JURNAL TEKNOLOGI DAN OPEN SOURCE
Vol. 8, No. 1, June 2025, pp. 352~361
e-ISSN: 2622-1659, accredited Four Grade by Kemenristekdikti, Decree No: 152/E/KPT/2023
DOI:10.36378/jtos.v8i1.4442

Transfer Learning Implementation with MobileNetV2 for Cassava Leaf Disease Detection

Muhammad Fathir Aulia[1], M. Khalil Gibran[2], Nur Shafwa Aulia Sitorus[3], Agung Nugroho[4], Nayla Faiza[5], Hervilla Amanda R. Siregar[6]
[1,2,3,4,5,6]Ilmu Komputer, Universitas Islam Negeri Sumatera Utara, Indonesia

Article Info

Article history:

Received 05 29, 2025
Revised 05 12, 2025
Accepted 05 27, 2025

Keywords:

Cassava Leaf
Disease Detection
Image Processing
MobileNetV2
Transfer Learning

ABSTRACT

Cassava (Manihot esculenta) is one of Indonesia's key agricultural commodities but is vulnerable to various leaf diseases, such as Cassava Bacterial Blight (CBB) and Cassava Mosaic Disease (CMD). These diseases often exhibit similar visual symptoms, making it challenging for farmers to accurately identify them through manual observation. This study aims to develop an automatic cassava leaf disease detection system based on transfer learning, utilizing the MobileNetV2 architecture. The dataset used consists of 1,500 images, evenly distributed across three categories: CBB, CMD, and healthy leaves. The data underwent preprocessing, augmentation, and model training, including fine-tuning of the last 20 layers of the MobileNetV2 model. Evaluation results indicated that the model achieved an accuracy of 67% on the test set, with the highest performance in detecting Cassava Mosaic Disease, reflected by an F1-score of 0.75. These results demonstrate the potential of MobileNetV2 as a lightweight and efficient solution for detecting cassava leaf diseases, particularly when supported by a larger and more diverse dataset. This research serves as a foundation for developing mobile-based diagnostic tools to help farmers make faster and more accurate decisions in the field.

Corresponding Author:

Muhammad Fathir Aulia
Ilmu Komputer
Universitas Islam Negeri Sumatera Utara
Medan, Indonesia
Email: muhammad0701223160@uinsu.ac.id

1. Introduction

Indonesia is known as an agricultural country with diverse and abundant agricultural products. One of the main commodities widely cultivated is cassava (Manihot esculenta)[1]. Cassava serves not only as an alternative carbohydrate source but also holds substantial economic value due to its versatility in being processed into a wide range of food products and industrial raw materials. Despite its potential, cassava cultivation continues to encounter major obstacles, especially leaf diseases that can lead to considerable decreases in crop productivity.

The three most common and damaging diseases affecting cassava plants are Cassava Bacterial Blight (CBB), Cassava Mosaic Disease (CMD), and healthy leaves as a control[2]. CBB is a bacterial disease that causes dark spots and damage to leaves, while CMD is a viral disease that causes mosaic patterns and curling of cassava leaves[2]. These three classes share similar visual symptoms, making it difficult for farmers to manually identify and diagnose the diseases.

To address this issue, this study employs transfer learning with the MobileNetV2 architecture to develop a fast and accurate automated system for detecting cassava leaf diseases[3]. MobileNetV2 was chosen for its ability to efficiently process images on devices with limited resources, making it highly suitable[4]. With this approach, it is hoped that the disease identification process can be automated, helping farmers make appropriate control decisions and improve crop productivity[5].

Digital image processing[6] technology integrated with artificial intelligence (AI) offers a promising solution for plant disease recognition. One effective AI method for image classification is the Convolutional Neural Network (CNN) [7]. However, CNN requires large training data and high computational power, which are not always available in agricultural environments. Therefore, transfer learning becomes an efficient alternative by leveraging pre-trained deep learning models on large datasets.

The MobileNetV2 model uses a lightweight architecture that combines depthwise separable convolution and inverted residuals, making it ideal for classifying cassava leaf images. With this approach, the system can recognize visual features of leaves infected with CBB, CMD, or healthy leaves. This capability is highly beneficial for making quick and accurate pest control decisions[8].

This research aims to apply transfer learning using MobileNetV2 to classify[9] cassava leaf diseases into three main classes based on leaf images. The results of this research are expected to serve as the basis for developing practical and efficient mobile-based disease detection applications, enabling farmers to monitor crop health independently and enhance national food security[10].

