Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control
Journal homepage: http://kinetik.umm.ac.id
ISSN: 2503-2267
Vol. 8, No. 3, August, Pp. 673-682

Classification of coffee leaf diseases using CNN

Dara Sucia*[1], Auliya Tara Shintya Larasabi[1], Yufis Azhar[1], Zamah Sari[1]
Universitas Muhammadiyah Malang, Malang, Indonesia[1]

Article Info

Keywords:
Images Classification, Coffee Leave Diseases, Deep Learning, CNN, VGG-19

Article history:
Received: May 31, 2023
Accepted: July 01, 2023
Published: August 31, 2023

Cite:
D. Sucia, A. T. Shintya Larasabi, Y. Azhar, and Z. Sari, "Classification of Coffee Leaf Diseases using CNN", KINETIK, vol. 8, no. 3, Jul. 2023.
https://doi.org/10.22219/kinetik.v8i3.1745

*Corresponding author.
Yufis Azhar
E-mail address:
yufis@umm.ac.id

Abstract

Indonesia's coffee industry plays a crucial role as a major export, making a significant contribution to the country's economy by generating foreign exchange. The quality and quantity of coffee production depend on various factors such as humidity, rain, and fungus that can cause rust diseases on coffee leaves. These diseases can spread quickly and affect other coffee plants quality, leading to decreased production. To address this issue, CNN with VGG-19 architecture model was utilized to identify coffee plant diseases using image data and the python programming language, which in previous studies used MATLAB as their platform. In addition, VGG-19 with image enhancement and contouring data for pre-processing step has a more profound learning feature than the method used in the previous studies, AlexNet which makes the structure of VGG- 19 more detailed. The dataset used in this paper is Robusta Coffee Leaf Images Dataset which have three classes, namely health, red spider mite, and rust. The VGG-19 model attained F1-Score of 90% when evaluated using the testing data with ratio 80:20, where 80% is training data, and 20% is validation data. This paper employed 0.0001 learning rate, batch size 15, momentum 0.9, 12 training iteration, and RMSprop optimizer.

1. Introduction

Indonesia is a country situated on both sides of the equator, which makes it an ideal location for coffee plantations due to the stable seasons and weather conditions in the Bean Belt region. Coffee is an important crop in Indonesia, playing a significant role in the country's economy and serving as a major export product that generates foreign exchange. In 2021, the total coffee production in Indonesia amounted to 762 thousand tons. Among this, smallholder plantations contributes 0.49% or 3.7 thousand tons, while private plantations made up the remaining 0.18% or 1.4 thousand tons [1].

Indonesia's coffee exports can only meet 4% European Union's demand due to various factors. According to a study conducted by the Center for Indonesia Policy Studies (CIPS) on the food sector, there are at least two reasons for the low coffee production in Indonesia. The first reason is that aging coffee trees are more vulnerable to diseases, which can harm their growth and productivity [2].

Physical alterations to coffee plant leaves can occur due to various factors, such as pest infestations and unpredictable weather conditions. Diseases, such as rust and red spider mite, can also attack coffee plant leaves, primarily due to humidity, rain, and the fungus Hemileia Vastatrix. To identify or visualize these diseases, many innovative technological advancements, including direct and laboratory convolution assays, have been developed. However, it is crucial to have adequate knowledge of the diseases condition to recognize it when observing it in person [3].

The process of classification involves sorting each image into specific categories based on its features. It is a crucial technique used for image recognition. Classification is a data mining method that enables the identification of items in a dataset and categorizes them according to their class or level [4]. The primary goal of classification is to anticipate the target class for every instance in the dataset. This target class is typically known as the training dataset, where the classification algorithm analyzes the data is process known as training. During the training phase, the algorithm generates model outcomes that can be used to categorize new data [5].

Currently, deep learning continues to be a machine learning approach that facilitates the automatic identification and detection of objects in digital images [6]. The researchers in this study have utilized a technique called Convolutional Neural Network (CNN) which is widely used in the field of deep learning for identifying and classifying various objects across different visual inputs.

CNN comprises various architectural models, including VGG-19, which was employed in this study. Despite the diversity in CNN architectures, they all share the same objective. Some of the most commonly used designs for image classification challenges include VGG, ResNet, AlexNet, GoogleNet, MobileNet, and others [7].

Previous research has utilized the CNN method for classifying diseases on leaves. In one such study [5], grape leaf diseases images were classified using the K-Means Clustering algorithm for segmentation. Feature extraction was performed using the VGG-16 transfer learning method, and classification was carried out using the CNN algorithm. The dataset comprised 4000 images of grape leaves obtained from Kaggle, with an additional 100 image data sourced for Google used as test data outside the dataset. The CNN model achieved an accuracy 99.50% on the training data, while the test accuracy on test data was 00.25%.

