SIMULASI SISTEM DETEKSI KERUSAKAN JALAN BERBASIS YoloV8 MENGGUNAKAN INPUT VIDEO DENGAN RESPONS PENGEREMAN OTOMATIS BERBASIS IoT

SKRIPSI

DISUSUN OLEH

ALMIFTAH NAZWA KHAIRUNNIZAM

NPM: 2209020081

PROGRAM STUDI TEKNOLOGI INFORMASI

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA

MEDAN

2026

SIMULASI SISTEM DETEKSI KERUSAKAN JALAN BERBASIS YoloV8 MENGGUNAKAN INPUT VIDEO DENGAN RESPONS PENGEREMAN OTOMATIS BERBASIS IOT

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer (S.Kom) dalam Program Studi Teknologi Informasi pada Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara

ALMIFTAH NAZWA KHAIRUNNIZAM

NPM: 2209020081

PROGRAM STUDI TEKNOLOGI INFORMASI

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA

MEDAN

2026

LEMBAR PENGESAHAN

Judul Skripsi	:	SIMULASI SISTEM DETEKSI KERUSAKAN JALAN BERBASIS YoloV8 MENGGUNAKAN INPUT VIDEO DENGAN RESPONS PENGEREMAN OTOMATIS BERBASIS IOT
Nama Mahasiswa	:	ALMIFTAH NAZWA KHAIRUNNIZAM
NPM	:	2209020081
Program Studi	:	TEKNOLOGI INFORMASI

Menyetujui

Komisi Pembimbing

(Dr. Firahmi Rizky, S.Kom., M.Kom)

NIDN. 0116079201

Ketua Program Studi

(Fatma Sari Hutagalung, S.Kom, M.Kom)

NIDN. 0117019301

Dekan

(Dr. [illegible], S.Kom., M.Kom.)

NIDN. 0127099201

PERNYATAAN ORISINALITAS

SIMULASI SISTEM DETEKSI KERUSAKAN JALAN BERBASIS YoloV8 MENGGUNAKAN INPUT VIDEO DENGAN RESPONS PENGEREMAN OTOMATIS BERBASIS IOT

SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, 1 Agustus 2026

Yang membuat pernyataan



Almiftah Nazwa KhairunNizam

NPM. 220902008

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan dibawah ini:

Nama	: Almiftah Nazwa KhairunNizam
NPM	: 2209020081
Program Studi	: Teknologi Informasi
Karya Ilmiah	: Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bedas Royalti Non-Eksekutif *(Non-Exclusive Royalty free Right)* atas penelitian skripsi saya yang berjudul:

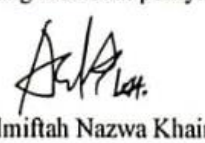
SIMULASI SISTEM DETEKSI KERUSAKAN JALAN BERBASIS YoloV8 MENGGUNAKAN INPUT VIDEO DENGAN RESPONS PENGEREMAN OTOMATIS BERBASIS IOT

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, 1 Agustus 2026

Yang membuat pernyataan



Almiftah Nazwa KhairunNizam

NPM. 2209020081

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap	: Almiftah Nazwa KhairunNizam
Tempat dan Tanggal Lahir	: Medan, 22 Agustus 2004
Alamat Rumah	: Jln. Tuar Raya Blok XI, Komplek Griya Martubung
Telepon/Faks/HP	: 082396071192
E-Mail	: allexmiftah@gmail.com
Instansi Tempat Kerja	: -
Alamat Kantor	: -

DATA PENDIDIKAN

SD : SD Muhammadiyah 02 Medan Timur (2016)

SMP : SMP Muhammadiyah 07 Medan Perjuangan (2019)

SMK : SMK Tritech Informatika Medan Tembung (2022)

KATA PENGANTAR



Assalamu'alaikum Warahmatullahi Wabarakatuh

Puji syukur Alhamdulillah Penulis ucapkan kehadirat Allah SWT. Yang telah melimpahkan rahmat, hidayah, serta karunia-Nya kepada penulis sehingga penulis dapat menyelesaikan skripsi di Universitas Muhammadiyah Sumatera Utara yang berjudul "SIMULASI SISTEM DETEKSI KERUSAKAN JALAN BERBASIS YOLO V8 MENGGUNAKAN INPUT VIDEO DENGAN RESPONS PENGEREMAN OTOMATIS BERBASIS IOT" skripsi ini disusun dengan baik dan tepat pada waktunya. Hasil penyusunan skripsi disusun berdasarkan observasi, Studi Pustaka dan juga Studi Literatur. Skripsi ini disusun sedemikian rupa dengan tujuan agar dapat diterima dan dapat dipakai sebagai usulan bagi semua pihak yang nantinya juga akan melakukan penelitian dan memenuhi tugas akhir penyusunan skripsi di Universitas Muhammadiyah Sumatera Utara.

Penulis tentunya berterima kasih kepada berbagai pihak dalam dukungan serta do'a dalam penyelesaian skripsi, terutama untuk kedua orang tua saya Ayah & Ibu yang telah banyak membantu saya dalam penyelesaian skripsi dan tak bosan-bosannya mengingatkan untuk mengerjakan skripsi. Penyelesaian skripsi ini juga tidak lepas dari bimbingan dan dukungan berbagai pihak. Untuk itu, penulis mengucapkan terima kasih sebesar-besarnya kepada:

1. Bapak Prof. Dr. Akrim, M.Pd., Rektor Universitas Muhammadiyah Sumatera Utara (UMSU).
2. Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom. Dekan Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
3. Ibu Dr. Firahmi Rizky, M.Kom., Sebagai Wakil Dekan I Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
4. Bapak Muhammad Basri, M.Kom. Sebagai Wakil Dekan III Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
5. Ibu Fatma Sari Hutagalung, M.Kom., Sebagai Ketua Program Studi Teknologi Informasi Universitas Muhammadiyah Sumatera Utara.
6. Bapak Okvi Nugroho, S.Kom., M.Kom., Sebagai Sekretaris Program Studi Teknologi Informasi Universitas Muhammadiyah Sumatera Utara.
7. Ibu Dr. Firahmi Rizky, M.Kom. Sebagai Dosen Pembimbing yang membantu Saya dalam penyelesaian skripsi ini.
8. Kepada sahabat saya dan teman saya sampai saat ini yang mana selama masa kuliah selalu ada dalam susah maupun senang dan banyak memberikan hal-hal baru dan pengalaman yang berkesan dalam hidup saya.
9. Seluruh civitas akademika UMSU dan semua pihak yang terlibat langsung maupun tidak langsung yang tidak dapat penulis ucapkan satu-persatu yang telah membantu penyelesaian skripsi ini.

Penulis menyadari skripsi ini masih jauh dari kata sempurna dan sangat terbuka terhadap kritik serta saran yang dapat membangun. Semoga karya ini dapat bermanfaat bagi pengembangan ilmu pengetahuan, khususnya di bidang Teknologi Informasi.

Medan, 1 Agustus 2026

Penulis



Almiftah Nazwa KhairunNizam

SIMULASI SISTEM DETEKSI KERUSAKAN JALAN BERBASIS YoloV8 MENGGUNAKAN INPUT VIDEO DENGAN RESPONS PENGEREMAN OTOMATIS BERBASIS IOT

ABSTRAK

Infrastruktur jalan yang rusak, seperti lubang, retakan, dan polisi tidur yang tidak terstandarisasi, menjadi salah satu penyebab utama kecelakaan lalu lintas dan penurunan efisiensi transportasi di Indonesia. Penelitian ini bertujuan untuk merancang dan membangun sebuah sistem simulasi otomatis yang mampu mendeteksi kerusakan jalan secara *near real-time* dan memberikan respons fisik berupa peringatan serta simulasi pengereman. Metode yang digunakan adalah algoritma YOLOv8 untuk pengolahan citra digital berbasis video dan integrasi Internet of Things (IoT) menggunakan mikrokontroler ESP32. Sistem ini memproses input video untuk mengidentifikasi tujuh jenis kerusakan jalan, yaitu *pothole, speed bump, alligator crack, lateral crack, longitudinal crack, edge cracking,* dan *ravelling*. Hasil penelitian menunjukkan bahwa model YOLOv8 mencapai nilai *mean Average Precision* (mAP@0.5) sebesar 0,454, dengan performa tertinggi pada deteksi *speed bump* (AP 0,903). Integrasi IoT berhasil memberikan respons melalui *buzzer* dan motor servo sebagai simulasi pengereman bertahap dengan waktu respons rata-rata di bawah 1 detik. Selain itu, sebuah antarmuka *website* monitoring telah berhasil diimplementasikan untuk merekam dan menampilkan data deteksi secara terstruktur. Kesimpulan dari penelitian ini adalah sistem mampu mengintegrasikan visi komputer dan IoT sebagai prototipe sistem peringatan dini untuk meningkatkan keselamatan berkendara, meskipun masih diperlukan pengembangan pada variasi dataset dan pengujian di kondisi jalan nyata

Kata Kunci: YOLOv8, Kerusakan Jalan, IoT, ESP32, Simulasi Pengereman, *Computer Vision*.

ROAD DAMAGE DETECTION SYSTEM SIMULATION BASED ON YoloV8 USING VIDEO INPUT WITH IOT-BASED AUTOMATIC BRAKING RESPONSE

ABSTRACT

Damaged road infrastructure, such as potholes, cracks, and non-standardized speed bumps, is a major cause of traffic accidents and transportation inefficiency in Indonesia. This research aims to design and build an automated simulation system capable of detecting road damage in near real-time and providing physical responses in the form of warnings and braking simulations. The method employs the YOLOv8 algorithm for video-based digital image processing and Internet of Things (IoT) integration using the ESP32 microcontroller. The system processes video input to identify seven types of road damage: pothole, speed bump, alligator crack, lateral crack, longitudinal crack, edge cracking, and ravelling. The results show that the YOLOv8 model achieved a mean Average Precision (mAP@0.5) of 0.454, with the highest performance in speed bump detection (AP 0.903). The IoT integration successfully provided responses via a buzzer and a servo motor as a gradual braking simulation with an average response time of less than 1 second. Additionally, a monitoring website interface was successfully implemented to record and display detection data in a structured manner. The conclusion of this study is that the system successfully integrates computer vision and IoT as an early warning system prototype to enhance driving safety, although further development in dataset variation and real-world road testing is still required

Keywords: *YOLOv8, Road Damage, IoT, ESP32, Braking Simulation, Computer Vision.*

DAFTAR ISI

LEMBAR PENGESAHAN ..... **i**
PERNYATAAN ORISINALITAS ..... **ii**
PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS ..... **iii**
RIWAYAT HIDUP ..... **iv**
KATA PENGANTAR ..... **v**
ABSTRAK ..... **viii**
ABSTRACT ..... **ix**
DAFTAR ISI ..... **x**
DAFTAR GAMBAR ..... **xii**
DAFTAR TABEL ..... **xiii**
BAB I PENDAHULUAN **1**
1.1. Latar Belakang 1
1.2. Rumusan Masalah 4
1.3. Tujuan Penelitian 4
1.4. Batasan Masalah 5
1.5. Manfaat Penelitian 7
BAB II LANDASAN TEORI **9**
2.1. Kerusakan Jalan 9
2.1.1. Definisi Dan Fungsi Jalan 9
2.2. Computer Vision 11
2.2.1. Pengertian Computer Vision 11
2.2.2. Fungsi dan Kegunaan Computer Vision dalam Pengolahan Citra 11
2.2.3. Peran Computer Vision dalam Deteksi Kerusakan Jalan 13
2.2.4. Keterbatasan dan Tantangan Implementasi 13
2.3. Yolo (You Only Look Once) 14
2.3.1. Sejarah dan Evolusi YOLO (v1 sampai v8) 14
2.3.2. Perbedaan YOLO dengan Model Deteksi Lain 16
2.3.3. Kelebihan dan Kekurangan YOLO pada Sistem Real-Time 17
2.4. IoT (Internet of Things) 19
2.4.1. Pengertian Internet of Things (IoT) 19
2.4.2. Arsitektur IoT 20
2.4.3. Peran IoT dalam Sistem Monitoring dan Peringatan 21
2.4.4. Tantangan dan Pertimbangan Praktis 22
BAB III METODOLOGI PENELITIAN **23**
3.1. Jenis dan Pendekatan Penelitian 23
3.1.1. Jenis Penelitian 23
3.1.2. Pendekatan Penelitian 23
3.2. Tahapan Penelitian 24
3.2.1. Identifikasi Masalah 25
3.2.2. Studi Literatur 25
3.2.3. Pengumpulan Dataset 25
3.2.4. Perancangan Sistem 25
3.2.5. Pelatihan Model Yolo v8 25
3.2.6. Implementasi IoT 26

3.2.7. Pengujian dan Evaluasi 26
3.2.8. Analisis Hasil 26
3.3. Lokasi dan Waktu Penelitian 27
3.4. Sumber dan Jenis Data 28
3.5. Teknik Pengumpulan Data 30
3.5.1. Observasi Lapangan 30
3.5.2. Pengambilan Citra dan Video Jalan 30
3.5.3. Anotasi Dataset 31
3.5.4. Studi Dokumentasi dan Dataset Sekunder 31
3.5.5. Dokumentasi Implementasi IoT 32
3.6. Arsitektur Perancangan Sistem 32
3.6.1. Arsitektur Sistem Secara Umum 33
3.6.2. Arsitektur Komponen Perangkat Keras *(Architecture Hardware)* 34
3.6.4. Arsitektur Komponen Perangkat Lunak *(Architecture Software)* 37
3.6.5. Alur Kerja Sistem 40
3.7. Metode Deteksi Kerusakan Jalan Menggunakan YOLO v8 41
3.7.1. Konfigurasi Model YOLO v8 42
3.7.2. Parameter Pelatihan Model 42
3.7.3. Proses Training dan Inference YOLO v8 43
3.8. Integrasi YOLOv8 dengan Perangkat IoT 44
3.8.1. Mekanisme Pengiriman Hasil Deteksi 45
3.8.2. Sistem Peringatan Otomatis 45
BAB IV HASIL DAN PEMBAHASAN 48
4.1. Implementasi Sistem 48
4.1.1. Implementasi YoloV8 48
4.1.2. Implementasi Sistem IoT 49
4.1.3. Implementasi Website / Monitoring 51
4.2. Hasil Pengujian Sistem 54
4.2.1. Analisis Performa Model YOLOv8 54
4.2.2. Pengujian sistem IoT 60
4.2.3. Pengujian Website Monitoring 63
4.2.4. Pengujian Sistem Terintegrasi 67
4.3. Kelebihan dan Keterbatan dari Sistem yang Dibangun 70
4.3.1. Kelebihan Sistem 70
4.3.2. Keterbatasan Sistem 71
BAB V PENUTUP 73
5.1. Kesimpulan 73
5.2. Saran 74
DAFTAR PUSTAKA 76

DAFTAR GAMBAR

Gambar 3. 1 Diagram Alur Tahapan Penelitian 24
Gambar 3. 2 Diagram Arsitektur Komponen Perangkat Keras 35
Gambar 3. 3 Diagram Arsitektur Komponen Perangkat Lunak 37
Gambar 3. 4 Rancangan Website Laporan Kerusakan Jalan 40
Gambar 3. 5 Flowchart Cara Kerja Sistem 41
Gambar 3. 6 Proses Pelatihan 43
Gambar 3. 7 Proses Kesimpulan/Uji 44
Gambar 3. 8 Contoh Deteksi Gambar Kerusakan Jalan 46
Gambar 4. 1 Implementasi Model YoloV8 Deteksi Jalan 49
Gambar 4. 2 Hasil Implementasi Alat IoT 50
Gambar 4. 3 Tampilan Full Website 52
Gambar 4. 4 Grafik Hasil Pelatihan Model YOLOv8 55
Gambar 4. 5 Grafik dari Precision dan Recall 55
Gambar 4. 6 Grafik Dari mAP50 dan mAP50-95 56
Gambar 4. 7 Kurva Precision-Recal Model YOLOv8 57
Gambar 4. 8 Hubungan antara Recall, F1-Score, & Precision Confidence Score 59
Gambar 4. 9 Rangkaian Perangkat ESP32 63
Gambar 4. 10 Pengujian Deteksi dari API lalu masuk ke Database 64
Gambar 4. 11 Hasil yang masuk ke database 65
Gambar 4. 12 Tampilan Halaman Website Monitoring Hasil Deteksi Kerusakan Jalan 66

DAFTAR TABEL

Tabel 3. 1 Perangkat Keras Untuk Percobaan Simulasi.....36
Tabel 3. 2 Standar Parameter Pengujian 43

BAB I

PENDAHULUAN

1.1. Latar Belakang

Infrastruktur jalan merupakan salah satu komponen vital dalam mendukung mobilitas masyarakat dan aktivitas ekonomi di suatu wilayah. Kualitas jalan yang baik berperan penting dalam melakukan aktivitas masyarakat dan juga memberikan keselamatan pengendara, kelancaran transportasi, serta efisiensi distribusi pengiriman barang dan juga jasa. Namun, pada kenyataanya masih banyak jalan di Indonesia yang mengalami kerusakan seperti retakan lubang, adanya polisi tidur dijalan, maupun permukaan yang tidak rata. Berdasarkan data Kementerian PUPR (2023), jalan yang rusak dibedakan menjadi dua yaitu jalan provinsi dan jalan kabupaten kota, pada tahun 2023 Kementerian PUPR mencatat adanya kerusakan jalan Nasional yang tidak mantap baik rusak maupun berlubang.

Adapun penyebab kerusakan jalan yang ditimbulkan berasal dari beban kendaraan yang melebihi kapasitas, ketika beban melebihi kapasitas desain jalan, maka akan mengalami kelelahan material *(fatigue)* yang menyebabkan retak dan deformasi permukaan. Perubahan suhu dan kelembapan dapat mempercepat degradasi permukaan jalan. Misalnya, suhu tinggi membuat aspal menjadi rapuh, sedangkan air yang masuk ke struktur jalan dapat melemahkan konstruksi jalan dan dapat menyebabkan retakan melebar lalu terjadinya pembentukan lubang jalan. Dampak dari kerusakan jalan beragam dan cukup signifikan. kerusakan yang tidak segera ditangani dapat menimbulkan

peningkatan biaya operasional kendaraan, penurunan kenyamanan perjalanan, waktu tempuh yang lebih lama, dan, yang paling krusial, peningkatan risiko kecelakaan lalu lintas terutama bagi pengguna kendaraan roda dua yang sangat terpengaruh oleh permukaan jalan yang tidak rata

Sebagai konsekuensinya, kerusakan jalan yang tidak tertangani tidak hanya memperburuk kondisi infrastruktur, tetapi juga berdampak pada aspek sosial-ekonomi dan keselamatan publik. Selama ini, proses pemantauan kondisi jalan masih banyak dilakukan secara manual melalui survei lapangan langsung. Pendekatan manual ini memiliki keterbatasan signifikan terkait waktu, biaya, dan skala pengamatan, serta risiko human error dalam penilaian kondisi jalan, sehingga tidak efektif untuk pemantauan infrastruktur yang luas dan dinamis. Karena itu, perlu dikembangkan suatu sistem otomatis dan cerdas yang mampu mendeteksi berbagai jenis kerusakan jalan secara *near real-time* untuk mempercepat respon perbaikan dan mengurangi dampak buruk terhadap pengguna jalan.

