Jurnal Teknik Informatika (JUTIF)
Vol. 5, No. 2, April 2024, pp. 571-577
DOI: https://doi.org/10.52436/1.jutif.2024.5.2.1926
p-ISSN: 2723-3863
e-ISSN: 2723-3871

LEAF DISEASE DETECTION IN TOMATO PLANTS USING XCEPTION MODEL IN CONVOLUTIONAL NEURAL NETWORK METHOD

Nurhikma Arifin[*1], Maratuttahirah[2], Juprianus Rusman[3], Muhammad Furqan Rasyid[4]

[1]Informatics, Engineering Faculty, Universitas Sulawesi Barat, Indonesia
[2]Information system, Institut Teknologi Bacharuddin Jusuf Habibie, Indonesia
[3]Informatics, Universitas Kristen Indonesia Toraja, Indonesia
[4]Information science, Nara Institute of science and technology, Japan
Email: [1]nurhikma_arifin@unsulbar.ac.id, [2]maratuttahirah.ir@ith.ac.id, [3]rusman.jr@ukitoraja.ac.id

(Article received: March 12, 2024; Revision: May 03, 2024; published: April 15, 2024)

Abstract

This study aims to detect leaf diseases in tomato plants by applying the Xception model in the Convolutional Neural Network (CNN) method. The study categorizes tomato conditions into three main categories: Early Blight, Late Blight, and Healthy. Early Blight is generally infected by specific pathogens that cause spots and damage in the early stages of plant growth, while Late Blight is infected by pathogens in the later stages of the growing season. Meanwhile, the healthy category indicates normal conditions without disease symptoms. The dataset used consists of 300 tomato images, with each category having 100 images. In the model training phase using the fit method in TensorFlow, 17 epochs were performed to teach the model to recognize patterns in tomato leaf disease images in the training dataset. The model testing results on 30 tomato leaf images showed an accuracy rate of 85.84%. This result indicates a positive indication that the developed CNN model performs well in detecting and classifying tomato leaf conditions. Thus, this research can contribute to improving the understanding and management of leaf diseases in tomato plants to support more productive and sustainable agriculture.

Keywords: *Convolutional Neural Network (CNN), Leaf Disease, Tomatoes, Xception model.*

1. INTRODUCTION

The tomato plant is one of the horticultural crops with high economic value and is a significant commodity in the agricultural industry. Despite having the potential for high yields, tomato plants are susceptible to disease attacks, especially leaf diseases that can significantly affect their growth and productivity. The presence of leaf diseases in tomato plants can lead to substantial economic losses for farmers and have a negative impact on the availability of tomato supplies in the market [1].

Leaf diseases in tomato plants can be caused by various factors, such as bacterial, fungal, or viral infections. In maintaining plant health and enhancing crop yields, early identification and proper management of leaf diseases play a crucial role. One type of leaf disease that requires attention is Early Blight, often caused by specific pathogens attacking the plant in its early growth stages. Symptoms like spots on the leaves can serve as early indications of infection, and a profound understanding of this condition enables farmers to take preventive measures early on.

On the other hand, Late Blight is a leaf disease that tends to emerge in later growth stages, causing significant damage to tomato plants towards the end of the growing season. Early identification of Late Blight plays a pivotal role in preventing the spread of the disease and helps farmers implement timely management strategies. Therefore, a thorough understanding of both types of leaf diseases is essential in efforts to improve agricultural productivity and reduce harvest losses.

However, the visual identification process conducted by farmers or agricultural experts requires specific expertise and may result in inaccurate diagnoses. Hence, there is a need for more sophisticated technological solutions to enhance accuracy and efficiency in detecting and managing leaf diseases in tomato plants.

The rapidly advancing technology, particularly in the field of artificial intelligence, has been extensively employed for the early detection of plant diseases through image analysis]. This involves extracting information from an image through object recognition and image classification [2]. A. Purnawati et al. conducted leaf disease detection on rice plants through image analysis by comparing the Decision Tree, Random Forest, Naïve Bayes, SVM, and KNN algorithms. The research results show that the best method among these five methods is KNN with an accuracy value of 87% [3]. Meanwhile, the accuracy result of the research on classifying apple plant diseases from leaf images using Convolutional Neural Network conducted by A. A. Paliwang et al. is 97.1% [4].

