

Sistem Deteksi Kendaraan Real-Time Menggunakan YOLOv5 dan Antarmuka Web Flask

Muhammad Fathsya Akbar¹, Muhammad Alaikan Ni'am², Dani Siva Vernanda³, Wahib Bagas Saputra⁴ Muhammad Munsarif⁵

¹Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer Universitas Muhammadiyah Semarang, Semarang, Indonesia

² Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer Universitas Muhammadiyah Semarang, Semarang, Indonesia

³ Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer Universitas Muhammadiyah Semarang, Semarang, Indonesia

⁴ Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer Universitas Muhammadiyah Semarang, Semarang, Indonesia

⁵ Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer Universitas Muhammadiyah Semarang, Semarang, Indonesia

Info Artikel

Riwayat Artikel:

Diterima 15, Oktober, 2025

Perbaikan 22, Desember, 2025

Disetujui 13, Januari, 2026

Keywords:

YOLOv5
Flask
Deteksi Objek
Kendaraan
Real-time
Deep Learning
Citra Video

ABSTRAK

Identifikasi kendaraan menjadi salah satu komponen krusial dalam teknologi pemantauan lalu lintas serta sistem keamanan transportasi. Permintaan terhadap solusi yang mampu mendeteksi secara langsung (real-time) dan dapat diakses melalui platform berbasis web mendorong lahirnya pendekatan baru berbasis deep learning yang efisien dan adaptif. Dalam studi ini, dirancang sebuah sistem pendekripsi kendaraan dengan memanfaatkan algoritma YOLOv5, yang dikenal unggul dalam kecepatan pemrosesan, ukuran model yang lebih ramping, serta kemudahan penerapannya bila dibandingkan dengan versi sebelumnya dari YOLO. Sistem tersebut diintegrasikan menggunakan framework Flask untuk menghadirkan antarmuka web yang menampilkan hasil deteksi dari berbagai sumber video, baik melalui kamera secara langsung maupun file yang diunggah. Berdasarkan pengujian, sistem ini mampu mengenali kendaraan seperti mobil, bus, dan truk dengan tingkat akurasi dan kecepatan tinggi, sekaligus menyajikan data jumlah kendaraan secara langsung. Temuan ini membuktikan bahwa penggabungan YOLOv5 dan Flask dapat menjadi solusi efektif untuk implementasi sistem pendekripsi kendaraan berbasis web dengan kinerja yang cepat dan akurat.

Vehicle identification is a crucial component in traffic monitoring technology and transportation security systems. The demand for solutions capable of real-time detection that are accessible via web-based platforms has driven the development of new, efficient, and adaptive deep learning approaches.

In this study, a vehicle detection system is designed utilizing the YOLOv5 algorithm, which is renowned for its superior processing speed, compact model size, and ease of deployment compared to previous YOLO iterations. The system is integrated using the Flask framework to provide a web interface that displays detection results from various video sources, including live camera feeds and uploaded files.

Testing results demonstrate the system's ability to recognize vehicles—such as cars, buses, and trucks—with high accuracy and speed, while simultaneously presenting real-time vehicle count data. These findings demonstrate that combining YOLOv5 and Flask offers an effective solution for implementing web-based vehicle detection systems characterized by fast and accurate performance.

Ini adalah artikel akses terbuka di bawah lisensi CC BY-SA.



Penulis Korespondensi:

Penulis Pertama

Program Studi Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Muhammadiyah Semarang
Alamat: Gedung GKB 2Lt. 7, Ruang 707, Jl. Kedungmundu Raya No.18, Semarang 50273, Indonesia
Email: penulispertama@unimus.ac.id

1. PENDAHULUAN

Kemajuan teknologi deep learning dalam deteksi objek mengalami pertumbuhan signifikan, khususnya sejak dikenalkannya algoritma You Only Look Once (YOLO) yang memungkinkan pendekripsi objek hanya melalui satu proses pemrosesan cepat. Di antara berbagai versi yang telah hadir, YOLOv5 menjadi salah satu pilihan utama karena memiliki ukuran model yang kecil dan mudah diintegrasikan dalam pengembangan aplikasi Python seperti Flask (Mulyana dkk., 2022). Walaupun versi lebih mutakhir seperti YOLOv6 hingga YOLOv8 menawarkan peningkatan dari sisi akurasi dan kecepatan, kebutuhan daya komputasinya yang besar membuatnya kurang ideal bagi sistem web ringan.

