SITEKNIK
Sistem Informasi, Teknik dan Teknologi Terapan
E-ISSN: 3032-3991 P-ISSN: 3090-1626
Vol. 2. No. 4 October 2025. Pages. 324-332

Chili Leaf Disease Classification Using Transfer Learning with VGG16 and MobileNetV2 Combined with Random Search Hyperparameter Tuning

Aryawijaya[1]✉, Kusnawi[2]
aryawijaya508@gmail.com[1], khusnawi@amikom.ac.id[2]
[1,2] Informatics, Faculty of Computer Science, University of Amikom Yogyakarta, Indonesia

Keywords: Chili Plant, Machine Learning, CNN, Transfer Learning, MobileNet V2, VGG16, Random Search

Submitted: 02/09/2025
Revised: 23/09/2025
Accepted: 24/09/2025

Abstrak

Chili is one of the main food commodities in Indonesia with considerable economic value. Frequent climate changes have made chili plants more vulnerable to pest and disease attacks. Early identification of these diseases is crucial, as delays can lead to crop failure. However, this process presents its own challenges, as it requires specific expertise and considerable time. This study employs the transfer learning method using the VGG16 and MobileNetV2 architectures to build a model capable of classifying diseases in chili plants based on leaf images, along with the use of Random Search hyperparameter tuning to improve model accuracy. The results show that the use of transfer learning for disease classification achieved high accuracy, with MobileNetV2 reaching an accuracy score of 88% without tuning. Meanwhile, the application of Random Search hyperparameter tuning proved effective in improving model accuracy, particularly with the VGG16 architecture, which saw a significant accuracy increase from 51% to 89%. It can be concluded that the transfer learning method is well-suited for identifying diseases in chili plants based on leaf images with high accuracy, and that the application of Random Search hyperparameter tuning successfully enhanced the model's performance.

Corresponding Author:
Aryawijaya
Informatics, Faculty of Computer Science, University of Amikom Yogyakarta
Jl. Ring Road Utara, Condong Catur, Sleman, Yogyakarta Telp: (0274) 884201-207
Email: aryawijaya508@gmail.com

INTRODUCTION

Indonesia is located along the equator, giving it a tropical climate and fertile soil (Manalu, 2024). This has been utilized by most Indonesians to make the agricultural sector one of the main sources of food commodities (Hasanah, 2022). One of the

This is an open access article under the CC BY 4.0 license

agricultural crops with high market demand is chili, which has significant economic value and is widely needed both domestically and internationally (Asir et al., 2023; Setyadi et al., 2024). According to the official website of the Badan Pangan Nasional (BPN), from January to October 2024, the average price of curly red chili was Rp 32,713, an increase compared to the same period in 2023, which had an average price of Rp 30,534 (Badan Pangan Nasional, 2024). The relatively high selling price of chili compared to other vegetables has led many farmers to prefer growing chili on their fields.

Climate change and high rainfall increase the vulnerability of chili plants to pest and disease attacks, such as armyworms, whiteflies, cotton aphids, thrips, aphids, as well as common diseases like anthracnose, yellow virus, fruit rot, stem rot, and leaf spot (Marianah, 2020; Naura, 2018). The most frequent pest infestation is caused by fruit flies, while the most common disease is anthracnose (Anafiotika et al., 2023). These pest and disease attacks can lead to reduced productivity of chili plants and may even result in crop failure. Currently, disease identification is still performed manually by farmers, which requires specific expertise, time, and experience, making the process inefficient. Therefore, the application of machine learning technology offers a promising solution for automatically detecting diseases in chili plants.

