Robusta coffee leaf diseases detection based on MobileNetV2 model

Yazid Aufar, Tesdiq Prigel Kaloka
Informatics Engineering, Politeknik Hasnur, Barito Kuala, Indonesia

Article Info

Article history:

Received Oct 21, 2021
Revised Jun 17, 2022
Accepted Jul 10, 2022

Keywords:

Convolutional neural network
Deep learning
Image detection
MobileNetV2
Plant diseases recognition

ABSTRACT

Indonesia is a major exporter and producer of coffee, and coffee cultivation adds to the nation's economy. Despite this, coffee remains vulnerable to several plant diseases that may result in significant financial losses for the agricultural industry. Traditionally, plant diseases are detected by expert observation with the naked eye. Traditional methods for managing such diseases are arduous, time-consuming, and costly, especially when dealing with expansive territories. Using a model based on transfer learning and deep learning model, we present in this study a technique for classifying Robusta coffee leaf disease photos into healthy and unhealthy classes. The MobileNetV2 network serves as the model since its network design is simple. Therefore, it is likely that the suggested approach will be deployed further on mobile devices. In addition, the transfer learning and experimental learning paradigms. Because it is such a lightweight net, the MobileNetV2 system serves as the foundational model. Results on Robusta coffee leaf disease datasets indicate that the suggested technique can achieve a high level of accuracy, up to 99.93%. The accuracy of other architectures besides MobileNetV2 such as DenseNet169 is 99.74%, ResNet50 architecture is 99.41%, and InceptionResNetV2 architecture is 99.09%.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Yazid Aufar
Informatics Engineering, Politeknik Hasnur
Barito Kuala, Kalimantan Selatan, Indonesia
Email: yazid.aufar.ya@gmail.com

1. INTRODUCTION

Coffee is one of the exports that generate foreign currency for the country. In the last decade, the export volume of coffee has fluctuated between (-) 40.15 percent and 12.82 percent. In 2011, the overall export volume was 346.49 thousand tons; by 2020, it will reach 373.35 thousand [1]. After Brazil, Indonesia is the second biggest exporter of coffee globally [2]. As a country with a high level of coffee exports, it is anticipated that Indonesia would be able to sustain the division's earnings by expanding coffee output consistently. Unfortunately, production and efficiency in commodity farming are already relatively low. Farmers who neglected plant cultivation, the agroecosystem, and the minimal degree of integrated pest management in the plantation sector have contributed to the fact that pests and plant diseases cause the majority of pesticide-related losses in this area. In recent years, incredible advancements have been made in diagnosing illnesses, including the prediction, identification, and quantification of many diseases via computer vision and artificial intelligence [3]. Deep learning (DL) is a subfield of artificial intelligence and machine learning that has recently shown remarkable performance on well-known computer vision datasets for image [4]–[6], and object detection tasks [7].

Using deep learning with supervised learning methods is one method for detecting coffee leaf disease. DL in agriculture aspires to surpass present human skills by providing a faster, more accurate, and more efficient method of identifying plant diseases [8], [9]. Deep learning architecture, namely convolutional neural network (CNN), has shown remarkable performance in image classification [10]–[12]. The CNN model was used to diagnose diseases on tomato leaves in the most recent research, which yielded an accuracy score of 95% with a batch size of 16 and a total loss of 0.1265 utilizing only 4,671 images of tomato leaves. The results demonstrated that CNN has a strong potential for detecting plant diseases [13]. CNN was also used to detect leaf disease in coffee in India by adding a data augmentation technique with an accuracy of 97% [14]. Many researchers have developed improved CNN architecture in recent years. It is essential to look into how well the new CNN design model for classifying Robusta coffee leaf disease works.

This paper will research the investigated model class, which has 1,560 leaf images split into six classes (rust level 1 to 4, red spider mite, and healthy), three classes (rust, red spider mite, and healthy), and two classes (unhealthy and healthy). MobileNetV2, ResNet50, DenseNet169, and InceptionResNetV2 are the four kinds of CNN architecture used in the model. In order to determine which CNN design offers the most effective method for detecting Robusta coffee leaf disease, a performance study will be carried out using all of the CNNs. The investigation is broken down into three sections: the second discusses the method used in the investigation, which covers the dataset, the preprocessing of the dataset, and the suggested model architecture. The third part covers the discussion of the model's performance assessment and the comparison of the model to other current state-of-the-art models in this sector. The fourth and last part presents the conclusion and some recommendations for the future.

J. Parraga-Alava, K. Cusme, A. Loor, and E. Santander, “RoCoLe: A robusta coffee leaf images dataset for evaluation of machine
learning based methods in plant diseases recognition,” Data in Brief, vol. 25, Aug. 2019, doi: 10.1016/j.dib.2019.104414.

2. RESEARCH METHOD

Method for detecting Robusta coffee leaf disease is described in the following sections. Figure 1 shows the research process for better comprehension. The method of research flow begins with data collecting. The data is then pre-processed by splitting it into training, validation, and testing data, followed by the image augmentation process. Following the augmentation phase comes the model building and training phase, followed by the model assessment phase. In this work, the augmentation method is used to maximize information on the training data by producing new training data and applying model disguises so that the model can identify pictures under different settings [15].

