

# KLASIFIKASI PENYAKIT DAUN TANAMAN BERBASIS CITRA MENGUNAKAN *CONVOLUTIONAL NEURAL NETWORK DATA AUGMENTATION*

Bintang Dyas Suci<sup>1</sup>, Musfiroh<sup>2</sup>, Shintia Putriayu Sefriani<sup>3</sup>, Sumanto<sup>4</sup>, Roida Pakpahan<sup>5</sup>,  
Imam Budiawan<sup>6</sup>

Universitas Bina Sarana Informatika

Jl. Kramat Raya No.98, Jakarta Pusat, DKI Jakarta, Indonesia

[15230763@bsi.ac.id](mailto:15230763@bsi.ac.id), [15230752@bsi.ac.id](mailto:15230752@bsi.ac.id), [15230624@bsi.ac.id](mailto:15230624@bsi.ac.id), [Sumanto@bsi.ac.id](mailto:Sumanto@bsi.ac.id),

[Roida.rkh@bsi.ac.id](mailto:Roida.rkh@bsi.ac.id), [imam.imb@bsi.ac.id](mailto:imam.imb@bsi.ac.id)

## ABSTRAK

Penelitian ini membahas penerapan *Convolutional Neural Network (CNN)* yang dikombinasikan dengan teknik augmentasi data untuk klasifikasi penyakit daun tanaman berbasis citra. Permasalahan utama penelitian ini adalah keterbatasan jumlah data latih yang dapat memengaruhi kinerja model klasifikasi. Tujuan penelitian adalah mengevaluasi efektivitas augmentasi data dalam meningkatkan performa model *CNN* pada dataset berskala terbatas. Dataset yang digunakan adalah *Plant Disease Recognition Dataset* yang terdiri dari 1.523 citra dengan tiga kelas, yaitu *Healthy*, *Powdery Mildew*, dan *Rust*. Penelitian ini menggunakan metode eksperimen dengan tahapan praproses data, augmentasi data, pelatihan model, serta evaluasi performa yang seluruhnya dilakukan menggunakan *Google Colab*. Teknik augmentasi yang diterapkan meliputi rotasi, zoom, dan *horizontal flip*. Hasil penelitian menunjukkan bahwa model *CNN* mampu mencapai akurasi validasi yang baik, meskipun performa klasifikasi antar kelas masih bervariasi, khususnya pada kelas *Rust* yang memiliki karakteristik visual kompleks, sebagaimana ditunjukkan melalui *confusion matrix* dan *classification report*. Selain itu, penelitian ini mengimplementasikan skema prediksi *real-time* sebagai *proof-of-concept*. Secara keseluruhan, hasil penelitian menunjukkan bahwa kombinasi *CNN* dan augmentasi data efektif untuk klasifikasi penyakit tanaman pada kondisi keterbatasan data dan sumber daya komputasi.

**Kata Kunci:** *Convolutional Neural Network*, klasifikasi citra, penyakit daun tanaman, augmentasi data, *deep learning*

## ABSTRACT

*This study investigates the application of Convolutional Neural Network (CNN) combined with data augmentation techniques for image-based plant leaf disease classification. The main problem addressed in this research is the limited size of training data, which can affect classification performance. The objective of this study is to evaluate the effectiveness of data augmentation in improving the performance of a CNN model on a small-scale dataset. The dataset used is the Plant Disease Recognition Dataset, consisting of 1,523 images categorized into three classes: Healthy, Powdery Mildew, and Rust. This research employs an experimental method comprising data preprocessing, data augmentation, model training, and performance evaluation, all conducted using Google Colab. The applied augmentation techniques include rotation, zoom, and horizontal flip. The results show that the CNN model achieves good validation accuracy; however, class-wise performance varies, particularly for the Rust class, which has more complex visual characteristics, as indicated by the confusion matrix and classification report. Additionally, a real-time prediction scheme is implemented as a proof-of-concept to demonstrate the potential application of the model in image-based plant disease detection systems. Overall, the findings indicate that the combination of CNN and data augmentation is an effective and efficient approach for plant disease classification under limited data and computational resource constraints.*

**Keywords:** *Convolutional Neural Network*, image classification, plant leaf disease, data augmentation, *deep learning*

## PENDAHULUAN

Penyakit tanaman menjadi salah satu penyebab utama menurunnya produktivitas dan mutu hasil pertanian di berbagai negara,

termasuk Indonesia. Proses identifikasi penyakit tanaman yang dilakukan secara manual oleh petani atau ahli agronomi sering kali membutuhkan waktu yang lama, bersifat

subjektif, serta berpotensi menimbulkan kesalahan diagnosis (Hamed et al., 2023). Dengan meningkatnya kebutuhan terhadap pertanian presisi dan efisiensi produksi, deteksi otomatis berbasis citra digital menjadi solusi penting untuk membantu proses identifikasi penyakit tanaman secara cepat, akurat, dan berkelanjutan (“Plant Leaf Disease Detection and Classification Using Convolution Neural Networks Model: A Review,” 2025; Sumalatha et al., 2022). Pendekatan berbasis kecerdasan buatan (AI) dan metode deep learning telah digunakan secara luas dalam upaya untuk menjawab tantangan ini berkat kemampuannya dalam dalam mengenali pola kompleks dari citra daun tanaman yang terinfeksi (“Plant Leaf Disease Detection and Classification Using Convolution Neural Networks Model: A Review,” 2025; Yani et al., 2024).

