

Helmet Use Detection Model For Motorcycle Riders Using YOLOv8

Muhammad Ikram Syafwan ^{1*}, T. Yudi Hadiwandra ^{2*}

* Jurusan Teknik Elektro, Program Studi Teknik Informatika, Universitas Riau, Pekanbaru, 28293, Indonesia.
muhammad.ikram2599@student.unri.ac.id ¹, tyudihw@lecturer.unri.ac.id ²

Article Info

Article history:

Received 2026-08-03

Revised 2026-09-11

Accepted 2026-10-05

Keyword:

Object Detection,
Helmet,
Machine Learning,
YOLOv8,
CRISP-DM.

ABSTRACT

Violations of helmet use regulations by motorcyclists in the Universitas Riau campus area still frequently occur. Supported by a preliminary survey indicating that 98.6% of 149 respondents perceive campus supervision as highly lenient, this lack of direct supervision creates a gap where students tend to ignore safety rules. This study aims to develop an automated helmet detection system using the YOLOv8 deep learning algorithm to classify riders. The research process was structured following the CRISP-DM (Cross-Industry Standard Process for Data Mining) framework. During the modeling phase, the algorithm was trained utilizing a combined dataset of 4,321 augmented images. The dataset comprises primary data from campus observations and secondary data from the Roboflow public dataset, functioning as a domain adaptation strategy. Evaluation of five architectural variants showed that YOLOv8n (Nano) provided the most optimal performance, successfully achieving a mean Average Precision (mAP@0.5) of 91.43% with a highly compact weight size of 11.9 MB. This model was then implemented into a web-based interface using the Flask framework for real-time monitoring. Field testing conducted at three campus entrance gates against 179 individuals demonstrated a robust operational accuracy of 89.4%, processing video streams at an average of 5 to 5.5 Frames Per Second (FPS). Nevertheless, occasional object tracking failures (lost tracking) were encountered due to motion blur effects when vehicles traveled at high speeds, primarily caused by the limited resolution and FPS specifications of the built-in webcam used during the test.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Keselamatan berkendara, khususnya bagi pengendara sepeda motor, sangatlah penting mengingat tingginya risiko cedera kepala akibat kecelakaan lalu lintas. Penggunaan helm berstandar telah terbukti secara ilmiah mampu mencegah cedera parah. Beberapa penelitian menunjukkan bahwa pemakaian helm dapat mengurangi risiko cedera otak hingga 69% dan menekan risiko kematian secara drastis [1]. Di Indonesia, aturan wajib memakai helm tertulis jelas dalam Undang-Undang Nomor 22 Tahun 2009 Pasal 106 ayat (8) dan Pasal 291 [2]. Walaupun aturannya sudah ada, pelanggaran masih sangat sering terjadi di mana-mana, termasuk di dalam area kampus [3]. Sebagai contoh nyata, pada tahun 2023 di Kota Bogor saja tercatat ada 2.781 kasus pelanggaran pengendara yang tidak memakai helm [4].

Di Universitas Riau yang wilayah kampusnya luas dan sangat padat oleh lalu lalang mahasiswa, masalah mahasiswa

yang tidak memakai helm sangat sering dijumpai. Urgensi permasalahan ini dibuktikan melalui data empiris dari survei pendahuluan yang melibatkan 149 responden di lingkungan kampus. Hasil survei menunjukkan bahwa hampir separuh responden mengaku jarang, bahkan tidak pernah, memakai helm saat berkendara di dalam area kampus. Berdasarkan hasil pengamatan di lapangan serta data tersebut, akar masalah utama dari pelanggaran ini bukanlah kondisi jalanan atau rambu lalu lintas, melainkan murni karena kurangnya pengawasan langsung dari pihak institusi. Fakta ini diperkuat oleh 98,6% responden yang menilai bahwa sistem pengawasan dari pihak keamanan saat ini sangat longgar, di mana seluruh peserta survei mengaku tidak pernah ditegur secara langsung oleh petugas meskipun melakukan pelanggaran. Karena tidak diawasi dengan ketat, mahasiswa merasa bebas dan beranggapan bahwa berkendara jarak dekat di dalam kampus itu aman-aman saja tanpa helm. Padahal, kelalaian ini sangat berbahaya dan sudah terbukti dengan

adanya beberapa insiden kecelakaan di dalam kampus (seperti di sekitar Simpang FKIP dan Jalan SM Amin) yang membuat korban mengalami benturan kepala yang parah.

