International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingénierie des Systèmes d'Information
Vol. 31, No. 3, March, 2026, pp. 763-771
Journal homepage: http://iieta.org/journals/isi

From Leaf to Data: A Hybrid CNN-KNN Strategy with Synthetic Augmentation for Early Detection of Robusta-Arabica Coffee Leaf Diseases

Apriani Apriani*, Ria Rismayati, Muhammad Zulfikri, Ivan Anggapranata, Made Arya Sutha Wijaya

Faculty of Engineering, Bumigora University, Mataram 83127, Indonesia

Corresponding Author Email: apriani@universitasbumigora.ac.id

https://doi.org/10.18280/isi.310310

Received: 28 September 2025
Revised: 15 December 2025
Accepted: 22 March 2026
Available online: 31 March 2026

Keywords:
coffee leaf disease, Convolutional Neural Network, K-Nearest Neighbor, synthetic augmentation, Generative Adversarial Network, Robusta, Arabica

ABSTRACT

Coffee is one of Indonesia's leading plantation commodities; however, its productivity is threatened by various leaf diseases such as *Cercospora*, leaf rust, leaf miner, and *Phoma*, which can reduce yields by up to 50%. Conventional disease inspection methods are still widely used but often suffer from subjectivity and delays in diagnosis. This study aims to develop an early detection model for coffee leaf diseases by integrating Convolutional Neural Networks (CNN) and K-Nearest Neighbors (KNN) with synthetic augmentation using Deep Convolutional Generative Adversarial Network (DCGAN). In this model, the CNN extracts image features and the KNN performs the final classification. Additionally, DCGAN-based augmentation is applied to balance and enrich the dataset with realistic synthetic images. The dataset consists of five classes—*Healthy*, *Cercospora*, *Leaf Rust*, *Miner*, and *Phoma*—each containing 9,000 images. The proposed model achieved a classification accuracy of 99.34%, with precision, recall, and F1-score values approaching 1.00. The 5-fold cross-validation confirmed the robustness of the model, resulting in an average accuracy of 97.63%. In conclusion, the hybrid CNN–KNN approach with DCGAN-based augmentation proved to be effective and reliable for the early detection of coffee leaf diseases, offering great potential as an artificial intelligence (AI)-based decision support tool for precision agriculture applications.

1. INTRODUCTION

Coffee is one of Indonesia's strategic plantation commodities that significantly contributes to the national economy and farmers' income [1-3]. As both a major producer and exporter, the coffee subsector not only supports the economic development of rural areas but also plays a key role in maintaining food security and market stability [4-6]. However, the productivity of Coffea arabica (Arabica) and Coffea canephora (Robusta) varieties continues to face serious challenges due to leaf diseases that directly affect yield quantity and quality [7-9]. Major coffee leaf diseases such as coffee leaf rust (Hemileia vastatrix) [10, 11], leaf spot (Cercospora coffeicola) [12, 13], and powdery mildew [14] have been reported to reduce production by up to 30–50% [15-17]. These conditions highlight the importance of developing effective early detection systems to minimize economic losses and support the sustainability of coffee agribusiness.

Traditionally, identification of coffee leaf diseases has relied on manual visual inspection in the field. Although simple and widely adopted, this method suffers from several limitations, including observer subjectivity, visual fatigue, and diagnostic inconsistency [18]. Such drawbacks often result in delayed disease control and inappropriate intervention strategies. Consequently, there is an increasing need for automated detection systems based on artificial intelligence (AI) and computer vision to ensure higher accuracy, reliability, and consistency in disease identification.

Recent advances in deep learning, particularly Convolutional Neural Networks (CNNs), have revolutionized plant disease detection due to their superior capability in extracting complex and relevant visual features from leaf images [19-29]. Prior studies have demonstrated CNN effectiveness across various architectures, including ResNet50 integrated with CenterNet [19], MobileNetV3 combined with Swin Transformer [20], and VGG-19 [25]. Several other studies have implemented CNNs in web-based systems [23] and hybrid configurations [22, 26]. In addition, the K-Nearest Neighbors (KNN) algorithm has been explored for classifying coffee diseases, producing promising results in terms of classification performance [28]. Collectively, these findings confirm the potential of CNNs for feature extraction in plant disease detection.

Nevertheless, a clear limitation remains in the final classification stage of CNN-based systems, which commonly depend on Softmax layers that are less adaptive to heterogeneous field data and variations in visual features [30, 31]. This limitation suggests the need to integrate CNNs with more flexible and non-parametric classification methods such as KNN. The KNN algorithm demonstrates advantages in handling small-scale datasets, robustness to noise and outliers, and effective recognition of local feature patterns through distance-based learning [28, 32].

Another persistent challenge in deep learning–based detection systems lies in the limited availability and diversity of labeled training data. Datasets for coffee leaf diseases are typically scarce and lack sufficient representation of real-world variations. Traditional augmentation techniques—such as rotation, flipping, and brightness adjustment—are often insufficient to emulate complex environmental conditions. To address this, the use of Generative Adversarial Networks (GANs) has emerged as a powerful alternative, capable of generating synthetic images that resemble real data distributions while providing higher intra-class variability [29].

Building on these considerations, this study proposes a hybrid CNN–KNN framework with synthetic augmentation using a GAN-based approach for the early detection of Robusta and Arabica coffee leaf diseases. The Inception-ResNet architecture serves as the CNN backbone due to its residual connections and multi-scale convolutional layers that capture rich hierarchical features. Extracted features are then classified using KNN to achieve adaptive classification performance across diverse visual patterns. Meanwhile, GAN-based augmentation is applied to balance and expand the dataset, enhancing generalization capability and mitigating overfitting risks.

