E-ISSN: 3031-8521
P-ISSN: 3032-1867
Halaman 1492-1509
Volume 3 Nomor 4 Tahun 2026

Sistem Klasifikasi Diagnosa Penyakit Daun Cabai Berbasis Pengolahan Citra Menggunakan Convolutional Neural Network (CNN)

Sientia Anggi Fadillah Siregar

Program Studi Sistem Informasi Fakultas Ilmu Komputer Dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara Medan
e-mail : anggifadilla2004@gmail.com

Abstrak

Identifikasi penyakit daun cabai secara manual menghadapi kendala berupa kemiripan gejala visual dan keterbatasan akses terhadap tenaga ahli pertanian. Penelitian ini bertujuan mengembangkan sistem klasifikasi diagnosis kondisi daun cabai berbasis *Convolutional Neural Network* (CNN) menggunakan arsitektur MobileNetV2 dan pendekatan *transfer learning*. Penelitian menggunakan pendekatan kuantitatif eksperimental dengan dataset sekunder dari Kaggle yang terdiri atas 1.200 citra daun cabai, meliputi kelas daun sehat, daun keriting, dan daun menguning. Instrumen penelitian berupa dataset citra digital dan perangkat lunak pengembangan model. Dataset dibagi menjadi 80% data latih, 10% data validasi, dan 10% data uji. Evaluasi dilakukan menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score*, sedangkan pengujian fungsional menggunakan metode *black-box testing*. Hasil pengujian menunjukkan akurasi model sebesar 100%, dengan nilai *precision*, *recall*, dan *F1-score* sebesar 1,0000 pada seluruh kelas. Seluruh skenario pengujian fungsional juga berjalan sesuai harapan. Kesimpulannya, sistem SIKDAC berpotensi menjadi alat bantu identifikasi awal kondisi daun cabai melalui aplikasi web dengan inferensi langsung menggunakan TensorFlow.js. Namun, pengujian pada citra lapangan yang lebih beragam masih diperlukan untuk memastikan kemampuan generalisasi sistem.

Kata Kunci: *Penyakit Daun Cabai, Convolutional Neural Network, Deep Learning, Klasifikasi Citra, Transfer Learning*

Abstract

Manual identification of chili leaf diseases is challenged by visual symptom similarities and limited access to agricultural experts. This study aimed to develop a chili leaf condition classification system using a Convolutional Neural Network (CNN), MobileNetV2 architecture, and transfer learning. An experimental quantitative approach was employed using a secondary dataset obtained from Kaggle, comprising 1,200 chili leaf images categorized as healthy, curled, and yellowing leaves. The research instruments consisted of digital image data and software for model development. The

dataset was divided into 80% training, 10% validation, and 10% testing data. Model performance was evaluated using accuracy, precision, recall, and F1-score, while functional testing employed black-box testing. The results demonstrated 100% test accuracy, with precision, recall, and F1-score values of 1.0000 for all three classes. All functional test scenarios also met the expected outcomes. These findings indicate that the SIKDAC system can support the initial identification of chili leaf conditions through a web application using browser-based inference with TensorFlow.js. However, further testing with more diverse field images is necessary to establish the system's generalization capability.

Keywords: *Chili Leaf Disease, Convolutional Neural Network, Deep Learning, Image Classification, Transfer Learning*

LATAR BELAKANG

Cabai merupakan salah satu komoditas hortikultura bernilai ekonomi tinggi yang memiliki permintaan untuk memenuhi kebutuhan dalam negeri maupun ekspor. Namun, produktivitas tanaman cabai sering mengalami gangguan akibat serangan penyakit, termasuk penyakit pada daun yang dapat memengaruhi hasil panen dan nilai jualnya. Kondisi tersebut menjadikan identifikasi penyakit daun sebagai salah satu aspek penting dalam menjaga produktivitas tanaman cabai (Mulyana Prayoga, Wahyuni, and Elisawati 2025).

Perkembangan teknologi informasi membuka peluang pemanfaatan sistem klasifikasi berbasis citra digital untuk membantu mengidentifikasi penyakit tanaman. Penggunaan *Deep Learning*, khususnya *Convolutional Neural Network* (CNN), memungkinkan sistem mengenali pola visual pada daun secara otomatis sehingga dapat mendukung proses diagnosis penyakit tanaman (Tri et al. 2024). Pendekatan ini juga berpotensi meningkatkan ketepatan identifikasi penyakit berdasarkan karakteristik citra daun (Zhang et al., 2021).

Permasalahan identifikasi penyakit daun cabai masih berkaitan dengan penggunaan pengamatan visual secara manual yang bergantung pada pengalaman dan pengetahuan petani. Perbedaan tingkat pemahaman mengenai gejala penyakit dapat menyebabkan kesalahan diagnosis dan pemilihan penanganan yang kurang sesuai. Selain itu, serangan penyakit pada daun dapat menghambat proses fotosintesis, mengganggu pertumbuhan tanaman, dan menurunkan hasil panen.

Kesulitan diagnosis juga disebabkan oleh kemiripan gejala visual pada beberapa kondisi daun. Perubahan warna, munculnya bercak, daun mengering, dan bentuk daun yang mengeriting dapat menyulitkan proses identifikasi jika hanya mengandalkan pengamatan langsung. Keterbatasan jumlah tenaga ahli pertanian dan akses terhadap penyuluh turut menjadi kendala dalam memperoleh diagnosis secara cepat dan tepat, sehingga keterlambatan identifikasi dapat meningkatkan risiko penyebaran penyakit dan kerugian hasil panen.

Penelitian sebelumnya menunjukkan bahwa CNN berpotensi digunakan untuk mengidentifikasi penyakit tanaman berdasarkan citra digital. Penelitian Tsany dan Dzaky

(2021) menggunakan arsitektur AlexNet untuk mengklasifikasikan kondisi daun cabai dan memperoleh akurasi sebesar 80% pada 10 epoch serta 90% pada 40 epoch. Sementara itu, Pratama et al. (2023) menerapkan CNN untuk mendeteksi penyakit daun hortikultura dengan akurasi 94,2%. Hasil tersebut menjadi dasar untuk mengembangkan sistem klasifikasi yang dapat membantu identifikasi kondisi daun cabai secara otomatis dan diimplementasikan melalui platform web.

