KHATULISTIWA SMART : JOURNAL OF ARTIFICIAL INTELLIGENCE

Khatulistiwa Smart : Journal of Artificial Intelligence
Vol. 2 No. 1 (2025) : 102 – 116
Available online at : https://journal.literasikhatulistiwa.org/index.php/smart

Web-Based Classification Optimization of Grape Leaf Diseases Using Transfer Learning in CNN to Improve Model Accuracy and Efficiency

Zainal Arifin[1] , Diana Indri Rukmana[2] , Fuadz Hasyim[3]
[123] Nurul Jadid University, Paiton, Probolinggo.
zainal@unuja.ac.id[1] ,kuliahdianaindri13@gmail.com[2] , fuadzhasyim@unuja.ac.id[3]

Doi

Submitted: 24 Nov 2025	Revision: 29 Nov 2025	Accepted: 12 Dec 2025	Published: 31 Dec 2025

Abstract Grapes are a high-value crop, but their yield is easily reduced due to leaf disease. This study focuses on developing a web-based grape leaf disease classification system that works automatically and in real time. The approach used is Convolutional Neural Network (CNN) with transfer learning using the EfficientNetB0 architecture. The research data consists of 8,000 grape leaf images divided into four classes (healthy, black rot, black spot, and leaf spot) with a composition of 80% for training, 10% for validation, and 10% for testing. The initial CNN model achieved an accuracy of 97%, which then increased to 99% after optimization using EfficientNetB0 with fine-tuning. The implementation of the system through Flask showed fast and accurate prediction results, proving that transfer learning plays an important role in improving classification performance.

Keywords *CNN; Transfer Learning; Grape Leaves; EfficientNetB0; Classification*

1. INTRODUCTION

Grapes are a horticultural commodity with high economic value that are widely cultivated in various regions, including Indonesia. (Juli et al., 2023). Grapes have high commercial value because they can be consumed directly or used as raw materials for the food and beverage industry. This potential makes grapes a strategic crop for improving farmers' welfare and promoting growth in the agroindustry sector. (Riyanto et al., 2021). However, grape productivity still faces serious obstacles, one of which is leaf disease, which results in reduced yields and fruit quality. Several types of diseases, such as black spot, leaf rot, and leaf blight, have been shown to significantly reduce production, causing losses for farmers. (Whardana et al., 2024).

Until now, grapevine leaf disease identification has generally been done manually through visual observation. This method requires special expertise, takes longer, and has the potential to result in misdiagnosis. Inaccurate identification can lead to excessive or inappropriate pesticide use, causing economic losses and environmental impacts. Therefore, a grapevine leaf disease detection system that is fast, accurate, and easy to use is needed. (Safitri et al., 2024).

The development of artificial intelligence technology, particularly computer vision with a deep learning approach, provides great opportunities in the field of modern agriculture. Convolutional Neural Network (CNN) has been proven to be capable of performing image classification effectively because it can extract visual features automatically (Juli et al., 2023). One of the most widely used CNN architectures is EfficientNet, where the EfficientNetB0 variant has the advantage of fewer parameters and low computational requirements, while still producing good accuracy (Setiawan et al., 2023). In addition, the use of transfer learning allows the use of pre-trained models, thereby improving performance even with limited training data. Hyperparameter optimization and the application of data augmentation techniques are also important to ensure that the resulting model is more accurate and resistant to data variation (Ansah et al., 2022).

Based on this background, this study proposes the use of EfficientNetB0-based CNN with a transfer learning approach for grapevine leaf disease classification. This system is implemented in the form of a web-based application so that it can be used directly by farmers and general users. This study is expected to contribute to the development of digital technology to support precision agriculture, particularly in the early detection of grapevine leaf diseases (Penyakit & Jagung, 2022).

2. RESEARCH METHODS

2.1. Research Framework

This study uses a quantitative approach with an experimental method to develop a grapevine leaf disease classification system based on Convolutional Neural Network (CNN) with transfer learning using the EfficientNetB0 architecture.

