Volume 11 Issue 1 Year 2026 | Page 160-170 ISSN: 2527-9866
Received: **18-12-2025** | *Revised:* **28-12-2025** | *Accepted:* **26-01-2025**

Classification of Herbal Plant Images Using Transfer Learning EfficientNetV2-B3

Rizki Ambarwati[1], Siska Devella[2]
[1,2] Multi Data Palembang University, Palembang, South Sumatra, Indonesia, 30113
e-mail: rizkiambarwati_2226250079@mhs.mdp.ac.id[1], siskadevella@mdp.ac.id[2]
*Correspondence: rizkiambarwati_2226250079@mhs.mdp.ac.id

Abstract: *Herbal plants are natural resources that have high economic and health value, but the identification process is still done manually, making it prone to errors due to morphological similarities between species. This study aims to develop a leaf image classification model for herbal plants using a Convolutional Neural Network (CNN) with the EfficientNetV2-B3 transfer learning approach and AdamW optimizer. The dataset used is the Indonesian Herb Leaf Dataset 3500, which consists of 3,500 leaf images from 10 types of Indonesian herbal plants, namely Belimbing Wuluh, Jambu Biji, Jeruk Nipis, Kemangi, Lidah Buaya, Nangka, Pandan, Pepaya, Seledri, and Sirih. The research stages included preprocessing, dataset division, and augmentation such as flipping, rotation, zooming, contrast and brightness changes, translation, and the addition of Gaussian noise and Salt-and-Pepper noise to increase data variation and test model robustness. Evaluation based on accuracy, precision, recall, and f1-score shows that the model without augmentation achieved 98.57% accuracy, 98.63% precision, 98.57% recall, and 98.58% f1-score, while the model with augmentation and noise addition achieved an accuracy of 97.71%, precision of 97.83%, recall of 97.71%, and an f1-score of 97.72%. These results prove that EfficientNetV2-B3 is capable of effectively classifying herbal plant leaves with good performance.*

Keywords: *EfficientNetV2-B3, Classification, Herbal Plants.*

1. Introduction

Indonesia is known as one of the countries with high biodiversity. This diversity includes various types of herbal plants that grow in various regions and have long been used by the community as traditional medicines to maintain health. The use of herbal plants as traditional medicines is considered safer than the use of modern medicines because they have fewer side effects [1]. The part of the herbal plant that is often used is the leaves, because they contain active compounds that are useful as ingredients for medicinal herbs [2]. This utilization makes herbal plants not only important from a health perspective, but also has economic value and the potential to be developed more widely in the pharmaceutical industry [3].

However, despite their significant benefits, the classification process for herbal plants still faces various challenges. One of them is that the leaves of several plant species have similar shapes, sizes, and textures, making them difficult to distinguish visually [4]. This situation causes non-experts who do not have botanical knowledge to experience difficulties in accurately identifying herbal plant species, given that conventional plant identification still relies on manual observation of leaf morphological characteristics by botanists [5]. However, errors in identifying herbal plants can pose risks to the effectiveness of treatment, method of use, timing of consumption, and suitability of the medicine for the type of disease [6]. Therefore, an approach that can classify herbal plants systematically and accurately is needed.

One approach that can solve this problem is computer vision, a technology that enables computers to recognize and analyze objects from digital images [7]. This technology is developing rapidly with the emergence of deep learning, a branch of machine learning that uses algorithms that mimic the working mechanisms of the human brain, known as artificial neural networks [8]. One of the algorithms used in deep learning is the Convolutional Neural Network (CNN), which is designed to process two-dimensional data such as images. The CNN method has a strong ability to automatically learn important features through the convolution and pooling processes, which enable more accurate and efficient object recognition [9]. This advantage makes CNN effective in distinguishing highly similar visual patterns, such as the leaf shapes of different species of herbal plants, which are difficult for humans to identify manually.

The superiority of CNN in recognizing complex visual patterns has been proven through various studies. One of them is a study that has proven the effectiveness of CNN in classifying herbal leaves using a dataset of 21,450 leaf images from 33 classes of herbal plants, achieving the highest accuracy of 97%, which states that CNN is effective in the process of classifying herbal plants [10]. Another study using a dataset of 480 herbal leaf images also produced an accuracy, precision, recall, and F1-score of 98%, higher than the Naïve Bayes method, which obtained an accuracy of 90% [11]. Research [12] using the Indonesian Herb Leaf Dataset 3500 with 10 classes of herbal plants and a total of 3,500 images achieved an accuracy of 97% with CNN and VGG16 architecture. Another study comparing EfficientNetB7 and MobileNetV2 for herbal plant classification using 10 classes of herbal plants in Indonesia achieved an accuracy of 98%, with EfficientNetB7 slightly superior in several metrics [1].

