

Deteksi Kerusakan Badan Kontainer di *Gate-In* PT Berlian Jasa Terminal Indonesia Menggunakan YOLOv11

1st Muhammad Najib Miftahul Huda
Teknologi Informasi
Universitas Telkom
Surabaya, Indonesia

najibmift@student.telkomuniversity.ac.id

2nd Yohanes Setiawan
Teknologi Informasi
Universitas Telkom
Surabaya, Indonesia

yohanessetiawan@telkomuniversity.ac.id

3rd Farah Zakiyah Rahmanti
Teknologi Informasi
Universitas Telkom
Surabaya, Indonesia

farahzakiyah@telkomuniversity.ac.id

Abstrak — Kerusakan pada badan kontainer menjadi tantangan utama dalam industri logistik, terutama bagi perusahaan pelabuhan seperti PT Berlian Jasa Terminal Indonesia (BJTI). Proses verifikasi kerusakan yang masih dilakukan secara manual menyebabkan potensi kesalahan, keterlambatan, dan penurunan kepercayaan pelanggan. Oleh karena itu, penelitian ini bertujuan untuk merancang, mengembangkan, dan mengevaluasi sistem deteksi otomatis kerusakan kontainer. Penelitian ini melakukan studi perbandingan antara YOLOv11 dengan model pembanding lainnya (YOLOv8, YOLOv9, dan YOLOv10) untuk menemukan arsitektur terbaik, yang kemudian diintegrasikan dengan sistem web sebagai antarmuka pemantauan. Model dilatih menggunakan dataset gabungan dari CCTV pelabuhan dan sumber publik. Hasil pengujian menunjukkan bahwa YOLOv11 menjadi model paling unggul, mencapai performa akurasi tertinggi dengan mAP@0.5 sebesar 53.44%, Precision 66.97%, dan Recall 50.69%. Dari sisi kecepatan, sistem yang diuji pada CPU menunjukkan waktu inferensi rata-rata 123.90 ms per gambar, yang juga merupakan kinerja tercepat dibandingkan model pembanding lainnya, membuktikan efisiensinya untuk implementasi operasional. Hasil deteksi secara otomatis diunggah ke server dan ditampilkan pada antarmuka web yang dibangun menggunakan framework CodeIgniter 4. Evaluasi menunjukkan bahwa integrasi antara model deteksi dan sistem web berjalan dengan baik dan stabil. Sistem ini diharapkan dapat menjadi solusi aplikatif berbasis computer vision untuk meningkatkan efisiensi dan akurasi dalam pengawasan kontainer di lingkungan pelabuhan.

Kata kunci — Deteksi kerusakan kontainer, YOLOv11, Pelabuhan

I. PENDAHULUAN

Kontainer merupakan elemen vital dalam logistik global, berfungsi melindungi barang selama pengiriman dari kerusakan akibat faktor eksternal [1]. Meskipun telah ada standar internasional untuk standardisasi kontainer, penerapannya sering kali menghadapi kendala, terutama di negara berkembang, akibat pengawasan yang lemah dan infrastruktur yang tidak memadai [2]. Akibatnya, standar sering dianggap hanya sebagai syarat administratif tanpa implementasi serius, yang dapat menimbulkan kerugian [3].

PT Berlian Jasa Terminal Indonesia (BJTI), sebagai perusahaan peti kemas domestik, sering menghadapi keluhan pelanggan terkait kerusakan kontainer seperti penyok dan lubang. Proses verifikasi kerusakan yang ada saat ini masih manual, mengandalkan rekaman CCTV yang memakan waktu dan rentan kesalahan. Kesalahan dalam menentukan lokasi dan waktu kerusakan tidak hanya merugikan perusahaan tetapi juga dapat menurunkan kepercayaan pelanggan [4].

Oleh karena itu, diperlukan sebuah solusi untuk mengotomatisasi proses deteksi kerusakan secara akurat dan efisien. Penelitian ini mengusulkan pengembangan sistem deteksi kerusakan badan kontainer menggunakan YOLOv11. Model ini dipilih karena arsitekturnya yang canggih terbukti unggul dalam hal akurasi, kecepatan, dan efisiensi komputasi untuk deteksi objek secara *real-time* [5] [6] [7].

Tujuan dari penelitian ini adalah merancang, mengembangkan, mengevaluasi, dan mengintegrasikan sistem deteksi kerusakan badan kontainer berbasis YOLOv11 yang dapat bekerja secara otomatis untuk mendukung efisiensi operasional dan memberikan bukti kondisi kontainer yang valid di *gate-in* PT Berlian Jasa Terminal Indonesia.

