Implementation and Comparative Analysis of CNN and Transfer Learning Models (EfficientNetB0, MobileNetV2, and ResNet50) for Rice Leaf Disease Detection Based on Digital Images

Tri Wahyu Utami [1]*, Mega Novita [2], Khoiriya Latifa [3]***
* Program Studi Informatika, Universitas Persatuan Guru Republik Indonesia Semarang
utamimi3008@gmail.com [1], novita@upgris.ac.id [2], khoiriyalatifah@upgris.ac.id [3]

Article Info

Article history:

Received 2025-10-31
Revised 2025-12-05
Accepted 2025-12-10

Keyword:

CNN,
Transfer Learning,
EfficientNetB0,
MobileNetV2,
ResNet50.

ABSTRACT

Rice leaf diseases significantly reduce agricultural productivity, making early and accurate detection essential, particularly in rice-producing regions such as Indonesia. This study proposes an automated rice leaf disease detection system based on Convolutional Neural Networks (CNN) and transfer learning. The dataset, obtained from the Mendeley Data Repository, consists of 6,889 images classified into eight categories: Bacterial Leaf Blight, Brown Spot, Healthy Rice Leaf, Leaf Blast, Leaf Scald, Narrow Brown Leaf Spot, Rice Hispa, and Sheath Blight. The dataset was divided into 70% training, 15% validation, and 15% testing. A baseline CNN model and three pre-trained models—EfficientNetB0, MobileNetV2, and ResNet50—were evaluated using accuracy, precision, recall, F1-score, and confusion matrix analysis. The baseline CNN achieved a test accuracy of 48.26%, while EfficientNetB0 achieved 58.41%. In contrast, MobileNetV2 and ResNet50 demonstrated significantly better performance, with test accuracies of 79.98% and 76.60%, respectively. MobileNetV2 exhibited the most balanced performance across all classes, showing superior generalization capability and computational efficiency. The best-performing model was integrated into a Streamlit-based application, enabling real-time rice leaf disease detection through image upload. The results confirm that transfer learning substantially improves classification accuracy and robustness compared to conventional CNNs. This study highlights the potential of lightweight deep learning models for practical implementation in smart agriculture systems and provides a reliable solution for automated rice disease detection in real-world conditions.

I. INTRODUCTION

In the modern era, developments in artificial intelligence (AI) technology have spurred significant improvements in a variety of industries. One notable application is in agriculture, specifically in the process of identifying and classifying various types of herbal plants using the Convolutional Neural Network (CNN) algorithm as an accurate and effective image processing method[1]. Rice is a staple food for over half of the world's population, so maintaining global food security depends on its sustainable production. However, rice production is often reduced by a number of leaf diseases, such as Brown Spot, Leaf Scald, Bacterial Leaf Blight, and Leaf Blast, which are usually caused by bacterial and fungal infections. Automatic seasonal recognition through visual image analysis is essential in a number of fields, such as weather monitoring and the development of geographic information systems based on digital image processing[2]. Rice leaves were chosen as the main object in the early stages of disease diagnosis because they have a larger surface area than other plant parts and make infection symptoms like yellowing and spots easier to see[3]. Images were classified using the Convolutional Neural Network (CNN) method, which is thought to be the most accurate and efficient deep learning algorithm in image identification when compared to other machine learning techniques This study can further our understanding of the application of Deep Learning for plant disease image classification and be a reference for future research in the domains of precision agriculture and pattern recognition technology[4]. According to an analysis of the efficacy of transfer learning architecture for identifying leaf diseases in food crops, deep learning and image classification have been used in numerous studies[5].

According to earlier research, the Convolutional Neural Network (CNN) approach has been widely used in plant leaf disease classification because of its ability to extract visual features without the need for manual engineering[6]. Based on a systematic review of more than 120 studies, CNN has emerged as a state-of-the-art method for diagnosing plant diseases using a range of dataset methodologies and plant species, according to another study[7]. Further studies show that CNN models trained on large-scale datasets can be applied to agriculture through transfer learning, which can improve accuracy and speed up training[8]. Some preliminary studies have classified plant leaf diseases with an accuracy of over 90% using pre-trained architectures such as VGG and ResNet[9]. Meanwhile, another study asserted that their objective was to use a Convolutional Neural Network (CNN) with a VGG16 architecture in order to ascertain the accuracy findings and make it easier for farmers to classify various diseases of rice plants [10]. As evidenced by the technique of diagnosing plant leaf diseases using the CNN and Random Forest algorithms, the combination of CNN feature extraction and Random Forest classification can improve accuracy when compared to traditional methods[11]. Then, another study successfully developed an autonomous detection system using CNN to classify various types of rice diseases using field-taken leaf photos and publicly accessible datasets[12]. In a different study, CNN artificial neural networks with the ResNet50V2 architecture and digital images were used to automatically identify eight different types of rice leaf diseases[13]. This study showed that the VGG-19 transfer learning approach improved the rice plant disease detection system's accuracy to 93%[14]. With a final accuracy of 97.1% on the Android platform, CNN was also successfully used to categorize apple plant diseases from leaf photos[15]. Additional research indicates that deep features that were retrieved using the VGG-19 architecture and classified using SVM were able to achieve the highest accuracy of 93.94%[16]. Another study claims that the CNN method can properly detect diseases and collect rice leaf image data from a range of rice plant health conditions[17]. Meanwhile, another study that used CNN and transfer learning to classify banana leaf disease found that the model overfits; as a result, dropout is used for regularization[18].

