MobileNetV3 for Durian Seedling Variety Classification Based on Leaf Images with Sobel Edge Detection Preprocessing

Rusly Nur Huda [1]*, Indah Susilawati [2]*
* Faculty of Information Technology, Informatics Study Program, Mercu Buana University Yogyakarta
ruslynurhuda1@gmail.com[1], Indah@mercubuana-yogya.ac.id [2]

Article Info

Article history:

Received 2026-05-25
Revised 2026-06-14
Accepted 2026-06-18

Keyword:

Convolutional Neural Network,
Deep Learning,
Durian Leaf Classification,
MobileNetV3Small,
Sobel Edge Detection,
Transfer Learning.

ABSTRACT

Durian is a high-value horticultural commodity in Indonesia, yet variety identification based on leaf morphology remains largely manual and prone to subjective error. This study develops a deep learning-based classification system for durian leaf varieties using MobileNetV3Small, designed for practical deployment on mobile and edge computing devices in agricultural field settings to support real-time variety identification by farmers and agricultural practitioners. A dataset of 1,680 images across four durian varieties Bawor, Musang King, Duri Hitam, and Super Tembaga was constructed through direct field collection and public dataset acquisition, followed by augmentation via rotation and flipping techniques. Preprocessing incorporated background removal, cropping, resizing to 224×224 pixels, and Sobel edge extraction to enhance morphological features such as leaf veins and contours. Transfer learning from ImageNet weights was applied, with training conducted in two phases: a 30-epoch warm-up with frozen base layers, followed by 90-epoch fine-tuning of the 20 deepest layers selected to adapt high-level semantic features while preserving general low-level representations and avoiding catastrophic forgetting using cosine annealing and early stopping. The model achieved a validation accuracy of 98.42% and a macro average F1-score of 0.9842, with only 4 misclassifications out of 253 test images. All misclassifications occurred exclusively between the morphologically similar Bawor and Duri Hitam classes. Comparative analysis against eight prior studies using similar durian leaf classification tasks suggests that the proposed approach achieves competitive performance relative to studies employing larger architectures such as ResNet and ConvNeXt, though direct comparison on an identical dataset was not conducted and remains a direction for future work.

I. Introduction

Durian is one of Indonesia's leading horticultural commodities with high economic value and continuously growing market demand [1]. Identification of durian leaf types and conditions is crucial in supporting cultivation and plant health monitoring, thereby improving production quality and reducing losses. However, the identification process, which is still carried out manually, risks errors due to the subjectivity of observation and limited experience. [2]. Therefore, a more objective and efficient solution is needed to facilitate the identification of durian leaf types and conditions.

Along with technological advancements, computer vision and deep learning-based approaches have been widely used in plant leaf image classification. This approach has been proven to improve accuracy and efficiency compared to conventional methods that rely more on manual analysis, thus providing a more optimal solution in plant variety identification. This technology enables automatic processing of large numbers of images, which can accelerate the identification process and analyze plant conditions in real-time [3].

Convolutional Neural Network (CNN) is one of the deep learning architectures that is effective in image processing [4]. CNN is designed to recognize patterns in images by extracting visual features such as edges and textures from images without requiring manual feature extraction [5]. With the ability to identify complex visual patterns, CNN can help recognize various features in leaf images, such as shape, size, and texture, thus facilitating more accurate and faster variety identification compared to conventional methods that rely more on human observation [6]. The use of CNN in durian leaf classification can produce higher and more consistent accuracy under various image conditions [7].

Several previous studies have developed durian leaf image classification methods using deep learning. Elroy [8] employed a standard CNN with normalization and augmentation preprocessing, achieving an accuracy of 80% with an F1-score of 0.82, however significant room for improvement remained. Kurniawan [9] demonstrated that optimizing training parameters such as learning rate and batch size in MobileNetV2 with background removal preprocessing could improve accuracy up to 90%. Kulsum [4]showed that ResNet50-based transfer learning with fine tuning from ImageNet weights yielded an accuracy of 91%, confirming the benefits of pre trained models on limited datasets.

Further research explored more complex architectures and more mature optimization strategies. Klangbunrueang [10] demonstrated that ConvNeXt with transfer learning achieved an accuracy of 98%, while MobileNetV3 remained competitive with an accuracy of 90.83% and faster inference time. Diana [11] proved that optimizing dropout parameters and layer architecture in Xception yielded an accuracy of 92%, while also highlighting the importance of handling dataset imbalance. Voo [12] underscored the significance of fine-tuning strategies, where accuracy increased dramatically from 8.22% to 94.64% after fine-tuning was applied. Eiamin [13] demonstrated that YOLOv5 excelled with an accuracy of 93.33% for real-time detection, whereas Wirabowo & Susilawati [14] showed that a simple CNN could achieve an accuracy of 90.03% with stable performance. Overall, this literature review confirms that the choice of architecture, fine-tuning strategy, and preprocessing technique are the primary determining factors in improving the accuracy and efficiency of durian leaf classification systems.

To further advance the development of that research, it can be explained that Convolutional Neural Networks (CNN) is a highly effective deep learning method that has been proven capable of improving performance in image recognition tasks [15]. However, there remains a question regarding the extent to which CNN can accurately recognize durian seedlings based on leaf images. One CNN architecture that can be used for this purpose is MobileNet-V3 [16]. Therefore, the use of CNN in this study enables a more efficient and accurate durian leaf classification process. Furthermore, in this study, MobileNetV3Small was selected as the model architecture due to its computational efficiency, which is suitable for resource-limited devices such as mobile phones or embedded devices [5]. The depthwise separable convolutions technique in MobileNetV3Small enables fast and efficient image processing while maintaining high accuracy [17].

In this study, transfer learning on MobileNetV3Small and Sobel edge detection were used to improve the accuracy of durian leaf classification [18]. The main focus of this study is to optimize the model through a two-stage fine-tuning process to improve classification accuracy compared to existing standard methods, in line with the findings of Voo [12]. The subjects of this study include four durian seedling varieties, namely Montong, Musang King, Bawor, and Duri Hitam, which will be tested to evaluate the effectiveness of the model in durian leaf classification.