Dengan kemajuan teknologi *Artificial Intelligence* (AI) dan *Computer Vision* (CV), proses pendeteksian objek dapat dilakukan secara otomatis menggunakan citra digital. Salah satu cara yang banyak digunakan untuk mendeteksi objek adalah melalui metode You Only Look Once (YOLO). Yolo adalah versi terbaru yang diluncurkan pada tahun 2023 oleh Ultralytics. YOLO v8 ini memiliki beberapa keunggulan yang sangat baik, termasuk kecepatan dalam deteksi, efisiensi model, dan tingkat akurasi yang tinggi untuk mendeteksi objek secara real-time (Ultralytics, 2023). Teknologi visi komputer yang memanfaatkan algoritma YOLO dapat digunakan untuk mengidentifikasi

berbagai kondisi jalan di Indonesia, seperti jalan yang berlubang, kerusakan jalan (retakan), atau polisi tidur. Selain itu, teknologi ini dapat diintegrasikan dengan perangkat edge.

Di samping itu, kemajuan Internet of Things (IoT) memungkinkan sistem cerdas untuk terhubung secara langsung dengan perangkat fisik. Dengan bantuan IoT, hasil dari deteksi yang menggunakan algoritma Yolo v8 dapat dihubungkan dengan mikrokontroler seperti ESP32. Hal ini bertujuan untuk memberikan umpan balik otomatis kepada pengendara, contohnya melalui peringatan LED yang berkedip atau sistem peringatan ringan saat mendeteksi kerusakan jalan seperti retakan, lubang, atau polisi tidur. Namun demikian, perlu ditegaskan bahwa penelitian ini dilakukan dalam bentuk simulasi prototype, bukan implementasi langsung pada motor atau kendaraan nyata. Simulasi dilakukan melalui pemrosesan video rekaman kondisi jalan sebagai input sistem untuk mengevaluasi kinerja model dan integrasi IoT secara terkontrol dalam lingkungan penelitian.

Adapun penelitian sebelumnya telah dilakukan terkait pendeteksian kerusakan jalan menggunakan metode berbasis citra. (Y. Li et al., 2024) dalam penelitiannya yang berjudul "RDD-YOLO: Road Damage Detection Algorithm Based on Improved You Only Look Once Version 8" berhasil mendapat f1-score 69.6%, mAP50 dan mAP50-95 62.5% dan validasi data 36.4%.

Berdasarkan uraian tersebut, dapat disimpulkan bahwa penelitian sebelumnya umumnya hanya berfokus pada deteksi visual tanpa adanya respon otomatis terhadap hasil deteksi. Oleh karena itu, penulis membuat penelitian ini untuk menggabungkan *Deep Learning* dan IoT untuk mendeteksi adanya

kerusakan jalan dan memberikan aksi peringatan secara *near real-time*. Sistem ini diharapkan dapat meningkatkan keselamatan pengguna di jalan terutama pengguna jalanan di indonesia yang mana banyaknya jalan yang masih buruk baik kabupaten/kota sampai dengan jalan nasional tingkat provinsi mengalami kerusakan yang cukup banyak dan belum terdata. Semoga dengan adanya solusi inovatif ini kedepannya pemantauan kondisi infrastruktur jalan di Indonesia jadi lebih baik.

1.2. Rumusan Masalah

1. Bagaimana proses simulasi deteksi berbagai jenis kerusakan jalan menggunakan algoritma YOLO v8 berbasis video rekaman?
2. Bagaimana mekanisme integrasi hasil deteksi YOLO v8 dengan perangkat Internet of Things (IoT) dalam lingkungan simulasi?
3. Bagaimana kinerja sistem simulasi deteksi kerusakan jalan berbasis YOLO v8 ditinjau dari parameter akurasi, presisi, recall dan mAP?
4. Seberapa efektif sistem simulasi yang diusulkan dalam memberikan peringatan otomatis terhadap kerusakan jalan melalui integrasi dengan IoT?

1.3. Tujuan Penelitian

Menghasilkan simulasi sistem deteksi otomatis untuk kerusakan jalan dan memberikan respon kepada pengguna kendaraan roda dua agar memberikan fitur keselamatan tambahan pada saat berkendara di jalan raya. Adapun beberapa point penjelasan yaitu:

1. Untuk merancang dan membangun simulasi sistem deteksi berbagai kerusakan jalan menggunakan algoritma Yolo v8 yang mana algoritma ini mampu mengidentifikasi jenis kerusakan seperti lubang di jalan, retakan jalan, dan polisi tidur secara otomatis.
2. Untuk mengintegrasikan hasil dari simulasi deteksi Yolo v8 dengan perangkat IoT berbasis mikrokontroler ESP32 beserta perangkat pendukungnya seperti LED, dan servo agar dapat memberikan respon otomatis kepada pengendara roda dua jika terdeteksi adanya kerusakan dijalan.
3. Untuk menguji kinerja dan tingkat akurasi Yolo v8 dalam mengenali berbagai jenis kerusakan di jalan pada kondisi pencahayaan pada saat mendung siang hari, jarak kerusakan jalan dengan pengendara dan juga sudut pengambilan gambar yang berbeda.
4. Untuk mengetahui dan mengevaluasi efektivitas sistem secara keseluruhan dalam memberikan peringatan otomatis terhadap pengendara kendaraan roda dua saat mendeteksi adanya kerusakan jalan di depan.

1.4. Batasan Masalah

1. Jenis kerusakan dijalan yang diteliti meliputi tiga kategori diantaranya: jalan berlubang *(Pothole),* retakan jalan *(Crack),* dan polisi tidur *(Speed Bump).* Jenis kerusakan lain seperti genangan air, marka jalan yang pudar, jalan bebatuan atau jalan yang berlumpur tidak akan dibahas di dalam penelitian ini.

2. Model deteksi yang digunakan adalah Yolo v8 merupakan versi terbaru dari Algoritma *You Only Look Once* yang dilatih menggunakan dataset gambar jalan yang sudah diberi label secara manual maupun diambil dari dataset publik yang relevan dengan penelitian ini.
3. Perangkat IoT yang digunakan mencakup mikrokontroler ESP32 yang terhubung dengan perangkat pendukung lainnya yang berfungsi sebagai sensor visual, serta servo motor yang berfungsi sebagai simulasi Sistem aktuator simulasi pengereman bertahap berbasis motor servo sebagai representasi sistem pengereman kendaraan. Servo tidak dipasangkan langsung ke kendaraan bermotor melainkan digunakan pada prototype simulasi handle rem untuk mempresentasikan respon sistem terhadap deteksi kerusakan jalan.
4. Integrasi sistem dilakukan secara simulatif, di mana hasil deteksi kerusakan jalan oleh YOLOv8 dikirimkan ke perangkat ESP32 untuk mengendalikan motor servo sebagai representasi mekanisme respon sistem, bukan sebagai sistem pengereman kendaraan sesungguhnya. Pergerakan servo digunakan semata-mata untuk mensimulasikan logika tahapan respon pengereman dalam bentuk perubahan sudut putar, tanpa terhubung pada sistem mekanis kendaraan nyata. Implementasi dilakukan pada skala laboratorium atau lingkungan terkendali, bukan di jalan raya sebenarnya. Uji coba hanya difokuskan pada kemampuan sistem dalam mendeteksi jenis kerusakan dan respon sensor terhadap sinyal peringatan.

5. Uji performa sistem dibatasi pada parameter teknik seperti akurasi deteksi (precision, recall, F1-Score), kecepatan respon sistem (delay), dan fungsi kerja servo simulasi. Faktor eksternal seperti pencahayaan ekstrem, cuaca, atau kondisi jalan basah pasca hujan tidak akan menjadi pembahasan penelitian ini.
6. Sistem tidak dirancang untuk mengontrol kendaraan secara langsung. Fungsi pengereman yang dimodelkan melalui servo hanya bersifat simulasi visual dan mekanik sebagai bukti konsep, tanpa terhubung dengan sistem rem sepeda motor yang sebenarnya, mengingat sistem keamanan kendaraan harus mengikuti standar internasional jadi hanya melakukan simulasi saja.

1.5. Manfaat Penelitian

1. Manfaat Penelitian Secara Teoritis
    a. Menambah wawasan dan pengetahuan dalam bidang kecerdasan buatan (AI), khususnya penerapan algoritma Yolo v8 dalam mendeteksi objek atau kerusakan jalan secara *near real-time.*
    b. Memberikan kontribusi ilmiah terhadap pengembangan integrasi teknologi *Computer Vision* dengan *Internet of Things* sebagai solusi inovatif dalam bidang transportasi cerdas *(smart transportation).* Menjadi referensi penelitian selanjutnya bagi para peneliti yang ingin mengembangkan sistem deteksi kondisi jalan dengan akurasi lebih tinggi atau integrasi dengan sistem kendaraan yang lain atau lebih kompleks.

2. Manfaat Penelitian Secara Praktis

   a. Meningkatkan keselamatan pengguna kendaraan bermotor dengan memberikan aksi secara otomatis ketika kendaraan bermotor mendekati area jalanan yang rusak atau berpotensi berbahaya.
   b. Menjadi prototipe awal sistem monitoring jalan cerdas, yang dapat diterapkan pada kendaraan roda dua untuk membantu pengendara mengenali kondisi jalan didepannya.
   c. Menjadi bahan pembelajaran dan pengembangan inovasi teknologi bagi mahasiswa atau peneliti lain yang tertarik dalam bidang *Artificial Intelligence, Deep Learning, Computer Vision,* dan IoT terapan.

BAB II

LANDASAN TEORI

2.1. Kerusakan Jalan

Kerusakan jalan dapat diartikan sebagai bentuk penurunan kualitas permukaan jalan yang mencakup lubang, retakan, perubahan bentuk, dan bentuk kerusakan lainnya yang langsung berdampak pada keselamatan lalu lintas serta pengeluaran untuk perawatan infrastruktur (Manurung, 2024), secara teknis, pendekatan berbasis pengolahan citra digital ditandai oleh rangkaian tahapan pra-pemrosesan (resizing, normalisasi, augmentasi dan peningkatan kontras), Metode ini memiliki keuntungan dibandingkan dengan inspeksi manual karena dapat secara otomatis mengekstrak fitur tekstur dan spasial dari gambar jalan dalam berbagai kondisi pencahayaan dan gangguan suara yang nyata (Suhendar, 2023).

2.1.1. Definisi Dan Fungsi Jalan

Jalan adalah fasilitas transportasi yang disediakan oleh pemerintah yang bertujuan untuk mendukung pergerakan dan aksesibilitas bagi pengguna kendaraan, baik menggunakan sepeda motor maupun mobil atau kendaraan yang lebih besar (Sakinah et al., 2025). Jalan yang baik dapat memudahkan aksesibilitas kemanapun dan dapat mempercepat pergerakan ekonomi suatu daerah juga dapat memberikan rasa aman, keselamatan, kenyamanan dan pastinya memberikan efisiensi waktu berkendara ke tempat yang akan dituju, dalam hukum di Indonesia definisi jalan adalah prasarana untuk transportasi darat yang meliputi bagian jalan termasuk juga

bangunan pelengkap dan perlengkapannya yang diperuntukan bagi lalu lintas, definisi ini menekankan bahwa “jalan” tidak hanya permukaan aspal/tanah saja yang dilewati oleh kendaraan, tetapi juga mencakup struktur pelengkap seperti halnya jembatan, saluran air, trotoar, dan fasilitas pendukung lain.

Fungsi jalan menjadi hal paling dasar bagi aktivitas manusia dan transportasi barang. Lebih spesifik, infrastruktur jalan yang baik mendukung integrasi ekonomi dan sosial antar daerah dan wilayah, seperti halnya dalam suatu penelitian: “jalan sangatlah penting dalam memperlancar kegiatan ekonomi dan kegiatan sosial lainnya” (Kansil et al., 2023). Dengan kata lain, jalan bukan hanya sekedar media bagi kendaraan roda dua maupun roda empat atau lebih, tetapi juga komponen vital bagi akses masyarakat terhadap layanan publik seperti pendidikan, kesehatan, pekerjaan, distribusi logistik, pergerakan komoditas dan juga mobilitas sosial-ekonomi. Peran yang bersifat strategis sangat penting untuk kemajuan pembangunan suatu daerah. Kualitas dan keberadaan peran ini menjadi syarat utama untuk meningkatkan pelayanan publik serta menarik investasi (Siswanto et al., 2016).

Pengelompokan jalan berdasarkan fungsi lazim dilakukan. Misalnya ada jalan arteri, kolektor dan lokal yang masing-masing memiliki karakteristik dan peran berbeda dalam sistem transportasi.

a. Jalan Arteri: melayani angkutan utama, perjalanan jarak jauh, kecepatan dan volume lalu lintas tinggi (https://kpbu.kemenkeu.go.id)

b. **Jalan Kolektor:** menghubungkan jalan lokal ke jalan arteri; melayani lalu lintas dengan jarak volume menengah (https://kpbu.kemenkeu.go.id).
c. **Jalan Lokal/Lingkungan:** melayani transportasi jarak pendek atau akses lokal misalnya (permukiman, lingkungan), volume dan kecepatan lebih rendah (https://kpbu.kemenkeu.go.id).

2.2. Computer Vision

2.2.1. Pengertian Computer Vision

Visi komputer adalah area dalam ilmu komputer yang berkaitan dengan cara komputer dapat mengumpulkan, mengolah, menganalisis, dan memahami data visual dari dunia nyata seperti gambar atau video. Tujuannya adalah untuk menghasilkan representasi yang bermanfaat untuk membantu dalam pengambilan keputusan atau tindakan secara otomatis (A\Arnita et al., 2022). Dalam literatur modern, Visi Komputer dipandang sebagai rangkaian teknik dan model terutama metode pembelajaran mendalam *(deep learning)* yang memungkinkan ekstraksi fitur visual tingkat rendah hingga tingkat tinggi misalnya (Tepi, tekstur, objek dan scene) dari data citra atau video dalam skala besar.

2.2.2. Fungsi dan Kegunaan Computer Vision dalam Pengolahan Citra

Secara fungsional, *Computer Vision* melayani beberapa tugas dasar dalam pengolahan citra, antara lain: akuisisi dan normalisasi citra *(preprocessing)*, ekstraksi fitur, segmentasi, deteksi objek, klasifikasi, pelacakan *(tracking)*, dan pemahaman scene *(scene understanding)*. Pada

tataran aplikasi, fungsi-fungsi ini diterjemahkan dalam bentuk simulasi sistem otomatis untuk inspeksi kualitas, pemantauan keselamatan, pengenalan pola, serta analisis perilaku yang semuanya bergantung pada pipeline pengolahan citra yang terdiri atas pra-pemrosesan, model inferensi, dan langkah validitas/evaluasi. Kemajuan dalam model jaringan syaraf konvolusional (CNN) dan model yang didasarkan pada transformer telah meningkatkan keakuratan dalam tugas-tugas ini. Selain itu, hal ini juga memperluas kemampuan penglihatan komputer untuk beroperasi dalam berbagai kondisi pencahayaan, sudut, dan jenis kebisingan (Steiniger et al., 2022). Fungsi praktis Visi Komputer dalam pengolahan citra meliputi:

a. Membersihkan dan menormalkan data masukan (resize, normalisasi, augmentasi),
b. Mengekstrak fitur representatif (melalui lapisan konvolusi atau attention),
c. Mengidentifikasi dan melokalisasi objek kepentingan (deteksi bounding box atau segmentasi),
d. Memproduksi keluaran yang dapat ditindaklanjuti misalnya (Peringatan, pelaporan otomatis, tugas kontrol pada perangkat IoT).

Jalur ini memungkinkan sistem untuk mengubah data yang belum diolah menjadi informasi yang teratur dan berguna bagi para pembuat keputusan atau sistem kontrol otomatis (Wang et al., 2023).

2.2.3. Peran Computer Vision dalam Deteksi Kerusakan Jalan

Dalam pengamatan kerusakan jalan, teknologi computer vision berfungsi sebagai elemen utama yang memungkinkan deteksi otomatis jenis dan tempat kerusakan di permukaan jalan. Ini termasuk masalah seperti retakan, lubang, deformasi, dan tambalan yang tidak baik, yang dapat dikenali dari gambar atau video yang ditangkap oleh kamera pada kendaraan atau alat pemantau. Teknik yang berlandaskan pembelajaran mendalam, seperti keluarga YOLO, Faster R-CNN, serta varian efisien seperti Tiny-YOLO atau model transformer ringan, telah terbukti mampu mendeteksi secara langsung dengan tingkat kesalahan yang semakin berkurang setelah memanfaatkan augmentasi data, penyesuaian pada dataset lokal, dan metode optimasi untuk perangkat edge. Penelitian di berbagai lokasi, termasuk kondisi jalan di Indonesia, menunjukkan bahwa akurasi deteksi dapat meningkat dengan penerapan kombinasi CNN/transformer serta pendekatan augmentasi dan penyeimbangan dataset (Rahman, 2023).

2.2.4. Keterbatasan dan Tantangan Implementasi

Walaupun memiliki potensi yang sangat besar, penggunaan computer vision untuk mendeteksi kerusakan pada jalan juga menghadapi beberapa tantangan praktis. Ini termasuk perbedaan dalam kondisi pencahayaan dan cuaca seperti gelap, basah, atau reflektif. Selain itu, variasi dalam jenis material perkerasan juga menjadi tantangan. Skala dan sudut pandang gambar yang diambil dari kamera kendaraan juga mempengaruhi hasil. Diperlukan dataset yang seimbang dan memiliki anotasi yang baik selain

itu, ada batasan komputasi pada perangkat edge yang memerlukan model yang lebih ringan atau teknik untuk mengurangi ukuran model (Wang et al., 2023).