Penelitian ini bertujuan mengembangkan sistem klasifikasi citra daun cabai menggunakan CNN untuk membedakan tiga kondisi, yaitu daun sehat, daun menguning, dan daun keriting. Penelitian ini penting sebagai upaya membantu proses identifikasi penyakit secara lebih cepat dan terukur, terutama ketika akses terhadap tenaga ahli pertanian terbatas. Kebaruan penelitian terletak pada penerapan *transfer learning* dengan arsitektur MobileNetV2 yang diimplementasikan dalam sistem web SIKDAC, dengan proses klasifikasi dijalankan langsung melalui peramban menggunakan TensorFlow.js.

METODE

Penelitian ini menggunakan pendekatan kuantitatif eksperimental untuk mengklasifikasikan citra daun cabai menggunakan metode *Convolutional Neural Network* (CNN). CNN dipilih karena mampu mempelajari fitur visual pada citra untuk mendukung proses klasifikasi penyakit tanaman (Alzubaidi et al., 2021). Model dikembangkan menggunakan pendekatan *transfer learning* dengan arsitektur MobileNetV2 untuk membedakan tiga kondisi daun cabai, yaitu sehat, menguning, dan keriting. Penggunaan *transfer learning* relevan untuk identifikasi penyakit tanaman berbasis citra karena memungkinkan pemanfaatan model yang telah dilatih sebelumnya (Chen et al., 2022).

Instrumen penelitian berupa dataset citra digital daun cabai yang diperoleh dari Kaggle serta perangkat lunak untuk pengembangan, pelatihan, dan pengujian model. Dataset tersebut merujuk pada data *Penyakit Daun Cabai* yang dipublikasikan oleh Ratnasari (2023). Kinerja model dianalisis menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score* untuk menilai ketepatan hasil klasifikasi. Selanjutnya, pengujian fungsional sistem dilakukan menggunakan metode *black-box testing* untuk memeriksa kesesuaian fungsi sistem dengan hasil yang diharapkan.

Data penelitian berupa citra digital daun cabai yang diperoleh dari repositori Kaggle. Sampel penelitian berjumlah 1.200 citra yang mencakup tiga kondisi daun, yaitu sehat, menguning, dan keriting. Dataset dibagi menjadi 80% data latih, 10% data validasi, dan 10% data uji. Pembagian ini digunakan untuk melatih model, memantau proses pelatihan, serta mengevaluasi kinerja klasifikasi pada data uji.

Penelitian diawali dengan pengumpulan dan prapemrosesan citra daun cabai, kemudian dilanjutkan dengan pembagian dataset menjadi data latih, validasi, dan uji. Model MobileNetV2 dikembangkan melalui dua tahap, yaitu *feature extraction* dengan membekukan lapisan dasar model dan *fine-tuning* dengan membuka 30 lapisan terakhir. Pendekatan ini menerapkan prinsip *transfer learning* dalam pemanfaatan model untuk klasifikasi citra penyakit tanaman (Chen et al., 2022). Setelah pelatihan, kinerja model

dievaluasi menggunakan *accuracy*, *precision*, *recall*, dan *F1-score*. Model yang telah dilatih kemudian diimplementasikan ke dalam sistem web SIKDAC menggunakan TensorFlow.js agar klasifikasi dapat dilakukan langsung melalui peramban. Tahap akhir adalah pengujian fungsional menggunakan *black-box testing* untuk memastikan fungsi sistem berjalan sesuai dengan kebutuhan yang telah ditentukan.

HASIL DAN PEMBAHASAN

Hasil Pelatihan Model Convolutional Neural Network

Persiapan dan Pembagian Dataset

Dataset yang digunakan dalam penelitian ini terdiri atas 1.200 citra daun cabai yang diperoleh dari repositori Kaggle, kemudian diekstraksi dan disimpan pada Google Drive untuk diakses melalui Google Colaboratory. Dataset asli tersimpan dalam tiga folder kategori, yaitu Daunsehat, Leafcurl (daun keriting akibat virus Gemini), dan yelowwiss (daun menguning), yang masing-masing berisi 400 citra.

Sebelum masuk ke tahap pelatihan, dataset terlebih dahulu dibagi menjadi tiga bagian menggunakan teknik pembagian acak (random split) dengan seed tetap (seed=42) agar hasil pembagian dapat direproduksi. Rasio pembagian yang digunakan adalah 80% data latih (train), 10% data validasi (validation), dan 10% data uji (test). Penggunaan data validasi terpisah dari data uji bertujuan untuk memonitor performa model selama proses pelatihan berlangsung tanpa mencampurnya dengan data uji akhir yang sama sekali belum pernah dilihat oleh model, sehingga evaluasi akhir dapat mencerminkan kemampuan generalisasi model secara lebih objektif.

Tabel 1. Hasil Pembagian Dataset per Kelas

Kelas	Data Latih (80%)	Data Validasi (10%)	Data Uji (10%)	Total
Daun Sehat (Daunsehat)	320	40	40	400
Daun Keriting (Leafcurl)	320	40	40	400
Daun Menguning (yelowwiss)	320	40	40	400
Total	**960**	**120**	**120**	**1.200**

Perlu dicatat bahwa rasio pembagian data yang diterapkan pada tahap implementasi (80:10:10) sedikit berbeda dari rancangan awal pada proposal penelitian yang menetapkan skema *hold-out validation* 80:20 (data latih dan data uji tanpa subset validasi terpisah). Penyesuaian ini dilakukan agar proses pemantauan performa model selama pelatihan (*monitoring val_accuracy* dan *val_loss*) dapat dilakukan pada data yang berbeda dari data uji akhir, sehingga hasil pengujian akhir benar-benar mencerminkan kemampuan model terhadap data yang belum pernah dilihat sama sekali.

