SMATIKA : STIKI Informatika Jurnal
Accredited Sinta 4 Ministry of Higher Education, Science and Technology Republic of Indonesia SK No. 79/E/KPT/2023
E-ISSN: 2580-6939 || P-ISSN: 2087-0256

Improving the Performance of Convolutional Neural Networks (CNN) in Identification of Agricultural Plant Diseases

Dedi Setiadi[1*], Fido Rizky[2]

[1] *Pagar Alam Institute of Technology, Informatics Engineering Study Program, Masik Siagim Street No. 75, Karang Dalo, Dempo Tengah District, Pagar Alam City, South Sumatra 31521, Indonesia*
[2] *Bina Insan University, Faculty of Engineering, Information Systems Study Program, H.M. Soeharto Street KM. 13, RT. 01, Lubuk Kupang Urban Village, South Lubuklinggau I District, Lubuklinggau City, South Sumatra 31629, Indonesia*

Keyword

CNN; Coffee; Diseases; Inception V3

****Corresponding Author:***
dedisetiadi1212@gmail.com

Abstract

This study aims to improve the performance of the Convolutional Neural Network (CNN) algorithm in detecting coffee leaf diseases using the Inception V3 architecture. The main challenge of this classification is the subtle visual similarity in color, texture, and symptom patterns between diseases. To overcome this, Inception V3 is implemented because of its superiority in multi-scale feature extraction through convolution factorization which reduces parameters while increasing accuracy. The dataset used consists of 1,120 images, evenly distributed into four classes (three types of diseases and one healthy class, each with 280 images), with a training, validation, and test data split ratio of 896:112:112. As a comparison, a conventional basic CNN architecture consisting of 3 convolution layers (3 X 3, stride 1), 3 max-pooling, and 1 dense layer, trained with the same hyperparameters (Adam optimizer, learning rate 0.001, batch size 32) is used. The experimental results show a significant performance improvement; Model accuracy increased from 74.4% on a standard CNN to 97.0% after integrating Inception V3. The scientific contribution of this research lies in mapping overlapping visual characteristics of coffee diseases through multi-scale feature optimization, which demonstrates that computational efficiency can go hand in hand with accuracy improvements on complex agricultural image datasets. These findings confirm that the Inception V3 architecture provides a robust and efficient solution for automating plant disease diagnosis in the field.

1. Introduction

Agricultural modernization is a crucial strategy for addressing various challenges in the agricultural sector through increased investment in mechanization and digital technology [1] to create more efficient and sustainable systems [2]. The application of modern technology can increase the productivity and income of farmers, [3] particularly smallholders, while simultaneously driving economic transformation and attracting younger generations to the agricultural sector [4]. One emerging innovation is digital imaging technology for detecting coffee plant diseases [5].With the help of specialized cameras, image analysis, and artificial intelligence[6], farmers can identify diseases more quickly and accurately than manual observation [7]. This technology also helps monitor overall plant health,[8] recognize disease progression patterns, and support more effective prevention and control measures, thereby increasing coffee plantation productivity [9].

One method that can be used to detect plant diseases is the Convolutional Neural Network (CNN) [10]. CNN is a deep learning model specifically designed for image classification [11],[12]. This model uses convolution and pooling layers to extract important features from images, then the results are processed in a fully connected layer to determine the classification [13]. The CNN architecture consists of several layers, such as input, convolution, activation, pooling, classification, and output [14]. In the convolution layer, filters are used to recognize certain patterns in the image through a convolution process using kernels and strides to produce new outputs [15].Although CNN has good capabilities in image processing, this model still has difficulty distinguishing leaves with similar disease symptoms, especially in coffee plants. One way to overcome this problem is to use a more diverse image dataset [16]. In CNN, convolution layers play a crucial role in extracting image features such as edges, corners, and specific points through a convolution process between the input matrix and filters [17].Inception V3 is one of the CNN architectures used for image recognition and classification [18].This architecture uses several convolution kernel sizes to capture image features more effectively. In addition, Inception V3 also applies regularization and dimensionality reduction techniques to reduce the risk of overfitting [19].

Inception V3 employs a deep and complex architecture that enables effective recognition of image features across multiple sizes and scales [20]. It further improves upon Inception V2 by introducing more efficient convolution operations and a reduced grid size, resulting in lower computational costs without compromising detection performance [20].The model also uses average pooling before the softmax layer to help reduce overfitting and improve accuracy[21]. To speed up data processing, Inception V3 uses dimensionality reduction techniques, such as 1x1 convolutions, resulting in fewer parameters [22]. In addition, the model applies batch normalization for faster and more stable training. During training, data augmentation techniques are also used to improve the model's ability to recognize a wide variety of images and reduce the risk of overfitting. Inception V3 is first trained on large datasets such as ImageNet before being used for specific classification tasks. This process allows the model to better learn image features, resulting in more accurate classification results and faster training. Although the Inception V3 architecture has been widely implemented in plant disease classification, most previous studies have used high visual contrast and homogeneous backgrounds. When this model is confronted with Robusta coffee leaf diseases, where disease spots often have very subtle similarities in texture, color, and visual patterns between classes, coupled with complex variations in field background lighting. Furthermore, previous studies applied standard architectures without analyzing the trade-offs of computational efficiency against limited dataset size variations. Therefore, this study focuses on optimizing Inception V3 multi-scale feature extraction combined with specific data augmentation techniques to address the problem of overlapping features in Robusta coffee images, as well as providing a fair direct comparison with standard CNN models.