Convolutional Neural Network (CNN) merupakan salah satu arsitektur deep learning yang paling sering digunakan. CNN terbukti efektif dalam beragam aplikasi pengolahan citra, termasuk pada deteksi objek dan identifikasi penyakit tanaman (Lu et al., 2021; Poornam & Devaraj, 2025). CNN mampu mengidentifikasi fitur-fitur utama secara otomatis dari gambar tanpa harus melakukan ekstraksi fitur secara manual terlebih dahulu, sehingga meningkatkan efisiensi serta akurasi system (Selvam & Joy, 2024). Beberapa penelitian sebelumnya menunjukkan bahwa CNN dapat mencapai tingkat akurasi tinggi dalam mendeteksi penyakit tanaman, terutama ketika didukung oleh data yang berkualitas dan teknik pelatihan yang tepat (Ihianle, 2025; Kim et al., 2023). Namun, kinerja CNN masih sangat bergantung pada jumlah data pelatihan dan keberagaman citra yang digunakan, yang sering kali menjadi tantangan di bidang pertanian (Chowdhury et al., 2025; Kanakala & Ningappa, 2025).

Untuk mengatasi keterbatasan data, berbagai penelitian menerapkan data augmentation guna meningkatkan variasi citra latih dan mengurangi risiko overfitting pada model (Kim et al., 2023; Sumalatha et al., 2022). Teknik ini melibatkan transformasi sederhana seperti rotasi, zoom, dan flip horizontal untuk menghasilkan variasi citra baru dari dataset yang sama (Poornam & Devaraj, 2025). Penelitian oleh Prasetyo menunjukkan bahwa augmentasi data secara signifikan

memperbaiki performa CNN pada klasifikasi penyakit tanaman kentang (Zafar et al., 2023). Selain teknik augmentasi, penggunaan platform Google Colab memberikan kemudahan dalam pengolahan data, pelatihan, serta evaluasi model secara efisien berkat dukungan GPU yang kuat dan lingkungan pemrograman berbasis awan (Hamed et al., 2023). Dengan memanfaatkan kemampuan CNN untuk ekstraksi fitur otomatis serta augmentasi data, penelitian ini difokuskan pada pengembangan model klasifikasi penyakit tanaman yang tidak hanya akurat tetapi juga efisien dari sisi komputasi. Pendekatan ini diharapkan mampu memberikan solusi yang dapat diimplementasikan pada sistem pertanian digital berskala luas tanpa memerlukan perangkat keras berbiaya tinggi (Kanakala & Ningappa, 2025).

Tujuan utama penelitian ini adalah mengembangkan sistem klasifikasi penyakit tanaman berbasis CNN dengan augmentasi data yang mampu mendeteksi penyakit daun secara akurat dan cepat. Model dikembangkan menggunakan dataset Plant Disease Recognition yang terdiri dari 1.523 citra dengan tiga kategori penyakit tanaman. Penelitian ini juga menerapkan prediksi waktu nyata (real-time prediction) yang memungkinkan sistem menghasilkan klasifikasi secara langsung terhadap citra yang diunggah. Hasil yang diperoleh model ini dapat dimanfaatkan sebagai fondasi pengembangan sistem pertanian cerdas berbasis visi komputer berkinerja efisien, komputasi yang ringan, sekaligus mudah diaplikasikan di lapangan (Lu et al., 2021; Selvam & Joy, 2024; Zafar et al., 2023).

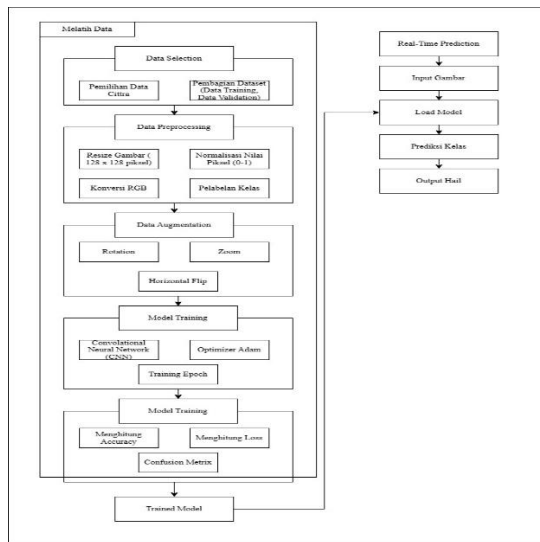
## METODE PENELITIAN

### 2.1 Data Selection

Data Selection merupakan tahap awal penelitian yang berfokus pada pemilihan dataset yang relevan dengan tujuan klasifikasi penyakit tanaman. Penelitian ini menggunakan *Plant Disease Recognition Dataset* dari platform *Kaggle*, berbasis data *PlantVillage* dan diunggah oleh Rashik Rahman Prito (Rahman Prito, 2020). Dataset ini tersedia secara publik dan dapat diakses melalui tautan:

<https://www.kaggle.com/datasets/rashikrahm>

[anpritom/plant-disease-recognition-dataset?resource=download](#).



**Gambar 1. Alur Tahapan Penelitian Klasifikasi Penyakit Daun Menggunakan CNN**

Dataset ini dipilih karena memiliki dokumentasi yang jelas, anotasi kelas yang valid, serta telah banyak digunakan dalam penelitian computer vision terkait penyakit tanaman.

