

Pengujian Fungsional dan Waktu Respon Inferensi Model Sistem Pelaporan Kerusakan Jalan Berbasis Android

^{1*}Ilham Khefi Ramadhanu, ²Wahyu Cahyo Utomo, ³Danang Wahyu Widodo

¹⁻³ Teknik Informatika, Universitas Nusantara PGRI Kediri

E-mail: ¹ilhamkhefiramadhanu@gmail.com, ²wahyu.utomo@unpkdr.ac.id,

³danangwahyuwidodo@unpkediri.ac.id

Penulis Korespondens : Ilham Khefi Ramadhanu

Abstrak— Kerusakan jalan merupakan permasalahan infrastruktur yang memerlukan penanganan cepat dan akurat. Penelitian ini menguji sistem pelaporan kerusakan jalan berbasis Android yang mengintegrasikan model YOLOv12 dan ResNet-50 melalui Django REST API. Pengujian dilakukan dalam dua tahap. Pertama, pengujian fungsional menggunakan metode *Black Box Testing* terhadap 26 skenario yang mencakup komponen register, lapor kerusakan, ambil lokasi GPS, daftar laporan, detail laporan, dan pemetaan, seluruhnya menunjukkan keberhasilan dengan tingkat keberhasilan 100%. Kedua, pengujian waktu respon inferensi terhadap 10 sampel gambar menghasilkan rata-rata unggah gambar 123,62 ms, inferensi ResNet-50 1.892,38 ms, inferensi YOLOv12 859,73 ms, dan total waktu respon 3.150,66 ms. Inferensi ResNet-50 menjadi kontributor terbesar dengan porsi sekitar 60% dari total waktu respon, mengindikasikan bahwa *bottleneck* utama sistem terletak pada komputasi model klasifikasi, bukan pada transmisi jaringan.

Kata Kunci—Android, Black Box Testing, Kerusakan Jalan, ResNet-50, YOLOv12

Abstract— Road damage is an infrastructure problem that requires fast and accurate handling. This study evaluates an Android-based road damage reporting system integrating YOLOv12 and ResNet-50 models through a Django REST API. Testing was conducted in two stages. First, Black Box Testing across 26 functional scenarios covering register, damage report, GPS location, report list, report detail, and mapping components — all yielding successful results with a 100% pass rate. Second, inference response time testing on 10 image samples produced an average image upload time of 123.62 ms, ResNet-50 inference of 1,892.38 ms, YOLOv12 inference of 859.73 ms, and a total response time of 3,150.66 ms. ResNet-50 inference was the largest contributor at approximately 60% of total response time, indicating that the primary bottleneck lies in classification model computation rather than network transmission.

Keywords— Android, Black Box Testing, Road Damage, ResNet-50, YOLOv12

This is an open access article under the CC BY-SA License.



I. PENDAHULUAN

Kerusakan jalan merupakan permasalahan infrastruktur yang berdampak langsung pada keselamatan pengguna jalan dan memerlukan penanganan yang cepat serta terkoordinasi [1]. Perkembangan teknologi kecerdasan buatan, khususnya di bidang *computer vision*, telah memungkinkan otomatisasi identifikasi kerusakan jalan melalui model deteksi objek [2], seperti

YOLOv12 dan model klasifikasi seperti ResNet-50 [3], [4] yang diintegrasikan ke dalam aplikasi berbasis Android melalui Django REST API [5].

Beberapa penelitian sebelumnya telah mengembangkan sistem deteksi dan pelaporan kerusakan jalan berbasis *deep learning*. Surahmanto et al. [6] mengembangkan sistem deteksi lubang berbasis YOLOv5 dengan akurasi 93%, Maeda et al. [7] mengusulkan metode deteksi menggunakan *smartphone* dan *deep learning*, serta Zhang et al. [8] menggunakan CNN untuk klasifikasi jenis kerusakan dengan akurasi 91%. Namun, sebagian besar penelitian tersebut berfokus pada akurasi model dan belum menyertakan pengujian fungsionalitas sistem secara menyeluruh maupun pengukuran performa latensi dari sisi integrasi server ke perangkat mobile.

