International Journal of Information Engineering and Science

E-ISSN: 3048-1953
P-ISSN: 3048-1902

(Research/Review) Article

Detection of Sugarcane Plant Diseases Based on Leaf Image Using Convolutional Neural Network Method

Arfian Hendro Priyono [1], Ema Utami [2], Dhani Ariatmanto [2]*

[1] Program Studi S2 Teknik Informatika Universitas Amikom Yogyakarta; e-mail: arfianhendro.amikom@gmail.com
[2] Program Pascasarjana Universitas Amikom Yogyakarta; e-mail: ema.u@amikom.ac.id
[3] Program Pascasarjana Universitas Amikom Yogyakarta; e-mail: dhani@amikom.ac.id
* Corresponding Author : Dhani Ariatmanto

Abstract: As the primary raw material for sugar and ethanol production, sugarcane is a highly significant plantation commodity. However, its relatively long growing period of approximately one year makes it more susceptible to diseases. Machine learning technology has been applied in the identification of sugarcane leaves, including through pre-processing methods and the development of disease classification models using Convolutional Neural Network (CNN) and Support Vector Machine (SVM) approaches. However, these methods exhibit limitations in terms of accuracy. Therefore, improving identification accuracy using VGG-16 is essential. The objective of this study is to enhance the accuracy of sugarcane leaf disease identification by utilizing VGG-16. The dataset consists of 2,521 sugarcane leaf images categorized into five classes. The results of this study indicate an accuracy improvement from 97.78% to 99.14%, reflecting an increase of 1.36%

Keywords: CNN; Machine learning; Sugarcane leaf disease; VGG-16

1. Introduction

Indonesia is an agricultural country with an economy that is highly dependent on the plantation sector, including sugarcane as the main commodity in sugar production. However, sugarcane productivity is often disrupted by diseases such as yellow disease, red rot, mosaic, and rust, which cause a decrease in yields. Detection of these diseases needs to be done immediately because it has a significant impact on the quality and quantity of sugarcane produced. Unfortunately, the manual identification process has weaknesses, such as being prone to human error and being less efficient, especially on large-scale plantations. There are a number of previous studies that examine diseases in sugarcane by observing the appearance of certain spots on sugarcane leaves, such as the results of the study. The study conducted did not discuss area-based segmentation or an improved adaptive histogram smoothing method. Instead, this study highlights a multilevel deep learning architecture with an attention mechanism to classify sugarcane leaf diseases, achieving an accuracy of 86.53% on an independently developed dataset.

In the study discussed region-based segmentation or improved adaptive histogram equalization model for sugarcane leaf disease classification. It focuses on CNN and SVM based predictive analysis to classify seven types of leaf diseases with high accuracy. Region-

Received: April, 23 2025
Revised: May, 07 2025
Accepted: May, 21 2025
Published: May, 23 2025
Curr. Ver.: May, 23 2025

based segmentation or improved adaptive histogram equalization model for sugarcane leaf disease classification. It focuses on deep learning models such as ResNet-50 and VGG-16 to classify diseases based on image datasets as in studies. The results obtained with region-based segmentation or improved adaptive histogram equalization model. It focuses on deep learning approach to classify healthy and diseased sugarcane leaves, achieving 97% accuracy using 1200 images. Convolutional Neural Network (CNN) approach for disease detection, achieving 95.2% accuracy on sugarcane leaf image datasets.

In the study [10], a model using CNN and SVM techniques to make accurate predictions about the severity of Grasshopper Disease in sugarcane cultivation was introduced, and its overall accuracy was 81.53%, and precision, recall, F1 score, and support values ranged from 65.71% to 85.37% depending on the severity. The studies conducted by provided evidence that deep learning models can perform better in classification problems and suggested some improvements to continue their contribution. Similarly, in the study a deep neural network was proposed for automatic identification of sugarcane diseases, involving 5 types of diseases and 1 healthy class, and a comparative analysis for different optimizers stochastic gradient descent, Adadelta and Adam was given.

