

EVALUASI KINERJA DETEKSI PENYAKIT BUAH CABAI RAWIT BERBASIS CITRA

Ni Wayan Widyasari Damayanti¹, Putu Manik Prihatini², I Putu Oka Wisnawa³

^{1,2,3}Teknologi Informasi, Politeknik Negeri Bali, Indonesia

¹widyasaridamayanti@gmail.com, ²manikprihatini@pnb.ac.id, ³okawisnawa@pnb.ac.id

Abstrak

Penyakit pada buah cabai rawit dapat menurunkan kualitas hasil panen dan menyebabkan kerugian ekonomi bagi petani. Proses identifikasi penyakit masih sering dilakukan secara manual sehingga berpotensi menimbulkan kesalahan akibat kemiripan karakteristik visual antarpenyakit. Di sisi lain, sebagian besar penelitian sebelumnya hanya berfokus pada klasifikasi citra tanpa mampu menunjukkan lokasi gejala pada objek. Selain itu, penggunaan dataset berbasis kondisi lapangan nyata masih relatif terbatas, padahal variasi pencahayaan, sudut pengambilan gambar, dan latar belakang lingkungan dapat memengaruhi performa model deteksi. Penelitian ini bertujuan mengembangkan sistem deteksi penyakit buah cabai rawit berbasis citra digital menggunakan metode *Single Shot Detector* (SSD). Model dilatih menggunakan sekitar 6.000 variasi citra hasil augmentasi data *online* yang berasal dari 1.200 citra asli dengan tiga kategori, yaitu sehat, antraknosa, dan lalat buah. Tahapan penelitian meliputi pengumpulan data, anotasi, pembagian dataset, pelatihan model, dan evaluasi menggunakan metrik *precision*, *recall*, *F1-Score*, serta *mean Average Precision* (mAP@0.5). Hasil pengujian menunjukkan bahwa konfigurasi terbaik diperoleh menggunakan *optimizer* AdamW dengan skema *learning rate* dinamis dan menghasilkan nilai mAP sebesar 83,78%. Sistem mampu memberikan keluaran berupa kategori penyakit, nilai *confidence score*, serta *bounding box* yang menunjukkan lokasi gejala pada buah cabai. Implementasi model ke dalam aplikasi berbasis web menunjukkan bahwa sistem dapat digunakan sebagai alat bantu identifikasi penyakit secara praktis pada kondisi lapangan nyata.

Kata kunci: deteksi objek, SSD, cabai rawit, penyakit tanaman, citra digital

1. Pendahuluan

Cabai rawit (*Capsicum frutescens* L.) ialah komoditas hortikultura krusial di Indonesia dengan permintaan pasar yang tinggi dan bernilai ekonomi bagi petani (Wehfany dkk., 2022). Di Kabupaten Badung, khususnya Kecamatan Mengwi, cabai rawit termasuk salah satu komoditas unggulan yang terus dikembangkan guna meningkatkan produktivitas pertanian daerah. Selain bernilai ekonomi, cabai rawit mengandung *capsaicin* dan senyawa bioaktif lain yang bermanfaat dalam bidang kesehatan maupun industri pangan (Arrufitasari dkk., 2025).

Dalam praktik budidaya, tanaman cabai cukup sensitif terhadap perubahan lingkungan. Curah hujan tinggi, perubahan suhu, serta kelembapan udara yang fluktuatif dapat meningkatkan risiko perkembangan hama dan penyakit tanaman (Budi dkk., 2023). Gangguan yang umum ditemukan pada buah cabai rawit meliputi penyakit antraknosa dan serangan lalat buah. Kedua gangguan tersebut dapat menyebabkan munculnya bercak pada permukaan buah, perubahan warna, pembusukan, serta menurunkan kualitas dan nilai jual hasil panen (Nursalim dkk., 2026). Di sisi lain, gejala visual yang hampir serupa antarpenyakit sering menyulitkan identifikasi secara manual, sehingga keputusan penanganan berpotensi kurang tepat (Shoaib dkk., 2023).

Pemanfaatan kecerdasan buatan telah berkembang sebagai solusi dalam bidang pertanian modern. Teknologi *machine learning* serta *deep learning* banyak dipakai guna mengenali penyakit tanaman melalui citra digital (Shoaib dkk., 2023; Upadhyay dkk., 2025). Sejumlah penelitian terdahulu menunjukkan bahwa metode berbasis *Convolutional Neural Network* (CNN) memiliki kinerja yang baik pada tugas klasifikasi citra. Setiyono dkk. (2023) menerapkan CNN untuk identifikasi tanaman obat Indonesia berdasarkan citra daun, sedangkan Aulia Zahra & Firmansyah (2025) menggunakan CNN guna memberi pembedaan jenis ikan air tawar dan air laut. Selain itu, Aldiani dkk. (2024) memanfaatkan CNN pada sistem absensi berbasis pengenalan wajah. Hasil dari beberapa penelitian tersebut memperlihatkan bahwa CNN efektif dalam mengenali pola visual pada data citra digital. Namun demikian, sebagian besar penelitian tersebut hanya menghasilkan keluaran berupa label kategori tanpa disertai informasi mengenai lokasi objek pada citra. Selain itu, penggunaan dataset yang berasal dari kondisi lapangan nyata masih terbatas, padahal variasi lingkungan seperti pencahayaan, sudut pengambilan gambar, dan latar belakang dapat memengaruhi performa model (Xu dkk., 2024).

Sejalan dengan hal tersebut, metode deteksi objek satu tahap seperti *Single Shot Detector* (SSD)

dapat digunakan sebagai pendekatan yang lebih sesuai. Metode ini mampu melakukan proses klasifikasi sekaligus menentukan lokasi objek dalam satu tahap komputasi, sehingga menghasilkan keluaran berupa label dan *bounding box* secara bersamaan (Juhartini dkk., 2025; Liu dkk., 2016). Penerapan deteksi objek juga telah menunjukkan potensi yang baik dalam berbagai bidang, termasuk sektor pertanian (Khan dkk., 2025).