Table 1. Related Research Reviews[11]

Previous Research	Methods Used	Research Results
Classification of Rice Plant Diseases	HSV color extraction (Hue and Saturation values) GLCM texture extraction (Gray-Level Co-occurrence Matrix) K-Nearest Neighbor (K-NN) classification	The system is capable of classifying rice leaf diseases (Blast, Blight, and Tungro) with the highest accuracy of 75% at a K value of 3. The use of a combination of HSV and GLCM features has proven effective for early detection of plant diseases[12].
Analysis of Pre-processing Use in Transfer Learning Methods for Detecting Cassava Leaf Diseases	Transfer Learning with MobileNetV2 & ResNet50 Pre-processing: augmentation, no augmentation, and rotation	MobileNetV2 without augmentation achieved the highest accuracy of 98.64%. MobileNetV2 with rotation also provided high validation accuracy (92.50%)[13].
Application of Transfer Learning with Inception-V3 and EfficientNet-B4 in a Case Study of Cassava Leaf Disease Classification	Transfer Learning using Inception-V3 and EfficientNet-B4 architectures Image augmentation techniques (random rotation & zoom) Hyperparameter tuning (Adam and SGD optimizers with learning rate variations)	The EfficientNet-B4 model with a learning rate of 0.0001 provides the highest validation accuracy of 72.84%. This model shows balanced and stable performance, even though the training accuracy is not the highest. A learning rate that is too high actually causes overfitting[14].
Cassava Leaf Disease Detection System Using Deep Learning with MobileNetV3 Architecture Based on Android	Convolutional Neural Network (CNN) MobileNetV3 architecture Compared to EfficientNet-B7 and ConvNext-Small Deployment to Android applications (real-time detection)	MobileNetV3 produced the highest accuracy (88.78%) compared to EfficientNet-B7 (71.57%) and ConvNext-Small (81.90%). The system successfully detected cassava leaf disease in real time on Android[15].
Disease Detection System for Cassava Leaves Using Deep Learning and TensorFlow Based on Android	CNN with TensorFlow + Keras framework Data augmentation and preprocessing Stratified K-Fold Cross Validation Model conversion to TFLite for Android	The system produces a detection accuracy of 86%. The application was tested using blackbox testing and usability testing, with a user satisfaction rate of 88.3%. The Android-based implementation was deemed effective and user-friendly[16].

2. Research Method

This study aims to classify cassava leaf diseases using a transfer learning approach with the MobileNetV2 architecture. The research stages were carried out systematically, from data processing to model evaluation, as described in the following flowchart:

2.1. Load Cassava Leaf Dataset

The dataset used is a compressed file (.zip) containing images of cassava leaves categorized into three classes: Cassava Bacterial Blight, Cassava Mosaic Disease, and Healthy. This dataset was manually uploaded to Google Colab[17].

(a) (b) (c)

Figure 1. Original Images (a) bacterial_blight, (b) healthy, (c) mosaic_disease

Figure 1 shows examples of original cassava leaf images that will be used in the study.

Table 2. Dataset

Label	Kategori	Jumlah
1	Cassava_bacterial_blight	500
2	Cassava_healthy	500
3	Cassava_mosaic_disease	500
	Total	1500

Table 2 shows that there are 1,500 cassava leaf datasets divided into three classes: Cassava_bacterial_blight, Cassava_healthy, and Cassava_mosaic_disease.

2.2. Dataset Extraction & Folder Structure

After uploading, the dataset was extracted and organized into three main folders: train/, val/, and test/. Each folder represents training, validation, and testing data. The data distribution consists of 350 images for training, 100 images for validation, and 50 images for testing.

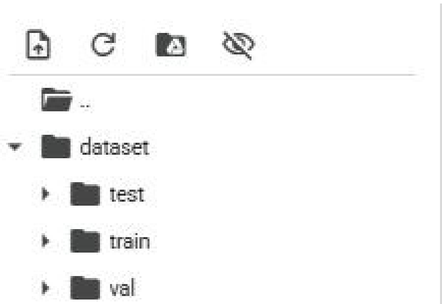

Figure 2. Results of the dataset after extraction

2.3. Data Preprocessing

The initial stage in image data processing is preprocessing[18]. All cassava leaf images in the dataset are color images with an original resolution of 256×256 pixels[19]. Before proceeding to the augmentation process, the images are resized to 224×224 pixels to match the input format required by the MobileNetV2 architecture, while also reducing the computational load during model training.