These results prove the great potential of CNN in recognizing types of herbal plants. CNN has achieved significant results in the field of classification [13]. However, most of these studies still use conventional CNN architecture, which requires long training times and is not yet optimal in handling morphological similarities between leaf species. This condition highlights the need for a more efficient and faster model that still maintains a high level of accuracy. The performance of CNN models for classification is highly dependent on the choice of architecture used [14]. One CNN architecture designed to address this need is EfficientNetV2-B3. This model was developed using a compound scaling approach, which is capable of balancing model size, training speed, and accuracy simultaneously, where the EfficientNetV2 variant uses Fused-MBConv and MBConv blocks that have been proven to accelerate the training process up to 11 times faster and make the model size 6.8 times smaller than the previous EfficientNet model without reducing accuracy [15]. The superiority of this architecture is also proven by research [16] using a dataset with a total of 3,076 images measuring 1500 x 1500 with 6 disease classes and 1 healthy leaf class. In this study, the EfficientNetV2-B3 model achieved an accuracy of 79.55% in potato leaf disease classification, higher than the Vision Transformer (ViT), which only achieved an accuracy of 77.92%. This indicates that EfficientNetV2-B3 is capable of learning complex visual features and handling variations in leaf shape and texture, as reported in previous studies.

In addition to architecture selection, the success of model training also depends on the selection of the right optimizer to improve accuracy [17]. This study uses the AdamW (Adaptive Moment Estimation with Weight Decay) optimizer, which is an extension of the Adam optimizer that applies a weight decay mechanism separately from the parameter update process. This approach aims to overcome Adam's limitations in weight regularization management, thereby improving training stability and model generalization performance [18]. The combination of the EfficientNetV2-B3 architecture and the AdamW optimizer is expected to produce a more stable, efficient, and accurate model for classifying herbal plant leaf images, especially in datasets with variations in shape and texture.

Based on previous research, most herbal leaf classification studies focus on improving classification accuracy using CNN architectures such as VGG16, MobileNetV2, and EfficientNetB7, while empirical analyses of training efficiency and model robustness are still limited. Therefore, this study does not propose a new architecture, but rather presents an empirical evaluation of the performance of EfficientNetV2-B3 with the AdamW optimizer at different levels of data complexity, and provides practical insights into the impact of augmentation and noise on classification performance.

2. Methods

This research implemented a method for classifying herbal plants, consisting of dataset collection, data preprocessing, dataset division, image augmentation and noise addition, EfficientNetV2-B3 model training, and model evaluation. The architecture of EfficientNetV2-B3 used in this study can be seen in Figure 1.

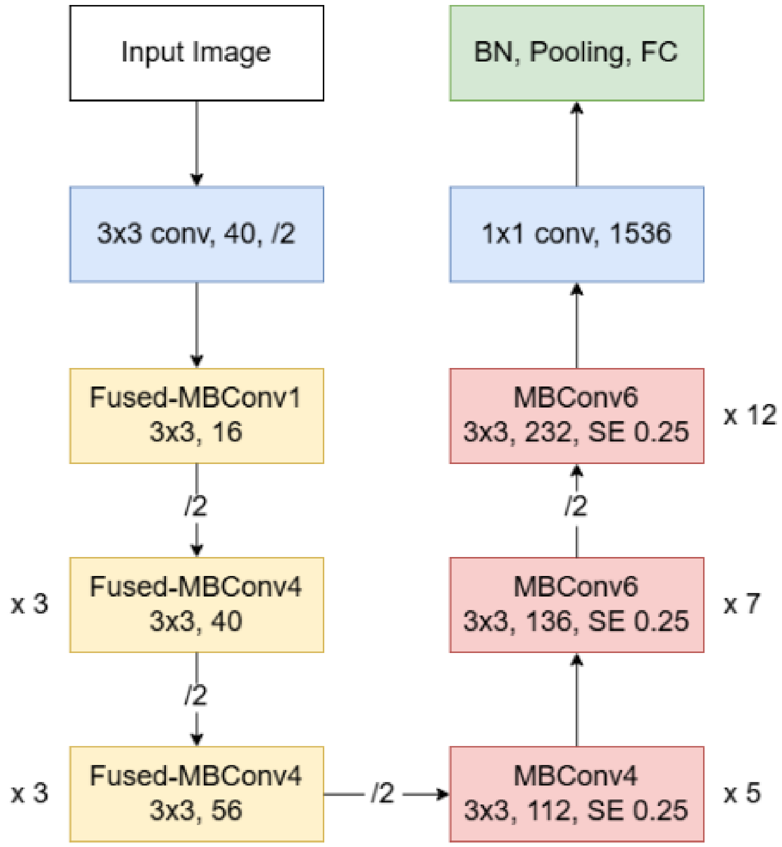

Figure 1. EfficientNetV2-B3 Architecture
[Source: [19]]

This study compares the results by applying two design schemes for testing. The first scheme represents the reference training condition, where the model is trained using the original dataset without applying data augmentation or noise addition. The scheme without augmentation is shown in Figure 2.

