Jurnal CoreIT, Vol.11, No.2, December 2025
ISSN 2460-738X (Print)
ISSN 2599-3321 (Online)

Classification of Herbal Leaves using EfficientNetB0

A. Nurul Aisya Alda[1], Dolly Indra[1*], Fitriyani Umar[1]
[1]Faculty of Computer Science, Universitas Muslim Indonesia, Indonesia
andialda6@gmail.com, dolly.indra@umi.ac.id*, fitriyani.umar@umi.ac.id

Abstract. The identification of herbal leaves remains a challenging task due to the high morphological and visual similarity among commonly used species, which often leads to misclassification when performed manually. This study addresses the challenge of identifying herbal leaves, namely *Sauropus androgynus*, *Moringa oleifera*, *Orthosiphon aristatus*, *Syzygium polyanthum*, and *Piper betle*, which are often difficult to distinguish due to high morphological and visual similarity. The proposed approach utilizes the EfficientNetB0 Convolutional Neural Network architecture and employs a two-stage fine-tuning strategy, combined with data augmentation, to enhance generalization performance. A total of 500 manually collected leaf images were used for training, resized to 224×224 pixels, and augmented through rotation and flipping. Model optimization was performed using the Adam and SGD optimizers. The trained model was evaluated on 235 previously unseen external images to assess robustness. The experimental results demonstrate that the proposed model achieved an overall classification accuracy of 88.94%, with particularly strong performance on leaf classes exhibiting distinctive morphological features, such as *Orthosiphon aristatus*, which obtained an F1-score of 0.96. However, the model exhibited limitations in distinguishing visually similar classes, especially between *Moringa oleifera* and *Sauropus androgynus*, both of which possess compound leaf structures, and performance degradation was observed under varying illumination conditions and complex backgrounds. The novelty of this study lies in the application of an EfficientNetB0-based fine-tuning strategy for multi-class herbal leaf classification using a limited, manually collected dataset, demonstrating its potential for deployment in mobile or other resource-constrained environments to support fast and reliable herbal plant identification.

Keywords: Adam, Convolutional Neural Network, EfficientNetB0, Fine-Tuning, Herbal Leaves, Image Classification.

Received August 2025/ **Revised** November 2025 / **Accepted** December 2025

INTRODUCTION

Indonesia is rich in biodiversity, including various herbal leaves that have long been recognized by local communities [1]. These herbal leaves are widely recognized for their use in traditional medicine, a practice that has been passed down through generations [2]. Their popularity continues to increase because they are considered healthier than chemical-based medicines [3]. This consumer preference for traditional remedies is fueled by the easy availability of raw ingredients and the perceived natural efficacy of plant-based self-medication. However, many people still do not fully understand the specific parts of these herbal plants that come from their leaves and their associated pharmacological benefits [3].

The diversity of herbal leaves can make accurate identification challenging for those without specialized knowledge [4]. Their highly varied shapes, patterns, and textures make herbal leaves such as bay leaves, curry, basil, cat's whisker, moringa, katuk, coriander, celery, binahong, and betel difficult to distinguish [5]. As a result, it is necessary to develop a system that can identify various types of herbal leaves by extracting features from images and analyzing parameters such as color, texture, and shape. This system must also be capable of performing classification to measure the level of accuracy in identifying the types of herbal leaves [6].

Automatic herbal plant recognition is gaining popularity with the increasing use of *deep learning* technology [1], [7]. *Deep learning* is a branch of machine learning that uses algorithms that mimic the structure and function of the brain through artificial neural networks [8]. One of the deep learning methods often used for image classification is *Convolutional Neural Network* (CNN) [9]. CNN is specifically designed to process two-dimensional data, such as images, with the ability to receive image input and

DOI: 10.24014/coreit.v11i2.38017

automatically recognize and identify objects or features within the image [10]. Through a learning process, CNN enables machines to understand complex patterns and distinguish between different types of images with high accuracy [11].