Berbeda dengan pendekatan klasifikasi citra yang hanya menghasilkan label kategori, metode deteksi objek memungkinkan penyajian informasi lokasi gejala secara lebih detail. Hal ini menjadi penting dalam kasus penyakit buah cabai rawit, karena posisi kerusakan pada buah dapat membantu dalam menentukan langkah penanganan yang tepat. Di sisi lain, keterbatasan dataset yang berasal dari kondisi lapangan nyata masih menjadi tantangan, mengingat variasi pencahayaan, sudut pengambilan, dan latar belakang lingkungan dapat memengaruhi kinerja model. Oleh karena itu, diperlukan pengembangan sistem deteksi objek yang mampu mengidentifikasi jenis penyakit sekaligus menunjukkan letak gejala secara akurat pada kondisi nyata.

Berdasarkan permasalahan tersebut, riset ini mempunyai tujuan guna melakukan pengembangan sistem deteksi penyakit pada buah cabai rawit memakai model *Single Shot Detector* (SSD). Sistem yang dirancang dapat mengidentifikasi tiga kategori utama, yaitu buah sehat, antraknosa, dan serangan lalat buah, serta menampilkan lokasi gejala dalam bentuk *bounding box*. Keunggulan pendekatan ini terletak pada integrasi proses identifikasi dan penentuan lokasi gejala dalam satu tahap, penggunaan data citra lapangan nyata, serta penyajian hasil dalam bentuk visual *bounding box*.

Ruang lingkup penelitian dibatasi pada proses deteksi penyakit buah cabai rawit di area persawahan Kecamatan Mengwi menggunakan dataset citra yang diperoleh dari kondisi lapangan. Kontribusi utama penelitian ini adalah: (1) penerapan metode SSD untuk deteksi penyakit buah cabai rawit berbasis citra lapangan nyata, (2) kemampuan sistem dalam mengidentifikasi kategori penyakit sekaligus menentukan lokasi gejala melalui *bounding box*, serta (3) implementasi model dalam aplikasi berbasis web untuk mendukung penggunaan praktis di lapangan.

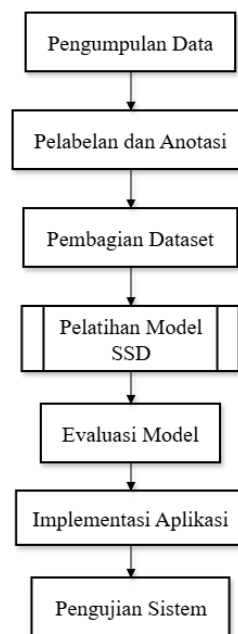
2. Metode

Riset ini memakai pendekatan kuantitatif melalui metode eksperimen guna melakukan pengembangan sistem deteksi penyakit pada buah cabai rawit berbasis citra digital. Pendekatan tersebut dipilih karena kinerja model dapat diukur secara objektif melalui indikator numerik. Pengujian dilakukan untuk mengetahui kemampuan model dalam mengenali jenis penyakit sekaligus menentukan letak objek pada citra melalui *bounding box*. Variasi latar belakang citra, seperti daun, batang, tanah, serta objek

yang saling menutupi, menjadi tantangan dalam proses deteksi (Xu dkk., 2024). Oleh sebab itu, pengevaluasian dilaksanakan memakai metrik *precision*, *recall*, *F1-Score*, dan *mean Average Precision* (mAP) yang umum digunakan pada penelitian deteksi objek (Juhartini dkk., 2025; Padilla dkk., 2021).

Metode yang diterapkan adalah *Single Shot Detector* (SSD), yaitu algoritma deteksi objek berbasis *Convolutional Neural Network* (CNN) yang sanggup melakukan klasifikasi dan lokalisasi objek dalam satu tahap komputasi. SSD dipilih karena memiliki waktu inferensi yang relatif cepat dan efisien, sehingga sesuai digunakan untuk kebutuhan deteksi secara langsung (*real time*) (Cheng dkk., 2022; Liu dkk., 2016).

Secara umum, riset ini dijalankan menggunakan sejumlah tahap, yakni mengumpulkan data, pelabelan dan anotasi citra, pembagian dataset, pelatihan model, evaluasi model, implementasi sistem berbasis web, serta pengujian akhir. Urutan tahapan riset dipaparkan dalam Gambar 1.



Gambar 1. Tahapan Pengembangan Model SSD untuk Deteksi Penyakit Buah Cabai Rawit dan Implementasi Web

2.1 Pengumpulan Data

Data penelitian diperoleh melalui pengambilan citra buah cabai rawit secara langsung di wilayah persawahan Kecamatan Mengwi, Kabupaten Badung. Proses akuisisi data dilakukan selama periode Januari hingga Maret 2026, yang mencakup kondisi musim pancaroba hingga musim penghujan. Pemilihan periode ini bertujuan untuk menangkap variasi kondisi lingkungan yang beragam, mengingat sebagian besar penyakit pada tanaman cabai rawit cenderung berkembang pada tingkat kelembapan tinggi selama musim hujan (Budi dkk., 2023).

Pengambilan citra dilakukan di beberapa lokasi, yaitu Desa Sading, Desa Mengwi, Desa Gulingan, Desa Anggungan, dan Desa Penarungan. Variasi kondisi pengambilan juga diperhatikan, di mana citra diambil pada pagi dan sore hari dengan kondisi cuaca yang beragam, mulai dari cerah hingga mendung. Hal ini dilakukan untuk memperkaya variasi basis pengetahuan dataset, sehingga model yang dikembangkan mempunyai kekuatan generalisasi yang lebih baik terhadap realita di lapangan.

Seluruh citra direkam menggunakan perangkat bergerak berbasis *Android* dan *iOS*. Penggunaan data lapangan secara langsung diharapkan dapat meningkatkan kemampuan model dalam beradaptasi terhadap kondisi implementasi sebenarnya (Xu dkk., 2024).

Jumlah data asli yang digunakan dalam penelitian ini sebanyak 1.200 citra. Dataset tersebut terdiri atas tiga kategori, yaitu antraknosa, lalat buah, dan cabai sehat dengan distribusi kelas yang relatif seimbang. Kategori penyakit mencakup 770 citra yang terdiri atas 380 citra antraknosa dan 390 citra lalat buah, sedangkan kategori sehat berjumlah 430 citra. Untuk meningkatkan keberagaman sampel pelatihan, diterapkan teknik augmentasi data pada tahap pelatihan model sehingga total variasi data pelatihan efektif meningkat sekitar 6.000 citra. Distribusi dataset asli dapat dilihat pada Tabel 1.