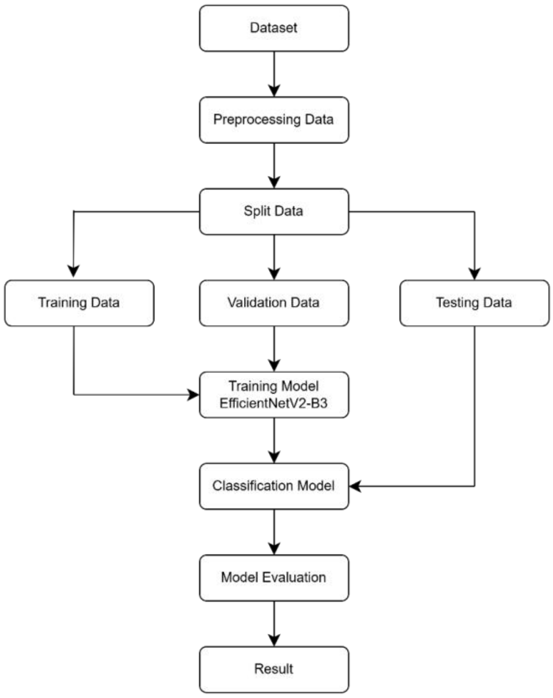

Figure 2. Design Scheme Without Augmentation

The second scheme uses data augmentation and noise addition. The results of the two designs are then compared to assess the effect of augmentation and noise addition on the accuracy and performance of the EfficientNetV2-B3 model in classifying herbal plants. The design scheme using data augmentation and noise addition is shown in Figure 3.

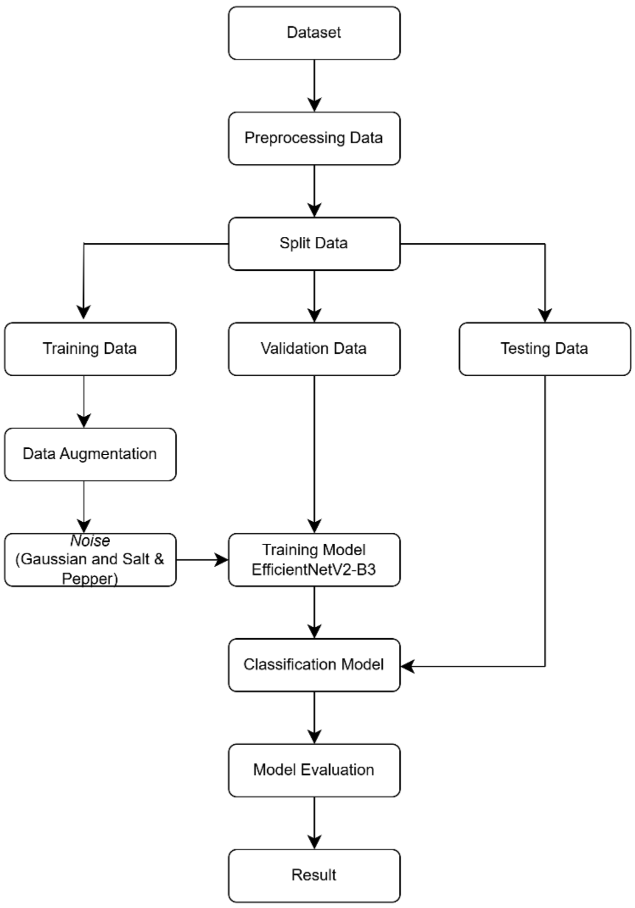

Figure 3. Augmentation and Noise Design Scheme

The stages of the research conducted are described as follows:

A. Dataset Collection

The dataset used in this study is the Indonesian Herb Leaf Dataset 3500, which is publicly available from Mendeley Data [20]. This dataset can be accessed via the following link: https://data.mendeley.com/datasets/s82j8dh4rr/1. This dataset consists of 3500 images of 10 types of herbal plants, with each class containing 350 leaf images, including Belimbing Wuluh (Averrhoa bilimbi), Jambu Biji (Psidium guajava), Jeruk Nipis (Citrus aurantiifolia), Kemangi (Ocimum africanum), Lidah Buaya (Aloe vera), Nangka (Artocarpus heterophyllus), Pandan (Pandanus amaryllifolius), Pepaya (Carica papaya), Seledri (Apium graveolens), and Sirih (Piper betle). All images are stored in (.jpg) format with dimensions of 1600 × 1200 pixels and a white background. It has been collected and published by previous researchers and is open source, so it can be used for research purposes. The research dataset can be seen in Table 1.