Several methods can be used to recognize and classify images, one of which is the CNN method with the EfficientNetB0 architecture [12]. The advantage of EfficiencyNetB0 is its ability to strike a balance between efficiency, accuracy, and flexibility. As in previous research related to "classification using the CNN method with the EfficiencyNetB0 architecture on corn and corn leaf diseases," the classification results showed a good level of accuracy, reaching 96% [13]. A study titled "Classification of Four Types of Herbal Leaves Using the Convolutional Neural Network Method" showed that the VGG16 model with transfer learning, data augmentation, and dropout techniques successfully achieved a validation accuracy of 96.2% in classifying four types of herbal leaves, namely curry leaves, mint leaves, betel leaves, and moringa leaves. Meanwhile, the CNN model without additional parameters produced a validation accuracy of 72%, indicating that VGG16 provides a significant performance improvement over the basic CNN [3].

Research by [14] employed the Convolutional Neural Network method with the proposed model, EfficientNet-B0, and two types of optimization algorithms, namely Adam and RMSprop. The accuracy obtained with both optimizations reached 91% [14]. Meanwhile, study [15] applied the EfficientNet-B6 deep learning model to classify images of rice leaf diseases into four classes. The analysis results show that the best performance was achieved with an input size of 224 and 50 epochs, resulting in the highest accuracy of 77.05%, precision of 77.11%, recall of 77.05%, and an F1 score of 76.29% on the fifth fold [15]. From the study [16] using Support Vector Machine and Naive Bayes in classifying herbal leaves. SVM with a linear kernel achieved the highest accuracy of 98% in dark conditions, while the sigmoid kernel produced the lowest accuracy of 44% in both dark and light conditions. The Naive Bayes method with a Multinomial kernel achieved the highest accuracy of 83% in bright conditions, and the Bernoulli kernel produced the lowest accuracy of 46% in both lighting conditions [16]. Then, research from [17] implemented a Convolutional Neural Network (CNN) method for classifying herbal plants based on leaf images. Thanks to its ability to classify a small number of parameters, this model is lightweight and suitable for devices with limited resources, yet still capable of providing high-accuracy results in tasks such as image classification, including the classification of herbal leaves [17].

Although many studies have utilized CNNs, the challenge of distinguishing between herbal leaves with high morphological similarity remains. This study focuses on five types of herbal leaves, namely *Sauropus androgynus* (katuk), *Moringa oleifera* (moringa), *Orthosiphon aristatus* (cat's whiskers), *Syzygium polyanthum* (Indonesian bay leaf), *and Piper betle* (betel). The objective of this research is to develop and evaluate an automatic classification system to distinguish five types of herbal leaves (katuk, kelor, kumis kucing, salam, and sirih) using the EfficientNetB0 Convolutional Neural Network (CNN) architecture with a two-stage fine-tuning strategy to achieve a high and objective level of accuracy.

METHODS

This study employs a structured methodological framework for the automatic identification of herbal leaves using image-based techniques. Details are provided in the following sections.

1. **Observation of Herbal Leaves:**
   At this stage, direct observation is conducted on the selected herbal leaves for research purposes. The aim is to recognize and understand the physical characteristics of each type of leaf. This observation includes details such as the original shape of the leaf, the pattern of the leaf veins, the surface texture, and any differences in color.
2. **Problem Analysis:**
   It can automatically recognize and distinguish types of herbal leaves, specifically *Syzygium polyanthum* (bay leaf), *Orthosiphon aristatus* (cat's whiskers), *Moringa oleifera* (moringa), *Sauropus androgynus* (katuk), and *Piper betle* (betel).
3. **Literature Study:**
   In this literature study stage, a review of previous research relevant to leaf image classification was conducted. This review encompassed various approaches, ranging from different deep learning architectures to conventional machine learning methods, in order to understand the existing solution landscape.
4. **Data Collection:**
   Data collection was carried out through two main scenarios:

a. Individual Leaf Photography: Each leaf was photographed separately to capture detailed features. Images were taken on both sides of the leaf, i.e., the front and back, to ensure that the model could learn the characteristics of the texture and veins of the leaf, which may differ on each side.

b. Contextual Photography: Leaves are also photographed in their natural state, i.e., while still attached to the stem or tree. This scenario aims to capture variations in lighting, shadows, and complex backgrounds that may be encountered when using the system in the field. Some examples of data acquisition results can be observed in Figures 1 to 5 below:

Figure 1. Example of *Sauropus androgynus*: single leaf (left, center) and natural condition (right)

Figure 2. Example of *Moringa oleifera*: single leaves (left, center) and natural conditions (right)

Figure 3. Example of *Orthosiphon aristatus*: single leaf (left, center) and natural condition (right)

Figure 4. Example of *Syzygium polyanthum*: single leaf (left, center) and natural condition (right)

Figure 5. Example of betel and piper betle linn: single leaf (left, center) and natural condition (right)

5. **Pre-Processing**

After the data is collected, a pre-processing stage is carried out to standardize the image format and address issues that could reduce accuracy, particularly shape distortion. As a solution, this study applies a method that maintains the original aspect ratio. The process begins by proportionally resizing the image, where the longest side is adjusted to 224 pixels. Next, to achieve the final dimensions of 224x224 pixels, the remaining area of the image is padded with black. This approach ensures that the original shape and silhouette of the leaves are preserved, allowing the model to learn them effectively. The entire transformation process is visualized in Figure 6 below.

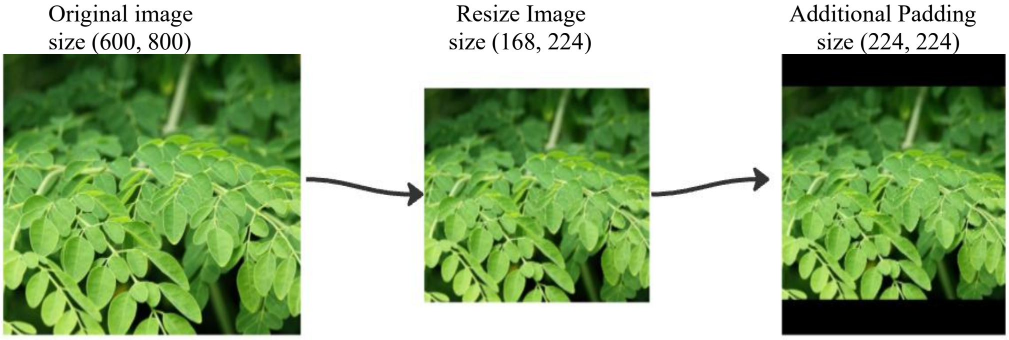

Figure 6. Resize and padding data

In addition to the resize and padding processes, the training data also goes through an augmentation stage that is applied on-the-fly. Data augmentation is a technique that artificially increases the diversity of a dataset, which is crucial for improving a model's generalization ability and preventing overfitting.

