

**KLASIFIKASI PENYAKIT DAUN PADI MENGGUNAKAN ALGORITMA
CONVOLUTIONAL NEURAL NETWORK(CNN)**



disusun oleh:
ADAM RIZQI PRAMUDYA 22.11.4702

Kepada
FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA

2025

1. Dokumentasi Proyek

1.1 Pendahuluan

- **Latar Belakang:** Produktivitas padi sebagai tanaman pangan pokok terancam oleh berbagai penyakit daun yang sulit dideteksi dini secara tradisional, menyebabkan kerugian hasil panen. Perkembangan teknologi AI, khususnya pengolahan citra digital, memungkinkan pengembangan sistem deteksi penyakit tanaman otomatis.
- **Tujuan Proyek:** Mengembangkan sistem klasifikasi otomatis untuk penyakit daun padi menggunakan metode *deep learning* berbasis Convolutional Neural Network (CNN) pada citra daun. Sistem ini diharapkan dapat membantu petani mendeteksi penyakit dini, meningkatkan hasil produksi, dan mengurangi kerugian.
- **Ruang Lingkup:**
 - Pengembangan sistem prediksi penyakit daun padi berbasis metode CNN.
 - Penilaian performa model CNN dalam mendeteksi dan mengklasifikasikan penyakit daun padi.
 - Kontribusi terhadap implementasi teknologi kecerdasan buatan di bidang pertanian.
 - Dataset yang digunakan terbatas pada “Rice Leaf’s Disease Dataset” dari Kaggle.
 - Jenis penyakit daun padi yang dianalisis hanya berdasarkan kategori penyakit yang tersedia pada dataset tersebut (bacterial_leaf_blight, brown_spot, healthy, leaf_blast, leaf_scald, dan narrow_brown_spot).
 - Proses klasifikasi dan deteksi dilakukan menggunakan model CNN tanpa membandingkan secara langsung dengan model lain.
 - Fokus penelitian adalah pada prediksi berbasis citra, bukan pada aspek biologis atau fisiologis tanaman.

1.2 Kebutuhan Fungsional:

- Sistem harus mampu mengklasifikasikan citra daun padi ke dalam enam kategori kondisi utama: bacterial leaf blight, brown spot, healthy, leaf blast, leaf scald, dan narrow brown spot.
- Sistem harus dapat menerima input berupa citra daun padi.
- Sistem harus dapat menampilkan hasil klasifikasi penyakit.

1.3 Kebutuhan Non-Fungsional:

- **Kinerja:** Model diharapkan mencapai akurasi tinggi dalam mengklasifikasikan citra daun padi. Model menunjukkan akurasi evaluasi sebesar 0.7902 dan *loss* sebesar 0.5330 pada data uji. Waktu pemrosesan model efisien, yaitu 287 ms per langkah.

- **Kemudahan Penggunaan:** Hasil dapat dimanfaatkan oleh petani, personel pertanian, peneliti, dan pengembang teknologi pertanian. Sistem ini dapat menjadi alat bantu dalam mengenali penyakit tanaman secara cepat dan tepat bagi petani.
- **Skalabilitas & Pemeliharaan:** Direkomendasikan penelitian lebih lanjut dengan dataset yang lebih besar dan penerapan model canggih seperti ResNet atau AI yang dapat dijelaskan (*explainable AI*) untuk meningkatkan akurasi dan transparansi sistem.