In another study [8], the researchers employed VGG and ResNet50 models to predict the age of tea leaves, using the CNN algorithm for digital image processing. The researchers collected 600 images and used the RMSprop optimizer, learning rate 0.01, batch size 32, and 100 epochs to segment 1800 images. The accuracy of the system was evaluated through testing and was found to be 97.5%.

In another research [9], the CNN technique was utilized to classify images of mango leaves. The images were collected from three categories of mango manalagi, and 90% of the data was used for training, while 10% was used for validation. The training data was iterated 60 times, which resulted in the best accuracy. The validation data had an accuracy of 89.20%, while the training data had an accuracy of 97.725%.

According to a research [10], Rice Leaf Disease was divided into three types: Bacterial Leaf Blight, Brown Spot, and Leaf Fungus. The transfer learning method employed ResNet101 pre-training model, and added architectural layers like Dense Layer, Dropout Layer, and Batch Normalization Layer. The validation data exhibited an accuracy rate of 100%, with a loss value of 5.61%.

In a previous article [2], MATLAB programming platform and CNN were utilized to detect coffee plant disease using images. The research employed 300 image data, which were categorized into three classes: health, rust, and red spider mite. This study obtained an accuracy rate of 80.56% on the training data using 240 image data, and an accuracy rate of 81.60% on the testing data using 60 image data.

The aim of this study was to detect coffee plant leaf disease images using the Python platform, and utilized a CNN model specifically VGG-19, which is known for its excellent image classification capabilities and has been a winner of ImageNet Challenge in 2014. The VGG-19 model is composed of 16 convolutional layers, and 3 fully- connected layers, making it powerful network. CNN consists of three layers: convolutional, pooling, and fully-connected layers [11].

The differences between the proposed study and previous study are the previous study used MATLAB as their platform to work on the program, it did not use the image enhancement and contouring data to classify the images data, and it only used one model, AlexNet. The reason the authors included ResNet50 model only in the results and discussion chapter is that in addition to overcoming problems with gradients by adding output from the previous layer to the next layer, ResNet50 can also remove redundant network structures to speed up data processing, which can cause problems with gradients. As a result, the authors conducted a comparison between two models, ResNet50 and VGG-19. VGG-19 has several advantages compared to ResNet50, and the model used in the main research journal, AlexNet, including the VGG-19 model using many convolutional layers with a small filter and layer pooling to enhance the effectiveness of image classification. VGG-19 also uses the ReLU activation function to minimize backpropagation errors and improve network performance, and uses the last layer, the softmax layer, to prevent overfitting.

2. Research Method

In prior research, CNN technique was used, specifically AlexNet, which is akin to an earlier CNN model named LeNet. However, AlexNet has a more complex and deeper structure, which slows down model training because of higher image resolution and more computationally intensive convolutions. The final two hidden layers of AlexNet necessitate 6400 × 4096, and 4096 × 4096 sizes. This translates to significant expenses in terms of memory, processing power, and floating-point operations per second (MFLOP), especially for smaller devices like mobile phone. As a results, more advanced architectures have surprassed AlexNet in terms of efficiency.

The presented Figure 1 depicts the design process that involves several stages, including load data images (input data), preprocessing, image enhancement, contour detection, augmentation using data train and validation, and evaluation model using validation data as data test. The first step involves inputting image data and splitting it into two sets, which are the training and validation data, in 80:20 ratio. Following this, image enhancement is carried out to enhance the quality of the visual information in the images, and contour detection is performed to analyze shapes and detect object recognition. The subsequent step is to resize the images data utilizing the Image Data Generator parameter, which can preprocessing all training images to smaller or larger to reduce overfitting while training the data. Ultimately, the processed data undergoes numerous iterations, and the models accuracy is evaluated using validation data.

