



SPECTA Journal of Technology

E-ISSN : 2622-9099

P-ISSN : 2549-2713

Homepage jurnal: <https://journal.itk.ac.id/index.php/sjt>



Deteksi Kepadatan Lalu Lintas pada CCTV Publik Pemerintah Kota Malang Menggunakan YOLOv8

Andi Surya¹, Ida Wahyuni^{2*}

^{12*} Program Studi Teknik Informatika, Fakultas Teknologi dan Desain, Institut Teknologi dan Bisnis Asia, Malang, Indonesia

*Corresponding email: idawahyuni@asia.ac.id

Received: 22/April/2025
Accepted: 15/August/2025

Revised: 15/July/2025
Published: 21/August/2025

To cite this article:

Surya, A & Wahyuni, I (2025). Deteksi Kepadatan Lalu Lintas pada CCTV Publik Pemerintah Kota Malang Menggunakan YOLOv8. *SPECTA Journal of Technology*, 9(2), 136 - 149. [10.35718/specta.v9i2.8481367](https://doi.org/10.35718/specta.v9i2.8481367)

Abstract

The increasing traffic congestion caused by the growing number of motor vehicles not only leads to reduced productivity but also contributes to economic losses and excessive fuel consumption, including on Jalan Ahmad Yani, Malang City. Until now, traffic monitoring has still been carried out manually, which requires significant time and human resources and can potentially result in greater economic losses if congestion is not addressed promptly. Therefore, an automated system is needed to detect and monitor traffic density. This study aims to develop a traffic density detection system using the YOLOv8 object detection model with video data from public CCTV provided by the Malang City Government. The method consists of several stages: data collection, data preprocessing, model development using YOLOv8, model testing, and traffic density calculation based on the number of detected vehicles. Model development was carried out by exploring various hyperparameter configurations, including epoch, batch size, learning rate, optimizer, imgsz, and early stopping mechanism to achieve optimal performance. The model was evaluated during training using Recall and Precision metrics, resulting in a Recall of 0.901 and a Precision of 0.811. Model testing was performed using CCTV video recordings by calculating the accuracy based on a comparison between the number of vehicles detected by the system and manual vehicle counts, yielding an average accuracy of 78,07%. These results indicate that the model is capable of detecting vehicles with sufficient accuracy in real-world conditions. This system can serve as a foundation for the development of automated traffic monitoring systems that utilize visual data from CCTV.

Keywords: YOLOv8, vehicle detection, CCTV, traffic density, Malang City.

Abstrak

Kemacetan lalu lintas yang meningkat akibat pertumbuhan jumlah kendaraan bermotor tidak hanya berdampak pada penurunan produktivitas, tetapi juga berkontribusi terhadap kerugian ekonomi dan borosnya konsumsi bahan bakar, termasuk di Jalan Ahmad Yani, Kota Malang. Selama ini, pengamatan lalu lintas masih dilakukan secara manual yang memerlukan waktu dan sumber daya manusia yang besar serta berpotensi menimbulkan kerugian ekonomi jika kemacetan tidak segera ditangani. Oleh karena itu, diperlukan sistem yang mampu mendeteksi dan memantau kepadatan lalu lintas secara otomatis. Penelitian ini bertujuan untuk membangun sistem deteksi kepadatan lalu lintas menggunakan model deteksi objek YOLOv8 dengan data video dari CCTV publik Pemerintah Kota Malang. Metode yang digunakan meliputi tahapan *data collection*, *data preprocessing*,

pembuatan model menggunakan YOLOv8, *model testing*, dan perhitungan tingkat kepadatan berdasarkan jumlah kendaraan yang terdeteksi. Pembuatan model dilakukan dengan mengeksplorasi beberapa *hyperparameter* seperti *epoch*, *batch size*, *learning rate*, *optimizer*, *imgsz*, dan mekanisme *early stopping* untuk memperoleh performa terbaik. Model dievaluasi selama pelatihan menggunakan metrik Recall dan Precision, dengan hasil Recall sebesar 0,901 dan Precision sebesar 0,811. *Model testing* dilakukan menggunakan rekaman video CCTV dengan menghitung akurasi berdasarkan perbandingan jumlah kendaraan yang terdeteksi oleh sistem terhadap hasil perhitungan manual, yang menghasilkan rata-rata akurasi sebesar 78,07%. Hasil tersebut menunjukkan bahwa model mampu mendeteksi kendaraan dengan cukup akurat dalam kondisi nyata. Sistem ini dapat digunakan sebagai dasar bagi pengembangan sistem pemantauan lalu lintas otomatis yang memanfaatkan data visual dari CCTV.

Kata Kunci: CCTV, deteksi kendaraan, kepadatan lalu lintas, Kota Malang, YOLOv8.

1. Pendahuluan

Pertumbuhan jumlah kendaraan bermotor di Indonesia terus meningkat melebihi laju pertumbuhan penduduk. Menurut Badan Pusat Statistik Indonesia (2024), jumlah penduduk Indonesia pada tahun 2023 mencapai 278,6 juta jiwa, meningkat sebesar 1,06% dari tahun sebelumnya. Pada periode yang sama, jumlah kendaraan bermotor naik sebesar 3,47%, dari 148,2 juta unit menjadi 153,4 juta unit. Pertumbuhan jumlah kendaraan bermotor yang signifikan ini berdampak pada berbagai permasalahan, salah satunya adalah sering terjadi kemacetan lalu lintas. Di sisi lain, kemacetan memberikan dampak yang kurang baik seperti hilangnya pendapatan akibat jam kerja yang hilang, mengurangi jam belajar bagi pelajar, hingga pemborosan bensin (Ashad Sadiq, 2021).

Salah satu kota besar yang mengalami kemacetan lalu lintas yang signifikan adalah Kota Malang. Kemacetan di kota ini menjadi masalah yang krusial karena mengakibatkan kerugian yang cukup besar, diperkirakan mencapai Rp 383 miliar per tahun (Ramadhani et al., 2021). Salah satu ruas jalan di Kota Malang yang mengalami kemacetan adalah ruas Jalan Ahmad Yani di Kecamatan Blimbing. Hal ini disebabkan karena ruas jalan tersebut sudah berada pada titik jenuh dan juga perlu adanya manajemen kapasitas jalan serta optimasi lalu lintas untuk mengurangi beban ruas jalan tersebut (Kiftiah, 2019). Namun, penelitian terdahulu (Kiftiah, 2019) memiliki kekurangan, yaitu pengamatan dan perhitungan kendaraan yang melintas masih dilakukan secara manual. Hal ini memerlukan waktu serta sumber daya manusia yang lebih banyak dan dapat mengakibatkan kerugian ekonomi yang lebih besar jika kemacetan tidak segera ditangani. Oleh karena itu, teknologi deteksi kendaraan untuk memberikan informasi kepadatan lalu lintas secara otomatis sangat dibutuhkan untuk mengurangi kemacetan serta mengurangi ketergantungan pada pengamatan secara visual dan manual.