Table 1. Indonesian Herb Leaf Dataset 3500

Class	Images	Number of Samples
Belimbing Wuluh		350
Jambu Biji		350
Jeruk Nipis		350
Kemangi		350
Lidah Buaya		350
Nangka		350
Pandan		350
Pepaya		350
Seledri		350
Sirih		350
	Total Data	3500

B. Data Preprocessing

Preprocessing is performed to standardize image size and ensure image quality before use in model training. Preprocessing includes:

1. Resizing images from 1600x1200 pixels to 300x300 pixels with 3 RGB channels, following the default input resolution of EfficientNetV2-B3.

2. Converting images to float32, with normalization performed using the EfficientNetV2-B3 preprocessing function to preserve key visual features.
3. Splitting the dataset into training, validation, and test sets with an 80:10:10 ratio, resulting in 2,800 training images, 350 validation images, and 350 test images from a total of 3,500 images. The training data is used to build the model, the validation data is used to monitor the model's performance during training, and the test data is used to assess the model's ability to recognize images it has never seen before.
4. The training data augmentation used was RandomFlip (horizontal–vertical), RandomRotation (0.2 radians), RandomZoom (20%), RandomContrast (30%), RandomBrightness (20%), and RandomTranslation (shifts the image position by 10%) to simulate variations in orientation, lighting, position, and scale.
5. Adding noise in the form of Gaussian noise with mean 0.0 and standard deviation 10.0, and salt-and-pepper noise with amount = 0.005, where 0.25% of pixels were set to white (salt) and 0.25% to black (pepper).

C. Architecture Modeling and Model Training

This study uses the pretrained ImageNet EfficientNetV2-B3 architecture for feature extraction, implemented using TensorFlow and Keras. The classification head consists of a Global Average Pooling layer, a Dropout layer with a rate of 0.5, and a Dense softmax output layer with the number of neurons equal to the number of classes. Training is performed using a fine-tuning. During epochs 1–9, all EfficientNetV2-B3 backbone layers were frozen to stabilize feature extraction. From epoch 10 onward, the last 100 layers were unfrozen to enable gradual fine-tuning of higher-level representations. The model is optimized using the AdamW optimizer with sparse categorical cross-entropy loss. The parameters used are:

1. Learning rate: 1×10^{-4}
2. Weight decay: 1×10^{-4}
3. Batch size: 32
4. Random seed: 123
5. Epochs: 50 (with EarlyStopping = 10)
6. Optimizer: AdamW
7. Callback: EarlyStopping and ReduceLROnPlateau
8. Model Evaluation: Model evaluation was strictly conducted using the original test dataset without any augmentation or noise to ensure fair assessment of generalization performance. Performance was evaluated using a confusion matrix, accuracy, precision, recall, and F1-score metrics.

Training was conducted using Google Colaboratory with T4 GPU acceleration. The experiments were prepared on a laptop with 16 GB RAM running a 64-bit Windows 11 operating system. The software environment used Python 3.x, supported by NumPy and Matplotlib.

3. Results and Discussion

This study evaluates the performance of the EfficientNetV2-B3 model under two scenarios: without data augmentation and with augmentation and noise, to analyze the effect of data variation on classification accuracy and generalization. Model performance is assessed using accuracy and loss curves, as well as confusion matrices. The results of the evaluation scenario without augmentation can be seen in Figure 4, which shows the accuracy graph, Figure 5 shows the loss graph, and Figure 6 shows the model performance evaluation using a confusion matrix.