Research conducted by introduced an AI-based Sugarcane Disease Prediction System using Convolutional Neural Network (CNN) to classify sugarcane images into Bacterial Blight, Healthy, or Red Rot categories, helping farmers and researchers in early disease detection through a user-friendly web application. In research [18] based on the results of the tests conducted, the best classifiers used were AlexNet and Support Vector Machine with accuracy, sensitivity, and specificity levels of 93.5%, 95.08%, and 93%, respectively. Likewise in research.

From previous studies, it can be seen that various CNN and SVM methods have been carried out but have weaknesses in accuracy, for this reason it is important in this study to improve the accuracy of identifying rice leaf diseases using VGG-16.

2. Literature Review

2.1 Definition of Image

Image is a visual representation on a two-dimensional plane formed from a two-dimensional analog image and continues into a discrete image, through a sampling step where the analog image is divided into M rows and N columns to form a discrete image (Purba, 2010). Figure 1 shows the coordinates of a digital image on the (x, y) axis on a two-dimensional flat plane.

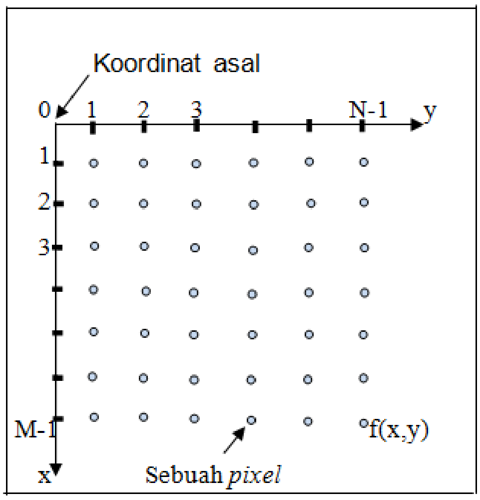

Figure 1. Digital image coordinates

Digital images can be mathematically written in matrix form as follows:

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix}$$

Figure 2. Digital image coordinates

2.2 Digital Image Processing

Digital image processing or can also be called image processing is the process of processing an image output in the form of an image (Nafi'iyah & Mujilahwati, 2018). The beginning of image processing is the improvement of images, then with the development of the era accompanied by the development of the computer world, image processing has developed and is widely used for various needs. Along with its development, image processing has the following main objectives:

a. Improving the quality of the image, where the output of this image processing is able to display information on the image more clearly.

b. Extracting features from an image, namely the output of this processing process is in the form of image data that provides information to humans about image information numerically, or in other words the computer interprets the information contained in the image through clearly distinguished data values.

2.3 Classification

Classification is widely used in various things, including fraud detection, customer processing, fruit ripeness detection, disease identification and many others. Classification itself has the meaning of a technique used to study a set of data which produces certain rules and provisions that can be used to recognize new and previously unseen data (Suyanto, 2017). Classification in machine learning is categorized into supervised learning, or supervised learning. Supervised learning means that the data used has been grouped into classes that have been labeled.

2.4 Convolutional Neural Network

Convolutional Neural Network or abbreviated as CNN is a development of Multilayer Perceptron (MLP) which can be used to process image data, especially two-dimensional images. Convolutional Neural Network (CNN) is often used to recognize objects or detect an object (Arrofiqoh & Harintaka, 2018). Convolutional Neural Network is part of a deep neural network that is generally used for image recognition and processing because of its high network depth and is widely applied to image data. In the Convolutional Neural Network (CNN) architecture, it can capture contextual information that is sometimes in data such as adjacent pixels in an image, which means that the CNN model can independently extract features from an image.

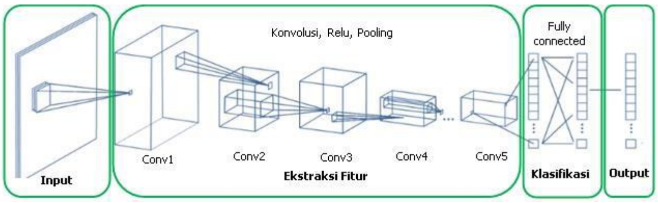

Figure 3. Convolutional Neural Network Architecture

A Convolutional Neural Network model basically consists of several screens for the image classification process as follows:

2.4.1 Input Layer

This layer is where the image that will be used as a dataset for the classification process is entered into the classification model.