The results of this study are expected to significantly contribute to disease classification based on leaf images, particularly by improving the accuracy and efficiency of Robusta coffee disease detection. This research also aims to provide a more engaging and informative approach, offering detailed insights for other researchers in identifying coffee plant diseases. Furthermore, the results of this study can serve as a reference and inspiration for future research in developing more effective and efficient detection methods.

2. Research Method

2.1 Research Approach

To overcome the limitations of conventional CNN algorithms in recognizing coffee leaf diseases, this study uses the Inception V3 architecture to achieve more accurate and efficient identification results [23].This model is able to better recognize image features through a combination of several filters that can capture various shapes, sizes, and disease patterns on coffee leaves. In addition, the model can also adjust the image processing process according to the characteristics of the input image and various background conditions [24].The study was conducted using a computer with an 11th generation Intel Core i7 processor, 16 GB of

RAM, and an NVIDIA GeForce RTX 3060 GPU. The model was trained using the transfer learning method with initial weights from ImageNet, the Adam optimizer, a learning rate of 0.001, and a batch size of 32. A seed setting of 42 was used to ensure consistent and repeatable training results.

2.2 Algorithm Design and Model

This section outlines the architectural structure of the Inception V3 CNN model and the optimization techniques applied, including the implementation of the Inception module for multi-scale feature extraction and the utilization of transfer learning weights. The selection of Inception V3 over other architectures is based on the characteristics of the coffee leaf disease dataset, which has visual similarities (subtle overlapping features) and random variations in symptom size. Unlike ResNet and DenseNet, which require large computational requirements and a massive parameter set for convergence, Inception V3 applies convolution factorization, significantly reducing the number of parameters without compromising the model's expressive capacity [25].Furthermore, compared to MobileNet, which is too compact and risks underfitting on vague features, or EfficientNet, which requires a large data volume to avoid overfitting, Inception V3 offers the most optimal trade-off. Parallel blocks with varying kernels (1X1, 3X3, and 5X5) in Inception V3 are proven to be superior in capturing spatial details of diseases on medium-sized datasets. Next, this section will explain the data preprocessing flow, fine-tuning of the trainable layer configuration, and final analysis on the coffee leaf image dataset.

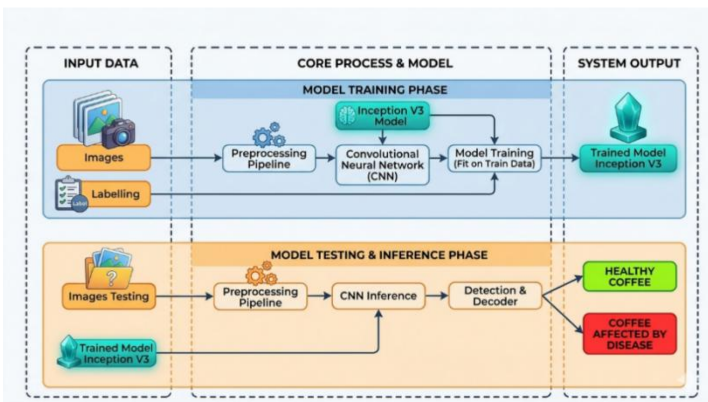

Figure 1. The CNN-Inception-V3 model structure

The following is an explanation and training stages in the block diagram of the CNN Inception V3 method system [26].

1. Input : The first stage contains the dataset. The dataset contains images and labels for each image. The images used are coffee leaves, and the labels used are healthy or diseased leaves.
2. Process : The second stage contains the processing. The images received by the computer are first processed to resize them, then converted into tensors, and these tensors are normalized. These tensors are then processed by CNN Inception V3 to classify coffee beans. The CNN architecture used is Inception V3. Next, the model.fit process is performed to train the Inception V3 model by inputting the preprocessed images and their labels.
3. Output : The output stage contains the results of CNN Inception V3, namely healthy or diseased leaves, as well as a new model from the trained CNN Inception V3 method.

The following is an explanation and testing stages in the block diagram of the CNN Inception V3 method system [27].

1. Input : The first stage contains test images and weights from the trained CNN Inception V3 model. The test images used are different from the training images.

2. Process : The second stage contains the processing. The received images are first processed to resize them, then converted into tensors, and these tensors are normalized. These tensors are then processed by the CNN to classify coffee diseases. The CNN architecture used is Inception V3. The classification predictions from the CNN Inception V3 model are then subjected to a predictive decoder process to determine the original label, since the model output is a tensor.
3. Output : At the output stage, the CNN Inception V3 results are output, namely healthy coffee and diseased coffee.

The overall process stages of research activities can be seen in the image below.

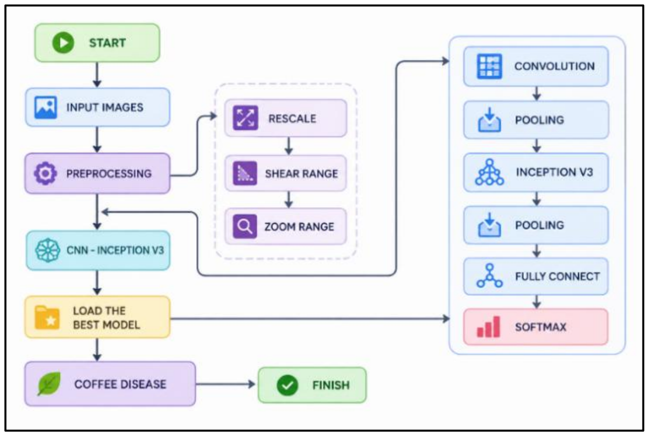

Figure 2. the CNN Inception V3 method

The following is an explanation of the overall flow of the CNN Inception V3 method [28].