Seiring dengan kemajuan teknologi, sistem pemantauan otomatis menggunakan kamera cerdas (*Computer Vision*) bisa menjadi solusi yang sangat membantu. Teknologi ini menggunakan metode *deep learning* untuk mengenali objek secara otomatis, sehingga bisa sangat meringankan beban kerja petugas keamanan [5]. Beberapa penelitian sebelumnya sudah mencoba membuat pendeteksi helm ini seperti sistem pemantauan tilang elektronik (ETLE) menggunakan YOLOv4 [5]. Selain itu, arsitektur *deep learning* lain seperti YOLOv5 juga terbukti efektif dalam mendeteksi penggunaan helm dari rekaman CCTV persimpangan jalan [6] maupun di area kampus [7].

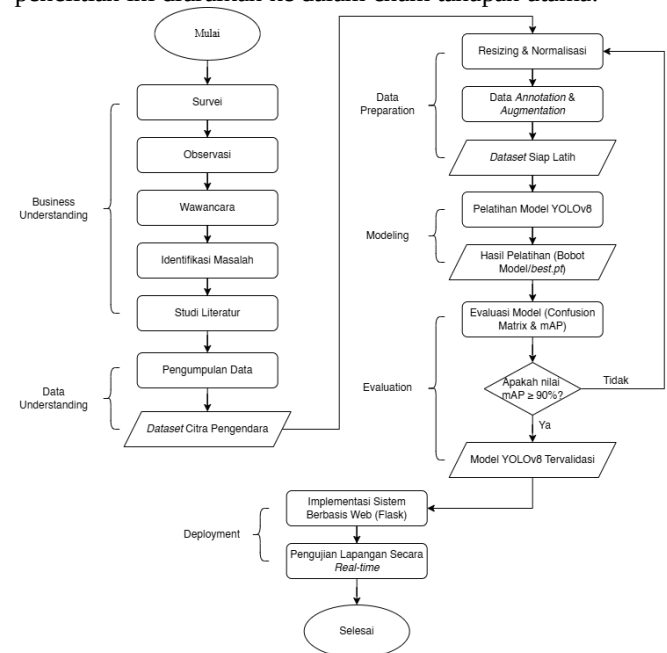
Penelitian ini memilih menggunakan algoritma *You Only Look Once* versi 8 (YOLOv8) sebagai otak utamanya. YOLOv8 dipilih karena keunggulannya yang mampu memproses seluruh citra masukan secara simultan dalam satu evaluasi jaringan (*single-shot detector*), yang menyeimbangkan antara kecepatan inferensi waktu nyata (*real-time*) dan akurasi tinggi [8]. Berdasarkan studi komparatif mutakhir, YOLOv8 memiliki stabilitas performa terbaik untuk mendeteksi objek dinamis di luar ruangan (seperti pengendara motor) dibandingkan algoritma penerusnya [9]. YOLOv8 juga terbukti memiliki latensi 27% lebih rendah pada perangkat *low-power embedded* dibandingkan YOLOv12, dan tidak mengalami penurunan akurasi yang signifikan seperti pada YOLOv10 [10]. Namun demikian, beberapa penelitian menemukan bahwa kinerja YOLOv8 sangat bergantung pada variasi data latihan; *dataset* yang monoton dari satu lokasi CCTV terbukti menurunkan akurasinya hingga ke angka 40-70% [11].