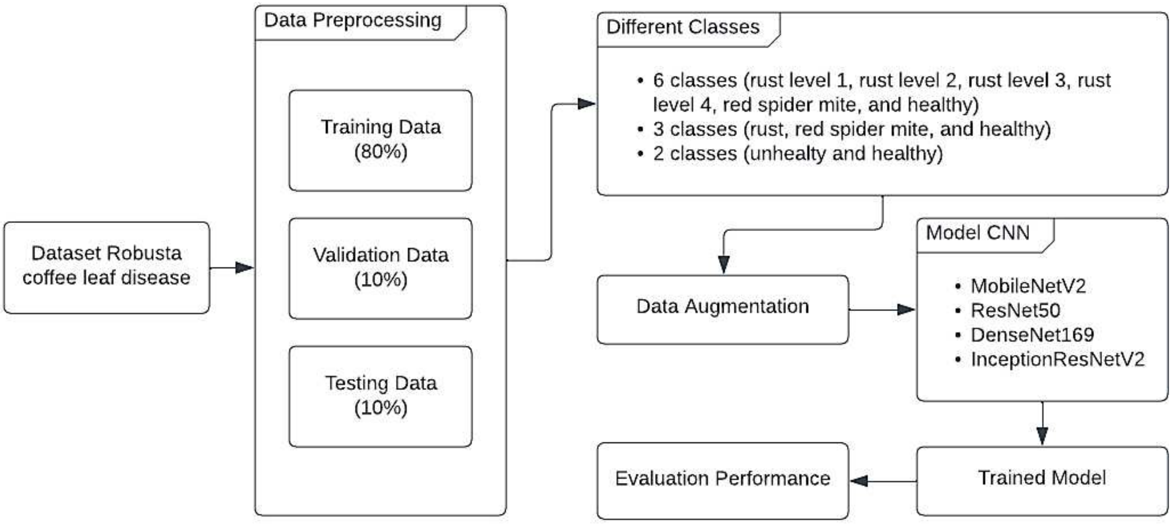
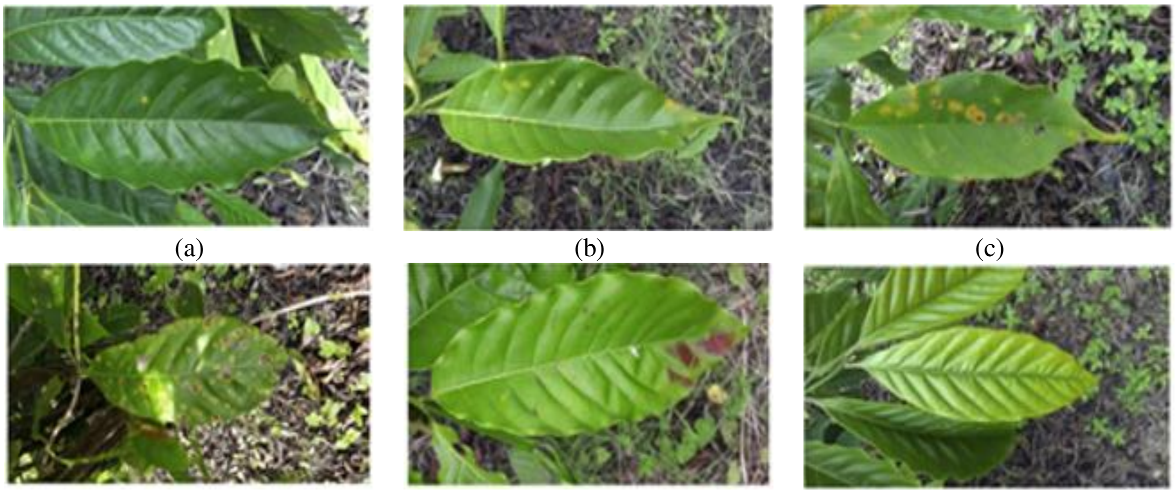


Figure 1. Research flow

2.1. Image datasets

In this study, datasets were collected from Data in Brief [16]. The collection includes 1,560 leaf images with apparent red mites and spots (indicating the presence of coffee leaf rust) for infection instances and images without such structures for health concerns. Each sample photo was taken with a 5-megapixel camera on a smartphone at a working distance of between 200 and 300 millimeters with no zoom. Furthermore, the dataset includes labels for six different classes: rust level 1, rust level 2, rust level 3, rust level 4, red spider mite, and healthy. Red spider mite, rust level 1, rust level 2, rust level 3, and rust level 4 are unhealthy. The total of the unhealthy is 769. Figure 2 shows images of the upper side of leaves from each class. Figures 2(a) to 2(d) show rust levels 1, 2, 3, and 4. Figure 2(e) shows a red spider mite, and Figure 2(f) shows that the plant is healthy. From this sort of picture, the labels are established by visual examination to determine the leaf condition. An expert validates the whole labeling procedure. The datasets labeled for Robusta coffee leaves were separated into training, validation, and testing groups. The training set was used to teach the deep learning model how to work. The approved enhancing methods were validated using the validation set. The improved deep learning model was evaluated using the testing set. Table 1 displays the titles of the classes and the total number of images gathered for each category.