II. KAJIAN TEORI

Bagian dasar teori memuat penjelasan mengenai teori-teori yang berhubungan dengan variabel-variabel penelitian. Teori-teori yang dibahas mencakup beberapa topik, diantaranya mengenai Studi terdahulu, YOLOv11, Kontainer, *Gate-in*, dan Standar kerusakan yang digunakan.

A. Konteks dan Studi Terdahulu dalam Deteksi Kerusakan

Penerapan *computer vision* untuk inspeksi visual otomatis terbukti efektif dalam meningkatkan efisiensi dan akurasi, khususnya di sektor logistik dan infrastruktur. Berbagai studi sebelumnya menunjukkan keberhasilan algoritma YOLO dalam mendeteksi kerusakan objek. Hidayah dan Kristian [8] menggunakan YOLOv7 untuk klasifikasi kerusakan kontainer, sementara Setyaningsih dan Edy [9] menunjukkan keunggulan YOLOv4 dibanding Mask R-CNN pada deteksi kerusakan karung. Kim dan Lee [10] menggabungkan YOLOv5 dan StyleGAN3 untuk deteksi paket rusak, sedangkan Pan dkk [11] mengembangkan YOLOX-s untuk

inspeksi cacat permukaan baja. Di bidang infrastruktur, Sasmito dkk. memakai YOLOv4-Tiny untuk mendeteksi kerusakan jalan. Temuan ini menegaskan bahwa algoritma YOLO merupakan solusi andal dan serbaguna dalam tugas deteksi kerusakan.

B. YOLOv11

YOLO adalah keluarga algoritma deteksi objek satu tahap yang terkenal karena kecepatan dan efisiensinya. Tidak seperti metode dua tahap, YOLO memproses seluruh gambar dalam satu kali proses untuk menghasilkan prediksi *bounding box* dan klasifikasi objek secara simultan [12]. Penelitian ini menggunakan YOLOv11, generasi terbaru yang dirancang untuk memberikan keseimbangan optimal antara presisi dan kecepatan [5]. Arsitektur YOLOv11 terdiri dari tiga komponen utama: *Backbone* untuk ekstraksi fitur, *Neck* untuk agregasi fitur multiskala, dan *Head* untuk prediksi akhir. Peningkatan pada YOLOv11, seperti modul C3K2, SPPF, dan C2PSA, meningkatkan kemampuan ekstraksi fitur halus dan pemahaman konteks, menjadikannya sangat efektif untuk mendeteksi objek dengan berbagai ukuran dan kompleksitas visual [7].

C. Kontainer (Peti Kemas)

Kontainer atau peti kemas adalah wadah berbentuk empat persegi panjang yang dirancang khusus dengan ukuran tertentu, terbuat dari material seperti besi atau aluminium. Peti kemas dilengkapi dengan pintu di salah satu sisinya dan dirancang untuk digunakan secara berulang kali. Fungsi utama peti kemas adalah sebagai tempat untuk menyimpan dan mengangkut muatan yang ada di dalamnya. Selain itu, peti kemas telah ditetapkan berdasarkan standar internasional yang dikeluarkan oleh International Organization for Standardization (ISO) [13].

D. Gate-In Terminal Peti Kemas

Gate merupakan titik keluar masuk bagi truk yang melakukan aktivitas operasional di sekitar terminal petikemas [14]. Sedangkan terminal peti kemas adalah fasilitas utama yang disediakan pelabuhan untuk mendukung kelancaran proses logistik. Terminal ini dirancang dengan infrastruktur seperti dermaga, lapangan penumpukan, dan peralatan bongkar muat modern untuk memastikan efisiensi proses logistik [15]. Berdasarkan pengertian di atas, *gate-in* terminal peti kemas merupakan titik masuk utama bagi truk pembawa peti kemas, yang dirancang untuk mendukung kelancaran proses logistik di pelabuhan. *Gate-in* berfungsi sebagai lokasi pemeriksaan administrasi dan fisik peti kemas sebelum memasuki terminal, serta merupakan elemen strategis dari infrastruktur terminal peti kemas.

E. Definisi dan Standar Kerusakan Kontainer

Untuk memastikan objektivitas dalam pendeteksian, penelitian ini mendefinisikan "kerusakan" berdasarkan keluhan yang paling sering diterima di PT Berlian Jasa Terminal Indonesia, yaitu penyok (*dent*) dan lubang (*hole*). Definisi ini diperkuat dengan mengacu pada pedoman standar industri pelayaran, *Unified Container Inspection and Repair Criteria* (UCIRC). Menurut UCIRC, kerusakan penyok pada badan kontainer dianggap signifikan dan memerlukan tindakan jika kedalamannya melebihi 50 mm. Sementara itu, lubang didefinisikan sebagai celah yang menembus badan

kontainer, yang berpotensi menyebabkan kontaminasi atau kerusakan muatan di dalamnya sehingga harus segera diperbaiki. Dengan mengacu pada standar ini, penelitian memastikan bahwa objek yang dideteksi oleh model YOLOv11 memiliki relevansi operasional dan sejalan dengan praktik industri.