2.3. Yolo (You Only Look Once)

2.3.1. Sejarah dan Evolusi YOLO (v1 sampai v8)

You Only Look Once (YOLO) dikenalkan sebagai pendekatan deteksi objek satu tahap yang mengubah masalah deteksi menjadi satu proses prediksi menyeluruh, memungkinkan inferensi yang sangat cepat menggunakan satu jaringan tunggal yang secara langsung memprediksi kotak pembatas dan kemungkinan kelas dari gambar keseluruhan. Sejak rilis awal, keluarga YOLO telah berkembang pesat dari segi arsitektur backbone, leher, dan strategi pelatihan untuk mencapai keseimbangan antara kecepatan dan ketepatan. Beberapa versi penting dalam perkembangan ini meliputi:

a. YOLOv1-v3: (Redmon et al., 2018) YOLOv1 memperkenalkan gagasan deteksi sebagai regresi end-to-end; YOLOv2 (YOLO9000) memiliki peningkatan akurasi dengan menggunakan teknik seperti anchor boxes dan batch-norm; YOLOv3 penambahan fitur *multi-scale prediction* dan *backbone* yang lebih baik sehingga mendapatkan akurasi meningkat tanpa adanya pengorbanan kecepatan.

b. YOLO v4 (2020): Diperkenalkan untuk mencapai kombinasi optimal antara kecepatan dan akurasi melalui integrasi sejumlah *bag-of-tricks* misalnya (CSP Darknet backbone, SPP, PANet, dan berbagai teknik augmentasi serta optimasi pelatihan) yang dibangun untuk efisiensi

pada GPU konvensional. YOLO v4 menjadi titik balik karena menitikberatkan praktik engineering yang pragmatis untuk deployment (Bochkovskiy et al., 2020).

c. **YOLO v5:** (Ultralytics Documentation, 2020) Walaupun YOLO v5 tidak diluncurkan sebagai sebuah publikasi resmi di arXiv, ia membawa dalam paketnya sebuah implementasi yang sederhana dan ringan. Terdapat juga variasi dari ukuran model (s,m,l,x) yang mempermudah penggunanya dalam menerapkan pada berbagai perangkat (cloud ↔ edge). Kode sumber dan dokumentasinya menjadi referensi yang cepat diadopsi oleh komunitas.

d. **YOLO v6 & YOLO v7 (2022):** Model ini menampilkan optimasi untuk aplikasi industri dan teknik pelatihan yang baru misalnya (*Bag-of-freebies, bag-of-specials, scaling rules*) sehingga meningkatkan mAP pada kecepatan inferensi kompetitif pada aplikasi *real-time* di berbagai hardware (C. Li et al., 2022).

e. **YOLO v8 (2023):** (Ultralytics Documentation, 2023) Peluncuran dari Ultralytics yang menggabungkan deteksi, segmentasi, dan klasifikasi dalam satu set API yang terintegrasi, serta perbaikan pada arsitektur neck dan proses pelatihan, dengan penekanan pada pengalaman pengguna dan integrasi dengan alat-alat penyebaran modern misalnya (ekspor ONNX atau perangkat). YOLO v8 menegaskan bahwa YOLO tidak hanya merupakan satu arsitektur, tetapi juga menjadi ekosistem model yang mendukung alur kerja dari awal hingga akhir.

Perkembangan di atas menunjukkan perubahan dari penelitian murni menuju rekayasa sistem yang menitik beratkan reproduksibilitas, kemudahan deploy dan penyesuaian model terhadap batasan perangkat (edge vs server).

2.3.2. Perbedaan YOLO dengan Model Deteksi Lain

Secara konseptual, detektor objek modern dibagi menjadi dua kelompok utama yaitu:

1. *Two-stage detectors* misalnya (R-CNN → Fast R-CNN → Faster R-CNN)
2. *One-stage detectors* misalnya (SDD, RetinaNet, YOLO)

Perbedaan pokoknya ialah:

a. Paradigma Pemrosesan: *Two-stage detectors* (Detektor bertahap dua) menghasilkan saran area dan selanjutnya melakukan klasifikasi atau penyempurnaan lokal pada setiap saran. Metode ini cenderung menghasilkan akurasi yang tinggi terutama untuk objek kecil dan yang saling tumpang tindih, tetapi dengan biaya latensi yang lebih tinggi. Di sisi lain, YOLO (tahap tunggal) mendefinisikan deteksi sebagai regresi langsung dari gambar keseluruhan, sehingga proses inferensi menjadi sangat cepat dan lebih sesuai untuk penggunaan waktu nyata (Zhang & Cloutier, 2022).

b. Kecepatan dan Akurasi: YOLO mengorbankan sebagian akurasi, khususnya dalam situasi objek kecil atau dengan tumpang tindih yang tinggi, untuk mendapatkan kecepatan inferensi yang lebih baik. Di sisi lain, model two-stage biasanya fokus pada peningkatan akurasi dengan

mengorbankan kecepatan. Namun, perkembangan YOLO dari versi 4 hingga 8 telah secara signifikan mengatasi kekurangan dalam akurasi melalui desain arsitektur yang lebih baik dan pendekatan pelatihan yang canggih. Penelitian terbaru dari tahun 2020 hingga 2025 menunjukkan bahwa versi terbaru YOLO dapat mencapai keseimbangan yang sangat bersaing dengan Faster R-CNN dan SSD dalam metrik mAP, sambil tetap mempertahankan tingkat frame yang jauh lebih tinggi (Shobaki & Milanova, 2025).

c. **Praktikalitas deployment:** Menurut (Ultralytics, 2023), YOLO memberikan komponen pengganti yang membuatnya gampang untuk beralih ke format ONNX/TensorRT/EdgeTPU, sehingga dapat dengan mudah digunakan pada perangkat IoT/embedded. Detektor dengan dua tahap biasanya sulit untuk dioptimalkan di perangkat edge tanpa mengorbankan akurasi secara signifikan.

2.3.3. Kelebihan dan Kekurangan YOLO pada Sistem Real-Time

Tinjauan pustaka dan kajian empiris terkini yang membahas seputar keunggulan dan kekurangan YOLO untuk aplikasi real-time meliputi:

a. **Inferensi cepat:** YOLO mengelola citra dalam satu evaluasi jaringan sehingga dapat menghasilkan frame rate yang cukup tinggi, kriteria utama untuk simulasi sistem peringatan waktu nyata di kendaraan. Kemampuan YOLO versi ringan dapat mencapai puluhan hingga ratusan FPS pada GPU dan puluhan FPS pada perangkat edge yang wajar (Bochkovskiy et al., 2020).

b. **Arsitektur yang mudah diskalakan:** Dikutip dari (Ultralytics, 2023) ukuran variasi (s, m, l, x) dan teknik pruning/klinis pengguna dapat memilih titik keseimbangan antara ukuran model, latensi, dan akurasi sesuai kemampuan perangkat. Hal ini dapat memudahkan penggabungan pada kendaraan roda dua dengan batasan daya dan komputasi.

c. **Ekosistem dan dukungan engineering:** Menurut (Ultralytics, 2023) Dokumentasi dan toolkit yang tersedia dapat mempercepat proses prototyping, pelatihan ulang menggunakan dataset lokal, serta mengekspor model untuk keperluan deployment. Meskipun YOLO banyak memiliki keunggulan yang praktis, penerapan YOLO untuk mendeteksi kerusakan jalan dan aplikasi real-time YOLO memiliki beberapa keterbatasan yang didokumentasikan di dalam beberapa kajian akademik, misalnya:

d. **Penurunan performa pada objek kecil:** Karena adanya desain grid atau sel prediksi dalam beberapa varian, deteksi objek yang sangat kecil atau yang letaknya sangat berdekatan bisa mengalami penurunan akurasi jika dibandingkan dengan detektor dua tahap (Zou et al., 2023).

e. **Sensitivitas terhadap domain shift:** Model YOLO dilatih menggunakan dataset umum (COCO, PASCAL) dapat mengalami penurunan performa ketika diaplikasikan ke domain spesifik misalnya (Tekstur aspal, variasi pencahayaan, jalan lokal) tanpa augmentasi yang benar. Oleh karena itu perlu adanya pengayaan data lokal dan transfer learning (Mittal, 2024)

f. **Kendala komputasi pada edge device sangat terbatas:** Walaupun ada varian yang lebih sederhana, biasanya model yang lebih besar diperlukan untuk mencapai tingkat akurasi yang tinggi. Ini menciptakan keseimbangan yang harus dipikirkan antara akurasi dan penggunaan daya atau waktu respons pada sepeda motor yang memiliki sumber daya terbatas. Sering kali, metode kuantisasi, pemotongan, dan pengoptimalan waktu jalan (seperti TensorRT dan ONNX) dibutuhkan (Bochkovskiy et al., 2020).

g. **Kesulitan dalam menangani kondisi lingkungan ekstrim:** Kondisi hujan, jalan basah, pantulan sinar, dan sudut kamera yang beragam dapat menurunkan nilai akurasi deteksi. Solusi yang diusulkan dalam literatur meliputi penggabungan multimodal (kamera dan sensor IMU/GPS) atau pipeline pre-processing khusus (Zhang & Cloutier, 2022).

2.4. IoT (Internet of Things)

2.4.1. Pengertian Internet of Things (IoT)

Menurut (Sorri et al., 2022) Internet of Things (IoT) merupakan konsep teknologi yang merujuk pada kumpulan objek fisik yang dilengkapi dengan sensor, aktuator, perangkat pemrosesan, dan kemampuan untuk berkomunikasi. Hal ini memungkinkan objek-objek tersebut untuk secara otomatis mengumpulkan, mengirim, dan menerima data guna mencapai tujuan tertentu tanpa memerlukan interaksi langsung dari manusia. Definisi yang lebih baru menyoroti pentingnya perangkat yang memiliki identitas,

pertukaran data, dan kemampuan untuk memberikan layanan berbasis data yang bermanfaat bagi pengguna atau sistem yang lebih besar. IoT bukan sekedar kumpulan perangkat melainkan sebuah ekosistem yang menggabungkan lapisan perangkat keras, lapisan middleware/komunikasi, dan lapisan aplikasi/presentasi.

2.4.2. Arsitektur IoT

Arsitektur IoT umumnya dijelaskan dalam model berlapis atau yang disebut *layer architecture* yang dapat memudahkan pemahaman terkait fungsi berbagai unsur dalam ekosistem. Ada beberapa lapisan yang sering digunakan menurut literatur:

a. **Lapisan Perangkat / Persepsi (Device / Perception Layer):** terdiri dari sensor, aktuator, dan perangkat endpoint misalnya (Kamera, modul GPS, sensor akselerasi) yang dimana berfungsi untuk mengumpulkan data dari lingkungan fisik (Rahaman, 2022).
b. **Lapisan Jaringan / Transport (Network / Transport Layer):** melakukan pengiriman data antar perangkat melalui platform edge server menggunakan protokol dan medium komunikasi (wifi, cellular, LoRa, MQTT, HTTP/REST) (Domínguez-Bolaño et al., 2022).
c. **Lapisan Middleware / Edge & Processing (Processing / Edge Layer):** melakukan proses awal seperti (filtering, agregasi, enkode), keputusan cepat di edge, atau pengiriman ke cloud untuk pemrosesan lebih lanjut (Tamboli et al., 2025).
d. **Lapisan Aplikasi / Presentasi (Application / Presentation Layer):** menyediakan tampilan akhir untuk pengguna seperti halaman

dashboard, notifikasi, API dan integrasi sistem manajemen, layanan akhir ini juga dapat menampilkan hasil analisis dalam bentuk yang dapat diubah sesuai keinginan (Rahaman, 2022).

Adapun beberapa studi lain menambahkan lapisan manajemen dan keamanan sebagai elemen transversal yang memantau beberapa aspek seperti autentikasi, otorisasi, enkripsi dan kebijakan privasi sepanjang aliran data, karena keamanan dan privasi pengguna merupakan tantangan kritis pada implementasi IoT (Tamboli et al., 2025).

2.4.3. Peran IoT dalam Sistem Monitoring dan Peringatan

IoT memainkan peran sentral dalam sistem monitoring dan peringatan (monitoring & alerting) melalui beberapa fungsi utama:

a. Akuisisi Data *Near real-time*: Sensor IoT memiliki aliran data kontinu yang dapat mendeteksi kondisi abnormal seperti getaran, deteksi kerusakan, atau perubahan lingkungan yang menandakan adanya risiko. Hal ini yang dapat memberikan peringatan otomatis berdasarkan nilai ambang batas atau model anomali (Witczak & Szymoniak, 2024).

b. Pemrosesan di Edge untuk Latensi Rendah: Dengan memindahkan sebagian inferensi atau logika deteksi ke edge, sistem dapat memberikan peringatan seketika tanpa harus menunggu round-trip ke cloud, fitur yang cocok pada pengaplikasian keselamatan seperti peringatan kondisi jalan pada kendaraan (Tamboli et al., 2025).

c. Kontekstualisasi & Integrasi Multimodal: IoT dapat menggabungkan data dari berbagai sumber sensor sehingga sistem monitoring dapat menurunkan false positive/negative dan memberikan

peringatan yang lebih akurat serta lokasi dan waktu yang nyata (Do**mínguez-Bolaño et al., 2022).**

d. **Otomatisasi Tindak lanjut (Actuation & Notification):** Setelah kondisi berisiko terdeteksi selanjutnya sistem IoT dapat mengirimkan pemberitahuan (push notif, SMS, atau indikator peringatan langsung pada perangkat pengendara) (Witczak & Szymoniak, 2024).

Ringkasnya, IoT memungkinkan transformasi dari monitoring pasif menjadi sistem peringatan proaktif yang bersifat terdistribusi, terhubung, dan dapat diskalakan untuk berbagai domain aplikasi termasuk untuk pemantauan jalan (Witczak & Szymoniak, 2024).

2.4.4. Tantangan dan Pertimbangan Praktis

Literatur mutakhir mengidentifikasi beberapa tantangan penting saat penerapan IoT untuk monitoring dan peringatan, antara lain: keterbatasan sumber daya perangkat (daya dan komputasi), masalah interoperabilitas protokol, keamanan dan privasi data, keandalan jaringan pada area terpencil, serta kebutuhan dataset dan model domain-spesifik untuk mengurangi kesalahan deteksi. Untuk mengatasi hal tersebut, peneliti merekomendasikan kombinasi desain arsitektur hybrid (edge + cloud), standar protokol terbuka, teknik enkripsi yang efisien, serta proses validasi lapangan yang ekstensif (Tamboli et al., 2025).

BAB III

METODOLOGI PENELITIAN

3.1. Jenis dan Pendekatan Penelitian

3.1.1. Jenis Penelitian

Penelitian yang penulis ambil berjudul *"SIMULASI SISTEM DETEKSI KERUSAKAN JALAN BERBASIS Yolo V8 MENGGUNAKAN INPUT VIDEO DENGAN RESPONS PENGEREMAN OTOMATIS BERBASIS IOT"* Penelitian ini termasuk kedalam Penelitian Rekayasa *(Engineering Research)* atau yang bisa disebut penelitian pengembangan sistem *(Research & Development/R&D)*.

Menurut (Hevner, 2020) Penelitian rekayasa bertujuan untuk merancang, membangun, dan menguji sebuah teknologi buatan berupa perangkat lunak, perangkat keras, maupun sistem yang terintegrasi dan langsung dapat digunakan lalu di evaluasi secara sistematis dalam konteks nyata.

3.1.2. Pendekatan Penelitian

Pendekatan penelitian yang penulis gunakan adalah pendekatan kuantitatif eksperimental, yang dikombinasikan dengan pendekatan rekayasa sistem berbasis eksperimen.

Pendekatan kuantitatif digunakan karena memiliki kinerja sistem yang dikembangkan dan diukur dengan parameter numerik dan matrik evaluasi objektif, seperti *precision, recall, mean Average Precision (mAP), frame per second (FPS),* dan waktu respon sistem. Data yang dihasilkan

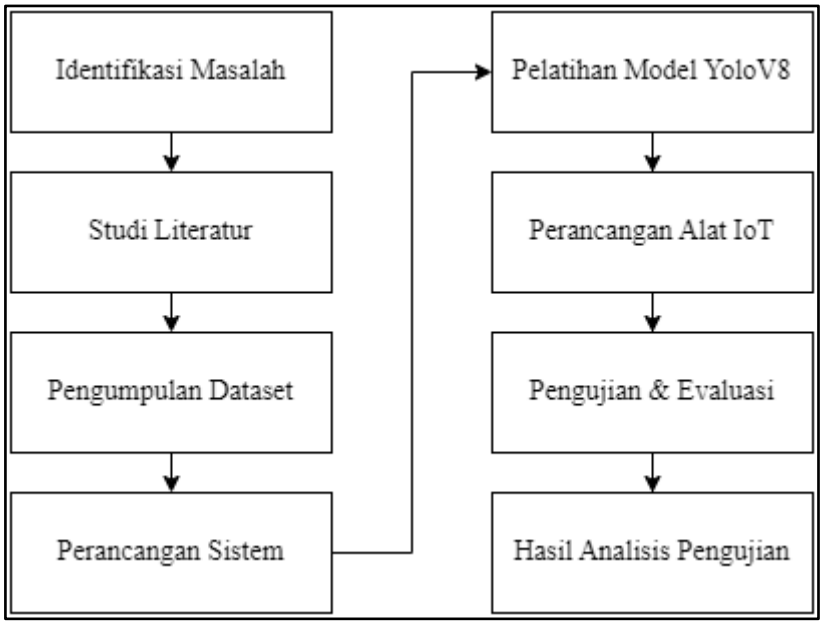
biasanya berupa angka-angka hasil pengujian model Yolov8 dan sistem IoT dalam berbagai skenario pengujian

Menurut (Sugiyono, 2020) “Pendekatan kuantitatif digunakan untuk menguji teori melalui pengukuran variabel-variabel penelitian dengan data numerik.”

Selain itu, penelitian ini menggunakan pendekatan eksperimental, karena melibatkan proses pelatihan model, pengujian sistem, serta evaluasi performa dengan perlakuan tertentu. Misalnya, variasi data, kondisi jalan, dan skenario pengujian *near real-time*.

3.2. Tahapan Penelitian

Penelitian ini dibuat secara sistematik melalui serangkaian tahapan yang saling berurutan mulai dari identifikasi masalah hingga hasil dari analisis, adapun diagram alur tahapan penelitian sebagai berikut:



Gambar 3. 1 Diagram Alur Tahapan Penelitian

3.2.1. Identifikasi Masalah

Tahap awal penelitian adalah dengan melakukan identifikasi permasalahan inti yang ingin diselesaikan, yaitu rendahnya kemampuan sistem konvensional untuk mendeteksi berbagai kerusakan jalan secara otomatis dan *near real-time* pada kendaraan roda dua. Identifikasi masalah ini penting agar penelitian terfokus pada gap yang jelas antara kebutuhan nyata dan kemampuan teknologi saat ini.

3.2.2. Studi Literatur

Selanjutnya menggunakan literatur review untuk memahami konsep-konsep utama seperti *computer vision, YOLO v8, IoT (Internet of Things),* serta metodologi evaluasi sistem deteksi objek.

3.2.3. Pengumpulan Dataset

Pengumpulan dataset tahapan paling penting untuk melatih sebuah model Deep Learning lalu menguji hasilnya setelah pengujian. Dataset yang diambil dalam penelitian ini bersumber dari repositori publik dan hasil pengambilan citra langsung menggunakan kamera yang ada pada kendaraan roda dua.