After resizing, pixel values are normalized by rescaling, which involves dividing each RGB pixel value by 255 so that the values are in the range [0, 1]. This normalization aims to stabilize the data distribution and accelerate the convergence process during training[20].

This preprocessing process is performed automatically using ImageDataGenerator from the Keras library, so there is no need for manual conversion from BGR to RGB format, which is usually required when using OpenCV.

Figure 3. Preprocessing Data Results

2.4. Data Augmentation

One of the main challenges in developing deep learning[21] models for cassava leaf disease detection is the limited amount of available data. Generally, the more training data there is, the better the model's ability to learn patterns and generate accurate predictions. However, when the available data is relatively limited, the model risks overfitting[22], a condition where the model adapts too closely to the training data, making it less capable of generalizing to new data.

To address this issue, this study employs data augmentation techniques. Augmentation is the process of transforming or manipulating existing images to automatically generate new data variations without the need for manual image addition. This technique aims to enrich the diversity of training data and enhance the model's ability to recognize cassava leaves under different conditions.

In this process, augmentation is performed after the preprocessing stage and is applied only to the training data. Various transformations are used, such as random rotation up to 30 degrees, horizontal and vertical position shifts, flipping (horizontal and vertical image reversal), zooming up to 20%, shear for perspective distortion, and brightness variations. These transformations enable the model to learn to recognize cassava leaves in various orientations and lighting conditions.

In addition to enriching data variation, augmentation also helps reduce the risk of class imbalance. Although the dataset in this study is relatively balanced, augmentation is still applied evenly across all classes to prevent the model from becoming overly reliant on patterns that may only appear in the original data.

By applying data augmentation in real-time during training using these techniques, this study successfully increased the number of training data variations while minimizing the risk of overfitting. Meanwhile, the validation and testing data only underwent normalization (rescaling) without augmentation so that the model's performance could be evaluated objectively.

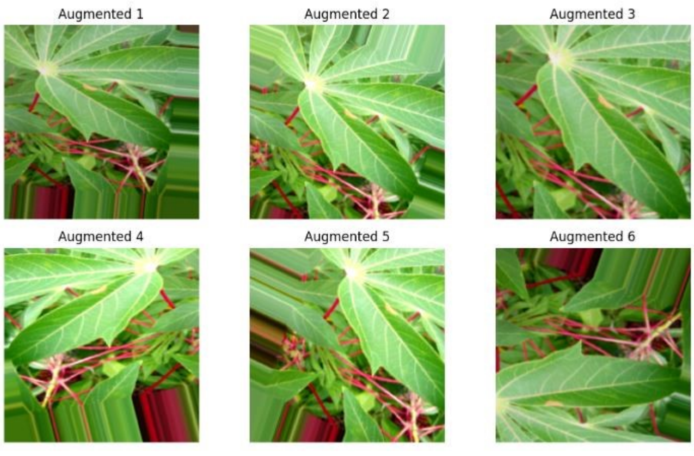

Figure 4. Data Augmentation Results

2.5. Data Split

The data was manually split into three main groups: training data, validation data, and test data, before the training process began. Each group was organized into separate folder structures to facilitate processing by the model, particularly when using Keras' ImageDataGenerator, which supports direct data loading from directories.

Proportional data distribution plays an important role in determining the performance of deep learning models. In this study, a total of 500 cassava leaf images were divided into three subsets as follows:

- Training data (train): 350 images (70%)
- Validation data (validation): 100 images (20%)
- Testing data (testing): 50 images (10%)

This proportion is designed to provide the model with sufficient data variation during training while maintaining previously unseen data for objective evaluation and validation of model performance.

Table 3. Train, Validation, and Test Data

Dataset	Jumlah
Train	350
Valid	100
Test	50

2.6. Transfer Learning with MobileNetV2

Transfer learning is applied by utilizing the MobileNetV2 model[9], which has been pre-trained using the ImageNet dataset. This model is used as a base model to extract features from cassava leaf images, thereby accelerating the training process and improving model performance with a relatively limited amount of data[23].

A. MobileNetV2 Architecture

MobileNetV2 is a lightweight and efficient CNN architecture for mobile and edge devices. This model is loaded without the top layer (include_top=False) so that a classification layer can be added as needed for the dataset.