(a) (b) (c)

(d) (e) (f)

Figure 2. Coffee leaf with many classes as an example: (a) rust level 1, (b) rust level 2, (c) rust level 3, (d) rust level 4, (e) red spider mite, and (f) healthy [16]

2.2. Model convolutional neural network

Researchers test convolutional neural networks (CNNs) more often than other deep learning techniques. Yann LeCun originally introduced CNN in 1988 [17]. The development of CNN has altered how image classification issues are handled [18]. MobileNetV2 [19], ResNet50 [20], DenseNet169 [21], and InceptionResNetV2 [22] are four kinds of CNN architecture employed in this paper to identify diseases in Robusta coffee leaves. Following is a concise explanation of the CNN architecture used in this research.

Table 1. A collection of images from various classifications

Class Name	No. of Collected Images
rust level 1	344
rust level 2	166
rust level 3	62
rust level 4	30
red spider mite	167
healthy	791

2.2.1. MobileNetV2

Inverted residual and linear bottleneck modules from MobileNetV1 were combined to produce MobileNetV2, which led to an increase in capacity. Convolution that was depth-separable was used in the MobileNet structure. Convolution in standard two dimensions treats each of the input channels individually to produce just one output channel by convolving the depth dimension (channel). After separating the input picture and the filter into separate channels, the depthwise convolution then convolves each of the input channels with the channel of the filter that best corresponds to it. The output channels are stacked back together once they have been filtered. The stacked output channels are blended into one channel by filtering them using a 1x1 convolution, also known as pointwise convolution, in inseparable depthwise convolution. The output of the depthwise separable convolution is identical to that of the standard convolution. However, the depthwise separable convolution is more efficient since it requires fewer parameters [19].

2.2.2. ResNet50

ResNet-50 is the design of the CNN that employs a residual network and a deep learning procedure. Microsoft developed the ResNet50 or residual network and won 2015 the annual ImageNet large scale visual recognition competition (ILSVRC) competition. ILSVRC is an annual competition where several teams compete to develop the best algorithm for computer vision tasks. ResNet50 is composed of 50 layers deep and over 25.6 million parameters. The set combines convolution, identity block (input=output), and a fully connected layer. The identity x corresponds to the input value of the original block or signal. The output value of the residual block is, therefore, the sum of the input value of the block with the output values of the internal layers of the block [23].

2.2.3. DenseNet169

DenseNet is a well-recognized CNN architecture. Individual layers are linked to each next layer. Thus, all layers are used for decision-making instead of CNN's single final layer. DenseNet was inspired by ResNet18 architecture, one of the greatest deep learning architectures, and has been used in several image classification research. Compared to the ResNet design, fewer connections between layers near the input and layers close to the output have improved training accuracy. The design of DenseNet is built on feed-forwarding connections between each layer and every other layer to prolong the shorter connections. DenseNet improves feature propagation and reduces the number of parameters. DenseNet169 is one of the DenseNet variants. DenseNet is available in variants with 121, 169, 201, and 264 weight layers [24].

2.2.4. InceptionResNetV2

InceptionResNetV2 has found widespread use in various fields, including semantic segmentation, target recognition, picture restoration, and others. It is widely considered among the best DNN frameworks available for image recognition. It is compared to other network models, which results in an increase in prediction performance and an improvement in the amount of time needed for training. The inceptionResNetV2 model is composed of Stem, a sequence of Resnet networks, a residual network, and a softmax classifier. The Stem has many layers of convolution and two layers of pooling. The pooling layers use parallel structures for convolution and pooling to avoid bottlenecks. The Stem is then followed by three different kinds of connections that include 20 convolution layers; the reduction modules within the three different types of convolution layers provide the function of pooling. Parallel architecture is also utilized to reduce bottlenecks. The final component consists of a global average layer (rather than a complete connection layer), a neuron that may be classified into three classifications (through soft-max function), and a dropout layer. The global average dropout and pooling of 0.8 help to avoid overfitting [25].

2.3. Evaluation performance

Conduct an analysis using the model classification techniques indicators to determine how well the waste classification model is performing in the relevant areas. In order to assess the efficacy of the waste classification model, this paper makes use of confusion matrices, accuracy, precision, recall, and F1-score [21]. The term "accuracy" refers to the percentage of the waste classification model that has been appropriately labeled, and it may be written down as (1).

$$Accuracy = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{False Positive} + \text{True Negative} + \text{False Negative}} \quad (1)$$

True positive indicates the number of positive classes that turned out to be positive classes. True negative indicates the number of negative classes that turned out to be negative classes. False positive indicates the number of negative classes that turned out to be positive. False negative indicates the number of positive classes that turned out to be negative. The term "precision" refers to the percentage of samples that tested positive out of the total number of samples anticipated to test positive. This proportion may be expressed mathematically as (2).

$$Precision = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \quad (2)$$

The recall is the percentage of all positive samples that can be accurately anticipated to be positive. This proportion may be expressed as (3).

$$Recall = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \quad (3)$$

The cumulative mean of accuracy and recall is the F1-score:

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

3. RESULTS AND DISCUSSION

This investigation was executed utilizing Tensorflow package and Python programming language. In the experiments that were carried out, a graphics processing unit (GPU) made by NVIDIA called the GeForce GTX 950M as well as 12 GB of DDR4 RAM running at a speed of 2,400 MHz were employed. Accuracy, sensitivity, specificity, and precision were used to assess the performance of the suggested approach. The accuracy reflects the proportion of test data that were properly categorized. The term "precision" refers to the percentage of correct positive class predictions that match the actual correct positive class. The term "recall" refers to the percentage of correct class predictions generated from all positive occurrences included in the training set. F1-score gives a single score addressing precision and recall problems in a single number. In addition, we record the running time and size of each model so that we may compare their efficacy. When training a classification model, an Adam optimizer with a learning rate of 0.001 is used to get a high level of accuracy.