2.4.2 Convolutional Layer

This layer functions for convolution operations on a number of nodes in the image using several filters. This operation helps the neural network to recognize objects such as leaf images based on their attributes. The convolution layer consists of an arrangement of neurons that form a filter. Figure 3 is an example of a convolution process using a 3x3 filter, in this convolution process the pixels in the image are multiplied by the filter pixels and produce an output commonly called a feature map.

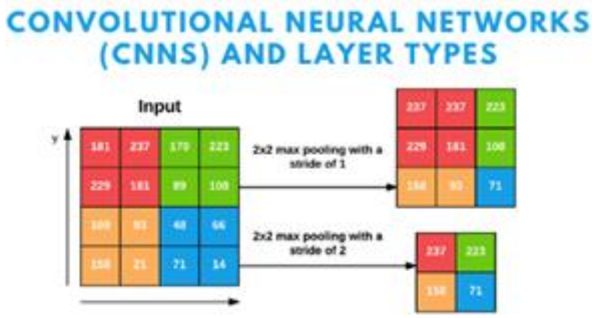

Figure 4. Convolution Layer

2.4.3 ReLU Layer

This layer is an activation function from the previous layer output. In the Convolutional Neural Network architecture, the activation function is located in the final calculation of the feature map output or after the convolution or pooling calculation process to produce a feature pattern. In the ReLU or Rectified Linear Unit function, the threshold is from 0 to infinity, meaning that this function inputs from neurons in the form of negative numbers, then this function will translate the value into a value of 0, and if the input is positive, the output of the neuron is the activation value itself.

2.4.4 Pooling Layer

The pooling layer is a layer that reduces the dimensions of the feature map. In this process, it is used to reduce the image resolution while maintaining the information in the image, so that it will provide greater speed in computing for classification. This layer is usually applied after the convolution layer and has several types, including max pooling and average pooling. The pooling layer used consists of a filter that has a certain size and also a certain stride, where stride is the amount of shift from the filter during the pooling process.

2.4.5 Full Connected Layer

This layer functions to calculate the output results from the convolution and pooling layers. In the convolution layer, the output produced is still in the form of a multi-dimensional array, so a process called flatten is needed where the meaning of flatten here is to change the output of the convolution result (feature map) into a vector form.

2.5 Confusion Matrix

Confusion Matrix can be used to evaluate the classification of models in estimating true or false objects. True positive (TP) and true negative (TN) provide information when the classification is correct, while false positive (FP) and false negative (FN) provide information when the classification is wrong (Pravina et al., 2019).

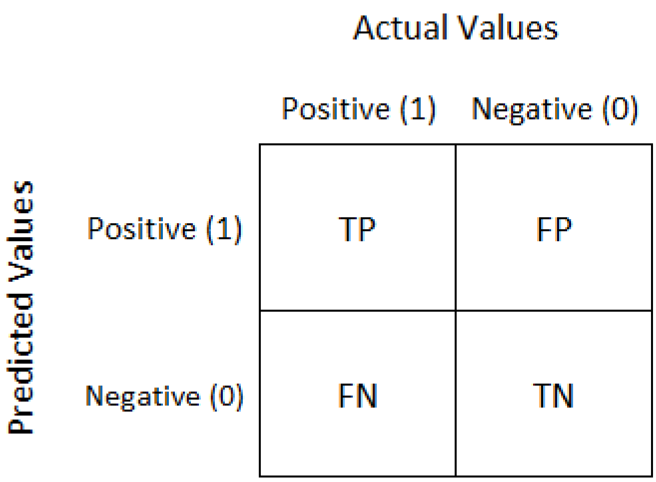

Figure 5. Confusion Matrix

Description:

a. TP (True Positive) is the amount of data whose actual class is positive class with the predicted class is positive class.

b. FN (False Negative) is the amount of data whose actual class is positive class with the predicted class is negative class.

c. FP (False Positive) is the amount of data whose actual class is negative class with the predicted class is positive class.

d. TN (True Negative) is the amount of data whose actual class is negative class with the predicted class is negative class.

3. Proposed Method

The research method can be shown in Figure 1.