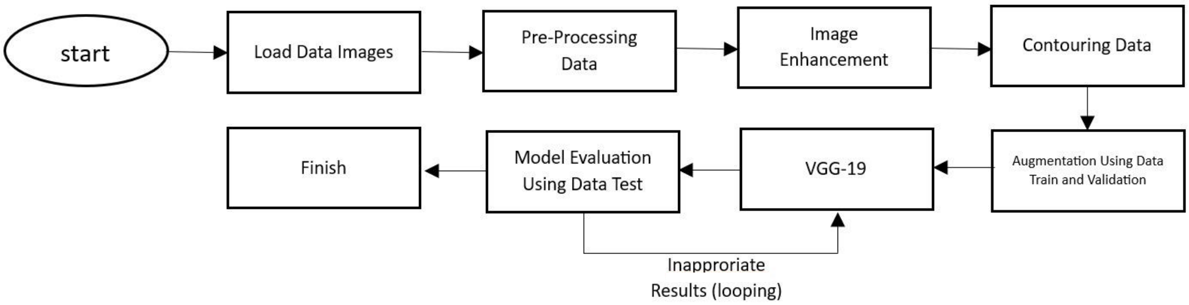


Figure 1. Flowchart of research method

2.1 Dataset

This research places its emphasis on the leaves of coffee plants, which can serve as indicators of potential impacts caused by different factors like pest invasions, unpredictable weather patterns, and the presence of the fungus Hemileia Vastatrix. These factors may result in observable alterations in the coffee plant leaves. Thus, researchers perform classification tasks to detect and identify diseases in coffee plants through the leaves. The Kaggle website provided the dataset for this study, which is titled RoCoLe: Robusta Coffee Leaf Images Dataset. It consists of 1560 images in jpg format and contains three category: health, rust, and red spider mites. Based on previous research, in order for the number of dataset for each class to be the same, 100 images were selected manually according to the research needs for each class. The data criteria to be used are images with a good focus on the main object without any other objects around it. Figure 2 shows some examples of the dataset for each classes.


(a) (b) (c)

Figure 2. Sample data from (a) Health, (b) Red spider mite, and (c) Rust

2.2 Preprocessing

The raw data is transformed into training data through a process called preprocessing to enhance the accuracy of the model. After importing the dataset from Google Drive to Google Colaboratory, the data is split into 240 images for training data, and 60 images for validation data. The images are then converted to the RGB color space and resized to 180 × 180 pixels to be used as input for the CNN model.

In this research, data normalization is one of the common steps performed in preprocessing. Data normalization aims to transform the scale of the data to have a consistent range or distribution. This ensures that all features have a similar scale, so that no element dominates or has a more significant influence than other features. Normalization also helps reduce the impact of outliers in the data. The normalization technique used in this research is normalizing the image by using min-max scaling, dividing the pixel value of each layer by 255, which scales the values to an internal of 0-1, with 1 being the highest pixel value in an RGB image.

2.3 Image Enhancement

The process of improving the quality of an image is essential before detecting its contours. The ultimate goal of this step is to improve the overall visual appearence of the image. Although it cannot restore the lost details, the image contrast can be improved. The outcomes of image enhancement have a significant impact on the evaluation results in this study. CNN is a widely used technique because it can learn complex visual features from numerous examples [12], [13].

Color and shape are crucial aspects of image processing, and they are part of one of the two main concepts in this field. Colors are used in design differentiate common object, and color spaces are integrated into image processing algorithms. From a mathematical perspective, digital images can be represented using a matrix that consists of a fixed number of pixels or dot cells. Grayscale images require one value to indicate pixel intensity, typically in the range of [0, 255], while color image use three value to represent the amount of red, green, and blue stored in them. A “binary image” is a representation that has only two possible intensity levels. Enhancing color and shape features is essential for accurate image processing, and CNN are well-known for their ability to learn intricate visual features from a large number of examples [14].

In this study, image enhancement is used after image pre-processing is complete. The method that used in this image enhancement process is changing all the original colors of the data images in each class to gray (grayscale). The two techniques used are sharpening, and color adjustment. The sharpening is used to emphasize the edges and details in the images so that it looks clear. Color adjustment is used to adjust the color balance, saturation, and hue to improve the quality images. After applying the sharpening and color adjustment methods, the results of the image enhancement are converted to a confusion matrix with and without normalization which can be seen in Figure 3.

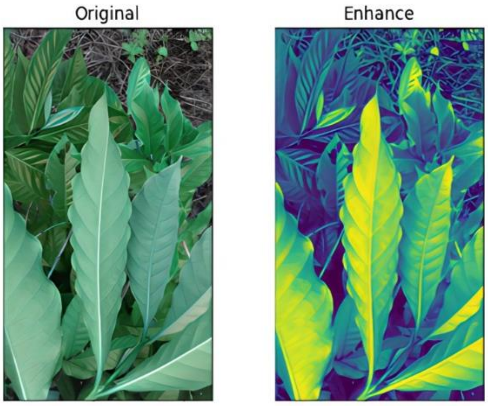


Figure 3. Image enhancement result

2.4 Contouring Data

The “contour” is a curved line that connects adjacent points of the same hue or value. It’s an essential tool in shape analysis and object identification. Contours are commonly shown as a set of circular points or polygons. The CNN technique successfully deals with intricate duties, including contour detection, classification, and segmentation. This technique can even distinguish between the edges and contours of the target object.