3.2.4. Perancangan Sistem

Perancangan sistem dilakukan dengan menyusun arsitektur perangkat keras dan perangkat lunak yang efektif untuk mendukung deteksi kerusakan jalan dan integrasi IoT.

3.2.5. Pelatihan Model Yolo v8

Pada tahap ini, model dari Yolo v8 dilatih menggunakan dataset yang telah diproses. Proses pelatihan meliputi optimasi parameter seperti

learning rate, batch size, serta penggunaan teknik augmentasi untuk meningkatkan performa model terhadap variasi kondisi citra jalan.

3.2.6. Implementasi IoT

Tahapan ini meliputi pengembangan dan integrasi komponen IoT seperti mikrokontroler, modul komunikasi, dan perangkat lunak penghubung (middleware), yang memungkinkan hasil deteksi model Yolo v8 dikirimkan secara *near real-time* ke unit antarmuka pengguna.

3.2.7. Pengujian dan Evaluasi

Pengujian dilakukan untuk menilai kinerja sistem secara keseluruhan, terutama kemampuan model dalam mendeteksi kerusakan jalan dan kecepatan respon sistem IoT dalam memberikan peringatan. Parameter evaluasi seperti *precision, recall, mean Average Precision (mAP), dan latency* digunakan untuk mengukur performa sistem secara kuantitatif.

3.2.8. Analisis Hasil

Analisis hasil dilakukan untuk memahami sejauh mana sistem yang dikembangkan memenuhi tujuan penelitian. Hasil pengujian dianalisis secara statistik dan dibandingkan dengan kriteria keberhasilan yang telah diterapkan.

Analisis ini juga mencakup interpretasi performa sistem dalam konteks aplikasi nyata dan rekomendasi perbaikan atau pengembangan lebih lanjut berdasarkan hasil empiris yang diperoleh.

3.3. Lokasi dan Waktu Penelitian

Lokasi pengambilan data dalam penelitian ini adalah jalan umum di sekitaran kota medan dari beberapa kecamatan terutama pada Kecamatan Medan Tembung, Kecamatan Medan Deli, Kecamatan Medan Labuhan, Kecamatan Medan Marelan, Kecamatan Medan Timur, Dan Kecamatan Medan Sunggal yang sengaja dipilih untuk mempertimbangkan faktor relevansi terhadap kondisi jalan yang sering dilalui kendaraan roda dua, dan tingkat variasi kerusakan jalan yang nyata di lapangan sehingga data yang diambil representatif terhadap fenomena yang diteliti.

Pemilihan lokasi jalan umum sebagai objek penelitian didasarkan pada prinsip bahwa lokasi yang harus dipilih harus mempunyai karakteristik lokasi yang sesuai dengan fokus penelitian agar memungkinkan pengumpulan data yang efektif dan efisien serta mampu meningkatkan kualitas hasil penelitian.

Menurut (Kustiawan et al., 2025) lokasi penelitian yang strategis dapat mempengaruhi kesesuaian data yang diperoleh dengan tujuan penelitian, sehingga pemilihannya harus mempertimbangkan kompatibilitas antara konteks penelitian dengan fenomena yang terjadi di lapangan.

Pemilihan lingkungan tertentu seperti ruas jalan yang mengalami kerusakan parah dan sedang dilakukan untuk memastikan bahwa sistem deteksi yang dikembangkan dapat diuji pada kondisi yang bervariasi sehingga hasil evaluasi model YOLO v8 dan IoT mencerminkan performa pada konteks nyata.

Sementara itu, rentang waktu penelitian ditentukan selama 4 minggu dimulai dari tanggal (20 november – 20 desember 2025) agar proses pengambilan data, pelatihan model, implementasi IoT, dan evaluasi sistem

dapat dilaksanakan secara bertahap dengan waktu yang memadai untuk melakukan perbaikan metode apabila diperlukan.

Waktu pengambilan data penelitian juga harus memperhatikan variasi kondisi lingkungan seperti cuaca dan tingkat lalu lintas, karena kondisi waktu sangat mempengaruhi kualitas citra dan deteksi kerusakan jalan, sehingga rentang waktu yang dipilih direncanakan mencakup waktu dengan variasi kondisi nyata.

Dengan demikian, lokasi pengambilan data yang dipilih secara strategis dan rentang waktu penelitian yang memadai menjadi elemen penting dalam metodologi ini untuk memastikan bahwa data yang terkumpul benar-benar mencerminkan fenomena kerusakan jalan yang sesungguhnya serta memberikan basis yang kuat bagi analisis performa sistem deteksi dan peringatan otomatis berbasis YOLO v8 dan IoT.

3.4. Sumber dan Jenis Data

Dalam penelitian ini data merupakan komponen paling penting yang menjadi dasar dalam proses pelatihan model Yolo v8 dan pengujian sistem IoT sehingga harus ditentukan sumber dan jenisnya secara jelas, sistematis, dan sesuai kaidah ilmu metodologi penelitian.

Sumber data penelitian dapat dibedakan menjadi dua jenis utama, yaitu data primer dan data sekunder, yang masing-masing memiliki karakteristik, fungsi, serta kontribusi yang berbeda terhadap penelitian ini.

Data primer data yang didapatkan secara langsung oleh peneliti dari sumber atau objek penelitian tanpa perantara pihak manapun, sehingga data ini bersifat

otentik dan khusus dikumpulkan untuk menjawab tujuan penelitian secara spesifik. Dalam hal ini, data primer mencakup citra atau video kerusakan jalan yang diambil langsung melalui kamera yang terpasang langsung pada kendaraan roda dua selama pengambilan data lapangan.

Sementara itu, data sekunder data yang telah tersedia sebelumnya, dikumpulkan dan dipublikasikan oleh pihak lain, kemudian dipakai kembali oleh peneliti lain. Data sekunder berfungsi sebagai pelengkap atau pembanding terhadap data primer serta mendukung proses analisis dan peninjauan teoritis.

Dalam penelitian ini, data sekunder yang mencakup dataset kerusakan jalan yang sudah tersedia di repository publik atau yang sudah disediakan pada lomba kompetisi deteksi objek (dataset) misalnya dataset *road damage* yang tersedia secara gratis di publikasi open search seperti kaggle, dan UCI Machine Learning.

Jenis data dalam penelitian ini dikategorikan berdasarkan cara perolehannya dan bentuk informasinya yang selaras dengan tujuan penelitian rekayasa sistem. Data primer berbentuk citra visual kondisi jalan yang diambil secara langsung melalui perangkat perekam pada kendaraan, sementara data sekunder berbentuk dataset citra digital, dan dokumentasi kerusakan jalan.

Data primer bersifat unik, karena dikumpulkan secara spesifik untuk penelitian ini, serta mencerminkan kondisi nyata di lokasi penelitian sehingga dapat digunakan untuk training maupun testing model Yolo v8.

Sedangkan data sekunder akan digunakan sebagai komplementer untuk pengayaan teori, pendalaman hasil observasi, serta validasi silang terhadap

performa model yang dibangun, sehingga pengambilan keputusan dan kesimpulan akhir penelitian memiliki pondasi empiris dan teoritis yang kuat.

Dengan demikian, kombinasi antara data primer dan data sekunder yang akan digunakan dalam penelitian ini dipilih sebagai sumber data karena keduanya saling melengkapi.

3.5. Teknik Pengumpulan Data

Teknik pengumpulan data merupakan tahapan penting dalam penelitian ini karena menentukan kualitas data yang akan digunakan dalam proses pelatihan model Yolo v8 serta nantinya untuk melakukan evaluasi sistem deteksi kerusakan jalan berbasis IoT.

Dalam penelitian ini, teknik pengumpulan data dilakukan dengan mengkombinasikan observasi lapangan, perekaman citra digital, anotasi data, dan studi dokumentasi, sebagaimana direkomendasikan dalam penelitian berbasis *computer vision* dan *machine learning.*

3.5.1. Observasi Lapangan

Observasi lapangan dilakukan dengan mengamati secara langsung kondisi jalan di lokasi penelitian untuk memperoleh pemahaman empiris mengenai jenis, tingkat, dan karakteristik kerusakan jalan yang sering dilalui kendaraan roda dua.

3.5.2. Pengambilan Citra dan Video Jalan

Pengambilan data visual dilakukan dengan merekam citra dan video kondisi jalan menggunakan kamera yang dipasang pada kendaraan roda dua

sebagai sumber data primer untuk pelatihan dan pengujian model YOLO v8.

Penggunaan kamera bergerak pada kendaraan memungkinkan pengambilan data secara dinamis dan *near real-time* sehingga sesuai dengan karakteristik sistem deteksi kerusakan jalan berbasis kendaraan.

Teknik ini banyak digunakan dalam penelitian deteksi objek jalan karena mampu menangkap variasi kondisi pencahayaan, sudut pandang, dan tekstur permukaan jalan yang beragam.

3.5.3. Anotasi Dataset

Setelah data citra dan video diperoleh selanjutnya adalah melakukan proses anotasi dataset dengan memberikan label dan *bounding box* pada setiap jenis kerusakan jalan yang teridentifikasi, sehingga data siap digunakan dalam proses pelatihan model Yolo v8.

Menurut (Bochkovskiy et al., 2020) kualitas anotasi sangat mempengaruhi performa model deteksi objek, karena anotasi yang tidak akurat dapat menurunkan nilai *precision* dan *recall* model.

3.5.4. Studi Dokumentasi dan Dataset Sekunder

Selain menggunakan data primer, penelitian ini juga menggunakan data sekunder yang diperoleh melalui studi dokumentasi berupa dataset kerusakan jalan dari repositori publik, artikel ilmiah, serta laporan penelitian sebelumnya yang relevan.

Data sekunder digunakan untuk memperkaya variasi data latih dan sebagai pembanding terhadap data primer guna meningkatkan kemampuan generalisasi model.

Penggabungan data primer dan sekunder dalam penelitian berbasis *machine learning* dapat mengurangi risiko overfitting.

3.5.5. Dokumentasi Implementasi IoT

Teknik pengumpulan data juga mencakup dokumentasi implementasi sistem IoT, seperti konfigurasi perangkat keras, arsitektur komunikasi, dan log pengiriman data hasil deteksi, yang digunakan sebagai bahan analisis performa sistem secara keseluruhan.

Dokumentasi teknis ini penting untuk memastikan bahwa sistem yang dibangun dapat direplikasi dan dievaluasi secara ilmiah sesuai prinsip penelitian rekayasa sistem. Dengan demikian, penerapan teknik pengumpulan data yang terstruktur melalui observasi lapangan, pengambilan citra, anotasi dataset, studi dokumentasi, dan pencatatan implementasi IoT diharapkan mampu menghasilkan data yang valid, reliable, dan relevan untuk mendukung tujuan penelitian ini.

3.6. Arsitektur Perancangan Sistem

Penelitian ini menerapkan pendekatan perancangan prototype sistem, yaitu membangun model awal sistem deteksi kerusakan jalan yang berfungsi sebagai representasi implementasi nyata untuk keperluan pengujian, evaluasi, dan pengembangan lebih lanjut. Pendekatan prototype dipilih karena memungkinkan peneliti memvalidasi konsep sistem sebelum diterapkan secara langsung pada lingkungan operasional sebenarnya.

Pada penelitian ini, sistem dirancang untuk memproses data video rekaman jalan sebagai sumber input, bukan citra dari kamera real-time. Video digunakan

sebagai simulasi kondisi nyata jalan yang dilewati kendaraan roda dua, sehingga proses deteksi dapat diuji secara terkontrol dan terukur.

3.6.1. Arsitektur Sistem Secara Umum

Arsitektur sistem yang diusulkan pada penelitian ini dirancang dalam bentuk prototype sistem deteksi kerusakan jalan berbasis video, yang terdiri dari beberapa lapisan utama, yaitu lapisan pengambilan data, lapisan pemrosesan dan deteksi, lapisan komunikasi IoT, serta lapisan aplikasi untuk penyajian informasi kepada pengguna.

Lapisan pengambilan data tidak menggunakan kamera secara langsung, melainkan memanfaatkan video rekaman kondisi jalan yang diperoleh dari dataset publik maupun hasil perekaman sebelumnya. Video ini digunakan sebagai media simulasi kondisi nyata kendaraan roda dua yang melintasi jalan, sehingga sistem dapat diuji secara terkontrol dan berulang.

Lapisan pemrosesan dan deteksi menggunakan model YOLO v8 yang telah dilatih dengan dataset kerusakan jalan untuk mengekstraksi fitur visual dari setiap frame video serta melakukan proses klasifikasi dan deteksi objek secara cepat dalam satu tahap inferensi, sesuai dengan karakteristik metode single-stage object detection.

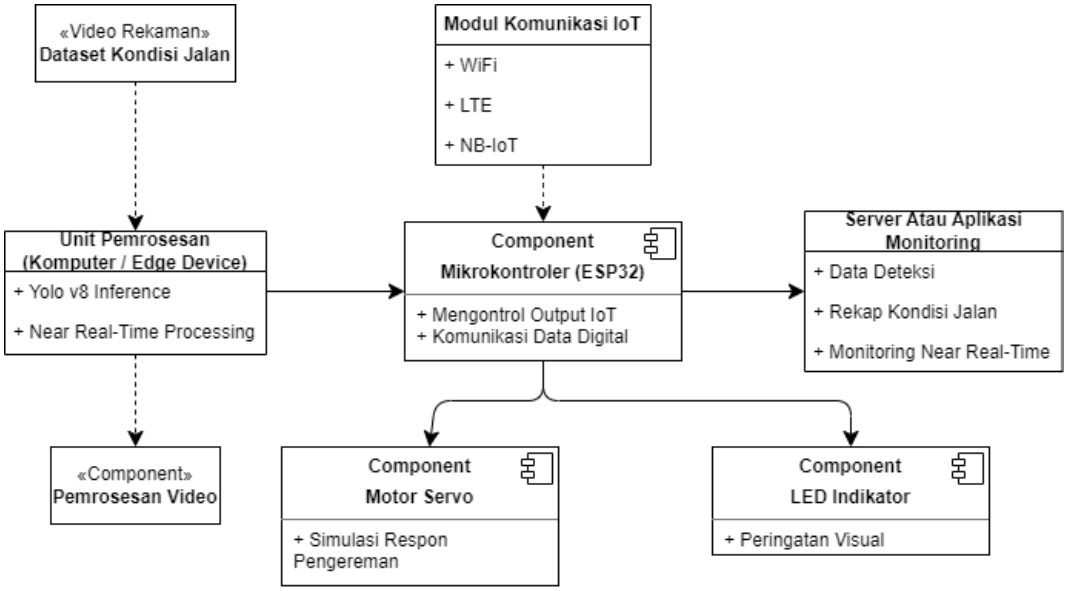
Selanjutnya, lapisan komunikasi IoT berperan dalam mentransmisikan hasil deteksi dari unit pemrosesan ke sistem IoT dalam bentuk data digital. Proses komunikasi ini memanfaatkan protokol seperti MQTT atau HTTP berbasis arsitektur REST, sehingga data hasil deteksi dapat dipantau dan diproses lebih lanjut secara *near real-time*.

Lapisan aplikasi berperan sebagai antarmuka pengguna yang menampilkan hasil deteksi kerusakan jalan dalam bentuk visualisasi bounding box pada video, informasi jenis kerusakan, serta rekapitulasi jumlah kerusakan jalan berdasarkan kategori yang terdeteksi selama proses pemutaran video berlangsung.

Dengan arsitektur berbasis prototype ini, sistem yang dikembangkan mampu merepresentasikan alur kerja sistem nyata secara simulatif, sehingga memudahkan proses pengujian, evaluasi performa model, serta analisis integrasi antara YOLO v8 dan perangkat IoT sebelum sistem diimplementasikan secara langsung pada kendaraan roda dua.

3.6.2. Arsitektur Komponen Perangkat Keras *(Architecture Hardware)*

Komponen perangkat keras dalam perancangan sistem prototype ini berfungsi sebagai media pemrosesan, komunikasi, dan pemberian peringatan, yang meliputi unit pemrosesan (komputer atau perangkat edge), perangkat mikrokontroler (MCU), modul komunikasi IoT, serta perangkat output peringatan. Berikut diagram arsitekturnya:



Gambar 3. 2 Diagram Arsitektur Komponen Perangkat Keras

"Garis utuh menunjukkan jalur komunikasi dan aliran data utama antar komponen sistem, sedangkan garis putus-putus menunjukkan hubungan konseptual atau proses pendukung, seperti sumber data awal dan koneksi simulatif pada tahap pengujian prototype."

Pada sistem prototype ini, kamera digital tidak digunakan sebagai sumber input langsung, karena data citra diperoleh dari video rekaman jalan. Oleh karena itu, kamera hanya berperan sebagai perangkat perekam pada tahap pengumpulan data sebelumnya dan tidak terlibat dalam proses inferensi sistem secara langsung.

Unit pemrosesan, berupa komputer atau perangkat edge, berfungsi untuk mengeksekusi model YOLO v8 yang telah dilatih guna mendeteksi kerusakan jalan dari frame video secara otomatis dengan latensi yang relatif rendah. Unit ini menjadi pusat komputasi dalam sistem prototype.

Mikrokontroler (MCU) digunakan sebagai pengendali sistem IoT yang menerima hasil deteksi dari unit pemrosesan dalam bentuk data digital,

kemudian memproses data tersebut untuk mengaktifkan sistem peringatan otomatis.

Modul komunikasi IoT, seperti Wi-Fi, LTE, atau NB-IoT, berperan dalam mengirimkan data hasil deteksi ke server atau aplikasi monitoring secara *near real-time*, sehingga informasi mengenai kondisi kerusakan jalan dapat dipantau oleh pengguna atau pihak terkait.