Augmentasi Data

Tabel 2. Parameter Augmentasi Data Latih

Parameter	Nilai	Keterangan
rotation_range	30°	Rotasi gambar secara acak
width_shift_range	0.2	Pergeseran horizontal
height_shift_range	0.2	Pergeseran vertikal
shear_range	0.15	Distorsi kemiringan (shear)
zoom_range	0.2	Perbesaran/perkecilan acak
horizontal_flip	True	Pembalikan horizontal
vertical_flip	True	Pembalikan vertikal
brightness_range	0.7 – 1.3	Variasi tingkat kecerahan
rescale	1/255	Normalisasi nilai piksel ke rentang 0–1

Dengan konfigurasi tersebut, sistem berhasil mendeteksi 960 gambar data latih, 120 gambar data validasi, dan 120 gambar data uji, yang masing-masing terbagi rata ke dalam tiga kelas.

Arsitektur Model: MobileNetV2 Transfer Learning

Implementasi menggunakan pendekatan transfer learning dengan arsitektur MobileNetV2 pretrained pada ImageNet sebagai feature extractor. Pendekatan ini dipilih karena lebih efisien dalam waktu pelatihan dan sesuai untuk dataset terbatas sebanyak 960 citra. Lapisan klasifikasi akhir diganti dengan classifier head untuk mengklasifikasikan tiga kelas, yaitu Sehat, Keriting, dan Menguning, yang terdiri atas GlobalAveragePooling2D, BatchNormalization, Dense 256 neuron (ReLU), Dropout 0,4, Dense 128 neuron (ReLU), Dropout 0,3, dan Dense 3 neuron (Softmax).

Tabel 3. Parameter Model

Komponen	Jumlah Parameter
Total Parameter	2.624.323 (± 10,01 MB)
Parameter Dapat Dilatih (Trainable)	363.779 (± 1,39 MB)
Parameter Tidak Dilatih (Non-trainable)	2.260.544 8,62 MB)

Ukuran input citra yang digunakan adalah 224 × 224 piksel dengan 3 kanal warna (RGB), konsisten dengan spesifikasi *input default* arsitektur MobileNetV2 serta sesuai dengan tahap prapemrosesan citra yang telah dirancang pada Bab III.

Proses Pelatihan Dua Tahap (Feature Extraction dan Fine-Tuning)

Tahap 1 *Feature Extraction* (*Base Model* Dibekukan)

Pada tahap pertama, seluruh lapisan MobileNetV2 dibekukan (trainable = False) sehingga bobot yang telah dilatih pada ImageNet tidak berubah. Hanya

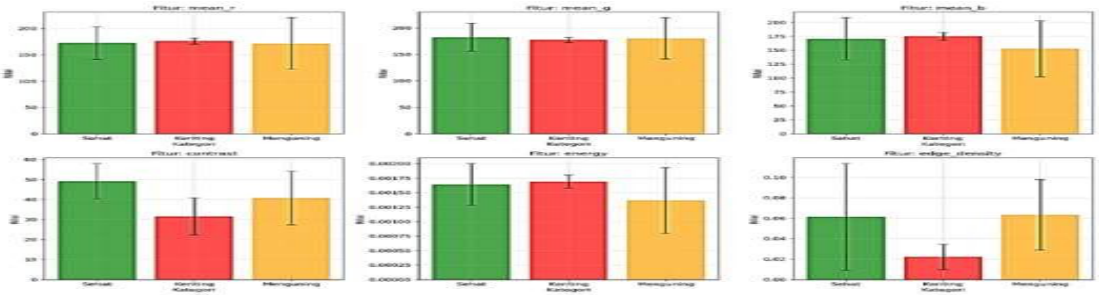
classifier head yang ditambahkan yang dilatih pada tahap ini. Model dikompilasi menggunakan optimizer Adam dengan learning rate 0.001 dan fungsi loss categorical_crossentropy. Pelatihan dijalankan hingga maksimal 25 epoch dengan mekanisme EarlyStopping (menghentikan pelatihan otomatis apabila val_loss tidak membaik selama 8 epoch berturut-turut) dan ReduceLROnPlateau (menurunkan learning rate otomatis jika val_loss stagnan). Pada epoch pertama, model telah mencapai val_accuracy sebesar 95,00%, yang kemudian terus meningkat pada epoch-epoch berikutnya.

Tahap 2 Fine-Tuning (Membuka Sebagian Lapisan MobileNetV2)

Setelah classifier head stabil pada tahap pertama, dilakukan fine-tuning dengan membuka kembali (unfreeze) 30 lapisan terakhir dari MobileNetV2, sementara lapisan-lapisan awal tetap dibekukan. Tahap ini bertujuan agar model dapat menyesuaikan fitur-fitur level tinggi secaralebih spesifik terhadap karakteristik visual daun cabai, seperti pola bercak, tekstur keriting, dan gradasi warna menguning. Learning rate diturunkan secara signifikan menjadi 0,00001 (1e-5) agar proses penyesuaian bobot pretrained berlangsung secara halus dan tidak merusak fitur yang telah dipelajari sebelumnya. Pelatihan tahap kedua dijalankan hingga maksimal 30 epoch dengan callback yang serupa. Pada epoch pertama tahap fine-tuning, model langsung mencapai val_accuracy sebesar 100% dengan val_loss sebesar 0,0058.

Tabel 4. Konfigurasi Pelatihan Dua Tahap

Parameter	Tahap 1 (*Feature Extraction*)	Tahap 2 (*Fine-Tuning*)
Status Base Model	Dibekukan (*frozen*)	30 layer terakhir dibuka
Learning Rate	0,001	0,00001 (1e-5)
Maksimal *Epoch*	25	30
Optimizer	Adam	Adam
Loss Function	*Categorical Crossentropy*	*Categorical Crossentropy*
Early Stopping (patience)	*8 epoch*	*10 epoch*



Gambar 1. Grafik Riwayat Pelatihan (Akurasi dan Loss)

Grafik pada Gambar 1 memperlihatkan tren peningkatan akurasi dan penurunan loss secara konsisten sepanjang proses pelatihan, baik pada data latih maupun data validasi. Garis putus-putus pada grafik menandai titik dimulainya tahap fine-tuning, di mana terlihat peningkatan performa yang cukup signifikan segera setelah lapisan-lapisan MobileNetV2 dibuka sebagian.