Tabel 1. Distribusi Dataset Asli

Kategori	Jumlah
Antraknosa	380
Lalat Buah	390
Sehat	430
Total	1200

Komposisi data yang mencakup berbagai kondisi lingkungan dan proporsi kelas yang relatif seimbang diharapkan mampu meminimalkan bias model terhadap kelas tertentu.

Selain dataset asli, proses augmentasi menghasilkan variasi citra tambahan yang digunakan selama pelatihan model. Rincian jumlah data setelah augmentasi ditunjukkan pada Tabel 2.

Tabel 2. Distribusi Data Setelah Augmentasi

Jenis Data	Jumlah
Dataset asli	1.200
Variasi hasil augmentasi	4.800
Total data pelatihan efektif	6.000

Kombinasi variasi lokasi pengambilan, kondisi pencahayaan, serta penerapan augmentasi selama proses pelatihan diharapkan mampu meningkatkan kemampuan generalisasi model dan mengurangi kecenderungan bias terhadap kondisi tertentu.

2.2 Pelabelan dan Anotasi Data

Setiap citra kemudian diberi label sesuai kondisi buah, yaitu antraknosa, lalat buah, atau sehat. Proses

identifikasi awal dilakukan berdasarkan gejala visual yang tampak pada permukaan buah, seperti bercak, perubahan warna, pembusukan, maupun kerusakan fisik lainnya.

Untuk meningkatkan ketepatan label, hasil klasifikasi awal diverifikasi oleh penyuluh pertanian dan tenaga ahli penyakit tanaman dari Badan Penyuluh Pangan Kecamatan Mengwi. Setelah label dinyatakan sesuai, objek buah pada citra ditandai menggunakan *bounding box*.

Data anotasi disimpan dalam format *COCO-JSON*, karena format tersebut banyak digunakan dalam pengembangan sistem deteksi objek modern (Tian dkk., 2024).

2.3 Pembagian Dataset

Dataset yang telah melalui proses anotasi selanjutnya dibagi menjadi tiga subset, yaitu data latih, data validasi, dan data uji. Pembagian data dilakukan menggunakan rasio 80:10:10 yang terdiri atas 960 citra data latih, 120 citra data validasi, dan 120 citra data uji. Rasio tersebut dipilih karena mampu memberikan proporsi data pelatihan yang cukup besar tanpa mengurangi representasi data validasi dan pengujian (Padilla dkk., 2021).

Proses pemisahan dataset dilakukan secara acak menggunakan *seed* 42 untuk menjaga konsistensi hasil eksperimen serta memastikan penelitian dapat direproduksi pada kondisi yang sama. Pembagian data dilakukan sebelum proses augmentasi untuk menghindari terjadinya data leakage antara data latih, validasi, dan data uji. Dengan demikian, citra hasil augmentasi hanya diterapkan pada data latih sehingga evaluasi model tetap dilakukan menggunakan data validasi dan data uji asli yang belum mengalami transformasi.

Selain mempertimbangkan proporsi jumlah data, distribusi kelas pada setiap subset juga dijaga agar tetap relatif seimbang. Pendekatan ini bertujuan untuk meminimalkan bias model terhadap kategori tertentu serta memastikan proses pelatihan dan evaluasi berlangsung secara lebih representatif. Distribusi pembagian dataset ditunjukkan pada Tabel 3.

Tabel 3. Pembagian Dataset

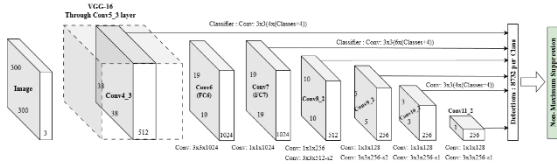
Jenis Data	Jumlah
Data Latih	960
Data Validasi	120
Data Uji	120

2.4 Pelatihan Model

Pelatihan model pada penelitian ini menggunakan arsitektur *Single Shot Detector* (SSD-300) dengan *backbone* VGG-16 sebagai ekstraktor fitur utama. *Backbone* VGG-16 berperan guna melakukan ekstraksi representasi fitur dari citra masukan melalui serangkaian lapisan konvolusi yang mendalam. Arsitektur ini dipilih karena mampu

mendeteksi objek pada berbagai skala, sehingga sesuai untuk kasus deteksi buah cabai rawit yang memiliki ukuran relatif kecil hingga sedang (Cheng dkk., 2022; Liu dkk., 2016).

Arsitektur SSD-300 dengan *backbone* VGG-16 dalam riset ini dipaparkan dalam Gambar 2.



Gambar 2. Arsitektur SSD-300 dengan *Backbone* VGG-16 (Sumber: Liu dkk., 2016)

Secara prinsip, mekanisme kerja SSD dimulai dengan memasukkan citra berukuran 300×300 piksel ke dalam jaringan VGG-16 untuk menghasilkan peta fitur (*feature map*). Selanjutnya, beberapa layer tambahan digunakan untuk menghasilkan *feature map* dengan resolusi berbeda (*multi-scale feature maps*). Pada setiap *feature map* tersebut, model secara langsung melakukan prediksi lokasi objek dalam bentuk *bounding box* serta klasifikasi kategori objek dalam satu tahap proses. Pendekatan ini menghasilkan proses deteksi yang lebih efisien dibandingkan metode dua tahap (Cheng dkk., 2022; Liu dkk., 2016). Setiap prediksi menggunakan *default box* dengan variasi skala dan rasio aspek, kemudian hasil deteksi disaring menggunakan *Non-Maximum Suppression* (NMS) dengan ambang IoU sebesar 0,5 untuk menghilangkan prediksi yang saling tumpang tindih.

Sebelum proses pelatihan dilakukan, seluruh citra pada data latih terlebih dahulu melalui tahap *preprocessing* berupa perubahan ukuran citra menjadi 300×300 piksel serta normalisasi data agar sesuai dengan spesifikasi input model. Selain *preprocessing*, diterapkan pula beberapa teknik augmentasi data yang terdiri atas *random crop resize*, *mosaic augmentation*, *color jitter*, dan *cutout augmentation*. *Random crop resize* digunakan untuk meningkatkan kemampuan model dalam mengenali objek pada variasi posisi dan ukuran yang berbeda. *Mosaic augmentation* diterapkan dengan mengombinasikan beberapa citra dalam satu frame sehingga model mampu mempelajari objek dalam latar yang lebih kompleks. *Color jitter* digunakan untuk menghasilkan variasi pencahayaan, kontras, dan saturasi warna agar model lebih adaptif terhadap perubahan kondisi lingkungan. Sementara itu, *cutout augmentation* diterapkan dengan menutupi sebagian area citra secara acak untuk meningkatkan ketahanan model terhadap gangguan visual maupun *occlusion*. Seluruh proses augmentasi dilakukan secara dinamis pada setiap epoch pelatihan sehingga model memperoleh variasi sampel citra yang lebih beragam selama proses training berlangsung (Xu dkk., 2023).