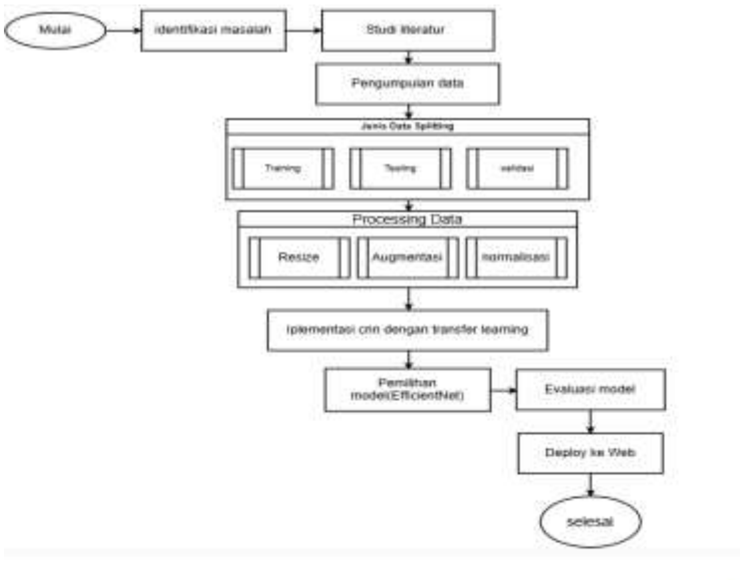


Figure 1 Research Flow

2.2. Research Method

This study aims to improve the accuracy of automatic grape leaf disease identification using a Convolutional Neural Network (CNN) with a Transfer Learning approach on the EfficientNetB0 architecture. The research dataset was obtained from Kaggle, consisting of approximately 8,000 grape leaf images divided into four classes: healthy, leaf spot, black spot, and black rot. The data was divided into training (80%), validation (10%), and testing (10%) sets (Pencurian et al., n.d.), The images were then processed through a preprocessing stage consisting of resizing the images to 224×224 pixels, normalization, and augmentation (rotation, flipping, zooming, and color changes) to increase the variety and generalization ability of the model. EfficientNetB0 was chosen for its computational efficiency and high accuracy through the compound scaling technique, as well as for utilizing pre-trained models to speed up the training process. Evaluation was carried out by monitoring the accuracy and loss in the training and validation data to detect overfitting or underfitting. After the model was trained, it was implemented in a Flask-based web application, allowing users to upload grape leaf images and obtain disease predictions directly (Ratri et al., 2023).

2.3. Design Model

This research designs a web-based grapevine leaf disease classification system using Flask integrated with a transfer learning-based CNN model. The main objective of this system is to provide accurate, efficient, and easy-to-use classification for end users.

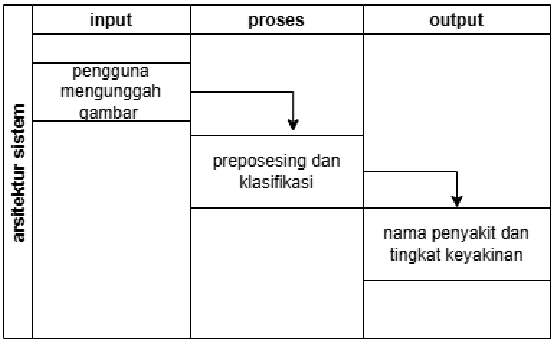
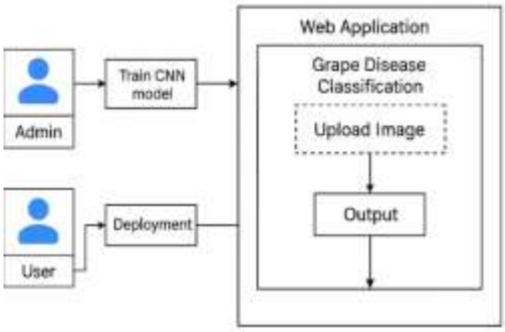


***Figure* 2.** system architecture

The system has three main components, namely input, process, and output. In the input stage, users upload grapevine images through an interactive web page. The process stage is carried out by an EfficientNetB0-based CNN model, which extracts visual features and classifies images into four categories of grapevine diseases. The output stage displays the classification results along with probability values (confidence scores) as indicators of the model's confidence.