The contour processing in the OpenCV library comprises two functions: the find contour function, which can scan the backdrop of the image data for objects, captures contour patterns, and creates approximations, and the draw contour function, which catalogs contour objects and empowers users to depict them with distinctive characteristics such as color and line width [15], [16], [17]. In Figure 4, this study utilizes the results of image enhancement, namely greyscale, then applies the draw contour function, which is used to draw outline lines in images, and also applies the find contour function, which is used to take the source image as a parameter and returns a list of contours found in the image.

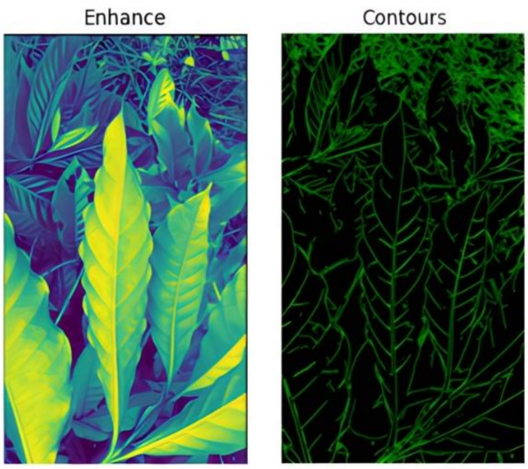


Figure 4. Contouring data results

2.5 Augmentation Data

During the training of CNN, overfitting can be a problem where spurious variations or errors are given too much weight in the explanation compared to the underlying relationship. One solution to this issue is to use augmentation data, which involves increasing the amount of available data [18]. Data augmentation is a commonly used technique in image classification scenarios, wherein each class of the dataset can be expanded by applying transformations such as rotation or resizing to the original image [19]. By improving augmentation techniques, it's possible to increase the amount data without introducing new data [3]. The aim of this process is to enhance the accuracy of data training by modifying ata within each class. Because of that, the authors use the Tensorflow library provides various techniques like rotation, zooming, and other methods that can be used to alter the data and influence the classification results.

2.6 Convolutional Neural Network (CNN)

The most prevalent method for detecting and recognizing particular objects in image data is deep learning. CNN is a technique used in deep learning to categorize and identify objects in images data. The reason for selecting it in this research is its similarity to human brain operations from input to output [3]. Futhermore, CNN outperform other methods in terms of accuracy. The method is comprised of multiple layers, including convolutional and pooling layers, which extract features from input image data. Afterward, the flattened image data is fed into the fully-connected layer process that classifies the data and activites it using the softmax layer.

Most researchers have used CNN because it does not require extensive preprocessing if image [20]. The CNN technique may be prone to overfitting, which results from a reduced number of datasets, leading to lower accuracy and longer processing time. For this reason, CNN implements a softmax layer that aids in binary and multi-classification to mitigate the possibility of overfitting [21].

2.7 Visual Geometry Group (VGG)

In order to enhance the accuracy of image classification, this study utilizes VGG-19, which is one of the models from the VGG family developed by Andrew Zisserman and Karen Simonyan specifically for the 2014 ImageNet Challenge. VGG-19 is preferred over AlexNet due to its superior generalization abilities. The VGG model utilizes multiple convolutional layers with small filters and pooling layers to enhance image classification effectiveness. To minimize backpropagation errors and enhance the network's perfomance, VGG-19 utilizes ReLU activation function. The feature extraction process in VGG-19 involves five convolution stages and five max-pooling layers [8], [20]. The final layer or fully-connected network, employs a softmax layer to prevent overfitting.

VGG-19 is chosen for this research due to its improved network depth compared to traditional CNN. It employs a structure consisting of multiple convolutional layers and non-linear activation layers, which is superior to a single convolution layer in extracting image features. Additionally, VGG-19 utilizes max pooling for downsampling and employs the Rectified Linear Unit (ReLU) as the activation function. This activation function selects the highest value within a specific image area as the pooled value. The downsampling layer is primaly utilized to enhance the network's ability to resist distortion in images while retaining the key features of the samples and reducing the number of parameters.

3. Results and Discussion

In a prior study [2], the RoCoLe dataset originally contains 1560 raw images, but only 300 were utilized. The images were preprocessed and sorted into three categories: health, red spider mite, and rust, with 100 images each. The study found that using 240 images for training data resulted in an accuracy 80.56%, while using the remaining 60 images for validation data yielded 81.60% accuracy use AlexNet model.

The objective of this research is to conduct a comparative analysis of two distinct CNN architectures – VGG-19 and ResNet50. The reason the authors included the ResNet50 model only in the results and discussion chapter is if there's problems with gradients, the model can remove redundant network structures to speed up data processing. Besides that, ResNet50 can solve problems with gradients by adding output from the previous layer to the next layer. Therefore, the authors conducted a comparison between the ResNet50 model and the VGG-19 model. The VGG-19 model presents several benefits when compared to both ResNet50 and the model utilized in the primary research publication, AlexNet. VGG-19 employs a greater number of convolutional layers with small filters and layer pooling to enhance the efficiency of image classification. Moreover, VGG-19 utilizes the ReLU activation function to minimize errors in backpropagation and enhance network performance. To prevent overfitting, VGG-19 incorporates a softmax layer as its final layer.