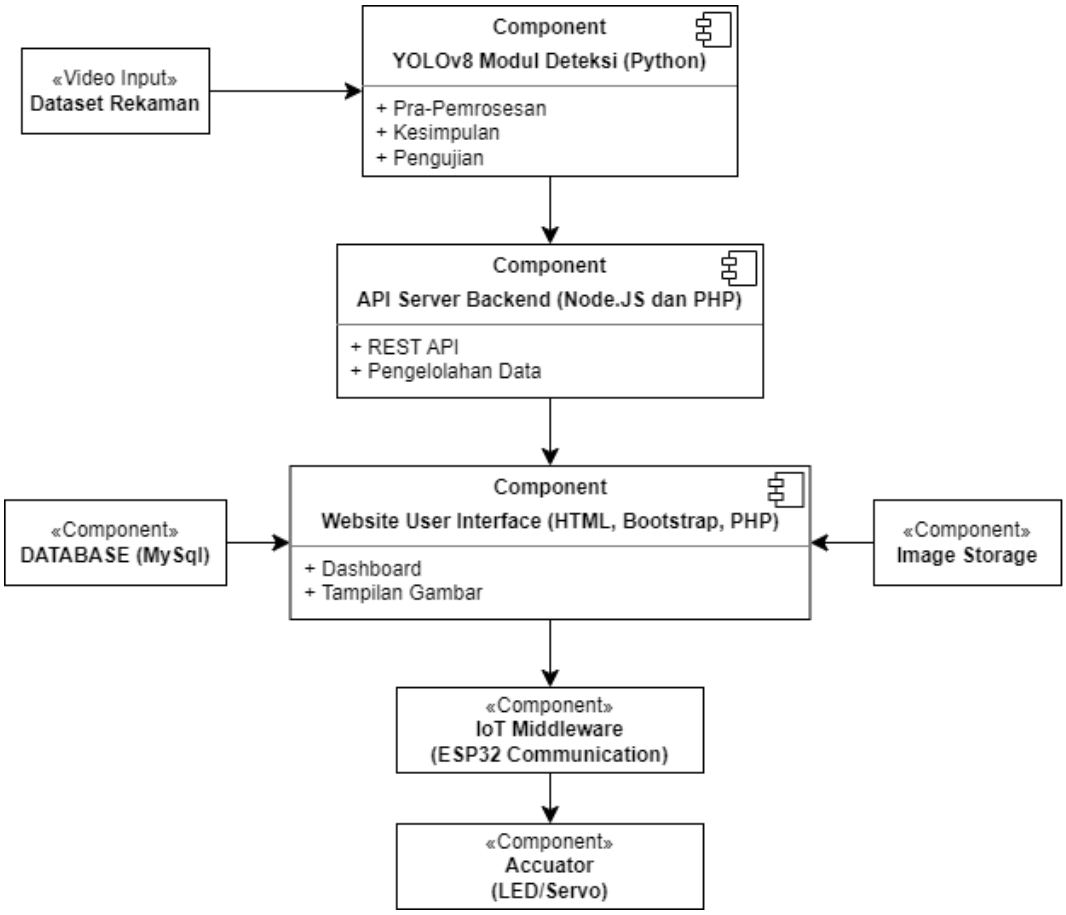
Dengan konfigurasi perangkat keras tersebut, prototype sistem mampu mensimulasikan proses kerja sistem nyata secara terintegrasi, meskipun input citra diperoleh dari video rekaman, sehingga tetap mendukung tujuan penelitian dalam mengevaluasi kinerja integrasi YOLO v8 dan IoT. Adapun rincian perangkat keras yang digunakan:

Tabel 3. 1 Perangkat Keras Untuk Percobaan Simulasi

Komponen	Fungsi
Laptop	YOLO + Server
ESP32	IoT Controller
Sensor Buzzer	Peringatan
Servo/DC Motor	Simulasi Rem
BreadBoard	Prototype
Kabel Jumper	Koneksi
Power Supply	Daya

3.6.4. Arsitektur Komponen Perangkat Lunak *(Architecture Software)*

Arsitektur komponen perangkat lunak pada penelitian ini dirancang untuk mendukung proses simulasi deteksi kerusakan jalan berbasis YOLO v8, pengiriman hasil deteksi ke server, serta penyajian informasi kepada pengguna melalui antarmuka berbasis web. Perangkat lunak disusun dalam beberapa komponen utama yang saling terintegrasi, yaitu modul deteksi YOLO v8, *server backend, middleware* komunikasi IoT, serta aplikasi website sebagai user interface (UI).



Gambar 3. 3 Diagram Arsitektur Komponen Perangkat Lunak

Modul deteksi YOLO v8 diimplementasikan menggunakan bahasa pemrograman Python dengan dukungan pustaka *deep learning* seperti PyTorch dan framework ultralytics Yolo v8 untuk melakukan proses

inferensi terhadap frame video. Modul ini bertanggung jawab dalam mengeksekusi model yang telah dilatih, mengekstrak fitur viusal dari setiap frame, serta menghasilkan keluaran berupa bounding box, klasifikasi jenis kerusakan jalan, dan nilai *confidence score.* Hasil inferensi kemudian dikemas dalam format data terstruktur untuk dikirimkan ke server backend.

Server backend berfungsi sebagai pusat pengelolaan data hasil deteksi serta sebagai penghubung antara sistem deteksi dan aplikasi antarmuka pengguna berbasis web. Server diimplementasikan menggunakan framework web Flask yang menyediakan layanan *Application Programming Interface* (API) berbasis HTTP dengan pendekatan REST (*RESTful API*).

API ini bertugas menerima data hasil deteksi dari modul YOLOv8, menyimpan metadata deteksi ke dalam basis data (*database*), serta mengelola penyimpanan file gambar hasil deteksi pada direktori server.

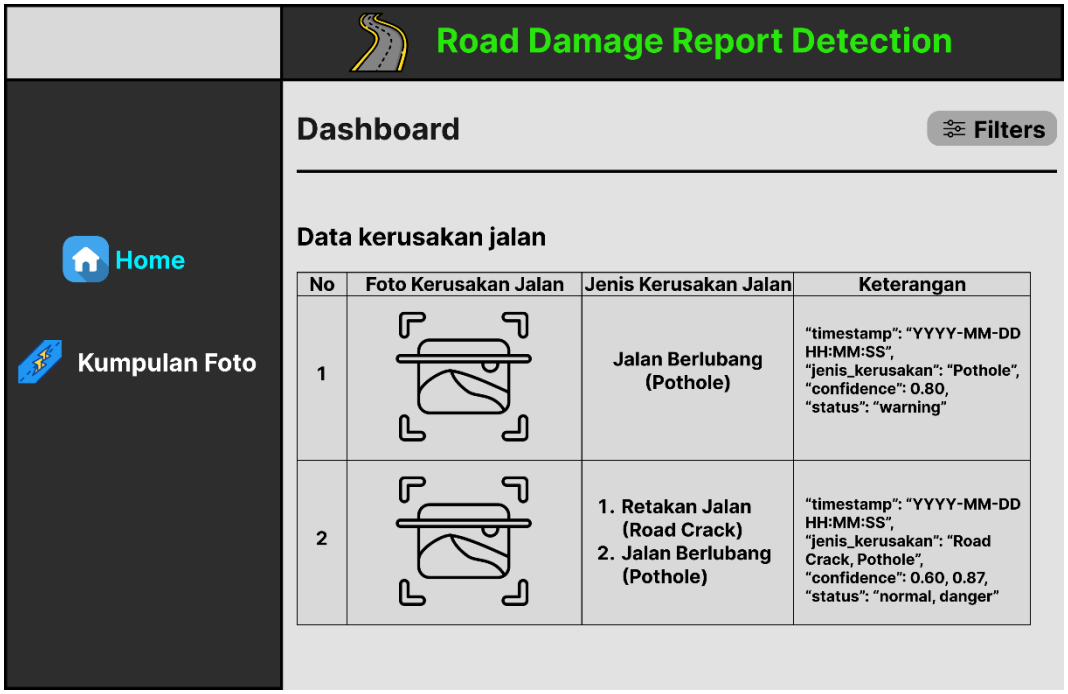
Proses pengiriman data dari sistem deteksi ke server dilakukan melalui mekanisme *HTTP request* ke endpoint API. Setiap kali kerusakan jalan terdeteksi, sistem akan mengirimkan informasi berupa jenis kerusakan, nilai *confidence*, waktu deteksi, serta file gambar hasil deteksi. Data tersebut kemudian diproses oleh server backend dan disimpan secara terstruktur sehingga dapat diakses dan ditampilkan oleh aplikasi web secara terpusat.

Middleware komunikasi IoT berperan sebagai lapisan perantara yang mengatur format data, protokol komunikasi, serta sinkronisasi antara sistem deteksi berbasis perangkat lunak dan perangkat IoT. Middleware ini

memastikan bahwa hasil deteksi dapat diteruskan secara konsisten baik ke perangkat IoT untuk memicu sistem peringatan maupun ke server backend untuk keperluan visualisasi pada website.

Aplikasi website berfungsi sebagai user interface (UI) yang digunakan oleh pengguna untuk memantau hasil deteksi kerusakan jalan. Website dikembangkan menggunakan teknologi web dan framework frontend seperti bootstrapdan terhubung langsung dengan server backend melalui API. Website secara periodeik mengambil data hasil deteksi dari server dan menampilkannya dalam bentuk gambar hasil deteksi, jenis kerusakan jalan, nilai confidence, serta informasi waktu deteksi.

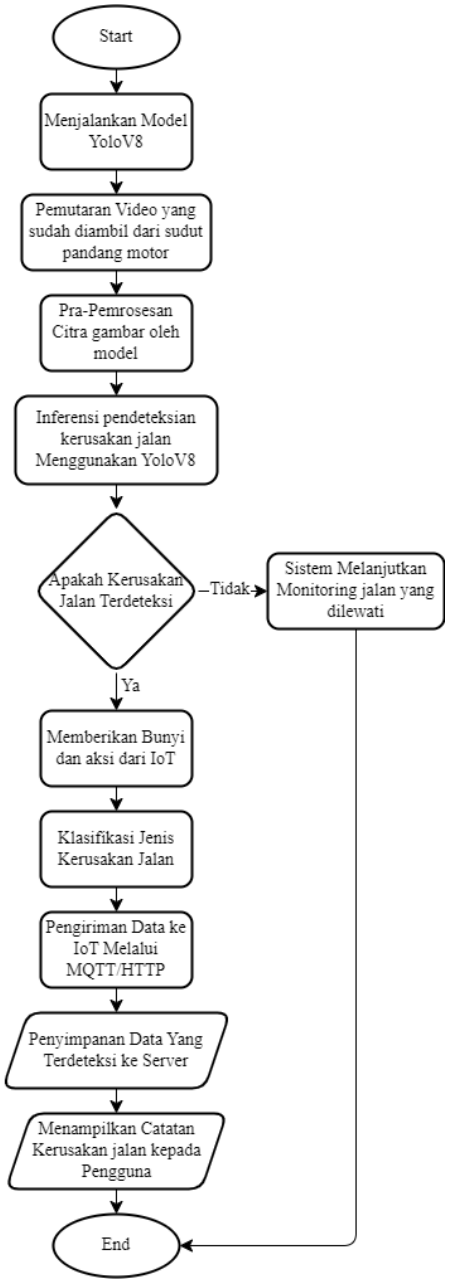
Selain menampilkan informasi visual, aplikasi website juga menyediakan fitur rekapitulasi data, seperti jumlah kerusakan jalan berdasarkan kategori tertentu dan riwayat hasil deteksi selama proses simulasi berlangsung. Dengan demikian, website bukan hanya sekedar memonitoring saja tetapi juga sebagai saran evaluasi performa sistem deteksi.



Gambar 3. 4 Rancangan Website Laporan Kerusakan Jalan

3.6.5. Alur Kerja Sistem

Secara logis alur kerja sistem pada penelitian ini dimulai dari pengambilan citra jalan, dilanjutkan dengan pemrosesan informasi oleh model YOLO v8, kemudian hasil deteksi dikirim melalui modul IoT ke server atau aplikasi pengguna, dan diakhiri dengan pemberian peringatan otomatis kepada pengguna jika kerusakan ditemukan, berikut alur kerja menggunakan flowchart:



Gambar 3. 5 Flowchart Cara Kerja Sistem

3.7. Metode Deteksi Kerusakan Jalan Menggunakan YOLO v8

Metode deteksi kerusakan jalan pada penelitian ini menggunakan algoritma YOLO v8 (You Only Look Once versi 8) yang merupakan model objek deteksi sendiri berbasis *deep learning* dan dirancang untuk kebutuhan deteksi objek secara tepat dan akurat pada sistem real-time.

YOLO v8 dipilih karena mampu melakukan proses deteksi dan klasifikasi objek dalam satu tahap inferensi, sehingga sangat sesuai untuk aplikasi kendaraan bergerak yang mampu membutuhkan respons cepat dan latensi yang rendah, seperti deteksi kerusakan jalan pada kendaraan roda dua.

3.7.1. Konfigurasi Model YOLO v8

Pada penelitian ini digunakan YOLO v8-s sebagai konfigurasi model karena memiliki keseimbangan antara akurasi dan kecepatan inferensi, sehingga cocok untuk implementasi pada perangkat edge atau sistem IoT berbasis kendaraan

Contoh Konfigurasi model:

1. Model: YOLOv8-s
2. Input Size: 640 x 640 piksel
3. Backbone: CSPDarknet
4. Neck: PAN-FPN
5. Head: Anchor-free detection head

Konfigurasi tersebut memungkinkan model mengekstraksi fitur visual kerusakan jalan seperti retakan *(crack)*, lubang *(pothole)*, dan permukaan rusak dengan kompleksitas komputasi yang masih dapat ditangani secara *near real-time*

3.7.2. Parameter Pelatihan Model

Parameter pelatihan sangat mempengaruhi performa akhir model YOLO v8 dalam mendeteksi kerusakan jalan.

Contoh parameter pelatihan yang digunakan:

Tabel 3. 2 Standar Parameter Pengujian

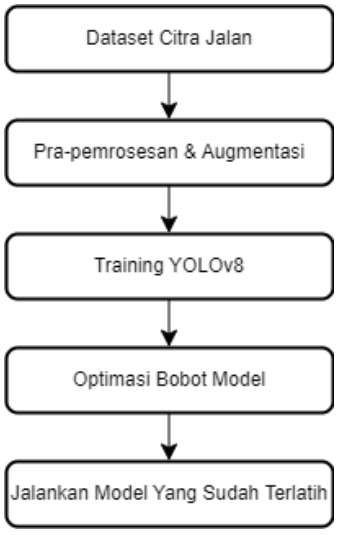
Parameter	Nilai Contoh	Keterangan
Epoch	80	Jumlah Iterasi Pelatihan
Batch Size	4	Jumlah Data Per Iterasi
Learning Rate	0.003	Kecepatan Pembaruan Bobot
Optimizer	Auto	Optimasi Konvergensi
Loss Function	CloU + BCE	Akurasi Bounding Box

Jumlah epoch 80 dipilih untuk memastikan model memiliki waktu belajar yang cukup terhadap variasi kerusakan jalan, sedangkan batch size 4 disesuaikan untuk pembacaan akurasi gambar dilatih.

3.7.3. Proses Training dan Inference YOLO v8

a. Proses Training

Proses pelatihan dilakukan menggunakan dataset citra kerusakan jalan yang telah dianotasi dalam format YOLO.

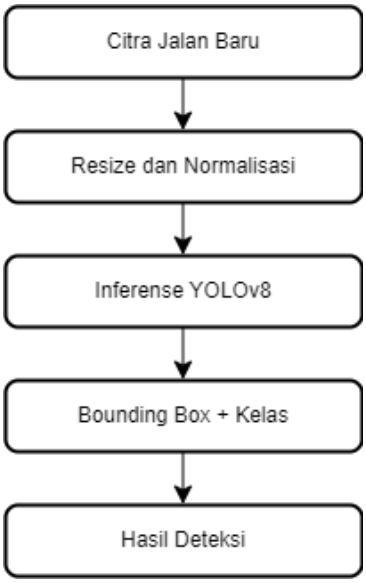


Gambar 3. 6 Proses Pelatihan

Pada tahap ini, model mempelajari pola visual kerusakan jalan melalui proses *forward propagation* dan *backpropagation* hingga mencapai nilai loss yang stabil.

b. Proses Inference

Proses *inference* dilakukan setelah model selesai dilatih, yaitu ketika model digunakan untuk mendeteksi kerusakan jalan pada citra atau video baru.



Gambar 3. 7 Proses Kesimpulan/Uji

Yolo v8 menghasilkan bounding box, kelas kerusakan, dan nilai confidence dalam satu kali proses inferensi, sehingga sangat efisien untuk aplikasi *near real-time*

3.8. Integrasi YOLOv8 dengan Perangkat IoT

Integrasi antara algoritma YOLO v8 dan Perangkat IoT pada penelitian ini bertujuan untuk membangun sistem deteksi kerusakan jalan yang tidak hanya mampu mengenali objek secara visual, tetapi juga dapat mengirimkan hasil

deteksi secara real-time dan memberikan peringatan otomatis kepada pengguna kendaraan. Integrasi ini memungkinkan sistem bekerja secara *end-to-end*, mulai dari akuisisi citra hingga penyampaian informasi kerusakan jalan dalam bentuk peringatan visual, audio, dan notifikasi digital.

Secara konseptual, YOLO v8 berperan sebagai modul analisis visual, sedangkan IoT berfungsi sebagai media komunikasi dan aktuator peringatan, sehingga keduanya membentuk sistem *intelligent monitoring and warning system.*

3.8.1. Mekanisme Pengiriman Hasil Deteksi

Hasil deteksi kerusakan jalan yang diperoleh dari proses inferensi YOLO v8 dikirimkan ke sistem IoT melalui jaringan nirkabel menggunakan protokol komunikasi ringan seperti MQTT atau HTTP REST API. Data yang dikirimkan bersifat ringkas agar sesuai dengan karakteristik sistem real-time dan keterbatasan bandwidth perangkat IoT.

```
{
  “timestamp”: “2025-01-15 10:32:45”,
  “jenis_kerusakan”: “pothole”,
  “confidence”: 0.87,
  “status”: “warning”
}
```

Data tersebut dikirim dari unit pemrosesan (edge device) menuju mikrokontroler IoT yang terhubung dengan modul output peringatan. Mekanisme ini sesuai dengan pendekatan *edge-IoT integration* yang direkomendasikan dalam sistem monitoring real-time.

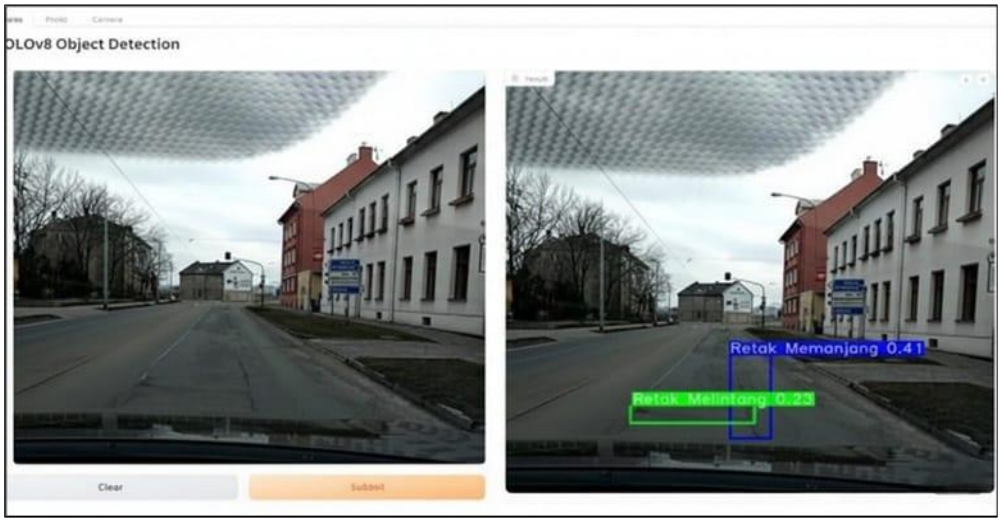
3.8.2. Sistem Peringatan Otomatis

Sistem peringatan otomatis dirancang untuk memberikan informasi kerusakan jalan kepada pengendara secara langsung, jelas, dan mudah

dipahami. Pada penelitian ini, sistem peringatan dibagi menjadi tiga bentuk utama, yaitu visual, audio, dan notifikasi digital.

a. Peringatan Visual

Peringatan visual ditampilkan melalui overlay bounding box dan label kelas pada tampilan layar Laptop. Informasi yang ditampilkan meliputi jenis kerusakan jalan dan tingkat kepercayaan *(confidence score)*.