1. Image Input: The input is an image of a coffee leaf to be classified.
2. Preprocessing: This stage involves preprocessing using rescale, shear range, and zoom range.
   a) Rescale: The rescale process changes the scale of the image. The scale value used in the rescale process for the yellow coffee berry image is 1/255, meaning each 1/255 value is multiplied, changing the value between 0 and 1.
   b) Shear Range: The shear range process implements the shear transformation method by adding variations to the image by rotating it by a certain degree.
   c) Zoom Range: The zoom range process implements zooming by enlarging the image by a certain scale from the original image. Values smaller than one will result in an enlarged image, while values greater than one will result in a reduced image.
3. CNN Inception V3 : [29] The image is then feature extracted and classified using the CNN method with the Inception V3 architecture.
   a) Convolution : At this stage, convolution is performed on the preprocessed image, forming a filter and producing a feature map output.
   b) Pooling : In the pooling layer, filter shifts are performed across the entire feature map area with a stride size. The stride is the calculation that determines the amount of filter shift.
   c) Inception V3 : This process is performed to reduce computation time and prevent overfitting.
   d) Pooling : At this stage, the pooling process is repeated.
   e) Linear : This process is used to activate linearly, so that the model results are proportional to the input.
   f) Softmax : This stage is performed to run the calculation process for each appropriate target class and assist the target class with the input.

3. Result and Discussions

3.1 Result

The study began by separating leaf objects from the background, followed by feature extraction using a model free from severe overfitting. Over the course of 100 epochs, the training (blue) and validation (orange) accuracy graphs increased rapidly and stabilized above 95% after the 20th epoch, supported by low loss values. When tested on 896 test data sets, the model achieved an overall accuracy of 97% and an average precision, recall, and F1-score of 0.97. Confusion matrix results confirmed this high accuracy, such as perfectly recognizing rust categories (56 out of 56 images) and minimal misclassification (e.g., 2 spots mistaken for rust and 1 sooty mold mistaken for spots). When tested using new images outside the dataset, the model proved to be still able to recognize the disease type with a very high level of confidence.

3.2 Discussions

3.2.1 Dataset

A public dataset of 1,120 images of coffee diseases was used to train and evaluate the InceptionV3 CNN model. While not a time-series dataset, this dataset offers a comprehensive overview of disease types and features. The images cover a wide range of agricultural disease species captured under varying environmental conditions, including varying lighting, backgrounds, and shooting angles, with illustrative examples presented in Figure 3.

Figure 3. Dataset

This dataset contains approximately 1,120 labeled images evenly distributed for training and testing models in complex real-world agricultural situations. The image collection includes diseases such as leaf rust, leaf spot, and sooty mold, captured from different perspectives, backgrounds (fields and greenhouses), and varying weather/lighting conditions. Each image has been labeled with the pest type and its location (bounding box). For modeling purposes, the dataset is proportionally divided into training (80%), validation (10%), and testing (10%) sets to improve the model's accuracy and robustness when used in the real world.

3.2.2 Resize Dataset

All images in the dataset were first resized to 150 × 150 pixels to ensure input uniformity and reduce the influence of image dimension variations on the classification process. The hyperparameter optimization stage was performed using Particle Swarm Optimization (PSO), with a learning rate range of 0.0001–0.001 and a batch size of 16–64. After obtaining the best hyperparameter combination, the robustness and generalization ability of the model were evaluated using 5-Fold Cross Validation. The dataset was divided into five subsets, where four subsets were used as training data and one subset as testing data alternately for up to five iterations. The test results showed an average accuracy of 98.2% with a standard deviation of 0.2%, indicating that the model has stable and consistent performance across various data divisions. This approach was used to minimize evaluation bias and provide a more reliable estimate of model performance compared to a single test.

3.2.3 Augmentation

Data augmentation was applied using ImageDataGenerator to increase the diversity of the training data consisting of 896 training images, 112 validation images and 112 test images divided into four coffee leaf classes [30].The augmentation parameters used include rescale=1/255, rotation_range=20°, width_shift_range=0.2, height_shift_range=0.2, shear_range=0.2, zoom_range=0.2, horizontal_flip=True, and fill_mode='nearest'. Rotation is used to simulate variations in leaf orientation during image capture. Horizontal and vertical shifts are applied to represent changes in object position in the image. The shear and zoom parameters are used to generate variations in leaf shape and size due to differences in image angle and distance. Meanwhile, horizontal flip is used to increase spatial variation without changing the disease characteristics of the leaves. The fill_mode='nearest' method was chosen to fill the empty areas that appear after image transformation. Through this combination of parameters, data variation becomes more diverse, thereby reducing the risk of overfitting and improving the model's generalization ability in recognizing new data.