Deep learning is a branch of machine learning that utilizes multilayered neural networks to automatically identify complex patterns in data such as images, text, or audio (Saputra & Krisyanti, 2022). One of its most widely used architectures is the Convolutional Neural Network (CNN), which is designed to process visual data by extracting key features and classifying them based on learned patterns (Setyadi et al., 2024). CNN has shown excellent performance in image recognition tasks, as it mimics the way the human and animal visual cortex processes visual information (Iswantoro & Handayani UN, 2022; Suyanto, 2022). To further improve model performance, transfer learning is often applied. This technique leverages pre-trained models to learn new tasks more efficiently and has been proven to significantly boost accuracy (Vo et al., 2023). Additionally, hyperparameter tuning is another strategy to enhance performance by automatically searching for optimal parameter combinations within a defined range (Saputra & Krisyanti, 2022; Suyanto, 2022).

A related study conducted by Setyadi et al. utilized several variations of transfer learning models, including AlexNet, GoogleNet, VGGNet, and ResNet, to identify diseases in chili leaves. All four models achieved high accuracy, reaching 93%, and demonstrated that the application of data augmentation methods can reduce overfitting by increasing the validation accuracy from 89% to 97%, bringing it closer to the training accuracy (Setyadi et al., 2024).

Another study by Vo et al. on the application of transfer learning and hyperparameter tuning in detecting driver drowsiness achieved an accuracy of over 96%. The architectures used were MobileNetV2, VGG16, and Xception. The three models will be tuned using two different methods: Bayesian Optimization and Random Search. This study aims to find the optimal values for various parameters, including dropout rate, activation function, number of units, optimizer, and learning rate. This study concludes that the use of hyperparameter tuning in training transfer learning models plays an important role in influencing how the model learns from the data by adjusting specific parameters to certain values (Vo et al., 2023).

The research conducted by Muna Ramadhani et al. compared the performance of the MobileNet V2 architecture with a Sequential CNN for detecting chili plant diseases. The model with the MobileNet V2 architecture demonstrated superior performance with an accuracy of 91%, compared to the Sequential CNN which only achieved 84%. The researchers concluded that the reason the MobileNet V2 architecture outperformed the Sequential CNN is because MobileNet V2 has deeper layers, making its hyperparameter settings more complex compared to the Sequential CNN. The researchers also concluded that using appropriate hyperparameter values can affect the model's ability to adapt to training data more quickly, resulting in higher accuracy (Muna Ramadhani et al., 2023).

This study aims to develop a model capable of accurately classifying various types of diseases in chili plants by implementing the transfer learning method due to its computational efficiency and performing fine-tuning using Random Search hyperparameter tuning to improve the model's performance and accuracy.

RESEARCH METHODS

This study utilizes the VGG16 and MobileNetV2 architectures to build a model that can predict leaf diseases in chili plants and uses Random Search to enhance the model's performance.