Similar research was also conducted by A. Lawi et al., which involved classifying leaf diseases in tomato plants using the CNN method, achieving the highest accuracy of 90.83% [5]. S. Sheila also conducted disease detection on rice leaves based on image processing using the CNN method. The training and testing results in the study showed a testing accuracy of 93.75% with a loss value of 0.3076 [6]. E. P. Gemilang et al., who classified types of diseases on tomato leaves using Convolutional Neural Network. The best accuracy result achieved in the designed model was 90.15%, with a loss rate of 0.3419 [7]. R. Soekarta et al, conducted the classification of ten types of tomato plant diseases, which include Bacterial spot, Early blight, Late blight, Leaf mold, Septoria leaf spot, Spider mite, Target spot, Mosaic virus, and Yellow leaf virus. This classification was achieved through the utilization of the VGG16 model in Convolutional Neural Network (CNN) method, resulting in an accuracy of 98% for training accuracy and 82% for validation accuracy [8]. S. Faisal et al, also conducted the identification of tomato leaf diseases by comparing the CNN and SVM algorithms. The test results indicated that the CNN method outperformed SVM, achieving a higher percentage with an accuracy of 97.5% compared to 95% [9]. A. R. Sujiwanto et al. focused on disease spread in tomatoes, employing VGG, MobileNet, and Inception V3 models, with Inception V3 exhibiting the best performance [10]. A. T. R. Dzaky et al. developed a system for detecting chili plant diseases, achieving over 90% accuracy using AlexNet [11]. H. Bastian et al. automated tomato disease identification with Alexnet, yielding 92.35% accuracy [12]. K. Muchtar et al. designed a portable tool for septoria leaf spot detection in tomatoes, obtaining an average accuracy of 95.89% [13]. A. J. Rozaqi et al. effectively managed potato farming challenges, achieving 95% training and 94% validation accuracy using CNN [14]. C. R. Kotta et al. utilized CNN for tomato disease identification, creating a reliable Android application with 94% gallery and 80% camera image accuracy [15].

Based on the literature review conducted, the utilization of the Convolutional Neural Network (CNN) method in leaf disease detection and classification offers an innovative solution. CNN can process information from images with high accuracy, enabling more precise and efficient identification of leaf diseases in tomato plants. Therefore, this research aims to detect leaf diseases in tomato plants using the CNN method. The study categorizes tomato conditions into three main categories: Early Blight, Late Blight, and Healthy. Thus, this research is expected to provide a positive contribution to the understanding and management of leaf diseases in tomato plants, supporting more productive and sustainable agriculture.

2. RESEARCH METHOD

The model developed for detecting diseases on tomato plant leaves utilizes the Convolutional Neural Network (CNN) method with a transfer learning architecture from the Xception model. Convolutional Neural Networks (CNNs) are a type of deep feed-forward artificial neural network in which the network maintains a hierarchical structure by learning internal feature representations and generalizing features in common image tasks such as object recognition and other computer vision problems [16]. The development is carried out using the Python programming language. The selection of the CNN model is driven by several key factors. Firstly, CNNs are highly effective in feature extraction, allowing them to learn hierarchical representations and patterns from image data. Their ability to maintain spatial hierarchies makes them particularly well-suited for tasks involving images, capturing both local and global dependencies within the input data. CNNs can also achieve high classification accuracy even with minimal preprocessing or segmentation [17]. Additionally, the use of convolutional layers with parameter sharing reduces computational complexity compared to fully connected networks, enhancing efficiency for image-related tasks. CNNs inherently possess translation invariance, enabling them to recognize patterns regardless of their position in the input image. The incorporation of transfer learning, particularly leveraging the pre-trained Xception model, provides the CNN with the advantage of utilizing knowledge gained from previous tasks and domains.