F. Metrik Evaluasi Kinerja

Untuk mengevaluasi performa model deteksi secara kuantitatif, penelitian ini menggunakan beberapa metrik standar. Metrik utama yang digunakan adalah *mean Average Precision* (mAP), yang mengukur rata-rata presisi di berbagai nilai *Intersection over Union* (IoU) dengan rumus

$$mAP = \frac{1}{n} \sum_{i=3}^n AP_i, \quad (1)$$

Kinerja ini juga diukur oleh *Precision*, yaitu proporsi deteksi positif yang benar dari total prediksi yang dibuat, yang dirumuskan sebagai

$$Precision = \frac{TP}{TP+FP}, \quad (2)$$

Selain itu, digunakan *Recall* untuk mengukur kemampuan model dalam menemukan semua objek kerusakan yang sebenarnya ada, dengan rumus

$$Recall = \frac{TP}{TP+FN}, \quad (3)$$

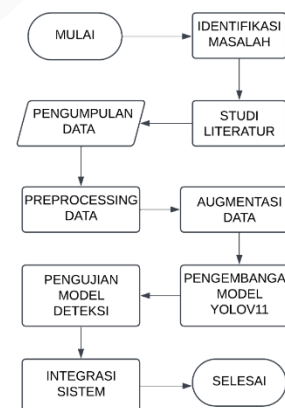
Keseimbangan antara keduanya dinilai menggunakan *F1 Score*, yang merupakan rata-rata harmonis dari *Precision* dan *Recall*, dihitung sebagai

$$F1 = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}, \quad (4)$$

Dalam rumus-rumus tersebut, TP adalah *True Positives* (prediksi benar), FP adalah *False Positives* (prediksi salah), dan FN adalah *False Negatives* (objek yang terlewat).

III. METODE

Penelitian ini mengikuti alur sistematis yang dimulai dari identifikasi masalah, studi literatur, pengumpulan data, *preprocessing*, augmentasi data, pengembangan model, pengujian, hingga integrasi sistem.



Gambar 1 Sistematis penyelesaian masalah

Tahap pertama adalah persiapan data. Data dikumpulkan dari dua sumber utama: data primer berupa gambar dari CCTV dan dokumentasi langsung di PT Berlian Jasa Terminal Indonesia, serta data sekunder dari repositori publik Roboflow untuk memperkaya variasi. Mengingat jumlah data primer yang menunjukkan kerusakan sangat terbatas (hanya 50 citra), strategi penggabungan dataset diterapkan untuk mencegah *overfitting* dan meningkatkan kemampuan generalisasi model. Seluruh data mentah kemudian melalui tahap *preprocessing* yang meliputi: (1) Pelabelan manual, di mana kerusakan seperti penyok dan lubang ditandai dalam satu kelas "*damage*"; (2) Perubahan ukuran gambar menjadi 640x640 piksel agar sesuai dengan arsitektur model; dan (3) Pembersihan data untuk menghilangkan gambar yang tidak valid. Untuk memperkaya variasi data latih, diterapkan teknik augmentasi data seperti *Horizontal Flip*, *Rotation* ($\pm 15^\circ$), dan perubahan *Brightness* ($\pm 25\%$). Dataset akhir kemudian dibagi dengan rasio 70% untuk data latih, 20% untuk validasi, dan 10% untuk pengujian.

Tabel 1 Parameter pelatihan model

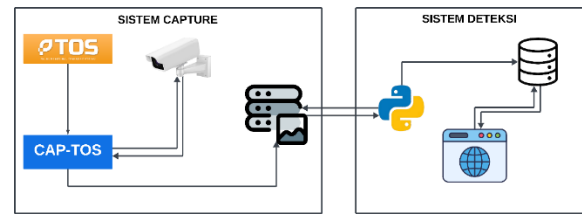
Percobaan	Training Hyperparameter			
	Epoch	Optimizer	lr0	Momentum
1	300	AdamW	0.0001	-
2	500	AdamW	0.0001	-
3	300	SGD	0.01	0.937
4	500	SGD	0.01	0.937

Tahap selanjutnya adalah pengembangan dan pelatihan model. Penelitian ini menggunakan arsitektur YOLOv11n karena keseimbangan antara kecepatan dan akurasi. Pelatihan model dilakukan di lingkungan komputasi awan Kaggle Notebook yang menyediakan akses ke GPU untuk mempercepat proses. Sebuah desain eksperimen dirancang untuk menemukan konfigurasi *hyperparameter* terbaik dengan menguji empat skenario yang membandingkan dua jenis *optimizer* (AdamW dan SGD) dan dua durasi pelatihan (300 dan 500 *epoch*).