Salah satu penelitian terdahulu yang pernah dilakukan untuk mendeteksi kendaraan pernah dilakukan dengan menggunakan hasil rekaman CCTV pada Kota Nganjuk (Mukhlis et al., 2024). Penelitian yang menggunakan YOLOv8 tersebut menghasilkan nilai akurasi yang cukup baik sebesar 93%. Penelitian lain untuk deteksi kendaraan menggunakan media CCTV juga pernah dilakukan, dengan menggunakan media rekaman CCTV Kota Banyumas dan Kota Bandung (Khatami, 2022). Penelitian yang menggunakan YOLOv3 ini menghasilkan nilai rata-rata akurasi lebih dari 80%. Dari beberapa penelitian ini, dapat disimpulkan bahwa YOLO merupakan salah satu metode yang dapat digunakan untuk mendeteksi kendaraan melalui media CCTV dengan cukup baik. Saat ini versi YOLO sudah sangat berkembang, salah satunya YOLOv8 yang merupakan algoritma deteksi objek turunan dari *convolutional neural network* (CNN) yang memiliki kecepatan deteksi dan akurasi yang tinggi (Hussain, 2024). YOLOv8 melakukan deteksi objek melalui satu tahapan, dengan membagi input menjadi beberapa segmen dan melakukan prediksi letak dan jenis objek pada segmen tersebut. YOLOv8 memiliki keunggulan dalam kecepatan, akurasi, dan penggunaan parameter deteksi yang lebih efisien daripada versi YOLO sebelumnya.

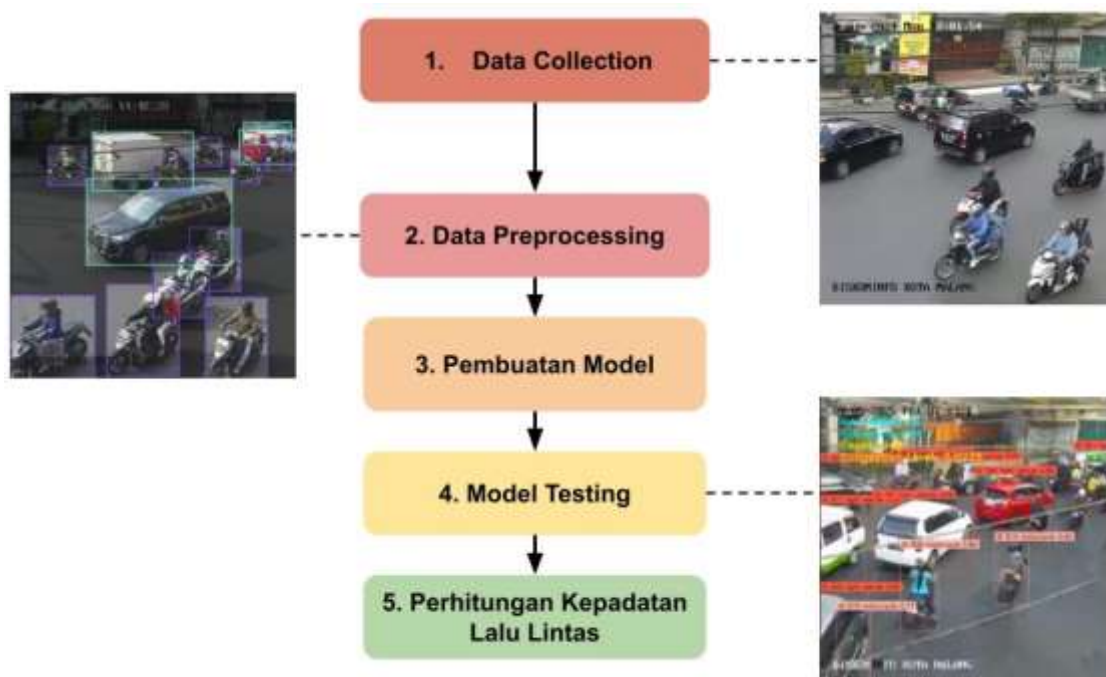
Meskipun beberapa penelitian terdahulu telah menggunakan YOLO dalam mendeteksi kendaraan menggunakan data visual dari CCTV, penelitian tersebut masih terbatas pada penggunaan YOLOv3 dan hanya menguji model yang telah dibuat menggunakan rekaman gambar statis, bukan rekaman video

secara utuh. Selain itu, belum ada penelitian yang menerapkan YOLOv8 untuk deteksi kepadatan lalu lintas di Kota Malang, khususnya dengan memanfaatkan rekaman video dari CCTV publik yang tersedia. Padahal, penggunaan video lebih merepresentasikan kondisi lalu lintas yang nyata karena dapat merekam pergerakan kendaraan serta adanya gangguan visual.

Oleh karena itu, keberadaan infrastruktur CCTV di Kota Malang yang telah dipasang oleh pemerintah dan dapat diakses oleh masyarakat menjadi potensi yang belum dimanfaatkan secara optimal. CCTV tersebut tersebar di beberapa titik strategis, terutama pada ruas jalan utama dan rawan kemacetan seperti di Jalan Ahmad Yani. Data visual dari CCTV ini dapat digunakan sebagai sumber informasi untuk menganalisis kepadatan kendaraan. Dengan memanfaatkan kondisi tersebut, penelitian ini menggunakan rekaman video dari CCTV Kota Malang pada ruas jalan tersebut. Metode yang digunakan untuk deteksi dan klasifikasi kendaraan adalah algoritma YOLOv8 yang mempunyai performa lebih baik dan kecepatan lebih unggul daripada algoritma YOLOv3. Secara umum, alur penelitian meliputi lima tahap utama, yaitu *data collection*, *data preprocessing*, pembuatan model, *model testing*, dan perhitungan kepadatan lalu lintas. Pada tahap pembuatan model, dilakukan penyesuaian terhadap konfigurasi *hyperparameter*, seperti pengaturan nilai *batch size*, *learning rate*, *epoch*, jenis *optimizer*, ukuran gambar (*imgsz*), serta nilai *patience*. Penyesuaian bertujuan untuk melihat konfigurasi terbaik saat tahapan tersebut. Penelitian ini bertujuan untuk membangun sistem deteksi kepadatan lalu lintas berbasis YOLOv8, yang dapat dimanfaatkan oleh publik, khususnya pemerintah daerah, dalam memantau kepadatan lalu lintas secara otomatis. Sistem ini diharapkan mampu menyajikan data kepadatan lalu lintas yang akurat dan cepat dibandingkan dengan pengamatan secara visual dan manual, sehingga mendukung pengambilan keputusan yang tepat terkait pengelolaan lalu lintas dan penanggulangan kemacetan di Kota Malang.

2. Metodologi

Metodologi yang digunakan terbagi dalam lima tahap utama, yaitu: *data collection*, *data preprocessing*, pembuatan model, *model testing*, dan perhitungan kepadatan lalu lintas. Tahapan metodologi tersebut ditunjukkan pada Gambar 1. *Dataset* berupa video akan dikumpulkan dari hasil rekaman kamera CCTV yang terpasang di Jalan Ahmad Yani Kota Malang. Semua program dan *script* menggunakan bahasa pemrograman Python, karena memiliki *libraries* atau *package* yang berhubungan dengan ilmu *data science* maupun *machine learning* (Abdul Kadhar & Anand, 2021).