The Xception model was chosen for the CNN architecture because it has a highly efficient architecture and is capable of producing good results in various computer vision tasks. Xception introduces deeper convolution blocks called separable convolutions, which significantly reduce the number of parameters that need to be learned by the model, thereby speeding up the training and inference process. This makes it an ideal choice for transfer learning, where a pre-trained model can easily be adapted to the task of detecting leaf diseases in tomato plants. Additionally, Xception has been proven successful in various computer vision applications, demonstrating consistent and reliable performance across different datasets. Therefore, the combination of CNN method and Xception architecture is expected to provide optimal results in detecting leaf diseases in tomato plants.

2.1. Data Collection

The dataset utilized in this study was sourced from Kaggle, consisting of 300 images of tomato leaves, with 100 images for each category: Early Blight, Late Blight, and Healthy. An illustration of the sample data used for each category is depicted in Figure 1.

Early Blight Late Blight Healthy

Figure 1. Sample data for each category

2.2. Labelling Data

The data labeling process in Convolutional Neural Network (CNN) involves assigning categories or classes to each training data. In this stage, each of the 300 data points used will be labeled accordingly. The example data for each category label is illustrated in Figure 2.

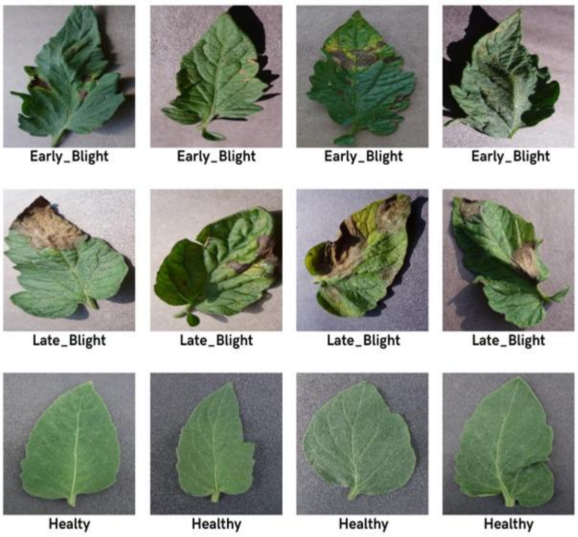

Figure 2. Sample data for each category label

2.3. Preprocessing

The preprocessing stage in this study involves resizing the images to meet the model's input requirements. This process aims to ensure that all images have uniform dimensions, facilitating efficient data processing by the model.

2.4. Modeling and Training

The model for detecting leaf diseases in tomato plants is developed using the Convolutional Neural Network (CNN) method with a transfer learning architecture from the Xception model. The choice of Xception is based on its powerful feature extraction capabilities and successful performance in computer vision tasks. The model architecture involves convolutional layers for feature extraction, max pooling for dimension reduction, dropout to prevent overfitting, and dense layers at the end for classification. The model is designed to understand and distinguish various types of leaf diseases in tomato plants based on visual features acquired during training. Transfer learning with Xception accelerates the model's convergence and enhances its performance on the tomato plant leaf disease dataset.

By leveraging knowledge from ImageNet, the model can recognize common patterns in tomato plants without requiring a large amount of training data. To address overfitting, the model utilizes a dropout layer that randomly ignores some units during training, promoting better generalization. With this approach, the model is expected to provide accurate and reliable leaf disease detection results in various conditions.

2.5. Model Evaluation

After the training is complete, the model is evaluated using the validation dataset to measure its performance and prediction accuracy. This evaluation ensures that the model can generalize well to data not used during training. The trained model is saved to a file with a format that includes the model's name, subject, and accuracy. This process enables the reuse of the trained model for inference on new data without the need for retraining.

The final step involves visualizing the results of the model training and performance graphically. The tr_plot function is used to create plots of the training and validation curves, while print_info is used to print performance information such as misclassified files, confusion matrix, and classification report. These visualizations assist users in interpreting and understanding the model's performance more effectively.