This study builds and analyzes a CNN-based and Transfer Learning-based rice leaf disease detection system using three pre-trained architectures—EfficientNetB0, MobileNetV2, and ResNet50—implemented through the Streamlit framework. The comparison methodology used in this study, which assesses model performance in terms of computing accuracy, efficiency, and potential application in digital technology-based smart agriculture systems, makes it unique.

II. METHODS

Convolutional Neural Networks (CNNs), a deep learning technique that works to gradually identify visual patterns in image data, was the primary method employed in this study. Convolution, pooling, and fully connected layers are some of the layers that make up the CNN architecture. These layers operate one after the other to extract and categorize picture characteristics. Because CNN has a multi-layered network architecture, it falls under the deep learning category. Through rigorous training, deep learning—a subfield of machine learning—allows computer systems to replicate how the human brain recognizes patterns and completes certain tasks[19].

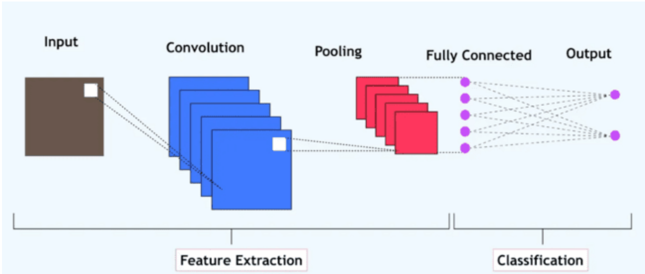


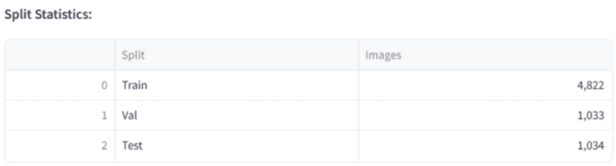
Figure 1.Architecture CNN

Figure 1 explains the working structure of CNN, which includes several main stages, namely input data processing, feature extraction, classification process, and generating the final output.

However, many previous studies still rely solely on accuracy as the primary evaluation metric, which may be misleading in multi-class and imbalanced datasets. Furthermore, limited research has comparatively analyzed lightweight transfer learning architectures for rice disease classification in a real-world web application context. Therefore, this study aims to systematically evaluate multiple CNN-based architectures using detailed per-class metrics such as precision, recall, F1-score, and confusion matrix analysis, as well as implementing the best-performing model in a Streamlit-based application.

A. Dataset and Pre-Proccessing Data

In order to improve data variety and decrease overfitting, image preprocessing often entails scaling, pixel normalization, and augmentation techniques like rotation, flipping, and brightness adjustments In order to enhance input quality prior to feature extraction, noise reduction and histogram balancing are also included in the preprocessing stage of numerous CNN application experiments conducted in Indonesia[20]. In order to retain minority class representation during training and validation, the dataset was split using stratified split to maintain class proportions[21]. The Rice Leaf Diseases Image Dataset was obtained from Mendeley Data and utilized in this investigation. This dataset covered eight types of rice leaf diseases and included 5,188 augmented photos in addition to 1,701 original photographs. To guarantee image size uniformity and enhance model generalization, pre-processing was carried out.



Split Statistics:

	Split	Images
0	Train	4,822
1	Val	1,033
2	Test	1,034

Figure 2.Data Subset Division

To guarantee a balanced class distribution in each subset, the dataset was then split into three subsets using the stratified split method, as seen in Figure 1, with a percentage of 70% training data, 15% validation, and 15% testing. In multi-class picture classification, this stratification strategy worked well to preserve the percentage of minority classes.

Using a Python package called data_prep.py, the entire process of directory scanning, class distribution computation, and dataset division was carried out automatically to guarantee experiment reproducibility using a random seed setting of 42. Table I shows the distribution of the number of photos in each class prior to data division.

TABEL I
DISTRIBUTION OF PHOTOS

No	Class	Number of Images
1	Leaf Blast	1.234
2	Sheath Blight	1.096
3	Brown Spot	1.077
4	Rice Hispa	879
5	Leaf Scald	749
6	Bacterial Leaf Blight	716
7	Healthy Rice Leaf	668
8	Narrow Brown Leaf Spot	470
Total		**6.889**

The underlying dataset lacks detailed metadata about shooting conditions, such as camera kind, shooting distance, and lighting intensity. In order to preserve model input consistency, all photos are standardized to a size of 224 by 224 pixels during the pre-processing stage.

Data preparation and augmentation are crucial procedures to increase the quality and generalization capabilities of models in identifying illnesses on rice leaves. All input photos were scaled to 224 × 224 pixels to match the input standards of the EfficientNetB0, MobileNetV2, and ResNet50 designs. After resizing, the pixel values were normalized by dividing each intensity value by 255 so that they were in the range [0,1]. Next, the images were standardized using ImageNet statistical parameters, namely mean [0.485, 0.456, 0.406] and standard deviation [0.229, 0.224, 0.225] in each RGB channel. This process aims to align the data distribution with the pre-trained model, thereby improving stability and convergence speed during network training.