Gambar 3. 8 Contoh Deteksi Gambar Kerusakan Jalan

Peringatan visual ini membantu pengendara mengenali jenis kerusakan jalan secara intuitif tanpa memerlukan interpretasi teknis yang kompleks

b. Peringatan Audio

Peringatan audio dihasilkan melalui buzzer yang terhubung ke mikrokontroler IoT. Buzzer akan aktif ketika nilai *confidence* hasil deteksi melebihi ambang batas tertentu, misalnya >= 70%

Peringatan Audio sangat penting dalam kondisi ketika pengendara tidak sempat untuk melihat jalan, sehingga meningkatkan aspek keselamatan berkendara (safety-critical system)

c. Notifikasi Digital

Notifikasi digital dikirimkan ke perangkat pengguna atau server monitoring melalui aplikasi berbasis web atau mobile. Informasi yang akan dikirimkan dapat berupa teks, ikon peringatan, atau notifikasi push.

Notifikasi digital memungkinkan data deteksi disimpan dan dianalisis lebih lanjut oleh pihak terkait seperti dinas perhubungan atau pengelola infrastruktur jalan.

BAB IV

HASIL DAN PEMBAHASAN

4.1. Implementasi Sistem

4.1.1. Implementasi YoloV8

Pada penelitian ini, implementasi model YOLOv8 dilakukan sebagai komponen utama dalam sistem deteksi kerusakan jalan berbasis pengolahan citra digital. Model ini digunakan untuk mengidentifikasi dan mengklasifikasikan berbagai jenis kerusakan jalan secara otomatis berdasarkan input berupa video rekaman kondisi jalan.

Hasil implementasi menunjukkan bahwa model YOLOv8 mampu melakukan deteksi kerusakan jalan secara real-time pada video dengan kecepatan pemrosesan yang cukup tinggi. Model dapat mengenali objek kerusakan dengan cukup baik pada berbagai kondisi jalan, meskipun performa masih dipengaruhi oleh faktor seperti pencahayaan, sudut pengambilan gambar, dan kualitas dataset.



Gambar 4. 1 Implementasi Model YoloV8 Deteksi Jalan

4.1.2. Implementasi Sistem IoT

Implementasi sistem IoT dalam penelitian ini berfungsi sebagai komponen yang menghubungkn hasil deteksi kerusakan jalan dengan mekanisme respon otomatis berupa peringatan dan simulasi pengereman. Sistem IoT dirancang untuk menerima data hasil deteksi dari model YOLOv8, memproses data tersebut, dan menghasilkan tindakan secara real-time melalui perangkat keras yang terintegrasi. Adapun sistem IoT pada penelitian ini terdiri dari beberapa komponen utama, yaitu:

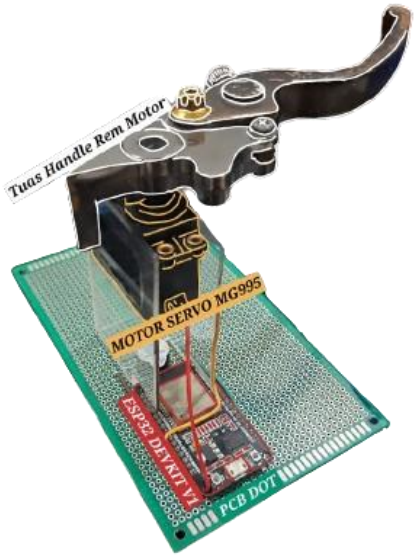
a. Mikrokontroler (ESP32) sebagai pusat kendali sistem.
b. Modul komunikasi berbasis jaringan (Wifi).
c. Aktuator, meliputi:
    - Buzzer sebagai indikator peringatan
    - Servo motor sebagai simulasi pengereman
d. Middleware komunikasi (MQTT/API) sebagai penghubung antara sistem deteksi dan perangkat IoT.

Arsitektur ini memungkinkan sistem bekerja secara terdistribusi, di mana proses deteksi dilakukan pada unit pemrosesan (komputer), sedangkan eksekusi respon dilakukan oleh perangkat IoT.

Adapun alur secara keseluruhan, alur kerja sistem IoT dalam penelitian ini dapat dijelaskan sebagai berikut:

a. Model YOLOv8 mendeteksi kerusakan jalan dari video
b. Hasil deteksi dikirim ke server atau broker dalam bentuk data digital
c. ESP32 menerima data melalui jaringan
d. Data diproses untuk menentukan tingkat bahaya
e. Sistem mengaktifkan aktuator sesuai kondisi.

Alur ini menunjukkan bahwa sistem IoT berperan sebagai eksekutor yang menerjemahkan hasil analisis visual menjadi aksi fisik secara otomatis.



Gambar 4. 2 Hasil Implementasi Alat IoT

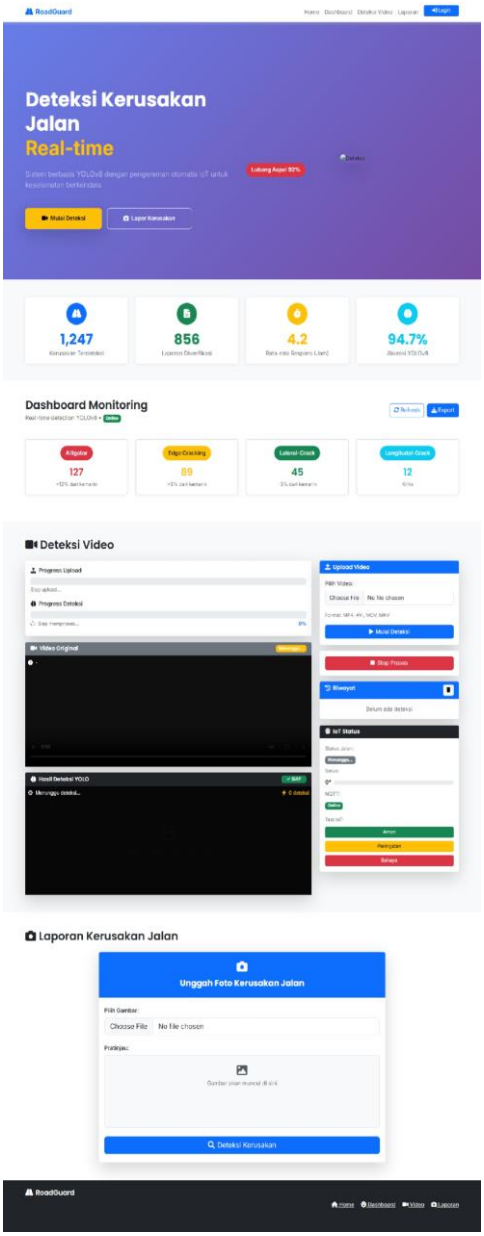
Hasil implementasi menunjukkan bahwa sistem IoT mampu menerima data dari sistem deteksi dan memberikan respon secara cepat

dengan latensi yang relatif rendah. Perangkat dapat bekerja secara sinkron dengan sistem deteksi, sehingga menghasilkan mekanisme peringatan yang responsif terhadap kondisi jalan yang terdeteksi.

Namun demikian, performa sistem masih dipengaruhi oleh kestabilan jaringan dan kecepatan komunikasi data, yang menjadi faktor paling penting dalam sistem berbasis IoT.

4.1.3. Implementasi Website / Monitoring

Implementasi website dalam penelitian ini berfungsi sebagai antarmuka pengguna yang digunakan untuk menampilkan, memantau, dan merekap hasil deteksi kerusakan jalan yang dihasilkan oleh model YOLOv8 serta sistem IoT. Website ini dirancang untuk memberikan visualisasi data secara informatif dan real time sehingga memudahkan pengguna dalam memahami kondisi jalan yang terdeteksi selama proses simulasi berlangsung.



Gambar 4. 3 Tampilan Full Website

Website dikembangkan sebagai bagian dari sistem berbasis client-server yang terintegrasi dengan backend, arsitektur ini terdiri dari:

a. Frontend (UI): menampilkan data kepada pengguna.

b. Backend (Server/API): mengelola data hasil deteksi.

c. Database: menyimpan hasil deteksi secara terstruktur.

Komunikasi antara frontend dan backend dilakukan melalui API berbasis HTTP dengan pendekatan RESTful, sehingga dapat diakses secara dinamis dan terpusat.

Backend sistem diimplementasikan menggunakan framework Flask yang berfungsi sebagai penyedia layanan API. Backend memliki beberapa fungsi utama, yaitu:

a. Menerima data hasil deteksi dari sistem YOLOv8
b. Menyimpan metadata deteksi kedalam database
c. Menyimpan file gambar hasil deteksi
d. Menyediakan endpoint API untuk diakses oleh frontend

Data yang diterima backend meliputi:

a. Jenis kerusakan jalan
b. Nilai confidence
c. Waktu deteksi
d. File gambar hasil deteksi

Setiap data yang masuk akan diproses dan disimpan secara terstruktur sehingga dapat digunakan untuk kebutuhan monitoring dan evaluasi sistem.

Implementasi frontend website dikembangkan menggunakan teknologi web seperti HTML, CSS, dan JS untuk menghasilkan tampilan yang responsif dan mudah digunakan. Antarmuka pengguna dirancang untuk menampilkan informasi hasil deteksi secara jelas dan terorganisir. Fitur utama yang tersedia pada website meliputi:

a. Informasi detail deteksi, seperti:

- Berbagai jenis kerusakan
- Nilai confidence
- Waktu deteksi

b. Tabel rekapitulasi data, yang menampilkan daftar hasil deteksi secara kronologis
c. Update data secara berkala, sehingga pengguna dapat memantau hasil deteksi secara near real-time

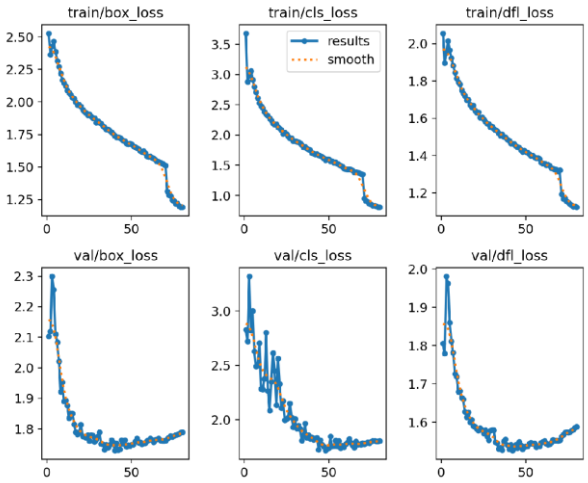
Hasil implementasi menunjukkan bahwa website mampu menampilkan data hasil secara terstruktur dan informatif. Pengguna dapat melihat hasil deteksi dalam bentuk visual maupun data numerik, sehingga memudahkan proses analisis terhadap performa sistem.

Selain itu, website juga mampu menyimpan riwayat deteksi yang dapat digunakan sebagai bahan evaluasi dan pengembangan sistem di masa mendatang.

4.2. Hasil Pengujian Sistem

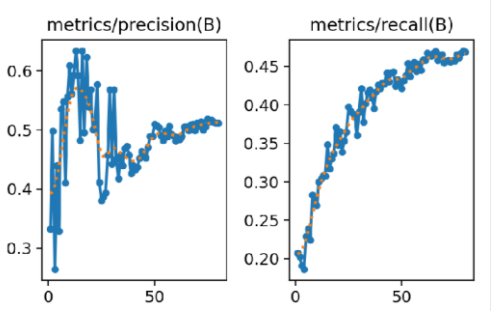
4.2.1. Analisis Performa Model YOLOv8

Berdasarkan hasil penelitian model YOLOv8 yang telah dilakukan, diperoleh beberapa indikator performa yang digunakan untuk mengevaluasi kemapuan model dalam mendeteksi kerusakan jalan. Evaluasi ini dilakukan dengan menganalisis grafik hasil training, nilai metrik evaluasi, serta kurva performa yang dihasilkan selama proses pengujian.



Gambar 4. 4 Grafik Hasil Pelatihan Model YOLOv8

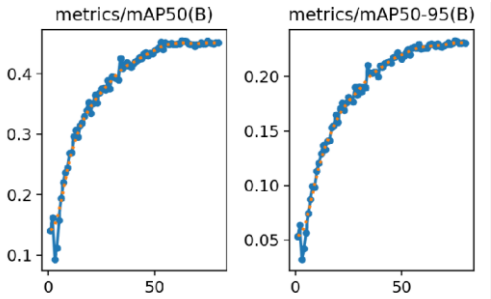
Dari grafik hasil penelitian, terlihat bahwa nilai *loss* pada data training yang meliputi *box loss*, *classification loss*, dan *distribution focal loss* mengalami penurunan secara konsisten seiring dengan bertambahnya jumlah epoch. Hal ini menunjukkan bahwa model mampu mempelajari pola dari dataset dengan baik. Selain itu, nilai *loss* pada datavalidasi juga menunjukkan tren penurunan pada awal pelatihan dan kemudian cendereung stabil pada epoch selanjutnya. Kondisi ini mengindikasi bahwa model tidak mengalami overfitting secara signifikan, karena performa antara data training dan data validasi relatif seimbang.



Gambar 4. 5 Grafik dari Precision dan Recall

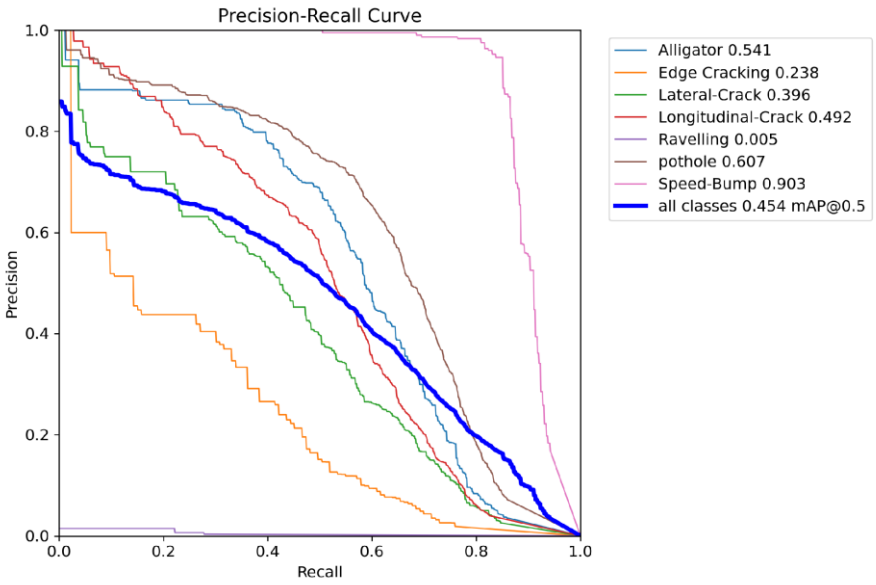
Selanjutnya, berdasarkan nilai metrik evaluasi, diperoleh nilai precision berada pada kisaran 0.50 hingga 0.52, sedangkan nilai recall

berada pada kisaran 0.46 hingga 0.47. nilai precision yang berada pada tingkat sedang menunjukkan bahwa model masih menghasilkan beberapa kesalahan dalam mendeteksi objek yang sebenarnya bukan merupakan kerusakan jalan *(false positive)*. Di sisi lain, nilai recall yang juga berada pada tingkat sedang mengindikasikan bahwa masih terdapat objek kerusakan jalan yang tidak berhasil terdeteksi oleh modle *(false negative)*. Hal ini menunjukkan bahwa model belum sepenuhnya optimal dalam menyeimbangkan antara kemampuan mendeteksi objek dan menghindari kesalahan deteksi.



Gambar 4. 6 Grafik Dari mAP50 dan mAP50-95

Evaluasi performa model juga dilakukan menggunakan metrik *mean Average Precision* (mAP). Berdasarkan hasil pengujian, diperoleh nilai mAP@0.5 sebesar 0.454 dan mAP@0.5:0.95 sebesar 0.23,. nilai mAP@0.5 menujukkan bahwa model memiliki kemampuan yang cukup baik dalam mendeteksi keberadaan objek kerusakan jalan secara umum. Namun demikian, nilai mAP@0.5:0.95 yang relatif rendah menunjukkan bahwa model masih memiliki keterbatasan dalam menghasilkan bounding box yang presisi pada berbagai tingkat *Intersection over Union* (IoU). Hal ini mengindikasikan bahwa meskipun objek dapat terdeteksi, akurasi posisi dan ukuran bounding box masih perlu ditingkatkan.



Gambar 4. 7 Kurva Precision-Recal Model YOLOv8

Analisis lebih lanjut dilakukan pada performa model untuk setiap kelas kerusakan jalan. Hasil pengujian menunjukkan bahwa kelas *speed bump* memiliki nilai Average Precision (AP) tertinggi, yaitu sebesar 0.903. Hal ini menunjukkan bahwa model sangat baik dalam mendeteksi objek dengan karakteristik visual yang jelas dan kontras. Selain itu, kelas *pothole* dan *alligator crack* juga menunjukkan performa yang cukup baik dengan nilai AP masing-masing sebesar 0.607 dan 0.541. Namun demikian, beberapa kelas lain seperti *lateral crack* dan *edge cracking* memiliki nilai AP yang lebih rendah, yaitu sebesar 0.396 dan 0.238, yang menunjukkan bahwa model mengalami kesulitan dalam membedakan pola kerusakan yang memiliki tekstur serupa.