3. RESULT AND DISCUSSION

In this study, the Convolutional Neural Network (CNN) model adopts an architecture that leverages transfer learning from the Xception model. The model comprises a defined number of layers, encompassing convolutional layers, max pooling layers, dropout layers, and a dense layer at the end. The specific configuration of these layers is outlined in Table 1.

Table 1. The specific configuration of these layers

Layet Type	Output Shape	Param #
Xception	(None, 7, 7, 2048)	20861480
conv2d_9	(None, 7, 7, 32)	589856
max_pooling2d_1	(None, 3, 3, 32)	0
dropout_1	(None, 3, 3, 32)	0
flatten_1	(None, 288)	0
dense_1	(None, 3)	867

Table 1 presents the detailed architecture and parameters of the CNN model for detecting tomato leaf diseases, known as "sequential_1," utilizing transfer learning with the Xception architecture. Initially, the Xception layer (Functional) is employed for transfer learning, generating a feature representation of (7, 7, 2048). This is followed by Conv2D convolutional layer (3x3 kernel, ReLU activation) for further feature extraction, succeeded by MaxPooling2D for dimension reduction. A

Dropout layer (0.5 dropout rate) is applied to prevent overfitting. The output is flattened with the Flatten layer and directed to the Dense (fully connected) layer with softmax activation for the classification output of three disease classes on tomato leaves. The total model parameters consist of 590,723 trainable parameters (2.25 MB) and 20,861,480 non-trainable parameters (79.58 MB). This separation reflects the model's complexity and the contribution of transfer learning.

The table provides comprehensive information on the CNN model structure, aiding in understanding the configuration and capacity of the model, and it is expected to be effective in detecting tomato leaf diseases with the robust feature representation from Xception.

The training results of the leaf disease detection model in tomato plants using the CNN method with 17 epochs are presented in Table 2.

Table 2. Training CNN Model using 17 Epochs

Epoch	Loss	Accuracy	Val_loss	Val_accuracy
1	0.4149	0.7833	0.4057	0.8333
2	0.4047	0.8042	0.3946	0.8000
3	0.3418	0.8583	0.4254	0.7333
4	0.2744	0.9083	0.4434	0.8000
5	0.2451	0.9167	0.4336	0.8000
6	0.2403	0.9167	0.3639	0.7667
7	0.2298	0.9125	0.3307	0.8667
8	0.1648	0.9292	0.3710	0.8667
9	0.1896	0.9333	0.4198	0.8333
10	0.1640	0.9375	0.3380	0.8333
11	0.1058	0.9708	0.3869	0.8333
12	0.1527	0.9458	0.4745	0.8000
13	0.1021	0.9667	0.4187	0.8333
14	0.0725	0.9833	0.4424	0.8333
15	0.0974	0.9667	0.4632	0.8667
16	0.0663	0.9833	0.3410	0.8667
17	0.0732	0.9750	0.5568	0.8667

Based on Table 2, 17 epochs were conducted to obtain the best training model by considering the values of loss, accuracy, validation loss, and validation accuracy. Loss functions to measure how much the model's predictions differ from the actual values. The goal during training is to minimize the loss values so that the model can provide more accurate predictions. Accuracy is used to measure how well the model can predict correctly on the training data. Validation Loss (Val_loss) is the loss function value on the validation data to assess how well the model can generalize unseen data during training. Validation Accuracy (Val_accuracy) measures the accuracy of the model's predictions on validation data as the ratio of correct predictions to the total predictions on the validation dataset. If Val_loss remains low and Val_accuracy is high on the validation dataset, it indicates that the model has successfully generalized and can be relied upon for new data. Monitoring Val Loss and Val Accuracy helps adjust the model during training to achieve better results on previously unseen data.

The training results in Table 2 indicate that epoch 7 is the most optimal and will be used in the testing phase, achieving high accuracy on the validation data (Val_accuracy) of 0.8667, and maintaining a low Val_loss of 0.3307. At this point, the model has generalized well and shows minimal signs of overfitting.