II. METHOD

A. Research Scheme and Workflow

This study began with the collection of a durian leaf image dataset, which was then augmented and processed using the Sobel Edge Detection technique. The pre-processed data was divided into three sets: training, validation, and testing. Model training was performed using MobileNetV3Small through transfer learning, with warm-up and fine-tuning stages. Evaluation was carried out using accuracy and loss metrics. The research workflow can be seen in Figure 2.1, which illustrates the steps from data collection to model evaluation.

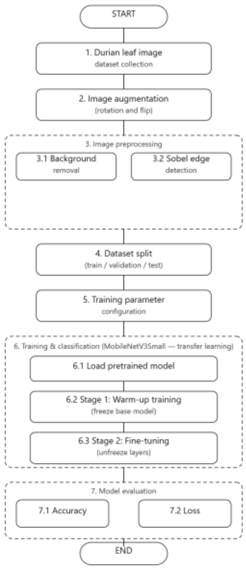


Figure 2. 1 Research Workflow

B. Data Collection

Data collection was carried out from two sources in equal proportions. The first source involved direct field collection at a durian orchard in Magelang using a smartphone camera, yielding 30 images per class. The second source was a public dataset from Kaggle, which also contributed 30 images per class. Consequently, the total initial dataset comprised 240 images spanning four durian leaf classes (Bawor, Musang King, Duri Hitam, and Super Tembaga), each containing 60 images with a composition of 50% field data and 50% public data.

Annotation quality validation was conducted in two stages. First, visual selection was performed manually to eliminate images that were blurry, unclear, or unrepresentative. Second, label consistency across classes was verified against variety morphology references from the literature and through direct confirmation with the durian orchard owner in Magelang for field data, while the label metadata available in the Kaggle public dataset was used as reference for external data. Field image capture was carried out with attention to data quality consistency, encompassing natural daylight illumination, a frontal shooting angle perpendicular to the leaf surface, and a focus on fully expanded mature leaves.

The collected data was then selected to ensure image quality, with blurry or unclear images being eliminated. To increase variation, the data was expanded using image augmentation techniques such as flipping, rotation, and the Random Nearest Rotation (RNR) method, resulting in a total dataset of 1,680 images after augmentation. This dataset was used for the preprocessing and model training stages. It should be noted that the collected dataset consists of leaf images in relatively good and clean condition, and therefore does not yet cover more diverse field conditions such as damaged, perforated, disease-affected, or partially occluded leaves.

C. Image Augmentation

Image augmentation is a technique to artificially increase the quantity and variety of a dataset without directly adding new data [9]. Its main purpose is to enrich image variation so that the model can learn from various conditions and achieve better generalization capability.

Augmentation was limited to rotation and flip, which were selected for their ability to preserve the morphological characteristics of leaves without altering color features that are also relevant to classification. Intensity-based augmentation techniques such as brightness adjustment, color jitter, and zooming were not applied to avoid distorting leaf color features that could affect the consistency of Sobel feature extraction. Nevertheless, exploring these additional augmentation techniques has the potential to improve model robustness against variations in field conditions and is recommended for future research.

In this study, image augmentation was performed prior to the preprocessing stage by applying rotation and flipping techniques, which were selected because they can preserve the main characteristics of leaf objects while adding variation in image shape and orientation.

1) Rotation

Rotation is a data augmentation technique that rotates an image at a certain angle to increase data variation [8]. Rotation shifts pixel positions based on the formula:

$$x' = x\cos\theta - y\sin\theta, \qquad (1)$$

$$y' = x\sin\theta + y\cos\theta$$

Description:
x, y: initial pixel coordinates
x′, y′: rotated pixel coordinates
θ: rotation angle (in degrees or radians)

However, rotation can cause empty areas at the edges of the image, as shown in Figure 2.2. To address this, image boundary handling techniques are used, namely Random Nearest Rotation (RNR), Random Reflect Rotation (RRR), and Random Wrap Rotation (RWR), which fill the empty areas with the nearest pixel values, mirror reflection, or repeating pixel patterns respectively. These techniques ensure that image information is preserved without reducing data quality.

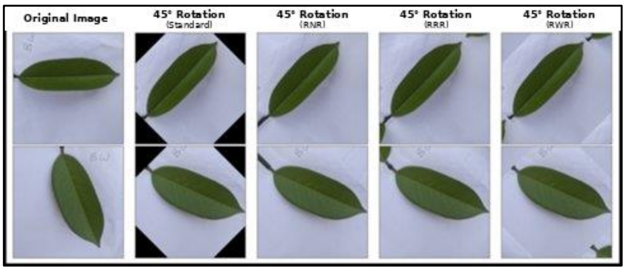


Figure 2. 2 Image Rotation

2) Flip

Flip (flipping) is a data augmentation technique based on geometric transformation that mirrors an image along a certain axis to increase data variation [1]. This technique includes horizontal flip, vertical flip, and a combination of both. Horizontal flip mirrors the image along the Y-axis, while vertical flip mirrors it along the X-axis. Mathematically, flipping can be represented by the following formula:

$$x' = W - 1 - x,\ y' = y;$$
$$x' = x,\ y' = H - 1 - y; \qquad (2)$$
$$x' = W - 1 - x,\ y' = H - 1 - y$$

Description:
x, y : initial pixel coordinates
x′, y′ : pixel coordinates after flip
W : image width
H : image height

The application of flipping, particularly horizontal flip, increases training data variation and helps the model learn from various object orientations. However, this technique is less suitable for asymmetric or direction-sensitive data, such as letters, numbers, or symbols, as it can cause changes in meaning or label inconsistencies [19]. A comparison of the results of all three flip types can be seen in Figure 2.3, which shows the change in orientation of durian leaf images from the original image to the results of horizontal flip, vertical flip, and a combination of both. This figure illustrates how the flip technique increases data variation for classification model training.