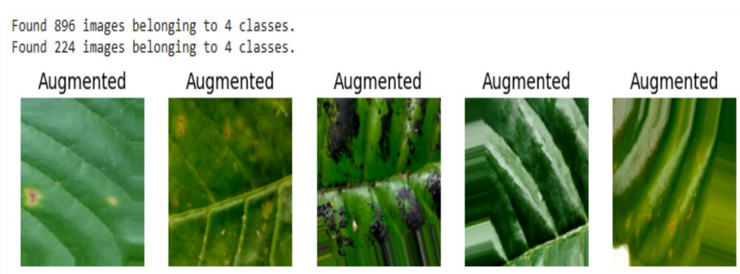

Figure 4. Augmentation

3.2.4 Feature extraction

Feature extraction in this study was performed automatically and end-to-end through convolutional layers in the Inception V3 architecture, rather than through manual color statistical calculations. Input images of coffee leaves, consisting of three color channels (RGB), were processed hierarchically by convolutional filters. In the initial (shallow) layers, the network extracted simple, high-resolution features, such as differences in color pixel intensity, edges, and the outer contour of the leaf.

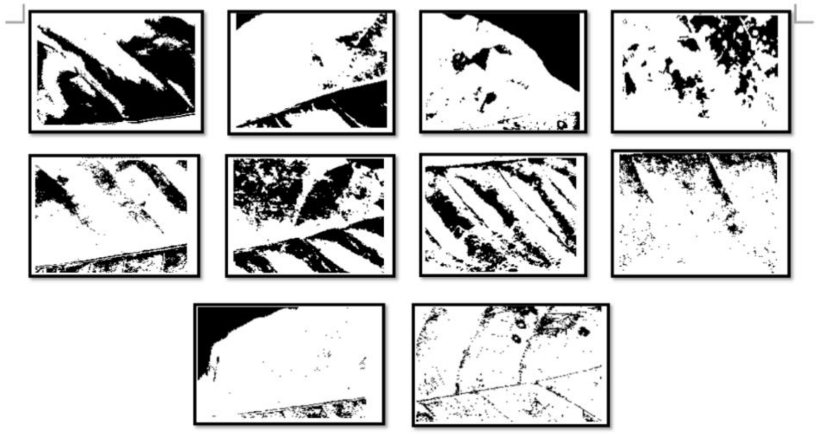

Figure 5. Feature extraction

As data entered deeper layers through the Inception module, the network combined spatial features in parallel using varying kernels (1X1, 3X3, 5X5). This process enabled the model to extract more complex and abstract feature representations, such as necrotic texture, spot distribution patterns, and color degradation characteristic of each type of coffee leaf disease. Through a global average pooling mechanism before the classification layer, all this spatial information was summarized into a dense feature vector without losing any important information. This process of internal segmentation and feature extraction based on multi-scale representation is what separates the disease lesion area from the leaf background automatically, so that the model can classify the type of coffee leaf disease accurately.

3.2.5 Modelling

After the data preparation process is complete, the next stage is the visualization of the main CNN function where if the Output Layer is in 4D (image), it is displayed as an image. If the Output Layer is in 2D, such as Fully Connected (FC) or Dropout, it is changed to a square shape so that it can be visualized. The Output is displayed as a long line. Layers that cannot be visualized will be skipped. The result of the CNN activation visualization are displayed in 8 images. the following is the Output from the CNN visualization.

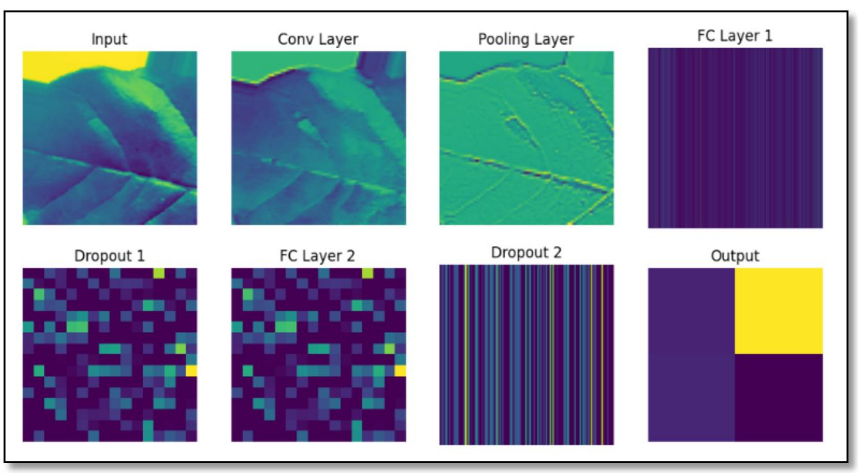

Figure 6. CNN activation visualization

This image shows the activation of the 8 main layers in the CNN model when processing an input image. Here is an explanation of each part :

1. Input Layer, The first image on the top left is the original image that enters the model. The colors in this visualization indicate the intensity of the pixels that have been processed by the model.
2. ConvLayer, This layer performs feature extraction with a convolution filter. There are clearer patterns and textures, indicating that the model has detected some important features in the image.
3. Pooling Layer, The Pooling Layer functions to reduce the dimensionality of the features while retaining important information. The results look simpler than the convolution layer, but still maintain the main shape of the recognized features.
4. FC Layer 1 (Fully Connected), The first fully connected layer connects the features into a vector shape. It appears as a dark block with some bright areas, indicating that only a few neurons are active after feature extraction.
5. Dropout 1, This layer reduces overfitting by randomly ignoring (dropping) some neurons. The varying colors indicate that some units are activated while others are deactivated.
6. FC Layer 2, The second fully connected layer continues feature processing. The visualization shows a vertical line pattern, which can reflect how neurons handle feature information.
7. Dropout 2, The second dropout again ignores some neurons to improve the model's generalization. The irregular color pattern is visible, indicating that most neurons are randomly turned off.
8. Output Layer, This layer is the final result of the CNN classification. The yellow and purple blocks show the probabilities that the model assigns to each class. The model has taken all the previously processed features and determined the final prediction.