The visualization of the Training and Validation Loss on leaf disease detection in tomato plants using convolutional neural network method is depicted in Figure 3.

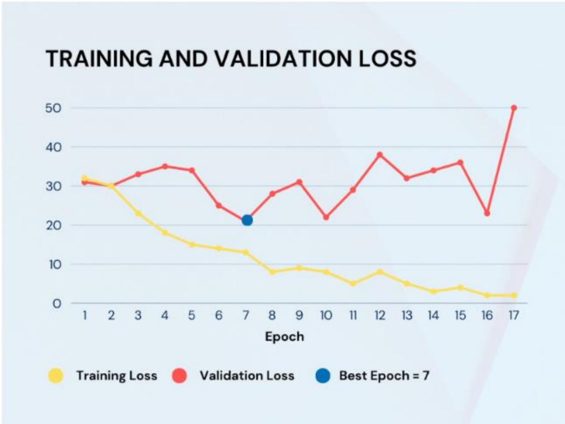

Figure 3. Visualization of the Training and Validation Loss

Based on the graph in Figure 3, the best training and validation loss results are achieved at epoch 7, with a Training Loss of 0.2298 and a Validation Loss of 0.3307. Epoch 7 emerges as the best epoch because it attains a balanced and stable value for both Training Loss and Validation Loss compared to other epochs. This outcome indicates that the model has learned well without overfitting excessively to the training data.

The visualization of the Training and Validation Accuracy on leaf disease detection in tomato plants using convolutional neural network method is depicted in Figure 4.

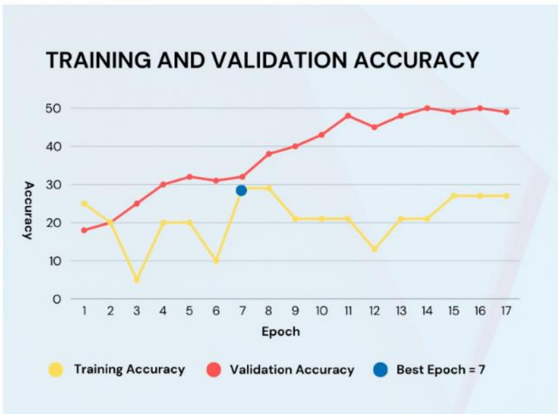

Figure 4. Visualization of the Training and Validation Accuracy

Based on the graph in Figure 4, the best training and validation accuracy results are achieved at epoch 7 with an accuracy of 0.9125 and a validation accuracy of 0.8667. Epoch 7 is considered the best epoch because it achieves high and balanced accuracy and validation accuracy values, indicating that the model can make accurate predictions on both training and unseen validation data. This is crucial to avoid overfitting, where the model focuses too much on the training data's details and fails to generalize well to new data.

The results of the training model in Table 3 are used to assess its performance on a dataset of 30 tomato images to see how well the model can generalize and make accurate predictions on new, unseen data during training. The following are the results of the testing using the Confusion Matrix.

Table 3. Testing results using the confusion matrix

Actual	Predicted			Accuracy
	Early Blight	**Late Blight**	**Healthy**	
Early Blight	**11**	2	0	81.81%
Late Blight	1	**6**	0	85.71%
Healthy	0	1	**9**	90.00%
Total Accuracy				**85.84%**

Based on the testing results presented in Table 3, it was found that the accuracy for the "Early Blight" category is 81.81%. This result is derived from a total of 11 Early Blight tomato data, with 2 misclassifications as the "Late Blight" category. This indicates that although the model successfully identified the majority of the data infected with "Early Blight," there are still some cases where the model misclassified these data as "Late Blight."

On the other hand, the "Late Blight" category achieved an accuracy of 85.71%, where out of a total of 7 data, only 1 was misclassified as "Early Blight." This suggests that the model tends to perform better in recognizing "Late Blight" compared to "Early Blight," but there is still room for improvement in reducing misclassification errors.