Beberapa studi sebelumnya telah membuktikan efektivitas penggunaan YOLOv5 untuk mengenali kendaraan dalam berbagai situasi. Mulyana dkk., (2022), contohnya, berhasil merancang sistem klasifikasi kendaraan secara langsung di Indonesia menggunakan algoritma YOLOv5, dengan tingkat akurasi hingga 90% terhadap kendaraan lokal seperti bajaj, becak, truk, hingga mobil pikap. Hasil ini menunjukkan bahwa YOLOv5 mampu beradaptasi dengan baik terhadap kondisi riil kendaraan di lapangan. Sementara itu, penelitian oleh Michael et al., (2024) membandingkan sejumlah varian YOLOv5 seperti v5n, v5s, dan v5m, dan menyimpulkan bahwa YOLOv5s menawarkan keseimbangan terbaik antara efisiensi pemrosesan dan akurasi, terutama dalam kondisi lalu lintas yang padat.

Selain itu, penelitian dari Zhang et al., (2024) memperkenalkan sistem pendekripsi kendaraan yang menggabungkan YOLOv5 dengan teknologi kamera stereo (binocular vision) guna meningkatkan ketepatan penentuan posisi kendaraan dalam aplikasi otomotif. Integrasi ini tidak hanya memperkuat kemampuan deteksi, tetapi juga memungkinkan sistem untuk menentukan letak kendaraan dengan lebih akurat. Di sisi lain, penelitian Zhang et al., (2022) melakukan penyempurnaan pada YOLOv5 dengan menambahkan metode augmentasi gambar seperti Flip-Mosaic, yang terbukti efektif dalam meningkatkan akurasi deteksi terhadap kendaraan berukuran kecil di area jalan yang rumit dan dinamis.

Walaupun berbagai penelitian tersebut menunjukkan keandalan YOLOv5, sebagian besar pengembangannya masih terbatas pada sistem offline dan belum menawarkan akses melalui antarmuka web secara langsung. Keterbatasan inilah yang menjadi alasan utama dilaksanakannya penelitian ini. Fokus utama dalam penelitian ini adalah merancang sistem deteksi kendaraan real-time yang langsung terhubung dengan antarmuka web menggunakan framework Flask. Sistem ini tidak hanya memperlihatkan visual hasil deteksi, tetapi juga menampilkan data statistik jumlah kendaraan secara interaktif. Dengan pendekatan semacam ini, penelitian ini bertujuan memberikan kontribusi dalam pengembangan teknologi pemantauan lalu lintas yang cerdas serta mendukung terciptanya infrastruktur kota pintar berbasis teknologi modern.