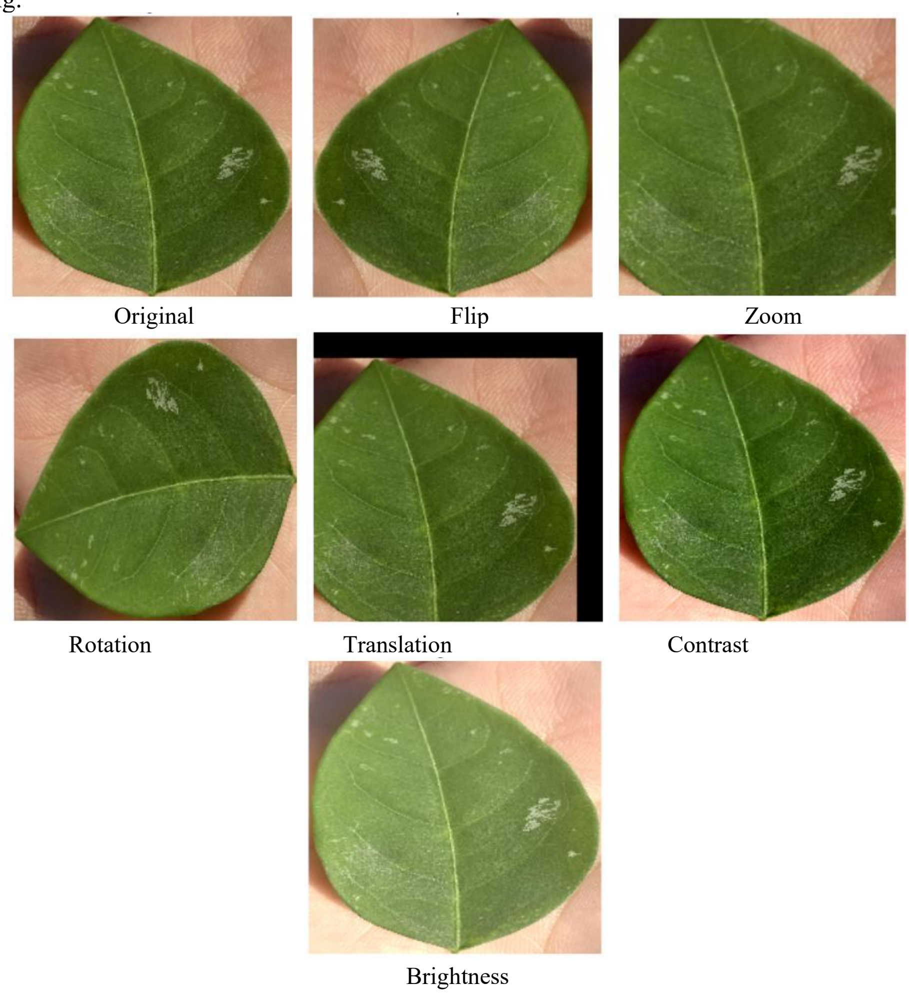

Figure 7. Data augmentation

Additionally, data labeling was conducted. Labeling in this study was applied using a directory-based organization method. All cleaned images were grouped into separate folders, where the name of each folder represented the relevant leaf class, namely *Sauropus androgynus* (katuk), *Moringa oleifera* (Moringa/ Kelor), *Orthosiphon aristatus* (cat’s whiskers/ Kumis Kucing), *Syzygium polyanthum* (Indonesian bay leaf/ Salam), *and Piper betle* (Sirih) as shown in Figure 8 below.

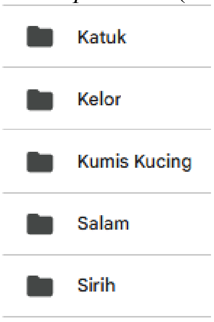

Figure 8. Dataset structure

The dataset is divided into a training set and a validation set before training. This division is done automatically using tf.keras.utils.image_dataset_from_directory function is shown in Figure 9.

```
Found 500 files belonging to 5 classes.
Using 400 files for training.
Found 500 files belonging to 5 classes.
Using 100 files for validation.
Ditemukan 5 kelas: ['Katuk', 'Kelor', 'Kumis Kucing', 'Salam', 'Sirih']
```

Figure 9. Dataset Division

By setting the parameter validation_split=0.2, this function proportionally separates 20% of the total images in each class directory to be used as validation data. The remaining 80% of the images are allocated as training data. Based on a total of 500 images, the resulting data allocation is 400 images for training and 100 images for validation, as shown in Figure 10. This process ensures that the model is evaluated using data it has never seen during the training phase, thereby providing a more objective assessment of its performance.

6. Architectural Model

The basic architecture of *EfficientNet*, specifically *EfficientNet-B0*, serves as a foundation that has been optimized through the Neural Architecture Search (NAS) process to achieve the optimal balance between efficiency and accuracy [11]. The main structure of EfficientNet-B0 is built from a series of fundamental blocks known as Mobile Inverted Bottleneck Convolution (MBConv) [18].

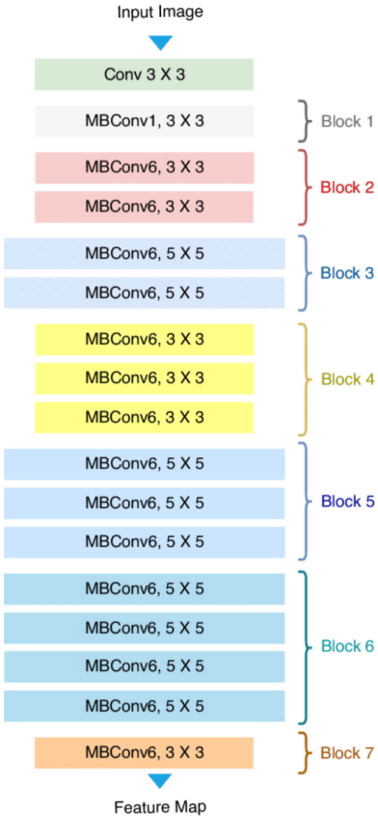

Figure 10. EfficientNetB0 Architecture [19]