Random data augmentation techniques were used to boost the training data's diversity and lower the chance of overfitting. Random horizontal and vertical flips (p = 0.5), brightness adjustments within ±20%, contrast and saturation adjustments within 0.8–1.2, and hue variations within ±0.1 were among the augmentation strategies. To preserve numerical stability, pixel values were trimmed between -2.5 and 2.5 after the augmentation process. The entire augmentation method was solely applied to the training data, while the validation and test data did not undergo augmentation to retain the objectivity of the evaluation and minimize potential data leaking. The detailed preprocessing and augmentation pipeline was implemented in the dataset module (dataset.py) using TensorFlow tf.data API, including on-the-fly augmentation to ensure memory efficiency and randomness for each epoch.

B. Architecture Baseline CNN

The basic CNN model, commonly used in local research, max pooling, a flatten layer, and dense as a classifier for image classification tasks[22]. The convolution operation formula and feature map output size are usually explicitly described in Indonesian research to clarify the architecture design[23]. The baseline CNN architecture built in this work consists of three primary convolution blocks. Each block has a Conv2D layer with a 3×3 kernel size, each with 32, 64, and 128 filters, followed by Batch Normalization and 2×2 MaxPooling to gradually lower the spatial dimensions. Dropout of 0.25 is applied to each block to decrease the danger of overfitting. After that, features are retrieved using Global Average Pooling and transmitted to a Dense layer having 256 neurons with a ReLU activation function, then Dropout 0.5 is applied before the output layer using the softmax function for multi-class classification.

The convolution operation in CNN is mathematically defined as:

$$Y(i,j) = \sum_{m} \sum_{n} X(i+m, j+n) \cdot K(m,n)$$

where X is the input image matrix and K is the convolution kernel.

The ReLU activation function is then applied to introduce non-linearity, with the equation:

$$f(x) = \max(0, x)$$

It aims to expedite the training process and avoid vanishing gradient problems.The output of the convolution layer is flattened and connected to a dense layer consisting of 256 neurons with a dropout rate of 0.4 in order to prevent overfitting. The output layer uses a softmax activation function for multi-class classification with eight disease categories.

The loss function used, categorical cross-entropy, can be written as follows:

$$L = -\sum_{i=1}^{C} y_i \log(\hat{y}_i)$$

Where y_i is the actual label and ŷ_i is the model's predicted probability.

C. *Transfer Learning*

The final layer was adjusted for specific agricultural domain adaptation using pre-trained weights from ImageNet and the transfer learning technique[24]. In order to improve classification performance and training efficiency, this study employed the Transfer Learning technique. The feature extractor layers from the pre-trained models (EfficientNetB0, MobileNetV2, and ResNet50) were reused, but the classification layers were adjusted to take into consideration the number of rice leaf disease classes. The fine-tuning process was restricted to the final layer in order to accelerate convergence and avoid overfitting. With a batch size of 32 and a learning rate of 1×10^{-3} for 30 epochs, the Adam Optimizer was employed for optimization. Adam was chosen because of its ability to combine the advantages of Momentum and RMSProp, producing more reliable convergence results. A multitude of callback mechanisms are used to dynamically regulate the model training process, rather than relying just on the number of epochs. Early halting is employed by monitoring the validation loss value and halting training if there is no improvement for 7 consecutive epochs, hence preventing overfitting. In addition, ReduceLROnPlateau is implemented to cut the learning rate by 50% if the validation loss stagnates for 3 epochs. ModelCheckpoint is also used to preserve the optimum model weights depending on the validation accuracy rating. Therefore, even though the maximum number of epochs is set at 30, the model can stop earlier when the optimum circumstances have been obtained. This approach maintains the best possible model performance while guaranteeing training efficiency. Model Backbone :

D. *EfficientNetB0*

EfficientNet's implementation in Indonesian studies, such as EfficientNet-B3/B0, demonstrates good parameter efficiency, which makes it appropriate for resource-constrained medical classification and agricultural picture tasks[25]. A model scaling technique that proportionately balances the network's depth, width, and resolution:

$$\text{depth} = \alpha^{\phi}, \text{width} = \beta^{\phi}, \text{resolution} = \gamma^{\phi}$$

A comparison of the performance of EfficientNet and other architectures in a local context highlights the advantages of scaling (depth/width/resolution), even though it requires careful fine-tuning[26].

E. *MobileNetV2*

MobileNetV2 is designed for devices with limited resources. MobileNetV2 utilizes depthwise separable convolution, which significantly reduces the number of parameters, making it ideal for real-time applications and edge devices[27]. Local research comparing MobileNetV2 states that this architecture achieves a good trade-off between accuracy and memory consumption, suitable for deployment on web-based and mobile systems[28]. This model utilizes depthwise separable convolution to reduce the number of parameters and computational load without significant loss of accuracy. The complexity of operations can be compared through the efficiency ratio:

$$\text{Rasio Efisiensi} = \frac{D_K^2 \cdot M + M \cdot N}{D_K^2 \cdot M \cdot N}$$

where D is the kernel size, M is the number of input channels, and N is the number of output channels[17].