Meanwhile, the "Healthy" category obtained the highest accuracy, which is 90%. However, there was one misclassification error on 1 data, which was erroneously identified as "Late Blight." Nevertheless, this high accuracy level indicates that the model is proficient in classifying data that is not infected with the disease.

Thus, the average accuracy obtained from the 30 test data points is 85.84%. Although the overall performance of the model is good, variations in accuracy among disease categories indicate challenges that need to be addressed.

4. DISCUSSION

The testing results reveal varied accuracies for each disease category, with "Early Blight" achieving an accuracy of 81.81%, "Late Blight" reaching 85.71%, and the "Healthy" category attaining the highest accuracy at 90%. However, there were several misclassification errors that need to be noted, particularly in cases where the model erroneously identified data infected with "Early Blight" as "Late Blight," as well as one misclassification error where a "Healthy" data point was incorrectly identified as "Late Blight." Despite the overall good performance of the model with an average accuracy of 85.84%, variations in accuracy among disease categories highlight the challenges that need to be addressed in improving the classification capability of the model.

The comparison between this study and the previous study conducted by R. Soekarta et al., which also focused on leaf disease detection in tomatoes, involves differences in the model architecture. While Soekarta used the VGG16 model with 16 convolutional layers, including 3x3 convolutional layers with padding, this study utilized the Xception model, which has a more complex architecture with separable convolution modules. However, there is a significant difference in the size of the dataset used. Soekarta's research utilized a larger dataset with 10,519 samples, whereas this study only used 300 samples. With a larger dataset size, the VGG16 model achieved higher training accuracy, reaching 98%, but with a lower validation accuracy of 82%. On the other hand, the Xception model in this study, despite using a smaller dataset, still managed to achieve 91% accuracy for training and 86% for validation. Nevertheless, the more complex architecture of Xception may enable it to handle more complex leaf disease variations. Therefore, the choice between the two models depends on the specific preferences and needs of the tomato leaf disease detection project.

This study also compares the results with the research conducted by A. J. Bastari, which achieved a very high accuracy of 99% with a larger dataset of 5632 samples. Meanwhile, this study utilized the Xception model with a significantly smaller dataset, only 300 samples, resulting in an accuracy of 85.84%. This highlights the significant impact of dataset size on model accuracy. Furthermore, differences in model architecture can also affect the final results, although this is influenced by other factors such as data quality and model configuration. Therefore, for future research endeavors, it is crucial to carefully consider both an adequate dataset size and the selection of an appropriate model to attain optimal outcomes in detecting leaf diseases in tomato plants.

The last study, conducted by A. J. Bastari and A. Cherid, also utilized a different model architecture, namely AlexNet, achieving an accuracy of 92.35% with a considerably larger dataset of 16,011 samples compared to the 300 samples used in this study, which had a significant impact on the accuracy results.

Based on the findings of this research and after comparing the results with other studies that also investigated the same subject, it can be concluded that the accuracy results are influenced by many factors that need to be carefully considered.

Firstly, the complexity of leaf diseases such as Early Blight and Late Blight complicates their differentiation due to similar visual symptoms. Both diseases exhibit spots on the leaves that appear almost identical in the early stages, making it difficult for models to distinguish between them. Additionally, variations within the dataset, including differences in lighting conditions and various stages of disease progression in tomato plants, further complicate the classification task. This uncertainty poses challenges for models to accurately differentiate between the existing categories.

Furthermore, limitations in the quantity and diversity of training data are crucial factors. Training datasets lacking various disease variations and environmental conditions make it difficult for models to generalize patterns observed in the training data to unseen test data.

Another highly significant factor is the choice of model architecture used in the CNN method. Using overly complex models or suboptimal hyperparameter settings can result in overfitting or underfitting, negatively impacting the model's performance on test data. Moreover, class distribution imbalances within the dataset can also affect the model's ability to predict less representative classes.

The presence of noise or anomalies in the data is equally significant, as it can affect the consistency and reliability of prediction results. This noise may originate from measurement errors or unexpected environmental differences affecting the tomato leaf images.