Next, at the stage of building a model that can identify types of leaf diseases in coffee plants. The image below shows the results regarding the number of parameters in a deep learning model, which is likely InceptionV3 based on the previous context.

```
Total params: 22,360,740 (85.30 MB)
Trainable params: 557,956 (2.13 MB)
Non-trainable params: 21,802,784 (83.17 MB)
```

Figure 7. The results regarding the number of parameters

The model has a total of 22,360,740 parameters (85.30 MB) covering all network weights and biases. Of these, only 557,956 parameters (2.13 MB) are trainable, representing the final layer that is updated for specific classification tasks such as leaf disease recognition. The remaining 21,802,784 parameters (83.17 MB) are untrainable due to being frozen; these are features from pre-trained models (such as ImageNet) that were intentionally reused to make the training process much more efficient in terms of time and resources.

Once built, the model was trained using 100 epochs (repetitions across the entire dataset) to recognize the training and testing data. The prediction results were determined by the highest probability value for each coffee leaf disease class. In this result, a value of 9.9574620e-01 (≈ 1.0 or 99.999%) was the highest, indicating that the model was highly confident that the leaf belonged to the Sooty Mold class. The next step was to classify the testing data, then evaluate the model's performance and accuracy using a confusion matrix, as visualized below.

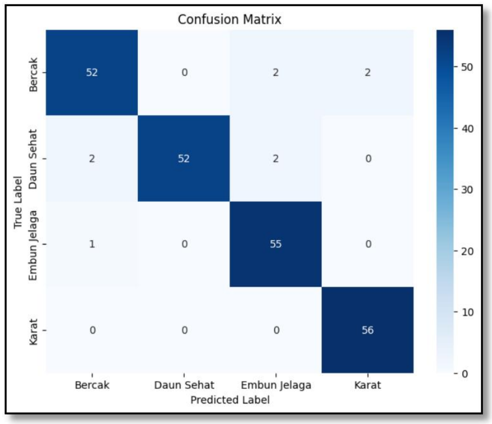

Figure 8. The confusion matrix graph

The confusion matrix graph (Y-axis: original class, X-axis: predicted results) proves the high model accuracy of 97% because the majority of data is on the main diagonal (correct predictions). In detail, the model successfully predicted 52 Spot samples (with 2 incorrectly assigned to Sooty Mold and 2 to Rust), 52 Healthy Leaf samples (with 2 incorrectly assigned to Spot and 2 to Sooty Mold), 55 Sooty Mold samples (1 incorrectly assigned to Spot), and all 56 Rust samples without error. These evaluation results are in line with the test report metrics, where the Spot class achieved a precision of 0.92 and a recall of 1.00; Healthy Leaf achieved a precision and recall of 0.98; Sooty Mold achieved a precision of 0.98 and a recall of 0.96; and Rust achieved a perfect precision of 1.00 but a recall of 0.93 due to samples that missed other classes. Although the performance is already very reliable and balanced (macro and weighted averages are stable at 0.97), model optimization can still be improved through parameter adjustments and adding dataset variations.

Tabel 1. Accuracy Testing Result

	precision	recall	F1-score	support
Healthy Leaf	0,98	0,9866	0,98	56
Spot	0,92	1,00	0,96	56
Sooty Mildew	0,98	0,96	0,97	56
Rust	1,00	0,93	0,96	56
Accuracy			0,97	224
Macro avg	0,97	0,97	0,97	224
Weight avg	0,97	0,97	0,97	224

This accuracy test report shows that the model is able to classify coffee leaves into four categories: Spots, Healthy Leaves, Sooty Mold, and Rust, with excellent results. The Spots category has a Precision of 0.92 and a Recall of 1.00, meaning that all Spot data were successfully recognized, although there were still small prediction errors. Healthy Leaves has a Precision and Recall of 0.98, indicating that most of the data was classified correctly. Sooty Mold also showed excellent results with a Precision of 0.98 and a Recall of 0.96. Meanwhile, Rust has a perfect Precision of 1.00 and a Recall of 0.93, meaning that all Rust predictions were correct, although there were still some data that were incorrectly placed into other categories. Overall, the model achieved 97% accuracy, meaning the model was able to correctly classify 97% of the samples. The macro-average and weighted average values also reached 0.97, indicating stable model performance across all classes. Although there were still small classification errors, the model is very good at detecting coffee leaf diseases. Model performance can still be improved by increasing dataset variations and adjusting model parameters. The results of this study indicate that the Inception V3 model achieves an accuracy of 97%, comparable to or exceeding several previous studies on multi-class coffee leaf disease classification using alternative CNN architectures. For instance, a five-class coffee leaf disease classification study reported that a fine-tuned ResNet50 model achieved 94% accuracy, while a separate four-class coffee disease study found that MobileNetV2 attained only 74.6% validation accuracy under comparable training conditions [31], [32]. The comparatively lower performance of MobileNetV2 in the cited study is consistent with its lightweight depthwise-separable convolutional design, which trades representational capacity for computational efficiency; this trade-off appears to become a limiting factor when disease classes exhibit subtle, overlapping visual symptoms, as is the case with Robusta coffee leaf diseases in the present study. In contrast, ResNet50's residual learning mechanism preserves more gradient information across deeper layers, which may explain its stronger relative performance, though it remains below the accuracy achieved through Inception V3's multi-scale convolution factorization. These comparisons should be interpreted with appropriate caution, as differences in dataset size, class composition, and image acquisition conditions across studies limit direct equivalence; nonetheless, the consistent pattern across independent studies lends support to the premise that architecture choice meaningfully interacts with the degree of inter-class visual similarity in disease symptoms. Under the experimental conditions of the present study, the proposed Inception V3 model achieved an overall classification accuracy of 97%, demonstrating competitive performance for coffee leaf disease classification. The high precision, recall, and F1-score obtained across all disease categories further indicate that the model effectively discriminates between healthy leaves, leaf spot, sooty mildew, and rust. The relatively lower precision observed for the Spot class (0.92) and lower recall for the Rust class (0.93) in this study reflect a pattern commonly reported in coffee leaf disease classification literature, where visually similar disease symptoms tend to produce the highest misclassification rates regardless of the underlying architecture. In a related five-class ResNet50-based study referenced above, the Phoma class likewise yielded the weakest performance among all classes (precision 0.88, recall 0.91), attributed to the visual overlap between necrotic lesion patterns across disease categories. This recurring pattern across independent studies and architectures suggests that the misclassification observed between Spot, Sooty Mold, and Rust in the present study is more plausibly attributable to genuine symptom overlap in the underlying disease pathology than to a deficiency specific to the Inception V3 architecture.