The image depicts a highly efficient convolutional neural network (CNN) architecture, known as Mobile Inverted Bottleneck Convolution (MBConv), comprising seven sequential blocks. The network begins with an Input Image, which passes through an initial Conv 3x3 convolution layer to extract basic features. These features are then processed in depth by Blocks 1 through 7, which are a series of MBConv layers. This architecture alternates between 3x3 and 5x5 kernels to capture both small details and broader context. The processing blocks have varying depths; for example, Block 6 is the longest block (4 layers of MBConv6, 5x5), indicating the most intensive stage of feature extraction. The process concludes with Block 7 and the final 3x3 MBConv6 layer, which generates a Feature Map, a concise representation of the image, ready for prediction tasks. This configuration is designed to achieve high performance with minimal parameters and computation.

7. Training Model

This research stage aims to measure the accuracy level of a model. During the training process, the model adjusts its weights and improves performance at each layer. This model is trained using training data with Adam and Stochastic Gradient Descent (SGD) parameters to obtain optimal performance.

8. Model Evaluation

In this stage, the model will be tested using separate test data to assess its performance after training. The evaluation is carried out by confusion matrix such as accuracy, precision, recall, and F1-score, to ensure that the model works well in classifying data. The following matrix equation is shown in Equation (1-4).

$$\text{Accuracy} = \frac{TP + TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FN} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$\text{F1 } Score = 2 \text{ X } \frac{Precision \times recal}{Precision + recall} \quad (4)$$

RESULTS AND DISCUSSION

This section presents the results of experiments classifying five types of herbal leaves using the Convolutional Neural Network (CNN) method with a modified EfficientNetB0 architecture. The dataset consists of 500 training images and 235 independent test images. The training process was carried out using a two-stage fine-tuning strategy. A two-stage adaptation strategy is applied to ensure optimal adaptation of the pre-trained model to specific tasks, such as herbal leaf classification.

The first stage involves Freezing Base Layers, where the weights of all main convolution layers of the EfficientNetB0 model are retained (frozen), and only the additional classification layers (fully connected layers) are trained. The goal of this stage is to establish initial connections between the features extracted by the base model and the new class labels in the herbal leaf dataset. Once initial performance is achieved, the process continues to the second stage, which is Unfreezing and Fine-Tuning All Layers. At this stage, all layers of the model are opened for training, but using a very small learning rate. The goal is to slightly adjust (fine-tune) the weights in the base layers of EfficientNetB0, allowing the model to internalize the highly specific and detailed features of the new dataset without the risk of forgetting the general knowledge it has learned previously (catastrophic forgetting).

1. **Model Training Results**

   The model training process for each optimizer scenario is run by applying the Early Stopping mechanism, which aims to automatically find the optimal number of epochs. The progress of the model learning process, until it is finally stopped by the Early Stopping callback, is monitored through the accuracy and loss metrics presented in Figure 11 below.