Kesenjangan ini menjadi motivasi utama penelitian ini. Sistem yang akurat secara model belum tentu andal secara fungsional dalam kondisi operasional nyata. Oleh karena itu, diperlukan dua pendekatan pengujian yang saling melengkapi: pengujian fungsional menggunakan metode *Black Box Testing* untuk memverifikasi seluruh fitur aplikasi Android berjalan sesuai rancangan, serta pengujian latensi inferensi untuk mengukur kecepatan respons dari server Django ke perangkat Android sebagai indikator kelayakan sistem dalam penggunaan *real-time*.

Penelitian ini mengembangkan sistem pelaporan kerusakan jalan berbasis Android yang mengintegrasikan model YOLOv12 untuk deteksi objek dan ResNet-50 untuk klasifikasi jenis kerusakan. Sistem dibangun menggunakan arsitektur *client-server* dengan Django REST API sebagai jembatan antara aplikasi Android dan model kecerdasan buatan yang berjalan di sisi server. Sistem mendeteksi empat jenis kerusakan jalan, yaitu lubang, retak melintang, retak memanjang, dan retak buaya.

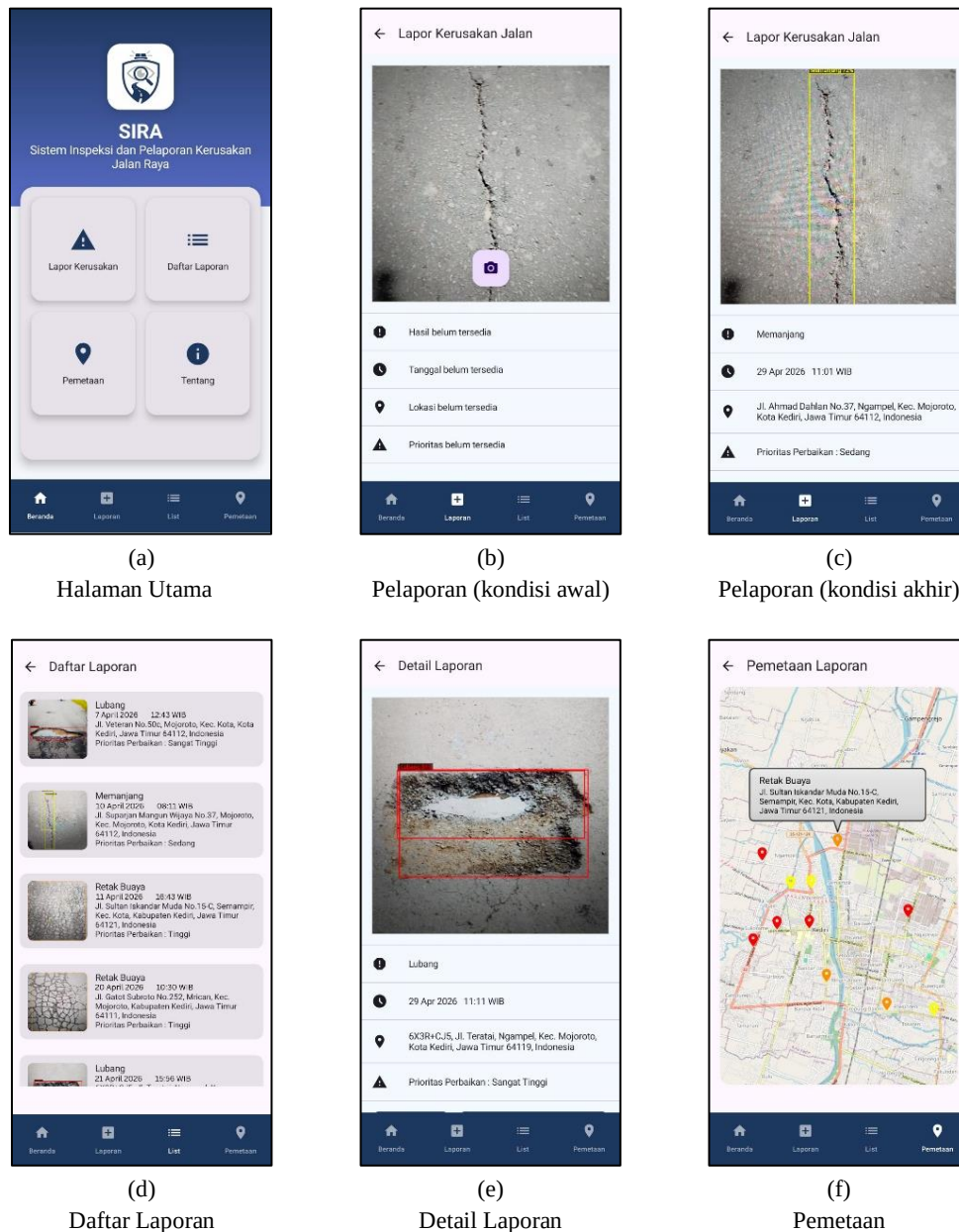
Aspek yang menjadi fokus dalam penelitian ini adalah pengujian fungsional sistem menggunakan metode *Black Box Testing* yang bersifat preventif dan korektif [9]. Pengujian dilakukan secara terstruktur terhadap dua lembar kerja utama sistem, yaitu lembar kerja pelaporan yang berjalan di sisi Android dan lembar kerja sistem cerdas yang berjalan di sisi server. Tujuan pengujian adalah memverifikasi bahwa seluruh komponen sistem berfungsi sesuai rancangan, mampu menangani input tidak valid, dan memberikan respons yang tepat dalam kondisi tidak terduga.