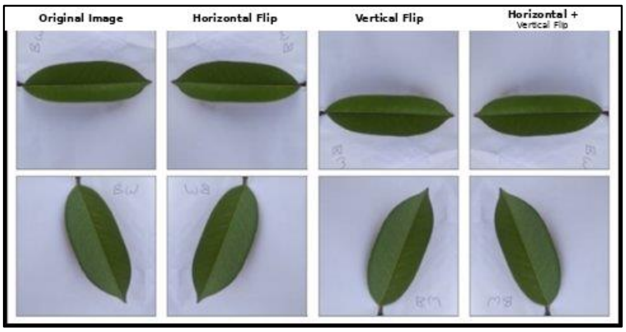


Figure 2. 3 Image Flip

D. Preprocessing

The image preprocessing stage aims to improve data quality and ensure compatibility with the training model [9]. This process begins with background removal to focus the image on the durian leaf, followed by cropping to highlight the leaf object. The image is then placed on a plain background with a larger size and cropped again into a square shape, following the bounding box dimensions. The image is finally resized to 224 × 224 pixels to match the requirements of the MobileNetV3Small model.

In addition, color space adjustment was also performed, considering the use of RGB as the primary color space for efficiency, although alternative color spaces can be used for more optimal feature representation [6]. This technique helps reduce interference from external elements, such as soil or branches, so that the model can focus more on relevant leaf shape features in the classification process.

This study also applies the Sobel Edge Detection method in the preprocessing stage to extract edge features from leaf images. The selection of Sobel Edge Detection over other edge extraction methods such as Canny and Laplacian was based on several considerations. First, Sobel produces smoother and more continuous edge representations compared to Canny, which tends to yield thin single-pixel binary edges, thereby preserving gradient information of leaf morphology such as vein width and contour edge thickness, making it more informative for the CNN feature learning process. Second, Sobel is more robust to noise than Laplacian, which relies on second-order derivatives and is more sensitive to small intensity variations, making it more stable when applied to field leaf images that exhibit surface texture variations. Third, Sobel computation is relatively lightweight and efficient, as it employs only two 3×3 convolution kernels, which aligns with this study's approach emphasizing computational efficiency for deployment on resource-constrained devices. Nevertheless, a direct comparison between Sobel, Canny, and Laplacian on the same dataset is acknowledged as a limitation of this study and is recommended for exploration in future research. The Sobel method uses horizontal (Gx) and vertical (Gy) convolution kernels to calculate changes in pixel intensity. The gradient results are then combined to obtain the gradient magnitude that describes the strength of edges in the image, which is formulated as:

$$G = \sqrt{G_x^2 + G_y^2} \quad (3)$$

Description:

G: gradient magnitude (edge strength)
Gx: result of image convolution with the horizontal kernel
Gy: result of image convolution with the vertical kernel

The application of Sobel aims to highlight important morphological structures such as leaf veins and leaf vein patterns, which are crucial in classification, particularly for distinguishing leaf varieties with similar colors but different shapes [20]. By integrating Sobel, images become clearer and more informative, supporting the feature learning process, improving classification accuracy, and strengthening the model's generalization capability against new data. This combination of background removal, cropping, resizing, and Sobel edge detection produces optimal input data for durian leaf classification model training.

TABLE 2. 1
PREPROCESSING DATA SAMPLE

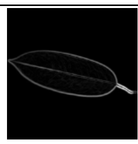
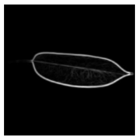
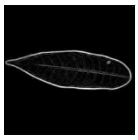
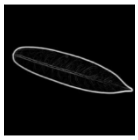
Class	Original Image	Background Removal	Sobel Edge
Bawor			
Duri Hitam			
Musang King			
Super Tembaga			

Table 2.1 displays the results of durian leaf image preprocessing on four classes, namely Bawor, Duri Hitam, Musang King, and Super Tembaga. Each class is presented

in three stages: the original image, the result of background removal and cropping, and the result of Sobel Edge Detection. These stages demonstrate that preprocessing is capable of reducing noise, focusing on the leaf object, and clarifying morphological features such as leaf veins, leaf vein patterns, and edge contours that are important in the classification process.

E. Data Splitting

After the preprocessing stage is complete, the dataset was divided into three main parts: training, validation, and testing data with a ratio of 70:15:15. The training data, consisting of 1,176 images, was used to train the model to learn the patterns and characteristics of durian leaf images. The validation data, consisting of 252 images, was used to evaluate model performance during training and assist in model parameter adjustment, as well as to prevent overfitting. The testing data, consisting of 252 images, was used to evaluate the final model performance after training is complete, providing an objective overview of the model's ability to classify new data.

F. Training Parameters

The training parameters for the MobileNetV3Small model include a batch size of 32 and a total of 120 epochs, divided into 30 epochs for warm-up (with base layers frozen) and 90 epochs for fine-tuning (with the last 20 layers unfrozen). The selection of the 20 deepest layers for fine-tuning was based on three primary considerations. First, these 20 layers represent high-level bneck blocks responsible for extracting domain-specific semantic features such as vein patterns and leaf edge contours. Second, freezing the earlier layers preserves general low-level features learned from ImageNet while preventing catastrophic forgetting. Third, this approach aligns with transfer learning practices in lightweight architectures, where fine-tuning a small subset of the final layers provides an optimal balance between domain adaptation and computational efficiency. The optimizer used is Adam with a learning rate of 0.001 during warm-up and 0.0001 during fine-tuning, supplemented by learning rate adjustment using cosine annealing. The loss function used is categorical crossentropy, and evaluation is performed using accuracy and F1-score (macro) metrics.

To prevent overfitting, early stopping is applied with a patience of 6 epochs during warm-up and 10 epochs during fine-tuning, along with model checkpointing to save the best model. Data leakage checking is also performed prior to training. This configuration aimed to produce a model with high classification performance and strong generalization capability, while mitigating the risk of overfitting through a synergistic combination of early stopping, cosine annealing, and Dropout operating across 120 training epochs.