Evaluasi model dilakukan dalam dua tahap. Pertama, evaluasi internal untuk menentukan konfigurasi terbaik dari empat skenario pelatihan YOLOv11. Kedua, studi perbandingan antara model YOLOv11 terbaik dengan tiga model pembanding lainnya (YOLOv8, YOLOv9, dan YOLOv10) untuk memvalidasi keunggulannya. Seluruh model pembanding dilatih dan diuji dalam kondisi yang identik untuk memastikan perbandingan yang adil. Kinerja model diukur menggunakan metrik standar deteksi objek, yaitu *mean Average Precision* (mAP@0.5), *Precision*, *Recall*, dan Waktu Inferensi.

Tahap terakhir adalah integrasi sistem, di mana model deteksi dihubungkan dengan alur kerja operasional. Desain sistem memisahkan antara Sistem *Capture* (infrastruktur PT BJTI yang sudah ada untuk menangkap gambar) dan Sistem Deteksi (aplikasi Python yang dikembangkan dalam penelitian ini). Sistem Deteksi secara otomatis memonitor folder di server, memproses gambar baru yang masuk menggunakan model YOLOv11, dan menyimpan hasilnya ke dalam *database* untuk ditampilkan pada antarmuka web.

A. Desain Sistem

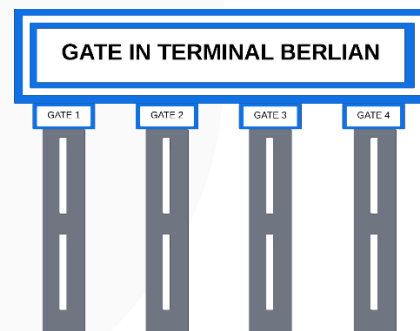


Gambar 2 Desain sistem

Arsitektur sistem yang dirancang memisahkan alur kerja menjadi dua subsistem utama yang saling berinteraksi: Sistem *Capture* dan Sistem Deteksi. Sistem *Capture* merupakan infrastruktur yang sudah ada di PT Berlian Jasa Terminal Indonesia, di mana aplikasi *registrasi* (P-TOS) memicu aplikasi lain (CAP-TOS) untuk menangkap serangkaian gambar kontainer dari kamera CCTV dan menyimpannya di server penyimpanan gambar. Sementara itu, Sistem Deteksi adalah sistem inti yang dikembangkan dalam penelitian ini. Sistem ini berupa aplikasi Python yang secara terus-menerus memonitor server tersebut. Ketika gambar baru tersedia, aplikasi akan menganalisisnya menggunakan model YOLOv11 yang telah dilatih, lalu menyimpan hasilnya ke dalam basis data untuk ditampilkan pada antarmuka web.

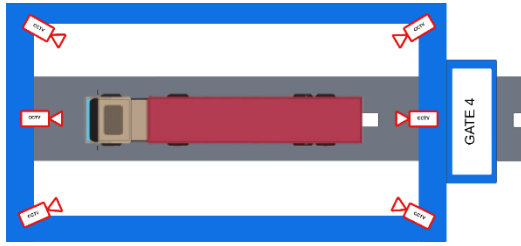
B. Desain Peletakan Kamera

Sistem deteksi kerusakan ini dirancang untuk bekerja dengan memanfaatkan infrastruktur kamera CCTV yang sudah terpasang di seluruh jalur *gate-in* PT Berlian Jasa Terminal Indonesia. Terminal tersebut memiliki beberapa gerbang masuk (Gate 1 hingga Gate 4), di mana semuanya memiliki konfigurasi kamera yang seragam untuk memastikan konsistensi pengambilan data.



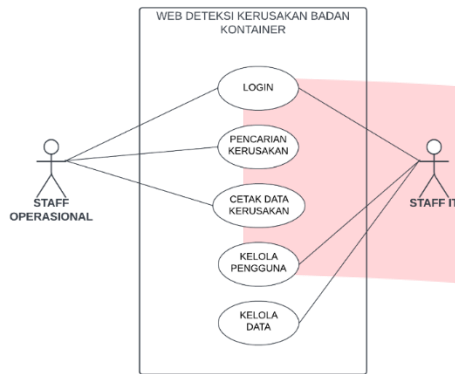
Gambar 3 Denah *gate-in* terminal berlian