Evaluasi Model pada Data Uji

Tabel 5. Hasil Evaluasi Model pada Data Uji

Metrik	Nilai
Akurasi (*Accuracy*)	100,00%
Loss	0,0003
Jumlah Data Uji	120 citra

Untuk memperoleh gambaran performa yang lebih rinci per kelas, dilakukan pula perhitungan *classification report* yang mencakup *precision, recall*, dan f1-*score*.

Tabel 6. *Classification Report* Model pada Data Uji

Kelas	*Precision*	*Recall*	*F1-Score*	*Support*
Daun Sehat	1,0000	1,0000	1,0000	40
Daun Keriting	1,0000	1,0000	1,0000	40
Daun Menguning	1,0000	1,0000	1,0000	40
Rata-rata (*macro* avg)	1,0000	1,0000	1,0000	120
Rata-rata (*weighted* avg)	1,0000	1,0000	1,0000	120

Berdasarkan tabel di atas, model berhasil mengklasifikasikan seluruh 120 data uji dengan tepat tanpa ada satu pun kesalahan klasifikasi (*misclassification*) antar kelas, sehingga menghasilkan nilai *precision, recall*, dan *f1-score* sempurna (1,0000) pada ketiga kelas. Meskipun hasil ini menunjukkan performa yang sangat baik, capaian akurasi sempurna (100%) perlu ditinjau secara kritis dan tidak serta-merta diartikan bahwa model telah mencapai kemampuan generalisasi yang sempurna pada kondisi dunia nyata. Pembahasan lebih lanjut mengenai hal ini disampaikan pada subbab 4.5 (Kelemahan Sistem).

Konversi Model ke Format TensorFlow.js

Setelah pelatihan dan evaluasi, model berformat .keras dan .h5 dikonversi ke TensorFlow.js menggunakan pustaka tensorflowjs agar prediksi dapat dijalankan langsung di browser tanpa backend Python/Flask. Pendekatan client-side inference mengurangi biaya hosting, beban server, dan latensi jaringan sehingga respons prediksi lebih cepat. Hasil konversi disimpan dalam direktori assets/model/, yang terdiri atas model.json sebagai struktur model dan metadata bobot, tiga berkas bobot (shard), serta class_labels.json sebagai pemetaan

indeks keluaran ke nama kelas.

Tabel 7. Pemetaan Indeks Kelas *Output* Model

Indeks Output Model	Nama Folder Dataset	Kode Kelas pada Sistem	Label yang Ditampilkan
0	Daunsehat	Sehat	Daun Sehat
1	Leafcurl	Keriting	Daun Keriting (Virus Gemini)
2	yelowwiss	Menguning	Daun Menguning

Pemetaan pada Tabel 7 bersifat krusial karena urutan indeks output softmax pada model mengikuti urutan alfabetis nama folder dataset yang digunakan pada tahap pelatihan. Kesalahan pada pemetaan ini akan menyebabkan hasil klasifikasi yang ditampilkan pada antarmuka web menjadi tertukar antar kelas, sehingga proses ini divalidasi secara cermat agar konsisten antara keluaran model, berkas class_labels.json, dan kode program pada sisi klien yang berada pada direktori (assets/js/diagnosa.js).

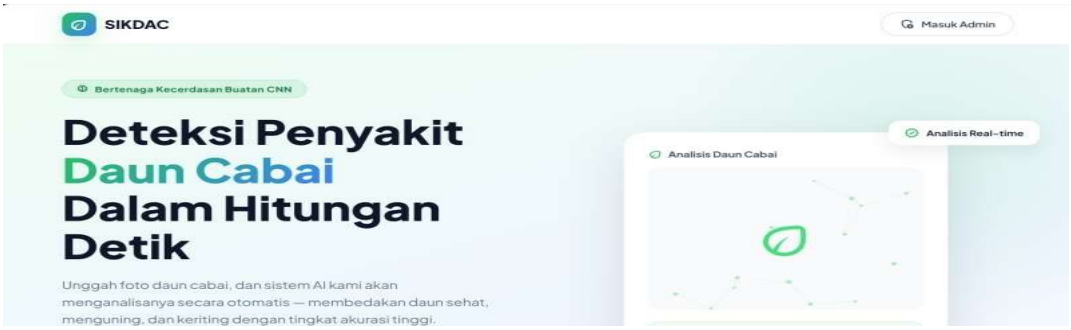
Implementasi Sistem Web SIKDAC

Tabel 8. Spesifikasi Teknis Implementasi Sistem

Komponen	Teknologi yang Digunakan
Bahasa Pemrograman (Server)	PHP (*native*, tanpa *framework*)
Basis Data	MySQL (MySQLi *Prepared Statement*)
Mesin Inferensi CNN	TensorFlow.js (*client-side*, berjalan di *browser*)
Antarmuka Pengguna	HTML5, CSS3, JavaScript (vanilla)
Pembuatan Dokumen PDF	*Library* FPDF
Server Pengembangan	Laragon (Apache/MySQL lokal)

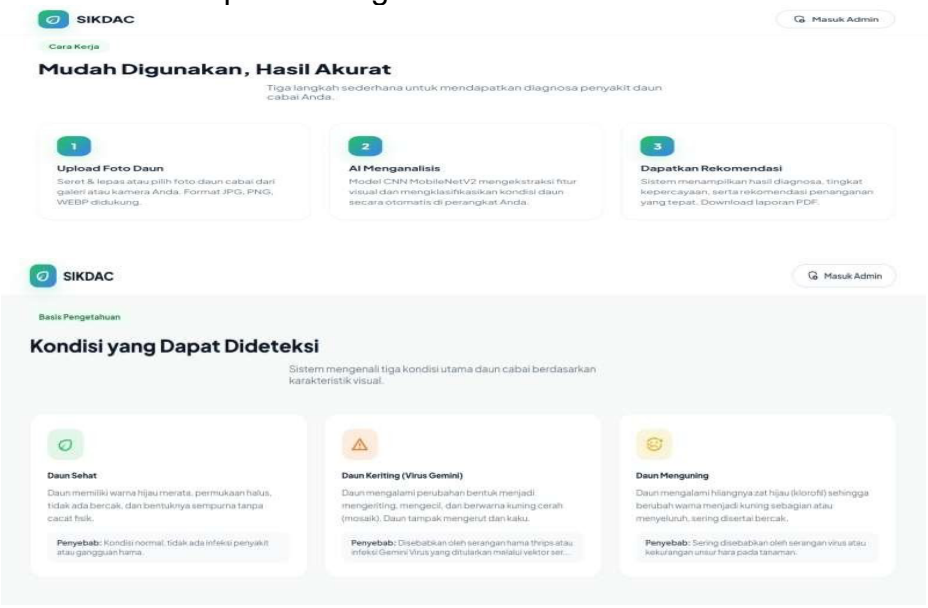
Halaman Landing (Beranda)

Halaman landing merupakan halaman pertama yang ditampilkan ketika pengguna mengakses sistem.