Proses pelatihan model dijalankan menggunakan GPU NVIDIA RTX 3060 6GB. Pada tahap awal

eksperimen, digunakan *optimizer* SGD untuk menguji berbagai kombinasi *hyperparameter*. Selanjutnya, pada tahap optimasi lanjutan digunakan *optimizer* AdamW dengan skema *learning rate* dinamis berbasis *cosine decay* untuk meningkatkan stabilitas proses pelatihan dan performa model. Fungsi *loss* yang digunakan adalah *MultiBox Loss* yang umum diterapkan pada SSD, serta proses seleksi *bounding box* dilakukan menggunakan *Non-Maximum Suppression* (NMS) dengan threshold IoU 0,5. Untuk memperoleh konfigurasi terbaik, dilakukan pengujian beberapa kombinasi *hyperparameter* sebagaimana pada Tabel 4.

Tabel 4. Kombinasi *Hyperparameter*

No	Epoch	Batch size	Learning rate
1	150	16	5e-4
2	150	8	5e-4
3	150	16	1e-3
4	150	8	1e-3
5	100	16	5e-4
6	100	16	1e-3
7	100	8	5e-4
8	100	8	1e-3

Tabel 4 menyajikan variasi kombinasi *hyperparameter* yang digunakan, meliputi jumlah *epoch*, *batch size*, dan *learning rate*. Parameter *epoch* diuji pada nilai 100 dan 150 untuk melihat pengaruh durasi pelatihan terhadap performa model. *Batch size* yang digunakan adalah 8 dan 16, yang memengaruhi jumlah data dalam satu iterasi pelatihan serta berdampak pada stabilitas proses pembelajaran. Sementara itu, *learning rate* diuji pada nilai 5e-4 dan 1e-3 untuk menentukan tingkat pembaruan bobot selama proses optimasi. Berbagai kombinasi ini diuji guna memperoleh parameter yang paling optimal dalam meningkatkan akurasi deteksi.

Pemilihan model terbaik dilakukan berdasarkan nilai mAP@0.5 tertinggi pada data validasi. Apabila hasil yang diperoleh belum menunjukkan performa yang optimal, maka akan dilakukan pengujian tambahan dengan variasi *hyperparameter* lainnya guna meningkatkan kinerja model secara lebih maksimal.

2.5 Evaluasi Model

Model yang sudah dilatih dilakukan evaluasi memakai data uji yang tidak dilibatkan dalam tahapan pelatihan ataupun validasi. Pengujian ini mempunyai tujuan guna mengukur kekuatan model guna mendeteksi jenis penyakit serta menentukan lokasi objek pada citra buah cabai rawit.

Kinerja model diukur memakai metrik *precision*, *recall*, *F1-Score*, *Average Precision* (AP), dan *mean Average Precision* (mAP). Dalam penelitian ini, *precision* merepresentasikan ketepatan

hasil deteksi penyakit, sedangkan *recall* menunjukkan kemampuan model guna memperoleh semua objek penyakit pada citra. *F1-Score* digunakan untuk mengevaluasi keseimbangan antara kedua metrik tersebut (Padilla dkk., 2021).

$$Precision = \frac{TP}{TP+FP} \tag{1}$$

Pada Persamaan (1), *TP (True Positive)* menyatakan jumlah prediksi yang sesuai dengan objek penyakit, sedangkan *FP (False Positive)* merupakan prediksi yang keliru namun teridentifikasi sebagai objek penyakit.

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

Pada Persamaan (2), *FN (False Negative)* menunjukkan jumlah objek penyakit yang tidak berhasil terdeteksi oleh model.

$$F1-Score = \frac{2TP}{2TP+FP+FN} \tag{3}$$

Nilai *F1-Score* pada Persamaan (3) ialah rerata harmonis yang menggabungkan *precision* dan *recall*, sehingga mampu merepresentasikan keseimbangan kinerja model secara keseluruhan.

Selain aspek klasifikasi, ketepatan dalam menentukan lokasi objek juga dievaluasi menggunakan *Intersection over Union (IoU)* dengan ambang batas 0,5. Prediksi dinyatakan benar apabila memiliki tingkat tumpang tindih yang memadai dengan anotasi sebenarnya.

$$IoU = \frac{Area(B_{pred} \cap B_{gt})}{Area(B_{pred} \cup B_{gt})} \tag{4}$$

Pada Persamaan (4), *B_{pred}* menunjukkan *bounding box* hasil prediksi model, sedangkan *B_{gt}* merupakan *bounding box ground truth*. Operasi irisan (\cap) merepresentasikan area yang saling tumpang tindih, sedangkan gabungan (\cup) menunjukkan total area kedua *bounding box* (Padilla dkk., 2021).

Selanjutnya, nilai *Average Precision (AP)* dihitung untuk setiap kelas, yaitu antraknosa, lalat buah, dan sehat. Nilai ini diperoleh dari luas area di bawah kurva *precision-recall*, sebagaimana dirumuskan berikut:

$$AP = \sum_{k=1}^n P(k) \Delta R(k) \tag{5}$$

Pada Persamaan (5), *P(k)* menyatakan nilai *precision* pada titik ke- *k*, sedangkan $\Delta R(k)$ menunjukkan perubahan nilai *recall* pada titik tersebut.

Untuk memperoleh gambaran performa model secara menyeluruh, nilai AP dari seluruh kelas dirata-ratakan menjadi *mean Average Precision (mAP)* sebagai berikut:

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \tag{6}$$

Pada persamaan (6), *N* menyatakan jumlah kelas yang diuji, sedangkan *AP_i* merupakan nilai AP pada masing-masing kelas. Metrik mAP digunakan sebagai indikator utama karena mampu merepresentasikan performa model secara komprehensif, baik dari aspek klasifikasi maupun lokalisasi objek.