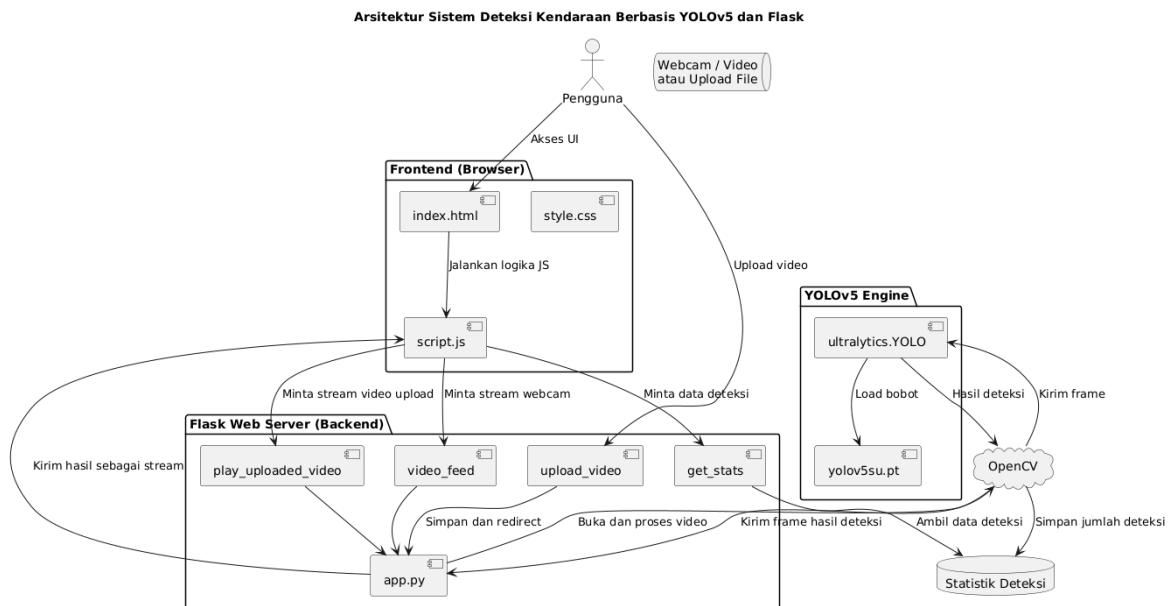
2. METODE

Dalam penelitian ini, pendekatan eksperimental diterapkan untuk merancang sistem deteksi kendaraan secara langsung (real-time) yang didukung oleh teknologi deep learning, dengan memanfaatkan model pralatih YOLOv5. Pengembangan sistem dilakukan menggunakan framework Flask sebagai server backend, sementara pemrosesan video dilakukan melalui bantuan library OpenCV. Di sisi frontend, JavaScript dikombinasikan dengan Chart.js untuk menyajikan hasil deteksi kendaraan dan visualisasi data dalam bentuk yang interaktif dan mudah dipahami pengguna.

2.1 Arsitektur Sistem

Struktur sistem ini dibagi menjadi tiga bagian utama, yaitu:

1. Komponen Backend, yang berfungsi untuk mengelola pengambilan data video baik melalui kamera langsung maupun file unggahan. Frame video yang diperoleh kemudian diproses menggunakan model YOLOv5 yang telah dilatih sebelumnya dengan bobot *yolov5su.pt*.
2. Komponen Deteksi Objek, yang bertugas menerapkan algoritma YOLO versi 5 (*You Only Look Once*) guna mendekripsi serta mengklasifikasi jenis kendaraan dalam setiap frame video, seperti mobil, bus, dan truk.
3. Komponen Frontend, yang dirancang untuk menampilkan hasil deteksi dalam bentuk video secara streaming dan menyajikan visualisasi jumlah objek dalam grafik batang menggunakan Chart.js, dengan pembaruan data yang dilakukan secara berkala.



Gambar 1. Arsitektur sistem deteksi kendaraan berbasis YOLOv5 dan Flask

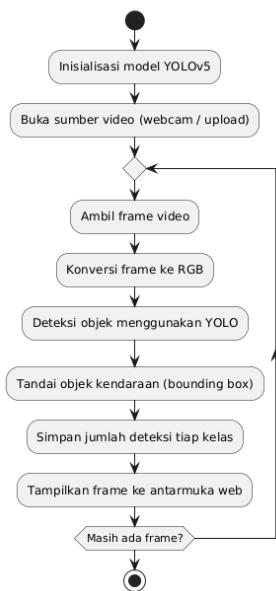
2.2 Alur Kerja Sistem

Proses deteksi objek dalam sistem ini dilakukan melalui beberapa tahapan berikut:

1. Inisialisasi Model: Sistem memuat model YOLOv5 bersama dengan bobot pralatih *yolov5su.pt* sebagai dasar deteksi.
2. Pengambilan Sumber Video: Video diperoleh baik dari kamera secara langsung (live streaming) maupun dari file yang diunggah oleh pengguna ke sistem.

- 3. Pra-pemrosesan Gambar: Setiap frame yang diambil diubah ke dalam format warna RGB dan disiapkan untuk proses deteksi lebih lanjut.
- 4. Identifikasi Objek Kendaraan: Model YOLOv5 kemudian menganalisis frame untuk mengenali objek, menghasilkan bounding box, tingkat kepercayaan (confidence score), dan identitas kelas objek (class ID).
- 5. Tampilan Visual Deteksi: Hasil deteksi divisualisasikan langsung pada frame video dan ditampilkan kepada pengguna melalui layanan streaming berbasis web.
- 6. Penyajian Data Statistik: Sistem menghitung jumlah kendaraan yang berhasil dikenali dan menyajikannya dalam bentuk grafik batang di antarmuka klien untuk kemudahan analisis visual.