Untuk memastikan cakupan deteksi yang komprehensif, setiap gerbang dilengkapi dengan beberapa unit kamera yang diposisikan secara strategis. Tata letak ini dirancang untuk menangkap citra kontainer secara menyeluruh dari enam sudut pandang berbeda saat truk melintas, yaitu sisi depan-kanan, depan-kiri, samping-kanan, samping-kiri, belakang-kanan, dan belakang-kiri. Proses pengambilan gambar ini terintegrasi dengan sistem operasional di gerbang; kamera secara otomatis akan menangkap serangkaian gambar ketika pengemudi melakukan proses identifikasi. Dengan memanfaatkan beberapa sudut pandang ini, sistem dapat melakukan pemeriksaan kerusakan pada lebih dari satu sisi kontainer, sehingga meningkatkan akurasi dan cakupan deteksi secara signifikan.



Gambar 4 Desain peletakan kamera

C. Use Case Diagram Website



Gambar 5 Use case diagram

Untuk memodelkan fungsionalitas sistem web dan interaksi antara pengguna dan sistem, sebuah *use case diagram* dirancang. Diagram ini mengidentifikasi dua peran pengguna (aktor) utama yang berinteraksi dengan sistem, yaitu Staff Operasional dan Staff IT. Berdasarkan rancangan, kedua aktor dapat mengakses sistem melalui fungsi Login yang aman. Setelah masuk, hak akses dibedakan berdasarkan peran. Staff Operasional memiliki akses ke fungsi utama untuk operasional harian, yang meliputi pencarian Kerusakan untuk menelusuri riwayat data berdasarkan nomor kontainer, serta Cetak Data Kerusakan untuk menghasilkan laporan fisik. Sementara itu, Staff IT memegang peran administratif dengan hak akses eksklusif. Fungsi mereka mencakup Kelola Pengguna untuk manajemen akun, dan Kelola Data untuk pemeliharaan data hasil deteksi di dalam basis data sistem. Pemisahan peran ini memastikan bahwa tugas operasional terpisah dari tugas administratif, sehingga menjaga integritas dan keamanan data.

IV. HASIL DAN PEMBAHASAN

A. Hasil Eksperimen dan Pemilihan Model Terbaik

Untuk menemukan konfigurasi model YOLOv11 yang paling optimal, dilakukan empat skenario eksperimen dengan memvariasikan *optimizer* dan jumlah *epoch*. Hasil pengujian performa dari keempat skenario pada data uji disajikan secara tabulasi pada Tabel 2, yang kemudian dianalisis secara deskriptif.

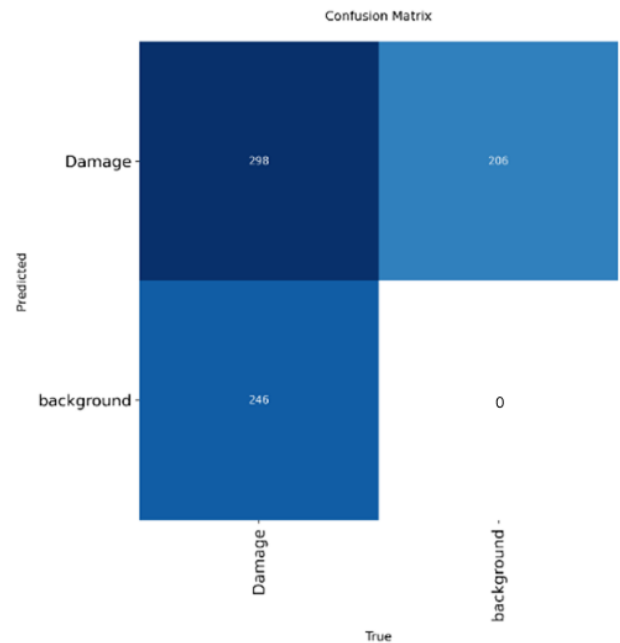
Tabel 2 Hasil Pengujian Empat Skenario Model YOLOv11

No	Optimizer	Epoch	Map @0.5	Map @0.5-95	Precision	Recall
1	AdamW	300	0.5344	0.2993	0.6697	0.5069
2	AdamW	500	0.5120	0.3112	0.7150	0.4504
3	SGD	300	0.5060	0.3054	0.6450	0.4706
4	SGD	500	0.5045	0.3087	0.7113	0.4619

Dari data pada Tabel 1, dapat ditarik kesimpulan bahwa Eksperimen 1 yang menggunakan *optimizer* AdamW dan 300 *epoch* memberikan performa terbaik. Model ini tidak hanya mencapai nilai mAP@0.5 tertinggi sebesar 53.44%, tetapi juga menunjukkan keseimbangan terbaik antara *Precision* (66.97%) dan *Recall* (50.69%). Temuan menarik dari data ini adalah bahwa penambahan durasi pelatihan menjadi 500 *epoch* justru menurunkan performa mAP@0.5, yang mengindikasikan adanya potensi *overfitting*. Berdasarkan analisis ini, model dari Eksperimen 1 dipilih sebagai model final untuk analisis lebih lanjut.