The main research motivation of this study is to design an AI-based diagnostic framework that not only achieves high detection accuracy but also maintains adaptability and robustness in real agricultural environments. The primary contribution lies in developing a hybrid methodology that combines CNN's feature extraction strength with KNN's non-parametric adaptability, reinforced by GAN-based synthetic augmentation to overcome dataset limitations. This hybrid strategy is expected to yield a more accurate, resilient, and field-applicable detection model for coffee leaf diseases. Consequently, this work provides both theoretical and practical contributions to the advancement of AI-driven plant disease detection systems, supporting sustainable coffee productivity and Indonesia's competitiveness in the global market.

2. METHOD

In this study, the dataset consists of two types of datasets: one containing images of healthy coffee leaves (Healthy) and the other containing images of coffee leaves affected by diseases (NonHealthy), including *Cercospora*, *Leafrust*, *Miner*, and *Phoma* [33]. The images were collected from various Arabica and Robusta coffee leaves. Therefore, the dataset in this study is divided into several classes: *Healthy* (8,983 images), *Cercospora* (7,681 images), *Leaf Rust* (8,201 images), *Miner* (6,978 images), and *Phoma* (6,571 images) (see Figure 1). The images in this dataset do not have a balanced number of images and lack a standard size, and the contrast levels vary across the images.

This article proposes a hybrid CNN-KNN method for identifying diseases in Arabica-Robusta coffee leaves, where there are two types of datasets: one containing images of healthy coffee leaves (Healthy) and the other containing images of diseased coffee leaves (NonHealthy), including *Cercospora*, *Leafrust*, *Milner*, and *Phoma*. The workflow for the proposed hybrid system is shown in Figure 2.

Deep learning is a part of machine learning that uses layered neural models. In image processing, this approach has recently achieved strong results. In our method, a CNN learns features from coffee leaf images and a KNN classifier uses these features to decide the disease class. The method is arranged into five stages, described in the following subsections.

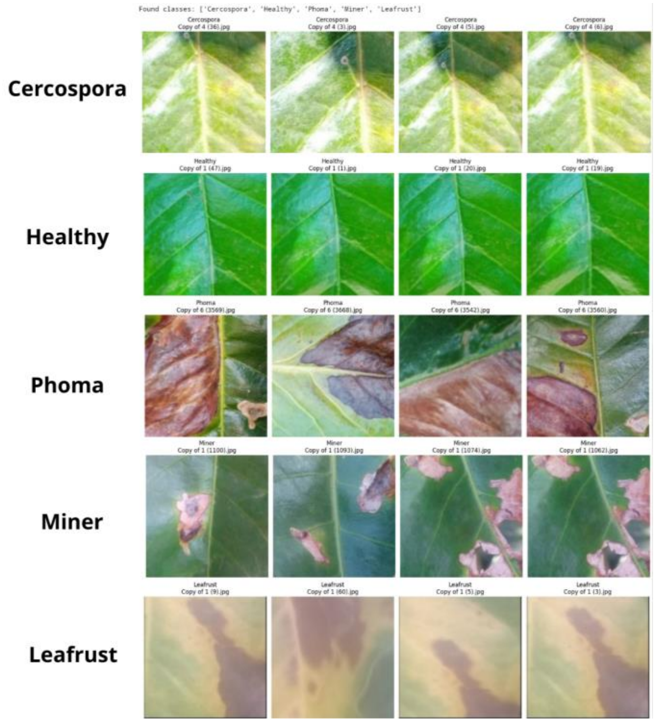

Figure 1. Arabica-Robusta coffee leaf images

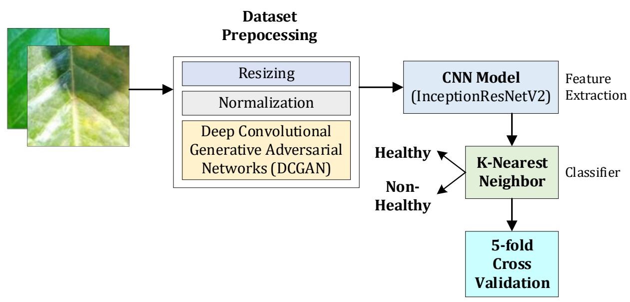

Figure 2. Working procedure of the proposed hybrid system

2.1 Preprocessing

In computer vision, preprocessing is used to improve raw images before analysis. In this work, several preprocessing steps are applied to the original coffee leaf images, as illustrated in Figure 3.

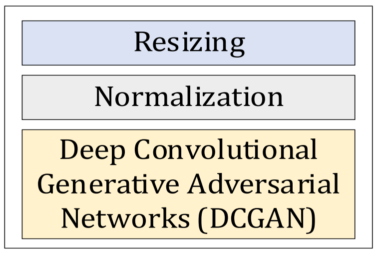

Figure 3. Preprocessing steps

2.1.1 Image resizing

The images obtained from the dataset have varying sizes. Therefore, the first step taken is to resize the images to a standardized size of 64 × 64 pixels. This resizing process ensures that the images used have consistent dimensions, which is essential for the efficiency and performance of the model in feature extraction using Inception ResNetV2 and classification using KNN [34, 35].

2.1.2 Normalization

After resizing, the images are then normalized. Normalization is performed to ensure that the pixel values are within an appropriate range, usually between 0 and 1. This is important because the Inception ResNetV2 model and KNN perform more effectively when the input data has a uniform scale. This normalization process helps accelerate convergence during training and improves the model's accuracy in both feature extraction and classification [36].

2.1.3 Data augmentation

Data augmentation aims to enrich the dataset by generating synthetic images that resemble the original images. In this study, data augmentation was performed using a Deep Convolutional Generative Adversarial Network (DCGAN) to generate synthetic coffee leaf images [37-39]. This technique enriches the limited dataset with new images that closely resemble the original ones, while still creating variations that strengthen the coffee leaf disease detection model. With this augmentation, the dataset becomes more diverse and balanced, helping to reduce the risk of overfitting and enhancing model performance, with the dataset from each class adjusted to 9,000 images.