Flowchart Sistem Deteksi Kendaraan Berbasis YOLOv5



Gambar 2. Alur Kerja Sistem

2.3 Desain Antarmuka Web

Tampilan antarmuka sistem dibangun menggunakan kombinasi HTML, CSS, dan JavaScript. Pengguna dapat mengontrol proses streaming video melalui tombol interaktif yang berfungsi untuk memulai maupun menghentikan tampilan secara langsung. Sementara itu, data jumlah kendaraan yang terdeteksi divisualisasikan dalam bentuk grafik batang yang diperbarui otomatis setiap dua detik dengan memanfaatkan fungsi `fetch()` dari endpoint API `/get_stats`.

2.4 Lingkungan dan Alat Pengujian

Pengujian sistem dilakukan pada lingkungan lokal dengan rincian konfigurasi sebagai berikut:

- A. Bahasa pemrograman yang digunakan adalah Python versi 3.10
- B. Framework pengembangan aplikasi backend menggunakan Flask
- C. Deteksi objek kendaraan dijalankan dengan library Ultralytics YOLOv5
- D. Untuk pengolahan gambar dan video, sistem memanfaatkan OpenCV
- E. Model yang digunakan adalah *yolov5su.pt*, yaitu model hasil pelatihan khusus (custom-trained)
- F. Antarmuka pengguna dikembangkan menggunakan HTML5, CSS3, JavaScript, serta Chart.js untuk visualisasi data.

2.5 Validasi dan Evaluasi

Proses validasi sistem dilakukan dengan menggunakan dua jenis sumber video, yaitu:

- A. Streaming secara langsung melalui kamera webcam
- B. File video yang berisi rekaman lalu lintas

Pengujian dilakukan untuk menilai efektivitas deteksi terhadap tiga kategori kendaraan, sekaligus mengevaluasi kestabilan tampilan video dan grafik visualisasi. Parameter performa yang diukur meliputi:

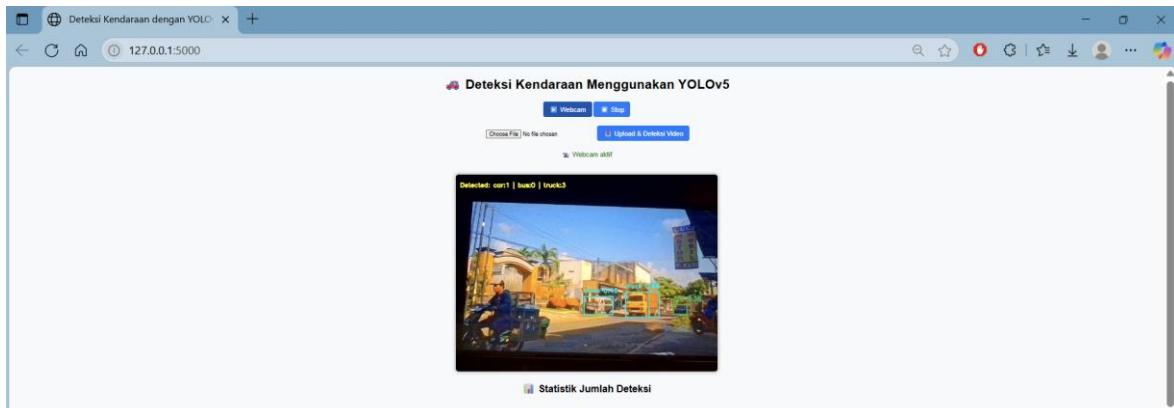
- A. Tingkat akurasi dalam mendeteksi objek
- B. Waktu pemrosesan setiap frame video
- C. Stabilitas pembaruan grafik secara berkelanjutan

3. HASIL DAN PEMBAHASAN

Setelah sistem deteksi kendaraan berhasil dibangun, tahap berikutnya adalah melakukan pengujian menggunakan dua tipe input: video langsung melalui webcam dan video yang diunggah pengguna melalui antarmuka web. Tampilan antarmuka dibuat agar ramah pengguna, dengan kemampuan menampilkan hasil deteksi dalam bentuk streaming video yang disertai informasi jumlah kendaraan yang teridentifikasi secara real-time.

Seperti ditunjukkan pada Gambar 1, pengguna dapat memilih sumber video yang diinginkan dan langsung menyaksikan hasil deteksi kendaraan. Setiap objek yang dikenali ditandai dengan *bounding box* dan label kategori pada frame video. Selain itu, sistem juga menyajikan data jumlah kendaraan berdasarkan jenisnya secara otomatis, sehingga memudahkan pemantauan.