In 2015, Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun created the ResNet50 model for the ILSVRC-2015 competition which is specifically engineered to solve gradient-related difficulties by incorporating skip connections. This connections involve the addition of the output from the previous layer to the subsequent layer, aiding in the optimization and training of deep neural networks. This approach helps to alleviate the degradation problem that can occur in deep neural networks and enables the model to be trained more effectively. Futhermore, it addressed the problem of multi-layer non-linearity in CNN models [23]. The ResNet50 architecture offers certain advantages, such as removing redundant network structures to expedite data processing, which can cause issues with gradients. It also resolves the problem of degradation in convetional CNN networks [24].

In this study, the images data is splitting into training data with ratio 80%, and validation data with ratio 20%, following the methodology of the previous studies. The researchers incorporated callbacks, checkpoints, and early stopping into their approach. Checkpoints help reduce CNN load by storing the model parameters as the accuracy of the matrix increases. Early stopping is used to halt the process if there is an increase in errors in the validation data, maintaining the model parameters. Tensorflow callback parameters instead of the latest values after training [25].

The subsequent action involves transforming the scale of the initial image data into grayscale. This procedure enhances the contrast to develop a unique object feature, namely grayscale. By converting the image from color or grayscale to binary, it simplifies and accelerates the image processing process. Moreover, this step may also improve the evaluation's accuracy results.

Once the original images data has been converted to grayscale, the subsequent action is to apply the drawing contour and find contour functions to draw a contour on top of the original RGB images data. By combining adjacent consecutive lines with the same value, a contour is formed which comprises of straight lines that connect a collection of curves.

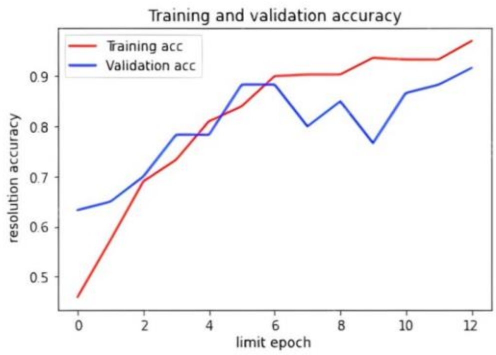
After collecting the images data, the researchers proceed with data augmentation using the image data generator. The images resolution is then reduced to 180 × 180 pixels before feeding it into a CNN for training and evaluation. In order to normalize data, the pixels values for each layer are adjusted to fit within a range of 0 to 1, where 1 represents the maximum pixel value in an RGB image.

Futhermore pada Table 1, the data is trained using two models, VGG-19 and ResNet50, with the RMSprop optimizer, and hyper parameter based on previous study [2] such as momentum value of 0.9, batch size 15, 12 training iterations, and a learning rate of 0.0001 after applying augmentation data. In this study, a learning rate of 0.0001 was used, while the maximum value was 0.006 [26]. The learning rate is a key parameter that can affect how quickly the model converges during training.

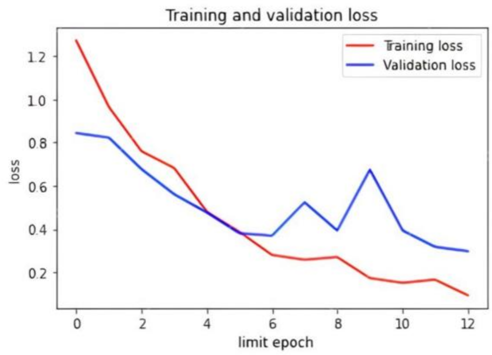
Table 1. Comparison of distribution of previous research datasets with proposed research

CNN Model	Total Training Data Images	Total Validation Data Images
AlexNet	240	60
VGG-19	240	60
ResNet50	240	60

The Figure 5 provides a graphical illustration of the VGG-19 models loss and accuracy. The X-axis of the graph shows the number of iterations employed in training, and Y-axis represents the accuracy during training, which ranges from 0 to 1. The loss graph, on the other hand, displays the number of iterations on the X-axis and the loss magnitude on the Y-axis, which ranges from 0 to 1.2 both graphs depict an accuracy value of 0.90.