In DCGAN, there are two main models: the Generator and the Discriminator [40]. The Generator is responsible for generating synthetic images from a latent vector (nz = 100) [41]. This vector is processed through several ConvTranspose2d layers, followed by BatchNorm2d and ReLU activation to generate images with higher resolution, 128 × 128 pixels. The Generator attempts to create synthetic images that closely resemble the original coffee leaf images. The architecture used is summarized in Table 1. On the other hand, the Discriminator is responsible for distinguishing between real and synthetic images generated by the Generator. The Discriminator uses several Conv2d layers equipped with LeakyReLU to produce an output probability that indicates whether the image is real or fake, and the architecture used is summarized in Table 2.

Table 1. Generator architecture

Layer	Operation	Parameters (out_channels, kernel_size, Stride, Padding)	Output Size (C × H × W)	Activation / Normalisation
ConvTranspose2d	Deconvolution	(1024, 4, 1, 0)	1024 × 4 × 4	BatchNorm2d + ReLU
ConvTranspose2d	Deconvolution	(512, 4, 2, 1)	512 × 8 × 8	BatchNorm2d + ReLU
ConvTranspose2d	Deconvolution	(256, 4, 2, 1)	256 × 16 × 16	BatchNorm2d + ReLU
ConvTranspose2d	Deconvolution	(128, 4, 2, 1)	128 × 32 × 32	BatchNorm2d + ReLU
ConvTranspose2d	Deconvolution	(64, 4, 2, 1)	64 × 64 × 64	BatchNorm2d + ReLU
ConvTranspose2d	Deconvolution	(3, 4, 2, 1)	3 × 128 × 128	Tanh

Table 2. Discriminator architecture

Layers	Operation	Parameters (out_channels, kernel_size, Stride, Padding)	Output Size (C × H × W)	Activation / Normalisation
Conv2d	Convolution	(64, 4, 2, 1)	64 × 64 × 64	LeakyReLU(0.2)
Conv2d	Convolution	(128, 4, 2, 1)	128 × 32 × 32	BatchNorm2d + LeakyReLU(0.2)
Conv2d	Convolution	(256, 4, 2, 1)	256 × 16 × 16	BatchNorm2d + LeakyReLU(0.2)
Conv2d	Convolution	(512, 4, 2, 1)	512 × 8 × 8	BatchNorm2d + LeakyReLU(0.2)
Conv2d	Convolution	(1024, 4, 2, 1)	1024 × 4 × 4	BatchNorm2d + LeakyReLU(0.2)
Conv2d	Convolution	(1, 4, 1, 0)	1 × 1 × 1	Sigmoid

Table 3. Hyperparameters for Deep Convolutional Generative Adversarial Network (DCGAN)

Parameter	Value	Description
nz	100	Latent vector size for Generator
ngf	64	Feature map size for Generator
ndf	64	Feature map size for Discriminator
num_epochs	300	Number of training epochs
Lr	0.0002	Learning rate for optimizer
beta1	0.5	First momentum parameter for Adam optimizer
Device	CUDA/CPU	Device used (GPU/CPU)

Some important parameters used in this model are nz = 100, which determines the size of the latent vector for the Generator. ngf = 64 and ndf = 64 are the feature map sizes for the Generator and Discriminator, which allow the model to capture various features from the coffee leaf images. Training is performed over 300 epochs, with a learning rate (lr) = 0.0002 and beta1 = 0.5 for momentum parameters in the Adam optimizer. The optimizer is used to update the model weights during training, and the device is used to determine whether a GPU or CPU is used for training. The hyperparameters used are summarized in Table 3.

Table 4. Optimizer

Parameter	Value	Description
Optimizer	Adam	Optimizer used for model training
Lr (Learning Rate)	0.0002	Learning rate used by Adam
Beta1	0.5	First momentum parameter for Adam
Betas	(0.5, 0.999)	Second momentum parameter for Adam

The optimizer used is Adam, which is very popular in deep learning model training, including GANs. Adam combines the advantages of AdaGrad and RMSProp, making it highly effective at handling very large parameters and highly variable data. The learning rate used is 0.0002, with beta1 = 0.5 for the first momentum parameter and betas = (0.5, 0.999) for the second parameter. This optimizer is responsible for updating the Generator and Discriminator weights efficiently, enabling them to learn better during training. The optimizer used is summarized in Table 4.

Unlike cGAN (which conditions on class labels), our DCGAN setup is unconditional; therefore, to achieve per-class balancing, we trained a separate DCGAN for each class (five models in total) and curated the generated images prior to merging into the 9,000-per-class dataset.

2.2 Proposed Convolutional Neural Networks architecture

The proposed CNN architecture for detecting diseases in coffee leaves uses a hybrid CNN-KNN model. This CNN architecture consists of several layers, including convolutional layers, max-pooling, dropout, fully connected, and softmax classifier layers. This model uses Inception ResNetV2 as part of feature extraction, which has proven effective in capturing more complex visual features in images. In each convolutional layer, filters of sizes 5 × 5 and 3 × 3 are used to identify basic features such as edges, corners, and textures in the coffee leaf images.

Each convolutional layer in the CNN generates feature maps that summarize important information for classification [42]. These features are then enhanced using Inception ResNetV2, which has the ability to combine feature extraction results from multiple pathways within a single architecture, enriching the feature representation. The main advantage of Inception ResNetV2 is its ability to handle image variability and ensure the model can recognize various types of patterns within limited data.

In the first convolutional layer, a 5 × 5 kernel with 32 filters is applied to the input image, generating feature maps. These filters will learn local features such as edges, corners, and textures. The formula for the output feature map generated by the convolutional layer is [43]:

$$M_L(x) = B_L(x) + \sum_{y=1}^{N_{L-1}} F_L(x,y) * M_{L-1}(y) \tag{1}$$

where, $M_L(x)$ is the output feature map, L is the layer, $F_L(x,y)$ is the filter, N_{L-1} is the number of filters, $B_L(x)$ is the bias, and $M_{L-1}(y)$ is the input map. These feature maps store detailed information about the image features. In the first convolutional layer, the input image of size 64 × 64 is processed with 32 filters. After each convolutional layer, batch normalization and ReLU activation layers are applied.