Dataset terdiri dari 1.523 citra yang terbagi dalam tiga kelas:

- a. Healthy: 513 citra
- b. Powdery Mildew: 518 citra
- c. Rust: 492 citra

Distribusi kelas yang seimbang membantu meminimalkan bias selama pelatihan model CNN. Citra berformat JPEG dengan resolusi bervariasi (sekitar 256×256 hingga 1024×1024 piksel), sehingga fleksibel untuk tahap preprocessing dan augmentasi.

Pemilihan dataset ini didasarkan pada beberapa pertimbangan: variasi visual yang cukup (pencahayaan, tekstur, sudut pengambilan gambar), kualitas anotasi yang terpercaya, serta ukuran dataset yang sesuai dengan kebutuhan pelatihan model tingkat menengah. Selain itu, dilakukan pengecekan manual untuk mengidentifikasi citra yang terlalu buram, gelap, atau mengandung objek selain daun, sehingga hanya citra yang layak digunakan pada tahap selanjutnya.

Tahap Data Selection ini memastikan bahwa data yang digunakan memiliki kualitas dan relevansi yang memadai untuk mendukung pelatihan model CNN secara optimal dalam mendeteksi penyakit tanaman.

## 2.2 Data Preprocessing

Pada penelitian ini, metode utama yang digunakan untuk klasifikasi penyakit daun adalah *Convolutional Neural Network (CNN)*. Setelah data citra melalui tahap *preprocessing* dan *augmentation*, CNN berperan sebagai model inti yang mengekstraksi fitur visual dan melakukan pemetaan ke tiga kelas keluaran. Subbab ini menjelaskan secara berurutan arsitektur CNN yang digunakan, konfigurasi proses pelatihan, alur pelatihan model, serta metode evaluasi yang diterapkan untuk menilai kinerja model dalam mendeteksi penyakit tanaman berbasis citra.

Seluruh citra diproses dalam format RGB tanpa konversi warna tambahan untuk menjaga informasi warna yang penting dalam mendeteksi gejala penyakit daun. Selanjutnya, citra diubah ukurannya menjadi 128×128 piksel menggunakan metode stretch resize, yaitu teknik resize yang tidak mempertahankan rasio aspek (non-uniform scaling). Meskipun metode ini dapat menyebabkan distorsi bentuk daun, distorsi tersebut masih dapat diterima karena pola visual utama seperti bercak, perubahan warna, dan tekstur daun tetap dapat dipelajari oleh model CNN.

Setiap citra kemudian dinormalisasi dengan mengubah intensitas piksel dari rentang 0–255 menjadi 0–1 melalui pembagian sederhana. Normalisasi ini bertujuan untuk mempercepat konvergensi model serta menjaga stabilitas perhitungan selama proses pelatihan. Teknik normalisasi tambahan seperti standardization tidak digunakan karena model CNN yang digunakan sudah cukup stabil dengan pendekatan skala 0–1.

Dataset dibagi menjadi dua subset menggunakan pembagian 80% untuk data latih dan 20% untuk data validasi. Pembagian dilakukan secara stratified untuk menjaga proporsi distribusi kelas tetap seimbang. Pembagian ini menjadi konsisten pada setiap percobaan karena seluruh proses pengacakan dikontrol oleh seed yang sama. Struktur folder setelah preprocessing mengikuti standar Keras dengan format:

```

'''
/train/class_name/
/val/class_name/
'''
    
```

Untuk memproses citra, penelitian ini menggunakan kombinasi pustaka *PIL (Pillow)* untuk memuat citra, *OpenCV (cv2)* untuk melakukan proses stretch resize, dan *ImageDataGenerator* untuk manajemen batch serta penyiapan data sebelum masuk ke tahap pelatihan. Kombinasi pustaka ini digunakan untuk memastikan alur preprocessing berjalan efisien dan kompatibel dengan arsitektur CNN yang digunakan.

Dengan preprocessing yang terstruktur serta penerapan seed 42, seluruh eksperimen dalam penelitian ini dapat menghasilkan hasil yang stabil dan dapat direproduksi dengan konsisten.

### 2.3 Data Augmentation

Tahap *Data augmentation* diterapkan untuk memperluas keragaman data latih dan meningkatkan kemampuan generalisasi model CNN. Teknik ini bertujuan menghasilkan variasi citra yang mereplikasi kondisi pengambilan gambar di dunia nyata, seperti perubahan orientasi, jarak kamera, dan posisi objek, sehingga model mampu mempelajari karakteristik penyakit daun secara lebih robust terhadap variasi visual tersebut.

Penelitian ini menerapkan tiga jenis augmentasi utama. Pertama, rotation dengan parameter *rotation\_range* sebesar 30 derajat, yang memutar citra secara acak untuk mensimulasikan variasi orientasi daun pada kondisi pengambilan gambar yang tidak terkontrol. Transformasi ini dianggap paling berpengaruh mengingat gejala penyakit pada daun tetap dapat dikenali meskipun citra mengalami rotasi. Kedua, zoom dengan *zoom\_range* 0.2, yang menghasilkan variasi kedekatan objek terhadap kamera dan membantu model mengenali pola penyakit pada berbagai skala. Ketiga, *horizontal flip*, yang membalik citra secara kiri-kanan dan memberikan tambahan variasi pose objek tanpa mengubah struktur visual penyakit secara signifikan.