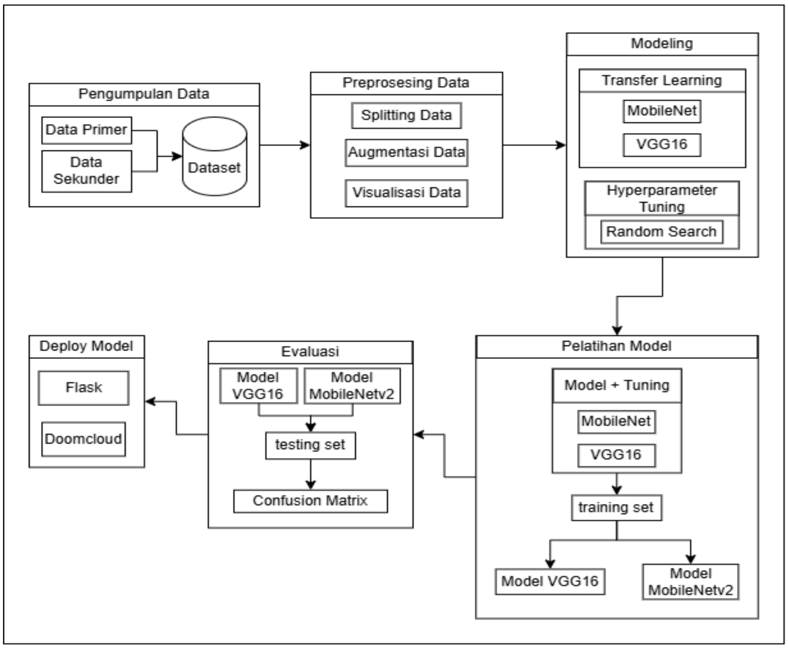


Figure 1. Research Flow

Data Collection

In this study, two types of data were used, namely primary and secondary data. Primary data was obtained from personal observations and documentation in the field. Secondary data was sourced from the Kaggle online dataset website titled “chilli_leaf_dataset,” which contains 5 classes with a total of 6,116 data points, and the “PlantVillage Dataset,” which contains 15 classes with approximately 20,600 data points (Emmanuel, 2018; Hubballi, 2023). Both types of data were combined into a single comprehensive dataset containing approximately 2,800 image data points, divided into five classes: leaf_spot, healthy_leaf, yellow_wish, whitefly, and leaf_curl.

Preprocessing Data

At this stage, the collected data is processed to ensure it is suitable for model training. This study applies three main preprocessing steps: data splitting, normalization, and augmentation. Data splitting divides the full dataset into training, validation, and test sets. Normalization scales pixel values in the images to a specific range to improve training efficiency. Augmentation generates new data by duplicating existing images and modifying attributes such as rotation, horizontal and vertical flipping, as well as zooming in and out. This helps increase dataset variation and reduce the risk of overfitting.

Modeling

This study employs the transfer learning method using the VGG16 and MobileNetV2 architectures. This method was chosen for its efficiency in both computation and data usage. Both models were pre-trained on ImageNet weights, and their final layers were modified to fit the dataset used in this study. Hyperparameter tuning was performed using Random Search, which offers faster computational performance compared to Grid Search.

Model Training

The model was trained in two sessions: the first session without applying Random Search, and the second session with Random Search to optimize the hyperparameters and improve model performance. In the second session, each model was tested five times to determine the best parameter values. The parameters being optimized were the learning rate and dropout rate.

Evaluation

The evaluation stage is the phase where the model's ability to predict diseases is measured. The evaluation method used is the confusion matrix. This method can analyze the model's performance in predicting diseases both overall and for each individual class. The evaluation results can also be used to calculate other metrics such as recall, precision, and F1-score.

Deployment Model

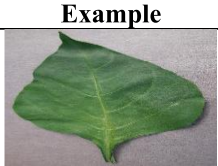
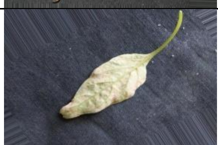
Model deployment is the final stage in the development of a machine learning model, where the trained and evaluated model can be directly used by end-users. One commonly used method for deployment is through a website. This process utilizes the Flask framework in Python and employs Doomcloud to host the website, making it accessible online.

RESULTS AND DISCUSSION

Data Collection

At this stage, two types of data are used: primary data and secondary data. Primary data was obtained through personal observation and documentation conducted directly in the field, while secondary data was sourced from online database websites, namely the “chilli_leaf_dataset” on Kaggle and the “PlantVillage Dataset.” These two types of data were combined into a single, unified dataset consisting of approximately 2,800 image data entries, divided into five classes: healthy_leaf, leaf_curl, leaf_spot, whitefly, and yellow_wish. The following is an explanation of the class distribution:

Table 1. Types of Chili Disease

Class	Example	Explanation
healthy_leaf		Healthy leaves are clean green in color without any spots.
leaf_curl		Leaf curl is a contagious plant disease caused by the Cucumber Mosaic Virus (CMV). As the name suggests, this disease is characterized by curled leaves, stunted plant growth, and smaller leaf size.
leaf_spot		Leaf spot is a disease caused by the Cercospora Capsici fungus, which spreads through wind, rainwater, or insect vectors. It is marked by the appearance of brownish spots on the leaves.
whitefly		Whitefly infestation is a plant disease caused by a pest known as the whitefly, which usually attacks legume crops. This pest is small, white in color, and often sticks to the underside of leaves.
yellow_wish		Yellowish disease in chili plants is caused by the Gemini virus. Infected plants are characterized by yellowing leaves.