Di sisi lain, dipakai *confusion matrix* guna menganalisis distribusi prediksi benar dan kesalahan antar kelas, sebagaimana ditunjukkan pada Tabel 5, sehingga dapat diketahui kecenderungan model dalam melakukan kesalahan klasifikasi pada masing-masing kategori penyakit.

Tabel 5. Ilustrasi *Confusion matrix* Tiga Kelas

Aktual / Prediksi	Lalat Buah	Antraknosa	Sehat
Lalat Buah	TP	FP	FP
Antraknosa	FN	TP	FP
Sehat	FN	FN	TP

Tabel 5 menunjukkan ilustrasi *confusion matrix* untuk tiga kelas penelitian. Baris menunjukkan kelas aktual, sedangkan kolom menunjukkan hasil prediksi model. Nilai *True Positive (TP)* menunjukkan prediksi yang sesuai, *False Positive (FP)* menunjukkan kesalahan klasifikasi, dan *False Negative (FN)* menunjukkan objek yang tidak terdeteksi. Analisis ini digunakan guna melakukan identifikasi kelas yang paling kerap ditemukan kesalahan, alhasil dapat diketahui kekurangan model dalam memberikan pembedaan jenis penyakit pada buah cabai rawit.

Model terbaik dipilih berdasarkan nilai mAP tertinggi. Apabila hasil yang diperoleh belum optimal, maka dilakukan pengujian lanjutan dengan variasi parameter untuk meningkatkan kinerja model.

2.6 Implementasi Aplikasi

Model dengan performa terbaik setelah itu dilakukan integrasi ke dalam aplikasi berbasis web memakai *Streamlit*. Platform ini dipilih karena mendukung integrasi model berbasis Python secara langsung sehingga proses inferensi, pengujian, dan visualisasi hasil deteksi dapat dilakukan secara efisien melalui antarmuka web.

Sistem dirancang agar pengguna dapat mengunggah citra buah cabai rawit melalui halaman web, kemudian model melakukan deteksi otomatis. Hasil keluaran sistem berupa kategori penyakit, nilai keyakinan prediksi, serta visualisasi *bounding box* yang menunjukkan lokasi gejala pada objek, sehingga memudahkan interpretasi hasil deteksi.

2.7 Pengujian Aplikasi

Pengujian aplikasi berbasis web dilakukan untuk menguji seluruh fungsi sistem setelah integrasi model. Pengujian dilakukan menggunakan metode *Black Box Testing* mencakup proses unggah citra, keberhasilan model melakukan inferensi, tampilan hasil deteksi berupa kategori penyakit dan *bounding box*, serta waktu respons sistem saat memproses gambar (Wulandari dkk., 2022).

Selain itu, dilakukan pengamatan terhadap kemampuan sistem dalam mendeteksi objek pada beberapa kondisi citra, seperti perbedaan pencahayaan, sudut pengambilan gambar, dan latar belakang lingkungan pertanian. Hasil pengujian ini digunakan untuk menilai kesiapan sistem sebagai media bantu identifikasi penyakit buah cabai rawit.

3. Hasil dan Pembahasan

Hasil deteksi pada riset ini diperoleh melalui proses pelatihan model *Single Shot Detector* (SSD) menggunakan dataset citra buah cabai rawit yang telah melalui tahap anotasi. Evaluasi kinerja model dilakukan memakai metrik *precision*, *recall*, *F1-Score*, serta *mean Average Precision* (mAP@0.5) untuk menilai kemampuan model dalam mengenali kategori objek sekaligus menentukan lokasi gejala pada citra.

Berdasarkan serangkaian pengujian yang dilakukan, diperoleh variasi performa model pada setiap kombinasi *hyperparameter*. Hasil pengujian delapan eksperimen awal disajikan pada Tabel 6, yang memperlihatkan perbandingan nilai evaluasi mAP untuk masing-masing konfigurasi.

Tabel 6. Hasil Pengujian Model SSD pada Berbagai Kombinasi *Hyperparameter*

ID	Epoch	Batch	LR	Optimizer	mAP (%)
EXP-001	150	16	5e-4	SGD	72
EXP-002	150	8	5e-4	SGD	72
EXP-003	150	16	1e-3	SGD	76
EXP-004	150	8	1e-3	SGD	76
EXP-005	100	16	5e-4	SGD	77
EXP-006	100	16	1e-3	SGD	72
EXP-007	100	8	5e-4	SGD	66
EXP-008	100	8	1e-3	SGD	76

Berdasarkan hasil pada Tabel 6, nilai mAP yang diperoleh berada pada rentang 66% hingga 77%, yang menunjukkan bahwa variasi parameter pelatihan memberikan pengaruh terhadap performa model secara keseluruhan. Perubahan nilai *epoch*, *batch size*, dan *learning rate* memberikan dampak yang berbeda terhadap stabilitas dan akurasi model.

Konfigurasi dengan *learning rate* yang lebih tinggi (1e-3) pada EXP-003 dan EXP-004 menunjukkan peningkatan performa dibandingkan konfigurasi lain, yang mengindikasikan bahwa laju pembelajaran yang lebih agresif mampu

mempercepat konvergensi pada tahap awal pelatihan (Upadhyay dkk., 2025). Namun demikian, penggunaan *batch size* kecil (8) cenderung menghasilkan fluktuasi performa, terutama pada nilai *recall* dan *F1-Score*. Sebaliknya, penggunaan *batch size* yang lebih besar (16) menghasilkan proses pelatihan yang lebih stabil dengan variasi performa yang lebih kecil antar *epoch*. Pada EXP-005, kombinasi *epoch* 100, *batch size* 16, dan *learning rate* 5e-4 menghasilkan nilai mAP sebesar 77%, yang merupakan performa terbaik pada tahap eksperimen awal.

Hasil ini menunjukkan bahwasanya kenaikan jumlah *epoch* tidak senantiasa memiliki perbandingan lurus terhadap kenaikan performa model, melainkan dipengaruhi oleh interaksi antar parameter pelatihan. Meskipun demikian, performa model pada delapan percobaan awal masih menunjukkan ketidakseimbangan antar kelas, terutama pada kategori antraknosa dan lalat buah. Kondisi ini menjadi dasar dilakukan percobaan lanjutan dengan pendekatan optimasi yang berbeda.