Gambar 1: Diagram Alur Metodologi Penelitian

2.1. Data Collection

Pengumpulan *dataset* video dilakukan dengan merekam kamera CCTV yang dipasang oleh Pemerintah Kota Malang, tepatnya di Jalan Ahmad Yani. Perekaman dilakukan secara otomatis menggunakan *script* Python yang terintegrasi dengan *livestream* CCTV di jalan tersebut. Perekaman dilakukan pada tiga rentang waktu yang berbeda untuk menangkap variasi tingkat lalu lintas, yaitu pagi (07:00 WIB), siang (12:00 WIB), dan sore hari (17:00 WIB), masing-masing berdurasi dua menit. Lima video digunakan dalam penelitian ini, dengan tiga video dialokasikan untuk pembuatan model dan 2 video untuk *model testing*.

2.2. Data Preprocessing

Data preprocessing dilakukan karena merupakan langkah penting sebelum algoritma atau model dapat membaca dan menganalisis data seperti yang dipaparkan oleh (Maharana et al., 2022). *Data preprocessing* dilakukan menggunakan *website* Roboflow, yang mempermudah proses anotasi serta *preprocessing* *dataset* menurut (Hayati et al., 2023). Ketiga video yang dialokasikan untuk pembuatan model akan dikonversi menjadi kumpulan gambar atau *frame* menjadi 1000 gambar. Selanjutnya, dilakukan proses anotasi pada setiap *frame* dengan memberikan *bounding box* serta *label* atau *class* pada objek kendaraan, dengan rincian *class* seperti pada Tabel 1. Setelah itu, setiap gambar diubah ukurannya menjadi 640 x 640 piksel agar sesuai dengan format input model YOLOv8. *Dataset* kemudian dibagi menjadi dua bagian, yaitu 80% untuk *training data* dan 20% untuk *validation data*.

Tabel 1: Pembagian Class Kendaraan

Class	Jenis Kendaraan
Motorcycle (MC)	Sepeda Motor
Light Vehicle (LV)	Sedan, Jip, Pick Up, Truck Kecil
Heavy Vehicle (HV)	Truk, Bus, Truk dengan 2 gandar atau lebih

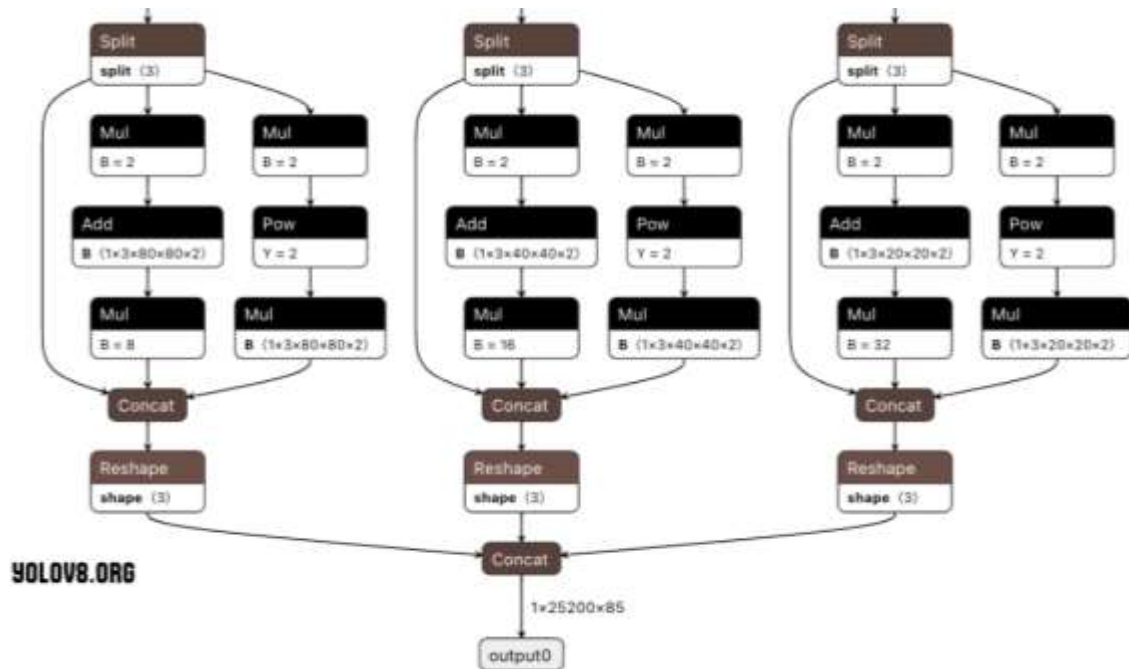
Sumber (Direktorat Jenderal Bina Marga, 2023), (Kiftiah, 2019)

2.3. Pembuatan Model

Penelitian ini bertujuan untuk membangun sistem deteksi kepadatan lalu lintas berdasarkan deteksi kendaraan pada video CCTV menggunakan model YOLOv8. Model deteksi kendaraan dibuat menggunakan arsitektur YOLOv8 dari *library* Ultralytics, yang diimplementasikan dengan bahasa pemrograman Python versi 3.11.2. YOLO merupakan salah satu algoritma *object detection* dalam *computer vision* yang berbasis CNN (Huang & Le, 2021). Berbeda dengan pendekatan algoritma *object detection* lain seperti Faster R-CNN, YOLO melakukan klasifikasi dan lokalisasi objek dari keseluruhan input dalam satu kali proses jaringan CNN, membuat algoritma ini sebagai *object detection* yang *real-time*. Sejak diperkenalkan tahun 2016, YOLO telah berkembang menjadi beberapa versi seperti YOLOv5 dan YOLOv8 (Hussain, 2024). Detail arsitektur dari YOLOv8 dapat dilihat pada Gambar 2.

YOLOv8 merupakan generasi terbaru dari keluarga algoritma YOLO (You Only Look Once) yang berfungsi untuk mendeteksi objek secara *real-time* dalam gambar atau video. Metode ini mengadopsi arsitektur deep learning yang menggabungkan kecepatan dan akurasi tinggi dalam satu framework tunggal. YOLOv8 mengoptimalkan proses pelatihan dan inferensi dengan menggunakan teknik pemrosesan citra yang lebih efisien serta arsitektur jaringan konvolusional yang lebih ramping namun tetap mampu menangkap fitur-fitur penting secara detail (Hussain, 2024).

Pembuatan model dilakukan dalam beberapa tahapan, yaitu *model training* dan *model evaluation* dengan mengeksplorasi kombinasi *hyperparameter*, menyimpan hasil *model evaluation*, mengurutkan hasil *model evaluation* berdasarkan nilai Recall dan Precision terbaik, serta memilih model terbaik yang telah diurutkan. Beberapa kombinasi *hyperparameter* dieksplorasi untuk mendapatkan model dengan performa terbaik. *Hyperparameter* yang digunakan terdiri dari *epoch*, *batch*, *learning rate*, *optimizer*, *patience*, serta *imgsz* dengan rincian nilai parameter terdapat pada Tabel 2.