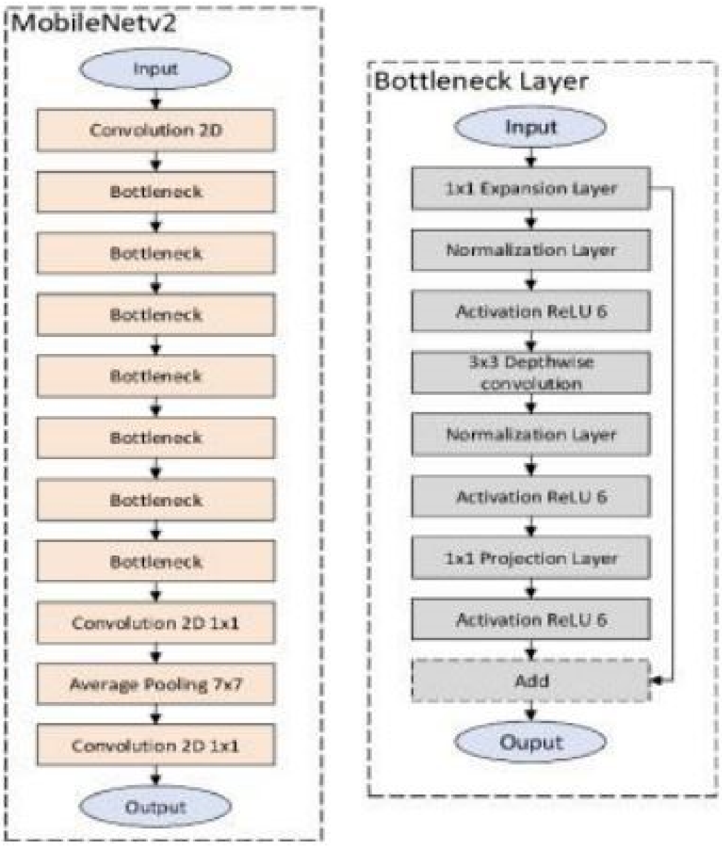

Figure 5. MobileNetV2 Architecture

Table 4 Custom Fully Connected Layer

Layer	Output Shade

Global Average Pooling	(None, 1280)
Dropout Layer_2	(None, 1280)
Dense Layer_1	(None, 128)
Dropout Layer_3	(None, 128)
Dense Layer_2	(None, 3)

B. Load Model and Fine-Tuning

The MobileNetV2 model is loaded with pretrained weights from ImageNet, without using the default top layer (fully connected layers). To adapt the model to the cassava leaf dataset, most layers in the base model are frozen, so their weights are not updated during training. However, the last 20 layers are made trainable for fine-tuning, which is retraining the end of the model to capture specific features of the cassava leaf dataset.

C. Addition of Classification Layers

After the base model, several special classification layers are added, namely:

- GlobalAveragePooling2D: This layer reduces the spatial dimensions of the feature map output to a single feature vector per filter, reducing the number of parameters and preventing overfitting.
- Dense layer with 128 neurons and ReLU activation: This layer learns more complex and non-linear feature representations.
- Dropout with a value of 0.5: Used twice to prevent the model from overfitting by randomly removing 50% of the neurons during training.
- Dense output layer with 3 neurons and softmax activation: This layer produces classification probabilities for the three cassava leaf disease classes, enabling the model to perform multi-class classification.

2.7. Model Training

The cassava leaf disease classification model was developed using the MobileNetV2 architecture, which was pre-trained on the ImageNet dataset. This model was modified by adding several specialized fully connected layers to classify three disease classes: Cassava_bacterial_blight, Cassava_healthy, and Cassava_mosaic_disease. During training, the following key hyperparameters were used:

- 30 epochs, to allow the model to learn from the data in depth.
- A batch size of 32, which is the number of samples processed in one training iteration.
- The optimizer uses the Adam algorithm with a learning rate of 1e-5 (0.00001), which provides an optimal balance between convergence speed and training stability.
- Dropout with a rate of 0.5 is applied to the fully connected layer to reduce the risk of overfitting by randomly deactivating a number of neurons during training.
- L2 regularization with a value of 0.001 is applied to the Dense layer to control the model weights so that they do not become too large, thereby improving generalization capabilities.

To improve the model's performance and resilience to data variations, data augmentation is performed on the training data. This augmentation includes image rotation, zoom, horizontal and vertical flipping, position shifting, shear, and brightness variation, making the model more resilient to changes in real-world leaf image conditions.

The model was trained with periodic evaluation on validation data every epoch to monitor the learning process and avoid overfitting.

After training was completed, the model's performance was evaluated on testing data that was not seen during training to measure the model's generalization ability in classifying cassava leaf disease images[24].