Figurer 3. system diagram

The system diagram shows the interaction between two types of users: admin and user. Admins are responsible for training the CNN model and deploying it to the web system, while users upload images and receive prediction results from the system. The CNN model uses a transfer learning approach with EfficientNetB0 loaded without the top layer (include_top=False) and uses pre-trained weights from ImageNet. All main layers are frozen so that the model functions as a feature extractor. Additional classification layers are added, including GlobalAveragePooling2D, Dropout 0.4 to prevent overfitting, and a Dense layer with a softmax activation function according to the number of disease classes. The model is compiled using the Adam optimizer with a learning rate of 0.0005 and categorical_crossentropy loss, and evaluated using the accuracy metric. This approach enables more efficient training and optimal classification results (Ratri et al., 2023) .

2.4. Testing Method

The testing method was conducted to evaluate the performance of the EfficientNetB0 transfer learning-based grape leaf disease classification model. The evaluation included measurements of accuracy, precision, recall, F1-score, and performance visualization through a confusion matrix. Accuracy was calculated as the percentage of correct predictions from all test data:

$$Akurasi = \frac{(TP)+(TN)}{TP+TN+FP+FN} \qquad \textbf{2.1}$$

Where TP (True Positive) and TN (True Negative) indicate correct predictions, while FP (False Positive) and FN (False Negative) indicate incorrect predictions. Precision indicates the model's ability to correctly classify positive classes, recall describes the model's ability to find all positive cases, and F1-score is the harmonic mean of precision and recall to assess the balance between the two:

$$Presisi = \frac{TP}{TP+FP}, Recall = \frac{TP}{TP+FN}, \; F1\,score = 2\frac{Precision\ x\ Recall}{Presicion+Recall}\,. \quad \textbf{2.2}$$

The confusion matrix is used to visualize the comparison between the original labels and the model predictions, making it easier to identify the types of classification errors. The four components TP, TN, FP, and FN form the basis for calculating evaluation metrics, which collectively describe the accuracy and effectiveness of the model in classifying grape leaf diseases (Simanjuntak et al., 2020).

3. Result and Discussion

The main focus of this research is to build a simple web-based classification system using the Flask framework. The implementation process was carried out by training a dataset that included four categories of diseased grape leaves and healthy grape leaves.

3.1. Dataset Description

Data collection is an important stage in research that aims to obtain accurate, relevant, and usable information to answer research questions and achieve research objectives. The data in this study was sourced from a secondary dataset downloaded from the Kaggle open data provider website using the keyword "grape leaf disease." The dataset contains a collection of images of grape leaves in both healthy and diseased conditions.

Table 3.1 Dataset Summary

Black Spot	Black Measles	Leaf Rot	Healthy
			

Overall, there are four classes of grape leaves, namely leaf spots, black pox, black rot, and healthy. Each class has 2,000 images, bringing the total dataset to 8,000 images.

Table 3.2 Data Distribution

Grape Leaf Type	Training Data	Validation Data	Test Data
Black Spot	1600	200	200
Black Measles	1600	200	200
Leaf Rot	1600	200	200
Healthy	1600	200	200
Total	6400	800	800

For model training purposes, the dataset was divided into three subsets, namely training data (80%), validation data (10%), and testing data (10%). Details of the dataset division are shown in Table 3.1. The use of secondary datasets from Kaggle provides a number of advantages, including time and cost efficiency and the availability of large amounts of data. However, visual suitability analysis and reclassification are still required to ensure that the data used is appropriate for the research context, namely grapevine leaf disease detection.

3.2. Preprocessing Results

The preprocessing stage is an important first step before data is used in the analysis process or machine learning model development. In this study, preprocessing was carried out by normalizing the image dataset through image augmentation using Keras and standardizing the image size to suit the model architecture requirements. The preprocessing process includes augmentation, image resizing, and pixel normalization to improve training data quality and support model performance. See Figure 3.1 for an example.

Figure 3.1 Preposesing Results

3.2.1 Resize

The image resizing process is carried out to adjust the image size to the input layer of the model used. In this study, the images were converted to 224×224 pixels, a standard size commonly used in CNN architectures such as EfficientNet. The batch size was set to 32 so that each training iteration processed 32 images simultaneously to improve computational efficiency. The training process is carried out in two stages, namely initial training for 15 epochs by training the final layer of the model, followed by fine-tuning for 10 epochs to refine the model.