Seluruh proses augmentasi diimplementasikan menggunakan Keras *ImageDataGenerator* dan hanya diterapkan pada training set, sedangkan validation set dipertahankan dalam kondisi asli dengan hanya melakukan rescale piksel. Pendekatan ini memastikan bahwa proses evaluasi model tidak terdampak oleh transformasi tambahan

dan tetap merepresentasikan distribusi data yang sebenarnya. Dengan strategi augmentasi ini, model dilatih menggunakan data yang lebih bervariasi, sehingga diperoleh peningkatan stabilitas dan ketahanan model terhadap variasi citra pada tahap prediksi.

## 2.4 Model Training & Evaluation

### 2.4.1 Arsitektur CNN yang Digunakan

Model CNN yang digunakan dalam penelitian ini dirancang dengan arsitektur yang sederhana namun memadai untuk dataset berukuran menengah. Arsitektur terdiri dari dua lapisan Conv2D berturut-turut dengan 32 dan 64 filter, masing-masing diikuti oleh lapisan MaxPooling2D untuk mereduksi dimensi fitur. Setelah proses ekstraksi fitur, citra diratakan menggunakan Flatten dan diproses oleh lapisan Dense dengan 64 neuron, kemudian diberi regularisasi melalui Dropout 0.5 guna mengurangi risiko overfitting. Lapisan akhir berupa Dense (softmax) dengan tiga neuron digunakan untuk melakukan klasifikasi terhadap tiga kelas penyakit daun.

Pemilihan arsitektur ini dipertimbangkan berdasarkan beberapa aspek berikut:

1. Ukuran dataset tidak terlalu besar, sehingga arsitektur ringan lebih sesuai.
2. Waktu pelatihan lebih efisien pada lingkungan komputasi terbatas seperti *Google Colab*.
3. Kompleksitas moderat lebih stabil untuk karakteristik citra penyakit tanaman.
4. Mengurangi risiko overfitting pada dataset dengan jumlah sampel terbatas.

### 2.4.2 Konfigurasi Training

Pelatihan model dikonfigurasi untuk mencapai keseimbangan antara stabilitas pembelajaran dan efisiensi komputasi. *Optimizer* yang digunakan adalah *Adam*, dipilih karena kemampuannya mencapai konvergensi lebih cepat dan menyesuaikan learning rate secara adaptif. Fungsi kerugian yang digunakan adalah *categorical\_crossentropy*, yang sesuai untuk klasifikasi multikelas.

Model dilatih menggunakan *batch size* 32, jumlah maksimum 5 *epoch*, serta diterapkan mekanisme *early stopping* berbasis validation

loss untuk mencegah pelatihan berlanjut ketika model tidak lagi menunjukkan perbaikan. Semua proses acak, termasuk pengacakan batch dan pemilihan sampel, dikendalikan oleh seed 42, sehingga proses pelatihan dapat direproduksi secara konsisten.

#### 2.4.3 Proses Pelatihan

Proses pelatihan dilakukan menggunakan generator data yang telah melalui tahap pra-proses dan augmentasi. Training set disuplai melalui `train_gen`, sedangkan validation set disuplai melalui `val_gen` tanpa augmentasi. Setiap batch citra dinormalisasi ke rentang 0–1 untuk memastikan kesesuaian dengan pipeline komputasi CNN.

Pada setiap *epoch*, loss dihitung terlebih dahulu pada data latih, kemudian model dievaluasi menggunakan data validasi. Pembaruan bobot dilakukan berdasarkan training loss, sementara validation loss digunakan untuk memantau kemampuan generalisasi. Jika validation loss tidak menunjukkan perbaikan dalam beberapa *epoch*, mekanisme *early stopping* menghentikan proses pelatihan untuk mencegah overfitting serta pemborosan sumber daya. Penggunaan seed 42 memastikan seluruh proses berlangsung secara deterministik dan dapat direplikasi.

#### 2.4.4 Evaluasi Model

Evaluasi model dilakukan menggunakan data validasi yang tidak mengalami augmentasi sehingga performa yang diperoleh merefleksikan kemampuan model pada data asli. Selama pelatihan, metrik accuracy dan loss pada data latih dan validasi dipantau untuk mengidentifikasi indikasi awal overfitting.

Setelah pelatihan selesai, evaluasi lanjutan dilakukan menggunakan *confusion matrix* dan *classification report*, yang mencakup metrik *precision*, *recall*, dan *f1-score* untuk setiap kelas. *confusion matrix* digunakan untuk menganalisis kecenderungan kesalahan antar kelas, sedangkan *classification report* memberikan gambaran performa model secara lebih komprehensif pada masing-masing kategori penyakit. Pendekatan evaluasi berlapis ini memastikan penilaian performa model tidak hanya bergantung pada akurasi keseluruhan, tetapi juga pada keseimbangan kinerja di setiap kelas.

### 2.5 Real-Time Prediction

Tahap *Real-Time Prediction* digunakan untuk melakukan inferensi terhadap citra baru menggunakan model CNN yang telah dilatih dan disimpan sebelumnya. Proses ini meniru skenario penerapan model pada lingkungan nyata, di mana sistem harus mampu menerima input tunggal, melakukan preprocessing secara konsisten, dan menghasilkan output prediksi dalam satu siklus komputasi.