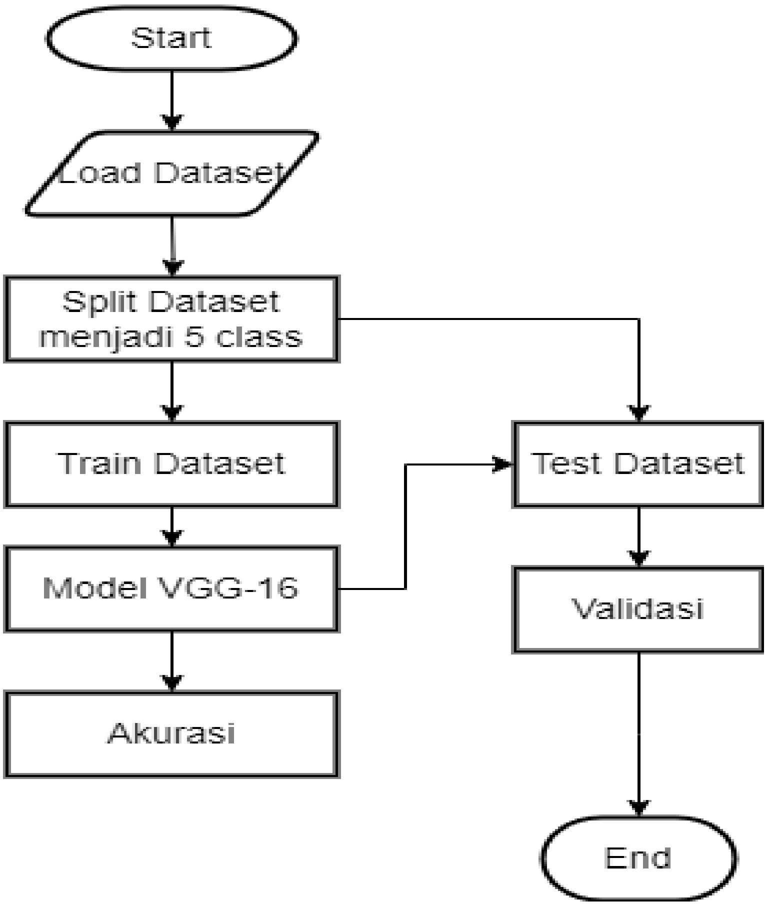

Figure 6 Research methods

a. Dataset

The dataset used in this study is a public dataset taken from Kaggle with the following link: https://www.kaggle.com/datasets/pritpal2873/sugarcane-leaf-disease-dataset, from the existing dataset there are five classes where each class consists of: Healthy as many as 522 files, Mosaic as many as 462 files, RedRot as many as 518 files, Rust as many as 514 files and Yellow 505 files, an example of a rice leaf image is shown in Figure 2.

Figure 7 Example of a sugarcane leaf image

b. Split Dataset into 5 Classes

The dataset that has been called from Kaggle is put into a data folder and each is saved in a folder according to the name of the type of sugarcane leaf disease.

c. Divide the Dataset into Train Data and Test Data

The dataset that is already in the folder according to the name of the sugarcane leaf disease, the Dataset is divided into data_Train and data_Test, the data is divided into 80:20. This means 80% data_Train and 20% data_Test.

d. VGG-16

VGG-16, introduced by Simonyan and Zisserman in 2014, is one of the popular convolutional neural network (CNN) architectures. This model combines simplicity of design with significant depth, utilizing a stack of small 3×3 convolutional layers, followed by a pooling layer to reduce the spatial dimension. This model consists of a total of 16 trained layers, consisting of; 13 convolutional layers, 3 fully connected (dense) layers, 1 output layer.

e. Main components of VGG-16:

Convolutional Layer, Each layer uses a 3×3 filter with the same padding, so that the output dimension is the same as the input except for pooling. ReLU activation

$$\boldsymbol{f\,(x) = \max(0, x)} \quad (1)$$

used to introduce non-linearity

Pooling Layer, Max pooling (2×2) with stride 2 is used to reduce the spatial dimension of the image.

Fully Connected Layer, This layer connects the spatial features of the convolutional layer to the output layer.