G. Training and Classification

The training and classification stage in this study aims to train the model to recognize durian leaf image patterns across four classes: Bawor, Musang King, Duri Hitam, and Super Tembaga, using images that have undergone augmentation, preprocessing, and dataset splitting. The model used is MobileNetV3Small with a transfer learning approach that utilizes initial weights from ImageNet. This model was selected due to its lightweight and efficient architecture, suitable for image classification with low computational requirements. The MobileNetV3Small architecture consists of an initial convolutional layer (Conv2D), followed by a series of inverted residual blocks (bneck) that serve as the main feature extractors, enabling the model to process images efficiently and effectively.

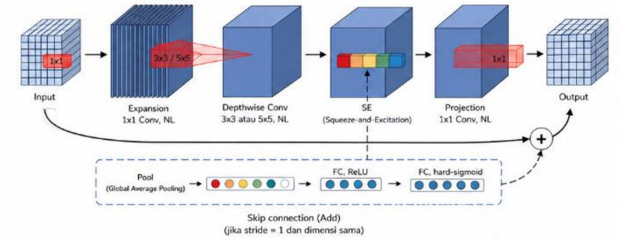


Figure 2. 4 MobileNetV3 Block

The MobileNetV3 Block (Inverted Residual Block) structure consists of an expansion layer with 1×1 convolution to expand the channels, depthwise convolution of 3×3 or 5×5 for feature extraction, a Squeeze-and-Excitation (SE) module to reinforce important features, and a projection layer with 1×1 convolution to restore the channel dimensions. Residual connections (skip connections) are used to maintain information flow and training stability when the input and output dimensions are equal (stride = 1).

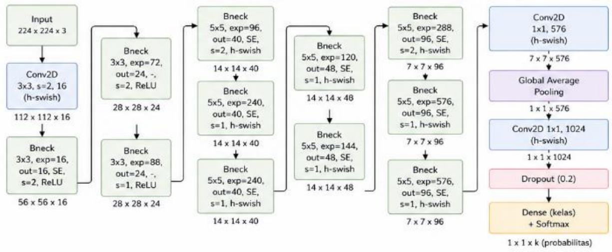


Figure 2. 5 MobileNetV3SmallGeneralArchitecture

The complete MobileNetV3Small architecture in this study illustrates the image processing flow from input to classification output, beginning with an input image of 224 × 224 × 3, which passes through an initial convolutional layer and a series of bneck blocks with varying configurations. Each block plays a role in progressively extracting features. After the feature extraction process, the image is processed using Global Average Pooling to reduce spatial dimensions, followed by 1×1 convolution to enhance feature representation, and Dropout as a regularization technique. In the final stage, a Dense layer with softmax activation is used to generate classification probabilities into four durian leaf classes.

In this study, the built-in top layer of MobileNetV3Small was not used (include_top=False), and was replaced with a new classification head for four-class durian leaf classification. The process begins by loading the durian leaf image dataset that has been processed with background removal, cropping, resizing, and Sobel edge detection. The dataset is divided into training (70%), validation (15%), and testing (15%) data. The use of ImageNet pre-trained weights potentially introduces domain bias, given that ImageNet is dominated by images of general objects such as animals, vehicles, and everyday items that are visually distinct from durian leaf images. Low-level features such as edges, textures, and intensity gradients learned from ImageNet are generally universal and transferable to the leaf image domain. However, high-level features that are more specific to ImageNet may be less relevant to the durian leaf classification domain. To mitigate the impact of this domain transfer, the study employed two primary strategies. First, Sobel Edge Detection preprocessing was applied to emphasize durian leaf-specific morphological features such as vein patterns and edge contours, making the model's input more representative of the target domain. Second, a two-stage fine-tuning strategy was applied to gradually adapt the high-level layer weights to better align with the visual characteristics of the durian leaf domain, without sacrificing the general feature representations already learned during the pre-training phase. Afterwards, the output layer is modified by adding Global Average Pooling, Dropout of 0.30, and a Dense layer with softmax activation.

The next stage is freezing the base model (trainable=False), where only the classification layers are trained. Training is carried out with a two-stage fine-tuning strategy, starting with a warm-up of 30 epochs at a learning rate of 0.001, followed by fine-tuning for 90 epochs at a learning rate of 0.0001. Cosine annealing is used to adjust the learning rate, and early stopping is applied to prevent overfitting. The best model is saved using model checkpointing and tested using the testing data to produce classification probabilities for each durian leaf image. The transfer learning approach with two-stage fine-tuning, along with the use of cosine learning rate scheduling and early stopping, produces a model that is efficient, accurate, and capable of generalizing well to new data.

H. Evaluation and Comparison

Comparative evaluation was conducted to determine the best model based on training schemes that examine the effect of Sobel Edge Detection and two-stage fine-tuning [21]. Accuracy and loss metrics are used to assess model performance on training, validation, and testing data. The best model is selected based on the highest accuracy, lowest loss, and performance stability between training and validation data, as well as to analyze the effectiveness of the applied methods.

1) Accuracy

Accuracy is an evaluation metric that measures the level of correctness of the model in classifying data [21]. Accuracy is calculated by comparing the number of correct predictions with the total number of all tested data. The accuracy formula can be expressed as follows:

$$A = \frac{b}{n} \times 100\% \qquad (4)$$

Keterangan:
A : Accuracy
b : Number of Correct Predictions
n : Total Data

In this study, accuracy is calculated on training, validation, and testing data to measure the consistency of model performance. However, accuracy alone is not sufficient to fully describe overall model performance, particularly in the case of multi-class classification, and therefore needs to be combined with other metrics such as loss.

2) Loss

Loss (loss function) measures the level of error in model predictions [21]. In this study, categorical crossentropy is used for multi-class classification. The smaller the loss value, the better the model performance. The loss function can be calculated using the formula:

$$L(\theta) = -\sum_{i=1}^{k} y_i \log \binom{\wedge}{y_i} \qquad (5)$$

Keterangan:
loss : model prediction error value
y_i : actual label (ground truth)
$\hat{y}_i$: predicted probability of the model for class i
k : number of classes

The loss value is monitored on training and validation data to track model development, with a stable decrease in loss value indicating that the model is not overfitting.