F. *ResNet50*

The ResNet50 architecture with residual blocks is frequently used for difficult visual pattern classification tasks because it enables deeper network training with vanishing gradient mitigation[19]. ResNet50 introduces the concept of residual learning using skip connections, where the input from the previous layer is instantly added to the output of the subsequent layer, to avoid performance loss in very deep networks. The basic equation is:

$$y = F(x, \{W_i\}) + x$$

dengan $F(x, \{W_i\})$ is a non-linear transformation function and x is the initial input[19]. This approach has been proven to overcome the vanishing gradient problem and improve the stability of model training in deep networks.

III. Result and Discussion

A. *CNN Baseline*

In this experiment, the baseline CNN model did not show satisfactory performance. Although the training accuracy gradually increased during the learning process, the validation accuracy fluctuated at a relatively low level, and the final test accuracy reached only 48.26%, as illustrated in Figure 3 below. This result indicates that the baseline architecture was not able to effectively capture the complex visual features of rice leaf diseases, leading to suboptimal classification performance.

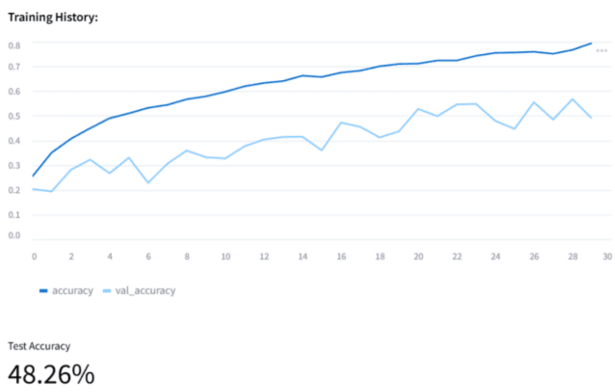


Figure 3.Result of Training Data with Baseline CNN Models

The confusion matrix shown in Figure3. Confusion Matrix Baseline CNN makes it clear that there is an imbalance in the model's ability to identify each class of rice leaf disease. Some classifications, like Leaf Blast and Bacterial Leaf Blight, have high rates of classification error and are often confused with other classes, like Sheath Blight and Narrow Brown Leaf Spot. However, the Rice Hispa and Sheath Blight classes have relatively more accurate predictions (True Positive) than other classes. This error distribution pattern demonstrates that the model still struggles to recognize certain visual characteristics in some diseases, especially those with similar leaf textures and color patterns, which results in high misclassification scores in some classes.

Figure 4. Confusion Matrix Baseline CNN Models

These findings are supported by the precision, recall, and f1-score values shown in the Baseline CNN Classification Report Table II.

TABEL II
CLASIFICATION REPORT BASELINE CNN MODELS

Class	Precision	Recall	F1-Score	Support
Bacterial Leaf Blight	1.0000	0.0187	0.0367	107
Brown Spot	0.8977	0.4877	0.6320	162
Healthy Rice Leaf	0.7143	0.5500	0.6215	100
Leaf Blast	0.7907	0.1838	0.2982	185
Leaf Scald	0.5610	0.2035	0.2987	113
Narrow Brown Leaf Spot	0.1657	0.4143	0.2367	70
Rice Hispa	0.8582	0.8712	0.8647	132
Sheath Blight	0.3418	0.9818	0.5070	165

The Bacterial Leaf Blight class had a very low recall (0.0187), indicating that only a tiny portion of the actual samples were successfully identified by the model, despite the precision value reaching 1.0000. Sheath Blight, on the other hand, has the highest recall (0.9818), indicating that the model can nearly perfectly identify this class; however, its precision value (0.3418) shows a significant rate of prediction errors in other classes that are also classified as Sheath Blight. In contrast, the Rice Hispa class has the best balanced performance with a precision of 0.8582 and a recall of 0.8712. Overall, the results in the Classification Report table show that the Baseline CNN architecture is not yet capable of optimizing feature generalization, thus reinforcing the need to apply more complex methods such as transfer learning to improve the accuracy and stability of the model's predictions. The large gap between training and testing accuracy indicates an overfitting problem, where the model learns the training data well but fails to generalize to unseen data. This limitation is likely caused by the simplicity of the baseline CNN architecture, possible class imbalance, and suboptimal hyperparameter settings. Consequently, the baseline model requires improvements such as deeper network structures, stronger data augmentation, regularization techniques, or the use of transfer learning. In this study, the baseline CNN serves primarily as a reference point for comparing the performance of more advanced transfer learning models, highlighting the importance of pre-trained features in achieving better generalization.

B. EfficientNetB0

The EfficientNetB0 model produced the lowest test accuracy, at 58.41%. Although this architecture's performance on this dataset was poor, it is theoretically possible to balance model depth, width, and resolution by using the compound scaling method.