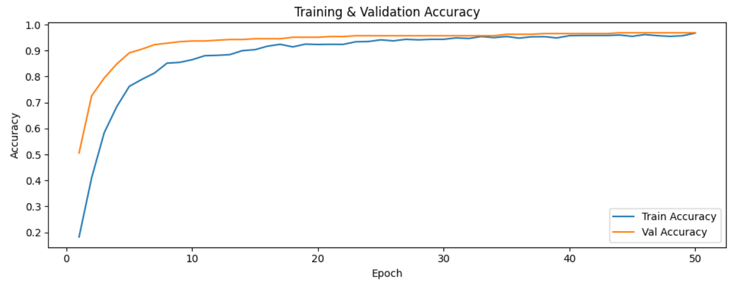

Figure 4. Accuracy Graph Without Augmentation and Noise

Figure 4 shows the training and validation accuracy over 50 epochs without data augmentation and noise. The accuracy increases rapidly during the early epochs (1–10), rising from approximately 0.20 to above 0.80, indicating fast learning of fundamental leaf features. Validation accuracy consistently remains slightly higher than training accuracy, suggesting the absence of overfitting. Performance improves steadily, reaching 94.02% at epoch 15, 96.86% at epoch 35, and achieving the highest accuracy of 98.57% at epoch 50, demonstrating stable learning and strong generalization without augmentation and noise.

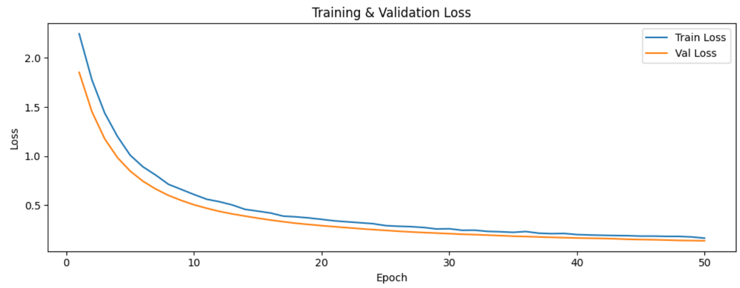

Figure 5. Loss Graph Without Augmentation and Noise

Figure 5 presents the training and validation loss curves for the scenario without data augmentation and Noise. The loss values decrease sharply during the early epochs from approximately 2.0–2.3, indicating rapid weight adjustment. After epoch 20, both training and validation losses continue to decline steadily and converge, reaching values below 0.2 by the end of training. The close alignment of the two curves indicates successful convergence and effective pattern learning using the original dataset without augmentation and noise.

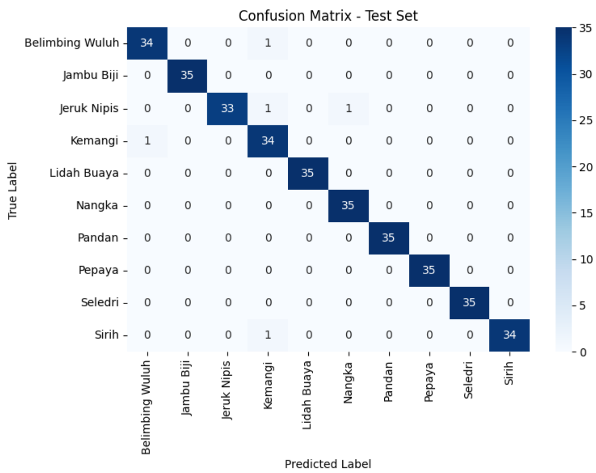

Figure 6. Confusion Matrix Without Augmentation and Noise

Figure 6 presents the confusion matrix for the test data without augmentation and noise. The diagonal dominance indicates very high classification accuracy across most herbal leaf classes, with 33–35 correct predictions per class. Several classes, including Belimbing Wuluh, Jambu

Biji, Lidah Buaya, Nangka, Pandan, Pepaya, Seledri, and Sirih, were classified perfectly or with negligible errors. Minor misclassifications occurred in the Jeruk Nipis and Kemangi classes (1–2 errors). The results of testing using augmentation and noise addition can be seen in Figures 7, 8, and 9 below:

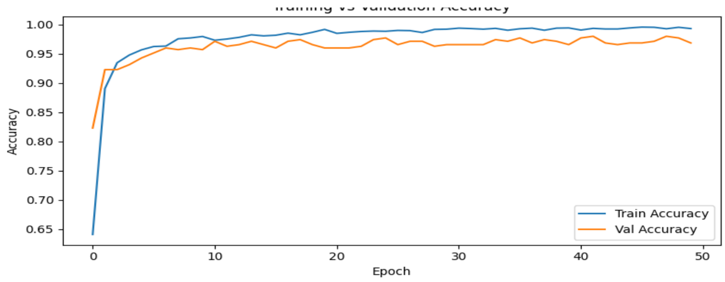

Figure 7. Augmentation and Noise Accuracy Graph

Figure 7 shows the training and validation accuracy over 50 epochs under the augmentation and noise scenario. Compared to the non-augmentation case, accuracy increases more gradually due to higher data complexity, reaching 76.86% at epoch 15, 88.57% at epoch 35, and 97.71% at epoch 50. This indicates that the model required more training time to learn from diverse visual variations, while maintaining stable learning behavior.