Untuk meningkatkan performa model, dilakukan percobaan lanjutan dengan menggunakan *optimizer* AdamW dan skema *learning rate* dinamis pada EXP-009. Hasil pengujian konfigurasi ini dipaparkan dalam Tabel 7.

Tabel 7. Hasil Optimasi Model SSD Menggunakan AdamW (EXP-009)

ID	Epoch	Batch	LR (Max-Min)	Optimizer	mAP (%)
EXP-009	150	6	3e-4 – 1e-6	AdamW	83,78

Hasil pada Tabel 7 menunjukkan bahwa penggunaan *optimizer* AdamW dengan skema *learning rate* dinamis mampu meningkatkan nilai mAP hingga mencapai 83,78%, yang merupakan performa tertinggi dibandingkan seluruh konfigurasi sebelumnya. Performa terbaik diperoleh pada *epoch* ke-125, yang menunjukkan titik optimal model sebelum memasuki kondisi stabil. Peningkatan ini mengindikasikan bahwa strategi optimasi yang adaptif mampu membantu model dalam menemukan solusi yang lebih optimal pada ruang parameter (*loss landscape*).

Selain nilai mAP, evaluasi juga dilakukan menggunakan metrik *precision*, *recall*, dan *F1-Score* rata-rata. Hasil evaluasi model terbaik ditunjukkan pada Tabel 8.

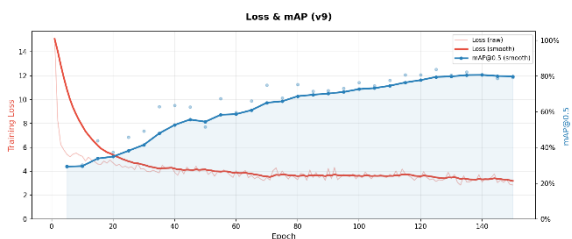
Tabel 8. Hasil Evaluasi Rata-Rata Model Terbaik (EXP-009)

Precision (%)	Recall (%)	F1-Score (%)
92	86	89

Nilai *precision* rata-rata sebesar 92% menunjukkan bahwa sebagian besar prediksi yang dihasilkan model telah sesuai dengan kategori objek sebenarnya. Sementara itu, *recall* sebesar 86%

menunjukkan bahwa model mampu mendeteksi sebagian besar objek penyakit pada citra. Nilai *F1-Score* sebesar 89% memperlihatkan bahwa model memiliki keseimbangan yang baik antara ketepatan prediksi dan kemampuan deteksi objek.

Perkembangan nilai *loss* dan *mAP* selama pelatihan ditunjukkan pada Gambar 3.

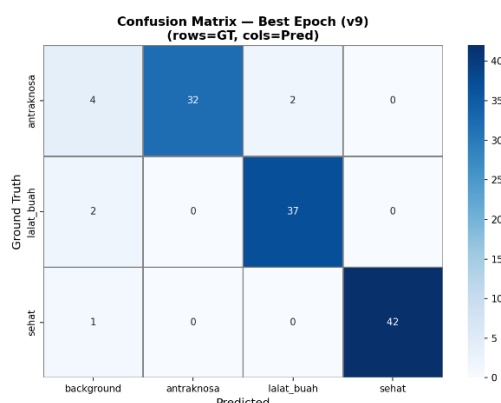


Gambar 3. Grafik *Loss* dan *mAP* terhadap *Epoch*

Grafik pada Gambar 3 memperlihatkan bahwa nilai *loss* mengalami penurunan secara signifikan pada fase awal pelatihan, kemudian melandai dan mencapai kondisi stabil pada *epoch* selanjutnya, yang menunjukkan proses konvergensi model. Di sisi lain, nilai *mAP* menunjukkan tren meningkat secara konsisten hingga mendekati nilai maksimum, yang mengindikasikan kemampuan model dalam melakukan generalisasi terhadap data uji semakin baik.

Evaluasi menggunakan data uji menunjukkan bahwa model mampu mendeteksi objek pada ketiga kategori yang digunakan dalam penelitian ini. Pada kategori buah sehat, hasil deteksi menunjukkan *confidence score* yang relatif tinggi. Sebaliknya, pada kategori antraknosa dan lalat buah, nilai *confidence* cenderung lebih bervariasi, terutama pada keadaan citra dengan pencahayaan tidak merata, *occlusion* (objek tertutup), serta latar belakang yang kompleks.

Untuk melihat distribusi kesalahan prediksi antar kelas, digunakan *confusion matrix* yang dipaparkan dalam Gambar 4.



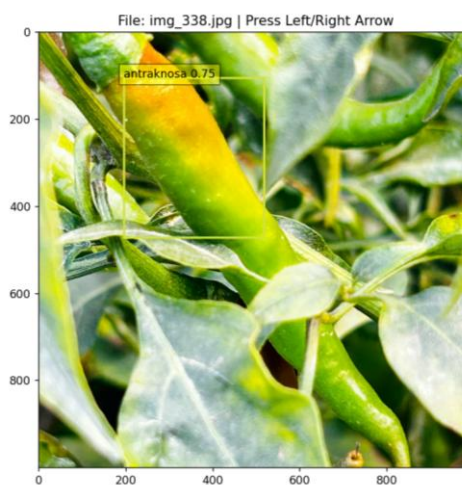
Gambar 4. *Confusion matrix* Hasil Deteksi Model

Berdasarkan *confusion matrix* yang ditampilkan pada Gambar 4, sebagian besar data pada kelas sehat berhasil diklasifikasikan dengan benar, sehingga menunjukkan tingkat *recall* yang tinggi pada kategori tersebut. Namun demikian, masih ditemukan

kesalahan klasifikasi antara kelas antraknosa dan lalat buah.

Fenomena ini menunjukkan bahwa kedua kategori memiliki karakteristik visual yang saling mendekati, terutama pada fase awal infeksi, di mana pola lesi yang muncul cenderung serupa alhasil mempersulit model guna memberikan pembedaan keduanya dengan cara yang konsisten.

Selama proses pengujian, masih ditemukan beberapa kesalahan deteksi, seperti objek yang tidak teridentifikasi atau diklasifikasikan ke kategori yang tidak sesuai. Contoh kesalahan deteksi dapat ditampilkan pada Gambar 5.