Preprocessing Data

At this stage, the collected dataset must first be processed before it can be used for model training. The data will be divided into three parts: training set, testing set, and validation set.

Table 2. Splitting Data

Subset	Amount
Training data	2030 data
Validation data	507 data
Testing data	305 data

The training data will be used to train the model, the validation data will be used to assess how accurately the model predicts during training, while the testing data will be used for the final evaluation stage. The result of the data augmentations process will be applied to the training data. The purpose of applying data augmentation is to address dataset limitations and reduce overfitting. This process works by modifying the training data into new data by altering several image attributes such as rotation, horizontal-vertical flipping, and zooming.

Figure 2. Example of Data Augmented

Figure 2 shows the results of the data augmentation process. It can be seen that the data becomes more varied. Upon closer inspection, there are distinctive features in each class. These specific features are what the model will learn in order to classify the disease based on those characteristics. The augmented data will then be used in the model training process.

Modeling

Two model architectures are used in this study: VGG16 and MobileNetV2. VGG16 was chosen because it is one of the most used architectures in research and offers stable performance, while MobileNetV2 was selected for its lightweight computation and faster training compared to other architectures (citation). Both models take input images of 192 x 192 pixels with 3 channels (RGB) and were previously trained on the ImageNet dataset. Most of the layers are frozen, with only the final layer modified to fit the current dataset, producing an output of five classes. Both models will be tuned using the Random Search method.

Model Training

The two designed models will be trained using the training data. There are two training sessions: one without tuning and one with tuning.

Table 3. Training Session

Session	Learning Rate	Dropout Rate	Optimizer	Trial	Epoch
1	0.0001	0.5	Adam	1	10
2	0.01, 0.001, 0.0001	0.3, 0.4, 0.6, 0.6, 0.7	Adam	5	10

As shown in Table 3, the first session involves training the model once using a learning rate of 0.0001 and a dropout rate of 0.5. In the second session, each model is trained five times using the Random Search method to find the best combination of learning rate and dropout rate values within the specified range.

Table 4. Model Training Accuracy

Model Architecture	Baseline Model		Random Search	
	Train Accuracy	Validation Accuracy	Train Accuracy	Validation Accuracy
VGG16	0.51	0.81	0.89	0.98
MobileNet V2	0.88	0.94	0.98	0.98

Table 5. Model Training Loss

Model Architecture	Baseline Model		Random Search	
	Train Loss	Validation Loss	Train Loss	Validation Loss
VGG16	1.31	1.27	0.29	0.14
MobileNet V2	0.34	0.20	0.05	0.03

From Tables 4 and 5, the performance differences between the two models after being tuned with Random Search. A significant improvement is seen in the VGG16 model, with accuracy increasing from 51% to 89%. Even the MobileNetV2 model, which already had a relatively high accuracy of 88%, experienced an improvement to 98%. This proves that applying Random Search hyperparameter tuning is quite effective in enhancing model performance.

Evaluation

At this stage, the VGG16 and MobileNetV2 models will be evaluated using several methods such as confusion matrix, precision, recall, and F1-score to measure the models' performance in identifying diseases in chili plants. This stage aims to assess and analyze the performance of both models before and after tuning using Random Search. The evaluation results of each model will then be compared to determine which architecture delivers the best performance. The following table summarizes the evaluation process of both models:

Table 6. Model Comparison of Evaluation Results

	Model							
	VGG16				MobileNet V2			
Matrix	**Accuracy**	**Recall**	**Precision**	**F1-Score**	**Accuracy**	**Recall**	**Precision**	**F1-Score**
Baseline Model	0.51	0.84	0.82	0.82	0.88	0.96	0.96	0.96
Tuned Model	0.89	0.97	0.97	0.97	0.98	0.99	0.99	0.99

Based on the evaluation results in Table 6, the MobileNetV2 model outperformed the VGG16 model both before and after tuning. The baseline VGG16 model only achieved an accuracy of 51%, indicating that it was not yet effective in predicting chili diseases across each class, with precision and recall scores of 84% and 82%, respectively. In contrast, the MobileNetV2 model already demonstrated high performance even before being tuned with Random Search. The baseline MobileNetV2 model achieved 88%

accuracy, with precision, recall, and F1-score consistently at 96%, indicating that it could accurately predict chili diseases across all classes.