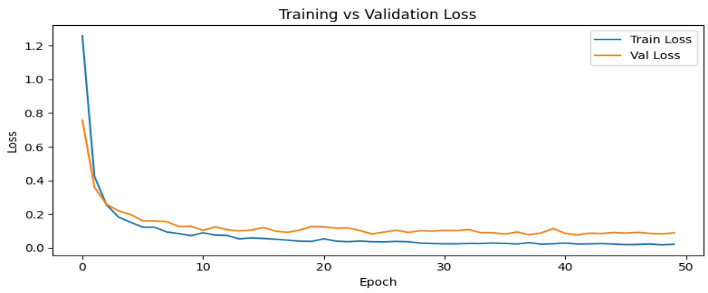

Figure 8. Augmentation and Noise Loss Graph

Figure 8 illustrates the training and validation loss curves for the same scenario. The initial loss values are higher than those without augmentation, reflecting increased data complexity. However, loss decreases consistently and converges between epochs 15 and 20, stabilizing below 0.2, which indicates effective optimization and successful learning under augmented and noisy training conditions.

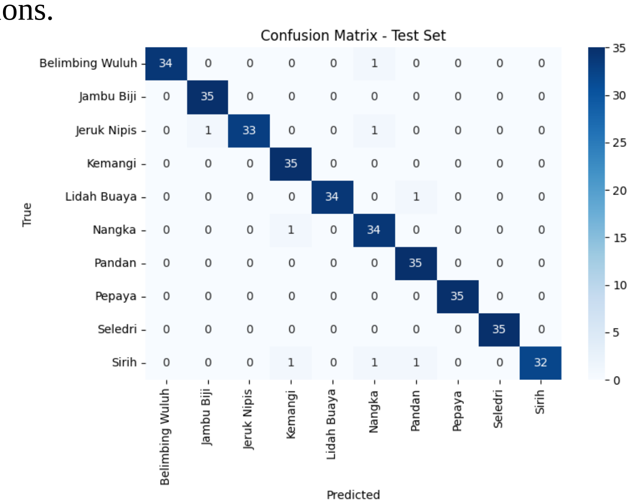

Figure 9. Confusion Matrix of the Model Trained with Augmentation and Noise

Figure 9 presents the confusion matrix obtained from the original test dataset using a model trained with data augmentation and noise. Most predictions are correct, with minor misclassifications observed in the Belimbing Wuluh, Jeruk Nipis, Lidah Buaya, Nangka, and

Sirih classes. These errors may be attributed to the increased visual variability introduced during training, which can slightly reduce the separability of certain morphological features.

A. Performance Evaluation and Comparative Analysis

Based on the results, the application of data augmentation and noise addition affects the performance of the EfficientNetV2-B3 model. Although these techniques increase feature diversity, they cause a slight decrease in overall performance compared to the non-augmented scenario. A comparison of the two scenarios is presented in Table 2 below:

Table 2. Comparison of Test Scenarios

Results	Without Augmentation and Noise	Augmentation and Noise
Architecture	EfficientNetV2-B3	EfficientNetV2-B3
Accuracy (%)	98.57	97.71
Precision (%)	98.63	97.83
Recall (%)	98.57	97.71
F1-Score (%)	98.58	97.72

Based on Table 2, the scenario without augmentation achieved the highest performance, with an accuracy of 98.57% and precision, recall, and F1-score above 98%, indicating high dataset consistency. In contrast, the augmentation and noise scenario showed a slight performance decrease to around 97.7% due to increased data complexity. These results provide an empirical observation of the EfficientNetV2-B3 model's behavior under different data complexity conditions. A comparison with previous studies using the Indonesian Herb Leaf Dataset 3500 is presented in Table 3.

Tabel 3. Performance Comparison of CNN Models

Study	Architecture	Optimizer	Augmentation	Accuracy (%)
[12]	VGG16	-	Yes	97
[12]	VGG16	-	No	96
[21]	EfficientNetV2-B0	Adam	No	97
[22]	MobileNet	Adam	Yes	98.86
This Study	EfficientNetV2-B3	AdamW	Yes	97.71
This Study	EfficientNetV2-B3	AdamW	No	98.57

Table 3 summarizes the best reported performance of CNN-based models on the Indonesian Herbal Leaf 3500 Dataset across various studies. This comparison is presented as a reference and not as a direct benchmark due to differences in experimental configurations. For studies evaluating multiple architectures, including MobileNet, VGG16, DenseNet121, ResNet50V2, and NASNetMobile, only the best-performing model is reported to ensure a concise comparison. MobileNet achieved the highest accuracy in a previous study [22], while the proposed EfficientNetV2-B3 with the AdamW optimizer showed comparable performance. A direct comparison with retraining the baseline model on an identical configuration has not been conducted and will be the subject of future research. The effects of AdamW, augmentation, and noise were further examined through controlled experiments using the same architecture, as shown in Table 4.