II. METODE

Penelitian ini menggunakan metode *Black Box Testing* untuk menguji fungsionalitas sistem pelaporan kerusakan jalan berbasis Android. *Black Box Testing* adalah metode pengujian yang berfokus pada input dan output sistem tanpa memperhatikan implementasi kode internal [10]. Pengujian dirancang bersifat preventif untuk memastikan sistem mampu mencegah dan menangani input tidak valid, serta bersifat korektif untuk memverifikasi respons sistem pada kondisi tidak terduga [9].

A. Arsitektur Sistem

Sistem yang diuji terdiri dari dua komponen utama yang saling terintegrasi. Pertama, aplikasi Android sebagai antarmuka pengguna yang mengelola proses pelaporan, mulai dari pengambilan gambar menggunakan kamera perangkat, pengambilan koordinat GPS secara otomatis, hingga pengiriman laporan ke server dan visualisasi laporan pada peta digital. Kedua, server Django yang menjalankan proses inferensi model kecerdasan buatan. Gambar yang dikirimkan dari Android diproses oleh model ResNet-50 untuk klasifikasi jenis kerusakan dan model YOLOv12 untuk deteksi dan lokalisasi objek kerusakan. Hasil inferensi dikembalikan ke Android dalam format JSON melalui REST API.



Gambar 1. Tampilan Antarmuka Aplikasi Pelaporan Kerusakan Jalan

Tampilan antarmuka aplikasi Android yang dikembangkan ditunjukkan pada Gambar 1. Antarmuka terdiri dari halaman utama, halaman pelaporan kerusakan, daftar laporan, detail laporan, dan pemetaan lokasi kerusakan jalan.

B. Struktur Lembar Kerja

Sistem dibagi menjadi dua lembar kerja berdasarkan lingkungan eksekusinya. Lembar kerja pelaporan mencakup enam komponen yang berjalan di sisi Android, yaitu komponen register, lapor kerusakan, ambil lokasi GPS, daftar laporan, detail laporan, dan pemetaan laporan. Lembar kerja sistem cerdas mencakup tiga komponen yang berjalan di sisi server Django, yaitu komponen penerimaan gambar, inferensi model, dan pengembalian hasil.

C. Rancangan Pengujian

Setiap skenario pengujian dirancang dengan mempertimbangkan tiga jenis kondisi uji. Kondisi normal untuk memverifikasi alur fungsi utama berjalan sesuai rancangan. Kondisi preventif untuk memastikan sistem mampu menolak atau menangani input yang tidak valid sebelum terjadi kesalahan. Kondisi koreksi untuk memverifikasi bahwa sistem memberikan respons yang informatif dan tepat ketika terjadi kondisi tidak terduga seperti koneksi terputus atau inferensi model gagal. Setiap skenario mencatat komponen yang diuji, kondisi uji, input yang diberikan, output yang diharapkan, output aktual, dan status hasil pengujian.