Max-pooling layers with a 2 × 2 kernel are applied after each convolutional block to reduce the spatial dimensions of the feature maps generated by the convolutional layers. This also reduces the number of parameters and computational cost, while summarizing features from portions of the generated feature maps.

The second convolutional layer applies 64 filters of size 3 × 3 to the output of the first max-pooling layer. Two deeper convolutional layers also use 3 × 3 kernels, with 80 filters in the third layer and 192 in the fourth. Details of kernel sizes and filter numbers are listed in Table 5.

The fully connected layers consist of two hidden layers, each with 520 neurons, and two neurons in the final layer to classify whether the coffee leaf is infected or not. A dropout layer with a rate of 0.5 is applied between the two fully connected layers for regularization of the model. Finally, the softmax layer is used to predict the class, providing probability-based predictions for each class, as summarized in Table 6.

Table 5. Filter specification for convolutional layers

Convolutional Layer	Number of Filters	Filter Size
conv2d_1	32	5 × 5
conv2d_2	64	3 × 3
conv2d_3	80	3 × 3
conv2d_4	192	3 × 3

Table 6. Total number of parameters

Layer	Output Shape	Parameter
Input Layer	(None, None, 3)	0
Conv2D (conv2d)	(None, None, 32)	864
BatchNorm	(None, None, 32)	96
Activation (ReLU)	(None, None, 32)	0
Conv2D (conv2d_1)	(None, None, 32)	9,216
BatchNorm	(None, None, 32)	96
Conv2D (conv2d_2)	(None, None, 64)	18,432
MaxPooling2D	(None, None, 64)	0
Conv2D (conv2d_3)	(None, None, 80)	5,12
Fully Connected (fc_1)	(None, 520)	1,398,280
Output Layer (fc_2)	(None, 2)	1,042
Total Parameters		21,802,784
Trainable params		21,768,352
Non-trainable param		34,432

After training, features for coffee leaf disease recognition are taken from the CNN. Early layers represent simple patterns, while deeper layers capture more complex structures. The activation vector from the last convolutional layer is used as the feature descriptor and is given to the KNN classifier to predict the class.

2.3 Parameters for Convolutional Neural Networks model

Convolutional and fully connected layers contain trainable weights and biases. These values form the parameters of the CNN model and are updated during training [44]. For a convolutional layer, the total number of parameters P_{conv} is computed as:

$$P_{conv} = F_h \times F_W \times C_{in} \times F_{num} \tag{2}$$

where, F_h and F_w are the height and width of the filter, F_{num} is the number of filters, and C_{in} is the number of input channels for the layer.

The parameters for the fully connected layer P_{FC} can be calculated as:

$$P_{FC} = A_{prev} \times N_{unit} \times N_{unit} \tag{3}$$

where, A_{prev} is the dimension of the activation from the previous layer, and N_{unit} is the number of units or neurons in the current fully connected layer. Max-pooling layers do not have trainable parameters. The parameters for the batch-normalization layer are calculated by multiplying the number of channels used in the previous convolutional layer by four.

In total, this model has 21,802,784 parameters, with

21,768,352 trainable parameters and 34,432 non-trainable parameters (mainly for batch normalization).

2.4 K-Nearest Neighbors classifier

In the KNN classifier, the Euclidean distance method is used to calculate the distance between two points in the feature space [45, 46]. The Euclidean distance formula can be written as follows:

$$R = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2} \tag{4}$$

where, R is the Euclidean distance between two points, and $(x_i - y_i)$ represents the difference in features between the two points. Here, i = 1, 2, 3, ..., N, where N is the number of features in the dataset. The choice of K, which is the number of nearest neighbors, is critical for determining classification accuracy. In this study, K is set to 3.

The KNN classifier calculates the distance between training features and testing features. The class label is then assigned based on the majority vote from the nearest neighbors [47]. The majority vote rule can be expressed mathematically as follows:

$$M.R = \text{majority rule}, L = \text{class label}, b_i = \text{class label of the } i - \text{the nearest neighbor} \tag{5}$$

Here, the function I returns a binary value (True or False) indicating whether a specific condition is met. The main advantage of KNN over more complex classifiers like CNN is its simplicity and effectiveness, as it does not require training time.

In this study, high-level features extracted from the fourth convolutional layer of the CNN are used as input to the KNN classifier. This classifier uses only the most relevant features to classify the coffee leaf images. The features from each test image are compared with the features from training images, and classification is performed based on the nearest neighbors.

2.5 The 5-fold cross-validation method

After performing classification with KNN, the model is validated using 5-fold cross-validation. In this method, the dataset is divided into five separate parts. Each part is alternately used as testing data, while the other four parts are used as training data [43, 48]. This process is repeated five times, so each data point is used as test data once and as training data four times. The results of the five tests are then averaged to provide more accurate results and reduce variance in the model. With 5-fold cross-validation, the model is thoroughly tested on each part of the dataset, providing a better overall picture of the model's performance. This method helps reduce overfitting and provides a more stable estimate of the model's ability to classify new data.

3. RESULT AND DISCUSSION

3.1 The impact of data augmentation using Deep Convolutional Generative Adversarial Network on model performance

Data augmentation using a DCGAN improved the limited dataset by synthesizing realistic, non-identical images. The procedure expanded each coffee-leaf disease class to 9,000 samples, increasing intra-class variability and restoring class balance, which in turn reduced overfitting and strengthened feature separability. Consequently, DCGAN-based augmentation contributed substantially to the observed performance gains by exposing the CNN to a richer and more diverse training distribution. Representative synthetic examples are shown in Figure 4.