Gambar 2: Arsitektur YOLOv8

Source: Jane Torres, 2024

Tabel 2: Kombinasi *Hyperparameter* Pembuatan Model

Nama Parameter	Nilai
Epoch	10, 20, 50, 100, 200
Batch Size	16, 32, 64
Learning rate	0.01, 0.001, 0.0001
Optimizer	Adam, RMSProp, SGD
Patience	10
Image Size	640

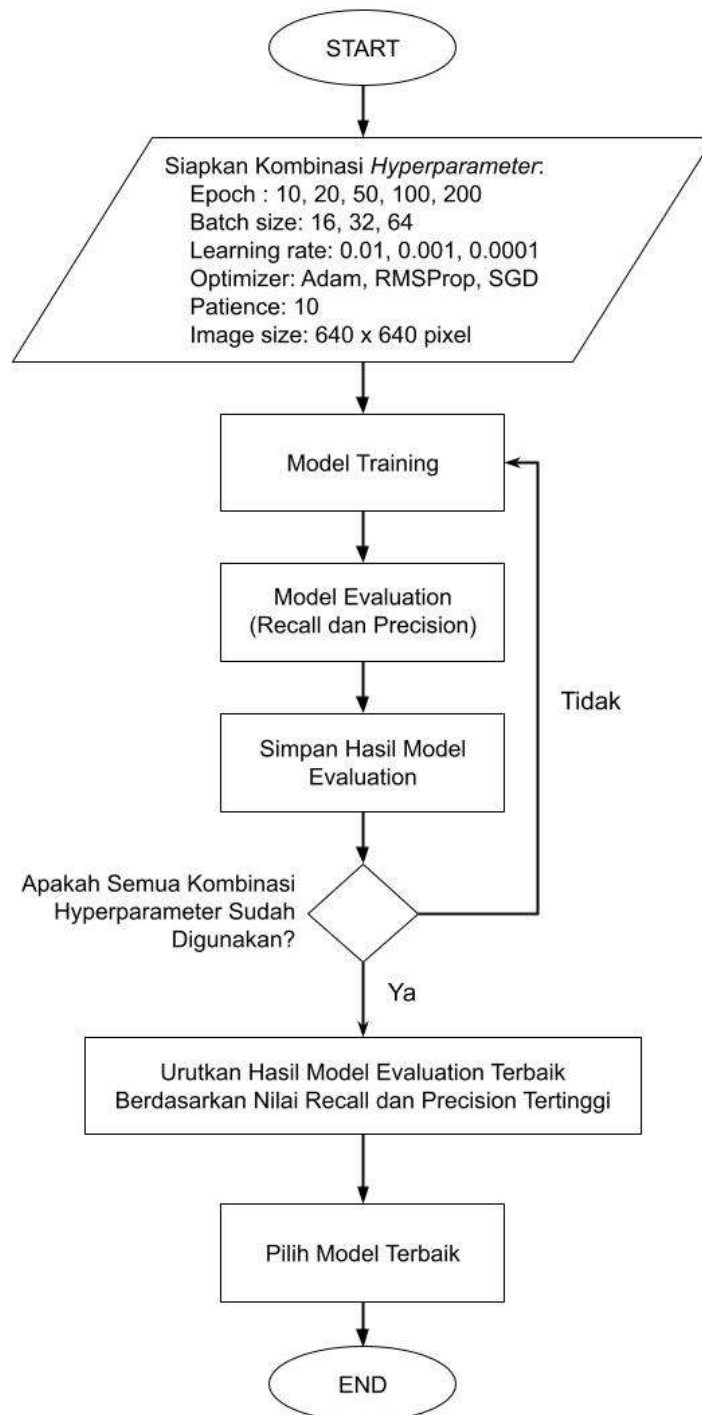
Pada Tabel 2, parameter epoch dengan nilai 10, 20, 50, 100, 200 digunakan untuk menentukan jumlah iterasi pelatihan yang dilakukan. Batch size dengan nilai 16, 32, 64 mengatur jumlah data yang diproses dalam satu iterasi pelatihan. Learning rate dengan nilai 0.01, 0.001, 0.0001 berfungsi mengatur seberapa besar perubahan bobot model pada setiap langkah pelatihan, sedangkan optimizer dengan nilai Adam, RMSProp, SGD menentukan metode terbaik dalam memperbarui bobot-bobot tersebut. Selain itu, digunakan mekanisme early stopping dengan parameter patience bernilai 10 untuk menghentikan pelatihan apabila tidak terjadi peningkatan performa model selama sejumlah 10 epoch berturut-turut. Sedangkan ukuran gambar (*image size*) digunakan untuk menentukan ukuran gambar input yang dimasukkan ke dalam model selama pelatihan yaitu 640x640 pixel. Detail dari tahapan ini ditunjukkan pada Gambar 3.

Model evaluation selama proses *model training* berfokus pada 2 metrik utama, yaitu Recall dan Precision. Kedua metrik yang ditunjukkan pada Persamaan 1 dan Persamaan 2 ini merupakan kriteria evaluasi model *object detection* (Zhang et al., 2024). Pada Persamaan 1 dan Persamaan 2, TP (*True Positive*) merepresentasikan jumlah deteksi yang benar, FP (*False Positive*) merepresentasikan jumlah deteksi yang salah, dan FN (*False Negative*) merepresentasikan jumlah deteksi yang salah namun terlewat oleh detektor.

$$Recall = \frac{TP}{TP + FP} \quad (1)$$

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

Hasil dari iterasi *model training* dan *model validation* diurutkan berdasarkan kombinasi nilai *Recall* dan *Precision* terbaik, namun memprioritaskan nilai *Recall* tertinggi. Kriteria ini digunakan karena dalam konteks deteksi kendaraan, kemampuan model dalam mendeteksi sebanyak mungkin kendaraan (*Recall*) lebih penting daripada hanya sekedar akurat (*Precision*). Jika kendaraan tidak terdeteksi (*False Negative*), maka akan menyebabkan kesalahan dalam perhitungan tingkat kepadatan lalu lintas karena jumlah kendaraan yang terdeteksi melintas oleh model kurang dari sebenarnya.



Gambar 3: Diagram Alur Pembuatan Model

2.4. Pengujian Model

Pengujian model dilakukan untuk mengevaluasi performa model YOLOv8 yang telah diimplementasikan dalam mendeteksi jumlah kendaraan dari rekaman video CCTV secara otomatis. Evaluasi dilakukan dengan membandingkan hasil deteksi model terhadap jumlah kendaraan hasil perhitungan manual pada video yang sama. Tujuan dari tahap ini untuk mengetahui seberapa akurat model dalam mendeteksi jumlah kendaraan dalam video. Akurasi model pada tahap pengujian model dihitung menggunakan *accuracy* yang dapat dilihat pada Persamaan 3.

$$Accuracy = \left(\frac{Detected\ Object}{Actual\ Object} \right) \times 100\% \quad (3)$$

Pengujian dilakukan menggunakan dua sampel video dari CCTV Kota Malang pada ruas Jalan Ahmad Yani, masing-masing diambil pada hari Kamis dan hari Minggu (Kiftiah, 2019). Kedua hari tersebut dipilih untuk merepresentasikan kondisi jalan saat hari kerja dan akhir pekan. Dua sampel video tersebut diambil dalam enam rentang waktu berbeda: pukul 06:00–07:00, 07:00–08:00, 11:00–12:00, 12:00–13:00, 16:00–17:00, dan 17:00–18:00. Untuk setiap segmen waktu, jumlah kendaraan yang terdeteksi oleh model dicatat, lalu dibandingkan dengan hasil hitung manual guna memperoleh akurasi sistem. Dengan pendekatan ini, pengujian dapat menilai sejauh mana model yang telah dibuat dapat memberikan informasi kepadatan lalu lintas dibandingkan dengan pengamatan secara manual (Kiftiah, 2019).