D. Lingkungan Pengujian

Pengujian dilakukan pada perangkat Android dengan spesifikasi RAM 8 GB dan sistem operasi Android 15. Server Django dijalankan pada laptop dengan prosesor Intel Core i7 Gen 7, RAM 32 GB, dan GPU NVIDIA GeForce 940MX. Koneksi antara perangkat Android dan server menggunakan jaringan lokal WiFi untuk meminimalkan latensi komunikasi. Pencatatan hasil pengujian fungsional dilakukan secara manual berdasarkan observasi langsung terhadap output yang ditampilkan aplikasi maupun respons yang dikembalikan oleh server, kemudian dibandingkan dengan output yang diharapkan untuk menentukan status keberhasilan setiap skenario. Pengukuran latensi inferensi dilakukan dengan mencatat selisih waktu antara pengiriman gambar dari Android hingga respons JSON diterima kembali oleh aplikasi, menggunakan 10 sampel gambar kerusakan jalan yang berbeda.

III. HASIL DAN PEMBAHASAN

A. Hasil Pengujian Lembar Kerja Pelaporan

Pengujian lembar kerja pelaporan dilakukan terhadap 26 skenario yang mencakup seluruh enam komponen pada sisi Android. Hasil pengujian disajikan pada Tabel 1 berikut.

Tabel 1. Hasil Pengujian Lembar Kerja Pelaporan

No	Nama Fitur	Skenario Uji	Input	Output yang Diharapkan	Output Aktual	Status
1	Register	Pendaftaran akun sukses dengan data yang memenuhi syarat	Nama pengguna dan kata sandi minimal 8 karakter	Akun berhasil dibuat dan pengguna dialihkan ke halaman beranda	Akun terdaftar dan aplikasi berpindah ke halaman beranda	✓
2	Register	Pendaftaran ditolak apabila nama pengguna telah terdaftar	Nama pengguna yang sudah ada dalam sistem	Aplikasi menampilkan notifikasi "username sudah digunakan"	Notifikasi "username sudah digunakan" berhasil ditampilkan	✓
3	Register	Pendaftaran ditolak saat kolom wajib tidak diisi	Salah satu kolom formulir dibiarkan kosong	Pesan validasi muncul pada kolom yang belum terisi	Pesan validasi tampil pada kolom yang kosong	✓
4	Register	Pendaftaran gagal jika	Kata sandi dan pengisian	Aplikasi menampilkan	Pesan tidaksesuai	✓

		pengisian ulang kata sandi tidak sesuai	ulangnya berbeda	pesan “Konfirmasi password tidak cocok”	kata sandi berhasil ditampilkan	
5	Lapor Kerusakan	Kamera perangkat berhasil diinisialisasi	Pengguna mengakses menu Lapor Kerusakan	Layar pratinjau kamera aktif dan tampil di antarmuka	Pratinjau kamera muncul pada antarmuka laporan	✓
6	Lapor Kerusakan	Foto berhasil diambil dan ditampilkan pada antarmuka	Pengguna menekan tombol pengambilan gambar	Hasil foto muncul di area pratinjau formulir laporan	Foto berhasil ditampilkan pada area pratinjau	✓
7	Lapor Kerusakan	Foto diunggah ke server segera setelah pengambilan	Gambar berhasil ditangkap oleh kamera	Permintaan API dikirimkan ke server beserta data gambar	Permintaan API berhasil dikirim ke server	✓
8	Lapor Kerusakan	Tombol pengiriman laporan tidak dapat diklik sebelum proses deteksi selesai	Gambar baru diambil namun hasil inferensi belum kembali	Tombol Kirim Laporan dalam kondisi tidak aktif	Tombol Kirim Laporan terkunci hingga deteksi selesai	✓
9	Lapor Kerusakan	Hasil inferensi ditampilkan setelah data dari server diterima	Data JSON hasil deteksi diterima dari server	Gambar dilengkapi anotasi bounding box, kategori kerusakan, dan tingkat prioritas	Gambar teranotasi beserta informasi kerusakan berhasil ditampilkan	✓
10	Lapor Kerusakan	Data laporan tersimpan ke basis data setelah dikirim	Pengguna mengonfirmasi pengiriman laporan	Rekaman laporan tersimpan di server dan notifikasi keberhasilan muncul	Data tersimpan dan notifikasi berhasil pengiriman tampil	✓
11	Lapor Kerusakan	Kolom keterangan tambahan tidak bersifat wajib diisi	Kolom keterangan dibiarkan kosong oleh pengguna	Laporan tetap dapat dikirim meskipun keterangan tidak diisi	Laporan terkirim tanpa perlu mengisi keterangan	✓
12	Ambil Lokasi GPS	Data koordinat GPS diperoleh secara otomatis oleh sistem	Sinyal GPS tersedia dan fitur lokasi perangkat aktif	Nilai latitude dan longitude tampil secara otomatis tanpa input manual	Koordinat lokasi muncul secara otomatis	✓
13	Ambil Lokasi GPS	Koordinat GPS dikonversi	Data koordinat telah berhasil diperoleh	Nama jalan atau alamat lokasi	Alamat lokasi berhasil ditampilkan	✓