3.1. Comparison of the proposed classes

After the image is trained, it is classified. Cross-validation divides the dataset 80:10:10 for training, validation, and testing data. The MobileNetV2 procedure is used throughout the training phase, which lasts for 50 epochs. The suggested classes were put through their paces using three different scenarios, as illustrated in Table 2. Classification accuracy 78.57% for 2 class classification (unhealthy and healthy), 34% for 3 class classification (rust, red spider mite, and healthy), and 16.67% for the 6-class classification (rust level 1 to 4, red spider mite, and healthy) generated by our proposed model. The comparison clearly shows that those using 2 class classifications provide the best performance while 6 class classifications perform poorly.

Table 2. Evaluation of the effectiveness of the suggested classes

Object	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Size (MB)	Time (Minute)
2 class classification	**78.57**	**53.52**	**100.00**	**69.72**	**34**	**69.25**
3 class classification	34.00	77.08	37.17	50.15	34	16.18
6 class classification	16.67	16.67	3.57	5.88	34	8.00

3.2. Comparison of the methods

In this experiment, we compared our proposed model, MobileNetV2, with other models, namely ResNet50, DenseNet169, and InceptionResNetV2, and then measured its performance on the Robusta coffee leaf disease dataset. We use 2 classification classes, namely healthy and unhealthy. The data before the augmentation technique was 1,538 data with the division of 769 healthy and 769 unhealthy. We perform augmentation techniques to multiply the data and improve accuracy. We experimented with a variety of enhancement methods, including random saturation, random contrast, random brightness, random left-right flip, and random jpeg quality. The goal of data replication in augmentation is to make the classification process more straightforward. Following the use of the augmentation strategy, the number of data grew to 15,380, with a breakdown of 7,690 healthy and 7,690 unhealthy. The data is divided into three parts, 12,304 for training data, 15,38 for validation data, and 15,380 for testing data, using a ratio of 80:10:10. During the course of data training, this experiment aims to identify the required accuracy, precision, recall, F1-score, size, and calculation time on every architectural model. The training epochs for this experiment will total 10 in total. Table 3 shows the accuracy, precision, recall, F1-score, size, and training calculation time of each CNN design. With a value of 0.0573, MobileNetV2 has the lowest training loss. DenseNet169, InceptionResnetV2, and ResNet50 all have values of 0.08174, 0.09990, and 0.21156, respectively. Figures 3 to 6 show the accuracy and loss charts from the training process.

Confusion matrix findings for the MobileNetV2 model with data testing are shown in Figure 7. One out of the 1,538 samples analyzed was incorrectly categorized. Figure 7 shows a single misclassified healthy class data sample. Confusion matrix findings from data testing of the ResNet50 architectural model are shown in Figure 8. Of the 1,538 samples analyzed, nine samples have been incorrectly labeled as such. Figure 8 depicts a sample of nine misclassified healthy class data samples. The DenseNet169 architectural model confusion matrix with data testing is shown in Figure 9. There are four misclassifications among the 1538 sample data evaluated. Figure 9 depicts a misclassification of four data samples from the healthy class. Confusion matrix findings from data testing of the InceptionResNetV2 architectural model are shown in Figure 10. Fourteen of the 1538 samples analyzed were incorrectly categorized. According to Figure 10, there are 14 misclassified data samples.