B. Analisis Kinerja dan Kesalahan Model

Untuk menafsirkan data kinerja dari model terbaik, dilakukan analisis mendalam terhadap *confusion matrix* untuk memahami karakteristik kesalahan yang dibuat.



Gambar 5 Confusion matrix model terbaik

Berdasarkan analisis *confusion matrix* tersebut, kinerja model dapat diuraikan secara rinci. Model berhasil mendeteksi 298 kasus kerusakan dengan benar (*True Positives*). Namun, model ini juga membuat kesalahan berupa 206 "alarm palsu" (*False Positives*), di mana area normal salah diklasifikasikan sebagai kerusakan. Jenis kesalahan yang paling signifikan adalah *False Negatives*, di mana model melewatkan 246 kasus kerusakan yang seharusnya terdeteksi. Interpretasi dari data ini menunjukkan bahwa kelemahan terbesar model terletak pada tingginya jumlah *False Negatives*. Hal ini berarti, meskipun deteksi yang dilaporkan cenderung akurat (tercermin dari nilai *Precision* yang baik), kemampuan model untuk mengidentifikasi seluruh kerusakan yang ada (*Recall*) masih terbatas.

C. Studi Perbandingan Performa

Untuk memvalidasi pilihan arsitektur YOLOv11, dilakukan studi perbandingan performa dengan tiga model lainnya dalam kondisi pengujian yang identik. Data hasil perbandingan akurasi dan kecepatan disajikan pada Tabel 3 dan Tabel 4.

Tabel 3 Perbandingan performa akurasi antar model

Model	Map@0.5	Map@0.5-95	Precision	Recall
YOLOv8	0.5052	0.2846	0.6681	0.4697
YOLOv9	0.4955	0.2752	0.6791	0.4587
YOLOv10	0.4926	0.2936	0.5880	0.4954
YOLOv11	0.5344	0.2993	0.6697	0.5069

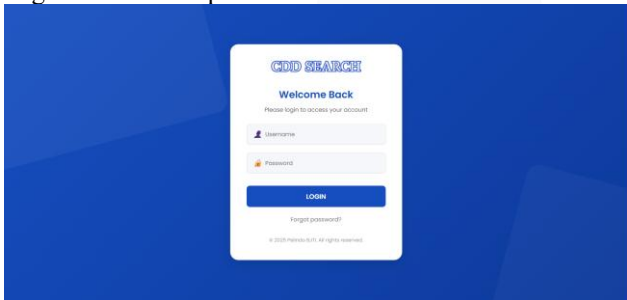
Tabel 4 Perbandingan Performa Kecepatan Antar Model

Model	Total Waktu Proses per Batch (6 gambar)	Waktu Inferensi (ms/gambar)
YOLOv8	34.49 detik	176.97 ms
YOLOv9	35.05 detik	184.66 ms
YOLOv10	29.77 detik	138.22 ms
YOLOv11	26.78 detik	123.90 ms

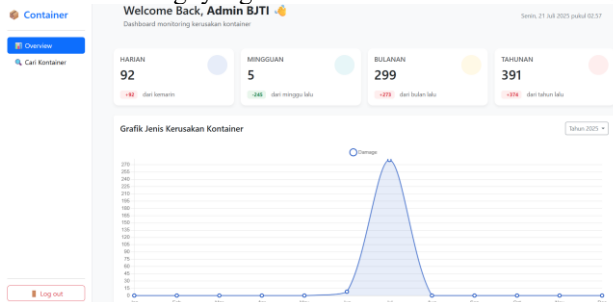
dari tabel di atas menunjukkan bahwa YOLOv11 unggul secara komprehensif. YOLOv11 tidak hanya mencapai akurasi keseluruhan (mAP@0.5) dan kelengkapan deteksi (*Recall*) tertinggi, tetapi juga merupakan yang tercepat, baik dari waktu proses total maupun waktu inferensi per gambar. Kombinasi keunggulan ini menjadikan YOLOv11 sebagai pilihan arsitektur yang paling sesuai untuk studi kasus ini.