3.2.2 Augemtassi

Augmentation is used to artificially increase the variety of training data through rotation of up to 30°, 20% zoom, 10% horizontal and vertical shifts, 20% shear, and horizontal flipping. In addition, pixel values are normalized to a range of 0–1 to speed up training and improve model stability.

3.2.3 Normalization

Normalization is performed by changing the pixel value range from [0, 255] to [0, 1] using the rescale=1./255 parameter. In the training data, normalization is combined with augmentation to increase data variation and prevent overfitting, while in the validation and testing data, only normalization is performed so that the evaluation results continue to reflect the original data conditions.

3.3 Model Training Results

This study uses the EfficientNetB0 architecture with a transfer learning approach. The top part of the model is removed (include_top=False) and the base weights are frozen (trainable=False) to maintain the pre-trained results. Additional layers consisting of Global Average Pooling and Dense Layer with softmax activation were added according to the number of classes (healthy, black rot, esca, and leaf blight). The model has a total of 4,054,695 parameters, with only 5,124 trainable parameters, while the rest are pre-trained weights. This strategy makes training more efficient and reduces the risk of overfitting.

In the early stages of training (9 epochs), the accuracy was still low (training accuracy 24.4% and validation accuracy 25%) with high loss, indicating that the model was not yet optimal. After fine-tuning by opening the EfficientNetB0 weights and adjusting the learning rate, the training results improved significantly, with a training accuracy of 98.47% and a validation accuracy of 99.37%, as well as a decrease in loss (training loss of 0.503 and validation loss of 0.0331).

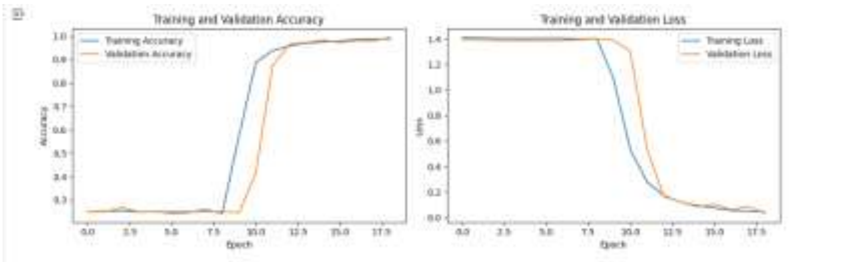


Figure 3.2 Accuracy and Loss Graph

The accuracy and loss graphs show a stable curve between the training and validation data, with no noticeable differences, indicating that the model does not suffer from overfitting. Thus, the application of transfer learning and fine-tuning on EfficientNetB0 has proven effective in improving the classification performance of grape leaf diseases to achieve near-perfect accuracy.

3.4 Model Evaluation

An evaluation was conducted to measure the performance of the grape leaf disease classification model after the training process using the EfficientNetB0 architecture with transfer learning techniques. The training process was carried out for 10 epochs, allowing the model to learn data patterns gradually.

Table 3.3 Accuracy Results

Epoch	Training Accuracy	Training Loss	Validation Accuracy	Accuracy Loss
10	0.9847	0.503	0.9937	0.0331

The high accuracy accompanied by low loss values indicates that the model is able to learn the data well and has good generalization capabilities for the validation data. There are no indications of overfitting because the difference between training and validation accuracy is very small. Next, the evaluation was carried out using test data that the model had never encountered before. This evaluation produced classification metrics in the form of precision, recall, and F1-score, and was visualized with a confusion matrix.

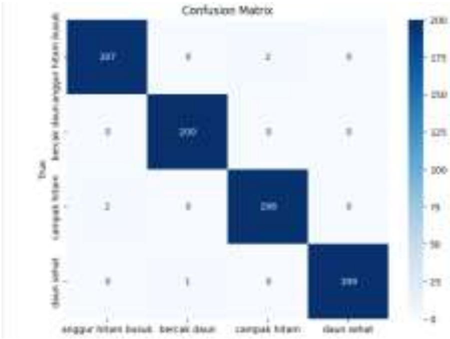


Figure 3.3 *Confusion Matrix*

Confusion matrix evaluation results

- Black rot: 197 correct, 3 incorrect → a small portion predicted as black spot.
- Leaf spot: 200 correct, no errors.
- Black spot: 198 correct, 2 incorrect → a small portion predicted as black rot.
- Healthy leaves: 199 correct, 1 incorrect → predicted as leaf spot.