		menjadi nama alamat		ditampilkan menggantikan koordinat numerik		
14	Ambil Lokasi GPS	Sistem meminta persetujuan akses lokasi bila izin belum diberikan	Izin akses lokasi belum diaktifkan pengguna	Dialog permohonan izin lokasi muncul pada layar	Dialog permintaan izin akses lokasi tampil	✓
15	Daftar Laporan	Riwayat laporan ditampilkan dengan urutan terbaru di atas	Pengguna membuka menu Daftar Laporan	Semua laporan tersaji dalam bentuk kartu, diurutkan berdasarkan waktu terbaru	Laporan tampil berurutan dari yang paling baru	✓
16	Daftar Laporan	Pesan kosong ditampilkan saat belum ada data laporan	Menu Daftar Laporan dibuka dalam kondisi tidak ada data	Muncul keterangan “Belum ada laporan yang tersimpan”	Keterangan data kosong berhasil ditampilkan	✓
17	Daftar Laporan	Setiap kartu menampilkan ringkasan informasi laporan	Data laporan tersedia di basis data	Kartu menyajikan foto, kategori kerusakan, lokasi, waktu pelaporan, dan prioritas	Seluruh informasi ringkas berhasil tampil pada kartu	✓
18	Daftar Laporan	Daftar dapat digulir tanpa gangguan saat data berjumlah banyak	Lebih dari satu laporan tersimpan di sistem	Pengguna dapat menggulir tampilan daftar tanpa mengalami error	Gulir daftar berjalan lancar tanpa hambatan	✓
19	Daftar Laporan	Laporan yang baru dikirim langsung muncul di posisi teratas	Laporan baru dikirim kemudian menu Daftar Laporan dibuka	Entri laporan terbaru tampil pada posisi paling atas daftar	Laporan terbaru muncul di urutan pertama	✓
20	Detail Laporan	Halaman detail terbuka ketika kartu laporan dipilih	Pengguna mengetuk salah satu kartu pada daftar laporan	Halaman detail terbuka dan menyajikan informasi laporan secara lengkap	Halaman detail berhasil terbuka dengan data yang sesuai	✓
21	Detail Laporan	Foto beranotasi berhasil ditampilkan pada halaman detail	Halaman detail laporan dibuka	Citra kondisi jalan dengan bounding box dan label jenis kerusakan tampil	Gambar dengan anotasi berhasil ditampilkan	✓

22	Detail Laporan	Seluruh atribut laporan tampil secara lengkap	Halaman detail laporan dibuka	Kategori kerusakan, koordinat, waktu, prioritas, dan keterangan tampil seluruhnya	Semua atribut laporan berhasil ditampilkan	✓
23	Detail Laporan	Tombol navigasi peta membuka aplikasi maps eksternal	Pengguna menekan tombol Lihat Lokasi di Map	Aplikasi peta pihak ketiga terbuka dan mengarah ke titik koordinat laporan	Google Maps terbuka pada koordinat lokasi yang sesuai	✓
24	Pemetaan Laporan	Antarmuka peta berhasil dimuat tanpa kendala	Pengguna membuka menu Pemetaan Laporan	Tampilan peta digital berhasil dirender pada layar	Peta digital tampil dengan baik tanpa error	✓
25	Pemetaan Laporan	Penanda lokasi laporan tampil pada koordinat yang sesuai	Data laporan tersedia di basis data	Marker lokasi muncul pada peta sesuai dengan koordinat masing-masing laporan	Penanda lokasi tampil pada posisi yang benar	✓
26	Pemetaan Laporan	Informasi singkat laporan muncul saat penanda dipilih	Pengguna mengetuk salah satu marker pada peta	Jendela popup menampilkan waktu pelaporan, kategori kerusakan, dan kode laporan	Popup informasi tampil dengan data yang akurat	✓