D. Hasil Uji Fungsionalitas Antarmuka Web

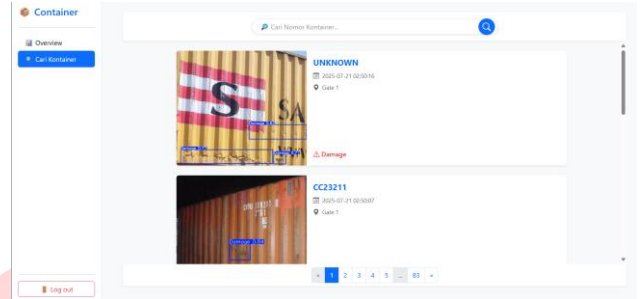
Sebagai bagian dari integrasi sistem, sebuah antarmuka web dikembangkan menggunakan *framework* CodeIgniter 4 untuk menyajikan hasil deteksi dalam format yang informatif dan mudah digunakan oleh pengguna. Evaluasi fungsionalitas dilakukan pada tiga halaman utama untuk memastikan sistem berjalan dengan baik, stabil, dan sesuai dengan kebutuhan operasional.

Gambar 6 Hasil halaman *login*

Fitur pertama yang diuji adalah halaman *login*, yang berfungsi untuk membatasi akses hanya kepada pengguna berwenang. Pengujian mengonfirmasi bahwa fitur autentikasi ini berhasil dijalankan dengan baik, di mana pengguna dengan kredensial valid dapat masuk ke sistem, sementara akses ditolak bagi yang tidak valid.

Gambar 7 Hasil halaman *dashboard*

Selanjutnya, halaman *dashboard* dievaluasi. Halaman ini berfungsi sebagai pusat tampilan statistik, menyajikan data ringkasan kerusakan kontainer dalam kategori harian, mingguan, bulanan, dan tahunan, serta menampilkan grafik tren kerusakan. Hasil evaluasi menunjukkan bahwa seluruh komponen pada *dashboard* tampil dengan baik dan menampilkan data statistik secara akurat dan konsisten dari basis data.



Gambar 6 Hasil halaman pencarian kontainer

Fungsi inti dari sistem, yaitu halaman pencarian, memungkinkan pengguna untuk mencari riwayat kerusakan berdasarkan nomor kontainer. Hasil pencarian menampilkan informasi detail seperti gambar hasil deteksi, jenis kerusakan, lokasi, dan waktu kejadian. Pengujian pada fitur ini menunjukkan bahwa sistem dapat menampilkan hasil pencarian dengan cepat dan akurat. Secara keseluruhan, evaluasi membuktikan bahwa sistem web berjalan stabil, responsif, dan berfungsi sesuai harapan, yang menandakan keberhasilan integrasi *end-to-end* dari model deteksi hingga penyajian hasil kepada pengguna.

V. KESIMPULAN

Penelitian ini telah berhasil merancang dan mengembangkan sebuah sistem fungsional *end-to-end* untuk mendeteksi kerusakan pada badan kontainer secara otomatis, yang terintegrasi dari proses akuisisi gambar hingga visualisasi hasil melalui antarmuka web. Melalui serangkaian eksperimen, arsitektur YOLOv11 dengan konfigurasi *optimizer* AdamW selama 300 *epoch* terbukti menjadi yang paling unggul, mencapai performa mAP@0.5 sebesar 53.44%, *Precision* 66.97%, dan *Recall* 50.69%. Kinerja ini tidak hanya melampaui konfigurasi lain tetapi juga terbukti lebih superior dibandingkan model pembanding seperti YOLOv8, YOLOv9, dan YOLOv10 dalam studi perbandingan yang dilakukan. Analisis lebih lanjut menunjukkan karakteristik model dengan presisi yang baik namun *recall* yang moderat, mengindikasikan bahwa sistem ini efektif sebagai alat bantu pendukung keputusan yang andal bagi operator, meskipun belum dapat beroperasi sepenuhnya secara otonom. Dari sisi kecepatan, sistem yang diuji pada CPU menunjukkan kinerja yang memadai untuk alur kerja operasional dengan waktu inferensi 123.90 ms per gambar, dan seluruh fungsionalitas web terintegrasi dengan stabil dan responsif.