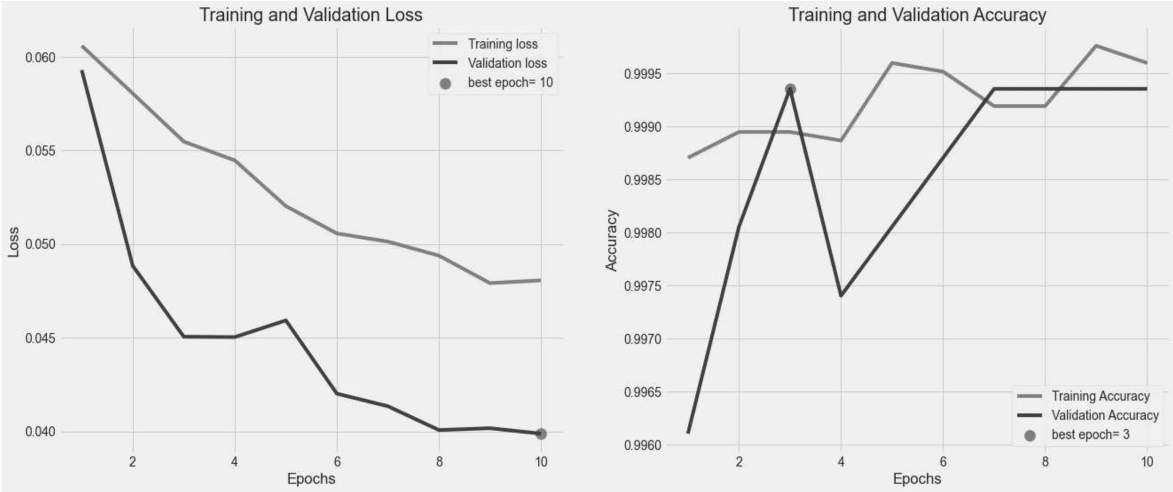


Figure 3. MobileNetV2's accuracy and loss of training

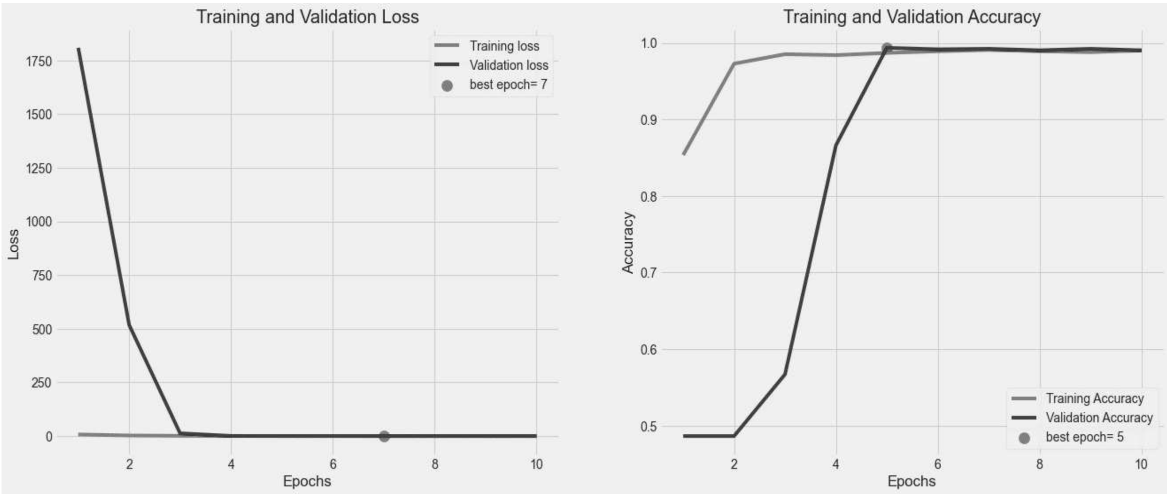


Figure 4. ResNet50's accuracy and loss of training

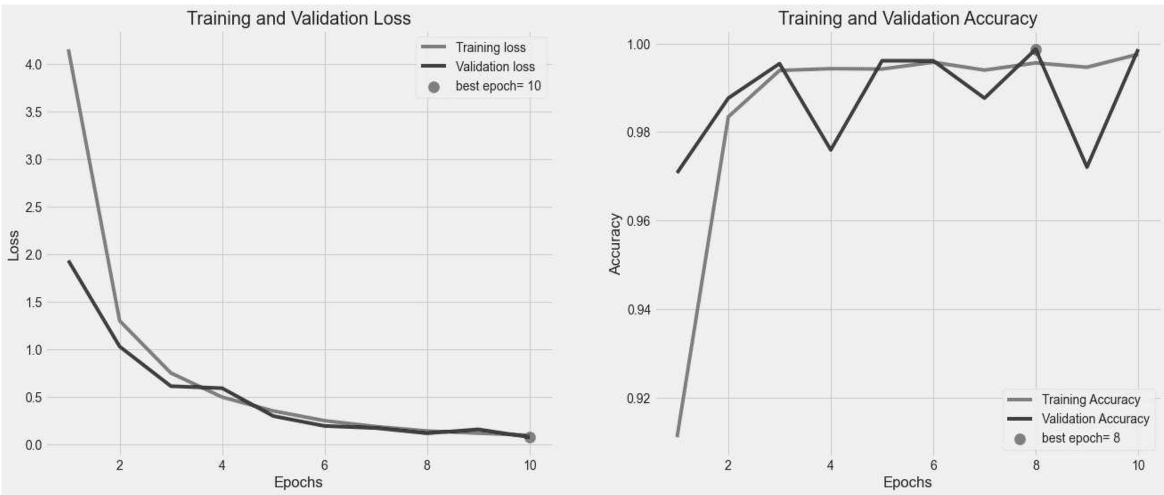


Figure 5. DenseNet169's accuracy and loss of training

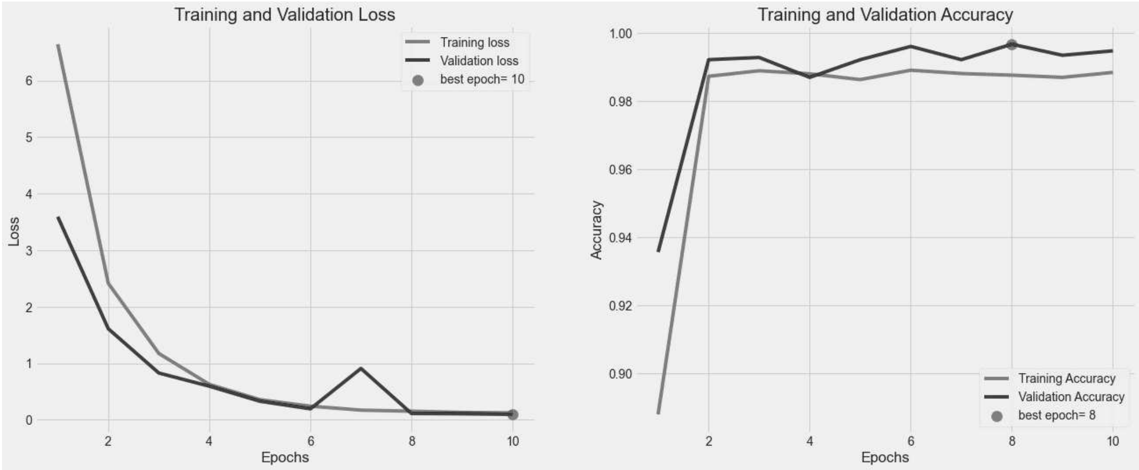


Figure 6. InceptionResNetV2's accuracy and loss of training

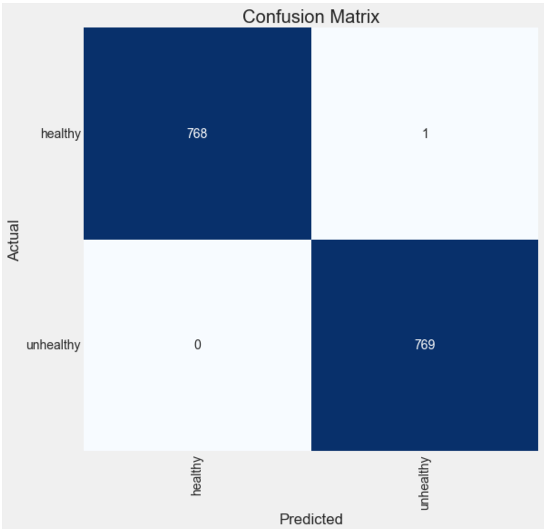


Figure 7. MobileNetV2's confusion matrix

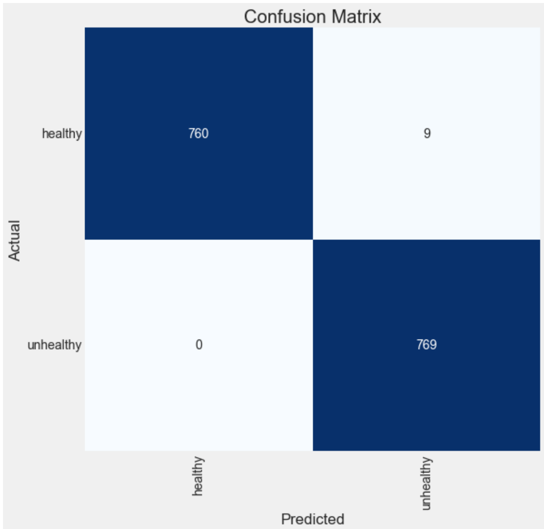


Figure 8. ResNet50's confusion matrix

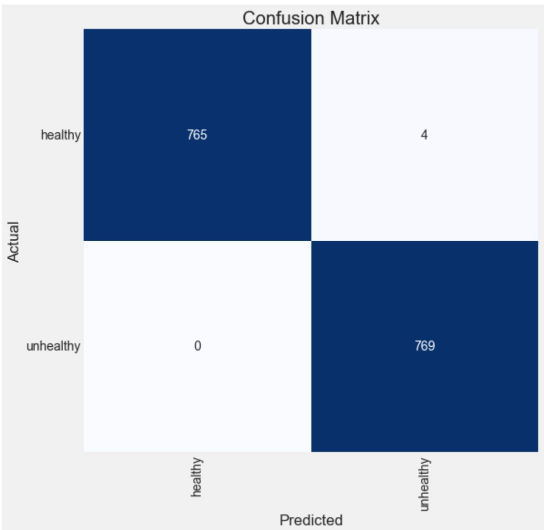


Figure 9. DenseNet169's confusion matrix

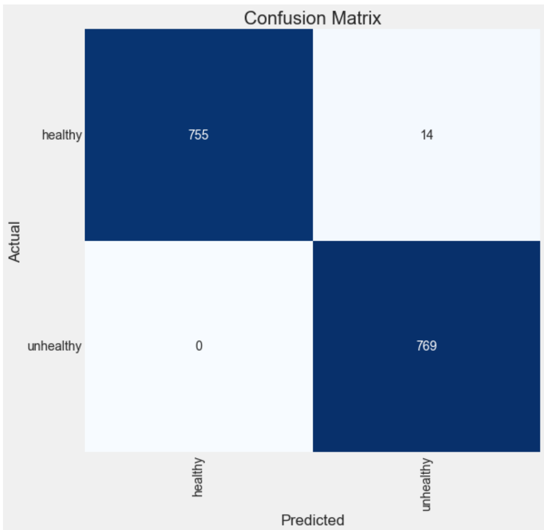


Figure 10. InceptionResNetV2's confusion matrix

A comparison matrix of training data and test data is provided in Table 3. The MobileNetV2 network, the foundational model of the proposed technique, has the best accuracy compared to other networks. In addition, the suggested technique provides the quickest execution time, confirming that MobileNetV2 is one of the lightest network topologies. MobileNetV2 got a score of 99.93% for accuracy, 99.87% for precision, 100% for recall, and 99.93% for F1-score. Then came the DenseNet169, which gave architectures scores of 99.74% for accuracy, 99.48% for precision, 100% for recall, and 99.74% for F1-score. A recall score of 100%, an accuracy score of 99.41%, a precision score of 99.83%, an F1-score of 99.41%, and an inception score of 99.41% are all part of ResNet50's impressive statistics. ResNetV2 has an accuracy value of 99.09%, precision 90.87%, recall 100%, F1-Score 99.93%.

Table 3. Comparison of the varying methods

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	Size (MB)	Time (Minute)
MobileNetV2 (Proposed)	**99.93**	**99.87**	**100.00**	**99.93**	**34**	**595.37**
ResNet50	99.41	98.83	100.00	99.41	282	1440.22
DenseNet169	99.74	99.48	100.00	99.74	156	2009.57