Tabel 4. Effect of Augmentation and Noise Using AdamW Optimizer

Augmentation	Noise	Optimizer	Accuracy (%)
No	No	AdamW	98.57
Yes	No	AdamW	98.29
Yes	Yes	AdamW	97.71

Table 4 shows that the AdamW optimizer provides consistently strong performance across different training configurations. The highest accuracy was achieved without augmentation and noise, while the introduction of augmentation and noise led to a gradual decrease in accuracy due to increased data complexity. Nevertheless, the results indicate that EfficientNetV2-B3 combined with AdamW remains stable and effective under varying data conditions.

4. Conclusions

This study evaluated the performance of the EfficientNetV2-B3 architecture combined with the AdamW optimizer for herbal leaf image classification under different data complexity conditions. Two experimental scenarios were analyzed, namely training without data augmentation and training with data augmentation and noise addition. The results show that the model achieved the highest accuracy of 98.57% without augmentation, indicating that the original dataset provides sufficient visual consistency for effective feature learning. When augmentation and noise were applied, accuracy slightly decreased to 97.71%, reflecting increased learning complexity while maintaining stable and high overall performance. The results indicate that data augmentation and noise addition increase task complexity while enabling the model to learn more diverse visual representations. However, since the dataset was collected under controlled conditions with uniform backgrounds and consistent lighting, the reported performance should be interpreted within this scope, and generalization to unconstrained environments remains limited.

This study does not aim to propose a new architecture or to claim superiority over other models. Instead, it provides an empirical analysis of the behavior of the EfficientNetV2-B3 model with the AdamW optimizer under varying data complexity conditions. A direct performance comparison with other deep learning architectures through retraining under identical experimental settings was not conducted and remains an important direction for future work. Future studies may include comparative evaluations with alternative CNN and transformer-based architectures, the use of datasets with more complex backgrounds, and the application of more comprehensive validation strategies such as k-fold cross-validation to further strengthen model generalization and reliability.

References

[1] S. Arnandito and T. B. Sasongko, “Comparison of EfficientNetB7 and MobileNetV2 in herbal plant species classification using convolutional neural networks,” *J. Appl. Informatics Comput.*, vol. 8, no. 1, pp. 176–185, 2024, doi: 10.30871/jaic.v8i1.7927.

[2] M. Faozi and Supatman, “Klasifikasi citra daun tanaman herbal untuk penyakit kanker-diabetes-hipertensi menggunakan model Xception,” *J. Sains Inform. Terap.*, vol. 4, no. 4, pp. 101–106, 2025, doi: 10.62357/jsit.v4i2.627.

[3] M. Andriani *et al.*, “Pemanfaatan tanaman obat keluarga jahe (Zingiber officinale) sebagai pengganti obat kimia di Dusun Tanjung Ale Desa Kemengking Dalam Kecamatan Taman Rajo Provinsi Jambi,” *Martabe J. Pengabdi. Kpd. Masy.*, vol. 4, no. 1, pp. 14–19, 2021, doi: 10.31604/jpm.v4i1.14-19.

[4] N. Kasim, M. B. Fadilah, W. Al Hidayat, and R. A. Saputra, “Klasifikasi jenis tanaman herbal berdasarkan citra menggunakan metode convolution neural network (CNN),” *J. Tekno Kompak*, vol. 19, no. 1, pp. 64–78, 2024, doi: 10.33365/jtk.v19i1.4536.

[5] I. P. Arisanti and Y. Yamasari, “Mengenali Jenis Tanaman Obat Berbasis Pola Citra Daun Dengan Algoritma K-Nearest Neighbors,” vol. 03, no. 02, pp. 95–103, 2021, doi: 10.26740/jinacs.v3n02.p95-103.

[6] B. Setiyono *et al.*, “Identifikasi tanaman obat Indonesia melalui citra daun menggunakan metode convolutional neural network (CNN),” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 2, pp. 385–392, 2023, doi: 10.25126/jtiik.20236809.

[7] I. Sanjaya, T. Lelita, and I. Yustiana, “Application of vision transformer for identifying Indonesian herbal plants based on visual images,” *J. Media Comput. Sci.*, vol. 4, no. 2, pp. 385–402, 2025, doi: 10.37676/jmcs.v4i2.8896.