Berdasarkan hasil uji awal, sistem menunjukkan performa yang cukup baik dalam mendeteksi objek seperti mobil, bus, dan truk, bahkan ketika diuji pada kondisi pencahayaan yang bervariasi serta sudut pengambilan gambar yang berbeda. Contoh visual dari deteksi objek dapat dilihat pada Gambar 3, yang memperlihatkan keberhasilan sistem dalam memberi label dan kotak pembatas pada kendaraan yang dikenali dari beberapa cuplikan video.



Gambar 3. Tampilan Website

Guna memperoleh hasil evaluasi yang lebih komprehensif, sistem diuji menggunakan lima video berbeda yang memiliki variasi dalam durasi pemutaran, jumlah kendaraan, serta jenis kendaraan yang muncul. Rekapitulasi hasil deteksi dari seluruh pengujian disajikan dalam Tabel 1, yang menampilkan jumlah kendaraan yang berhasil dikenali berdasarkan kategorinya untuk masing-masing video.

Video ke-	Mobil	Bus	Truk	Total
Video 1	12	1	2	15
Video 2	8	2	4	14
Video 3	15	0	3	18
Video 4	10	1	1	12
Video 5	20	3	6	29

Tabel 1. Jumlah Kendaraan yang terdeteksi

Dari aspek kinerja, sistem mampu memproses antara 15 hingga 25 frame per detik (FPS) saat dijalankan di perangkat dengan spesifikasi menengah, seperti prosesor Intel i5 dan RAM 8GB. Hasil ini menunjukkan bahwa sistem dapat beroperasi secara real-time tanpa bergantung pada GPU khusus, menjadikannya solusi yang efisien dan layak digunakan di lingkungan dengan keterbatasan perangkat keras.

Dari segi akurasi, performa deteksi tergolong tinggi, terutama saat mengenali kendaraan berukuran sedang hingga besar seperti mobil dan bus. Namun, deteksi pada objek truk mengalami sedikit penurunan presisi, khususnya saat objek berada terlalu jauh atau tertutup sebagian. Meskipun begitu, sistem backend yang dikembangkan menggunakan Flask tetap berjalan lancar, mampu menampilkan video streaming dan menyuguhkan statistik kendaraan tanpa mengalami hambatan teknis atau bottleneck yang berarti.

Keunggulan utama dari sistem ini terletak pada kemudahan dalam penggunaannya. Karena berbasis web, pengguna hanya perlu mengaksesnya melalui browser tanpa instalasi perangkat lunak tambahan. Penggabungan antara YOLOv5 dan Flask tidak hanya membuat sistem ini ringan dari sisi beban komputasi, tetapi juga memberikan fleksibilitas tinggi dalam implementasi.

Meski demikian, sistem masih memiliki keterbatasan, terutama pada cakupan klasifikasi kendaraan. Model YOLOv5 yang digunakan dalam penelitian ini dilatih menggunakan dataset COCO, sehingga belum mampu mengenali tipe kendaraan khas Indonesia seperti angkot, mikrobus, maupun kendaraan niaga lokal lainnya. Secara keseluruhan, penelitian ini membuktikan bahwa integrasi YOLOv5 dan Flask menghasilkan sistem deteksi kendaraan yang ringan, real-time, dan responsif untuk keperluan pemantauan lalu lintas berbasis web.

4. KESIMPULAN

Penelitian ini berhasil mengembangkan sistem deteksi kendaraan secara real-time dengan mengimplementasikan algoritma YOLOv5 yang terintegrasi ke dalam antarmuka web menggunakan framework Flask. Sistem yang dirancang mampu mengenali kendaraan seperti mobil, truk, dan bus dengan kecepatan serta akurasi yang memadai, baik melalui input kamera langsung maupun dari file video. Informasi jumlah kendaraan yang terdeteksi juga ditampilkan secara interaktif di halaman web, memberikan pengalaman pemantauan yang informatif dan real-time bagi pengguna.

Berdasarkan hasil pengujian, sistem menunjukkan performa yang stabil dan responsif meskipun dijalankan pada perangkat tanpa GPU. Temuan ini mengindikasikan bahwa kombinasi YOLOv5 dan Flask efektif digunakan untuk membangun aplikasi pendekripsi objek berbasis video streaming yang ringan, mudah diakses melalui browser, dan tidak memerlukan instalasi software tambahan.