REFERENSI

- [1] M. Haris, A. Abdillah, and I. Kirono, "Penggunaan Container Sebagai Pengganti Terpal Untuk Angkut Pupuk Pt Gresik Cipta Sejahtera," *AkMen (Akuntansi dan Manajemen) JURNAL ILMIAH*, vol. 20, no. 3, pp. 250–258, 2023, doi: <https://doi.org/10.37476/akmen.v20i3.4371>.
- [2] D. Agus Susanto, M. Haekal Habiebie, and B. Basuki, "Peran Penerapan Standardisasi Dan Penilaian Kesesuaian Dalam Peningkatan Kinerja Perusahaan Dalam Perdagangan Internasional," *Jurnal Ekonomi & Kebijakan Publik*, vol. 14, no. 2, pp. 93–108, 2023, doi: 10.22212/jekp.v14i2.2129.
- [3] D. Willar, D. Debora, G. Pangemanan, and P. Korespondensi, "Hambatan Signifikan Implementasi Sistem Manajemen Mutu Pelaksana Konstruksi," *TEKNIK*, vol. 41, no. 2, pp. 100–110, 2020, doi: 10.14710/teknik.v41n2.27252.
- [4] Juanda, April Gunawan Malau, and Agung Kwartama, "Pengaruh Kualitas Layanan Dan Pengembangan Pelabuhan Terhadap Kepercayaan Yang Berimplikasi Pada Loyalitas Pengguna Jasa Bongkar Muat Di Pt Jakarta International Container Terminal (JICT)," *Journal of Economic, Bussines and Accounting (COSTING)*, vol. 7, no. 5, 2024, doi: <https://doi.org/10.31539/costing.v7i5.12395>.
- [5] A. Sharma, V. Kumar, and L. Longchamps, "Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN models for detection of multiple weed species," *Smart Agricultural Technology*, vol. 9, Dec. 2024, doi: 10.1016/j.atech.2024.100648.
- [6] R. Sapkota, Z. Meng, M. Churuvija, X. Du, Z. Ma, and M. Karkee, "Comprehensive Performance Evaluation of YOLOv12, YOLO11, YOLOv10, YOLOv9 and YOLOv8 on Detecting and Counting Fruitlet in Complex Orchard Environments," *arXiv preprint*, Feb. 2025, [Online]. Available: <http://arxiv.org/abs/2407.12040>
- [7] L. H. He, Y. Z. Zhou, L. Liu, W. Cao, and J. H. Ma, "Research on object detection and recognition in remote sensing images based on YOLOv11," *Sci Rep*, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-96314-x.
- [8] F. Hidayah and Y. Kristian, "Identifikasi Kerusakan Badan Kontainer Pada Waktu Pengiriman Berdasarkan Citra CCTV Memanfaatkan YOLO dan Deep Transfer Learning," *Teknika*, vol. 13, no. 1, pp. 10–17, Jan. 2024, doi: 10.34148/teknika.v13i1.718.
- [9] E. R. Setyaningsih and M. S. Edy, "YOLOv4 dan Mask R-CNN Untuk Deteksi Kerusakan Pada Karung Komoditi," *Teknika*, vol. 11, no. 1, pp. 45–52, Mar. 2022, doi: 10.34148/teknika.v11i1.419.
- [10] S. Kim and S. D. Lee, "YOLO-Based Damage Detection with StyleGAN3 Data Augmentation for Parcel Information-Recognition System," *Computers, Materials and Continua*, vol. 80, no. 1, pp. 195–215, 2024, doi: 10.32604/cmc.2024.052070.
- [11] J. Z. Pan, C. H. Yang, L. Wu, X. Huang, and S. Qiu, "One improved YOLOX-s algorithm for lightweight section-steel surface defect detection," *Advances in Mechanical Engineering*, vol. 16, no. 8, Aug. 2024, doi: 10.1177/16878132241266456.
- [12] C.-Y. Wang and H.-Y. M. Liao, "YOLOv1 to YOLOv10: The fastest and most accurate real-time object detection systems," *APSIPA Trans Signal Inf Process*, pp. 1–38, Aug. 2024, doi: 10.1561/116.20240058.
- [13] R. Dormawaty, R. S. Wulandari, and M. E. Tumeko, "Pengaruh Penanganan Repair Container Guna Memenuhi Kebutuhan Ekspor di PT.Evergreen Shipping Agency Indonesia," *Meteor STIP Marunda*, vol. 14, no. 2, pp. 158–169, Dec. 2021, doi: 10.36101/msm.v14i2.204.
- [14] R. Septianarta Putra and I. Dewi Ratih, "Analysis of Import Gate Out Service Using Control Chart in PT Terminal Petikemas Surabaya," *IPTEK The Journal of Engineering*, vol. 7, no. 1, pp. 23–30, 2021, doi: <http://dx.doi.org/10.12962/j23378557.v7i1.a7460>.
- [15] Prayono, Riyanto, Kundori, and I. N. Gede Muliawan, "Optimalisasi Relokasi Petikemas Di Pelabuhan Tanjung Perak Dalam Menunjang Kegiatan Bongkar Muat Pada Pt.Abc," *Jurnal Sains Teknologi Transportasi Maritim*, vol. 2, no. 2, pp. 33–40, Nov. 2020, doi: <https://doi.org/10.51578/j.sitektransmar.v2i2.26>.