The implementation of Random Search hyperparameter tuning proved effective in improving model performance. The most significant improvement occurred in the VGG16 model, which increased from 51% to 89% accuracy, with precision, recall, and F1-score all reaching a consistent 97%, indicating a much better capability in predicting diseases across classes. Additionally, the MobileNetV2 model, which already had a relatively high accuracy of 88%, was further improved to 98% after applying Random Search.

Deployment Model

Deployment is the final stage in this research, aimed at enabling the model to be used by other users. The deployment process is carried out using the Flask framework and hosted on Doomcloud.

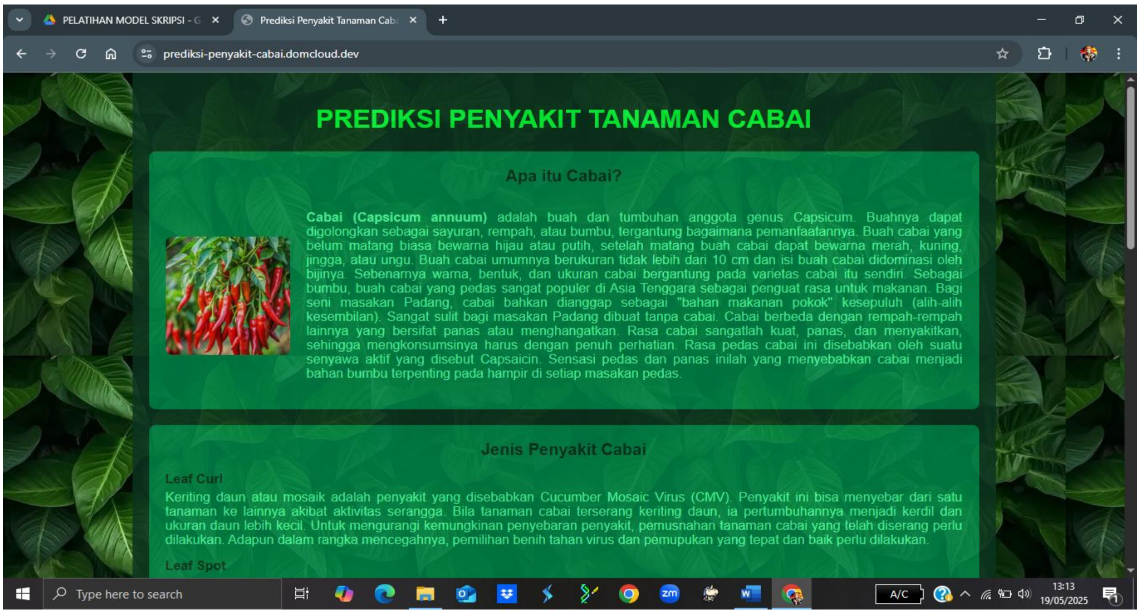


Figure 3. Home Page Website

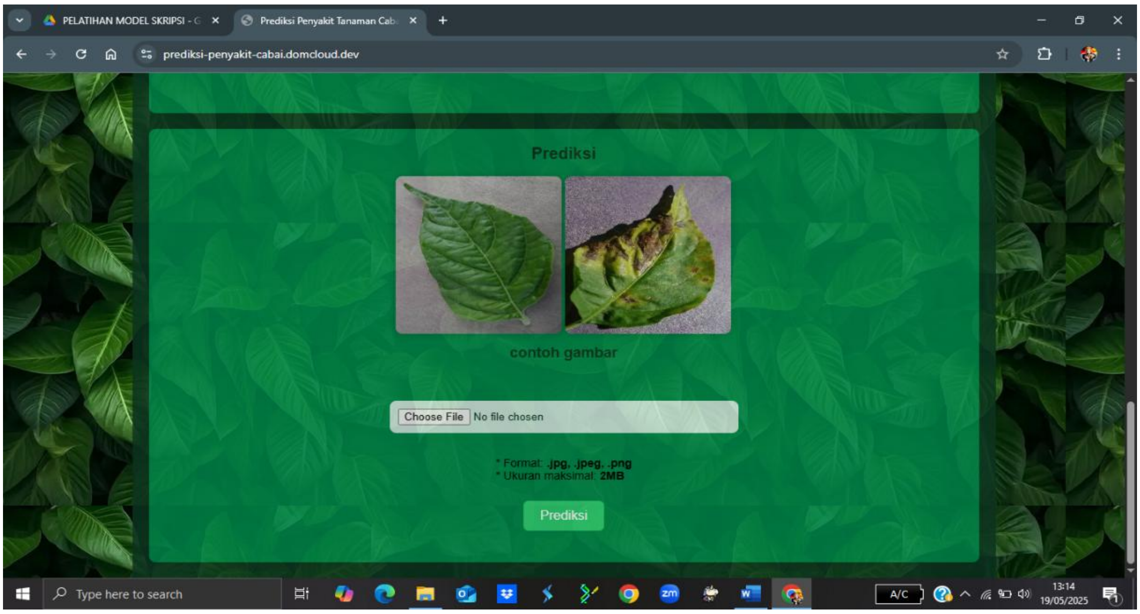


Figure 4. Prediction Page Website

The website includes information about chili plants, various diseases, and how to treat them. The disease prediction feature is located at the bottom section of the website. Users can upload images of chili leaves, following specific requirements: the file must be in .jpg, .jpeg, or .png format and must not exceed 2 MB in size. Once uploaded, the deployed model will predict the type of disease, and the result will be displayed on the website.

CONCLUSIONS AND SUGGESTIONS

Conclusion

The models developed using the transfer learning method achieved relatively high average accuracy, where only the baseline VGG16 model had an accuracy below 60%, while the baseline MobileNetV2, tuned VGG16, and tuned MobileNetV2 models all achieved accuracy levels above 85%. This demonstrates that applying the transfer learning method can be an effective way to enhance model performance. Furthermore, adjusting specific parameter values can influence how the model learns