Considering all these factors, ongoing efforts to improve the model, add high-quality training data, optimize parameters, and reduce noise or anomalies in the data can help enhance accuracy and consistency in prediction results in subsequent tests.

5. CONCLUSION

Based on the testing conducted on 30 tomato data using the Xception model in Convolutional Neural Network (CNN), with a Loss of 0.2298 and a Validation Loss of 0.3307, the results yielded an accuracy for the "Early Blight" category of 81.81%, "Late Blight" reaching 85.71%, and the "Healthy" category attaining the highest accuracy at 90%. Consequently, for the overall dataset, it can be concluded that the implemented Convolutional Neural Network (CNN) model has successfully achieved an accuracy level of 85.84%. This success demonstrates the model's ability to comprehend and recognize visual patterns present in tomato leaf images. The application of transfer learning techniques using the pre-trained Xception model also contributed positively to the model's performance, leveraging the knowledge acquired by the model from previous tasks. The evaluation results of the model using the Confusion Matrix provide a more detailed overview of the classification performance for each disease class.

Based on the findings of this research and after comparing the results with other studies that also investigated the same subject, it can be concluded that the accuracy results are influenced by many factors that need to be carefully considered, especially the increase in training data, adjustment of hyperparameters, and exploration of more complex and appropriate model architectures.

The importance of considering the quantity of data and the selection of model architecture and parameter adjustments used in the study is crucial as they have a significant impact on accuracy results and model performance. Adequate data volume is necessary for the model to learn effectively and represent various possible cases that may occur in real-world environments. With sufficient data, the model has a greater opportunity to identify relevant patterns and represent the population comprehensively. On the other hand, selecting the appropriate model architecture is also important because each architecture has different strengths and weaknesses. Complex architectures may be required to address complex issues, while simpler ones may be sufficient for simpler tasks.

Furthermore, the parameters used in the model also significantly influence the performance and accuracy of the model. Proper parameter settings can help the model optimize the learning process and improve its ability to generalize to new data. Therefore, by carefully considering these factors, research can achieve better results in classification and prediction, as well as improve the model's ability to adapt to new data.

This conclusion serves as a foundation for further research in plant disease identification using deep learning technology, encouraging continued exploration and development in this field.

REFERENCES

[1] A. J. Bastari and A. Cherid, “Klasifikasi Penyakit Tanaman Tomat Menggunakan Convolutional Neural Network Dan Implementasi Model H5 Pada Aplikasi Desktop,” *J. Sist. Inf. Dan Sist. Komput.*, vol. 8, no. 2, pp. 199–207, 2023, doi: 10.51717/simkom.v8i2.194.

[2] N. Awalia and A. Primajaya, “Identifikasi Penyakit Leaf Mold Daun Tomat Menggunakan Model Densenet121 Berbasis Transfer Learning,” *J. Ilm. Ilmu Komput.*, vol. 8, no. 1, pp. 49–54, 2022, doi: 10.35329/jiik.v8i1.212.

[3] A. Purnamawati, W. Nugroho, D. Putri, and W. F. Hidayat, “Deteksi Penyakit Daun pada Tanaman Padi Menggunakan Algoritma Decision Tree, Random Forest, Naïve Bayes, SVM dan KNN,” *J. Nas. Inform. dan Teknol. Jar.*, vol. 5, no. 1, pp. 212–215, 2020, doi:doi.org/10.30743/infotekjar.v5i1.2934.

[4] A. A. A. Paliwang, M. R. D. Septian, and M. Cahyanti, “Klasifikasi Penyakit Tanaman Apel Dari Citra Daun Dengan Convolutional Neural Network,” *J. Sebatik*, vol. 24, no. 2,

pp. 207–212, 2020.

[5] A. Lawi, N. S. Intizhami, R. Mukhtarom, and S. Amir, “Klasifikasi Penyakit Citra Daun Tanaman Tomat Dengan Ensemble Convolutional Neural Network,” *Semin. Nas. Tek. Elektro Dan Inform.,* vol. 8, no. 1, pp. 239–243, 2022.