The performance evaluation of the InceptionV3 CNN model over 100 epochs is visualized through accuracy and loss graphs. In the accuracy graph (X-axis: epochs, Y-axis: accuracy), the blue (training) and orange (validation) lines increase rapidly at the beginning and then stabilize consistently above 95% after the 20th epoch, indicating the model has learned the pattern well. Meanwhile, in the loss graph (X-axis: epochs, Y-axis: error rate), the training and validation loss values decrease sharply and reach very low convergence after the 20th epoch. Although the validation loss fluctuates slightly, it remains stable and low, thus not indicating severe overfitting.

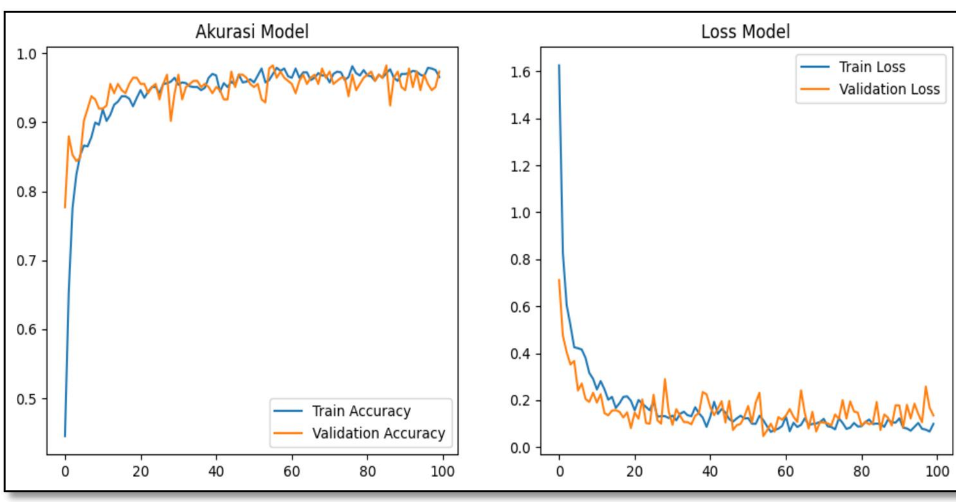

Figure 9. The visualized through accuracy and loss graphs

3.2.6 Evaluation

After the artificial neural network model is successfully built, the next stage is to test the performance of the model. CNN model prediction results for uploaded coffee leaf images. Users upload images via the Choose Files feature, with a file named "(4).jpg". After being uploaded, the image is processed by the CNN model to detect whether the leaf has a disease or is in a healthy condition. It can be seen that the class prediction in the results above is a spot where the correct pesticide recommendation is to use daconil pesticide. The recommendation for Daconil is based on the fact that it contains chlorothalonil, a fungicide effective in controlling various fungal diseases in coffee plants, such as leaf spot and rust. The recommendation is based on the classification results of a CNN model that recognizes visual characteristics of the disease, such as brown or yellowish spots and leaf discoloration. When the model detects patterns consistent with fungal disease symptoms and assigns the highest confidence level to that class, the system automatically recommends Daconil as the appropriate control measure.

3.2.7 Deployment

The process of moving an application or system that has been developed from a development environment to a production environment, so that the application can be used by end users. This process involves installation, configuration, and testing to ensure that the application runs properly in the production environment.

4. Conclusions and Future Works

Based on the research results, it can be concluded that the Convolutional Neural Network (CNN) method with the InceptionV3 architecture was successfully used to identify coffee leaf diseases easily and accurately. This optimization was proven to drastically increase the model accuracy from 74.4% to 97% using 896 training data sets and 112 test data sets. The success of this training process was also marked by a continuously increasing accuracy graph because the model was able to recognize patterns well, as well as a decreasing loss graph as prediction errors decreased. This accuracy gain is consistent with the broader trend observed in recent coffee leaf disease classification research, where multi-scale or attention-enhanced architectures tend to outperform lightweight models on datasets with high inter-class visual similarity, reinforcing the conclusion that architecture selection should be guided by the degree of symptom overlap inherent to the target disease classes rather than computational efficiency alone.Ultimately, this research not only facilitates the identification of coffee leaf diseases in the field, but also deserves to be an important reference for further research based on CNN algorithms.