Softmax Output, the softmax function is used in the final layer to provide probabilities for each class The output dimension for each convolutional layer can be calculated using the formula:

$$\boldsymbol{W_{out} = \frac{W_{in} - F + 2P}{S} + 1} \quad (2)$$

$$\boldsymbol{H_{out} = \frac{H_{in} - F + 2P}{S} + 1} \quad (3)$$

Where:

W_in,H_in:Input width and height dimensions

F:Filter size (3x3)

P:Padding (usually 1 for same padding)

S:Stride (usually 1)

W_out,H_out:Output width and height dimensions

For pooling:

$$\boldsymbol{W_{out}} = \frac{\boldsymbol{W_{in}}}{\boldsymbol{S}} \quad (4)$$

$$\boldsymbol{H_{out}} = \frac{\boldsymbol{H_{in}}}{\boldsymbol{S}} \quad (5)$$

f. Accuracy and Validation

Accuracy measures the extent to which the model predictions match the actual labels in the dataset.

Accuracy formula:

$$\boldsymbol{Akurasi} = \frac{\boldsymbol{TP+TN}}{\boldsymbol{TP+TN+FP+FN}}\boldsymbol{x100\%} \quad (6)$$

Where:

TP (True Positive): Correct prediction for the positive class.

TN (True Negative): Correct prediction for the negative class.

FP (False Positive): Wrong prediction for the positive class.

FN (False Negative): Wrong prediction for the negative class.

Validation is done to measure the performance of the model on data that is not involved in training (validation set). Typically, metrics such as accuracy, precision, recall, and F1-score are used in the validation stage.

g. Precision

Measuring the prediction accuracy for the positive class:

$$\boldsymbol{Presisi} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} \quad (7)$$

Recall (Sensitivity)

Measuring the model's ability to detect all instances of the positive class:

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}} \quad (8)$$

F1-Score

Combines precision and recall in one metric to address data imbalance:

$$\boldsymbol{F1-Score} = \boldsymbol{2} \cdot \frac{\boldsymbol{Presisi.Recall}}{\boldsymbol{Presisi+Recall}} \quad (9)$$

4. Results And Discussion

A series of processes carried out to classify diseases in images of sugarcane leaves are as follows:

a. Dataset Preparation

The initial stage after calling the dataset from Kaggle is the process of storing images according to their class, classes = ['Healthy', 'Mosaic', 'RedRot', 'Rust', 'Yellow'] The results are automatically saved in the Healthy, Mosaic, RedRot, Rust and Yellow folders, as shown in Figure 3.

Figure 8 Image folders according to class

The percentage of results from the number of images used for each class used can be shown in Figure 4.

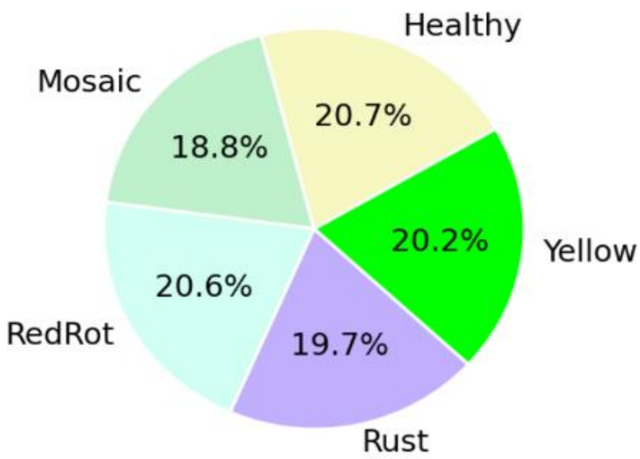

Figure 9 Image data proposition according to the class used

The existing dataset is then split into Train data and Test data. The results of the data split from the testing process can be shown in Figure 5.