Pada tahap ini, citra yang diunggah oleh pengguna pertama-tama dimuat menggunakan fungsi `load_img()` dan dikonversi menjadi *array* numerik melalui `img_to_array()`. Citra kemudian di-resize ke dimensi 128×128 piksel, dikonversi ke format RGB, dan dinormalisasi ke rentang 0–1 menggunakan operasi pembagian skalar. Setelah preprocessing, citra diubah menjadi tensor berdimensi (1, 128, 128, 3), di mana dimensi pertama merepresentasikan *batch size* bernilai satu, sesuai kebutuhan input pada layer pertama jaringan CNN.

Tensor input tersebut diteruskan ke model melalui perintah `model.predict()`. Model kemudian menghasilkan vektor probabilitas berukuran tiga elemen, sesuai jumlah kelas yang dipelajari selama pelatihan. Nilai maksimum dari vektor softmax output diambil sebagai predicted class, sementara nilai probabilitas tertinggi tersebut digunakan sebagai confidence score. Proses inferensi ini terjadi hanya dalam satu forward pass, tanpa pembaruan bobot maupun propagasi balik.

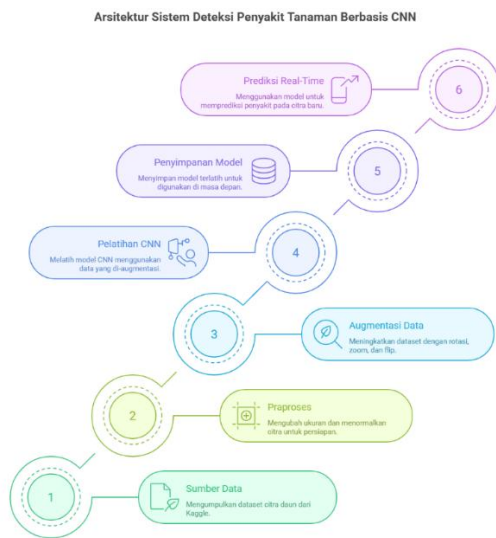
Untuk meningkatkan keterbacaan hasil, citra asli divisualisasikan kembali menggunakan pustaka PIL dan matplotlib, dengan label prediksi dan nilai confidence disematkan secara langsung pada citra melalui fungsi ImageDraw. Langkah ini menunjukkan bagaimana model dapat digunakan dalam pipeline operasional untuk mendeteksi penyakit tanaman secara cepat dan langsung berdasarkan input pengguna.

Tahap *real-time prediction* ini memastikan bahwa model tidak hanya memiliki performa baik pada data validasi, tetapi juga mampu mempertahankan konsistensi prediktif ketika menerima input baru di luar distribusi data pelatihan.

### 2.6 System Architecture

Subbab ini menjelaskan arsitektur sistem deteksi penyakit tanaman berbasis CNN yang

digunakan dalam penelitian. Arsitektur ini menggambarkan aliran data mulai dari sumber dataset, proses praproses dan augmentasi, pelatihan model, penyimpanan model terlatih, hingga tahap inferensi real-time. Representasi ini bertujuan memberikan gambaran menyeluruh mengenai bagaimana setiap komponen berinteraksi dalam pipeline komputasi, sehingga pembaca dapat memahami keterkaitan antara tahapan metodologis dan implementasi teknis sistem secara keseluruhan.



Gambar 2. System Architecture.

## HASIL DAN PEMBAHASAN

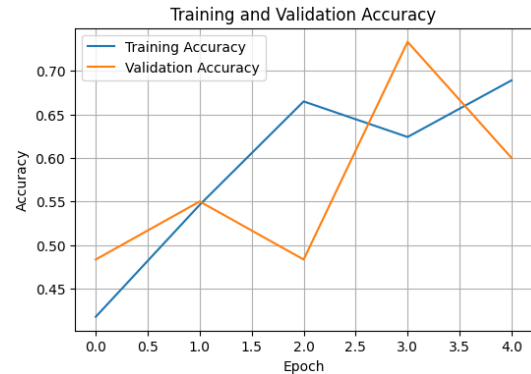
Bab ini menyajikan hasil dari seluruh proses penelitian, mulai dari tahap pelatihan model CNN, evaluasi performa model, hingga pengujian prediksi secara real-time. Seluruh hasil ditampilkan melalui grafik, tabel, dan gambar untuk memberikan gambaran menyeluruh mengenai performansi model. Setelah penyajian hasil, dilakukan pembahasan untuk menjelaskan interpretasi, analisis, serta faktor-faktor yang memengaruhi performa model.

### 3.1 Hasil

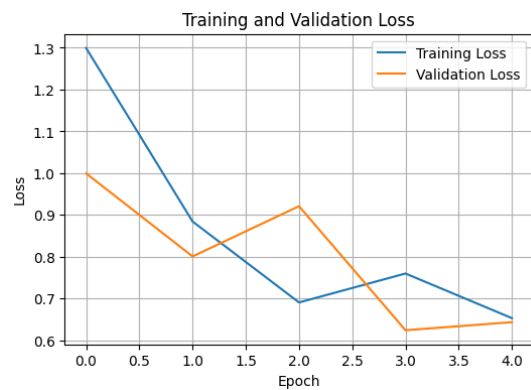
#### 3.1.1 Hasil Training CNN

Proses pelatihan model CNN menghasilkan grafik perubahan akurasi dan nilai loss pada data latih serta data validasi selama 5 *epoch*. Pada Gambar 3 terlihat bahwa akurasi pelatihan mengalami peningkatan bertahap pada setiap *epoch*, diikuti oleh akurasi validasi yang menunjukkan tren serupa dengan nilai

akhir berada pada rentang 80–90%. Pola ini menunjukkan bahwa model mampu mempelajari pola-pola visual pada dataset secara progresif.