Berdasarkan Tabel 1, seluruh 26 skenario pengujian pada lembar kerja pelaporan menunjukkan hasil berhasil. Pada komponen register, lima skenario preventif memastikan sistem mampu menangani input tidak valid seperti email yang sudah terdaftar, kolom yang kosong, format email salah, dan password terlalu pendek. Komponen lapor kerusakan berhasil memvalidasi pengiriman gambar, menampilkan hasil deteksi dari server, dan menangani kondisi koneksi terputus tanpa kehilangan data. Komponen ambil lokasi GPS berhasil mendapatkan koordinat secara otomatis serta memberikan peringatan yang tepat ketika GPS tidak aktif atau izin lokasi ditolak. Komponen daftar laporan, detail laporan, dan pemetaan berhasil menampilkan data dengan benar termasuk menangani kondisi ketika data belum tersedia.

B. Hasil Pengujian Latensi Inferensi Model

Pengujian latensi inferensi model dilakukan untuk mengukur kecepatan respons sistem dari server Django ke perangkat Android. Pengukuran dilakukan terhadap 10 sampel gambar kerusakan jalan yang berbeda, mencatat waktu total sejak permintaan dikirim dari Android hingga respons JSON diterima kembali oleh aplikasi. Hasil pengujian disajikan pada Tabel 2 berikut.

Tabel 2. Hasil Pengujian Latensi Inferensi Model

No	Upload Gambar (ms)	Inferensi ResNet-50 (ms)	Inferensi YOLOv12 (ms)	Total (ms)
1	151,37	1.900,71	845,65	3.123,65
2	106,44	1.887,30	778,84	2.991,16
3	108,57	1.889,10	837,22	3.169,95
4	130,17	1.928,41	877,45	3.209,94
5	132,88	1.945,10	837,78	3.097,84
6	107,94	1.869,16	955,46	3.181,32
7	115,82	1.874,19	903,78	3.311,53
8	110,06	1.840,75	859,92	3.151,33
9	145,47	1.927,10	863,23	3.229,87
10	127,46	1.861,99	838,01	3.039,97
Rata-Rata	123,62	1.892,38	859,73	3.150,66

Berdasarkan Tabel 2, hasil pengujian waktu respon inferensi model terhadap 10 sampel gambar kerusakan jalan menunjukkan rata-rata total waktu respon sebesar 3.150,66 ms dengan nilai minimum 2.991,16 ms dan maksimum 3.311,53 ms. Komponen yang paling mendominasi waktu respon adalah inferensi ResNet-50 dengan rata-rata 1.892,38 ms atau sekitar 60% dari total waktu, diikuti inferensi YOLOv12 dengan rata-rata 859,73 ms, dan proses unggah gambar dengan rata-rata 123,62 ms. Hal ini mengindikasikan bahwa *bottleneck* utama sistem terletak pada proses inferensi model klasifikasi ResNet-50, bukan pada transmisi data jaringan. Meskipun rata-rata total waktu respon berada di kisaran 3 detik, seluruh sampel menunjukkan performa yang konsisten dan stabil, mengonfirmasi bahwa sistem mampu memberikan hasil identifikasi kerusakan jalan secara andal dalam kondisi jaringan lokal.

C. Rekapitulasi Hasil Pengujian

Rekapitulasi keseluruhan hasil pengujian fungsional dan latensi disajikan pada Tabel 3 berikut.

Tabel 3. Rekapitulasi Hasil Pengujian

Jenis Pengujian	Jumlah Data	Hasil
Black Box Testing (Pelaporan Android)	26	26 (100%)
Latensi Inferensi Model	10 sampel	Rata-rata 3.150,66 ms
Total	26 skenario + 10 sampel	26 (100%) + rata-rata 3.150,66 ms

Tabel 3 merangkum dua jenis pengujian yang dilakukan. Pengujian fungsional *Black Box* terhadap lembar kerja pelaporan mencatat 26 skenario dengan tingkat keberhasilan 100%, membuktikan bahwa seluruh fitur Android berfungsi sesuai rancangan. Pengujian waktu respon inferensi terhadap 10 sampel gambar menunjukkan rata-rata total waktu respon 3.150,66 ms, dengan dominasi waktu pada komponen inferensi ResNet-50 sebesar rata-rata 1.892,38 ms. Secara keseluruhan, hasil ini menunjukkan bahwa sistem telah diimplementasikan secara *robust* dan konsisten, siap digunakan dalam kondisi operasional nyata.