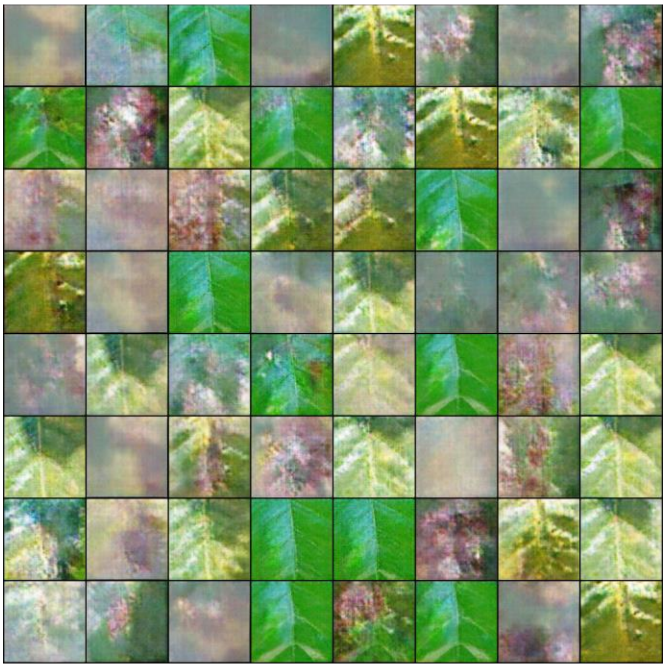

Figure 4. Images generated by Deep Convolutional Generative Adversarial Network

3.2 Evaluation of the proposed Convolutional Neural Networks architecture

The proposed CNN architecture consists of 23 layers, including convolutional layers, max-pooling, dropout, fully connected, and softmax classifier layers. Each convolutional layer uses filters of sizes 5 × 5 and 3 × 3, with varying numbers of filters: 32, 64, 80, and 192, designed to extract important visual features from the coffee leaf images. This model is trained with input images of size 128 × 128 and has a total of 21,802,784 parameters, with 21,768,352 of them being trainable parameters.

```
[1.10934234e+00 3.42922807e-01 0.00000000e+00 3.33312559e+00
 2.75575590e+00 1.79900169e+00 2.43580055e+00 1.41392982e+00
 1.61267471e+00 0.00000000e+00 4.89569128e-01 1.56378841e+00
 3.18965524e-01 1.06609964e+00 3.10471743e-01 2.71857738e+00
 4.23849761e-01 1.93348810e-01 1.53098106e+00 8.29758167e-01
 5.88541555e+00 9.81568694e-01 2.62173724e+00 0.00000000e+00
 1.24163747e+00 3.46653461e-01 9.76929665e-02 1.67841601e+00
 0.00000000e+00 3.79982328e+00 3.35842490e-01 2.61270928e+00
 2.44424534e+00 1.30488026e+00 4.28514421e-01 1.16072094e+00
 4.27901363e+00 7.10200548e-01 4.79500484e+00 8.27078938e-01
 5.56556702e-01 5.30604482e-01 3.54689980e+00 0.00000000e+00
 2.39699268e+00 0.00000000e+00 2.12451720e+00 0.00000000e+00
```

Figure 5. Feature extraction results using Inception-ResNetV2

The evaluation results show that this model successfully captures relevant visual characteristics from coffee leaf images, as shown in Figure 5. With its deep architectural structure, this model is capable of effective feature extraction, contributing to improved performance in classifying various types of coffee leaf diseases.

3.3 Model performance with K-Nearest Neighbors

After performing feature extraction using CNN, KNN is used to improve classification accuracy. KNN is implemented to reduce the bias in results that might occur due to the use of the softmax activation function in CNN, which tends to bias the results towards the training data. Evaluation of the KNN model shows that its performance is very good, with very high precision, recall, and F1-score for each class. For the *Cercospora*, *Healthy*, *Leafrust*, *Miner*, and *Phoma* classes, precision and recall each reached 100% for most classes, with F1-scores approaching 1.00. This shows that using KNN after CNN successfully improved classification result stability and provided more accurate results in detecting coffee leaf diseases. This evaluation result is clearly shown in Figure 6, which presents the precision, recall, and F1-score values for each class, along with the average and weighted values of the obtained results. Additionally, the KNN evaluation also shows an overall accuracy of 0.9802 (or 98.02%), which indicates that this model performs very well in classifying coffee leaf diseases.

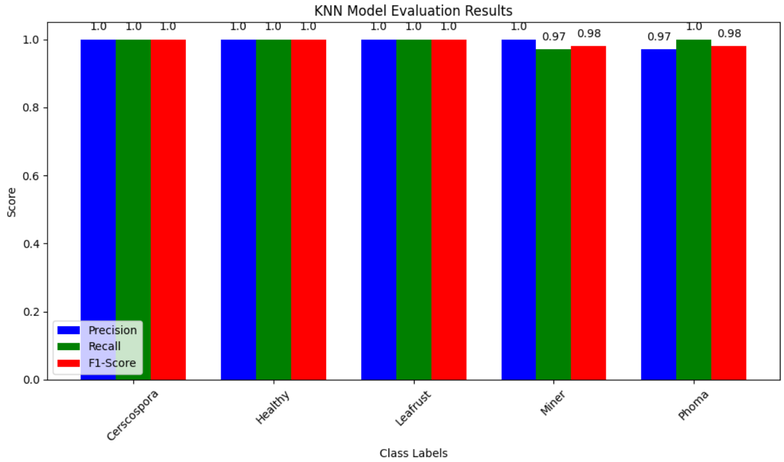

Figure 6. K-Nearest Neighbors (KNN) model evaluation results

In addition, the evaluation was carried out using a confusion matrix. The results of the confusion matrix can be seen in Figure 7.