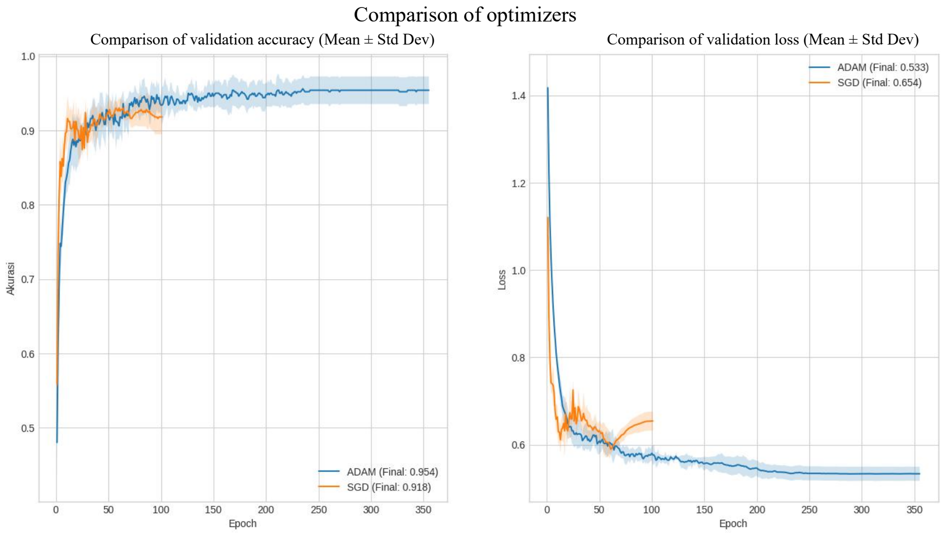

Figure 11. Comparison of Adam and SGD Performance Results

   The graph above shows a direct comparison between the performance of the Adam optimizer (blue) and SGD (orange). The thick line shows the average performance of five trials, while the shaded area around it (standard deviation) represents the level of consistency or variation in results between trials.

2. **Test Results**

   After the training process was completed and the best model (trained with the Adam optimizer) was obtained, a final evaluation was conducted to objectively test the model's generalization ability. This testing is not performed on the validation data used during training, but rather on a completely separate and new test set. This test set contains 235 images collected from the internet to simulate real-world conditions, as shown in Figure 12 below.

	dataset_test
Salam	50
Sirih	50
Kelor	42
Kumis Kucing	37
Katuk	56
Total	235

Figure 12. Testing Dataset Collection

The model performance results on this external test dataset are summarized in Table 1 and visualized in the confusion matrix in Figure 13.

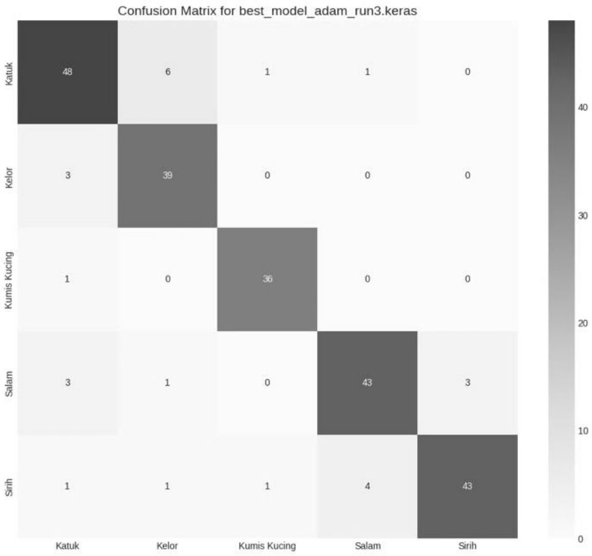

Figure 13. Confusion Matrix of Model Testing Results

From Figure 13, it can be seen that the values on the main diagonal, which is the row of cells stretching from the upper left corner to the lower right corner, contain the values 48, 39, 36, 43, and 43 in this example. These values indicate all data that has been correctly classified according to its original class. Meanwhile, the values outside the diagonal indicate the number of data points that were misclassified.

Table 1. Model Performance Evaluation Metrics

Leaf Class	Precision	Recall	F1-Score	Number of Samples
Sauropus Androgynus	0.86	0.86	0.86	56
Moringga Oleifera	0.83	0.93	0.88	42
Orthosiphon Aristatus	0.95	0.97	0.96	37
Syzygium polyanthum	0.90	0.86	0.88	50
Piper Betle	0.93	0.86	0.90	50
Average (Macro)	0.89	0.89	0.89	235
Average (Weighted)	0.89	0.89	0.89	235

Table 1 presents the detailed results of the model performance evaluation metrics on the test dataset. To provide a deeper understanding of the results in Table 1, the calculation details for the Precision, Recall, and F1-Score metrics for each class are presented below. These calculations are based on the True Positive (TP), False Positive (FP), and False Negative (FN) values extracted from the confusion matrix. Overall, the following formula is used to calculate model accuracy:

$$
\begin{aligned}
Accuracy &= \frac{Number\ of\ correct\ classifications\ per\ class\ (TP)}{Total\ number\ of\ classifications} \\
&= \frac{48 + 39 + 36 + 43 + 43}{235} \\
&= \frac{209}{235} \approx 88.94\%
\end{aligned}
$$