5. References

[1] A. Morchid, R. El Alami, A. A. Raezah, and Y. Sabbar, "Applications of Internet of Things (IoT) and Sensors Technology to Increase Food Security and Agricultural Sustainability: Benefits and Challenges," *Ain Shams Eng. J.*, vol. 15, no. 3, p. 102509, 2024, doi: 10.1016/j.asej.2023.102509.

[2] A. Khaidar, "Integrasi Teknologi Internet Of Things Dalam Smart Farming Sebagai Strategi Digitalisasi Pertanian Menuju Era Industri 4.0," *Improve*, vol. 17, no. 2, pp. 34–40, 2025.

[3] A. Hussain, T. Senapati, S. Bibi, K. Ullah, L. Jin, and S. Moslem, "Sustainable Agriculture Method Selection Using Spherical Fuzzy Bonferroni Mean-Based Approach," *Franklin Open*, p. 100558, 2026, doi: 10.1016/j.fraope.2026.100558.

[4] K. L. Himayanta and D. F. Wardhani, "Prediksi Hasil Panen Padi Dengan Machine Learning," *Inov. Pembang. J. Kelitbangan*, vol. 13, no. 1, 2025.

[5] S. Matarru, G. A. N. Pongdatu, and J. Rusman, "Klasifikasi Penyakit pada Tanaman Kopi Arabika Menggunakan Metode K-Nearest Neighbor (KNN) Berbasis Citra," *Indones. J. Comput. Sci.*, vol. 12, no. 2, 2023, doi: 10.33022/ijcs.v12i2.3172.

[6] A. I. Saputra, I. Weni, and U. Khaira, "Implementasi Metode Convolutional Neural Network Untuk Deteksi Penyakit Pada Tanaman Kopi Arabika Melalui Citra Daun Berbasis Android," *Decod. J. Pendidik. Teknol. Inf.*, vol. 4, no. 1, pp. 41–51, 2023, doi: 10.51454/decode.v4i1.231.

[7] Y. P. Putra, A. Susanto, and W. Novrian, "Penerapan Jaringan Syaraf Konvolusional Untuk Deteksi Penyakit Tanaman Kopi Berdasarkan Citra Daun," *J. Pustaka AI*, vol. 5, no. 3, pp. 507–513, 2025, doi: 10.55382/jurnalpustakaai.v5i3.1123.

[8] G. A. de Melo and others, "Performance Management in Coffee Growing: A Multivariate Analysis Applied to Coffee Production in the Main Brazilian Regions," *SSRN Electron. J.*, 2023.

[9] M. H. Alhazmi, "An Explainable AI Framework (XAI-CoffeeNet) with Deep Learning and Vision Transformers for Saudi Coffee Disease Recognition and Severity Assessment," *Smart Agric. Technol.*, vol. 14, p. 102259, 2026, doi: 10.1016/j.atech.2026.102259.

[10] N. P. D. A. S. Dewi, I. G. Hendrayana, and I. W. A. W. K. Putra, "Optimasi Hyperparameter Convolutional Neural Network Dengan Arsitektur Mobilenet Pada Klasifikasi Penyakit Daun Jagung," *J. Mnemon.*, vol. 8, no. 1, pp. 92–99, 2025, doi: 10.36040/mnemonic.v8i1.11744.

[11] D. Setiadi and Y. I. Mukti, "An Integrated Framework for Rice Disease Detection and Smart Irrigation Using EfficientNet-B0 and IoT," *Ing. des Syst. d'Information*, vol. 30, no. 9, 2025, doi: 10.18280/isi.300907.

[12] S. Irannejad and H. Bagheri, "Gaussian Mixture Integration of CNN and Transformer Models for Forest Biomass Estimation: A Multi-Sensor Fusion Approach with Uncertainty Quantification," *Smart Agric. Technol.*, p. 102241, 2026, doi: 10.1016/j.atech.2026.102241.

[13] M. A. Mohammed, Y. S. Alsahafi, R. Salama, A. M. Elshewey, and K. M. Hosny, "Explainable Hybrid CNN Model Using DenseNet121 and EfficientNetB0 for Brain Tumor Detection and Classification," *Intell. Syst. with Appl.*, p. 200670, 2026, doi: 10.1016/j.iswa.2026.200670.

[14] Y. Findawati, "Implementation Of Convolutional Neural Network (CNN) To Detect Hate Speech And Emotions On Twitter."

[15] B. Baso and N. Huda, "Hybrid Algoritma Convolutional Neural Network dengan Support Vector Machine untuk Klasifikasi Jenis Tenun Timor," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 12, no. 6, pp. 1233–1242, 2025.

[16] E. Firasari and F. L. D. Cahyanti, "Implementation of Deep Learning for Potato Leaf Disease Detection Using the InceptionResNetV2 Architecture," *Indones. J. Comput. Sci.*, vol. 4, no. 1, pp. 76–81, 2025, doi: 10.31294/n09m6677.