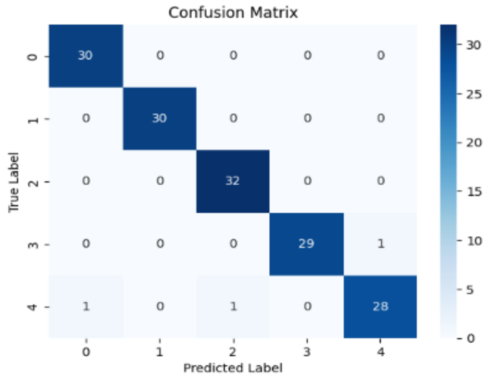

Figure 7. Confusion matrix

3.4 Performance evaluation under 5-fold cross-validation

The 5-fold cross-validation is used to evaluate the proposed CNN model and shows that its performance is stable across the folds. The accuracy obtained for each fold is as follows: Fold 1–98.03%, Fold 2–98.68%, Fold 3–96.71%, Fold 4–96.71%, and Fold 5–98.01%. The average accuracy from the 5-fold cross-validation is 97.63%, which shows that the model is reliable and produces very good accuracy across all folds. This process strengthens the validity of the proposed CNN model, which not only provides high accuracy but also ensures stable performance in detecting coffee leaf diseases across different test data. The results of the 5-fold cross-validation evaluation are presented in detail in Table 7, which shows the accuracy for each fold and the average accuracy obtained.

Table 7. The 5-fold cross-validation results

Fold	Accuracy (%)
Fold 1	98.03
Fold 2	98.68
Fold 3	96.71
Fold 4	96.71
Fold 5	98.01
Average	97.63

In addition, there is also a standard deviation = 0.005 and a confidence interval = (0.970, 0.980) with k-fold = 5. These results show that the resulting model is very accurate, stable, and has good generalization.

3.5 Discussion

The hybrid approach that combines CNN-based feature extraction with distance-based decision rules through KNN, accompanied by data rebalancing using a DCGAN, exhibits consistent performance across five coffee leaf disease classes. Methodologically, the DCGAN enriches sample diversity and normalizes inter-class distributions—bringing each class to approximately 9,000 images—thereby reducing minority-class bias and sharpening separability in the learned representations. Performance stability is further evidenced by

5-fold cross-validation (mean accuracy 97.63%; standard deviation 0.005; confidence interval 0.970–0.980), indicating low variability across folds.

Notwithstanding these strengths, several limitations warrant consideration. Reliance on synthetic images may introduce artifacts that are not fully representative of field conditions, and the evaluation remains centered on a single data source, leaving applicability across locations, cultivars, imaging devices, and illumination regimes insufficiently characterized. In addition, the architecture's size (≈ 21.8 million parameters) imposes nontrivial computational demands for deployment on resource-constrained edge devices. From an implementation perspective, a standardized image-acquisition protocol and targeted optimization of the inference pipeline are necessary to preserve observed performance under real-world use.

4. CONCLUSION

This study confirms the effectiveness of a hybrid CNN–KNN approach with DCGAN-based synthetic augmentation for detecting five coffee leaf disease classes; however, the findings should be interpreted alongside two principal limitations, namely reliance on synthetic images and an evaluation scope confined to a single data source. Accordingly, future work will prioritize external validation across diverse locations and devices and expansion/diversification of real-world data to strengthen generalization, together with computational optimization for edge deployment, exploration of stronger CNN architectures/pretraining, additional augmentation techniques, and alternative classifiers (e.g., SVM, Random Forest) to enhance stability and deployment readiness.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to the Ministry of Higher Education, Science, and Technology of the Republic of Indonesia, through the Directorate of Research, Technology, and Community Service, for funding this research under the Fundamental Grant scheme for the 2025 fiscal year (Contract No.: 0070/C3/AL.04/2025). dated 23 May 2025 concerning the Recipients of the Operational Assistance Program for State Universities for Research and Community Service Programs, Fiscal Year 2025. We also extend our appreciation to the Institute for Research and Community Service (LPPM) of Bumigora University for facilitating and supporting the implementation of this study. Finally, we thank all resource persons and parties who contributed to the successful completion of this research.

REFERENCES

[1] Darmanto, E.B., Suhartono, S., Pratiwi, Y.S., Adenan, M. (2025). Competitiveness of Indonesian coffee commodities in global market. EKOMBIS Review: Jurnal Ilmiah Ekonomi dan Bisnis, 13(2): 1219-1224. https://doi.org/10.37676/ekombis.v13i2.7293

[2] Purwawangsa, H., Irfany, M.I., Haq, D.A. (2024). Indonesian coffee exports' competitiveness and determinants. Journal of Management Agribisnis, 21(1): 59-71. http://doi.org/10.17358/jma.21.1.59

[3] Harmiansyah, D., Diptaningsari, D., Wardani, N., Meidaliyantisyah, M., Mawardi, R., Hendra, J. (2023). Intensity of leaf rust disease on four Robusta Coffee clones in Natar, South Lampung. IOP Conference Series: Earth and Environmental Science, 1230: 012097. https://doi.org/10.1088/1755-1315/1230/1/012097

[4] As-Sadili, A.H., Syaukat, Y., Falatehan, F. (2023). Pendapatan dan kerentanan petani Kopi Robusta di sekitar kawasan Taman Nasional Bukit Barisan Selatan. Jurnal Agribisnis Indonesia, 11(2): 220-235. https://doi.org/10.29244/jai.2023.11.2.220-235

[5] Irawan, A. (2025). The smallholder coffee farmer's livelihood adaptation strategies in Bengkulu, Indonesia. Journal of Strategy and Management, 18(1): 7395. https://doi.org/10.1108/JSMA-04-2023-0082

[6] Ashardiono, F., Trihartono, A. (2024). Optimizing the potential of Indonesian coffee: A dual market approach. Cogent Social Sciences, 10(1). https://doi.org/10.1080/23311886.2024.2340206