2.8. Model Evaluation

After the training process for 30 epochs was completed, an evaluation was conducted using test data to measure the performance of the classification model[25] in detecting cassava leaf disease. This evaluation was carried out using several approaches:

A. Accuracy and Loss

During training, the model showed a steady improvement in performance [26]. Accuracy and loss were recorded at each epoch to monitor progress. The final accuracy achieved on the test data was 67%, indicating that the model was able to classify cassava leaf images with moderate accuracy.

B. Confusion Matrix

A confusion matrix was utilized to analyze the distribution of the model's predictions in relation to the actual labels. This matrix provides insights into how accurately the model assigns images to their correct categories and helps identify patterns of misclassification.

C. Classification Report

A more detailed evaluation was conducted using a classification report that included precision, recall, and f1-score for each class. The results showed that the model performed best in detecting the Cassava Mosaic Disease class with an f1-score of 0.75. Meanwhile, the highest precision was obtained in the Cassava Bacterial Blight class (0.85), although the recall was lower (0.44), indicating an imbalance between correct and incorrect predictions for that class.

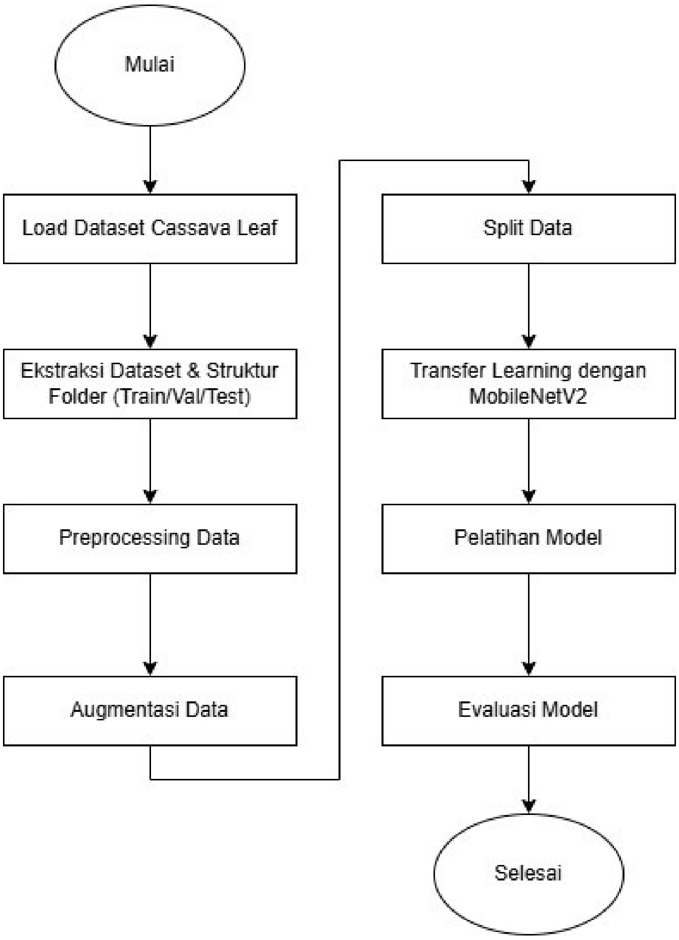

Figure 6. Research Stages

3. Result and Discussion

3.1. Model Training Results

The modified and fine-tuned MobileNetV2 model trained on the cassava leaf dataset shows a trend of improving performance over 30 epochs. The accuracy and loss plots in Figure 7 indicate that accuracy on the training and validation data increases gradually as the number of epochs increases, while the loss value continues to decrease. This suggests that the training process is proceeding smoothly without significant signs of overfitting.