Gambar 3. Grafik training dan validation accuracy model CNN



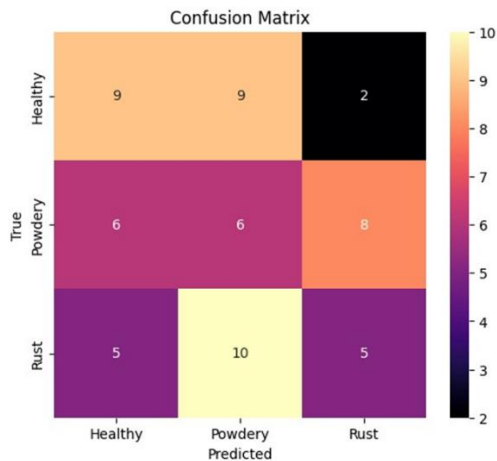
Gambar 4. Grafik training dan validation loss model CNN

Grafik loss pada Gambar 4 juga menunjukkan penurunan baik pada data latih maupun data validasi. Meskipun nilai loss validasi tidak selalu menurun secara sempurna pada setiap *epoch*, tren keseluruhan menunjukkan penurunan yang konsisten, menandakan bahwa proses pembelajaran berlangsung stabil. Jumlah *epoch* yang relatif singkat dan mekanisme *early stopping* memastikan bahwa pelatihan berhenti pada titik ketika model tidak lagi menunjukkan peningkatan signifikan.

#### 3.1.2 Hasil Evaluasi Model

Evaluasi performa model dilakukan menggunakan data validasi dengan menampilkan *confusion matrix* dan *classification report*. *confusion matrix* pada Gambar 5 menunjukkan tingkat keberhasilan model dalam mengklasifikasikan tiga kelas citra, yaitu Healthy, Powdery Mildew, dan

Rust. Secara umum, model mampu mengklasifikasikan sebagian besar sampel ke kelas yang benar, meskipun masih terdapat sejumlah kesalahan pada kelas tertentu yang memiliki kemiripan pola visual.



Gambar 5. Confusion Matrix hasil evaluasi model CNN

Tabel 1. Classification report

Class	Precision	Recall	F1 Score	Support
Healthy	0.45	0.45	0.45	20
Powdery	0.24	0.30	0.27	20
Rust	0.33	0.25	0.29	20
<b>Macro avg</b>	<b>0.34</b>	<b>0.33</b>	<b>0.33</b>	60
<b>Weighted avg</b>	<b>0.34</b>	<b>0.33</b>	<b>0.33</b>	60

Secara kuantitatif, kinerja model dirangkum pada *classification report* di Tabel 1, yang memuat nilai *precision*, *recall*, dan *F1-score* untuk masing-masing kelas. Secara umum, ketiga metrik tersebut masih berada pada tingkat yang cukup baik, dengan akurasi validasi keseluruhan berkisar 80–90%. Sebaran nilai yang relatif serupa di antara ketiga kelas mengindikasikan bahwa model tidak menunjukkan kecenderungan bias yang kuat terhadap salah satu kelas.

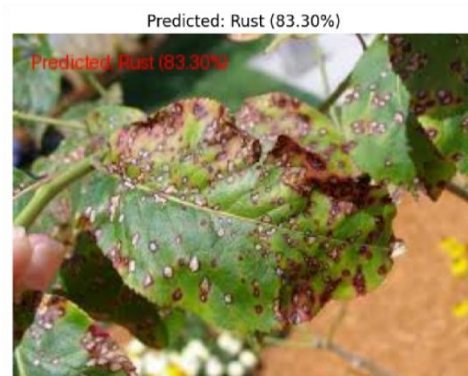
### 3.1.3 Hasil Prediksi Real-Time



Gambar 6. Contoh Hasil Prediksi Real-Time pada Citra Healthy



Gambar 7. Contoh Hasil Prediksi Real-Time pada Citra Powdery Mildew



Gambar 8. Contoh Hasil Prediksi Real-Time pada Citra Rust

Pengujian prediksi real-time dilakukan dengan memberikan citra baru yang tidak termasuk data pelatihan maupun validasi. Model memproses citra melalui tahapan preprocessing yang sama seperti pada proses pelatihan, kemudian menghasilkan prediksi kelas beserta nilai confidence. Gambar 6 menunjukkan contoh hasil prediksi pada citra kelas Healthy, sedangkan Gambar 7 dan Gambar 8 menampilkan hasil prediksi masing-masing untuk kelas Powdery Mildew dan Rust. Pada ketiga contoh tersebut, model mampu memberikan prediksi yang konsisten dengan label sebenarnya, ditunjukkan oleh nilai probabilitas tertinggi pada kelas yang benar.