Gambar 5. Kesalahan Deteksi

Kesalahan tersebut umumnya terjadi pada objek dengan ukuran relatif kecil, tingkat kontras rendah terhadap latar belakang, atau kondisi objek yang tertutup sebagian oleh daun (*occlusion*), sehingga fitur visual yang ditangkap menjadi kurang representatif bagi model.

Model terbaik yang diperoleh kemudian diintegrasikan ke dalam aplikasi berbasis web. Tampilan antarmuka sistem mampu diamati dalam Gambar 6.

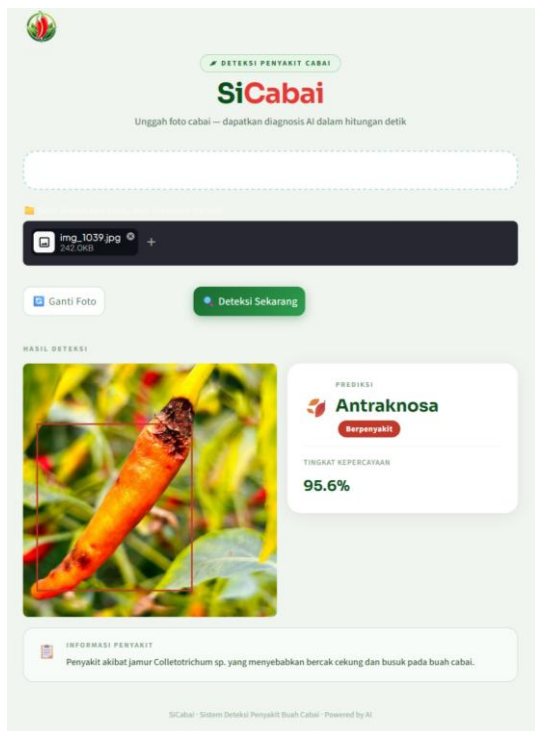


Gambar 6. Tampilan Antarmuka Aplikasi

Antarmuka dirancang dengan pendekatan sederhana dan informatif, sehingga memudahkan pengguna dari kalangan non-teknis, seperti petani atau praktisi pertanian, dalam mengoperasikan sistem

tanpa memerlukan pemahaman teknis yang mendalam.

Setelah citra diproses, sistem menampilkan hasil berupa kategori penyakit, *confidence score*, serta *bounding box* yang menunjukkan lokasi objek. Hasil deteksi ditampilkan pada Gambar 7.



Gambar 7. Hasil Deteksi dengan *Bounding Box*

Selain menampilkan hasil prediksi, sistem juga dilengkapi dengan informasi singkat terkait penyakit yang terdeteksi, sehingga pengguna tidak hanya memperoleh hasil identifikasi, tetapi juga pemahaman tambahan mengenai kondisi buah cabai yang dianalisis.

Waktu yang dibutuhkan untuk memproses satu citra berada pada kisaran 2–3 detik. Hal tersebut memperlihatkan bahwasanya sistem mempunyai tingkat respons yang cukup baik dan berpotensi digunakan dalam skenario penggunaan semi *real time* pada aplikasi berbasis web.

Secara keseluruhan, hasil penelitian menunjukkan bahwa pendekatan deteksi objek mampu mengidentifikasi penyakit buah cabai rawit sekaligus menentukan lokasi gejala pada citra.

Dibandingkan dengan metode klasifikasi citra, pendekatan ini memberikan keunggulan dalam menyediakan informasi spasial berupa posisi objek yang terdeteksi, sehingga mendukung analisis visual yang lebih komprehensif terhadap kondisi penyakit tanaman. Hal ini sejalan dengan penelitian Juhartini dkk. (2025) yang menyatakan bahwa metode deteksi objek mampu memberikan informasi lokasi selain klasifikasi.

Walaupun demikian, riset ini masih mempunyai sejumlah keterbatasan. Keterbatasan jumlah dataset menyebabkan variasi kondisi objek belum

sepenuhnya terakomodasi, sehingga berdampak pada kemampuan generalisasi model terhadap data baru. Kondisi ini juga dipengaruhi oleh kompleksitas data lapangan yang memiliki variasi pencahayaan, latar belakang, dan kondisi objek yang beragam, sebagaimana dijelaskan oleh Xu dkk. (2024).

Selain itu, kemiripan visual antar kategori masih menyebabkan kesalahan deteksi pada kondisi tertentu. Waktu inferensi yang masih berada pada kisaran beberapa detik menunjukkan bahwa sistem belum sepenuhnya optimal untuk implementasi *real time*, sehingga menjadi aspek yang dapat ditingkatkan pada penelitian selanjutnya.

Untuk pengembangan lebih lanjut, riset mampu difokuskan dalam penambahan jumlah serta variasi dataset, peningkatan kualitas anotasi, serta optimalisasi arsitektur model guna meningkatkan akurasi deteksi dan efisiensi waktu inferensi.

4. Kesimpulan

Hasil penelitian menunjukkan bahwa penerapan metode *Single Shot Detector* (SSD) pada deteksi penyakit buah cabai rawit menunjukkan bahwa metode SSD mampu mengidentifikasi kategori buah sehat, antraknosa, dan alat buah sekaligus menampilkan lokasi gejala melalui *bounding box*. Model terbaik diperoleh dengan konfigurasi *optimizer* AdamW dan skema *learning rate* dinamis dengan nilai mAP sebesar 83,78%. Sistem yang dikembangkan telah diimplementasikan dalam aplikasi berbasis web dan dapat melakukan proses deteksi dengan waktu pemrosesan sekitar 2–3 detik per citra. Meskipun demikian, masih terdapat kesalahan prediksi pada objek dengan karakteristik visual yang serupa. Pengembangan selanjutnya dapat difokuskan pada penambahan variasi dataset, peningkatan kualitas pelatihan model, serta pengujian dengan arsitektur lain untuk meningkatkan kemampuan deteksi. Selain itu, optimalisasi sistem dan pengembangan ke platform mobile dapat dilakukan agar pemanfaatannya lebih luas di lapangan.