III. RESULTS AND DISCUSSION

This study evaluates the performance of a durian leaf image classification model using MobileNetV3Small with Sobel Edge Detection preprocessing and a two-stage fine-tuning strategy. Evaluation was conducted through analysis of training and validation loss curves, accuracy curves, confusion matrix, classification report, and comparison with previous studies.

At the beginning of training, the training loss and validation loss values were approximately 1.29 and 1.04, respectively. The initial difference between these two values is considered normal given that the new classification layer was randomly initialized, causing predictions on the training data to tend to be unstable in the early epochs. The sharp decrease in loss during the early epochs reflects a productive learning phase, in which the classification layer rapidly learns discriminative patterns from the ImageNet feature representations stored in the base layers through the prior pre-training process.

The small spike observed around epoch 30 is a typical manifestation of the transition from warm-up to fine-tuning, when the previously frozen base layers begin to be unfrozen and fully optimized. This transition causes a temporary change in the distribution of the network's internal activations as the weights of all layers begin to be adjusted simultaneously. Nevertheless, this spike is temporary and does not indicate overall training instability.

After the transition phase, both curves show a slower but consistent and stable decrease. In the final epochs, both training loss and validation loss converge toward values of 0.03 to 0.05. The consistently small gap between the two curves throughout the training process confirms the absence of significant overfitting, while also demonstrating that the model is capable of learning patterns that generalize well to previously unseen data. This learning dynamic can be seen visually in Figure 3.1.

Although the model was trained for a total of 120 epochs, the potential for overfitting was minimized through several mechanisms applied simultaneously. First, early stopping with a patience of 6 epochs during the warm-up phase and 10 epochs during the fine-tuning phase automatically halted training when no improvement in validation loss was observed, thereby preventing the model from over-adapting to the training data. Second, cosine annealing kept the learning rate controlled so that weight updates in the final epochs were smooth and non-aggressive. Third, a Dropout rate of 0.30 in the classification head served as regularization to reduce the model's dependency on specific neurons. Empirical evidence of the absence of overfitting is reflected in the consistently small gap between training loss and validation loss throughout training, with both converging to values approaching 0.03 to 0.05 in the final epochs.

.

Figure 3. 1 Training & Validation Loss Graph

In the early epochs, training accuracy was in the range of 0.39 and validation accuracy was in the range of 0.55. The phenomenon in which validation accuracy surpasses training accuracy in these early stages can be explained by two main factors. First, the features extracted by the pre-trained base layers from ImageNet are already sufficiently representative and capable of producing reasonably correct predictions on the validation data even before the fine-tuning process takes place. Second, the newly randomly initialized classification layer still produces inconsistent predictions on the training data, causing training accuracy to be suppressed at low values in the beginning.

As training progresses, the accuracy fluctuations observed around epoch 30 are consistent with the transition between training stages, as also reflected in the loss curves. After the transition phase has passed, both accuracy curves increase consistently and steadily without significant oscillations. At the end of training, training accuracy approaches a value of 1.0 and validation accuracy is very close to that value. The convergence of both at high values demonstrates the model's strong generalization capability as well as the success of the two-stage fine-tuning strategy in avoiding both overfitting and underfitting phenomena. The development of accuracy throughout training is displayed in Figure 3.2.

.

Figure 3. 2 Training & Validation Accuracy Graph

The features extracted by the pre-trained base layers were already sufficiently representative to produce good predictions on validation data even before fine-tuning commenced, indicating that despite the domain gap between ImageNet and durian leaf images, the learned low-level features remained relevant and generalizable. The convergence of training accuracy and validation accuracy at high values toward the end of training confirms that the two-stage fine-tuning strategy successfully adapted the feature representations from the ImageNet domain to the durian leaf-specific domain, effectively minimizing the adverse impact of domain transfer bias.

The confusion matrix in Figure 3.3 provides a detailed overview of the classification error patterns produced by the model. Out of a total of 253 test images used, the model produced only 4 classification errors, reflecting an overall very high level of accuracy.

On a per-class basis, the Bawor class successfully classified 61 out of 63 images correctly, with 2 images misclassified as DuriHitam. The DuriHitam class classified

62 out of 63 images correctly, with 1 error in the form of misclassification to the Bawor class. The MusangKing class classified 63 out of 64 images correctly, with 1 error to the DuriHitam class. The SuperTembaga class achieved perfect performance with all 63 images correctly classified without a single error.

A notable finding is that all classification errors were concentrated between the Bawor and DuriHitam classes. Morphological analysis of these two varieties reveals several visual similarities that inherently complicate the classification process. First, both varieties share a similar leaf shape — an elongated ellipse with an acuminate apex — causing the edge contour features extracted by Sobel to tend toward overlapping representations between the two classes. Second, the venation patterns of Bawor and DuriHitam both exhibit a pinnate type with relatively similar secondary vein density, making the gradient features produced by Sobel less capable of consistently distinguishing between them. Third, the leaf sizes of both varieties fall within a closely adjacent range, in contrast to MusangKing, which tends to be broader, and SuperTembaga, which possesses a more distinctive surface texture. The more distinguishing differences between Bawor and DuriHitam generally lie in leaf surface color and texture characteristics — precisely the information that becomes less prominent after Sobel preprocessing is applied.

.

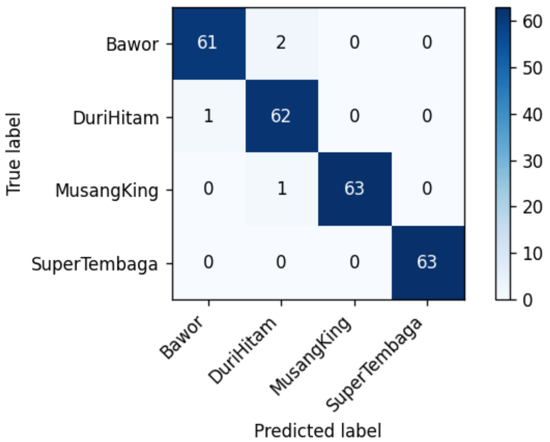


Figure 3. 3Confusion matrix

The classification report in Figure 3.4 presents a more granular evaluation of model performance at the per-class level. Precision values were recorded at 0.9839 for the Bawor class, 0.9538 for DuriHitam, 1.0000 for MusangKing, and 1.0000 for SuperTembaga. The lower precision value of DuriHitam compared to the other classes reflects the presence of false positives stemming from the misclassification of Bawor as DuriHitam.