InceptionResNetV2	99.09	99.87	100.00	99.93	633	1689.77

4. CONCLUSION

MobileNetV2 is the best architecture for diagnosing Robusta coffee leaf disease, with a 99.93% accuracy rate, based on an examination of the trial outcomes of each CNN architectural model. DenseNet169 architectural accuracy is 99.74%, ResNet50 architecture accuracy is 99.41%, and InceptionResNetV2 architecture accuracy is 99.09%. In order to carry out this experiment, Adam's optimization was used, and the batch size and learning rate were both changed accordingly. In order to continue this line of research, a CNN model that has faster and more accurate training will need to be developed. It is essential to provide further details on the diseases and insects that may be found on Robusta coffee leaves. Additionally, additional enhancements are required for the suggested strategy to tackle the issue of imbalanced datasets. In addition, the suggested approach has the potential to be further applied on mobile devices to aid in the early diagnosis of Robusta coffee leaf disease

ACKNOWLEDGEMENTS

We want to take this opportunity to thank Yayasan Hasnur Centre for providing the Internal Research Funding that was used to assist this investigation.

REFERENCES

[1] D. M. Yuwono, *Statistika kopi Indoenesia 2020*, 5504006th ed. Jakarta: ©Badan Pusat Statistik/BPS – Statistics Indonesia Baru, 2021.

[2] ICO, “Word coffee consumption,” *International Coffee Organization*, 2021. https://www.ico.org (accessed Sep. 15, 2021).

[3] P. K. Sethy, N. K. Barpanda, A. K. Rath, and S. K. Behera, “Rice false smut detection based on faster R-CNN,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 19, no. 3, pp. 1590–1595, Sep. 2020, doi: 10.11591/ijeecs.v19.i3.pp1590-1595.