2.5. Perhitungan Kepadatan Lalu Lintas

Tahap terakhir adalah menentukan tingkat kepadatan lalu lintas. Kepadatan lalu lintas terjadi karena arus lalu lintas melebihi kapasitas suatu jalan. Jika dilihat dari waktu tempuh, kepadatan terjadi karena arus lalu lintas normal terganggu oleh banyaknya kendaraan yang menyebabkan waktu tempuh menjadi bertambah (Afrin & Yodo, 2020) (Aftabuzzaman, 2007).

Kepadatan lalu lintas ditentukan berdasarkan nilai derajat kejenuhan (*Dk*). Perhitungan nilai tersebut dilakukan dengan pembagian rasio arus lalu lintas (*Q*) terhadap kapasitas jalan (*c*) (Direktorat Jenderal Bina Marga, 2023). Rumus untuk menghitung rasio arus lalu lintas ditunjukkan pada Persamaan 4.

$$Q = (1 \times LV) + (1,3 \times HV) + (0,4 \times MC) \quad (4)$$

Keterangan:

Q = Rasio arus lalu lintas
LV = Light Vehicle
HV = Heavy Vehicle
MC = Motorcycle

Nilai kapasitas jalan (*c*) ditetapkan sebesar 6649 satuan mobil penumpang / jam, berdasarkan penelitian (Kiftiah, 2019) di lokasi yang sama. Derajat kejenuhan (*Dk*) dihitung menggunakan Persamaan 5 (Direktorat Jenderal Bina Marga, 2023). Nilai *Dk* yang diperoleh digunakan untuk mengklasifikasikan tingkat kepadatan dalam lima kategori dengan rincian seperti pada Tabel 3. Pada Tabel 3 ditunjukkan bahwa semakin besar nilai derajat kejenuhan, maka kondisi lalu lintas akan semakin macet dengan penjelasan pada kolom karakteristik operasi terkait. Sebaliknya, semakin kecil nilai derajat kejenuhan, maka kondisi lalu lintas semakin lancar.

$$Dk = \frac{Q}{c} \quad (5)$$

Tabel 3: Tingkat Kepadatan Lalu Lintas

Tingkat Pelayanan	Karakteristik Operasi Terkait	Nilai Derajat Kejenuhan
A	Kondisi arus bebas dengan kecepatan tinggi, pengemudi dapat memilih kecepatan yang diinginkan tanpa hambatan.	0-0,20
B	Arus stabil, tetapi kecepatan operasi mulai dibatasi oleh kondisi lalu lintas, pengemudi memilih kebebasan yang cukup untuk memilih kecepatan.	0,20-0,44
C	Arus stabil, tetapi kecepatan dan gerak dikendalikan, pengemudi dibatasi dalam memilih kecepatan.	0,45-0,74
D	Arus mendekati tidak stabil, kecepatan masih dikendalikan.	0,75-0,84
E	Arus lalu lintas mendekati atau berada pada kapasitas arus tidak stabil, kecepatan terkadang berhenti.	0,85-1,00
F	Arus yang dipaksakan atau macet kecepatan rendah, volume dibawah kapasitas. Antrian panjang dan terjadi hambatan-hambatan yang besar.	>1,00

Sumber: (Fatoni & Asmaroni, 2022)

3. Hasil dan Pembahasan

3.1. Data Collection

Proses *data collection* dimulai dengan mengakses *website* System Monitoring CCTV Kota Malang (http://cctv.malangkota.go.id/cameras_dua). Titik lokasi yang dipilih terletak di Jalan Ahmad Yani Kota Malang. *Link streaming* dari CCTV tersebut diperoleh dengan menggunakan fitur *Inspect Element* pada *browser* untuk menemukan *Request URL* yang mengarah ke sumber video CCTV tersebut. *Link streaming* yang diperoleh digunakan sebagai input kode Python untuk merekam video selama dua menit dari kamera CCTV tersebut. Total tiga video dikumpulkan dari lokasi yang sama sebagai dasar *dataset* untuk pembuatan model deteksi kendaraan.

3.2. Data Preprocessing

Setelah *data collection* selesai, dilakukan *data preprocessing* untuk mengubah video menjadi *dataset* gambar. Konversi dilakukan di *website* Roboflow dengan mengambil 3 *frame* per detik dari durasi video berjumlah 120 detik, menghasilkan total 361 gambar per video, atau 1083 gambar dari 3 video yang dikumpulkan untuk pembuatan model. Selanjutnya, dilakukan anotasi secara manual pada tiap gambar dengan memberikan *bounding box* berdasarkan tiga *class* kendaraan: Light Vehicle, Heavy Vehicle, dan Motorcycle. Hasil anotasi disimpan dalam *file* berformat .txt. Pemberian *bounding box* menggunakan warna yang berbeda dengan Light Vehicle berwarna hijau toska, Heavy Vehicle berwarna merah, dan Motorcycle berwarna ungu. Contoh anotasi ditunjukkan pada Gambar 4.



Gambar 4: Anotasi Gambar Pada Website Roboflow

3.3. Pembuatan Model

Pembuatan model dilakukan pada *website* Kaggle menggunakan bahasa Python 3.11.2 dan YOLOv8 dari *library* Ultralytics. *Dataset* yang sudah di-*preprocessing* dari *website* Roboflow diakses menggunakan *code snippet* yang telah disediakan oleh *website* tersebut. Eksplorasi kombinasi *hyperparameter* menghasilkan 135 kali iterasi *model training* dan *model validation* yang berisi nilai Recall serta Precision. Hasil ini diurutkan berdasarkan kombinasi nilai Recall dan Precision terbaik, namun nilai Recall tertinggi menjadi prioritas. Tabel 4 menunjukkan rincian hasil 10 model terbaik berdasarkan nilai Recall dan Precision. Model terbaik memiliki performa dengan nilai Recall sebesar 0.904 dan Precision sebesar 0.811. Hasil ini diperoleh dari kombinasi *epoch* 100, *batch size* 16, *learning rate* 0.01, dan *optimizer* menggunakan SGD.