Gambar 2. Tampilan Halaman *Landing* (Beranda) Sistem SIKDAC

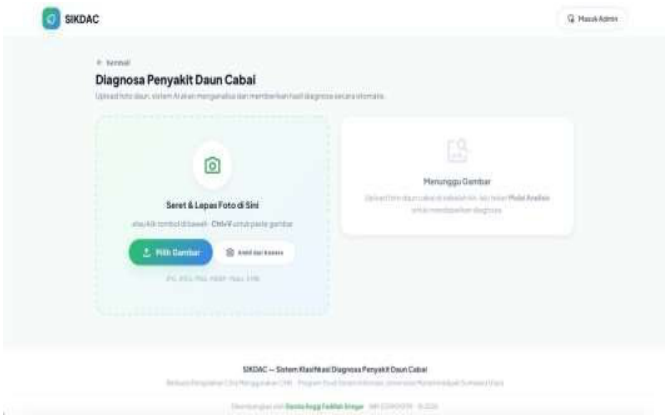
Halaman ini dirancang untuk memberikan gambaran umum mengenai fungsi sistem sebelum pengguna diarahkan ke halaman proses diagnosa, sehingga pengguna baru dapat memahami cara kerja sistem terlebih dahulu. Halaman ini menampilkan judul utama sistem, statistik singkat (jumlah data latih, jumlah kondisi yang dapat dideteksi, dan target akurasi), penjelasan alur kerja sistem dalam tiga langkah, informasi mengenai tiga kondisi daun cabai yang dapat dideteksi beserta karakteristik visualnya, serta tombol ajakan (call-to-action) untuk memulai proses diagnosa.



Gambar 3. Tampilan Bagian "Cara Kerja" dan Kondisi yang Dideteksi

Halaman Diagnosa — Unggah Citra Daun Cabai

Setelah pengguna menekan tombol "Mulai Diagnosa Sekarang" pada halaman landing, sistem akan menampilkan halaman proses analisis. Pada perangkat dengan lebar layar desktop, halaman ini menerapkan tata letak dua kolom, yaitu kolom kiri untuk area unggah dan pratinjau citra, serta kolom kanan untuk menampilkan tahapan proses analisis dan hasil diagnosa, sehingga tampilan tidak memenuhi seluruh layar dan tetap proporsional. Pengguna dapat mengunggah citra daun cabai melalui tiga cara, yaitu menyeret dan melepas berkas (drag and drop), menekan tombol "Pilih Gambar" untuk membukapenjelajah berkas, atau menempelkan (paste) gambar langsung dari papan klip menggunakan kombinasi tombol Ctrl+V.



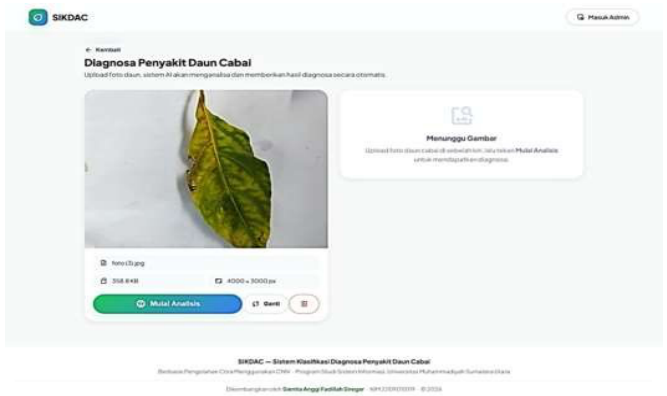
Gambar 4. Tampilan Halaman Unggah Citra (Sebelum Ada Gambar)

Sistem

Menerapkan validasi pada sisi klien terhadap berkas yang diunggah, meliputi validasi jenis berkas (hanya menerima format JPG, JPEG, PNG, dan WEBP) serta validasi ukuran berkas (maksimal 5 MB). Apabila berkas yang diunggah tidak memenuhi kriteria tersebut, sistem akan menampilkan notifikasi peringatan kepada pengguna beserta alasan penolakan.

Pratinjau Citra Setelah Diunggah

Setelah citra berhasil diunggah dan lolos validasi, sistem menampilkan pratinjau citra beserta informasi metadata berkas, meliputi nama berkas, ukuran berkas, dan resolusi gambar. Pada tahap ini pengguna dapat memilih untuk melanjutkan ke proses analisis, mengganti gambar, atau menghapus gambar yang telah dipilih.