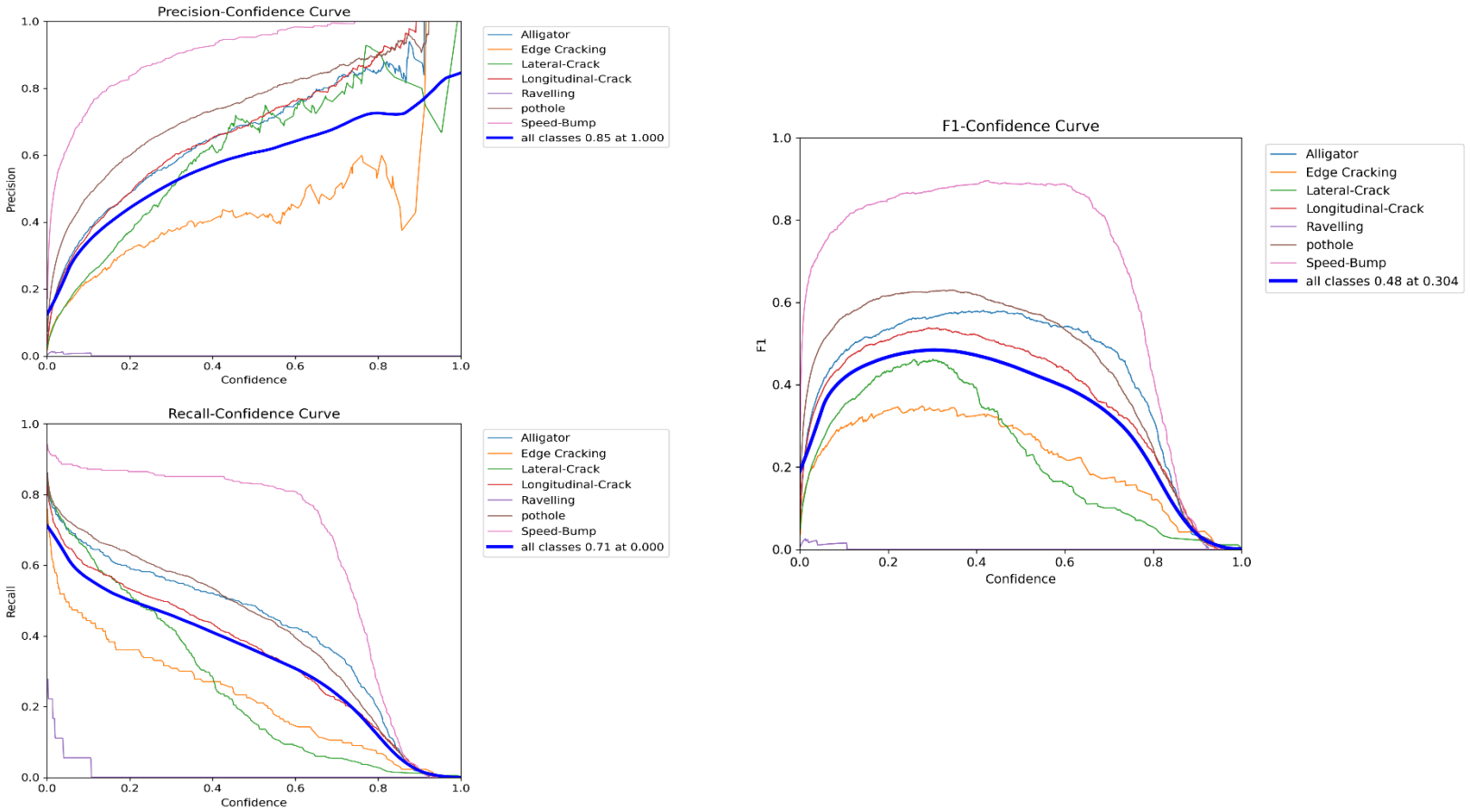
Kelas dengan performa terendah adalah *ravelling*, dengan nilai AP sebesar 0.005. Nilai ini menunjukkan bahwa model hampir tidak mampu mendeteksi jenis kerusakan tersebut. Hal ini kemungkinan disebabkan oleh keterbatasan jumlah dataset pada kelas tersebut, serta karakteristik visual

yang tidak konsisten dan sulit dibedakan dari kondisi jalan normal umumnya.

Ada juga analisis terhadap kurva *precision-recall* menujukkan bahwa kurva yang dihasilkan belum mendekati sudut kanan atas, yang mengindikasikan bahwa performa model masih belum optimal.

Sementara itu, pada kurva F1- score terhadap confidence sekitar 0,3. Hal ini menunjukkan bahwa nilai ambang (thresshold) yang optimal untuk sistem deteksi berada pada kisaran tersebut, karena memberikan keseimbangan terbaik antara precision dan recall.

Berdasarkan kurva hubungan antara recall dan confidence, terlihat bahwa peningkatan nilai confidence akan menyebabkan penurunan nilai recall secara signifikan. Hal ini menunjukkan bahwa jika sistem menggunakan nilai confidence yang terlalu tinggi, maka banyak objek kerusakan jalan yang tidak dapan terdeteksi. Oleh karena itu, pemilihan nilai confidence yang tepat menjadi faktor penting dalam menjaga kesimbangan performa sistem.



Gambar 4. 8 Hubungan antara Recall, F1-Score, & Precision Confidence Score

Secara keseluruhan, hasil pengujian menunjukkan bahwa model YOLOv8 yang dikembangkan dalam penelitian ini mampu medeteksi kerusakan jalan dengan tingkat akurasi yang cukup baik, terutama pada objek dengan karakteristik visual yang jelas seperti *speed bump* dan *pothole*. Namun demikian, model masih mengalami keterbatan dalam mendeteksi jenis kerusakan dengan pola yang kompleks dan tidak terstruktur, seperti *ravelling* dan *edge cracking*. Oleh karena itu, diperlukan peningkatan kualitas dan distribusi dataset, serta optimasi parameter pelatihan untuk meningkatkan performa model secara keseluruhan.

4.2.2. Pengujian sistem IoT

Pengujian sistem *Internet of Things* (IoT) pada penelitian ini dilakuka untuk mengevaluasi kemampuan sistem dalam merespon hasil deteksi kerusakan jalan yang dihasilkan oleh model YOLOv8. Pengujian difokuskan pada aspek waktu respon, akurasi respon aktuator, serta kestabilan komunikasi antara sistem deteksi dan perangkat IoT. Sistem IoT diharapkan mampu bekerja secara real-time dalam menerima data dan menghasilkan aksi berupa peringatan maupun simulasi pengereman.

a. Pengujian Waktu Respon Sistem

Pengujian waktu respon dilakukan untuk mengukur durasi yang dibutuhkan oleh sistem sejak kerusakan jalan terdeteksi hingga perangkat IoT memeberikan respon. Waktu respon dihitung mulai dari proses pengiriman data dari sistem deteksi ke server atau broker, hingga diterima dan diproses oleh ESP32 untuk mengaktifkan aktuator.

Berdasarkan hasil pengujian, sistem menunjukkan waktu respon yang relatif cepat dengan rata-rata waktu respon berada pada kisaran beberapa ratus milidetik hingga kurang dibawah 1 detik. Hal ini menunjukkan bahwa sistem mampu bekerja dalam kondisi near real-time. Namun demikian, terdapat variasi waktu respon yang dapat dipengaruhi oleh kondisi jaringan, seperti kestabilan koneksi WiFi dan latency komunikasi data.

b. Pengujian Akurasi Respon Aktuator

Pengujian ini dilakukan untuk mengetahui apakah aktuator yang digunakan, yaitu buzzer dan servo motor, dapat merespon dengan benar sesuai dengan kondisi kerusakan jalan yang terdeteksi. Sistem dikonfigurasikan dengan tiga kondisi utama, yaitu aman, peringatan, dan kritis.

Hasil pengujian menunjukkan bahwa sistem mampu mengaktifkan buzzer pada kondisi kritis dan mengaktifkan servo motor dimulai pada kondisi peringatan sekitar 15derajat secara kosisten sampai deteksi kerusakan berikutnya. Pada kondisi aman, sistem tidak memberikan respon, yang menunjukkan bahwa logika pengambilan keputusan telah berjalan dengan baik. Hal ini mmebuktikan bahwa integrasi antara hasil deteksi dan aksi fisik dapat berjalan sesuai dengan perancangan sistem.

c. Pengujian Konsistensi Sistem

Pengujian konsistensi dilakukan dengan menjalankan sistem secara berulang untuk memastikan bahwa perangkat IoT dapat merespon setiap data yang dikirimkan tanpa kegagalan. Pengujian dilakukan dengan

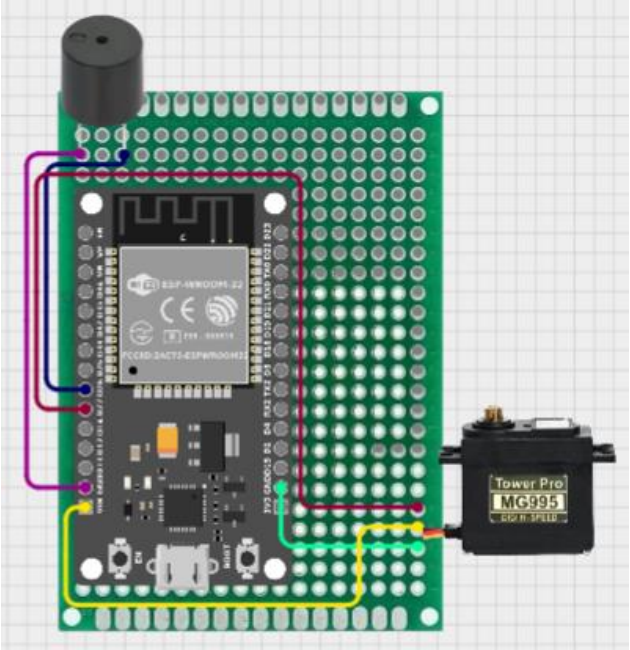
berbagai variasi kondisi input untuk melihat kestabilan sistem dalam jangka waktu tertentu.

Berdasarkan hasil pengujian, sistem menunjukan tingkat konsistensi yang baik, di mana sebagian besar data yang dikirimkan berhasil diterima dan diproses oleh ESP32 tanpa mengalami kegagalan. Meskipun demikian, terdapat beberapa kondisi di mana respo sistem mengalami keterlambatan, yang umumnya disebabkan oleh gangguan pada jaringan komunikasi.

d. Analisis Kinerja Sistem IoT

Secara keseluruhan, sistem IOT yang dikembangkan dalam penelitian ini mampu berfungsi dengan baik dalam menghubungkan hasil deteksi berbasis YOLOv8 dengan aksi fisik melalui perangkat keras. Sistem menunjukkan kemampuan respon yang cepat, akurat, dan cukup stabil dalam berbagai kondisi pengujian.

Namun demikian, performa sistem masih di pengaruhi oleh faktor eksternal, terutama kestabilan jaringan komunikasi. Oleh karena itu, untuk pengembangan lebih lanjut, diperlukan optimasi pada aspek komunikasi data serta penggunaan protokol yang lebih handal untuk meningkatkan keandalan sistem secara keseluruhan.



Gambar 4. 9 Rangkaian Perangkat ESP32

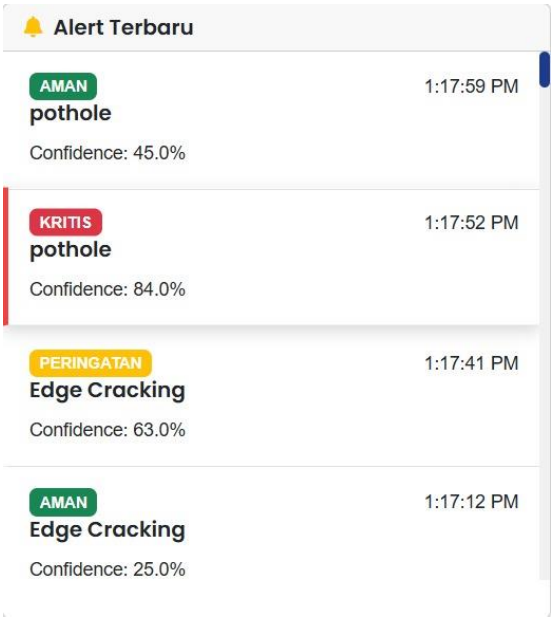
4.2.3. Pengujian Website Monitoring

Pengujian website monitoring dilakukan untuk mengevaluasi kemampuan sistem dalam menampilkan, menyimpan, dan mengelola data hasil deteksi kerusakan jalan yang diperoleh dari model YOLOv8. Pengujian ini bertujuan untuk memastikan bahwa proses pertukaran data antara sistem deteksi, server backend, database, dan antarmuka pengguna dapat berjalan dengan baik sesuai dengan rancangan sistem yang telah dibuat.

Website monitoring pada penelitian ini berperan sebagai media visualisasi hasil deteksi sekaligus sebagai sarana dokumentasi data yang memungkinkan pengguna untuk memantau kondisi kerusakan jalan secara terpusat. Oleh karena itu, pengujian dilakukan pada beberapa aspek utama yang meliputi proses penerimaan data, penyimpanan data, serta penyajian infomrasi kepada pengguna.

a. Pengujian Penerimaan Data Dari Sistem Deteksi

Pengujian ini dilakukan untuk memastikan bahwa data hasil deteksi yang dikirimkan oleh sistem YOLOv8 dapat diterima dengan baik oleh server backend melalui Application Programming Interface (API). Data yang dikirimkan meliputi jenis kerusakan jalan, nilai confidence, waktu deteksi.



Gambar 4. 10 Pengujian Deteksi dari API lalu masuk ke Database

Berdasarkan hasil pengujian, seluruh data yang dikirimkan oleh sistem deteksi berhasil diterimaoleh server tanpa mengalami kehilangan data. Setiap hasil deteksi yang dikirimkan melalui API dapat diproses dan diteruskan ke database untuk disimpan. Hasil ini menunjukkan bahwa mekanisme komunikasi antara sistem deteksi dan backend telah berjalan sesuai dengan perancangan yang telah ditetapkan.

b. Pengujian Penyimpanan Data ke Database

Pengujian penyimpanan data dilakukan untuk memastikan bahwa seluruh informasi hasil deteksi dapat tersimpan dengan baik pada database.

Data yang tersimpan kemudian digunakan sebagai sumber informasi bagi website monitoring dalam menampilkan riwayat deteksi.

id	object_type	confidence	status	timestamp	source_type	source_name	frame_number	x1	y1
1	pothole	0.4500	aman	2026-07-07 14:30:22	realtime	camera_1	1024	150	200
2	pothole	0.8400	bahaya	2026-07-07 14:30:45	video	jalan_raya_video_20260707.mp4	2456	320	180
3	Edge Cracking	0.6300	peringatan	2026-07-07 14:31:02	image	jalan_rusak_20260707.jpg	0	100	250
4	Edge Cracking	0.2500	aman	2026-07-07 14:31:18	realtime	camera_2	567	450	120

x2	y2	crop_image_path	full_image_path	image_width	image_height	file_size	is_processed	created_at	updated_at
350	400	/static/crops/realtime/2026-07-07/pothole_20260707...	/static/uploads/full/realtime/2026-07-07/frame_202...	640	480	NULL	1	2026-07-07 14:30:22	2026-07-07 13:03:43
480	420	/static/crops/video/2026-07-07/pothole_20260707_14...	/static/uploads/full/video/2026-07-07/frame_202607...	1920	1080	NULL	1	2026-07-07 14:30:45	2026-07-07 13:03:43
280	390	/static/crops/image/2026-07-07/edge_cracking_20260...	/static/uploads/full/image/2026-07-07/frame_202607...	1280	720	NULL	1	2026-07-07 14:31:02	2026-07-07 13:03:43
520	300	/static/crops/realtime/2026-07-07/edge_cracking_20...	/static/uploads/full/realtime/2026-07-07/frame_202...	640	480	NULL	1	2026-07-07 14:31:18	2026-07-07 13:03:43

Gambar 4. 11 Hasil yang masuk ke database

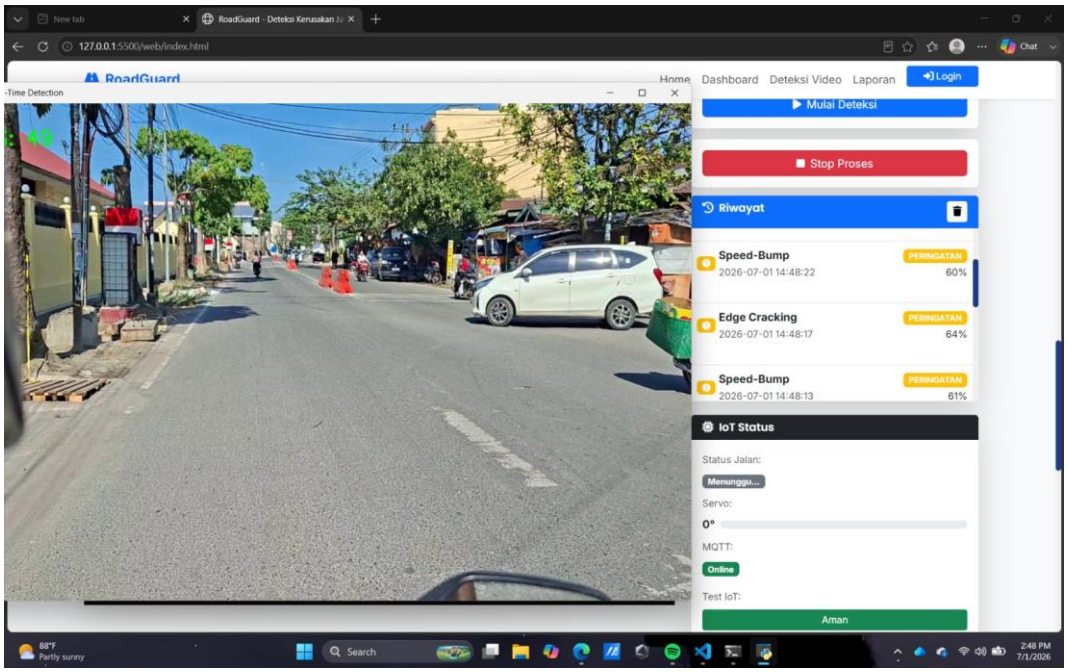
Hasil pengujian menunjukkan bahwa setiap data yang diterima oleh backend berhasil disimpan ke dalam database tanpa mengalami kesalahan penyimpanan. Data yang tersimpan dapat dipanggil kembali dan ditampilkan pada halaman website sesuai dengan informasi yang dikirimkan oleh sistem deteksi.

c. Pengujian Tampilan Website Monitoring

Pengujian tampilan website dilakukan untuk memastikan bahwa data yang telah tersimpan kedalam database dapat ditampilkan dengan benar pada antarmuka pengguna. Pengujian ini mencakup visualisasi gambar hasil deteksi, infomrasi jenis kerusakan jalan, nilai confidence, serta waktu deteksi.