Tabel 4: Hasil 10 Model Terbaik Berdasarkan Nilai Recall dan Precision

No	Epochs	Batch Size	Learning Rate	Optimizer	Recall	Precision
1	100	16	0.01	SGD	0.904	0.811
2	100	16	0.001	Adam	0.900	0.881
3	50	16	0.01	Adam	0.880	0.836
4	50	16	0.001	Adam	0.874	0.855
5	100	64	0.01	Adam	0.861	0.908
6	100	64	0.001	Adam	0.859	0.853
7	100	64	0.01	SGD	0.858	0.856
8	200	64	0.001	Adam	0.853	0.929
9	200	32	0.001	Adam	0.844	0.894
10	50	64	0.001	Adam	0.839	0.838

3.4. Pengujian Model

Setelah model dengan performa dan kombinasi *hyperparameter* terbaik dipilih, dilakukan *model testing* menggunakan dua sampel video yang direkam pada hari Kamis, 22 Mei 2025 dan hari Minggu, 25 Mei

2025. Masing-masing video dibagi ke dalam enam segmen waktu yang merepresentasikan kondisi lalu lintas pagi, siang, dan sore hari. Hasil deteksi model dibandingkan dengan hasil hitung manual sebagai acuan, untuk memperoleh nilai akurasi pada tiap segmen waktu. Tahapan ini menambahkan *bounding box* berwarna hijau agar model dapat menghitung kendaraan yang terdeteksi melewati area tersebut. Rincian hasil *model testing* ditunjukkan pada Tabel 5 dan Tabel 6.

Tabel 5: Hasil Akurasi Pengujian Model pada Video Pertama (Kamis)

Waktu	Jumlah Kendaraan Terdeteksi	Jumlah Kendaraan Perhitungan Manual	Akurasi (%)
06:00-07:00	7176	9086	78.98
07:00-08:00	6752	8885	75.99
11:00-12:00	6160	8273	74.46
12:00-13:00	4296	6074	71.79
16:00-17:00	5056	6841	73.91
17:00-18:00	6007	7690	78.98

Tabel 6: Hasil Akurasi Pengujian Model pada Video Kedua (Minggu)

Waktu	Jumlah Kendaraan Terdeteksi	Jumlah Kendaraan Perhitungan Manual	Akurasi (%)
06:00-07:00	3959	4833	81.92
07:00-08:00	4230	5178	81.69
11:00-12:00	5173	6294	82.19
12:00-13:00	3404	4217	80.72
16:00-17:00	4739	5879	80.61
17:00-18:00	5178	6600	78.45

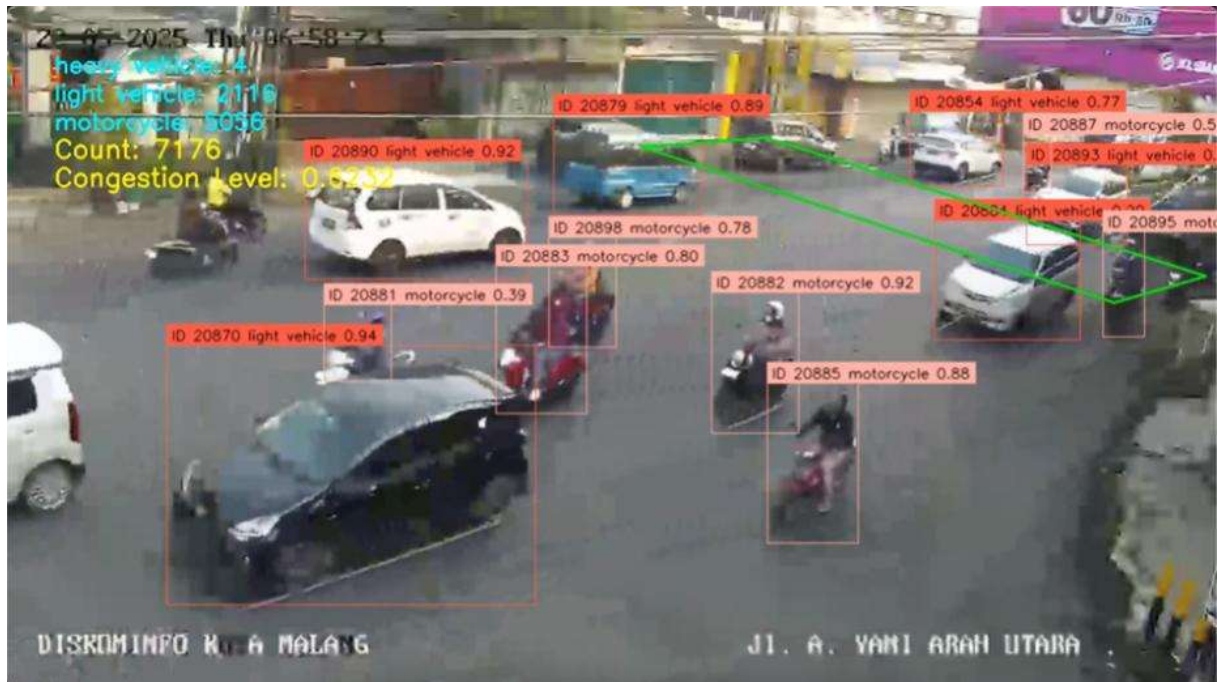
Pada *model testing* hari Kamis pada Tabel 5, model menunjukkan akurasi bervariasi, dengan nilai tertinggi sebesar 78,98% pada rentang waktu 06:00–07:00 dan 17:00–18:00, serta nilai terendah sebesar 71,79% pada pukul 12:00–13:00. Secara umum, akurasi cenderung menurun pada waktu siang hari, yang dipengaruhi oleh kondisi pencahayaan yang akibat cuaca mendung dan kepadatan kendaraan yang padat, sehingga memengaruhi kualitas deteksi.

Sebaliknya, *model testing* pada hari Minggu pada Tabel 7 menunjukkan performa model yang lebih stabil dengan akurasi rata-rata lebih tinggi dibandingkan hari Kamis. Nilai tertinggi dicapai pada pukul 11:00–12:00 dengan akurasi 82,19%, sedangkan nilai terendah terjadi pada pukul 17:00–18:00 sebesar 78,45%. Performa yang lebih baik ini karena volume lalu lintas pada hari Minggu relatif lebih rendah dan kepadatan kendaraan lebih renggang, sehingga memudahkan proses deteksi oleh model.

Untuk lebih jelasnya, ditunjukkan Gambar 5 yang memperlihatkan bahwa model mampu mendeteksi kendaraan jenis Light Vehicle dengan akurasi mencapai 94%. Hal ini karena kendaraan tersebut tidak tertutupi oleh kendaraan lain. Model juga mampu untuk mendeteksi kendaraan jenis Motorcycle hingga akurasi sebesar 92%, karena kendaraan tersebut juga tidak tertutupi oleh kendaraan lain, serta memiliki jarak yang tidak terlalu dekat dengan kendaraan. Namun, ada kecenderungan model tidak dapat mendeteksi kendaraan jenis Motorcycle karena berada pada area gelap akibat bayangan akibat pohon. Selain itu, model tidak mampu mendeteksi Motorcycle yang berada dekat dengan Motorcycle lain. Hal ini karena kualitas citra dari CCTV tidak terlalu jelas, sehingga model kesulitan dalam mendeteksi kendaraan tersebut.