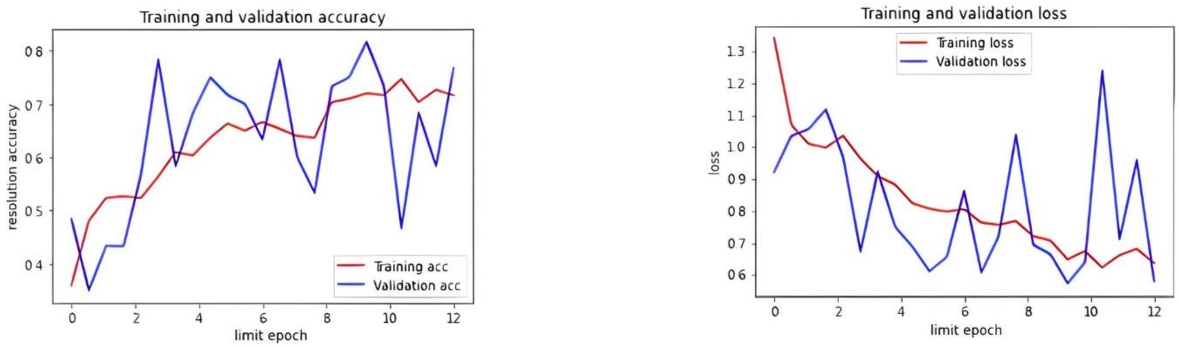
(a)



(b)

Figure 5. VGG-19 (a) accuracy chart, and (b) loss chart

The ResNet50 model's accuracy and loss are depicted in Figure 6. The accuracy graph represents the relationship between the number of iterations during training, displayed on the X-axis, and the corresponding accuracy level, ranging from 0 to 1, depicted on the Y-axis. The X- axis of the loss graph represents the number of iterations during training, while the Y-axis represents the loss magnitude, ranging from 0 to 1.3. The findings from both graphs shows a value of 0.81.



(a) (b)

Figure 6. ResNet50 (a) accuracy chart, and (b) loss chart

Table 2 displays the finding of the VGG-19 classification report, while Table 3 displays the outcomes of the ResNet50 classification report. The VGG-19 validation data yielded an accuracy of 90%, whereas the ResNet50 validation data yielded an accuracy of 80%. When precision approaches 1, the models perfomance is considered good, while when it is closer to 0, the models perfomance is considered poor.

Table 2. Classification Report VGG-19

	Precision	Recall	F1-Score	Support
Health	0.94	0.80	0.86	20
Red Spider Mite	1.00	0.95	0.97	20
Rust	0.85	0.85	0.85	20
Micro Avg	0.93	0.87	0.90	60
Macro Avg	0.93	0.87	0.90	60
Weighted Avg	0.93	0.87	0.90	60
Samples Avg	0.87	0.87	0.87	60

Table 3. Classification Report ResNet50

	Precision	Recall	F1-Score	Support
Health	0.75	0.90	0.82	20
Red Spider Mite	0.86	0.60	0.71	20
Rust	1.00	0.80	0.89	20
Micro Avg	0.85	0.77	0.81	60
Macro Avg	0.87	0.77	0.80	60
Weighted Avg	0.87	0.77	0.80	60
Samples Avg	0.77	0.77	0.77	60

Various matrices, such as accuracy, recall, and f1-score, can be employed to assess the models efficacy. The accuracy of the models predictions is measured based on the precentage of all observations that are positive. The f1-score is higher when a greater number of data predictions are correct for the total observations [27].

The Equation 1, Equation 2, Equation 3, and Equation 4 mentioned earlier can be determined by using the confusion matrix, which helps evaluate the efficiency of deep learning models by comparing their output with the target output. These metrics can aid researchers in creating precise and effective models for studying coffee leaf diseases. This scoring matrices used are True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 - score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (4)$$

The diagonal of the matrix indicates the correctly labeled class categories, while the ff-diagonal elements show the misclassification of the classes into false positives and false negatives, and the correct identification of true negatives and true positives.

Based of the data shown in Figure 7, it can be seen that for the health category, there were 18 cases where the data was correctly identified as positives, 2 cases where it was falsely identified as negatives, 3 cases where it was falsely identified as positive, and 37 cases where it was correctly identified as negative. As for the red spider mite category, there were 19 cases where the data was correctly identified as negative, 0 cases where it was falsely identified as positive, and 40 cases where it was correctly identifed as negative.

In contrast, the rust category has 17 instances of correctly identified positive data, 3 instance of falsely identified negative data, 3 instances of falsely identified positive data, and 37 instances of correctly identified negative data. The results of the confusion matrix indicate that the accuracy of the validation data is remarkable and reaches 90%.