[4] S. Chaivivatrakul, J. Moonrinta, and S. Chaiwiwatrakul, “Convolutional neural networks for herb identification: plain background and natural environment,” *International Journal on Advanced Science, Engineering and Information Technology (IJASEIT)*, vol. 12, no. 3, pp. 1244–1252, Jun. 2022, doi: 10.18517/ijaseit.12.3.15348.

[5] D. I. Swasono, H. Tjandrasa, and C. Fatichah, “Classification of air-cured tobacco leaf pests using pruning convolutional neural networks and transfer learning,” *International Journal on Advanced Science, Engineering and Information Technology (IJASEIT)*, vol. 12, no. 3, pp. 1229–1235, May 2022, doi: 10.18517/ijaseit.12.3.15950.

[6] M. S. I. Prottasha and S. M. S. Reza, “A classification model based on depthwise separable convolutional neural network to identify rice plant diseases,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 4, pp. 3642–3654, Aug. 2022, doi: 10.11591/ijece.v12i4.pp3642-3654.

[7] M. Karthikeyan and T. S. Subashini, “Automated object detection of mechanical fasteners using faster region based convolutional neural networks,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 6, 5430, Dec. 2021, doi: 10.11591/ijece.v11i6.pp5430-5437.

[8] S. Pudaruth, M. F. Mahomoodally, N. Kissoon, and F. Chady, “MedicPlant: A mobile application for the recognition of medicinal plants from the Republic of Mauritius using deep learning in real-time,” *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 4, pp. 938–947, Dec. 2021, doi: 10.11591/ijai.v10.i4.pp938-947.

[9] A. Julianto and A. Sunyoto, “A performance evaluation of convolutional neural network architecture for classification of rice leaf disease,” *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 4, pp. 1069–1078, Dec. 2021, doi: 10.11591/ijai.v10.i4.pp1069-1078.

[10] A. Y. Saleh, C. K. Chin, V. Penshie, and H. R. H. Al-Absi, “Lung cancer medical images classification using hybrid CNN-SVM,” *International Journal of Advances in Intelligent Informatics*, vol. 7, no. 2, Jul. 2021, doi: 10.26555/ijain.v7i2.317.