Recall values for each class are 0.9683 for Bawor, 0.9841 for DuriHitam, 0.9844 for MusangKing, and 1.0000 for SuperTembaga. The slightly lower recall of Bawor is consistent with the two Bawor images that were misclassified as DuriHitam, while the relatively high recall of DuriHitam indicates that the majority of DuriHitam instances were correctly identified.

The F1-score per class reached 0.9760 for Bawor, 0.9688 for DuriHitam, 0.9921 for MusangKing, and 1.0000 for SuperTembaga. A macro average F1-score of 0.9842 and a weighted average F1-score of 0.9843 indicate high and consistent performance across all classes. The balanced support distribution, with each class containing 63 or 64 test images, ensures that the overall accuracy of 98.42% is representative and unbiased by class imbalance.

	precision	recall	f1-score	support
Bawor	0.9839	0.9683	0.9760	63
DuriHitam	0.9538	0.9841	0.9688	63
MusangKing	1.0000	0.9844	0.9921	64
SuperTembaga	1.0000	1.0000	1.0000	63
accuracy			0.9842	253
macro avg	0.9844	0.9842	0.9842	253
weighted avg	0.9845	0.9842	0.9843	253

Figure 3. 4 Classification report

Table 3.1 summarizes the comparison of the validation accuracy of this study with eight previous studies that used various deep learning architectures for durian leaf image classification as well as similar plant images. The model developed in this study achieved the highest validation accuracy of 98.42%, surpassing all comparison studies.

TABLE 3. 1
COMPARISON OF VALIDATION ACCURACY WITH PREVIOUS STUDIES

No	Researcher (Year)	Main Model Architecture	Validation Accuracy
1	Elroy, 2024	CNN Standar	80,00%
2	Kurniawan & Ariatmanto, 2024	MobileNetV2	90,00%
3	Wirabowo & Susilawati, 2025	CNN (*Lightweight*)	90,03%
4	Kulsum & Cherid, 2023	ResNet50	91,00%
5	Diana et al., 2025	Xception	92,00%
6	Eiamin, 2025	YOLOv5	93,33%
7	Voo , 2026	MobileNet (*Fine-Tuning*)	93,69%
8	Klangbunrueang , 2026	ConvNeXt	98,00%
9	This Study	MobileNetV3Small + *Sobel*	98,42%

The closest study is Klangbunrueang [4] with 98.00% using ResNet and ConvNeXt architectures, followed by Voo [5] with 93.69% using MobileNet with conventional fine-tuning, Kurniawan [6] with 90.00% using MobileNetV2, and Elroy [7] with 80.00% using standard CNN without transfer learning. To facilitate visual comparison, the results of all studies are presented in two different visualization forms in Figure 3.5 and Figure 3.6.

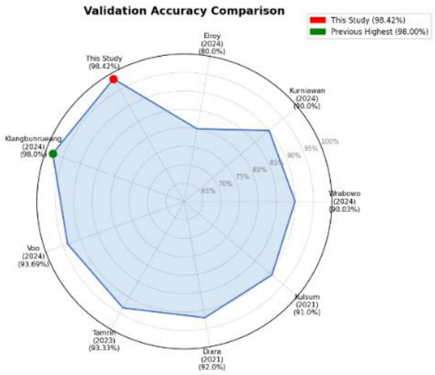


Figure 3. 5 Radar Chart of Validation Accuracy Comparison

Figure 3.5 presents a comparison of the validation accuracy of all studies in the form of a radar chart (spider chart). Each vertex on the radar chart represents one study, while the distance of the point from the center reflects the magnitude of the accuracy value achieved. The farther the point from the center, the higher the accuracy obtained. From this visualization, it can be seen that the point representing this study is positioned furthest from the center compared to all other points, indicating that the developed model consistently outperforms all comparison studies. The radar chart also intuitively illustrates a considerably large performance gap between the group of studies with accuracy below 94% and the two top-performing studies, namely this study and Klangbunrueang [10], both of which are positioned well beyond the reach of the other points.

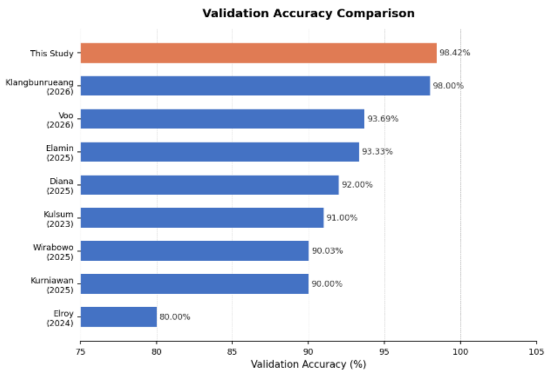


Figure 3. 6 Bar Chart of Validation Accuracy Comparison

Figure 3.6 presents the same comparison in the form of a bar chart sorted from the lowest to the highest accuracy value. This visualization facilitates more precise reading of the numerical differences between studies compared to the radar chart. The bar representing this study appears as the tallest bar, with a value of 98.42%, slightly surpassing the bar of Klangbunrueang [10] at 98.00%. The bar chart also reinforces the presence of a fairly significant performance leap between the group of studies based on conventional lightweight architectures such as MobileNetV2 and Lightweight CNN in the range of 90%, and the group of architectures implementing transfer learning and more mature optimization strategies in the range of 93% and above. The position of this study at the top of the chart confirms that the combination of MobileNetV3Small, Sobel Edge Detection, and two-stage fine-tuning is a competitive approach even against architectures with far larger parameter capacities.