[7] Koutouleas, A., Collinge, D.B., Boa, E. (2023). The coffee leaf rust pandemic: An ever-present danger to coffee production. Plant Pathology, 73(3): 522-534. https://doi.org/10.1111/ppa.13846

[8] Peck, L.D., Boa, E. (2023). Coffee wilt disease: The forgotten threat to coffee. Plant Pathology, 73(3): 506-521. https://doi.org/10.1111/ppa.13833

[9] Ayalew, B., Hylander, K., Adugna, G., Zewdie, B., Zignol, F., Tack, A.J.M. (2024). Impact of climate and management on coffee berry disease and yield in coffee's native range. Basic and Applied Ecology, 76: 25-34. https://doi.org/10.1016/j.baae.2024.01.006

[10] Aristizábal, L.F. (2024). Achievements and challenges in controlling coffee leaf rust (*Hemileia vastatrix*) in Hawaii. Agrochemicals, 3(2): 147-163. https://doi.org/10.3390/agrochemicals3020011

[11] Julca-Otiniano, A., Alvarado-Huamán, L., Castro-Cepero, V., Borjas-Ventura, R., et al. (2024). New races of *Hemileia vastatrix* detected in Peruvian coffee fields. Agronomy, 14(8): 1811. https://doi.org/10.3390/agronomy14081811

[12] Juárez-Sánchez, J.P., Ramírez-Valverde, B., Ramírez-Suárez, J.G. (2024). Pests and diseases in coffee (Coffea arabica L.) production in two municipalities of the State of Puebla. Agro Productividad, 17(12): 153-159. https://doi.org/10.32854/agrop.v17i12.3192

[13] Nam, H.S., Park, H.S., Kim, Y.C. (2023). First report of coffee leaf spot caused by *Curvularia geniculata*. Journal of Phytopathology, 172(1): e13245. https://doi.org/10.1111/jph.13245

[14] Sidauruk, A., Suseno, P., Satria, B., Sulistiyono, M. (2024). Diagnosis penyakit tanaman Kopi Robusta menggunakan metode dempster shafer berbasis sistem pakar. Indonesian Journal of Computer Science, 13(4): 6020-6030. https://doi.org/10.33022/ijcs.v13i4.3953

[15] Koutouleas, A. (2023). Coffee leaf rust: Wreaking havoc in coffee production areas across the tropics. Plant Health Cases. https://doi.org/10.1079/planthealthcases.2023.0005

[16] Rahmah, D.M., Purnomo, D., Filianty, F., Ardiansah, I., Pramulya, R., Noguchi, R. (2023). Social life cycle assessment of a coffee production management system in a rural area: A regional evaluation of the coffee industry in West Java, Indonesia. Sustainability, 15(18): 13834. https://doi.org/10.3390/su151813834

[17] Revadiana, R.A., Trimo, L. (2021). Determining factors for coffee business success (case study in SML company, West Java Province). Jurnal Ekonomi Pertanian dan Agribisnis, 5(1): 016-026. https://doi.org/10.21776/ub.jepa.2021.005.01.02

[18] Salamai, A.A. (2024). Towards automated, efficient, and interpretable diagnosis coffee leaf disease: A dual-path visual transformer network. Expert Systems with Applications, 255(Part A): 124490. https://doi.org/10.1016/j.eswa.2024.124490

[19] Nawaz, M., Nazir, T., Javed, A., Amin, S.T., Jeribi, F., Tahir, A. (2024). CoffeeNet: A deep learning approach for coffee plant leaves diseases recognition. Expert Systems with Applications, 237(Part A): 121481. https://doi.org/10.1016/j.eswa.2023.121481

[20] Faisal, M., Leu, J.S., Darmawan, J.T. (2023). Model selection of hybrid feature fusion for coffee leaf disease classification. IEEE Access, 11: 62281-62291. https://doi.org/10.1109/ACCESS.2023.3286935

[21] Milke, E.B., Gebiremariam, M.T., Salau, A.O. (2023). Development of a coffee wilt disease identification model using deep learning. Informatics in Medicine Unlocked, 42: 101344. https://doi.org/10.1016/j.imu.2023.101344

[22] Pham, T.C., Nguyen, V.D., Le, C.H., Packianather, M., Hoang, V.D. (2023). Artificial intelligence-based solutions for coffee leaf disease classification. IOP Conference Series: Earth and Environmental Science, 1278: 012004. https://doi.org/10.1088/1755-1315/1278/1/012004

[23] Aufar, Y., Abdillah, M.H., Romadoni, J. (2023). Web-based CNN application for Arabica coffee leaf disease prediction in smart agriculture. Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 7(1): 71-79. https://doi.org/10.29207/resti.v7i1.4622

[24] Hitimana, E., Sinayobye, O.J., Ufitinema, J.C., Mukamugema, J., et al. (2023). An intelligent system-based coffee plant leaf disease recognition using deep learning techniques on Rwandan arabica dataset. Technologies, 11(5): 116. https://doi.org/10.3390/technologies11050116

[25] Sucia, D., Larasabi, A.T.S., Azhar, Y., Sari, Z. (2023). Classification of coffee leaf diseases using CNN. Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control, 8(3): 673-682. https://doi.org/10.22219/kinetik.v8i3.1745

[26] Singh, M.K., Kumar, A. (2024). Coffee leaf disease classification by using a hybrid deep convolution neural network. SN Computer Science, 5: 618. https://doi.org/10.1007/s42979-024-02960-9

[27] Abuhayi, B.M., Mossa, A.A. (2023). Coffee disease classification using Convolutional Neural Network based on feature concatenation. Informatics in Medicine Unlocked, 39: 101245. https://doi.org/10.1016/j.imu.2023.101245

[28] de Oliveira Aparecido, L.E., Lorençone, P.A., Lorençone, J.A., Torsoni, G.B., Lima, R.F., Padilha, F., Souza, P.S., Rolim, G.S. (2024). Addressing coffee crop diseases: Forecasting Phoma leaf spot with machine learning. Theoretical and Applied Climatology, 155: 2261-2282. https://doi.org/10.1007/s00704-023-04739-z