From the calculations obtained, the model achieved an accuracy rate of 88.94% on the testing dataset. The table also shows the macro average and weighted average. The macro average does not account for the weight of each class when calculating the average score, indicating how the model performs regardless of the number of samples per class. Meanwhile, the weighted average considers the number of samples when calculating the average, meaning that classes with more samples will have a greater influence on the final score.

CONCLUSION

This study developed a herbal leaf classification model based on the EfficientNetB0 architecture combined with fine-tuning and data augmentation strategies. The proposed model achieved strong classification performance, attaining an accuracy of 88.94% on the external test dataset, which consisted of previously unseen images. The Adam optimizer was employed to facilitate stable training and effective loss minimization during the learning process. The experimental results indicate that the model performs particularly well in identifying leaf classes with distinctive visual characteristics, such as the *Orthosiphon aristatus* (kumis kucing) class, which achieved the highest F1-score of 0.96, with a precision of 0.95 and a recall of 0.97. Nevertheless, the model exhibited limitations when distinguishing leaf classes with high morphological similarity, notably between *Sauropus androgynus* (katuk) and *Moringa oleifera* (kelor). In addition, performance degradation was observed under varying illumination conditions and complex background scenarios. Overall, these findings demonstrate that EfficientNetB0 is an effective approach for automated herbal leaf classification. However, improving discrimination among morphologically similar classes remains a key challenge and represents an important direction for future research.

REFERENCES

[1] Adrian, R. A. Syahputra, N. A. Juwita, R. Astyka, and M. F. Lubis, “Andaliman (Zanthoxylum acanthopodium DC.) a herbal medicine from North Sumatera, Indonesia: Phytochemical and pharmacological review,” *Heliyon*, vol. 9, no. 5, p. e16159, May 2023, doi: 10.1016/J.HELIYON.2023.E16159.

[2] M. Rahayu *et al.*, “Ethnobotany and diversity of Citrus spp. (Rutaceae) as a source of ‘Kem-kem’ traditional medicine used among the Karo sub-ethnic in North Sumatra, Indonesia,” *Heliyon*, vol. 10, no. 9, p. e29721, May 2024, doi: 10.1016/J.HELIYON.2024.E29721.

[3] M. H. Ahmad, M. Hana, T. Ghazi Pratama, and H. Aulida, “Klasifikasi Empat Jenis Daun Herbal Menggunakan Metode Convolutional Neural Network,” *Jurnal Ilmu Komputer dan Matemtika*, vol. 4, no. 2, pp. 69–76, 2023.

[4] R. Bashyal and D. L. Roberts, “Assessing the identification uncertainty in plant products traded as traditional Asian medicines,” *J Nat Conserv*, vol. 73, p. 126410, Jun. 2023, doi: 10.1016/J.JNC.2023.126410.

[5] N. L. Marpaung, R. J. H. Butar Butar, and S. Hutabarat, “Implementasi Deep learning untuk Identifikasi Daun Tanaman Obat Menggunakan Metode Transfer learning,” *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, vol. 9, no. 3, p. 348, 2023, doi: 10.26418/jp.v9i3.63895.

[6] M. Meiriyama, S. Devella, and S. M. Adelfi, “Klasifikasi Daun Herbal Berdasarkan Fitur Bentuk dan Tekstur Menggunakan KNN,” *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, vol. 9, no. 3, pp. 2573–2584, 2022, doi: 10.35957/jatisi.v9i3.2974.

[7] D. Irfansyah, M. Mustikasari, and A. Suroso, “Arsitektur Convolutional Neural Network (CNN) Alexnet Untuk Klasifikasi Hama Pada Citra Daun Tanaman Kopi,” *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 6, no. 2, pp. 87–92, 2021, doi: 10.30591/jpit.v6i2.2802.