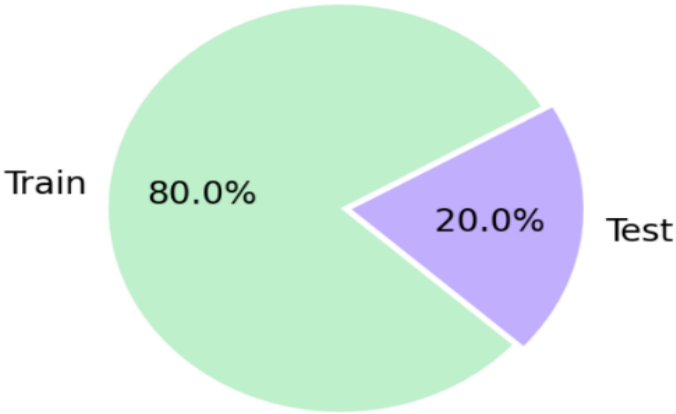

Figure 10 Results of dividing the dataset into Train data and Test data

After the data is split, the next step is to carry out testing. The stages of the testing process on VGG-16 can be shown in Figure 6,

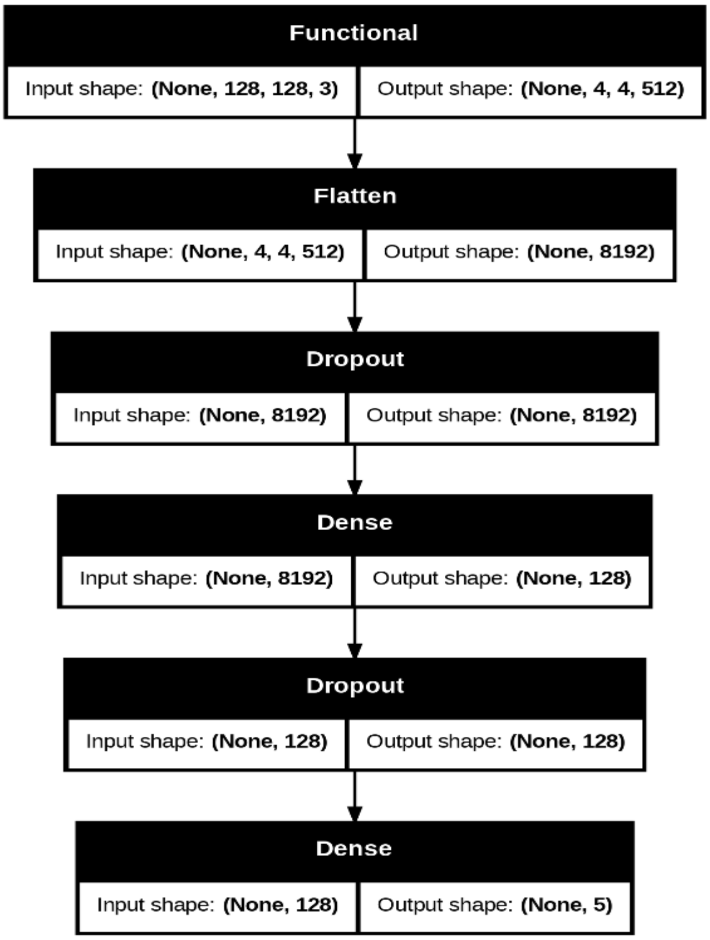

Figure 11 VGG-16 model process

Testing using the VGG-16 model. The test results can be shown in Figure 7.

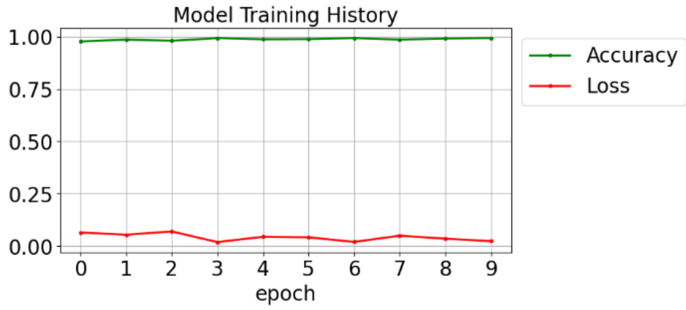

Figure 12 Training history results of the VGG-16 model

From Figure 7, it can be seen that the green line on this graph, the model accuracy tends to be stable at a high value of 0.9914, indicating that the model has learned well to distinguish the five classes. While the Loss result of 0.0240 the loss value tends to decrease and then becomes stable, indicating that the model has found the optimal parameters to minimize errors. The validation results can be shown in Table 1