Salah satu nilai lebih dari sistem ini terletak pada kemudahan instalasi serta efisiensi kinerjanya, menjadikannya ideal untuk digunakan dalam berbagai kebutuhan, mulai dari pemantauan lalu lintas hingga sistem pengawasan visual di ruang publik maupun area privat.

Sebagai arah pengembangan ke depan, sistem ini memiliki potensi untuk ditingkatkan dengan melatih ulang model YOLOv5 menggunakan dataset lokal, agar mampu mengenali kendaraan khas Indonesia seperti angkot atau truk niaga dengan lebih presisi. Selain itu, sistem juga dapat diperluas dengan mengintegrasikannya ke dalam jaringan pemantauan lalu lintas yang lebih canggih untuk mendukung pengambilan keputusan otomatis dan mendukung realisasi konsep kota pintar (smart city).

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Dalam banyak kasus, ucapan terima kasih kepada sponsor dan dukungan finansial.

REFERENSI

- (Barhatte M.AU3 - Miglani H.AU4 - Deo P., 2023)
- Barhatte, A., Gupta, M., Miglani, H., & Deo, P. (2023). Real-time traffic control and monitoring. *e-Prime Advances in Electrical Engineering*.
- (Lin J. Y., 2022) Lin, C. J., & Jhang, J. Y. (2022). Intelligent traffic-monitoring system based on YOLO and convolutional fuzzy neural networks. *IEEE Access*.
- (Azimjonov A., 2021) Azimjonov, J., & Özmen, A. (2021). A real-time vehicle detection and a novel vehicle tracking system. *Advanced Engineering Informatics*.
- (Prathap D. S.AU3 - Bansal R. K., 2022)
- Prathap, K. S. V., Reddy, D. S., & Bansal, R. K. (2022). Intelligent Traffic Light System Using YOLO. *Springer Innovations in Signal Processing and Communication*.
- (Rahman M. A.AU3 - Ayman U., 2025)
- Rahman, M. M., Rahim, M. A., & Ayman, U. (2025). Real-Time Vehicle Type Detection and Counting for Emission Monitoring.
- (Deoghare Y.AU3 - Gaikwad S., 2025)
- Deoghare, S. U., Dhond, Y., Gaikwad, S., et al. (2025). Dynamic Traffic Signal Optimization Using YOLO. *IEEE IoT and Smart City*.
- (Moumen N., 2024) Moumen, I., & Rafalia, N. (2024). A Next-Generation Platform for Law Enforcement Vehicle Detection. *IEEE. Link*
- (Goud, 2025) Goud, M. D. (2025). Real-Time Object Detection and Voice-Based Recognition Using YOLO and Webcam. *EAJET*.
- (Azevedo, 2022) Azevedo, J. P. (2022). YOLO Neural Networks on Reconfigurable Logic for Vehicle Traffic Control. *ProQuest Dissertation*.
- (Mulyana & Rofik, 2022) Mulyana dan A. Rofik, "Implementasi Deteksi Real Time Klasifikasi Jenis Kendaraan di Indonesia Menggunakan Metode YOLOv5," *Jurnal Pendidikan Tambusai*, vol. 6, no. 3, pp. 14512–14520, 2022.
- (Michael, 2024) M. Michael, "Identifikasi Kendaraan Beroda Menggunakan Algoritma YOLOv5," *Jurnal Buana Informatika*, vol. 13, no. 1, pp. 45–53, 2024.
- (Zhang et al., 2024) Y. Zhang, J. Xu, dan Q. Li, "Research on YOLOv5 Vehicle Detection and Positioning System Based on Binocular Vision," *World Electric Vehicle Journal*, vol. 15, no. 2, p. 62, 2024.
- (Zhang et al., 2022) Y. Zhang, J. Liu, dan W. Zhang, "Real-Time Vehicle Detection Based on Improved YOLOv5," *Sustainability*, vol. 14, no. 19, p. 12274, 2022.
- (Geetha, 2024) S. Geetha, "Comparing YOLOv5 Variants for Vehicle Detection: A Performance Analysis," *arXiv preprint*, arXiv:2408.12550, 2024.