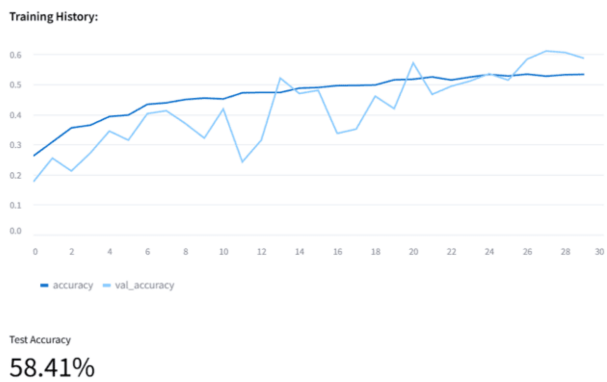


Figure 5.Result of Training Data with Transfer Learning and EfficientNetB0 Backbone

The training graph's as shown in Figure 5. significant variances in validation accuracy show that the model has trouble achieving convergence stability. The small training data set and the differences in visual distribution between the ImageNet and rice leaf datasets could be the cause of insufficient fine-tuning.

Figure 6. Confusion Matrix EfficientNetB0

Based on the confusion matrix in Figure 6, the transfer learning model with the EfficientNetB0 backbone shows more stable classification skills than the baseline CNN, even though prediction errors still happen in several classes with similar visual characteristics. The Rice Hispa and Healthy Rice Leaf classes perform the best, with 98 and 75 samples of accurate predictions on the major diagonal, respectively. The Brown Spot and Leaf Blast classes also performed fairly well, with 110 and 108 correct predictions, respectively. However, a considerable percentage of samples were mistakenly assigned to other classes, particularly Leaf Blast and Sheath Blight, suggesting a greater level of confusion in the Narrow Brown Leaf Spot and Leaf Scald classes. This pattern indicates that although EfficientNetB0 is capable of extracting more representative features than the basic architecture, the overlap of textures and visual patterns between rice leaf diseases remains a major challenge in the multi-class classification process.

TABEL III
CLASIFICATION REPORT EFFICIENTNETB0

Class	Precision	Recall	F1-score	Support
Bacterial Leaf Blight	0.6264	0.5327	0.5758	107
Brown Spot	0.5882	0.6790	0.6304	162
Healthy Rice Leaf	0.5859	0.7500	0.6579	100
Leaf Blast	0.6067	0.5838	0.5950	185
Leaf scald	0.5354	0.4690	0.5000	113
Narrow Brown Leaf Spot	0.2400	0.4286	0.3077	70
Rice Hispa	0.7259	0.7424	0.7341	132
Sheath Blight	0.8022	0.4424	0.5703	165
accuracy			**0.5841**	1034
macro avg	0.5888	0.5785	0.5714	1034
weighted avg	0.6176	0.5841	0.5886	1034

The classification report results in Table X show that the EfficientNetB0 model produced an overall accuracy of 58.41%, with average macro precision, recall, and F1-score values of 0.5888, 0.5785, and 0.5714, respectively. The Rice Hispa class had the highest F1-score of 0.7341, followed by Healthy Rice Leaf (0.6579) and Brown Spot (0.6304), indicating that the model was able to recognize more unique visual patterns in these classes. The Narrow Brown Leaf Spot class continued to perform the worst, with an F1-score of only 0.3077, indicating that the model was unable to distinguish this class from other illnesses with similar visual characteristics. Overall, these results confirm that the application of transfer learning with EfficientNetB0 provides a significant performance improvement compared to the baseline model, although further optimization is still needed to address class imbalance and improve feature separability in classes with high similarity.

C. MobileNetV2

MobileNetV2 performed better than the other two Transfer Learning models, achieving a test accuracy of 79.98%. Our model achieved a validation accuracy above 80% with a fast rate of convergence. This was due to the use of depthwise separable convolution and inverted residual blocks, which, when compared to conventional CNN models, reduced computational complexity by up to 70% while still extracting important features with exceptional efficiency. as shown in Figure 7 in below.

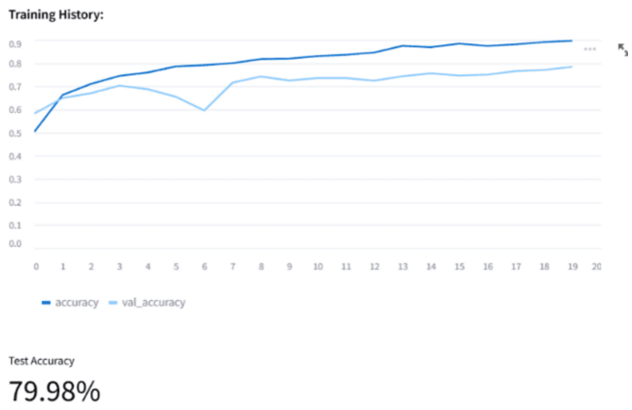


Figure 7.Result of Training Data with Transfer Learning and MobileNetV2 Backbone

Based on the MobileNetV2 confusion matrix on Figure 8., the model performs significantly better in classification than both the baseline and EfficientNetB0, with most predictions falling on the main diagonal of the matrix. This implies that the majority of images were appropriately classified. The Sheath Blight, Rice Hispa, and Bacterial Leaf Blight classes show a high number of accurate predictions with 145, 106, and 95 correctly identified photos, respectively. This pattern illustrates how effectively MobileNetV2 can identify the key visual characteristics of these three diseases.