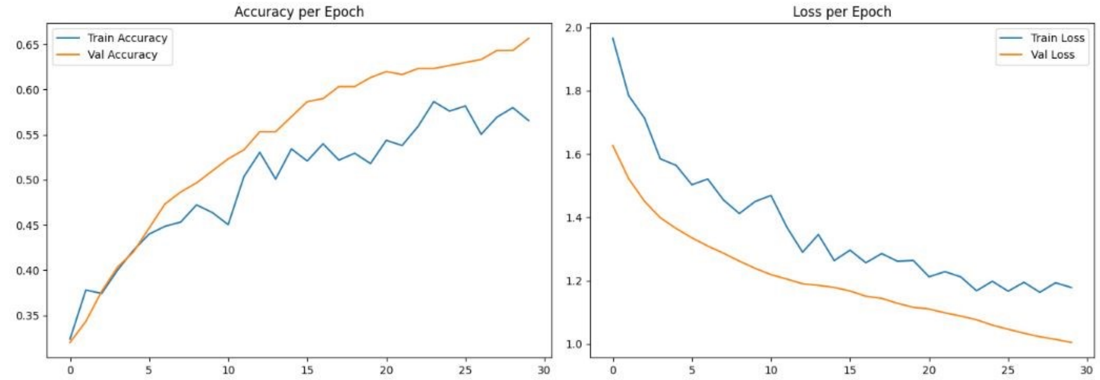

Figure 7. Accuracy and Loss Graph per Epoch

3.2. Model Evaluation

After the training process, the model was evaluated using test data to assess its classification performance. Based on the evaluation results in Figure 8, the model achieved an accuracy of 67% with a final loss value of 1.2249.

```
5/5 ━━━━━━━━━━━━━━━━━━━━ 8s 2s/step - accuracy: 0.5694 - loss: 1.2249
Test Accuracy: 0.67
```

Figure 8. Accuracy and Loss Results on Test Data

3.3. Confusion Matrix

The confusion matrix in Figure 9 shows the distribution of model predictions across each class. Out of the total 150 test data:

• Cassava bacterial blight class: 22 out of 50 data points were correctly classified.

• Cassava healthy class: performed better, with 35 data points correctly classified.

• Cassava mosaic disease class: showed the best results, with 43 out of 50 data points correctly classified.

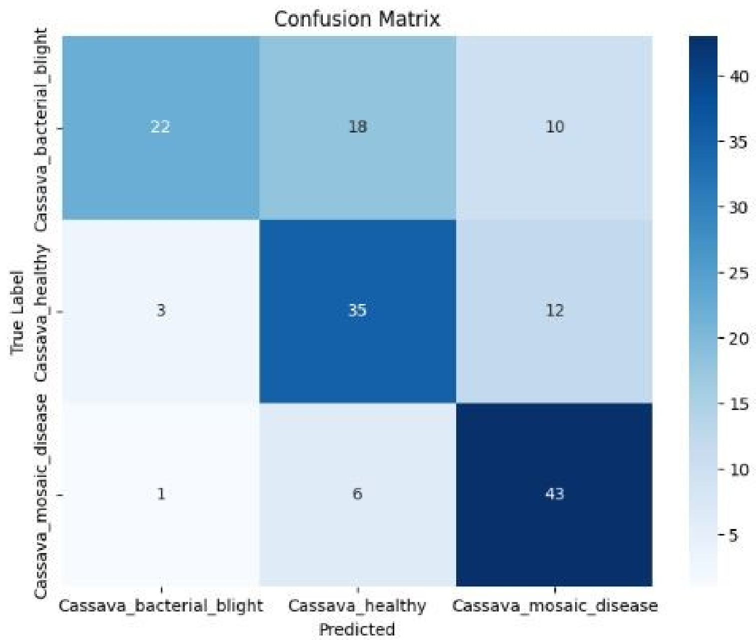

Figure 9. Confusion Matrix Classification Results

3.4. Classification Report

The classification report in Figure 10 provides detailed evaluation metrics for each class:

```
5/5 ━━━━━━━━━━━━━━━━━━━━ 10s 2s/step
Classification Report:

                          precision    recall  f1-score   support

Cassava_bacterial_blight       0.85      0.44      0.58        50
         Cassava_healthy       0.59      0.70      0.64        50
   Cassava_mosaic_disease      0.66      0.86      0.75        50
```

Figure 10. Classification Report Model

Overall, the Cassava mosaic disease class has the highest f1-score of 0.75, indicating that the model is most effective in identifying this type of disease.

3.5. Discussion

This study successfully built a disease classification model for cassava leaves using a transfer learning approach with a MobileNetV2 architecture that was fine-tuned on a specialized cassava leaf dataset. The training process was conducted over 30 epochs using the Adam optimizer, with dropout and L2 regularization applied to minimize overfitting.

From the evaluation results, the model achieved an accuracy of 67% on the test data. The best performance was achieved in the Cassava bacterial blight class in terms of precision (85%), while Cassava mosaic disease excelled in the recall metric (86%) and f1-score (0.75). These findings indicate that the model is quite effective in recognizing several types of cassava leaf diseases, although it still faces challenges in detecting all classes evenly.

The low recall value in the Cassava bacterial blight class indicates that the model often fails to detect positive cases for that class. This is likely due to the limited amount of training data and the lack of variation in the data for that class. Therefore, an increase in the amount of data and improved augmentation techniques are needed to enhance the model's generalization ability.