2. Dokumen Desain Sistem (System Design Document)

2.1 Arsitektur Sistem: Sistem ini menggunakan arsitektur *deep learning* berbasis Convolutional Neural Network (CNN).

- **Modul Utama:**
 - *Import Library.*
 - *Load Dataset.*
 - *Exploratory Data Analysis (EDA)* dan Visualisasi Data.
 - *Preprocessing* (resize, normalisasi, augmentasi).
 - *Split Data.*
 - *Modeling* (Pelatihan model CNN).
 - Evaluasi Performa Model.
 - *Testing Model.*

2.2 Desain Model CNN:

- **Arsitektur Spesifik:** Model CNN menggunakan 4 lapis konvolusi.
- **Pola Lapisan:** Conv2D → ReLU → MaxPooling.
- **Lapisan Akhir:** Flatten → Dense (512 unit) → Dropout (0.5) → *Output layer* (softmax 6 kelas).
- **Fungsi Aktivasi:** ReLU (implisit dari pola Conv2D → ReLU). Meskipun fungsi sigmoid disebutkan dalam teori, praktiknya tidak banyak digunakan karena kelemahannya.
- **Parameter Model:** Jumlah lapisan dan neuron pada masing-masing lapisan dianggap sebagai *hyperparameter* dan dioptimasi menggunakan pendekatan *searching*.
- **Proses Pelatihan:** Model dilatih selama 30 *epoch* dengan data validasi dari *validation_generator*.

2.3 Desain Pra-pemrosesan Data:

- **Resizing:** Mengubah ukuran gambar menjadi dimensi seragam (misalnya 255x255 piksel) agar sesuai dengan *input* model CNN.
- **Normalisasi:** Nilai piksel dinormalisasi ke skala 0-1 dengan membagi 255.

- **Augmentasi Data:** Dilakukan augmentasi data seperti rotasi, geser, *zoom*, dan *horizontal flip* untuk memperbesar dan memperkaya dataset secara buatan, mengurangi *overfitting*, dan meningkatkan akurasi model.
- **Pembagian Data:** Dataset sudah dibagi ke dalam *train*, *test*, dan *valid* dalam struktur direktori.

ImageDataGenerator akan memuat data sesuai pembagian tersebut. Label kelas diubah menjadi *one-hot encoding* dengan *to_categorical()*.

2.4 Teknologi yang Digunakan:

- **Bahasa Pemrograman:** Python.
- **Framework/Library:** TensorFlow / Keras , scikit-learn , Matplotlib , Seaborn , Numpy , OS (untuk manipulasi direktori dan *path* file).
- **Lingkungan Pengembangan:** Google Colab (platform komputasi awan).
- **Hardware:** PC untuk menjalankan proses komputasi, pelatihan model, dan analisis data. Penggunaan GPU pada Google Colab dimanfaatkan untuk mempercepat proses pelatihan.

3. Dokumen Teknis/Implementasi (Technical/Implementation Document)

3.1 Struktur Direktori Proyek: Dataset diatur dalam direktori

train, *test*, dan *valid*, dengan subdirektori untuk setiap kelas penyakit (*bacterial_leaf_blight*, *brown_spot*, *healthy*, *leaf_blast*, *leaf_sclad*, dan *narrow_brown_spot*).

3.2 Detail Implementasi Setiap Modul:

- **Import Library:** Menyambungkan Google Colab ke Google Drive sebagai lokasi penyimpanan dataset. Mengimpor TensorFlow, Keras, scikit-learn, Matplotlib, dan Seaborn.
- **Load Dataset:** Dataset dimuat dari direktori *train*, *test*, dan *valid* menggunakan *ImageDataGenerator.flow_from_directory()* yang secara otomatis mengidentifikasi kelas dari nama subdirektori.
- **Exploratory Data Analysis (EDA) dan Visualisasi Data:** Mengeksplorasi dan menganalisis dataset, menampilkan contoh gambar dari setiap kelas, dan membandingkan proporsi jumlah gambar per kelas melalui *bar chart* dan/atau *pie chart*.
- **Preprocessing:** Meliputi *resize* gambar ke 255x255 piksel, normalisasi nilai piksel ke skala 0-1, dan augmentasi data (rotasi, geser, *zoom*, *horizontal flip*).
- **Split Data:** Dataset sudah terbagi secara default. Label kelas diubah menjadi *one-hot encoding*.
- **Modeling:** Arsitektur CNN dengan 4 lapis konvolusi (Conv2D → ReLU → MaxPooling), diakhiri dengan Flatten → Dense (512 unit) → Dropout (0.5) → *Output layer* (softmax 6 kelas). Pelatihan model selama 30

epoch.