[8] Ratnawati, F., & Tedyyana, A. (2020, October). Warning System Design to Detect Suspicious

Activities in a Network. In *2020 International Conference on Applied Science and Technology (iCAST)* (pp. 59-62). IEEE.
[9] S. A. E. Albakia and R. A. Saputra, "Identifikasi jenis daun tanaman obat menggunakan metode convolutional neural network (CNN) dengan model vgg16," *J. Inform. Polinema*, vol. 9, no. 4, pp. 451–460, 2023, doi: 10.33795/jip.v9i4.1420.
[10] R. Pujiati and N. Rochmawati, "Identifikasi citra daun tanaman herbal menggunakan metode convolutional neural network (CNN)," *J. Informatics Comput. Sci.*, vol. 3, no. 03, pp. 351–357, 2022, doi: 10.26740/jinacs.v3n03.p351-357.
[11] A. R. Azzahra, Purnawansyah, H. Darwis, and D. Widyawati, "Klasifikasi daun herbal menggunakan metode CNN dan Naïve Bayes dengan fitur GLCM," *Indones. J. Comput. Sci.*, vol. 12, no. 4, pp. 2183–2194, 2023, doi: 10.33022/ijcs.v12i4.3362.
[12] B. D. Mardiana, W. B. Utomo, U. N. Oktaviana, G. W. Wicaksono, and A. E. Minarno, "Herbal leaves classification based on leaf image using CNN architecture model VGG16," *J. RESTI*, vol. 7, no. 1, pp. 20–26, 2023, doi: 10.29207/resti.v7i1.4550.
[13] T. Sulistyorini, E. Sova, N. Sofie, and R. I. Napitupulu, "Penerapan hyperparameter convolutional neural network (CNN) dalam membangun model segmentasi gambar menggunakan arsitektur U-Net dengan TensorFlow," *J. Ilm. Inform. Komput.*, vol. 28, no. 2, pp. 112–121, 2023, doi: 10.35760/ik.2023.v28i2.6959.
[14] M. Nasihin and S. Supatman, "Klasifikasi citra produk font menggunakan convolution neural network," vol. 7, no. 02, pp. 658–670, 2025, doi: 10.53863/kst.v7i02.1293.
[15] M. Tan and Q. V. Le, "EfficientNetV2: Smaller models and faster training," *Proc. Mach. Learn. Res.*, 2021.
[16] J. H. Sinamenye, A. Chatterjee, and R. Shrestha, "Potato plant disease detection: leveraging hybrid deep learning models," *BMC Plant Biol.*, vol. 25, no. 647, pp. 1–15, 2025, doi: 10.1186/s12870-025-06679-4.
[17] Ramdaniah and B. A. Ashad, "Evaluasi perbandingan convolutional neural network dan deep learning optimizer untuk klasifikasi penyakit pada tanaman mangga," *Pros. SNEKTI (Seminar Nas. Energi, Kelistrikan, Tek. dan Inform.*, pp. 20–26, 2024.
[18] N Tedyyana, A., Ratnawati, F., & Kurniati, R. (2019). Rancangan sistem informasi penelitian dan pengabdian masyarakat Politeknik Negeri Bengkalis menggunakan metode UML (Unified Modeling Language). *Sistemasi: Jurnal Sistem Informasi*, *8*(3), 413-423. [19] N. H. Shabrina, S. Indarti, R. A. Lika, and R. Maharani, "A comparative analysis of convolutional neural networks approaches for phytoparasitic nematode identification," *Commun. Math. Biol. Neurosci.*, pp. 1–27, 2023, doi: 10.28919/cmbn/7993.
[20] A. E. Minarno, G. W. Wicaksono, Y. Azhar, and M. Y. Hasanuddin, "Indonesian herb leaf dataset 3500 [Dataset]," *Mendeley Data*, 2022. doi: 10.17632/s82j8dh4rr.1.
[21] R. P. S. Putra, C. S. K. Aditya, and G. W. Wicaksono, "Herbal leaf classification using deep learning model EfficientNetV2B0," *JITK (Jurnal Ilmu Pengetah. dan Teknol. Komputer)*, vol. 9, no. 2, pp. 301–307, 2024, doi: 10.33480/jitk.v9i2.5119.
[22] S. Salsabila, A. Suharso, and P. Purwantoro, "Comparison of deep learning architectures in identifying types of medicinal plant leaf images," *J. Appl. Informatics Comput.*, vol. 8, no. 1, pp. 39–46, 2024, doi: 10.30871/jaic.v8i1.6289.

.