[6] S. Sheila, M. K. Anwar, A. B. Saputra, F. R. Pujianto, and I. P. Sari, “Deteksi Penyakit Pada Daun Padi Berbasis Pengolahan Citra Menggunakan Metode Convolutional Neural Network (CNN),” *J. Multinetics,* vol. 9, no. 1, pp. 27–34, Mei 2023.

[7] E. P. Gemilang and C. Lubis, “Klasifikasi Jenis Penyakit Pada Daun Tomat Dengan Menggunakan Convolutional Neural Network,” *J. Ilmu Komput. Dan Sist. Inf.,* vol. 10, no. 1, pp. 1–6, 2022, doi: https://doi.org/10.24912/jiksi.v10i1.17839.

[8] R. Soekarta, N. Nurdjan, and A. Syah, "Classification of Tomato Plant Diseases Using Convolutional Neural Network (CNN) Method," *in Insect Informatics and Security Journal of Information Technology*, vol. 8, no. 2, pp. 143–151, March 2023, doi: 10.33506/insect.v8i2.2356.

[9] S. Faisal, T. F. M. Butarbutar, and P. Sirait, “Implementasi CNN dan SVM untuk Identifikasi Penyakit Tomat via Daun,” *J. SIFO Mikroskil,* vol. 20, no. 2, pp. 117–134, oktober 2019, doi: https://doi.org/10.55601/jsm.v20i2.670.

[10] A. R. Sujiwanto, R. R. B. Pratomo, and P. Palupingsih, “Analisis Perbandingan Performa Model Klasifikasi Kesehatan Daun Tomat menggunakan Arsitektur VGG, MobileNet, dan Inception V3,” *J. Ilmu Komput. Dan Agri Inform.*, vol. 10, no. 1, pp. 98–110, 2023, doi: doi.org/10.29244/jika.10.1.98-110.

[11] A. T. R. Dzaky, “Deteksi Penyakit Tanaman Cabai Menggunakan Metode Convolutional Neural Network,” *Telkom Univ. EProceedings Eng.*, vol. 8, no. 2, pp. 3039–3055, 2021.

[12] H. Bastian, A. Muqoddas, R. A. Pramunendar, and D. P. Prabowo, “Deteksi Penyakit Daun Tomat Menggunakan Konfigurasi Layer pada Metode CNN,” *Sci. Eng. Natl. Semin.*, vol. 8, no. 1, pp. 248–252, 2023.

[13] K. Muchtar, Y. Nurdin, and A. Afdhal, “Pendeteksian Septoria pada Tanaman Tomat dengan Metode Deep Learning berbasis Raspberry Pi,” *J. Rekayasa Sistem Dan Teknol. Inf*., vol. 5, no. 1, pp. 107–113, 2021, doi: https://doi.org/10.29207/resti.v5i1.2831.

[14] A. J. Rozaqi, A. Sunyoto, and R. Arief, “Deteksi Penyakit pada Daun Kentang Menggunakan Pengolahan Citra dengan Metode Convolutional Neural Network,” *Citec J.*, vol. 8, no. 1, pp.22-3, Jan. 2021.

[15] C. R. Kotta, D. Paseru, and M. Sumampouw, “Implementasi Metode Convolutional Neural Network untuk Mendeteksi Penyakit Pada Citra Daun Tomat,” *J. Pekommas*, vol. 7, no. 2, pp. 123-132, Dec. 2022, doi: 10.56873/jpkm.v7i2.4961.

[16] B. Raharjo, *Deep Learning dengan python. Yayasan Prima Agus Teknik*, 2022.

[17] A. Jinan and B. H. Hayadi, “Klasifikasi Penyakit Tanaman Padi Mengunakan Metode Convolutional Neural Network Melalui Citra Daun (Multilayer Perceptron),” *J. Comput. Eng. Sci.*, vol. 1, no. 2, pp. 37–44, Apr. 2022.