Berdasarkan hasil pengujian, website mampu menampilkan seluruh informasi yang dibutuhkan secara lengkap dan terstruktur. Gambar hasil

deteksi dapat ditampilkan dengan baik beserta bounding box yang menunjukkan lokasi kerusakan jalan. Selain itu, informasi pendukung seperti jenis kerusakan dan nilai confidence juga dapat ditampilkan secara akurat sesuai data yang tersimpan pada database.



Gambar 4. 12 Tampilan Halaman Website Monitoring Hasil Deteksi Kerusakan Jalan

Website mampu menampilkan informasi hasil deteksi deteksi dalam bentuk visual maupun data tekstual. Informasi yang ditampilkan meliputi gambar hasil deteksi, jenis kerusakan, nilai confidence, dan waktu deteksi sehingga memudahkan pengguna dalam melakukan pemantauan kondisi jalan.

d. Analsisis Hasil Pengujian Website Monitoring

Berdasarkan seluruh pengujian yang telah dilakukan, website monitoring menunjukkan kinerja yang baik dalam menerima, menyimpan, dan menampilkan data hasil deteksi kerusakan jalan. Integrasi antara sistem

deteksi, backend, database, dan antarmuka pengguna dapat berjalan secara terkoordinasi sehingga informasi yang dihasilkan dapat disajikan kepada pengguna secara cepat dan terstruktur.

Selain berfungsi sebagai media monitoring, website juga berperan sebagai saran dokumentasi yang menyimpan riwayat hasil deteksi. Dengan adanya penyimpanan data secara terpusat, pengguna dapat melakuka evaluasi terhadap kondisi jalan berdasarkan data yang telah direkam oleh sistem. Hasil pengujian ini menunjukkan bahwa website yang dikembangkan telah memenuhi fungsi yang dirancang dalam penelitian sebagai media monitoring hasil deteksi kerusakan jalan berbasis YOLOv8 dan Internet of Things.

4.2.4. Pengujian Sistem Terintegrasi

Pengujian sistem terintegrasi dilakukan untuk memastikan bahwa seluruh komponen yang dikembangkan dalam penelitian ini dapat bekerja secara terkoordinasi sebagai satu kesatuan sistem. Pengujian ini bertujuan untuk mengevaluasi proses komunikasi data mulai dari tahap deteksi kerusakan jalan menggunakan model YOLOv8 hingga proses aktivasi perangkat IoT dan penyajian informasi pada website monitoring

Dalam pengujian ini, seluruh komponen sistem dioperasikan secara bersamaan sehingga dapat diketahui apakah data hasil deteksi mampu diteruskan dengan baik melalui API dan MQTT tanpa mengalami kehilangan informasi maupun kegagalan komunikasi

a. Skenario Pengujian Sistem Terintegrasi

Pengujian dilakukan dengan menggunakan video yang mengandung beberapa jenis kerusakan jalan sebgai input sistem. Ketika model YOLOv8 mendeteksi objek kerusakan jalan, hasil deteksi akan dikirimkan ke backen melalui API untuk disimpan pada database dan ditampilkan pada website monitoring. Selanjutnya data yang sama diteruskan melalui protokol MQTT menuju ESP32 untuk diproses sebagai dasar pengambilan keputusan terhadap aktuator.

Alur kerja sistem yang diuji dapat dijelaskan sebagai berikut:

1. Video diproses oleh model YOLOv8.
2. Model deteksi jenis kerusakan jalan.
3. Hasil deteksi dikirimkan ke backend melalui API.
4. Data disimpan pada database.
5. Website menampilkan hasil deteksi.
6. Data dikirimkan melalui MQTT.
7. ESP32 menerima data hasil deteksi.
8. Sistem menentukan tingkat kondisi jalan.
9. Buzzer dan servo memberikan respon sesuai kondisi yang terdeteksi.

b. Hasil Pengujian Sistem Terintegrasi

Berdasarkan hasil pengujian yang telah dilakukan, seluruh komponen sistem berhasil berkomunikasi dan menjalankan fungsinya sesuai dengan rancangan yang telah dibuat. Model YOLOv8 mampu mendeteksi objek kerusakan jalan dan menghasilkan data berupa jenis kerusakan jalan serta

nilai confidence. Data tersebut berhasil dikirimkan ke backend melalui API dan tersimpan pada database tanpa mengalami kehilangan data.

Selain itu, website monitoring mampu menampilkan hasil deteksi secara real-time sehingga pengguna dapat memantau kondisi jalan yang terdeteksi. Data hasil deteksi juga berhasil diteruskan melalui MQTT dan diterima oleh ESP32 sebagai dasar pengambilan keputusan terhadap aktuator.

Ketika sistem mendeteksi kondisi yang telah dikategorikan sebagai peringatan atau kritis, ESP32 mampu mengaktifkan buzzer maupun servo sesuai dengan logika yang telah ditentukan. Hal ini menunjukkan bahwa proses komunikasi data antar komponen berlangsung dengan baik dan menghasilkan respon fisik sesuai dengan hasil deteksi yang diperoleh.

c. Analisis Hasil Pengujian Sistem Terintegrasi

Hasil pengujian menunjukkan bahwa sistem yang dikembangkan telah berhasil mengintegrasikan teknologi computer vision, website monitoring, dan Internet of Things dalam satu platform yang saling terhubung. Seluruh proses mulai dari deteksi objek hingga respon aktuator dapat berjalan secara otomatis tanpa memerlukan intervensi pengguna.

Integrasi sistem memungkinkan informasi hasil deteksi tidak hanya digunakan sebagai data visual, tetapi juga dimanfaatkan untuk menghasilkan tindakan secara langsung melalui perangkat IoT. Dengan demikian, sistem yang dikembangkan mampu berfungsi sebagai sistem peringatan dini yang dapat membantu pengguna dalam mengenali kondisi jalan yang berpotensi membahayakan.

Meskipun demikian, performa sistem secara keseluruhan masih dipengaruhi oleh beberapa faktor eksternal seperti kualitas jaringan komunikasi, kondisi lingkungan saat proses deteksi, serta kulaitaas dataset yang digunakan dalam pelatihan model. Faktor-faktor tersebut perlu diperhatikan untuk meningkatkan keandalan sistem pada pengembangan selanjutnya.

4.3. Kelebihan dan Keterbatan dari Sistem yang Dibangun

4.3.1. Kelebihan Sistem

Pada bagian ini dibahas keunggulan yang berhasil dicapaiberdasarkan hasil implementasi dan pengujian.

a. Kemampuan Deteksi Otomatis

Salah satu kelebihan utama sistem yang dikembangkan adalah kemampuan dalam mendeteksi kerusakan jelan secara otomatis menggunakan model YOLOv8. Sistem mampu mengenali beberapa jenis kerusakan jalan berdasarkan karakteristik visual yang terdapat pada citra atau video sehingga proses identifikasi tidak lagi dilakukan secara manual.

b. Integrasi Dengan Sistem IoT

Berbeda dengan penelitian yang hanya berfokus pada proses deteksi objek, sistem yang dikembangkan dalam penelitian ini telah terintegrasi dengan perangkat Internet of Things (IoT). Integrasi tersebut memungkinkan hasil deteksi digunakan sebagai dasar pengambilan keputusan untuk mengaktifkan buzzer maupun servo secara otomatis sesuai tingkat kerusakan yang terdeteksi.

c. Monitoring Berbasis Website

Sistem menyediakan antarmuka website yang mampu menampilkan hasil deteksi secara terpusat sehingga pengguna dapat memantau kondisi jalan yang terdeteksi beserta informasi pendukung seperti jenis kerusakan, nilai confidence, dan waktu deteksi.

d. Arsitektur Modular

Arsitektur sistem yang dibangun memiliki struktur modular yang memisahkan proses deteksi, komunikasi data, monitoring, dan eksekusi aktuator. Pendekatan ini memudahkan proses pengembangan, pemeliharan, maupun integrasi dengan teknologi lain pada penelitian selanjutnya.

4.3.2. Keterbatasan Sistem

a. Keterbatasan Dataset

Performa model masih dipengaruhi oleh distribusi dataset yang digunakan pada proses pelatihan. Beberapa kelas kerusakan menunjukkan nilai Average Precision yang relatif rendah sehingga kemampuan model dalam mengenali jenis kerusakan tertentu belum optimal.

b. Sensitif Terhadap Kondisi Lingkungan

Akurasi deteksi masih dipengaruhi oleh faktor lingkungan seperti pencahayaan, bayangan, kualitas video, dan sudut pengambilan gambar. Perubahan kondisi tersebut dapat menyebabkan penurunan performa deteksi pada beberapa situasi tertentu.

c. Pengujian Masih Menggunakan Video Simulasi

Implementasi sistem pada penelitian ini masih menggunakan video rekaman sebagai sumber input sehingga belum sepenuhnya

merepresentasikan kondisi operasional kendaraan roda dua dalam lingkungan nyata secara langsung.

d. Ketergantungan Pada Koneksi Jaringan

Kinerja sistem IoT sangat bergantung pada stabilitas jaringan komunikasi yang digunakan. Gangguan koneksi dapat menyebabkan peningkatan latensi maupun keterlambatan respon perangkat terhadap hasil deteksi yang diterima.

e. Simulasi Pengereman Masih Menggunakan Servo

Mekanisme pengereman pada peneltian ini masih berupa simulasi menggunakan servo motor sehingga belum dapat diterapkan secara langsung pada sistem pengereman kendaraan yang sebenarnya. Oleh karena itu, diperlukan pengembangan lebih lanjut terkait aspek keamanan dan integrasi mekanik sebelum sistem dapat digunakan pada kendaraan operasional.

BAB V

PENUTUP

5.1. Kesimpulan

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan pada penelitian ini, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Penelitian ini berhasil mengimplementasikan model YOLOv8 untuk mendeteksi berbagai jenis kerusakan jalan, seperti *pothole*, *speed bump*, *alligator crack*, *lateral crack*, *longitudal crack*, *edge cracking*, dan *ravelling*. Model yang dikembangkan mampu melakukan deteksi objek berdasarkan karakteristik visual pada citra dengan tingkat performa yang cukup baik, meskipun masih terdapat beberapa kelas dengan akurasi yang rendah akibat keterbatasan dataset.
2. Sistem *Internet of Things* (IoT) yang dibangun menggunakan ESP32 berhasil diintegrasikan dengan sistem deteksi untuk menghasilkan respon otomatis berupa peringatan suara melalui buzzer dan simulasi pengereman menggunakan servo motor. Berdasarkan hasil pengujian, sistem IoT mampu merespon data hasil deteksi dengan waktu respon yang relative cepat dan tingkat akurasi yang baik sesuai dengan logika yang telah dirancang.
3. Website monitoring yang dikembangkan mampu menerima, menyimpan, dan menampilkan data hasil deteksi secara terstruktur. Website dapat menyajikan informasi berupa gambar hasil deteksi, jenis kerusakan jalan, nilai confidence, serta waktu deteksi, sehingga memudahkan pengguna dalam melakukan pemantauan kondisi jalan secara real-time maupun berdasarkan data historis.

4. Integrasi antara model YOLOv8, sistem backend, website monitoring, dan perangkat IoT berhasil diimplementasikan dengan baik sehingga seluruh komponen sistem dapat bekerja secara terkoordinasi. Sistem mampu mengalirkan data mulai dari proses deteksi hingga menghasilkan respon fisik secara otomatis tanpa intervensi pengguna, sehingga dapat berfungsi sebagai sistem peringatan dini terhadap kondisi jalan yang berpotensi membahayakan.

Secara keseluruhan, sistem yang dikembangkan dalam penelitian ini telah memenuhi tujuan penelitian, yaitu membangun sistem deteksi kerusakan jalan berbasis komputer visi yang terintegrasi dengan IoT dan website monitoring.

5.2.Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa saran yang dapat digunakan sebagai bahan pengembangan pada penelitian selanjutnya:

1. Peningkatan kualitas dan jumlah dataset sangat disarankan untuk meningkatkan performa model dalam mendeteksi berbagai jenis kerusakan jalan, khuksusnya pada kelas dengan akurasi yang rendah seperti ravelling dan edge cracking. Dataset yang lebih beragam akan membantu model dalam melakukan generalisasi terhadap kondisi jalan yang berbeda.
2. Pengujian sistem sebaiknya dilakukan secara langsung pada kondisi nyata menggunakan kamera yang dipasangkan pada kendaraan langsung, sehingga hasil penelitian dapat lebih merepresentasikan kondisi operasional yang sebenarnya.

3. Sistem IoT dapat dikembangkan lebih lanjut dengan menggunakan metode komunikasi yang lebih stabil dan memiliki latensi rendah untuk meningkatkan kecepatan dan keandalan respons sistem, terutama pada kondisi jaringan yang tidak stabil.
4. Mekanisme pengereman yang saat ini masih berupa simulasi menggunakan servo motor dapat dikembangkan menjadi sistem yang lebih realistis dengan mempertimbangkan aspek keamanan dan integrasi dengan sistem kendaraan yang sesungguhnya.
5. Website monitoring dapat dikembangkan dengan menambahkan fitur visualisasi lokasi kerusakan dengan menambahkan teknologi *Global Positioning System* (GPS), sehingga pengguna dapat mengetahui lokasi kerusakan jalan secara presisi dengan geografis pengguna.

Penelitian selanjutnya dapat melakukan perbandingan antara YOLOv8 dengan metode deteksi lainnya untuk memperoleh model dengan performa yang lebih optimal dalam mendeteksi kerusakan jalan

DAFTAR PUSTAKA

A\Arnita, Marpaung, F., Aulia, F., Suryani, N., & Cyra Nabila, R. (2022). *Computer Vision and Digital Image Processing*. www.pustakaaksara.co.id

Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*. http://arxiv.org/abs/2004.10934

Domínguez-Bolaño, T., Campos, O., Barral, V., Escudero, C. J., & García-Naya, J. A. (2022). An overview of IoT architectures, technologies, and existing open-source projects. *Internet of Things (Netherlands)*, *20*, 100626. https://doi.org/10.1016/j.iot.2022.100626

Hevner, A. R. (2020). DESIGN SCIENCE IN INFORMATION SYSTEMS RESEARCH. *AI and Society*, *10*(2), 199–217. https://doi.org/10.1007/BF01205282

Kansil, K. K., Walewangko, E. N., & Masloman, I. (2023). *ANALISIS INFRASTRUKTUR TERHADAP PERTUMBUHAN EKONOMI KOTA MANADO*. *23*(8), 121–132.

Kustiawan, W., Raihan, M. D., & Thariq, M. I. (2025). Pendekatan Metode Penelitian Lokasi Dan Waktu InformanPenelitian Dan Penyusunan. *Jurnal Pendidikan Tambusai*, *9*, 5051–5055.

Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., Nie, W., Li, Y., Zhang, B., Liang, Y., Zhou, L., Xu, X., Chu, X., Wei, X., & Wei, X. (2022). *YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications*. http://arxiv.org/abs/2209.02976

Li, Y., Yin, C., Lei, Y., Zhang, J., & Yan, Y. (2024). *RDD-YOLO : Road Damage Detection Algorithm Based on Improved You Only Look Once Version 8*.

Manurung, R. B. (2024). *DETEKSI DAN ESTIMASI DIMENSI LUBANG JALAN SECARA REAL-TIME BERBASIS DEEP LEARNING*.

Mittal, P. (2024). A comprehensive survey of deep learning-based lightweight object detection models for edge devices. In *Artificial Intelligence Review* (Vol. 57, Issue 9). Springer Netherlands. https://doi.org/10.1007/s10462-024-10877-1

Rahaman, M. M. (2022). A Review on Internet of Things-IoT- Architecture , Technologies , Future Applications & Challenges. *International Journal of Science and Business*, *14*(1), 80–92. https://doi.org/10.5281/zenodo.7066810

Rahman, M. M. M. (2023). *Road Damage Detection Based on Deep Learning*.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2018). You Only Look Once: Unified, Real-Time Object Detection. *ACM International Conference Proceeding Series*. https://doi.org/10.1145/3243394.3243692

Sakinah, L., Haryatmi, E., & Riyadi, T. A. (2025). *Implementasi Algoritma Yolo Untuk Mendeteksi Jalan Berlubang dan Retak*. *6*(3), 251–257. https://doi.org/10.62527/jitsi.6.

Sasmito, B., Setiadji, B. H., & Isnanto, R. (2023). *Deteksi Kerusakan Jalan Menggunakan Pengolahan Citra Deep Learning di Kota Semarang*. *44*(1), 7–14. https://doi.org/10.14710/teknik.v44i1.51908

Shobaki, W. A., & Milanova, M. (2025). A Comparative Study of YOLO, SSD, Faster R-CNN, and More for Optimized Eye-Gaze Writing. *Sci*, *7*(2). https://doi.org/10.3390/sci7020047

Siswanto, H., Sulistio, H., Djakfar, L., & Wicaksono, A. (2016). *SISTEM MANAJEMEN JALAN DAN KONDISI KERUSAKAN JALAN DI*

INDONESIA : SEBUAH KAJIAN PUSTAKA.

Sorri, K., Mustafee, N., & Seppänen, M. (2022). Revisiting IoT definitions: A framework towards comprehensive use. *Technological Forecasting and Social Change*, *179*(January). https://doi.org/10.1016/j.techfore.2022.121623

Steiniger, Y., Kraus, D., & Meisen, T. (2022). Survey on deep learning based computer vision for sonar imagery. *Engineering Applications of Artificial Intelligence*, *114*(March), 105157. https://doi.org/10.1016/j.engappai.2022.105157

Sugiyono. (2020). *Metodologi Penelitian Kuantitatif, Kualitatif dan R & D*.

Suhendar, H. (2023). Road Crack Detection using Yolo-V5 and Adaptive Thresholding. *Journal of Applied Intelligent System*, *8*(3), 425–431. https://doi.org/10.33633/jais.v8i3.9437

Tamboli, K., Mane, V., Pulgam, N., & Maktum, T. (2025). *Study of IoT Layers Vulnerabilities and Attack Surfaces*. *IC3Com 2024*, 76–84. https://doi.org/10.5220/0013264000004646

Wang, Y., Han, Y., Wang, C., Song, S., Tian, Q., & Huang, G. (2023). *Computation-efficient Deep Learning for Computer Vision: A Survey*. 1–47. http://arxiv.org/abs/2308.13998

Witczak, D., & Szymoniak, S. (2024). Review of Monitoring and Control Systems Based on Internet of Things. *Applied Sciences (Switzerland)*, *14*(19). https://doi.org/10.3390/app14198943

Zhang, H., & Cloutier, R. S. (2022). Review on One-Stage Object Detection Based on Deep Learning. *EAI Endorsed Transactions on E-Learning*, *7*(23), 174181. https://doi.org/10.4108/eai.9-6-2022.174181

Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. (2023). Object Detection in 20 Years: A Survey. *Proceedings of the IEEE*, *111*(3), 257–276. https://doi.org/10.1109/JPROC.2023.3238524