[11] A. Musha, A. Al Mamun, A. Tahabilder, J. Hossen, and B. Jahan, “A deep learning approach for COVID-19 and pneumonia

detection from chest X-ray images," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 4, pp. 3655–3664, 2022, doi: 10.11591/ijece.v12i4.pp3655-3664.
[12] J. O. Pinzón-Arenas and R. Jiménez-Moreno, "Comparison between handwritten word and speech record in real-time using CNN architectures," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 4, pp. 4313–4321, 2020, doi: 10.11591/ijece.v10i4.pp4313-4321.
[13] S. Z. M. Zaki, M. Asyraf Zulkifley, M. Mohd Stofa, N. A. M. Kamari, and N. Ayuni Mohamed, "Classification of tomato leaf diseases using MobileNet v2," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 9, no. 2, pp. 290–296, Jun. 2020, doi: 10.11591/ijai.v9.i2.pp290-296.
[14] M. Kumar, P. Gupta, P. Madhav, and Sachin, "Disease detection in coffee plants using convolutional neural network," in *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, Jun. 2020, pp. 755–760, doi: 10.1109/ICCES48766.2020.9138000.
[15] C. R. Rahman *et al.*, "Identification and recognition of rice diseases and pests using convolutional neural networks," *Biosystems Engineering*, vol. 194, pp. 112–120, Jun. 2020, doi: 10.1016/j.biosystemseng.2020.03.020.
[16] J. Parraga-Alava, K. Cusme, A. Loor, and E. Santander, "RoCoLe: A robusta coffee leaf images dataset for evaluation of machine learning based methods in plant diseases recognition," *Data in Brief*, vol. 25, Aug. 2019, doi: 10.1016/j.dib.2019.104414.
[17] L. Alzubaidi *et al.*, "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *Journal of Big Data*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-021-00444-8.
[18] N. Alrefai and O. Ibrahim, "Deep learning for COVID-19 diagnosis based on chest X-ray images," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 5, pp. 4531–4541, Oct. 2021, doi: 10.11591/ijece.v11i5.pp4531-4541.
[19] R. Indraswari, R. Rokhana, and W. Herulambang, "Melanoma image classification based on MobileNetV2 network," *Procedia Computer Science*, vol. 197, pp. 198–207, 2022, doi: 10.1016/j.procs.2021.12.132.
[20] M. B. Hossain, S. M. H. S. Iqbal, M. M. Islam, M. N. Akhtar, and I. H. Sarker, "Transfer learning with fine-tuned deep CNN ResNet50 model for classifying COVID-19 from chest X-ray images," *Informatics in Medicine Unlocked*, vol. 30, 2022, doi: 10.1016/j.imu.2022.100916.
[21] Q. Zhang, Q. Yang, X. Zhang, Q. Bao, J. Su, and X. Liu, "Waste image classification based on transfer learning and convolutional neural network," *Waste Management*, vol. 135, pp. 150–157, Nov. 2021, doi: 10.1016/j.wasman.2021.08.038.
[22] J. Wang, X. He, S. Faming, G. Lu, H. Cong, and Q. Jiang, "A real-time bridge crack detection method based on an improved inception-resnet-v2 structure," *IEEE Access*, vol. 9, pp. 93209–93223, 2021, doi: 10.1109/ACCESS.2021.3093210.
[23] M. Aatila, M. Lachgar, H. Hrimech, and A. Kartit, "Diabetic retinopathy classification using ResNet50 and VGG-16 pretrained networks," *International Journal of Computer Engineering and Data Science (IJCEDS)*, vol. 1, no. 1, pp. 1–7, 2021.
[24] B. Bayram *et al.*, "Mersin photogrammetry Journal A deep learning integrated mobile application for historic landmark recognition: A case study of Istanbul," *Mersin Photogrammetry Journal*, vol. 2, no. 2, pp. 38–50, 2020.
[25] G. Deng *et al.*, "Image classification and detection of cigarette combustion cone based on inception resnet V2," in *2020 5th International Conference on Computer and Communication Systems (ICCCS)*, May 2020, pp. 395–399, doi: 10.1109/ICCCS49078.2020.9118570.

BIOGRAPHIES OF AUTHORS

Yazid Aufar Obtained his Bachelor Computer degree in Computer Science from Lambung Mangkurat University, South Borneo, Indonesia in 2015. He received his master, in Computer Science from IPB University, Bogor, Indonesia, in 2019. Currently, he teaches Informatics Engineering at Politeknik Hasnur in South Borneo, Indonesia. Image processing, pattern recognition, machine learning, and deep learning are his core research interests. He can be contacted at email: yazid.aufar.ya@gmail.com.

Tesdiq Prigel Kaloka is working as a lecturer in the field of Informatics Engineering at the Politeknik Hasnur in South Borneo, Indonesia. He obtained his Master in Applied Mathematics in 2019 from University of Indonesia, Depok, Indonesia. For the research, he focuses on bioinformatics, data science, and applied mathematics. He can be contacted at email: tesdiq@sci.ui.ac.id.