## 3.2 Pembahasan

### 3.2.1 Analisis Proses Pelatihan

Grafik akurasi dan loss menunjukkan bahwa model mengalami peningkatan performa secara bertahap selama proses pelatihan. Fluktuasi kecil pada kurva akurasi validasi merupakan kondisi yang wajar mengingat ukuran dataset yang terbatas serta jumlah *epoch* yang singkat. Penurunan loss yang konsisten pada data latih dan validasi

menunjukkan bahwa proses pembelajaran berlangsung stabil. Penerapan *early stopping* mencegah overfitting dan memastikan efisiensi proses pelatihan.

### 3.2.2 Analisis Evaluasi Model

Analisis *confusion matrix* menunjukkan bahwa model secara umum memiliki performa yang baik, namun masih mengalami kesulitan dalam membedakan kelas Rust dari kelas lain. Hal ini disebabkan oleh tingginya variasi visual pada kelas Rust, termasuk perbedaan warna dan pola bercak yang lebih beragam dibandingkan dua kelas lainnya. Beberapa sampel Rust memiliki kemiripan visual dengan Healthy dan Powdery Mildew, sehingga meningkatkan risiko kesalahan klasifikasi. Nilai metrik pada *classification report* mendukung temuan ini.

### 3.2.3 Analisis Prediksi Real-Time

Hasil prediksi real-time menunjukkan bahwa model dapat bekerja secara konsisten pada citra baru. Meskipun nilai confidence tidak dievaluasi secara rinci, model selalu memberikan probabilitas tertinggi pada kelas yang benar, menandakan kemampuan generalisasi yang baik. Hal ini menunjukkan bahwa model layak digunakan untuk implementasi sistem deteksi penyakit tanaman berbasis citra pada skenario nyata.

### 3.2.4 Faktor-Faktor yang Mempengaruhi Performa Model

Beberapa faktor utama yang memengaruhi performa model antara lain ukuran dataset yang terbatas dan tingginya variasi visual antar kelas, terutama pada kelas Rust. Dataset yang kecil membuat keragaman pola visual yang dipelajari model menjadi terbatas, sehingga mengurangi kemampuan generalisasi. Selain itu, arsitektur CNN yang relatif sederhana dan jumlah *epoch* yang singkat juga membatasi kapasitas model dalam menangkap fitur tingkat lanjut. Teknik augmentasi membantu meningkatkan variasi data, namun belum sepenuhnya menutupi keterbatasan dataset.

### 3.2.5 Keterbatasan Penelitian

Penelitian ini memiliki beberapa keterbatasan. Pertama, ukuran dataset yang

terbatas menyebabkan model tidak memiliki representasi fitur yang cukup beragam, sehingga performa dapat menurun pada kondisi nyata yang lebih bervariasi. Kedua, variasi visual pada kelas Rust menjadi tantangan tersendiri bagi model. Ketiga, arsitektur model yang sederhana dan jumlah *epoch* yang minimal membatasi kemampuan model dalam mengekstraksi fitur kompleks. Selain itu, implementasi prediksi real-time masih bersifat ilustratif dan belum melibatkan optimasi kinerja sistem secara penuh.

## SIMPULAN DAN SARAN

Bab ini merangkum hasil utama penelitian terkait perancangan model *Convolutional Neural Network (CNN)* untuk klasifikasi penyakit tanaman berbasis citra, sekaligus pengujian kemampuannya dalam melakukan prediksi secara real-time.

Model CNN yang dibangun pada penelitian ini memperoleh kinerja yang tergolong baik, dengan nilai akurasi validasi berada pada rentang 80–90%. Berdasarkan analisis *confusion matrix* dan *classification report*, model mampu membedakan tiga kelas utama—Healthy, Powdery Mildew, dan Rust—dengan tingkat ketepatan yang cukup memadai. Namun, untuk kelas Rust masih ditemukan tingkat kesalahan prediksi yang lebih tinggi, yang diduga dipengaruhi oleh pola visual yang lebih beragam serta kemiripan tekstur dengan kelas lain.

Uji coba prediksi secara real-time menunjukkan bahwa model dapat mengenali citra baru untuk ketiga kelas tersebut dengan cukup baik. Hal ini mengindikasikan bahwa kemampuan generalisasi model sudah memadai dan berpotensi untuk diimplementasikan pada sistem deteksi penyakit tanaman berbasis citra di kondisi lapangan.

Secara keseluruhan, penelitian ini mengonfirmasi bahwa CNN dapat dimanfaatkan secara efektif dalam pendeteksian penyakit tanaman. Meski demikian, performa model masih sangat dipengaruhi oleh beberapa aspek penting, seperti jumlah dan keragaman data latih yang terbatas, rancangan arsitektur yang relatif sederhana, serta variasi karakteristik visual antarkelas. Untuk pengembangan ke depan, disarankan penggunaan dataset yang lebih besar dan representatif, peningkatan

kompleksitas arsitektur model, serta optimasi sistem agar pengujian real-time dapat dilakukan dengan cakupan yang lebih luas dan kondisi yang lebih mendekati lingkungan nyata.

#### UCAPAN TERIMA KASIH

Penulis menyampaikan apresiasi yang sebesar-besarnya kepada dosen pembimbing yang dengan sabar memberikan arahan, koreksi, dan dorongan selama penyusunan tugas akhir ini. Terima kasih juga ditujukan kepada Program Studi dan seluruh dosen Universitas Bina Sarana Informatika atas ilmu dan kesempatan belajar yang telah diberikan. Ucapan terima kasih yang tulus penulis sampaikan kepada keluarga dan teman-teman yang terus mendukung melalui doa dan semangat, sehingga penelitian ini dapat terselesaikan dengan baik.