Selain itu, berdasarkan Gambar 6, model menunjukkan kemampuan yang cukup baik dalam mendeteksi kendaraan jenis Motorcycle yang tidak berada dalam jarak dekat maupun tidak tertutup oleh kendaraan lain, dengan akurasi mencapai 87%. Selain itu, model mampu mendeteksi Light Vehicle dengan akurasi

hingga 94%. Tingginya tingkat akurasi model dalam mendeteksi berbagai jenis kendaraan diakibatkan karena jarak antar kendaraan yang tidak rapat, sehingga memudahkan proses deteksi kendaraan.



Gambar 5: Visualisasi Pengujian Model pada Video Pertama (Kamis)



Gambar 6: Visualisasi Pengujian Model pada Video Kedua (Minggu)

Berdasarkan hasil *model testing* pada dua hari berbeda, yaitu hari Kamis dan hari Minggu, model menunjukkan rata-rata akurasi sebesar 78,07%. Angka ini menunjukkan performa deteksi kendaraan yang cukup baik dalam kondisi lalu lintas nyata. Akurasi lebih tinggi cenderung terjadi pada hari Minggu yang lalu lintasnya lebih lengang, dibandingkan Kamis yang merupakan hari kerja dengan volume lalu lintas yang lebih padat. Akurasi ini menunjukkan bahwa model terpilih memiliki akurasi yang cukup baik dalam mendeteksi kendaraan. Sehingga, pengamatan dan perhitungan kendaraan untuk menentukan nilai kepadatan lalu lintas dapat dilakukan secara otomatis dengan menggunakan model YOLOv8, tanpa perlu melakukan perhitungan dan pengamatan secara manual.

3.5. Perhitungan Tingkat Kepadatan Lalu Lintas

Setelah diperoleh jumlah kendaraan hasil deteksi model, dilakukan perhitungan derajat kejenuhan (Dk) untuk menentukan tingkat kepadatan lalu lintas. Pertama, dilakukan perhitungan rasio arus lalu lintas (Q) dilakukan berdasarkan jumlah kendaraan yang terdeteksi oleh model menggunakan Persamaan 2, dibagi dengan nilai kapasitas jalan (c). Perhitungan tingkat kepadatan lalu lintas (Dk) dilakukan pada video pertama (Kamis) dan video kedua (Minggu) pada semua *timeline*. Nilai tersebut kemudian dipetakan ke dalam tingkatan kepadatan lalu lintas berdasarkan

Tabel 3. Detail hasil perhitungan tingkat kepadatan lalu lintas dapat dilihat lebih jelas pada Tabel 7.

Tabel 7: Hasil Perhitungan Tingkat Kepadatan Lalu Lintas

Hari, Tanggal	Waktu	Jumlah Kendaraan Terdeteksi	Jumlah Kendaraan Hitung Manual			Q	Dk	Tingkat Kepadatan
			LV	MC	HV			
Kamis, 22 Mei 2025	06:00-07:00	7176	2116	5056	4	4,143.60	0.62	C
	07:00-08:00	6752	2529	4220	3	4,220.90	0.63	C
	11:00-12:00	6160	2487	3662	11	3,966.10	0.60	C
	12:00-13:00	4296	1642	2643	11	2,713.50	0.41	B
	16:00-17:00	5056	1850	3200	6	3,137.80	0.47	C
	17:00-18:00	6007	2050	3950	7	3,639.10	0.55	C
Minggu, 25 Mei 2025	06:00-07:00	3959	1064	2888	7	2,228.30	0.34	B
	07:00-08:00	4230	2124	2102	4	2,970.00	0.45	C
	11:00-12:00	5173	2042	3124	7	3,300.70	0.50	C
	12:00-13:00	3404	1254	2142	8	2,121.20	0.32	B
	16:00-17:00	4739	3025	1710	4	2,925.20	0.44	B
	17:00-18:00	5178	3132	2041	5	3,300.30	0.50	C

Dari Tabel 7, didapatkan bahwa keadaan lalu lintas di Jalan Ahmad Yani Kota Malang pada video pertama (hari Kamis) di sebagian besar waktu berada pada tingkat pelayanan C, yakni pada pukul 06:00–07:00, 07:00–08:00, 11:00–12:00, 16:00–17:00, dan 17:00–18:00. Tingkat pelayanan C menunjukkan bahwa arus stabil, tetapi kecepatan dan gerak dikendalikan, dan pengemudi dibatasi dalam memilih kecepatan. Hanya pada pukul 12:00–13:00 kondisi berkurang menjadi tingkat pelayanan B, yang berarti arus stabil, kecepatan operasi mulai dibatasi oleh kondisi lalu lintas, namun pengemudi dapat memilih kebebasan yang cukup untuk memilih kecepatan.

Sementara itu, pada video kedua (hari Minggu), situasi lalu lintas cenderung lebih lengang. Hal ini ditunjukkan dengan tiga dari enam rentang waktu yang masuk dalam tingkat pelayanan B, yaitu pukul 06:00–07:00, 12:00–13:00, dan 16:00–17:00. Karakteristik arus pada tingkat pelayanan B mengindikasikan bahwa arus stabil, tetapi kecepatan operasi mulai dibatasi oleh kondisi lalu lintas, dan pengemudi memilih kebebasan yang cukup untuk memilih kecepatan. Sisanya berada pada tingkat pelayanan C, yang mengindikasikan arus stabil, tetapi kecepatan dan gerak dikendalikan, dan pengemudi dibatasi dalam memilih kecepatan.

Dari keseluruhan hasil, dapat disimpulkan bahwa hari pertama (hari Kamis) cenderung memiliki lalu lintas yang lebih padat dibandingkan hari kedua (hari Minggu). Hal ini terlihat dari banyaknya tingkat pelayanan C pada seluruh segmen waktu pengamatan, kecuali satu segmen yang berada pada tingkat B. Hal ini menunjukkan bahwa pada hari kerja seperti Kamis, aktivitas masyarakat seperti berangkat kerja, sekolah, dan distribusi logistik kemungkinan besar meningkatkan volume kendaraan di ruas jalan tersebut. Sebaliknya, pada hari Minggu yang merupakan hari libur, arus kendaraan cenderung lebih stabil dan lancar, ditandai dengan lebih banyaknya segmen waktu yang berada pada tingkat pelayanan B, yang merepresentasikan arus stabil, kecepatan operasi mulai dibatasi oleh kondisi lalu lintas, namun

pengemudi memilih kebebasan yang cukup untuk memilih kecepatan. Perbedaan ini menunjukkan adanya variasi pola lalu lintas antara hari kerja dan akhir pekan yang dapat dimanfaatkan dalam pengelolaan dan pengaturan lalu lintas di waktu-waktu tertentu.

4. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan, dapat disimpulkan bahwa deteksi kepadatan lalu lintas secara otomatis di Jalan Ahmad Yani Kota Malang dapat direalisasikan dengan memanfaatkan teknologi *computer vision* berbasis YOLOv8. Model mampu mendeteksi dan menghitung jumlah kendaraan pada rekaman video CCTV secara otomatis, sehingga dapat diterapkan sebagai bagian dari sistem pemantauan lalu lintas. Sistem tersebut dapat diintegrasikan ke dalam *dashboard* pemantauan milik pemerintah daerah, untuk menyediakan data kepadatan lalu lintas secara *real-time*, yang selanjutnya dapat digunakan dalam pengambilan keputusan terkait pengelolaan lalu lintas dan penanggulangan kemacetan. Selain itu, sistem juga dapat dikembangkan menjadi layanan informasi lalu lintas publik berbasis web atau aplikasi.