D. Pembahasan

Keberhasilan seluruh skenario pengujian mengonfirmasi bahwa integrasi antara aplikasi Android dan server Django melalui REST API telah berjalan dengan baik. Pendekatan pengujian yang bersifat preventif terbukti efektif dalam mengidentifikasi potensi kegagalan sistem akibat input yang tidak sesuai, seperti pengiriman request tanpa gambar, format file yang salah, dan penolakan izin akses GPS. Sementara itu, skenario koreksi membuktikan bahwa sistem mampu memberikan umpan balik yang informatif kepada pengguna ketika terjadi kondisi di luar kendali, seperti koneksi jaringan terputus atau kegagalan proses inferensi model.

Dibandingkan dengan penelitian terdahulu yang umumnya hanya menguji aspek akurasi model [6], [7], [8], penelitian ini memberikan kontribusi tambahan berupa verifikasi fungsionalitas sistem secara menyeluruh melalui Black Box Testing yang terstruktur. Hal ini sejalan dengan pandangan yang menyatakan bahwa *Black Box Testing* merupakan pendekatan yang efektif untuk memverifikasi perilaku sistem dari perspektif pengguna akhir tanpa bergantung pada pengetahuan implementasi internal.

Hasil pengujian waktu respon inferensi menunjukkan bahwa komponen yang paling berpengaruh terhadap total waktu respon adalah inferensi ResNet-50 dengan rata-rata 1.892,38 ms, yang menyumbang sekitar 60% dari keseluruhan waktu respon rata-rata 3.150,66 ms. Sementara itu, proses unggah gambar hanya memerlukan rata-rata 123,62 ms, mengindikasikan bahwa *bottleneck* sistem bukan pada transmisi jaringan melainkan pada komputasi model klasifikasi. Hal ini wajar mengingat ResNet-50 merupakan arsitektur CNN yang dalam dengan 50 lapisan, sehingga membutuhkan waktu komputasi lebih lama dibandingkan YOLOv12 yang dioptimalkan untuk kecepatan deteksi *real-time*. Pengujian lanjutan pada kondisi jaringan yang berbeda, seperti jaringan seluler 4G, serta optimasi model melalui teknik seperti *quantization* atau *pruning* direkomendasikan untuk meningkatkan responsivitas sistem secara keseluruhan.

IV. KESIMPULAN

Pengujian sistem pelaporan kerusakan jalan berbasis Android telah dilakukan dalam dua tahap. Pertama, pengujian fungsional menggunakan metode *Black Box Testing* terhadap lembar kerja pelaporan yang mencakup 26 skenario, seluruhnya menunjukkan hasil berhasil dengan tingkat keberhasilan 100%. Pengujian yang bersifat preventif memastikan sistem mampu menangani input tidak valid secara tepat sebelum terjadi kegagalan, sedangkan pengujian korektif mengonfirmasi bahwa sistem memberikan respons yang informatif pada kondisi tidak terduga. Kedua, pengujian waktu respon inferensi model terhadap 10 sampel gambar kerusakan jalan menunjukkan rata-rata total waktu respon sebesar 3.150,66 ms, dengan komponen inferensi ResNet-50 sebagai kontributor terbesar (rata-rata 1.892,38 ms), diikuti YOLOv12 (rata-rata 859,73 ms), dan proses unggah gambar (rata-rata 123,62 ms).