patterns in the data, and the use of the Random Search method proves highly beneficial in automatically determining the appropriate parameter values, allowing the model to be trained with its best parameter combination and achieve high accuracy. Among all the models tested, the MobileNetV2 architecture with tuning delivered the best performance, reaching 98% accuracy and maintaining precision, recall, and F1-score at 99%, indicating the model's excellent capability to classify chili diseases across all classes.

Suggestion

To improve future research, it is recommended to increase the variation of the dataset. In this study, most of the data used had black backgrounds, which may have influenced the prediction results as the model might become overfitted to that pattern. Therefore, adding more diverse backgrounds and incorporating multiple objects per image is suggested so that the model can generalize better when applied to different environments. In addition, enhancing the data augmentation process is important, particularly by introducing adjustments related to brightness and clarity. Considering that field conditions are unpredictable (sometimes sunny and sometimes cloudy), such augmentation can help improve the robustness and accuracy of the model when deployed in real-world scenarios.

Furthermore, since the target users of this system are farmers, it is essential to prioritize practicality, ease of use, and efficiency. For this reason, deploying the trained model on mobile devices is a preferable option, as it would allow farmers to quickly identify chili diseases directly in the field. Mobile-based deployment not only increases accessibility but also supports faster decision-making, thereby contributing to improved productivity and better disease management practices in agricultural settings.