The superiority of this study over Klangbunrueang [10], which uses architectures far larger in terms of parameter count, has significant implications from a computational efficiency standpoint. The fact that MobileNetV3Small, which is a lightweight architecture, is capable of surpassing far larger architectures proves that the contribution of Sobel Edge Detection as preprocessing and the two-stage fine-tuning strategy are effective in maximizing the representational capacity of a compact architecture. This approach practically opens up opportunities for model deployment on resource-limited devices, such as mobile devices or edge computing systems in agricultural fields, without significantly sacrificing accuracy.

It should be noted that the comparison in Table 3.1 was conducted against prior studies that used different datasets; therefore, differences in accuracy cannot be attributed solely to architectural superiority, as they are also influenced by the characteristics of each respective dataset. A fair comparison would require testing all competing architectures on an identical dataset, which is acknowledged as a limitation of this study and recommended as a direction for future research. Nevertheless, the results achieved by MobileNetV3Small with Sobel preprocessing and two-stage fine-tuning remain competitive among all studies compared within the durian leaf classification domain.

To evaluate the contribution of Sobel Edge Detection preprocessing more concretely, a direct comparison was conducted between MobileNetV3Small with and without Sobel preprocessing on an identical dataset. The comparison results presented in Table 3.2 show that the application of Sobel Edge Detection yielded a significant improvement in validation accuracy from 89.72% to 98.42%, equivalent to an increase of 8.70 percentage points. This finding empirically confirms that Sobel preprocessing plays a crucial role in maximizing the representational capability of MobileNetV3Small for the durian leaf classification domain, by emphasizing domain-specific morphological features such as vein patterns and edge contours that serve as the primary discriminating characteristics between varieties. Without Sobel preprocessing, the model tends to struggle in extracting discriminative morphological features directly from RGB images, resulting in substantially lower accuracy.

TABLE 3. 2
VALIDATION ACCURACY COMPARISON OF MOBILENETV3SMALL WITH AND WITHOUT SOBEL EDGE DETECTION

Model	Validation Accuracy
MobileNetV3Small with Sobel Edge Detection	98,42%
MobileNetV3Small without Sobel Edge Detection	89,72%

In addition to comparisons against prior studies, a direct comparative experiment was conducted between MobileNetV3Small and two larger architectures, namely ResNet50 and ConvNeXt, on the same dataset. The results presented in Table 3.3 show that MobileNetV3Small with Sobel preprocessing achieved the highest validation accuracy of 98.42%, surpassing ResNet50 at 97.88% and ConvNeXt at 98.20%. Although the accuracy margins among the three models are relatively small, these results carry significant implications from a computational efficiency standpoint. MobileNetV3Small, as a lightweight architecture, achieved the highest accuracy compared to ResNet50 and ConvNeXt, which possess substantially larger parameter capacities, demonstrating that the combination of Sobel preprocessing and a two-stage fine-tuning strategy effectively maximizes the representational capacity of a lightweight architecture, making it competitive even against more complex architectures.

TABLE 3. 3
VALIDATION ACCURACY COMPARISON ACROSS ARCHITECTURES ON THE SAME DATASET

Model	Validation Accuracy
MobileNetV3Small	98,42%
ResNet50	97,88%
ConvNeXt	98,20%

Model evaluation in this study was conducted on a dataset consisting of leaf images captured under controlled and relatively clean conditions. Consequently, the model's performance of 98.42% may not fully reflect its robustness under more complex real-world field conditions. Several real-world disturbance scenarios that have not yet been evaluated include leaves damaged or perforated by insect activity, leaves exhibiting disease symptoms, leaves partially occluded by other objects such as branches or overlapping leaves, and variations in field lighting conditions. These conditions have the potential to affect the quality of both Sobel feature extraction and the background removal process, and may significantly reduce model accuracy if deployed directly in the field without further adaptation.

IV. CONCLUSIONS AND RECOMMENDATIONS

This study successfully developed a durian leaf image classification model based on MobileNetV3Small with Sobel Edge Detection preprocessing and a two-stage fine-tuning strategy. The model achieved a validation accuracy of 98.42% and a macro average F1-score of 0.9842, with only 4 classification errors out of 253 test images. The SuperTembaga class achieved perfect precision, recall, and F1-score of 1.0000, while the other classes recorded values above 0.96 for all evaluation metrics. These results indicate that the combination of Sobel Edge Detection as a preprocessing stage and a two-stage fine-tuning strategy is capable of maximizing the performance of the lightweight MobileNetV3Small architecture on the dataset used, although direct comparisons with other architectures on the same dataset are necessary to confirm the superiority of the proposed method more conclusively. It should be emphasized that the scope of this study is limited to the development and evaluation of the classification model, and does not yet encompass the application implementation stage or real-world field system testing. For future research, it is recommended to expand the scope of durian leaf varieties studied, test model performance under more diverse field conditions, and explore combinations with other preprocessing methods to improve model robustness against variations in image quality. As well as exploring additional augmentation techniques such as brightness adjustment, zooming, and color jitter to improve model robustness against variations in lighting conditions and image capture angles, while simultaneously evaluating model robustness under more diverse real-world field conditions, including damaged, perforated, disease-affected, or partially occluded leaf images, and applying model interpretability techniques such as Grad-CAM to visualize the leaf regions underlying classification decisions, as well as developing a mobile application based on TensorFlow Lite or ONNX as an offline durian seedling variety identification system usable by farmers in the field, in order to ensure model reliability and transparency prior to practical implementation in the agricultural sector.

ACKNOWLEDGMENTS

The authors express sincere gratitude to their parents for their endless prayers and support, as well as to all parties who have accompanied and provided valuable input throughout the course of this research.

REFERENCES

[1] Y. Miftahuddin and F. Zaelani, "Perbandingan Metode Efficientnet-B3 dan Mobilenet-V2 Untuk Identifikasi Jenis Buah-buahan Menggunakan Fitur Daun," 2022.