Hasil pengujian membuktikan bahwa integrasi model YOLOv12 dan ResNet-50 melalui Django REST API ke dalam aplikasi Android telah berfungsi secara *robust* dan konsisten. Sistem mampu mendeteksi empat jenis kerusakan jalan dengan rata-rata total waktu respon 3.150,66 ms pada jaringan lokal, dengan *bottleneck* utama pada komponen inferensi ResNet-50. Optimasi lebih lanjut pada model klasifikasi, seperti penerapan teknik *quantization* atau *pruning*, direkomendasikan untuk meningkatkan kecepatan respon sistem. Penelitian ini diharapkan dapat menjadi referensi dalam pengembangan sistem pelaporan infrastruktur berbasis kecerdasan

buatan yang memperhatikan aspek *robustness* melalui pengujian fungsional terstruktur dan pengujian waktu respon inferensi yang komprehensif.

DAFTAR PUSTAKA

- [1] A. Nidia Agustin, M. Mulyadi, and I. R. Amirulloh, "Peran Dinas Pekerjaan Umum dalam Pemeliharaan dan Perbaikan Infrastruktur Jalan di Ruas Jalan Pangleseran-Cibatu Kabupaten Sukabumi," *Innov. J. Soc. Sci. Res.*, vol. 5, no. 1, pp. 6606–6617, Feb. 2025, doi: 10.31004/INNOVATIVE.V5I1.17922.
- [2] T. A. Dompeipen, S. R. U. A. Sompie, and M. E. . Najoan, "Computer Vision Implementation for Detection and Counting the Number of Humans," *J. Tek. Inform.*, vol. 16, no. 1, pp. 65–76, Mar. 2021, doi: 10.35793/JTI.V16I1.31471.
- [3] M. A. Ghofur and A. B. Ulum, "Analisis Perbandingan Kinerja Model ResNet50 Dan EfficientNet-B0 Dalam Klasifikasi Kerusakan Jalan," *J. Media Inform.*, vol. 6, no. 5, pp. 2504–2511, Sep. 2025, doi: 10.55338/JUMIN.V6I5.6843.
- [4] N. Deng and Q. Song, "Research on Improved Pavement Distress Detection Algorithm of Yolov12 Acknowledgments," *Mod. Econ. Manag. Forum*, vol. 6, no. 5, pp. 730–732, 2025, [Online]. Available: <https://front-sci.com/journal/article?doi=10.32629/memf.v6i5.4524>
- [5] M. A. Saputra, W. C. Utomo, A. B. Setiawan, and I. K. Ramadhanu, "Pengamatan Cuaca Lokal secara Multi Node dengan Internet of Things dan Django Framework," *JITU J. Inform. Technol. Commun.*, vol. 8, no. 1, pp. 31–40, May 2024, Accessed: Oct. 28, 2025. [Online]. Available: <https://ejournal.uby.ac.id/index.php/jitu/article/view/1334>
- [6] M. Surahmanto, S. Aras, M. Rifki, I. Adhim, and P. Ussalama, "Deteksi Jalan Berlubang Menggunakan Algoritma Yolov5: Pothole Detection Using the Yolov5 Algorithm," *J. Digit. Bus. Inf. Technol.*, vol. 1, no. 1, pp. 1–8, Jun. 2024, doi: 10.23971/JOBIT.V1I1.198.
- [7] H. Maeda, T. Kashiyama, Y. Sekimoto, T. Seto, and H. Omata, "Generative adversarial network for road damage detection," *Comput. Civ. Infrastruct. Eng.*, vol. 36, no. 1, pp. 47–60, Jan. 2021, doi: 10.1111/mice.12561.
- [8] Y. Zhang *et al.*, "Road damage detection using UAV images based on multi-level attention mechanism," *Autom. Constr.*, vol. 144, Dec. 2022, doi: 10.1016/j.autcon.2022.104613.
- [9] A. Suderajat *et al.*, "Penerapan Manajemen Risiko dalam Software Quality Assurance: Studi Kasus Pengujian Black Box pada Aplikasi Inventori Maintenance," *J. Indones. Manaj. Inform. dan Komun.*, vol. 6, no. 1, pp. 438–456, Jan. 2025, doi: 10.35870/JIMIK.V6I1.1214.
- [10] M. Maulida, F. Zahro, R. Hakim, M. S. Akbar, S. Pd, and M. Kom, "PENGUJIAN BLACK BOX TESTING PADA SISTEM WEBSITE PEMESANAN ONLINE TOKO AYAM KRISPY," *J. Media Akad.*, vol. 3, no. 5, pp. 3031–5220, May 2025, doi: 10.62281/V3I5.1908.