[8] R. Pujiati and N. Rochmawati, “Identifikasi Citra Daun Tanaman Herbal Menggunakan Metode Convolutional Neural Network (CNN),” *Journal of Informatics and Computer Science (JINACS)*, vol. 3, no. 03, pp. 351–357, 2022, doi: 10.26740/jinacs.v3n03.p351-357.

[9] N. P. D. A. S. Dewi, M. W. A. Kesiman, I. M. G. Sunarya, G. A. A. D. Indradewi, and I. G. Andika, “Klasifikasi Jenis Daun Tumbuhan Herbal Berdasarkan Lontar Usada Taru Pramana Menggunakan CNN,” *Techno.Com*, vol. 23, no. 1, pp. 271–283, 2024, doi: 10.62411/tc.v23i1.9510.

[10] X. Zhao, L. Wang, Y. Zhang, X. Han, M. Deveci, and M. Parmar, “A review of convolutional neural networks in computer vision,” *Artif Intell Rev*, vol. 57, no. 4, Apr. 2024, doi: 10.1007/s10462-024-10721-6.

[11] R. Kusumastuti, T. Dwi Putra, and Z. Zulfahmi Yudam, “Klasifikasi Citra Penyakit Daun Jagung Menggunakan Algoritma Cnn Effcientnet,” *Multitek Indonesia*, vol. 17, no. 2, pp. 143–153, 2024, doi: 10.24269/mtkind.v17i2.10085.

[12] S. Arnandito and T. B. Sasongko, “Comparison of EfficientNetB7 and MobileNetV2 in Herbal Plant Species Classification Using Convolutional Neural Networks,” *Journal of Applied Informatics and Computing*, vol. 8, no. 1, pp. 176–185, 2024, doi: 10.30871/jaic.v8i1.7927.

[13] W. G. Pamungkas, M. I. P. Wardhana, Z. Sari, and Y. Azhar, “Leaf Image Identification: CNN with EfficientNet-B0 and ResNet-50 Used to Classified Corn Disease,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 2, pp. 326–333, 2023, doi: 10.29207/resti.v7i2.4736.

[14] S. A. Sabrina and W. F. Al Maki, “Klasifikasi Penyakit pada Tanaman Kopi Robusta Berdasarkan Citra Daun Menggunakan Convolutional Neural Network,” *eProceedings of Engineering*, vol. 9, no. 3, pp. 1919–1927, 2022.

[15] A. C. Milano, “Klasifikasi Penyakit Daun Padi Menggunakan Model Deep Learning Efficientnet-B6,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 1, 2024, doi: 10.23960/jitet.v12i1.3855.

[16] M. Samir, H. Darwis, F. Umar, and P. Korespondensi, “Fourier Descriptor Pada Klasifikasi Daun Herbal Menggunakan Support Vector Machine Dan Naive Bayes Fourier Descriptor on Classification of Herbal Leaves Using Support Vector Machine and Naive Bayes,” vol. 10, no. 6, pp. 1205–1212, 2023, doi: 10.25126/jtiik.2023107309.

[17] D. Intan Permatasari, “Implementasi Metode Convolutional Neural Network (Cnn) Untuk Klasifikasi Tanaman Herbal Berdasarkan Citra Daun,” *Kohesi: Jurnal Sains dan Teknologi*, vol. 3, no. 9, pp. 1–10, 2024.

[18] C. Mahaputri and I. D. Wisana, “Introduction Makanan Tradisional Nusantara dengan Menggunakan Metode Convolutional Neural Network (CNN),” *(Journal of Information System and Computer)*, vol. 1, no. 1, pp. 1–8, 2022.

[19] H. T. Gorji *et al.*, “Gambar Arsitektur EfficientNetB0,” Scientific Reports. [Online]. Available: https://r.search.yahoo.com/_ylt=Awr98kelUK9nKqMYM92jzbkF;_ylu=c2VjA2ZwLWF0dHJpY gRzbGsDcnVybA-- /RV=2/RE=1739571493/RO=11/RU=https%3A%2F%2Fwww.researchgate.net%2Ffigure%2FA-A-concise-representation-of-the-EfficientNet-B0-model-B-The-building-blocks-of_fig4_