[29] Somanna, H.P., Stynes, P., Muntean, C.H. (2024). A deep learning-based plant disease detection and classification for arabica coffee leaves. In the 5th International Conference on Deep Learning Theory and Applications (DeLTA 2024), Dijon, France, pp. 19-37. https://doi.org/10.1007/978-3-031-66694-0_2

[30] Salehi, A.W., Khan, S., Gupta, G., Alabduallah, B.I., Almjally, A., Alsolai, H., Siddiqui, T., Mellit, A. (2023). A study of CNN and transfer learning in medical imaging: Advantages, challenges, future scope. Sustainability, 15(7): 5930. https://doi.org/10.3390/su15075930

[31] Barburiceanu, S., Meza, S., Orza, B., Malutan, R., Terebes, R. (2021). Convolutional neural networks for texture feature extraction. Applications to leaf disease classification in precision agriculture. IEEE Access, 9: 160085-160103. https://doi.org/10.1109/ACCESS.2021.3131002

[32] Shanjida, S., Islam, M.S., Mohiuddin, M. (2022). MRI-image based brain tumor detection and classification using CNN-KNN. In 2022 IEEE IAS Global Conference on Emerging Technologies (GlobConET), Arad, Romania, pp. 900-905. https://doi.org/10.1109/GlobConET53749.2022.9872168

[33] Fragoso, J., Silva, C., Paixão, T., Alvarez, A.B., Castro Júnior, O., Florez, R., Palomino-Quispe, F., Savian, L.G., Trazzi, P.A. (2025). Coffee-leaf diseases and pests detection based on YOLO models. Applied Sciences, 15(9): 5040. https://doi.org/10.3390/app15095040

[34] Pardede, J., Purohita, A.S. (2023). The advantage of transfer learning with pre-trained model in CNN towards CT-scan classification. Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika, 9(2): 155-161. https://doi.org/10.23917/khif.v9i2.19872

[35] Benbrahim, H., Behloul, A. (2021). Fine-tuned Xception for image classification on Tiny ImageNet. In 2021 International Conference on Artificial Intelligence for Cyber Security Systems and Privacy (AI-CSP), El Oued, Algeria, pp. 1-4. https://doi.org/10.1109/AI-CSP52968.2021.9671150

[36] Dhaka, V.S., Meena, S.V., Rani, G., Sinwar, D., Kavita, K., Ijaz, M.F., Woźniak, M. (2021). A Survey of deep convolutional neural networks applied for prediction of plant leaf diseases. Sensors, 21(14): 4749. https://doi.org/10.3390/s21144749

[37] Motamed, S., Rogalla, P., Khalvati, F. (2021). Data augmentation using Generative Adversarial Networks (GANs) for GAN-based detection of Pneumonia and COVID-19 in chest X-ray images. Informatics in Medicine Unlocked, 27: 100779. https://doi.org/10.1016/j.imu.2021.100779

[38] Lim, W., Yong, K.S.C., Lau, B.T., Tan, C.C.L. (2024). Future of generative adversarial networks (GAN) for anomaly detection in network security: A review. Computers & Security, 139: 103733. https://doi.org/10.1016/j.cose.2024.103733

[39] Lu, Y.Z., Chen, D., Olaniyi, E., Huang, Y.B. (2022). Generative adversarial networks (GANs) for image augmentation in agriculture: A systematic review. Computers and Electronics in Agriculture, 200: 107208. https://doi.org/10.1016/j.compag.2022.107208

[40] Behara, K., Bhero, E., Agee, J.T. (2023). Skin lesion synthesis and classification using an improved DCGAN classifier. Diagnostics, 13(16): 2635. https://doi.org/10.3390/diagnostics13162635

[41] Goyal, M., Mahmoud, Q.H. (2024). A systematic review of synthetic data generation techniques using generative AI. Electronics, 13(17): 3509. https://doi.org/10.3390/electronics13173509

[42] Gumma, L.N., Thiruvengatanadhan, R., Lakshmi, P.D., LakshmiNadh, K. (2022). A binary multi class and multi level classification with dual priority labelling model for COVID-19 and other thorax disease detection. Revue d'Intelligence Artificielle, 36(5): 657-664. https://doi.org/10.18280/ria.360501

[43] Sejuti, Z.A., Islam, M.S. (2023). A hybrid CNN–KNN approach for identification of COVID-19 with 5-fold cross validation. Sensors International, 4: 100229. https://doi.org/10.1016/j.sintl.2023.100229

[44] Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., et al. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data, 8: 53. https://doi.org/10.1186/s40537-021-00444-8

[45] Dinata, R.K., Akbar, H., Hasdyna, N. (2020). Algoritma K-Nearest Neighbor dengan euclidean distance dan manhattan distance untuk klasifikasi transportasi bus. Ilkom: Jurnal Ilmiah, 12(2): 104-111. https://doi.org/10.33096/ilkom.v12i2.539.104-111

[46] Hidayati, N., Hermawan, A. (2021). K-Nearest Neighbor (K-NN) algorithm with Euclidean and Manhattan in classification of student graduation. Journal of Engineering and Applied Technology, 2(2): 86-91. https://doi.org/10.21831/jeatech.v2i2.42777

[47] Gaikwad, V.P., Musande, V. (2023). Advanced prediction of crop diseases using cetalatran-optimized deep KNN in multispectral imaging. Traitement du Signal, 40(3): 1093-1106. https://doi.org/10.18280/ts.400325

[48] Teodorescu, V., Brașoveanu, L.O. (2025). Assessing the validity of k-fold cross-validation for model selection: Evidence from bankruptcy prediction using Random Forest and XGBoost. Computation, 13(5): 127. https://doi.org/10.3390/computation13050127