Figure 8. Confusion Matrix MobileNetV2

The classification report results show that the MobileNetV2 model achieved an overall accuracy of 79.98%, with a macro-average F1-score of 79.10% and a weighted-average F1-score of 80.21%. These figures show relative class balance and good classification performance. Sheath Blight (F1-score = 0.8735), Bacterial Leaf Blight (0.8444), and Rice Hispa (0.8249) are among the classes that show good precision and recall, indicating the model's ability to consistently recognize the unique patterns of these diseases.

TABEL IV
CLASIFICATION REPORT MOBILENETV2

Class	Precision	Recall	F1-score	Support
Bacterial Leaf Blight	0.8051	0.8879	0.8444	107
Brown Spot	0.8312	0.7901	0.8101	162
Healthy Rice Leaf	0.9589	0.7000	0.8092	100
Leaf Blast	0.8103	0.7622	0.7855	185
Leaf Scald	0.7008	0.7876	0.7417	113
Narrow Brown Leaf Spot	0.5521	0.7571	0.6386	70
Rice Hispa	0.8480	0.8030	0.8249	132
Sheath Blight	0.8683	0.8788	0.8735	165
Accuracy			**0.7998**	1034
Macro Avg	0.7968	0.7958	0.7910	1034

Additionally, MobileNetV2's lightweight and effective architecture tends to make it more resistant to overfitting. Which demonstrates that MobileNetV2 strikes the ideal balance between inference speed and accuracy in image-based plant disease identification.

D. ResNet50

ResNet50 is more stable during training, but it performs slightly worse than MobileNetV2 with a test accuracy result of 76.60%.

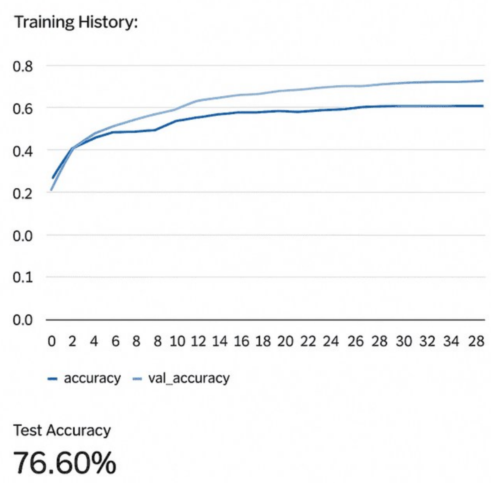


Figure 9.Result of Training Data with Transfer Learning and ResNet50 Backbone

As shown in Figure 9, ResNet50 is highly effective at learning complex hierarchical features, especially when it comes to recognizing disease patterns like Leaf Blast and Brown Spot that are similar across classes. However, this model takes longer to train than MobileNetV2 and requires more GPU RAM because of its complexity. Despite its higher precision, ResNet50 performs less well in real-time applications. This study compares the baseline CNN model with three Transfer Learning architectures: ResNet50, MobileNetV2, and EfficientNetB0. Each model was trained using the same dataset and the same training parameters: 30 epochs, a batch size of 32, and a learning rate of 1×10^{-3}.

Based on the ResNet50 confusion matrix on Figure 10, the model has relatively good classification performance, with values mostly on the main diagonal, indicating that most images were correctly classified. The high recognition rates of classes such as Sheath Blight (126), Leaf Blast (130), Brown Spot (118), and Rice Hispa (117 accurate predictions) demonstrate that ResNet50 can capture important visual features in most disease classes. Even though there are still prediction errors in several classes with similar textures, particularly between Bacterial Leaf Blight, Brown Spot, and Leaf Scald, the distribution of errors is fairly balanced and not concentrated in any particular class, indicating the stability of the model on the test data.

Figure 10. Confusion Matrix ResNet50

The results of the categorization report show that ResNet50 achieved a total accuracy of 76.60%, with a macro-average F1-score of 75.67% and a weighted-average F1-score of 76.42%. The Rice Hispa (F1-score = 0.8571) and Sheath Blight (0.8235) classes outperformed Healthy Rice Leaf (0.7960), indicating the model's ability to identify traits specific to these diseases. Some classes, such as Narrow Brown Leaf Spot (0.6538), performed worse despite having visual similarities with other classes. Although ResNet50 performed better overall than the baseline CNN in this experiment, it was still inferior to MobileNetV2 and EfficientNetB0.

TABEL V
CLASIFICATION REPORT RESNET

Class	Precision	Recall	F1-score	Support
Bacterial Leaf Blight	0.6225	0.8785	0.7287	107
Brown Spot	0.7421	0.7284	0.7352	162
Healthy Rice Leaf	0.7921	0.8000	0.7960	100
Leaf Blast	0.7975	0.7027	0.7471	185
Leaf Scald	0.7935	0.6460	0.7122	113
Narrow Brown Leaf Spot	0.5930	0.7286	0.6538	70
Rice Hispa	0.8298	0.8864	0.8571	132
Sheath Blight	0.8936	0.7636	0.8235	165
Accuracy			**0.7660**	1034
Macro Avg	0.7580	0.7668	0.7567	1034

The training results show significant variations in test accuracy between models, as summarized in Table IV below:

TABEL VI
TRAINING RESULT

Model	Train Accuracy)	(Val Accuracy)	(Test Accuracy)	General Characteristics
CNN Baseline	58.7%	30.2%	48.26%	Weak generalization, high misclassification, strong class confusion, unstable learning pattern
EfficientNet B0	49.8%	45.3%	58.41%	Stable but slow convergence
MobileNetV 2	87.2%	81.5%	79.98%	Fast convergence, computationally efficient
ResNet50	74.9%	73.5%	76.60%	High precision, but high complexity

IV. CONCLUSION

This study successfully built and examined a rice leaf disease detection system based on Convolutional Neural Networks (CNN) and Transfer Learning. An interactive interface was implemented using Streamlit. Comparing Transfer Learning to the conventional CNN model, the experimental results show a significant improvement in performance. MobileNetV2 outperformed the other two pre-trained designs with a test accuracy of 79.98%; ResNet50 came in second with 76.60% and EfficientNetB0 with 58.41%. Because of its exceptional computing efficiency, rapid convergence, and ability to generalize to complex variations in images of rice leaves, the MobileNetV2 model is the most suitable option for use in real-time web-based disease detection systemsThe ResNet50 architecture performs better than the others in terms of classification accuracy and training stability, despite requiring more processing power. Although EfficientNetB0 offers a balance between model capacity and efficiency, it hasn't yet been able to match the accuracy of the other two models. The results of the study generally lend credence to the idea that the Transfer Learning methodology greatly improves the accuracy, efficacy, and reliability of plant disease classification systems, especially when training data is limited. The implementation of this Streamlit-based system holds great potential for assisting Indonesia in embracing smart agriculture, especially with regard to expediting the accurate and automated detection of plant diseases. Future research is recommended to expand the dataset, more systematically adjust the hyperparameters, and integrate the best models with edge computing or mobile devices in order to make the system usable in the field. Thus, this study contributes empirically to the development of CNN-based image classification models and transfer learning, and provides practical suggestions for improving AI-based agricultural systems in the digital age.

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. I. Burhanuddin, Adam Syaifullah, Setiawan Adeka Putra Jaya, and Muhammad Gabriel Somoal, "Analisis Komparatif Model MobilenetV1 Dan EfficientnetB0 Dalam Klasifikasi Citra Empat Musim Menggunakan Transfer Learning," *JEKIN - J. Tek. Inform.*, vol. 5, no. 2, pp. 508–521, 2025, doi: 10.58794/jekin.v5i2.1378.

[3] A. M. Y. Abdu, "Tanaman padi menggunakan arsitektur densenet-169," p. 8, 2024, [Online]. Available: repository.uma.ac.id

[4] U. M. Area, "Mobilenetv3 Dengan Cbam Untuk Klasifikasi Sikripsi Oleh : Tarsius Tulus Hati Buuolo Fakultas Teknik Universitas Medan Area Medan Sikripsi Diajukan sebagai Salah Satu Syarat untuk Memperoleh Gelar Sarjana di Fakultas Teknik Universitas Medan Area Oleh : Tarsius Tulus Hati Buulolo Fakultas Teknik Medan," 2025.

[5] T. S. Winanto, C. Rozikin, and A. Jamaludin, "Analisa Performa Arsitektur Transfer Learning Untuk Mengindentifikasi Penyakit Daun Pada Tanaman Pangan," J. Appl. Informatics Comput., vol. 7, no. 1, pp. 68–81, 2023, doi: 10.30871/jaic.v7i1.5991.

[6] J. Lu, L. Tan, and H. Jiang, "Review on convolutional neural network (CNN) applied to plant leaf disease classification," Agric., vol. 11, no. 8, pp. 1–18, 2021, doi: 10.3390/agriculture11080707.

[7] A. Abade, P. A. Ferreira, and F. de Barros Vidal, "Plant diseases recognition on images using convolutional neural networks: A systematic review," Comput. Electron. Agric., vol. 185, 2021, doi: 10.1016/j.compag.2021.106125.

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

[9] T. D. Salka, M. B. Hanafi, S. M. S. A. A. Rahman, D. B. M. Zulperi, and Z. Omar, "Plant leaf disease detection and classification using convolution neural networks model: a review," Artif. Intell. Rev., vol. 58, no. 10, 2025, doi: 10.1007/s10462-025-11234-6.

[10] A. C. Anwari, "Klasifikasi Penyakit Tanaman Tomat Melalui Citra Daun Menggunakan Metode Convolutional Neural Network," JIMUJurnal Ilm. Multidisipliner, vol. 2, no. 03, pp. 639–647, 2024, doi: 10.70294/jimu.v2i03.421.

[11] M. Jeffry Setiawan, B. Nugroho, and A. Puspita Sari, "Klasifikasi Penyakit Daun Tanaman Menggunakan Algoritma CNNdan Random Forest," Teknologi, vol. 12, no. 1, pp. 1–8, 2022, [Online]. Available: https://journal.unipdu.ac.id/index.php/teknologi/article/view/2403

[12] M. Tiara, C. Hia, D. R. Hutasuhut, and E. M. Hutauruk, "Deteksi Penyakit Blas, Tungro & Bercak Coklat Pada Tanaman Padi Menggunakan Metode Convolutional Neural Network," J. Media Inform., vol. 6, no. 3, pp. 2221–2232, 2025.