Daftar Pustaka:

- Aldiani, D., Dwilestari, G., Susana, H., Hamonangan, R., & Pratama, D. (2024). IMPLEMENTASI ALGORITMA CNN DALAM SISTEM ABSENSI BERBASIS PENGENALAN WAJAH. *JIP (Jurnal Informatika Polinema)*, 10(2), 197–202.
- Arrufitasari, P. N., Fikri, G. El, Handini, A. S., Sutanto, O. P., & Hidayatullah, C. S. R. (2025). STUDI KANDUNGAN CAPSAISIN CABAI RAWIT (*Capsicum frutescens* L.) AKIBAT PEMBERIAN MUTAGEN KIMIA EMS (ETHYL METHANE SULFONATE). *AGRICA*, 18(1), 139–149. <https://doi.org/10.37478/agr.v18i1.5678>

- Aulia Zahra, F., & Firmansyah, N. (2025). KLASIFIKASI CITRA JENIS IKAN AIR TAWAR DAN AIR LAUT MENGGUNAKAN ALGORITMA CNN (CONVOLUTIONAL NEURAL NETWORK). *JIP (Jurnal Informatika Polinema)*, 11(4), 495–502.
- Budi, I. S., Mariana, M., & Fauziah, A. (2023). Resistance of cayenne pepper varieties (*Capsicum frutescens*) to anthracnose disease (*Colletotrichum gloeosporioides*) isolates from swampy areas. *Jurnal Ilmiah Pertanian*, 20(3), 232–246. <https://doi.org/10.31849/jip.v20i3.16042>
- Cheng, L., Ji, Y., Li, C., Liu, X., & Fang, G. (2022). Improved SSD network for fast concealed object detection and recognition in passive terahertz security images. *Scientific Reports*, 12(1). <https://doi.org/10.1038/s41598-022-16208-0>
- Juhartini, Dwinita Arwidiyarti, & Desmiwati. (2025). Single Shot Multibox Detector (SSD) in Object Detection: A Review. *IJACI: International Journal of Advanced Computing and Informatics*, 1(2), 118–127. <https://doi.org/10.71129/ijaci.v1i2.pp118-127>
- Khan, A., Syafeeza, A. R., Amsan, A. N., Abdul Hamid, N., Abd Razak, N., & Rahaman, S. (2025). Disease Detection in Solanaceous Crops using One-Stage Detectors. *Journal of Advanced Research Design*, 134(1), 1–13. <https://doi.org/10.37934/ARD.134.1.113>
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). *SSD: Single Shot MultiBox Detector*. https://doi.org/10.1007/978-3-319-46448-0_2
- Nursalim, Y. A., Suwandi, ;, Muslim, A., Putri, ;, Wahyuni, I., Hama, J., Tumbuhan, P., & Sriwijaya, U. (2026). Evaluasi Fusarium non-patogenik terhadap Serangan Antraknosa pada Cabai Merah di Indralaya Ogan Ilir, Sumatra Selatan Evaluation of non-pathogenic Fusarium for Controlling Anthracnose on Red Chili in Indralaya, South Sumatra. *Agriprima Journal of Applied Agricultural Sciences Maret*, 10(1), 11–24. <https://doi.org/10.25047/agriprima.v9i1.782>
- Padilla, R., Passos, W. L., Dias, T. L. B., Netto, S. L., & Da Silva, E. A. B. (2021). A comparative analysis of object detection metrics with a companion open-source toolkit. *Electronics (Switzerland)*, 10(3), 1–28. <https://doi.org/10.3390/electronics10030279>
- Setiyono, B., Riv'an Arif, M., Aini, Q. Q., Soegianto, T. H., Ohanna, J., Andrean, R., Gunawan, F., & Rizkia, A. P. (2023). IDENTIFIKASI TANAMAN OBAT INDONESIA MELALUI CITRA DAUN MENGGUNAKAN METODE CONVOLUTIONAL NEURAL NETWORK (CNN). *Jurnal Teknologi Informasi dan Ilmu Komputer*, 10(2), 385–392. <https://doi.org/10.25126/jtiik.2023106809>
- Shoaib, M., Shah, B., El-Sappagh, S., Ali, A., Ullah, A., Alenezi, F., Gechev, T., Hussain, T., & Ali, F. (2023). An advanced deep learning models-based plant disease detection: A review of recent research. Dalam *Frontiers in Plant Science* (Vol. 14). Frontiers Media SA. <https://doi.org/10.3389/fpls.2023.1158933>
- Tian, J., Jin, Q., Wang, Y., Yang, J., Zhang, S., & Sun, D. (2024). Performance analysis of deep learning-based object detection algorithms on COCO benchmark: a comparative study. *Journal of Engineering and Applied Science*, 71(1). <https://doi.org/10.1186/s44147-024-00411-z>
- Upadhyay, A., Chandel, N. S., Singh, K. P., Chakraborty, S. K., Nandede, B. M., Kumar, M., Subeesh, A., Upendar, K., Salem, A., & Elbeltagi, A. (2025). Deep learning and computer vision in plant disease detection: a comprehensive review of techniques, models, and trends in precision agriculture. *Artificial Intelligence Review*, 58(3). <https://doi.org/10.1007/s10462-024-11100-x>
- Wehfany, F. Y., Timisela, N. R., & Luhukay, J. M. (2022). Analisis Faktor Yang Mempengaruhi Pendapatan Usahatani Cabai Rawit (*Capsicum frutescens* L.). *JURNAL AGRICA*, 15(2), 123–133. <https://doi.org/10.31289/agrica.v15i2.7314>
- Wulandari, A. S., Saepudin, A., Kinanti, M. P., Sudesi, Z., Saifudin, A., & Yulianti, Y. (2022). Pengujian Aplikasi Sistem Informasi Akademik Berbasis Web Menggunakan Metode Black Box Testing Equivalence Partitioning. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 5(2), 102. <https://doi.org/10.32493/jtsi.v5i2.17561>
- Xu, M., Park, J. E., Lee, J., Yang, J., & Yoon, S. (2024). Plant disease recognition datasets in the age of deep learning: challenges and opportunities. *Frontiers in Plant Science*, 15. <https://doi.org/10.3389/fpls.2024.1452551>
- Xu, M., Yoon, S., Fuentes, A., & Park, D. S. (2023). A Comprehensive Survey of Image Augmentation Techniques for Deep Learning. *Pattern Recognition*, 137. <https://doi.org/10.1016/j.patcog.2023.109347>

Halaman ini sengaja dikosongkan