Melalui proses eksplorasi *hyperparameter*, diperoleh konfigurasi terbaik untuk model deteksi kendaraan, dengan nilai *epoch* sebanyak 100, *batch size* sebesar 16, *learning rate* sebesar 0.01, dan *optimizer* menggunakan SGD. Kombinasi parameter tersebut menghasilkan performa model dengan nilai Recall sebesar 0.901 dan Precision sebesar 0.811 menggunakan *validation data*. Nilai ini mengindikasikan bahwa model memiliki kemampuan yang cukup baik dalam mengenali kendaraan. Pada tahap *model testing* menggunakan video sampel CCTV Jalan Ahmad Yani Kota Malang, model mampu mendeteksi rata-rata 78,07% kendaraan secara akurat dibandingkan dengan jumlah kendaraan aktual yang dihitung secara manual. Hal ini menunjukkan bahwa model yang telah dilatih memiliki performa yang cukup baik dalam menghadapi data baru, serta dapat digunakan dalam mendukung analisis kepadatan lalu lintas secara otomatis.

Saran yang dapat diberikan dalam upaya untuk memperbaiki penelitian ini di masa mendatang diantaranya variasi *dataset* dapat ditambahkan dengan merekam video CCTV Jalan Ahmad Yani Kota Malang pada malam hari, karena *dataset* yang dikumpulkan pada penelitian ini hanya direkam saat pagi hari, siang hari, dan sore hari. Penelitian selanjutnya dapat menggunakan model YOLO versi terbaru seperti YOLOv12, karena saat penelitian ini dilakukan YOLOv12 belum *release* dan masih menggunakan YOLOv8.

Referensi

- Abdul Kadhar, K. M., & Anand, G. (2021). Basics of Python Programming. In *Data Science with Raspberry Pi* (pp. 13–47). Apress. https://doi.org/10.1007/978-1-4842-6825-4_2
- Afrin, T., & Yodo, N. (2020). A Survey of Road Traffic Congestion Measures towards a Sustainable and Resilient Transportation System. *Sustainability*, 12(11), 4660. <https://doi.org/10.3390/su12114660>
- Aftabuzzaman, M. (2007). Measuring traffic congestion-a critical review. *30th Australasian Transport Research Forum*, 1.
- Ashad Sadiq, A. M. (2021). Analisis Dampak Besarnya Pendapatan Pengguna Jalan Yang Hilang Akibat Kemacetan (Studi Kasus Kecamatan Manggala Kota Makassar). *ILTEK: Jurnal Teknologi*, 16(02), 85–89. <https://doi.org/10.47398/iltek.v16i02.51>
- Badan Pusat Statistik Indonesia. (2024a, February 20). *Jumlah Kendaraan Bermotor Menurut Provinsi dan Jenis Kendaraan (unit)*. <https://www.bps.go.id/id/statistics-table/3/VjJ3NGRGa3dkRk5MTIU1bVNFOTVVbmQyVURSTVFUMDkjMw==/jumlah-kendaraan-bermotor-menurut-provinsi-dan-jenis-kendaraan--unit---2023.html?year=2023>
- Badan Pusat Statistik Indonesia. (2024b, February 29). *Perkembangan Jumlah Kendaraan Bermotor Menurut Jenis*. <https://www.bps.go.id/id/statistics-table/2/NTcjMg==/perkembangan-jumlah-kendaraan-bermotor-menurut-jenis--unit-.html>

- Badan Pusat Statistik Indonesia. (2024c, June 28). *Jumlah Penduduk Pertengahan Tahun, 2022-2024*. <https://www.bps.go.id/id/statistics-table/2/MTk3NSMy/jumlah-penduduk-pertengahan-tahun--ribu-jiwa-.html>
- Direktorat Jenderal Bina Marga. (2023). PKJI - Pedoman Kapasitas Jalan Indonesia. *Kementrian Pekerjaan Umum dan Perumahan Rakyat*.
- Fatoni, A., & Asmaroni, D. (2022). Analisis Tingkat Pelayanan Jalan Pada RuasJalan Kh. Amin Jakfar Ditinjau Dari Arus Pergerakan Lalu Lintas. *Ge-STRAM: Jurnal Perencanaan Dan Rekayasa Sipil*.
- Hayati, N. J., Singasatia, D., & Muttaqin, M. R. (2023). Object Tracking Menggunakan Algoritma You Only Look Once (YOLO)v8 untuk Menghitung Kendaraan. *Komputa : Jurnal Ilmiah Komputer Dan Informatika*, 12(2), 91–99. <https://doi.org/10.34010/komputa.v12i2.10654>
- Huang, S.-C., & Le, T.-H. (2021). *Principles and Labs for Deep Learning*. Elsevier. <https://doi.org/10.1016/C2020-0-03408-0>
- Hussain, M. (2024). *YOLOv5, YOLOv8 and YOLOv10: The Go-To Detectors for Real-time Vision*.
- Jane Torres. (2024, January 15). *YOLOv8 Architecture: A Deep Dive into its Architecture*. <https://Yolov8.Org/>. <https://yolov8.org/yolov8-architecture/>
- Khatami, M. S. (2022). *Deteksi Kendaraan Menggunakan Algoritma You Only Look Once (Yolo) V3*.
- Kiftiah, N. (2019). *Manajemen Lalu Lintas Berdasarkan Evaluasi Kinerja Jalan Dan Persimpangan Di Jalan Ahmad Yani – Jalan Raden Intan Kecamatan Blimbing Kota Malang*.
- Maharana, K., Mondal, S., & Nemade, B. (2022). A review: Data pre-processing and data augmentation techniques. *Global Transitions Proceedings*, 3(1), 91–99. <https://doi.org/10.1016/j.gltp.2022.04.020>
- Mukhlis, M. N. S. A., Wulanningrum, R., & Sanjaya, A. (2024). Implementasi YOLO Dalam Deteksi Jumlah Kendaraan. *Prosiding SEMNAS INOTEK (Seminar Nasional Inovasi Teknologi)*, 8(3).
- Ramadhani, R., Aini, A. F., & Hidayah, N. (2021). Urgensi Transportasi LRT Dalam Menekan Kerugian di Bidang Ekonomi Akibat Kemacetan di Perkotaan: Studi Kasus Kota Malang. *Jurnal Wilayah Dan Kota*, 7(01), 9–17. <https://doi.org/10.34010/jwk.v8i01.6852>
- Zhang, L. J., Fang, J. J., Liu, Y. X., Feng Le, H., Rao, Z. Q., & Zhao, J. X. (2024). CR-YOLOv8: Multiscale Object Detection in Traffic Sign Images. *IEEE Access*, 12, 219–228. <https://doi.org/10.1109/ACCESS.2023.3347352>