The confusion matrix displays the performance of the classification model by comparing the prediction results with the actual labels.

Table 3.4 *Confision matrix data*

Label	TP	FP	FN	TN
Black Spot	197	2	2	598
Leaf Spots	200	1	0	598
Black Measles	198	2	2	597
Healthy	199	0	1	599

The confusion matrix shows the performance of the grapevine leaf disease classification model. The model is able to identify all classes with high accuracy, as indicated by the large number of True Positives and low number of errors (False Positives and False Negatives), thereby effectively distinguishing between diseased and healthy leaves.

Predicted

	precision	recall	f1-score	support
anggur hitam busuk	0.99	0.99	0.99	199
bercak daun	1.00	1.00	1.00	200
campak hitam	0.99	0.99	0.99	200
daun sehat	1.00	0.99	1.00	200
accuracy			0.99	799
macro avg	0.99	0.99	0.99	799
weighted avg	0.99	0.99	0.99	799

Figure 3.3 Prediction Results

The EfficientNetB0 model with transfer learning showed excellent performance with 99% accuracy, as well as an average precision, recall, and F1-score of 0.99. The confusion matrix showed consistent classification across all classes with minimal errors, indicating that the model was optimal for detecting grape leaf diseases. These results show that the model successfully classified 99% of the total 799 test data, with only 5 data misclassified. The errors that occurred were mostly due to the similarity of visual characteristics between disease classes.

With very high accuracy, precision, recall, and F1-score values, it can be concluded that the EfficientNetB0 transfer learning-based classification model performs very well, is able to accurately distinguish between healthy and diseased leaves, and has strong generalization capabilities. This model has great potential for implementation in an automatic grape leaf disease detection system.

3.5 Application Implementation (Flask/Streamlit)

The implementation stage was carried out by building a web-based application using the Flask framework. This application functions to classify grape leaf diseases based on images uploaded by users. The model used is EfficientNetB0 from previous training, with the main format of the app.py program running through Visual Studio Code.

Figure 3.4 *Flask Web terminal*

The entire system is run using the main app.py file, which is executed through the terminal in Visual Studio Code. The application then creates a local web server at the default address http://127.0.0.1:5000, which can be accessed through a browser to perform grape leaf disease classification directly. The main interface (index.html) is used to upload grape leaf images, while the result.html page displays the classification results. This Flask-based system utilizes the EfficientNetB0 model to classify images in real-time into four categories (Healthy, Black Rot, Black Spot, and Leaf Spot), complete with accuracy rates and uploaded images as verification of the results.

a) Initial Display on the Flask Web

Figure 3.5 *Initial Display*

The main interface page of the web-based grapevine leaf disease classification system allows users to upload leaf images to be automatically classified into four disease categories using the pre-trained EfficientNetB0 deep learning model.

b) Image Selection for detection

Figure 3.5 Image Selection

The initial display provides a "Detect Now" button to upload images from the user's device. After processing, the system displays the prediction results in the form of disease class names and confidence levels (percentages), and displays the uploaded images again so that users can verify the accuracy of the classification results.

c) Results from rotten black grapes

Figure 3.6 *Results from rotten black grapes*

The system successfully classified leaves as affected by black rot disease with a confidence level of 99.14% and an unhealthy status. Visually, this disease is characterized by dark brown to black circular spots on the leaves, often surrounded by a yellowish ring. The spots can enlarge, causing the leaves to dry out and fall off, and sometimes small black dots (fungal structures) appear in the center of the spots. Treatment recommendations: avoid excessive moisture and spray with a copper-based fungicide.

d) Results of black spot

Figure 3.7 *Results of black spot*

The prediction results show that the system successfully identified black leaf spot disease with a confidence level of 99.92%. The visual characteristics of this disease are small black spots on the leaf surface that spread unevenly, enlarge over time, and are accompanied by a change in leaf color to brown or yellow. The edges of the leaves may also dry out and curl due to tissue damage. Recommended treatments include pruning infected branches and avoiding injury during pruning.