[2] C. Algemayel, D. Abou Jaoude, S. Talhouk, I. Issa, and C. Ghassibe, "Advances in machine learning models for plant species identification: A scoping review," Feb. 01, 2026, *Elsevier B.V.* doi: 10.1016/j.ecoinf.2025.103464.

[3] S. Nassor, M. Mushthofa, and K. Priandana, "Deep learning model for detection and classification of banana diseases based on leaf images," in *IOP Conference Series: Earth and Environmental Science*, Institute of Physics, 2024. doi: 10.1088/1755-1315/1359/1/012010.

[4] U. Kulsum and A. Cherid, "Penerapan Convolutional Neural Network Pada Klasifikasi Tanaman Menggunakan ResNet50," *SIMKOM*, vol. 8, no. 2, pp. 221–228, Jul. 2023, doi: 10.51717/simkom.v8i2.191.

[5] C. Y. Kim, K. S. Um, and S. W. Heo, "A novel MobileNet with selective depth multiplier to compromise complexity and accuracy," *ETRI Journal*, vol. 45, no. 4, pp. 666–677, Aug. 2023, doi: 10.4218/etrij.2022-0103.

[6] C. P. Lee, K. M. Lim, Y. X. Song, and A. Alqahtani, "Plant-CNN-ViT: Plant Classification with Ensemble of Convolutional Neural Networks and Vision Transformer," *Plants*, vol. 12, no. 14, Jul. 2023, doi: 10.3390/plants12142642.

[7] A. Naufal Abiyyu and M. Rahardi, "Comparison of Transfer learning Models MobileNetV3-Large and EfficientNet-B0 for Rice Leaf Disease Classification," 2026. [Online]. Available: http://jurnal.polibatam.ac.id/index.php/JAIC

[8] S. J. Elroy, F. Nurdiyansyah, G. Priyandoko, and K. Kunci, "3 RD MDP Student Conference (MSC) 2024 56 | Universitas Multi Data Palembang Klasifikasi Daun Durian Pada Citra Dalam Menentukan Jenis Menggunakan Convolutional Neutral Network," 2024.

[9] D. Kurniawan and D. Ariatmanto, "Identifikasi Varietas Bibit Durian Menggunakan Mobilenetv2 Berdasarkan Gambar Daun," 2024. [Online]. Available: http://e-journal.stmiklombok.ac.id/index.php/jireISSN.2620-6900

[10] R. Klangbunrueang, W. Chansanam, N. Iam-On, and T. Boongoen, "AI-Driven Durian Leaf Disease Classification Using Benchmark CNN Architectures for Precision Agriculture," *Applied Sciences*, vol. 16, no. 9, p. 4062, Apr. 2026, doi: 10.3390/app16094062.

[11] D. Diana *et al.*, "Convolutional Neural Network Based Deep Learning Model for Accurate Classification of Durian Types," *Journal of Applied Data Sciences*, vol. 6, no. 1, pp. 101–114, Jan. 2025, doi: 10.47738/jads.v6i1.480.

[12] N. M. Voo, T. M. Lim, and Y. M. Lim, "Fine-Tuning MobileNet for Durian Variety Classification," in *IEEE ICCBE 2025*, Basel Switzerland: MDPI, Mar. 2026, p. 46. doi: 10.3390/engproc2026128046.

[13] P. Eiamin, P. Janoum, J. Puangmanee, and P. Chophuk, "Comparative Evaluation of YOLOv5, YOLOv8, and TensorFlow for Optimal Model Selection in Durian Leaf Disease Detection via Mobile Application," in *Frontiers in Artificial Intelligence and Applications*, IOS Press BV, Mar. 2025, pp. 401–411. doi: 10.3233/FAIA241595.

[14] I. Wirabowo and I. Susilawati, "Implementasi Convolution Neural Network (CNN) untuk Deteksi Penyakit pada Daun Jagung Berbasis Citra Digital," *Jurnal Pustaka Data (Pusat Akses Kajian Database, Analisa Teknologi, dan Arsitektur Komputer)*, vol. 5, no. 1, pp. 233–241, Jun. 2025, doi: 10.55382/jurnalpustakadata.v5i1.1046.

[15] S. A. Wagle, R. Harikrishnan, S. H. M. Ali, and M. Faseehuddin, "Classification of plant leaves using new compact convolutional neural network models," *Plants*, vol. 11, no. 1, Jan. 2022, doi: 10.3390/plants11010024.

[16] H. Pebrian and E. Hartati, "Potato Leaf Disease Classification Using MobileNetV3 Architecture with Adam and Stochastic Gradient Descent Optimizers," *Research Article*, vol. 6, no. 1, pp. 81–95, 2026, doi: 10.53623/gisa.v6i1.1063.

[17] P. Sridhara Acharya, "MobileNetV3Small-CM FusionNet: A Lightweight Deep Learning Framework for Multi-Class Arecanut Disease Classification Using Feature Fusion," 2024. [Online]. Available: https://www.jisem-journal.com/

[18] K. Navinkumar, R. Logesh, P. VishnuBabu, and A. V. Ananthalakshmi, "FPGA implementation of sobel edge detection algorithm," *EAI Endorsed Transactions on Internet of Things*, vol. 10, 2024, doi: 10.4108/eetiot.5148.

[19] K. Alomar, H. I. Aysel, and X. Cai, "Data Augmentation in Classification and Segmentation: A Survey and New Strategies," *J. Imaging*, vol. 9, no. 2, Feb. 2023, doi: 10.3390/jimaging9020046.

[20] Y. Apriansyah *et al.*, "Implementation Of Edge Detection Using The Sobel Operator On Papaya Leaf Images," *Jurnal Ilmiah Informatika dan Komputer (Informatech)*, vol. 2, no. 2, pp. 164–168, 2025, doi: 10.69533.

[21] S. Farhadpour, T. A. Warner, and A. E. Maxwell, "Selecting and Interpreting Multiclass Loss and Accuracy Assessment Metrics for Classifications with Class Imbalance: Guidance and Best Practices," *Remote Sens. (Basel).*, vol. 16, no. 3, Feb. 2024, doi: 10.3390/rs16030533.