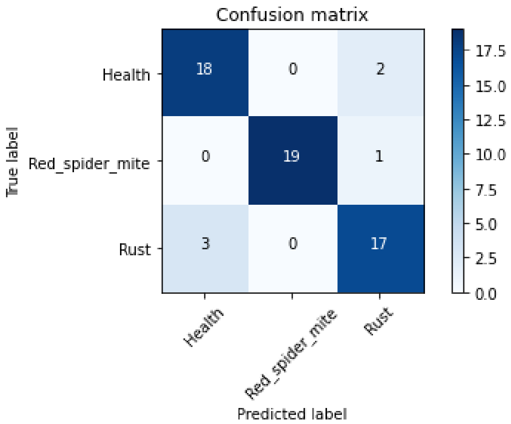


Figure 7. Confusion Matrix of VGG-19

The results shown in Figure 8 indicate that for the health category, there were 19 instances of the correctly identified data, 1 instances of incorrectly identified data, 11 instances of incorrectly data, and 29 instances of correctly identified health data. In the red spider mite category, there were 12 instances of correctly identified data, 8 instances of incorrectly identified data, 2 instances of incorrectly identified data, and 38 instances of correctly identified data.

Lastly, in the rust category, there were 16 instances of correctly identified data, 4 instances of incorrectly identified data, 0 instances of incorrectly identified data, and 40 instances of correctly identified data. Overall, the accuracy of the ResNet50 model in validation data is 81%.

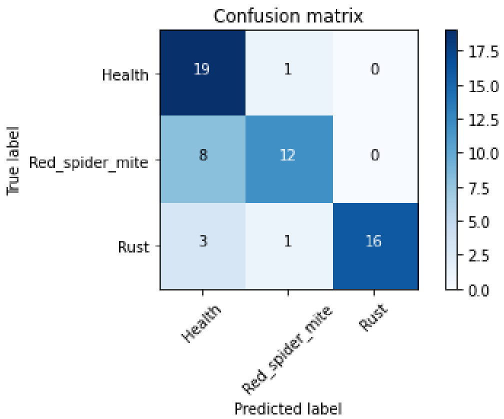


Figure 8. Confusion matrix of ResNet50

Table 4 below presents a comparison of the accuracy, precision, recall, and f1-score of the VGG-19 and ResNet50 models, as conducted by the researchers.

Table 4. Validation data results for accuracy, precision, recall, and F1-score

	VGG-19	ResNet50
Accuracy	0.90	0.81
Precision	0.93	0.85
Recall	0.87	0.77
F1-Score	0.90	0.81

4. Conclusion

This study suggests that python programming language and CNN approach can be utilized to classify coffee plant leaf diseases. The results demonstrate that the proposed model using VGG-19 method accurately identified coffee leaf diseases with a classification accuracy of 90% on validation data as testing data. This study employed 0.0001 learning rate, 15 batch size, 0.9 momentum, RMSprop optimizer, and 12 training iterations.

This study concentrates exclusively on the classification of coffee plant leaf diseases, thus future research can expand the range of the analysis to include other areas of the coffee plant, such as roots, fruits, and other leaves. Further study also can utilize wider range of programming languages and deep learning approaches.