e) Result of leaf spots

Figure 3.8 *Result of leaf spots*

The Flask-based system successfully classified leaf spot disease with 99.96% accuracy. The main characteristics are brownish to purplish spots on the leaves and brittle stems. Treatment is recommended using balanced fertilizer and systemic fungicide.

f) Results from healthy leaves

Figure 3.9 *Results from healthy leaves*

The system successfully identified healthy leaves with a confidence level of 99.63%. Healthy leaves are characterized by a uniform fresh green color without spots or damage, clear veins, and a flexible texture. It is recommended to maintain soil moisture and check the leaves regularly. The application also provides a "redetection" feature for quick reclassification.

Based on the results of testing the grape leaf disease classification model, it was found that there were classification errors in three classes. These errors indicate that the model still experiences confusion in distinguishing between several categories of leaf diseases that have visual similarities. The following is a table of misclassified data, showing the original labels (actual classes) and the model's predicted labels:

Table 3.5 Classification Error Analysis Results

Image Results	Original Label	Predicted Label
	Black Spot	Black Measles
	Black Measles	Rotten Black Grapes

		
	Healthy Leaves	Leaf Measles

4. CONCLUSION

This study successfully developed a web-based grapevine leaf disease classification system using CNN optimized with transfer learning on the EfficientNetB0 architecture, improving accuracy from 97% to 99% and F1-Score between 0.99–1.00 in four classes (healthy leaves, leaf spots, black pox, black rot) from 8,000 images. This system makes it easy for farmers to perform classification directly without technical knowledge. For further development, it is recommended to add more variety to the dataset from real field conditions to make the model more robust, develop a mobile application with a classification history feature to monitor plant development, and apply this method to other plants with similar problems, such as bananas, tomatoes, or peppers.

5. REFERENCES

Ansah, M. A., Kasih, P., Ayu, M., & Widya, D. (2022). *Identification of Grape Leaf Diseases Based on Color and Texture Features Using the Android-Based Backpropagation Method. c*, 265–271.

Juli, V. N., Ollivia, A., & Pratiwi, C. (2023). *Classification of Grape Varieties Based on Leaf Shape Using Convolutional Neural Network and K-Nearest Neighbor. 3*(2).

Pencurian, A., Indonesia, B., Perlindungan, R. U., Pribadi, D., & Indonesia, B. (n.d.). *Legal Analysis of Data Leaks in the Banking System in Indonesia (Case Study of Data Leaks at Bank Indonesia) A5*(01), 46–63.

Penyakit, K., & Jagung, T. (2022). *Classification of Corn Plant Diseases Using the Convolutional Neural Network (CNN) Method. 22*(2), 900–905. https://doi.org/10.33087/jiubj.v22i2.2065

Ratri, K. E. N., Wardani, R., & Leonardi, L. (2023). *Classification of Diseases in Grape Leaves using the Convolutional Neural Network Method. 17*(2), 112–126.

Riyanto, Y., Riana, D., Riyanto, Y., & Riana, D. (2021). *Grape leaf image disease classification using the CNN-VGG16 model. 9*(July), 218–223. https://doi.org/10.14710/jtsiskom.2021.14013

Safitri, E., Heppy, R., Sibarani, R., Kiswanto, D., Komputer, I., Medan, U. N., Baru, K., Percut, K., Tuan, S., Serdang, K. D., & Utara, S. (2024). *CLASSIFICATION OF GRAPE LEAF DISEASES BASED ON IMAGES USING THE K-NEAREST NEIGHBORS (KNN) METHOD*. *8*(6), 12633–12642.

Setiawan, M. J., Nugroho, B., & Sari, A. P. (2023). *Classification of Plant Leaf Diseases Using CNN and Random Forest Algorithms. 12*(1), 1–7.

Simanjuntak, S. S., Sinaga, H., & Telaumbanua, K. (2020). *Classification of Grape Leaf Diseases Using GLCM, Color Moment, and K * Tree Methods*. *21*(2), 93–104.

Whardana, A. K., Febriyanto, D., Katanka, M. J., Oktavia, N. A., & Desta, T. (2024). *Classification of Grape Leaf Diseases Using Convolutional Neural Networks and Transfer Learning from VGG16*.