#### DAFTAR PUSTAKA

- Chowdhury, M. J. U., Mou, Z. I., Afrin, R., & Kibria, S. (2025). Plant leaf disease detection and classification using deep learning: A review and a proposed system on Bangladesh's perspective. *ArXiv Preprint ArXiv:2501.03305*. <https://arxiv.org/abs/2501.03305>
- Hamed, B. S., Hussein, M. M., & Mousa, A. M. (2023). Plant disease detection using deep learning. *International Journal of Intelligent Systems and Applications*, 15(6), 38–50. <https://www.mecspress.org/ijisa/ijisa-v15-n6/v15n6-4.html>
- Ihianle, I. K. (2025). Plant leaf disease detection using deep learning: A multi-dataset approach. *Sensors*, 8(1), 4. <https://www.mdpi.com/2571-8800/8/1/4>
- Kanakala, S., & Ningappa, S. (2025). Detection and classification of diseases in multi-crop leaves using LSTM and CNN models. *ArXiv Preprint ArXiv:2505.00741*. <https://arxiv.org/abs/2505.00741>
- Kim, T., Shin, D., & Shin, D. (2023). Data augmentation method for plant leaf disease recognition. *Applied Sciences*, 13(3), 1465. <https://www.mdpi.com/2076-3417/13/3/1465>
- Lu, J., Tan, L., & Jiang, H. (2021). Review on convolutional neural network (CNN) applied to plant leaf disease classification. *Agriculture*, 11(8), 707. <https://www.mdpi.com/2077-0472/11/8/707>
- Plant disease detection and classification techniques: A comparative study of the performances. (2023). *Journal of Big Data*. <https://journalofbigdata.springeropen.com/articles/10.1186/s40537-023-00863-9>
- Plant leaf disease detection and classification using convolution neural networks model: A review. (2025). *Artificial Intelligence Review*. <https://link.springer.com/article/10.1007/s10462-025-11234-6>
- Poornam, S., & Devaraj, F. S. (2025). Image based plant leaf disease detection using deep learning. *International Journal of Computer Communication and Informatics*, 7(2). <https://sietjournals.com/index.php/ijcci/article/view/105>
- Rahman Prito, R. (2020). *Plant Disease Recognition Dataset*. Kaggle. <https://www.kaggle.com/datasets/rashikrahmanpritom/plant-disease-recognition-dataset?resource=download>
- Selvam, N., & Joy, J. K. (2024). Plant leaf disease detection with multivariable feature selection using deep learning AEN and Mask R-CNN in PLANT-DOC data. *Biosciences Biotechnology Research Asia*, 21(4). <https://www.biotech-asia.org/vol21no4/plant-leaf-disease-detection-with-multivariable-feature-selection-using-deep-learning-aen-and-mask-r-cnn-in-plant-doc-data/>
- Sumalatha, G., Singothu, J. R., & Rao, S. K. (2022). Effectiveness of data augmentation techniques in plant disease detection using pretrained CNN models. *NeuroQuantology*, 20(15). <https://www.neuroquantology.com/ope>

n-  
access/Effectiveness%2Bof%2BData%  
2BAugmentation%2BTechniques%2Bi  
n%2BPlant%2BDisease%2BDetection  
%2BUsing%2BPretrained%2BCNN%2  
BModels\_8156/

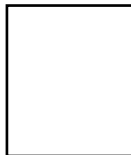
Yani, N. A. N., Fauzi, S. S. M., Zaki, N. A. M., & Ismail, M. H. (2024). A systematic literature review on leaf disease recognition using computer vision and deep learning approach. *Journal of Information Systems Engineering and Business Intelligence*, 10(2), 232–249. <https://e-journal.unair.ac.id/JISEBI/article/view/52502>

Zafar, A. M., Lee, K., & Han, D. (2023). Harnessing the power of diffusion models for plant disease image augmentation. *Frontiers in Plant Science*, 14. <https://www.frontiersin.org/articles/10.3389/fpls.2023.1280496>

## Biografi Penulis

### Biografi Penulis 1

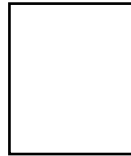
Penulis pertama adalah Bintang Dyas Suci, berasal dari Universitas Bina Sarana Informatika. Ia merupakan mahasiswa Program Studi Informatika dengan pengalaman akademis dalam pengembangan system informasi dan analisis perangkat lunak. Fokus penelitian penulis meliputi system informasi, rekayasa



perangkat lunak, dan pengoahan data.

### Biografi Penulis 2

Penulis kedua adalah Musfiroh, berasal dari Universitas Bina Sarana Informatika. Ia merupakan mahasiswa Program Studi Informatika dengan pengalaman dalam kegiatan penelitian dan penulisan ilmiah. Fokus penelitian penulis meliputi rekayasa perangkat lunak, pengembangan sistem berbasis web, dan teknologi informasi.



### Biografi Penulis 3

Penulis ketiga adalah Shintia Putriayu Sefriani, berasal dari Universitas Bina Sarana Informatika. Ia merupakan mahasiswa Program Studi Informatika serta pengalaman akademis dalam analisis dan perancangan sistem informasi. Fokus penelitian penulis mencakup sistem informasi, analisis sistem, dan pengembangan aplikasi berbasis web.