[13] N. N. Saragih and R. Kurniawan, "(Journal of Computer Engineering, System and Science)," J. Comput. Eng. Syst. Sci., vol. 10, no. 1, pp. 299–311, 2025.

[14] A. Noor Hidayah, "Sistem Deteksi Penyakit Tanaman Padi Dan Rekomendasi Pupuk Menggunakan Metode Transfer Learning Dengan Model VGG19," pp. 1–9.

[15] A. Asrafil, A. Paliwang, M. R. D. Septian, M. Cahyanti, and R. Swedia, "Klasifikasi Penyakit Tanaman Apel Dari Citra Daun Dengan," Sebatik, no. 1410–3737, pp. 207–212, 2020.

[16] D. Margarita, "Klasifikasi Penyakit Padi Menggunakan Support Vector Machine," vol. 9, no. 4, pp. 2256–2270, 2024, [Online]. Available: https://doi.org/10.29100/jipi.v9i4.5634

[17] S. Sariah, N. Suarna, I. Ali, and D. Solihudin, "Penerapan Convolutional Neural Network (CNN) Untuk Prediksi Penyakit Tanaman Padi Melalui Citra Daun," J. Komtika (Komputasi dan Inform., vol. 9, no. 1, pp. 1–10, 2025, doi: 10.31603/komtika.v9i1.12852.

[18] D. Pratama, "Jurnal Teknologi Terpadu LEARNING," J. Teknol. Terpadu, vol. 8, no. 1, pp. 89–94, 2022.

[19] A. M. Agista and Dedy Kurniadi, "Implementasi Arsitektur Resnet50 Pada Klasifikasi Motif Batik Indonesia Menggunakan Metode Convolutional Neural Network (Cnn)," J. Rekayasa Sist. Inf. dan Teknol., vol. 2, no. 4, pp. 1324–1343, 2025, doi: 10.70248/jrsit.v2i4.2375.

[20] Y. J. Krisnawan and W. Setiawan, "Implementasi Convolutional Neural Network (CNN) untuk Deteksi Katarak pada Citra Mata," J. Ekon. Manaj. Sist. Inf., vol. 6, no. 1, pp. 443–450, 2024, doi: 10.38035/jemsi.v6i1.3066.

[21] L. Informatika et al., "Klasifikasi Cengkeh Menggunakan Metode Convolutional Neural Network (CNN)," vol. 2, no. 3, pp. 341–349, 2025.

[22] K. Prasetyo and R. Mahenra, "Analisis Kinerja Convolutional Neural Networks Baseline untuk Identifikasi Jenis Jenis Penyakit Kentang: Performance Analysis of Baseline Convolutional Neural Networks for Identifying Potato Disease Types," MALCOM Indones. J. Mach. Learn. Comput. Sci., vol. 5, no. 2, pp. 609–615, 2025, [Online]. Available: https://doi.org/10.57152/malcom.v5i2.1722

[23] A. Hibatullah and I. Maliki, "Penerapan Metode Convolutional Neural Network Pada Pengenalan Pola Citra Sandi Rumput," J. Informatics Comput. Sci., vol. 1, no. 2, pp. 1–8, 2019.

[24] A. Mufidatuzzainiya and M. Faisal, "Penggunaan Teknik Transfer Learning pada Metode CNN untuk Pengenalan Tanaman Bunga," JISKA (Jurnal Inform. Sunan Kalijaga), vol. 10, no. 2, pp. 195–206, 2025, doi: 10.14421/jiska.2025.10.2.195-206.

[25] M. Ferry Fernanda, Y. Azhar, and A. E. Minarno, "Implementasi Transfer Learning menggunakan Model Efficientnet-B3 pada Klasifikasi Pigmen Kanker Kulit," Repositor, vol. 7, no. 2, pp. 177–188, 2025.

[26] F. Zaelani and Y. Miftahuddin, "Perbandingan Metode EfficientNetB3 dan MobileNetV2 Untuk Identifikasi Jenis Buah-buahan Menggunakan Fitur Daun," J. Ilm. Teknol. Infomasi Terap., vol. 9, no. 1, pp. 1–11, 2022, doi: 10.33197/jitter.vol9.iss1.2022.911.

[27] S. F. D. Wardhana and A. Nugroho, "Perbandingan Arsitektur MobileNetV2 dan MobileNetV3 Dalam Klasifikasi Jenis Jeruk," J. Ilmu Komput. dan Bisnis, vol. 16, no. 1, pp. 25–34, 2025, doi: 10.47927/jikb.v16i1.916.

[28] A. R. Hermanto, A. Aziz, and S. Sudianto, "Perbandingan Arsitektur MobileNetV2 dan RestNet50 untuk Klasifikasi Jenis Buah Kurma Comparison of MobileNetV2 and RestNet50 Architectures for Date Fruit Classification by Type," J. Sist. dan Teknol. Inf., vol. 12, no. 4, pp. 630–637, 2024, doi: 10.26418/justin.v12i4.80358.