Overall, the use of MobileNetV2 shows good potential, but the model's performance can be significantly improved by improving the quality and quantity of training data.

4. Conclusion

The results show that the MobileNetV2 model with a transfer learning approach is capable of classifying cassava leaf diseases with a fairly adequate level of accuracy, even when using a limited dataset.

The Cassava mosaic disease class recorded the highest f1-score value, while the low recall in the Cassava bacterial blight class indicates that the model still needs to be improved in terms of generalization.

The poor performance of the model is primarily due to the limited quantity and diversity of the training data, rather than any weaknesses in the model architecture itself. Therefore, future research is recommended to increase the quantity and variety of the dataset, apply more advanced data augmentation techniques, and compare performance with other architectures such as EfficientNet or ResNet.

References

[1] A. E. Nugraha, S. Rizal, and N. K. C. Pratiwi, "Klasifikasi Penyakit Pada Tanaman Singkong Menggunakan Arsitektur VGGNET Berbasis Deep Learning," *e-Proceeding Eng.*, vol. 8, no. 6, pp. 3240–3246, 2022.

[2] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, "Deep learning for image-based cassava disease detection," *Front. Plant Sci.*, vol. 8, no. October, pp. 1–7, 2017, doi: 10.3389/fpls.2017.01852.

[3] Y. A. and S. A. M. D. Fida Zubair, Moutaz Saleh, "A Robust Ensemble Model for Plant Disease Detection Using Deep Learning Architectures," *AgriEngineering is an Int.*, vol. 7, no. 159, p. 25, 2025.

[4] S. M. Hassan, A. K. Maji, M. Jasiński, Z. Leonowicz, and E. Jasińska, "Identification of plant-leaf diseases using cnn and transfer-learning approach," *Electron.*, vol. 10, no. 12, 2021, doi: 10.3390/electronics10121388.

[5] I. G. D. Dwijayana and I. G. A. Wibawa, "Implementasi Transfer Learning Dalam Klasifikasi Penyakit Pada Daun Teh Menggunakan MobileNetV2," *J. Nas. Teknol. Inf. dan Apl.*, vol. 1, no. 1, pp. 379–387, 2022.

[6] A. Saleh, A. Ridwan, and M. K. Gibran, "Machine Learning and Fuzzy C-Means Clustering for the Identification of Tomato Diseases," *Indones. J. Comput. Sci.*, vol. 12, no. 5, pp. 2401–2413, 2023, doi: 10.33022/ijcs.v12i5.3379.

[7] F. S. Anam, M. R. Muttaqin, and Y. R. Ramadhan, "Klasifikasi Penyakit Pada Daun dan Buah Jambu Menggunakan Convolutional Neural Network," *JOINTECS (Journal Inf. Technol. Comput. Sci.*, vol. 8, no. 3, p. 115, 2023, doi: 10.31328/jointecs.v8i3.4823.

[8] J. Yao, S. N. Tran, S. Sawyer, and S. Garg, *Machine learning for leaf disease classification: data, techniques and applications*, vol. 56, no. s3. Springer Netherlands, 2023. doi: 10.1007/s10462-023-10610-4.

[9] J. Fathur Rahman, Irfansyah, Rivaldi Dwi Andhika, "WhatsApp Chat Fraud Analysis Using Support Vector

Machine Method," *J. Teknol. DAN OPEN SOURCE*, vol. 4, no. 2, pp. 174–179, 2021, doi: 10.36378/jtos.v4i2.1791.

[10] A. F. Ridwan Dwi Irawan, "Application of Deep Learning Algorithm to Detect Fraud in Online Transaction Networks Ridwan," *J. Teknol. DAN OPEN SOURCE*, vol. 7, no. 2, pp. 167–177, 2024, doi: 10.36378/jtos.v7i2.3890.

[11] R. Sari and R. Y. Hayuningtyas, "Particle Swarm Optimization-based Support Vector Machine Method for Sentiment Analysis in OVO Digital Payment Applications," *J. Teknol. Dan Open Source*, vol. 4, no. 2, pp. 232–239, 2021, doi: 10.36378/jtos.v4i2.1776.

[12] M. F. Siti Saniah, "Classification Of Rice Plant Diseases Using K-Nearest Neighbor Algorithm Based On Hue Saturation Value Color Extraction And Gray Level Co-Occurrence Matrix Features," *J. Teknol. DAN OPEN SOURCE*, vol. 7, no. 2, pp. 212–223, 2024, doi: 10.36378/jtos.v7i2.3972.