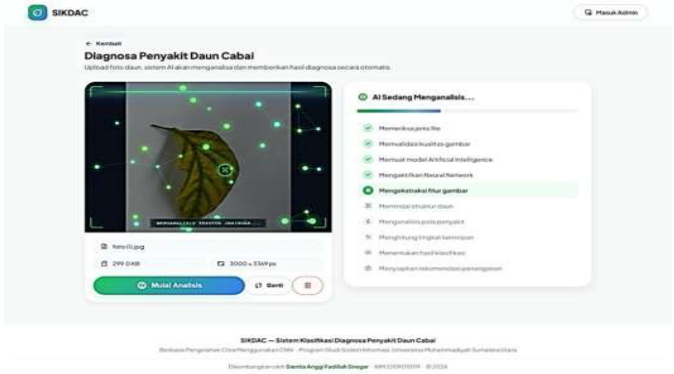


Gambar 5. Tampilan Pratinjau Citra Daun Cabai yang Telah Diunggah

Proses Analisis — Animasi Neural Network Mesh

Saat tombol "Mulai Analisis" ditekan, sistem menjalankan sepuluh

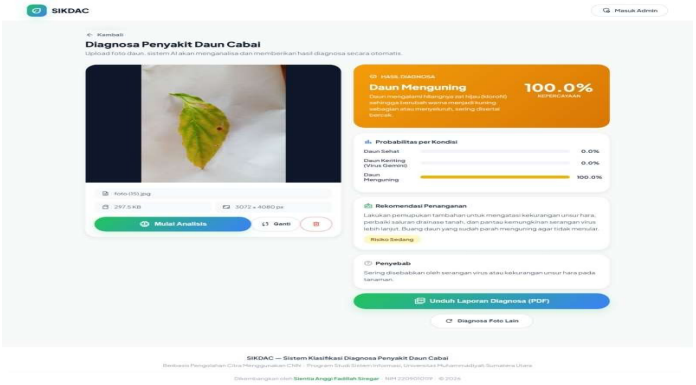
tahapan pemrosesan secara berurutan, mulai dari pemeriksaan berkas hingga penyusunan rekomendasi penanganan, yang ditampilkan melalui progress bar dan panel proses. Selama analisis, animasi Neural Network Mesh berupa titik hijau neon yang saling terhubung ditampilkan pada overlay transparan di atas citra menggunakan HTML5 Canvas. Animasi berhenti secara otomatis dengan efek fade-out setelah model CNN menghasilkan klasifikasi.



Gambar 6. Tampilan Saat Proses Analisis Sedang Berlangsung

Halaman Hasil Diagnosa

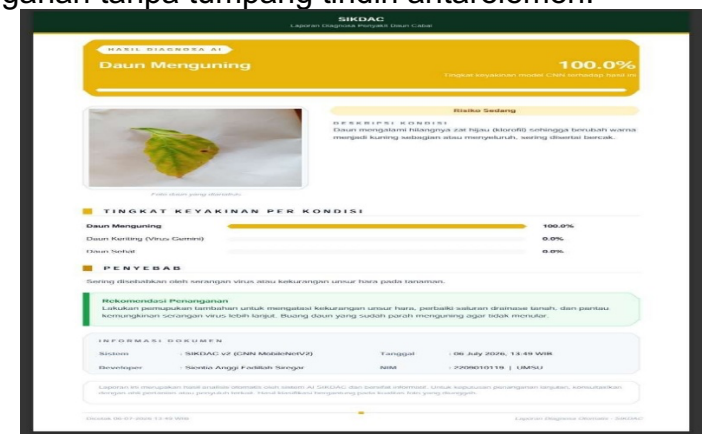
Setelah proses analisis selesai, sistem menampilkan hasil diagnosa pada kolom kanan yang terdiri atas beberapa bagian, yaitu kartu hasil utama yang menampilkan nama kondisi terdeteksi beserta persentase tingkat kepercayaan (confidence level) dengan warna latar yang menyesuaikan tingkat risiko kondisi tersebut, grafik probabilitas untuk ketiga kelas yang menunjukkan seberapa yakin model terhadap masing-masing kemungkinan kondisi, rekomendasi penangananberdasarkan basis pengetahuan yang telah disusun, informasi penyebab kondisi, serta tombol untuk mengunduh laporan hasil diagnosa dalam format PDF.



Gambar 7. Tampilan Hasil Diagnosa dan Grafik Probabilitas

Laporan Hasil Diagnosa dalam Format PDF

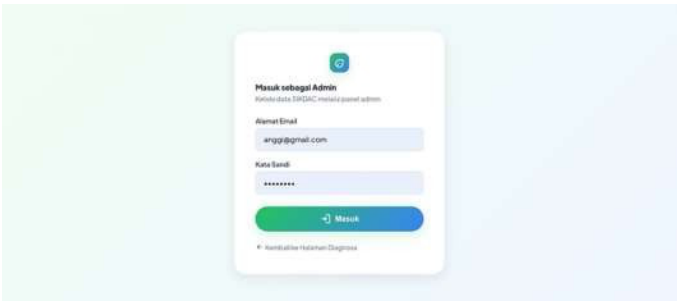
Sistem menyediakan fitur pengunduhan laporan diagnosis dalam format PDF dengan desain modern, tata letak rapi, ukuran citra proporsional, dan teks yang mudah dibaca. Laporan memuat kondisi terdeteksi, tingkat kepercayaan model, deskripsi visual, gejala, penyebab, rekomendasi penanganan, serta langkah pencegahan tanpa tumpang tindih antarelemen.



Gambar 8. Tampilan Laporan PDF Hasil Diagnosa

Halaman Masuk (Login) Admin

Panel admin hanya dapat diakses oleh pengguna yang memiliki hak akses administrator melalui proses autentikasi menggunakan alamat surel dan kata sandi. Kata sandi disimpan pada basis data dalam bentuk hash menggunakan algoritma bcrypt untuk menjaga keamanan data kredensial admin.



Gambar 9. Tampilan Halaman Login Admin

Halaman Dashboard Admin

Setelah berhasil masuk, admin diarahkan ke halaman *dashboard* yang menampilkan ringkasan statistik sistem, seperti jumlah total diagnosa yang telah dilakukan, distribusi kondisi yang paling sering terdeteksi, serta grafik aktivitas

diagnosa dari waktu ke waktu.

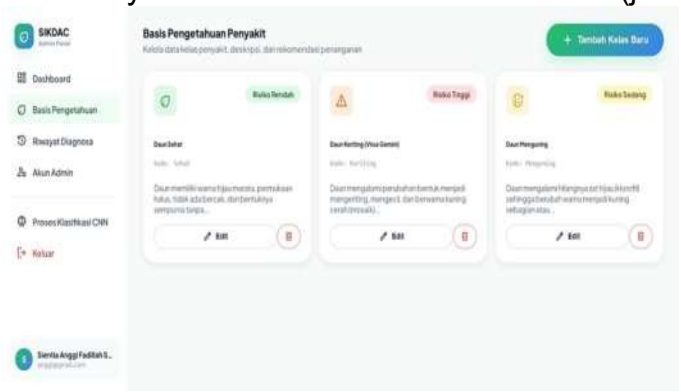


Gambar 10. Tampilan *Dashboard* Admin

Pada bagian menu navigasi (*sidebar*), tersedia menu "Proses Klasifikasi CNN" yang memungkinkan admin untuk langsung mengakses dan mencoba fitur diagnosa pada antarmuka publik tanpa perlu membuka tab atau perangkat lain.