- **Evaluasi:** Menampilkan akurasi pada *test set*, visualisasi akurasi dan *loss*, *confusion matrix*, dan *classification report*.
- **Testing Model:** Mengambil data gambar acak dari folder *test* dan memprediksinya dengan model.

3.3 Dataset:

- **Nama Dataset:** “Rice Leaf’s Disease Dataset”.
- **Sumber:** Kaggle (<https://www.kaggle.com/datasets/nirmalsankalana/rice-leaf-disease-image>).
- **Ukuran:** Total 1476 gambar. Terbagi rata menjadi 40 gambar untuk setiap kategori di set pengujian (*test*). Jumlah total citra lebih dari seribu gambar yang dibagi ke dalam enam kelas utama.
- **Kategori Penyakit:** bacterial_leaf_blight, brown_spot, healthy, leaf_blast, leaf_scald, dan narrow_brown_spot.
- **Format Data:** .jpg (atau .png, .jpeg).
- **Pelabelan:** Citra telah dilabeli secara manual oleh ahli pertanian profesional.

4. Dokumen Pengujian (Testing Document)

4.1 Metrik Evaluasi:

- Akurasi.
- *Confusion matrix*.
- *Classification report* (precision, recall, f1-score).
- Visualisasi akurasi dan *loss* pelatihan dan validasi.

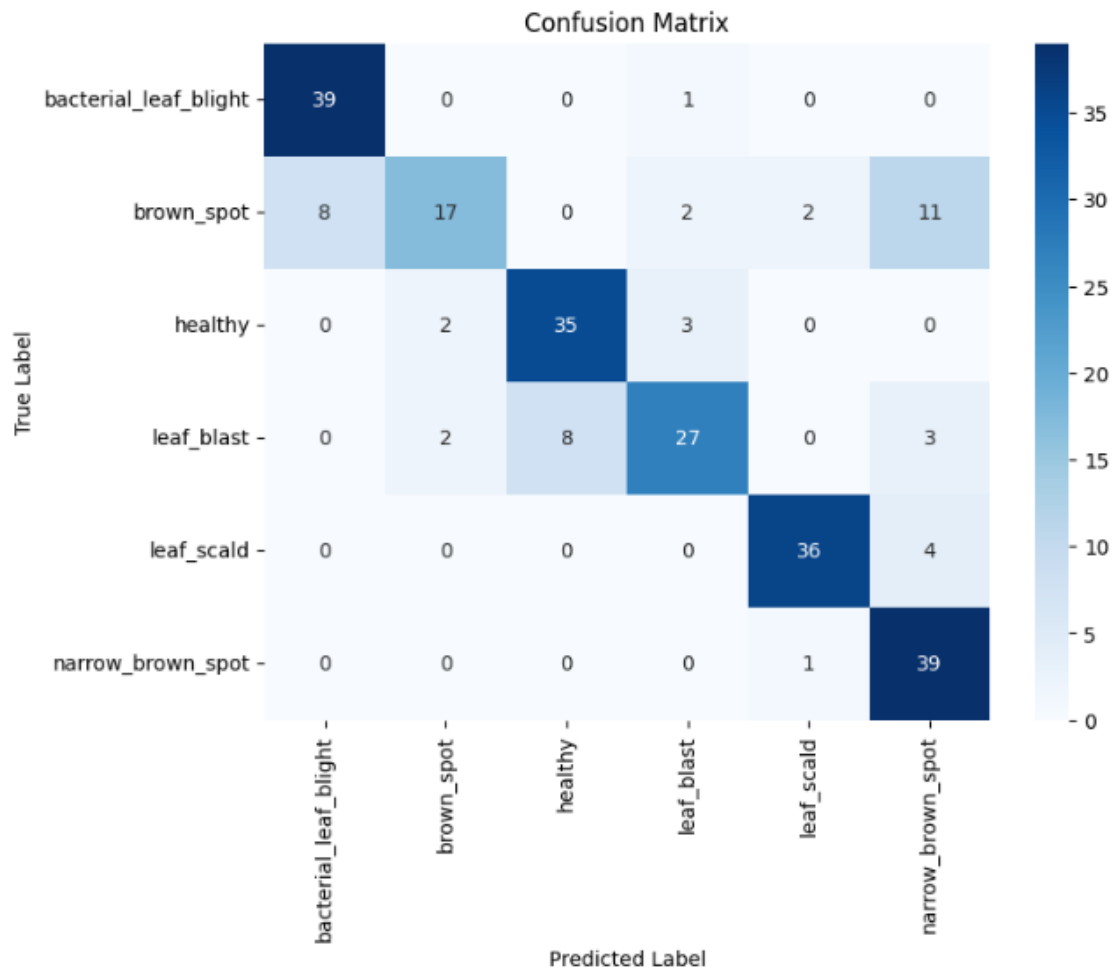
4.2 Hasil Pengujian:

- **Akurasi pada data uji:** 0.7902 (79.02%).
- **Loss pada data uji:** 0.5330.
- **Waktu Pemrosesan:** 287 ms per langkah.
- **Classification Report (Gambar 4.2):**

Classification Report:

	precision	recall	f1-score	support
bacterial_leaf_blight	0.83	0.97	0.90	40
brown_spot	0.81	0.42	0.56	40
healthy	0.81	0.88	0.84	40
leaf_blast	0.82	0.68	0.74	40
leaf_scald	0.92	0.90	0.91	40
narrow_brown_spot	0.68	0.97	0.80	40
accuracy			0.80	240
macro avg	0.81	0.80	0.79	240
weighted avg	0.81	0.80	0.79	240

- **bacterial_leaf_blight:** Precision: 0.83, Recall: 0.97, F1-score: 0.90.
- **brown_spot:** Precision: 0.81, Recall: 0.42, F1-score: 0.56.
- **healthy:** Precision: 0.81, Recall: 0.88, F1-score: 0.84.
- **leaf_blast:** Precision: 0.82, Recall: 0.68, F1-score: 0.74.
- **leaf_scald:** Precision: 0.92, Recall: 0.90, F1-score: 0.91.
- **narrow_brown_spot:** Precision: 0.68, Recall: 0.97, F1-score: 0.80.
- **Akurasi Keseluruhan:** 0.80 (80%).
- **Macro average:** Precision: 0.81, Recall: 0.80, F1-score: 0.79.
- **Weighted average:** Precision: 0.81, Recall: 0.80, F1-score: 0.79.



- **Confusion Matrix (Gambar 4.3):** Menunjukkan jumlah *True Positive*, *False Positive*, *False Negative*, dan *True Negative* untuk setiap kelas.

5. Dokumen Pemeliharaan (Maintenance Document)

5.1 Log Perubahan (Change Log): Belum ada log perubahan spesifik yang disediakan dalam dokumen ini, namun skripsi mencakup sistematika penulisan standar untuk laporan penelitian.

5.2 Saran untuk Peningkatan:

- Validasi lebih lanjut menggunakan dataset eksternal yang lebih kompleks dan beragam secara klinis.
- Pengembangan model dengan arsitektur CNN yang lebih dalam atau kombinasi dengan metode *transfer learning* seperti ResNet atau EfficientNet.
- Integrasi teknik *image segmentation* pada tahapan *preprocessing* untuk meningkatkan akurasi dan pemahaman klinis, serta potensi untuk mengukur tingkat keparahan penyakit.
- Mengoptimalkan *hyperparameter* lebih lanjut.
- Meningkatkan jumlah dan kualitas dataset untuk kelas-kelas yang performanya masih rendah, seperti *brown_spot* dan *leaf_blast*.