[17] A. T. Adiana, J. Jumadi, and E. Nurlatifah, "Klasifikasi Penyakit Pada Daun Padi Menggunakan Model Hybrid CNN-SVM," *SMATIKA*, vol. 15, no. 2, pp. 258–267, 2025, doi: 10.32664/smatika.v15i02.1548.

[18] A. Iqbal, K. Iqbal, Y. A. Shah, F. Ullah, J. Khan, and S. Yaqoob, "Multi-filter Stacking in Inception V3 for Enhanced Alzheimer's Severity Classification," *Neuroscience*, 2025, doi: 10.1016/j.neuroscience.2025.08.049.

[19] G. Meena, K. K. Mohbey, and S. Kumar, "Sentiment Analysis on Images Using Convolutional Neural Networks Based Inception-V3 Transfer Learning Approach," *Int. J. Inf. Manag. Data Insights*, vol. 3, no. 1, p. 100174, 2023, doi: 10.1016/j.jjimei.2023.100174.

[20] M. W. Sardjono, V. Ramadhan, M. Cahyanti, and E. R. Swedia, "Klasifikasi Bentuk Bingkai (Frame) Kacamata Menggunakan CNN dengan Arsitektur Inception V3 dan Augmented Reality Berbasis Android," *J. Syst. Comput. Eng.*, vol. 5, no. 2, pp. 204–218, 2024, doi: 10.61628/jsce.v5i2.1292.

[21] U. Ungkawa and G. Al Hakim, "Klasifikasi Warna pada Kematangan Buah Kopi Kuning menggunakan Metode CNN Inception V3," *ELKOMIKA*, vol. 11, no. 3, p. 731, 2023, doi: 10.26760/elkomika.v11i3.731.

[22] I. P. Agus, K. Hidjah, N. Sulistianingsih, G. Hendro, and S. Syahrir, "Implementasi Arsitektur Deep Convolutional Neural Network (CNN) dengan Transfer Learning untuk Klasifikasi Penyakit Kulit," *J. Teknol. Inf. dan Multimed.*, vol. 7, no. 3, pp. 461–477, 2025.

[23] K. Z. Maulana and A. Susanto, "Implementasi Arsitektur CNN Inception V3 untuk Identifikasi Spesies Burung Endemik di Indonesia," *J. Pustaka Robot Sister*, vol. 2, no. 1, pp. 22–27, 2024, doi: 10.55382/jurnalpustakarobotsister.v2i1.775.

[24] R. P. Lailatul'Ismi and S. R. Nudin, "Deteksi Penyakit pada Tanaman Tomat Menggunakan Model Inception V3 Berbasis Mobile," *J. Informatics Comput. Sci.*, vol. 7, no. 2, pp. 520–524, 2025.

[25] R. Gunawan, R. Fathurrahman, A. I. S. Widyaningrum, F. Issandra, M. A. Abdurachman, and Y. E. Putra, "Pendekatan Transfer Learning untuk Klasifikasi Penyakit Mata Menggunakan Citra dengan CNN InceptionV3," *J. Comput. Sci. Inf. Technol.*, vol. 6, no. 1, pp. 60–67, 2025, doi: 10.37859/coscitech.v6i1.8509.

[26] A. B. Dash, S. Dash, S. Padhy, N. Kumar, G. K. Pati, and K. Uthansingh, "Leveraging Inception-V3 CNN Model for Efficient Image Classification," in *Intelligent Computing and Communication Techniques*, CRC Press, 2025, pp. 341–348. doi: 10.1201/9781003635680-52.

[27] C. V Kwatra, H. Kaur, M. Mangla, A. Singh, S. N. Tambe, and S. Potharaju, "Early Detection of Gynecological Malignancies Using Ensemble Deep Learning Models: ResNet50 and Inception V3," *Informatics Med. Unlocked*, vol. 53, p. 101620, 2025, doi: 10.1016/j.imu.2025.101620.

[28] M. M. Baig and others, "Enhancing Crop Disease Identification and Management Using Inception V3 Transfer Learning," in *2025 IEEE 1st International Conference on Recent Trends in Computing and Smart Mobility (RCSM)*, IEEE, 2025, pp. 1–5. doi: 10.1109/RCSM67767.2025.11507538.

[29] M. Ahmed, N. Afreen, M. Ahmed, M. Sameer, and J. Ahamed, "An Inception V3 Approach for Malware Classification Using Machine Learning and Transfer Learning," *Int. J. Intell. Networks*, vol. 4, pp. 11–18, 2023, doi: 10.1016/j.ijin.2022.11.005.

[30] M. A. Setiawan, R. Dijaya, C. Taurusta, and A. S. Fitrani, "Mountain Climbing Safety Induction for Beginners Using Augmented Reality," *SMATIKA J.*, vol. 16, no. 1, pp. 45–56, 2026, doi: 10.32664/smatika.v16i01.2215.

[31] B. Mehta, S. Bharany, D. H. Elkamchouchi, A. U. Rehman, R. Singh, and S. H. Adem, "EffResViT-SE FusionNet: A Hybrid Deep Learning Framework for Accurate Classification of Coffee Leaf Diseases," *Food Sci. Nutr.*, vol. 13, no. 12, p. e71311, 2025, doi: 10.1002/fsn3.71311.

[32] D. Novtahaning, H. A. Shah, and J.-M. Kang, "Deep Learning Ensemble-Based Automated and High-Performing Recognition of Coffee Leaf Disease," *Agriculture*, vol. 12, no. 11, p. 1909, 2022, doi: 10.3390/agriculture12111909.