Table 1 Validation Results

	precision	recall	f1-score	support
Healthy	0,75	0,9	0,82	104
Mosaic	0,72	0,72	0,72	83
RedRot	0,84	0,73	0,78	103
Rust	0,89	0,74	0,81	117
Yellow	0,74	0,83	0,78	98
Accuracy			0,79	505
Macro avg	0,79	0,79	0,78	505
weighted avg	0,79	0,79	0,79	505

The results of the confusion matrix are shown in

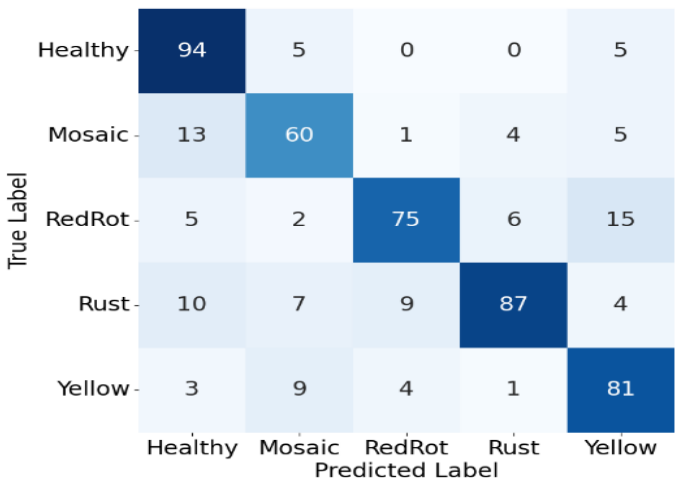

Figure 13 Confusion matrix results

The results of the confusion matrix can be explained from the total Healthy class image data, 94 identified data indicate the Healthy class (true label), while 5 identified data are included in the Mosaic class and 5 identified data are included in the Yellow class. Mosaic image class data from 83 identified data are correctly included in the Mosaic class as many as 60 image data, while 13 identified image data are included in the Healthy class, 1 identified data is included in the RedRot class, 4 identified data are included in the Rust class and 5 mask image data in the Yellow class. RedRot image data consisting of 103 data, are correctly identified as the RedRot class as many as 75, while 5 identified data are included in the Healthy class, 2 data are included in the Mosaic class, 6 identified data are included in the Rust class, and 15 data are included in the Yellow class. For Rust class image data consisting of 117 data, 87 image data were identified as Rust class, 10 image data were identified as Healthy class, 7 images were identified as Mosaic image, and 9 images were identified as RedRot class and 4 images were identified as Yellow class. Meanwhile, the final test results for Yellow class consisting of 98 image data, 81 images were identified as Yellow image, 3 images were identified as Healthy image, 9 images were identified as Mosaic image, 4 images were identified as ReddRot image and 1 image was identified as Rust image.

It can be seen from the results of previous tests using CNN, the results of the testing process obtained an accuracy of 0.9778 and a loss result of 0.0593. The comparison of the results can be shown in the table.

Table 2 Comparison of train results

	Akurasi	Loss
CNN	0,9778	0,0593
VGG-16	0,9914	0,0240

5. Conclusions

The training results show that the VGG-16 model has been successfully trained to diagnose sugarcane leaf diseases with good accuracy. From the previous test results using CNN with an accuracy of 0.9778 or 97.78% and using VGG-16 with an accuracy of 0.9914 or 99.14%, there was an increase in accuracy of 1.36%. As for the loss results, there was a reduction in results from 5.93% to 2.40% so that the decrease occurred by 3.53%. The increase in accuracy and decrease in loss indicate that your VGG-16 model has been successfully trained for the task of classifying sugarcane leaf diseases.