References

[1] BPS, "Statistik Kopi Indonesia," Badan Pusat Statistik Indonesia, 2021.
[2] D. Irfansyah et al., "Arsitektur Convolutional Neural Network (CNN) Alexnet Untuk Klasifikasi Hama Pada Citra Daun Tanaman Kopi," vol. 6, no. 2, pp. 87–92, 2021.
[3] M. Ilhamsyah and U. Enri, "Identification Of Bacterial Spot Diseases On Paprika Leaves Using CNN And Transfer Learning," vol. 18, no. 1, 2022. https://doi.org/10.33480/pilar.v18i1.2755
[4] A. Waheed, M. Goyal, D. Gupta, A. Khanna, and A. Ella, "An optimized dense convolutional neural network model for disease recognition and classification in corn leaf," Comput. Electron. Agric., vol. 175, no. April, p. 105456, 2020. https://doi.org/10.1016/j.compag.2020.105456
[5] M. A. Hasan, Y. Riyanto, and D. Riana, "Klasifikasi penyakit citra daun anggur menggunakan model CNN-VGG16 Grape leaf image disease classification using CNN-VGG16 model," vol. 9, no. December 2020, pp. 218–223, 2021. https://doi.org/10.14710/jtsiskom.2021.14013
[6] A. S. Paymode and V. B. Malode, "Arti fi cial Intelligence in Agriculture Transfer Learning for Multi-Crop Leaf Disease Image Classi fi cation using Convolutional Neural Network VGG," Artif. Intell. Agric., vol. 6, pp. 23–33, 2022. https://doi.org/10.1016/j.aiia.2021.12.002
[7] G. M. Esgario, P. B. C. De Castro, L. M. Tassis, and R. A. Krohling, "An app to assist farmers in the identification of diseases and pests of coffee leaves using deep learning," no. xxxx, pp. 1–10, 2021. https://doi.org/10.1016/j.inpa.2021.01.004
[8] N. U. R. Ibrahim, G. A. Y. U. Lestary, and F. S. Hanafi, "Klasifikasi Tingkat Kematangan Pucuk Daun Teh menggunakan Metode Convolutional Neural Network," vol. 10, no. 1, pp. 162–176, 2022.
[9] F. Teknologi, I. Universitas, and J. Barat, "Klasifikasi Jenis Citra Daun Mangga Menggunakan Convolutional Neural Network," pp. 223–238, 2020.
[10] R. Sistem and U. M. Malang, "Klasifikasi Penyakit Padi berdasarkan Citra Daun Menggunakan Model," vol. 5, no. 158, pp. 9–11, 2021.
[11] A. Zhang, Z. C. Lipton, M. Li, and A. J. Smola, "Dive into Deep Learning," 2022.
[12] A. Luque-chang, E. Cuevas, M. Pérez-cisneros, F. Fausto, A. Valdivia-gonzález, and R. Sarkar, "Knowledge-Based Systems Moth Swarm Algorithm for Image Contrast Enhancement," *Knowledge-Based Syst.*, vol. 212, p. 106607, 2021. https://doi.org/10.1016/j.knosys.2020.106607
[13] J. V. C. I. R, "End-to-end single image enhancement based on a dual network cascade model q," *J. Vis. Commun. Image Represent.*, vol. 61,pp. 284–295, 2019. https://doi.org/10.1016/j.jvcir.2019.04.008
[14] V. Tyagi, "Understanding Digital Image Processing," no. November, 2018. https://doi.org/10.1201/9781315123905
[15] D. M. Hakim *et al.*, "Convolutional Neural Network untuk Pengenalan Citra Notasi Musik," vol. 18, no. 3, pp. 214–226, 2019.
[16] I. Type, R. Bruno, E. Jorge, S. Reader, and C. Science, "Detection of Brain Tumor in Magnetic Resonance Imaging (MRI) Images using Fuzzy C-Means and Thresholding Detection of Brain Tumor in Magnetic Resonance Imaging (MRI) Department of Computer Sciences Utica , New York In Partial fulfillment Of the Requirements of the Master of Science Degree," 2023.
[17] N. Modrzyk, *OpenCV on the JavaVM*.
[18] E. Mehdi, M. Lachgar, H. Hrimech, and A. Kartit, "Arti fi cial Intelligence in Agriculture Optimization techniques in deep convolutional neuronal networks applied to olive diseases classi fi cation," *Artif. Intell. Agric.*, vol. 6, pp. 77–89, 2022. https://doi.org/10.1016/j.aiia.2022.06.001
[19] E. Bisong, *Building Machine Learning and Deep Learning Models on Google Cloud Platform*.
[20] S. Cheng and G. Zhou, "Facial Expression Recognition Method Based on Improved VGG Convolutional Neural Network," vol. 34, no. 7, pp. 1– 16, 2020. https://doi.org/10.1142/S0218001420560030
[21] N. Krishnamoorthy, L. V. N. Prasad, C. S. P. Kumar, B. Subedi, H. Baraki, and V. E. Sathishkumar, "Rice leaf diseases prediction using deep neural networks with transfer learning," *Environ. Res.*, vol. 198, no. May, p. 111275, 2021. https://doi.org/10.1016/j.envres.2021.111275
[22] Keras, "About Keras."
[23] R. Gabriela and C. Dobre, "ResNet interpretation methods applied to the classification of foliar diseases in sunflower," *J. Agric. Food Res.*, vol. 9, no. 313, p. 100323, 2022. https://doi.org/10.1016/j.jafr.2022.100323
[24] Z. Yan, H. Liu, T. Li, J. Li, and Y. Wang, "Two dimensional correlation spectroscopy combined with ResNet: Efficient method to identify bolete species compared to traditional machine learning," *LWT*, vol. 162, no. December 2021, p. 113490, 2022. https://doi.org/10.1016/j.lwt.2022.113490
[25] N. Razfar, J. True, R. Bassiouny, V. Venkatesh, and R. Kashef, "Weed detection in soybean crops using custom lightweight deep learning models," *J. Agric. Food Res.*, vol. 8, no. May 2021, p. 100308, 2022. https://doi.org/10.1016/j.jafr.2022.100308
[26] A. Beikmohammadi, K. Faez, and A. Motallebi, "SWP-LeafNET : A novel multistage approach for plant leaf identification based on deep CNN," *Expert Syst. Appl.*, vol. 202, no. May, p. 117470, 2022. https://doi.org/10.1016/j.eswa.2022.117470
[27] A. Sagar and D. Jacob, "On Using Transfer Learning for Plant Disease Detection," no. July, 2020. http://dx.doi.org/10.13140/RG.2.2.12224.15360/1