REFERENCE

Anafiotika, R., Suparman, S., Fauziah, Z., Meira Zhafirah, A., Margareta, G., Dwi Rani, F., Wardani, A., Tegar Yusniawan, M., Studi Proteksi Tanaman Jurusan Hama dan Penyakit Tumbuhan, P., & Pertanian, F. (2023). Intensitas Serangan Hama Dan Penyakit Cabai Rawit Di Provinsi Sumatera Selatan. *Seminar Nasional Pertanian Pesisir*, *2*(1), 535–547.

Asir, M., Wahab, A., Yani, N. F., Arum, R. A., & Ramlah, R. (2023). Strategi peningkatan penjualan produk pertanian cabai di Kabupaten Sinjai. *JPPI (Jurnal Penelitian Pendidikan Indonesia)*, *9*(2), 725. https://doi.org/10.29210/020231803

Badan Pangan Nasional. (2024). *Badan Pangan Nasional - Home*. https://badanpangan.go.id/

Emmanuel, T. O. (2018). *PlantVillage Dataset*. Kaggle. https://www.kaggle.com/datasets/emmarex/plantdisease

Hasanah, L. (2022). Analisis Faktor-Faktor Pengaruh Terjadinya Impor Beras di Indonesia Setelah Swasembada Pangan. *Growth: Jurnal Ilmiah Ekonomi Pembangunan*, *1*(2), 57–72. https://e-journal.unimaju.ac.id/index.php/GJIEP/article/view/6

Hubballi, M. (2023). *chilli_leaf_dataset*. Kaggle. https://www.kaggle.com.datasets/mahaninghubballi/chilli-leaf-dataset

Iswantoro, D., & Handayani UN, D. (2022). Klasifikasi Penyakit Tanaman Jagung Menggunakan Metode Convolutional Neural Network (CNN). *Jurnal Ilmiah Universitas Batanghari Jambi*, *22*(2), 900. https://doi.org/10.33087/jiubj.v22i2.2065

Manalu, S. (2024). Dampak Perubahan Iklim Terhadap Produksi Pertanian di Indonesia. *Literacy Notes*, *2*. https://liternot.com/index.php/ln/article/view/169

Marianah, L. (2020). Serangga Vektor dan Intensitas Penyakit Virus pada Tanaman Cabai Merah. *AgriHumanis: Journal of Agriculture and Human Resource Development Studies*, *1*(2), 127–134. https://doi.org/10.46575/agrihumanis.v1i2.70

Muna Ramadhani, A. N., Saraswati, G. W., Agung, R. T., & Santoso, H. A. (2023). Performance Comparison of Convolutional Neural Network and MobileNetV2 for Chili Diseases Classification. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, *7*(4), 940–946. https://doi.org/10.29207/resti.v7i4.5028

Naura, A. (2018). *Dampak Perubahan Iklim Terhadap Produksi dan Pendapatan Usaha Tani Cabai Merah (Kasus Dusun Sumberbendo, Desa Kucur, Kabupaten Malang*. *2*. https://doi.org/10.21776/ub.jepa.2018.002.02.8

Saputra, I., & Krisyanti, D. A. (2022). *Machine Learning Untuk Pemula*. Informatika Bandung.

Setyadi, R. A., Rahman, S., Manurung, D., Hasanah, M., & Indrawati, A. (2024). Implementasi Transfer Learning Untuk Klasifikasi Penyakit Pada Daun Cabai Menggunakan Cnn. *Djtechno: Jurnal Teknologi Informasi*, *5*(2), 304–315. https://doi.org/10.46576/djtechno.v5i2.4642

Suyanto. (2022). *Machine Learning: Tingkat Dasar dan Lanjut Edisi - 2* (2nd ed.). Informatika Bandung.

Vo, H. T., Ngoc, H. T., & Quach, L. Da. (2023). An Approach to Hyperparameter Tuning in Transfer Learning for Driver Drowsiness Detection Based on Bayesian Optimization and Random Search. *International Journal of Advanced Computer Science and Applications*, *14*(4), 828–837. https://doi.org/10.14569/IJACSA.2023.0140492