[13] A. D. P. Ariyanto, S. Hasanah, M. B. Subkhi, and N. Suciati, "Analisis Penggunaan Pra-proses pada Metode Transfer Learning untuk Mendeteksi Penyakit Daun Singkong," *Techno.Com*, vol. 22, no. 2, pp. 336–347, 2023, doi: 10.33633/tc.v22i2.7769.

[14] D. A. Tri Anton, Arief Setyanto, "Penerapan Transfer Learning Dengan Inception-V3 Dan Efficientnet-B4 Pada Studi Kasus Klasifikasi Penyakit Pada Daun Singkong," *J. Compr. Sci.*, vol. 3, no. 12, pp. 37–48, 2024.

[15] D. Firdaus, I. Sumardi, and R. R. Aziz, "Sistem Deteksi Penyakit Daun Singkong Menggunakan Deep Learning dengan Arsitektur MobilenetV3 berbasis Android," *J. Wahana Inform.*, vol. 3, no. 2, pp. 71–80, 2024.

[16] M. Faturrachman, I. Yustiana, and . S., "Sistem Pendeteksi Penyakit Pada Daun Tanaman Singkong Menggunakan Deep Learning Dan Tensorflow Berbasis Android," *IJIS - Indones. J. Inf. Syst.*, vol. 7, no. 2, p. 176, 2022, doi: 10.36549/ijis.v7i2.225.

[17] W. Shafik, A. Tufail, C. De Silva Liyanage, and R. A. A. H. M. Apong, "Using transfer learning-based plant disease classification and detection for sustainable agriculture," *BMC Plant Biol.*, vol. 24, no. 1, pp. 1–19, 2024, doi: 10.1186/s12870-024-04825-y.

[18] A. R. D. Muhammad Gabriel Somoal, "KOMPARASI MOBILENETV2 DENGAN KUSTOMISASI TRANSFER LEARNING DAN HYPERPARAMETER UNTUK IDENTIFIKASI TUMOR OTAK," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 12, no. 1, pp. 229–240, 2025, doi: 10.25126/jtiik.2025129582.

[19] M. A. and TE Ogunbiyi, AM Mustapha, EJ Eturhobore and T. Sessi, "Development of a Web-based Tomato Plant Disease Detection and Diagnosis System using Transfer Learning Techniques," *Ann. Civ. Environ. Eng.*, vol. 8, no. 1, pp. 76–86, 2024.

[20] W. B. Demilie, "Plant disease detection and classification techniques: a comparative study of the performances," *J. Big Data*, vol. 11, no. 1, 2024, doi: 10.1186/s40537-023-00863-9.

[21] A. K. Fathorazi Nur Fajri, Mohammad Dzikrillah, "Digital Fish Image Segmentation Using U-Net for Shape Feature Extraction," *J. Teknol. DAN OPEN SOURCE*, vol. 7, no. 2, pp. 195–201, 2024, doi: 10.36378/jtos.v7i2.3968.

[22] P. Kaur, S. Harnal, R. Tiwari, S. Upadhyay, S. Bhatia, and A. Mashat, "Recognition of Leaf Disease Using Hybrid Convolutional," *Sensors*, vol. 22, 2022.

[23] F. A. Arafat, M. N. Ichsan, and M. F. Pramoedya, "Pemanfaatan Arsitektur MOBILENET-CNN Untuk Mendiagnosis Penyakit Pada Daun Singkong Melalui Teknologi Citra Digital," *Pros. Semin. Nas. Teknol. DAN SAINS*, vol. 4, pp. 73–78, 2025.

[24] A. Dolatabadian, T. X. Neik, M. F. Danilevicz, S. R. Upadhyaya, J. Batley, and D. Edwards, "Image-based crop disease detection using machine learning," *Plant Pathol.*, no. September 2024, pp. 18–38, 2024, doi: 10.1111/ppa.14006.

[25] A. S. M. Khalil Gibran, "A Hybrid RBF Neural Network and FCM Clustering for Diabetes Prediction Dataset," *J. Comput. Sci. Inf. Technol. Telecommun. Eng.*, vol. 4, no. 2, pp. 395–401, 2023, doi: 10.30596/jcositte.v4i2.15573.

[26] I. Mudzakir and T. Arifin, "Klasifikasi Penggunaan Masker dengan Convolutional Neural Network Menggunakan Arsitektur MobileNetv2," *Expert J. Manaj. Sist. Inf. dan Teknol.*, vol. 12, no. 1, p. 76, 2022, doi: 10.36448/expert.v12i1.2466.