References

[1] S. D. Daphal, "Enhanced deep learning technique for sugarcane leaf disease classification and mobile application integration," Heliyon, vol. 10, no. 8, 2024.
[2] S. Singh, "Enhancing Sugarcane Crop Health: CNN and SVM-Based Predictive Analysis of Leaf Diseases," in 2024 3rd Int. Conf. Innovation in Technology (INOCON), 2024.
[3] M. A. R. Yead, "Deep Learning-Based Classification of Sugarcane Leaf Disease," in Proc. 6th Int. Conf. Electrical Engineering and Information and Communication Technology (ICEEICT), pp. 818–823, 2024.
[4] V. Tanwar, "Deep Learning-based Hybrid Model for Severity Prediction of Leaf Smut Sugarcane Infection," in Proc. 3rd Int. Conf. Artificial Intelligence and Smart Energy (ICAIS), pp. 1004–1009, 2023.
[5] V. Tanwar, "Deep Learning-based Approach for Leaf Disease of Sugarcane Classification," in Proc. 2023 12th IEEE Int. Conf. Communication Systems and Network Technologies (CSNT), pp. 176–180, 2023.
[6] A. Atheeswaran, "Deep Learning-based Diagnosis of Sugarcane Leaf Scald Diseases: A Cutting-Edge Approach," in 15th Int. Conf. Advances in Computing, Control, and Telecommunication Technologies (ACT), vol. 1, pp. 242–249, 2024.
[7] R. Plant, D. Diagnosis, and U. Deep, "CROP GURU: PRECISE AND RAPID PLANT DISEASE," vol. 3, pp. 160–168, 2024.
[8] G. M. Reddy, "A Survey on Sugarcane Leaf Disease Identification Using Deep Learning Technique (CNN)," Int. J. Recent Innov. Trends Comput. Commun., vol. 11, no. 5, pp. 248–254, 2023.
[9] D. Li, "Application of Deep Reinforcement Learning Based Graph Convolutional Neural Network for Sugarcane Leaf Disease Identification," ACM Int. Conf. Proceeding Series, pp. 13–17, 2023.
[10] D. B. V. K. S. H. V. J. S. Dutta, "An Intelligent Framework for Grassy Shoot Disease Severity Detection and Classification in Sugarcane Crop," in 2023 2nd Int. Conf. Applied Artificial Intelligence and Computing (ICAAIC), 2023.
[11] A. A. Hernandez, "Classification of Sugarcane Leaf Disease using Deep Learning Algorithms," in 2022 IEEE 13th Control and System Graduate Research Colloquium (ICSGRC), pp. 47–50, 2022.
[12] R. Maurya, "A Deep Convolutional Neural Network for Leaf Disease Detection of Sugarcane," in 2023 14th Int. Conf. Computing Communication and Networking Technologies (ICCCNT), 2023.
[13] V. Tanwar, "AI-Driven Deep Learning Models for Efficient Sugarcane Leaf Disease Diagnosis," in 4th Int. Conf. Sustainable Expert Systems (ICSES), pp. 1250–1254, 2024.
[14] S. Srivastava, "A Novel Deep Learning Framework Approach for Sugarcane Disease Detection," SN Comput. Sci., vol. 1, no. 2, 2020.
[15] N. Amarasingam, "Detection of White Leaf Disease in Sugarcane Crops Using UAV-Derived RGB Imagery with Existing Deep Learning Models," Remote Sens., vol. 14, no. 23, 2022.
[16] S. D. Daphal, "Efficient Use of Convolutional Neural Networks for Classification of Sugarcane Leaf Diseases," Lecture Notes in Electrical Engineering, vol. 828, pp. 675–680, 2022.
[17] Y. Chauhan, "Artificial Intelligence Based Sugarcane Leaf Disease Prediction System for Smart Farming," in Proc. Int. Conf. Circuit Power and Computing Technologies (ICCPCT), pp. 106–111, 2024.
[18] M. Syarief and W. Setiawan, "Convolutional neural network for maize leaf disease image classification," Telkomnika (Telecommunication Comput. Electron. Control.), vol. 18, no. 3, pp. 1376–1381, 2020.
[19] K. Rajput, "Enhancing Crop Health: CNN-SVM Fusion for Sugarcane Leaf Disease Analysis," in 2024 3rd Int. Conf. Innovation in Technology (INOCON), 2024.
[20] V. S. Kumar, "Recognition and Classification of Apple and Sugarcane Plant Leaf Diseases using SVM with DAE Models," in Int. Conf. Distributed Computing and Optimization Techniques (ICDCOT), 2024.
[21] U. Vignesh, "EnC-SVMWEL: Ensemble Approach using CNN and SVM Weighted Average Ensemble Learning for Sugarcane Leaf Disease Detection," in 2nd Int. Conf. Sustainable Computing and Data Communication Systems (ICSCDS), pp. 1663–1668, 2023.