Halaman Manajemen Basis Pengetahuan Penyakit

Halaman ini digunakan oleh admin untuk mengelola data basis pengetahuan penyakit daun cabai yang meliputi kode kelas, nama penyakit, deskripsi visual, gejala, penyebab, rekomendasi penanganan, langkah pencegahan, tingkat bahaya, ikon, dan warna tema untuk masing-masing kondisi. Admin dapat menambah, mengubah, maupun menghapus data kelas penyakit melalui formulir yadisediakan dalam bentuk modal (jendela dialog).

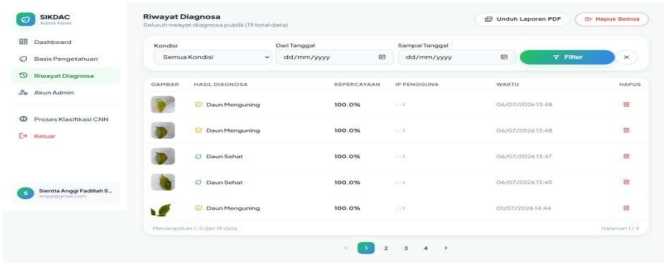


Gambar 11. Tampilan Halaman Manajemen Basis Pengetahuan Penyakit

Halaman Riwayat Diagnosa

Halaman riwayat diagnosis menampilkan gambar yang diunggah, hasil diagnosis, tingkat kepercayaan, alamat IP, dan waktu diagnosis. Fitur yang tersedia meliputi penyaringan berdasarkan kondisi dan rentang tanggal, pagination, penghapusan data secara individual maupun massal dengan

konfirmasi, serta pengunduhan laporan riwayat diagnosis dalam format PDF yang rapi dan formal.



Gambar 12. Tampilan Halaman Riwayat Diagnosa



Gambar 13. Tampilan Laporan PDF Riwayat Diagnosa (Admin)

Pengujian Sistem

Pengujian Fungsional Sistem

Tabel 9. Hasil Pengujian Fungsional Sistem (*Black-Box Testing*)

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil
1	Mengunggah citra dengan format JPG/PNG/WEBP	Citra berhasil diunggah dan ditampilkan pada pratinjau	Sesuai
2	Mengunggah citra dengan format tidak didukung (mis. PDF)	Sistem menampilkan notifikasi penolakan	Sesuai
3	Mengunggah citra melebihi ukuran maksimal (>5MB)	Sistem menampilkan notifikasi ukuran berkas terlalu besar	Sesuai
4	Menekan tombol "Mulai Analisis"	Animasi Neural Network Mesh tampil dan proses inferensi berjalan	Sesuai
5	Hasil klasifikasi ditampilkan	Nama kondisi, persentase kepercayaan, dan rekomendasi tampil	Sesuai

6	Mengunduh laporan PDF hasil diagnosa	Berkas PDF terunduh dengan data yang sesuai	Sesuai
7	Login admin dengan kredensial salah	Sistem menolak akses dan menampilkan pesan error	Sesuai
8	Login admin dengan kredensial benar	Admin diarahkan ke halaman dashboard	Sesuai
9	Menambah data kelas penyakit baru	Data tersimpan dan tampil pada daftar	Sesuai
10	Menghapus satu data riwayat diagnosa	Data dan berkas gambar terkait terhapus	Sesuai
11	Filter riwayat diagnosa berdasarkan tanggal	Data yang tampil sesuai dengan rentang tanggal	Sesuai
12	Mengunduh laporan PDF riwayat diagnosa	Berkas PDF terunduh dengan data sesuai filter aktif	Sesuai

Berdasarkan hasil pengujian pada Tabel 9, seluruh skenario pengujian fungsional (*functional requirement*) yang diujikan menunjukkan hasil yang sesuai dengan harapan, sehingga dapat disimpulkan bahwa sistem telah berjalan sebagaimana mestinya dari sisi fungsionalitas.

Pengujian Sistem Klasifikasi dan Deteksi Daun

Tabel 10. Pengujian Deteksi "Bukan Daun Cabai" (Leaf-Gate)

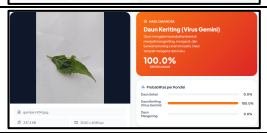
No	Bukti Pengujian	Hasil yang Diharapkan	Hasil Aktual Sistem	Kesimpulan
1		Ditolak sistem — "Gambar Bukan Daun Cabai"	Sistem Menolak: Gambar Bukan Daun Cabai	Sesuai
2		Ditolak sistem — "Gambar Bukan Daun Cabai"	Sistem Menolak: Gambar Bukan Daun Cabai	Sesuai
3		Ditolak sistem — "Gambar Bukan Daun Cabai"	Sistem Menolak: Gambar Bukan Daun Cabai	Sesuai
4		Ditolak sistem — "Gambar Bukan Daun Cabai"	Sistem Menolak: Gambar Bukan Daun Cabai	Sesuai
5		Ditolak sistem — "Gambar Bukan Daun Cabai"	Sistem Menolak: Gambar Bukan Daun Cabai	Sesuai

E-ISSN: 3031-8521
P-ISSN: 3032-1867
Halaman 1492-1509
Volume 3 Nomor 4 Tahun 2026

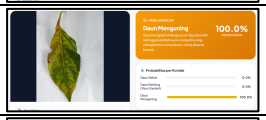
Tabel 11. Pengujian Klasifikasi Daun Sehat

No	Bukti Pengujian	Hasil yang Diharapkan	Hasil Aktual Sistem	Kesimpulan
1		Daun Sehat	Daun Sehat 100.0% Kepercayaan	Sesuai
2		Daun Sehat	Daun Sehat 100.0% Kepercayaan	Sesuai
3		Daun Sehat	Daun Sehat 100.0% Kepercayaan	Sesuai
4		Daun Sehat	Daun Sehat 100.0% Kepercayaan	Sesuai
5		Daun Sehat	Daun Sehat 100.0% Kepercayaan	Sesuai

Tabel 12. Pengujian Klasifikasi Daun Keriting (Virus Gemini)

No	Bukti Pengujian	Hasil yang Diharapkan	Hasil Aktual Sistem	Kesimpulan
1		Daun Keriting (Virus Gemini)	Daun Keriting (Virus Gemini) 99.8% Kepercayaan	Sesuai
2		Daun Keriting (Virus Gemini)	Daun Keriting (Virus Gemini) 100.0% Kepercayaan	Sesuai
3		Daun Keriting (Virus Gemini)	Daun Keriting (Virus Gemini) 100.0% Kepercayaan	Sesuai
4		Daun Keriting (Virus Gemini)	Daun Keriting (Virus Gemini) 100.0% Kepercayaan	Sesuai
5		Daun Keriting (Virus Gemini)	Daun Keriting (Virus Gemini) 99.8% Kepercayaan	Sesuai

Tabel 13. Pengujian Klasifikasi Daun Menguning

No	Bukti Pengujian	Hasil yang Diharapkan	Hasil Aktual Sistem	Kesimpulan
1		Daun Menguning	Daun Menguning 100.0% Kepercayaan	Sesuai
2		Daun Menguning	Daun Menguning 76.2% Kepercayaan	Sesuai
3		Daun Menguning	Daun Menguning 100.0% Kepercayaan	Sesuai
4		Daun Menguning	Daun Menguning 100.0% Kepercayaan	Sesuai
5		Daun Menguning	Daun Menguning 100.0% Kepercayaan	Sesuai

SIMPULAN

Penelitian ini berhasil mengembangkan Sistem Klasifikasi Diagnosa Penyakit Daun Cabai (SIKDAC) berbasis Convolutional Neural Network (CNN) dengan pendekatan transfer learning menggunakan arsitektur MobileNetV2. Sistem dirancang untuk mengklasifikasikan tiga kondisi daun cabai, yaitu daun sehat, daun keriting akibat virus Gemini, dan daun menguning, melalui aplikasi berbasis web yang menjalankan proses inferensi secara langsung menggunakan TensorFlow.js pada peramban. Model dikembangkan menggunakan 1.200 citra yang dibagi menjadi 960 citra data latih, 120 citra data validasi, dan 120 citra data uji. Hasil evaluasi menunjukkan akurasi sebesar 100% dengan nilai precision, recall, dan F1-score sebesar 1,0000 pada seluruh kelas. Selain itu, pengujian fungsional menggunakan metode black-box testing menunjukkan bahwa seluruh skenario yang diuji berjalan sesuai dengan hasil yang diharapkan. Temuan ini menunjukkan bahwa integrasi MobileNetV2 dan TensorFlow.js berpotensi mendukung identifikasi awal kondisi daun cabai secara otomatis, cepat, dan praktis melalui sistem berbasis web.

Meskipun memperoleh hasil evaluasi yang sangat baik, penelitian ini memiliki keterbatasan karena hanya menggunakan tiga kategori kondisi daun cabai dan dataset sebanyak 1.200 citra, sehingga kemampuan generalisasi model

terhadap variasi kondisi lapangan belum dapat dipastikan. Akurasi 100% pada data uji juga belum cukup untuk menyimpulkan bahwa sistem memiliki ketepatan yang sama ketika digunakan pada citra dengan variasi pencahayaan, latar belakang, sudut pengambilan gambar, dan tingkat keparahan gejala yang berbeda. Penelitian selanjutnya disarankan menggunakan dataset yang lebih besar dan beragam, menguji model pada citra lapangan yang dikumpulkan secara independen, serta menambahkan kelas penyakit lain agar kemampuan klasifikasi menjadi lebih komprehensif. Pengujian lanjutan juga diperlukan untuk mengevaluasi kecepatan inferensi dan konsistensi hasil pada berbagai perangkat. Secara praktis, SIKDAC dapat dikembangkan sebagai alat bantu identifikasi awal bagi petani dan pengguna lainnya untuk memperoleh informasi mengenai kondisi daun cabai beserta rekomendasi penanganan dan pencegahan. Namun, hasil klasifikasi perlu diposisikan sebagai informasi pendukung, bukan pengganti diagnosis oleh tenaga ahli pertanian.

DAFTAR PUSTAKA

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., & Farhan, L. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data, 8*(1), Article 53.

Chen, J., Chen, J., Zhang, D., Sun, Y., & Nanehkaran, Y. A. (2022). Using deep transfer learning for image-based plant disease identification. *Computers and Electronics in Agriculture, 173*, Article 105393.

Dzaky, A. T. R. (2021). Deteksi penyakit tanaman cabai menggunakan metode *Convolutional Neural Network*. *E-Proceeding of Engineering, 8*(2), 3039–3055.

Mulyana Prayoga, Y., Wahyuni, D., & Elisawati. (2025). Deteksi penyakit tanaman cabai berbasis deep learning menggunakan *Convolutional Neural Network* secara realtime. *Jurnal Unitek, 18*(2), 292–302. https://ejurnal.sttdumai.ac.id/index.php/unitek/article/view/1467

Pratama, R., et al. (2023). Penerapan deep learning dengan algoritma CNN untuk deteksi penyakit daun hortikultura. *Jurnal Teknik Elektro, 12*(1), 15–23.

Ratnasari. (2023). *Penyakit daun cabai* [Data set]. Kaggle. https://www.kaggle.com/datasets/ratnasari/penyakit-daun-cabai

Tri, B. N., Ningrum, C., Ni'mah, E. N., Arifin, M. P., Dusea, A., & Dara, W. (2024). Klasifikasi dan pengenalan pola penyakit cabai dengan metode CNN. *Universitas Nusantara PGRI Kediri, 3*, 2828–299.

Zhang, Y., Liu, B., & Chen, X. (2021). Plant disease detection using deep convolutional neural networks. *Procedia Computer Science, 187*, 38–45.