Meskipun algoritma YOLO telah lazim digunakan, sebagian besar penelitian terdahulu masih berfokus pada penerapan model di lingkungan lalu lintas perkotaan dengan sistem pengawasan skala besar. Kebaruan (*novelty*) dan kontribusi dari penelitian ini terletak pada tiga aspek utama yang membedakannya dari penelitian sebelumnya. Pertama, dari sisi kondisi lingkungan, penelitian ini secara spesifik mengkaji deteksi di area institusi pendidikan (kampus) yang memiliki karakteristik anomali visual tersendiri, seperti kepadatan kendaraan di gerbang sempit, pergerakan acak, serta mayoritas pengendara perempuan yang menggunakan berbagai model jilbab. Kedua, dari sisi *dataset*, penelitian ini mengusulkan strategi adaptasi domain (*domain adaptation*); mengombinasikan data sekunder berskala besar sebagai dasar pembelajaran fitur umum, dengan data primer lokal untuk mengenali ciri khusus lingkungan kampus agar model tidak mengalami *overfitting*. Ketiga, dari sisi implementasi, penelitian ini berfokus pada efisiensi komputasi dengan mengoptimalkan varian teringan YOLOv8n (Nano) agar dapat beroperasi sebagai sistem *monitoring* otonom berbasis *web* yang andal, tanpa menuntut spesifikasi perangkat keras kamera pengawas yang mahal.

Oleh karena itu, penelitian ini bertujuan untuk merancang dan membuat sistem deteksi helm menggunakan YOLOv8 di kampus Universitas Riau. Dengan menggunakan metode yang terstruktur dan kumpulan gambar pelatihan yang sangat banyak dan beragam, diharapkan sistem web ini bisa mendeteksi pelanggaran dengan cepat, mampu beradaptasi dengan kondisi cahaya kampus, dan benar-benar berguna untuk membantu petugas keamanan Universitas Riau mendata pelanggaran lalu lintas. Untuk membuktikan kesiapan operasional sistem pemantauan otomatis ini, evaluasi tidak hanya dilakukan pada *dataset* statis, melainkan diuji langsung di kondisi nyata. Adapun batasan ruang lingkup pada pengujian lapangan ini difokuskan pada siang hari dengan kondisi pencahayaan normal, dan tidak mencakup kondisi cuaca ekstrem seperti hujan deras atau malam hari dengan pencahayaan minim.

II. METODE

Penelitian ini dijalankan menggunakan pendekatan kerangka kerja *Cross-Industry Standard Process for Data Mining* (CRISP-DM). Model referensi ini menyediakan struktur yang teruji dan berulang (*iterative*) yang memastikan proyek *machine learning* selaras dengan kebutuhan permasalahan di dunia nyata [12]. Siklus CRISP-DM dalam penelitian ini diuraikan ke dalam enam tahapan utama.



Gambar 1. Diagram Alur Penelitian

A. Business Understanding

Tahap awal difokuskan pada analisis mendalam terhadap permasalahan kepatuhan berlalu lintas di Universitas Riau. Proses ini didukung oleh berbagai metode pengumpulan data secara paralel. Pelaksanaan survei daring yang melibatkan 149 responden mengonfirmasi bahwa sebagian besar mahasiswa menganggap pengawasan sangat longgar dan setuju bahwa teknologi deteksi otomatis amat diperlukan.

Selain itu, observasi lapangan yang dilakukan di berbagai titik gerbang membuktikan bahwa meskipun regulasi tertulis mengenai wajib helm telah dipampang dengan jelas, kepatuhan faktual tetap sangat rendah. Temuan ini kemudian diperkuat melalui wawancara komprehensif bersama Komandan Satpam Universitas Riau yang menyatakan bahwa keterbatasan personel membatasi pengawasan hanya pada tingkat teguran lisan tanpa adanya rekapitulasi data yang konkret.

B. Data Understanding

Kualitas model *deep learning* sangat bergantung pada representasi kelengkapan data. Oleh karena itu, penelitian ini menggunakan total 1.889 citra asli yang kemudian ditingkatkan kuantitasnya menjadi 4.321 citra siap latih melalui proses augmentasi. Dataset asli ini merupakan gabungan strategis dari 152 citra primer (8,05%) dan 1.737 citra sekunder (91,95%). Perbandingan sumber gambar yang digunakan bisa dilihat pada tabel di bawah ini.

TABEL I
PERBANDINGAN DAN JUMLAH DATASET

Jenis Data	Sumber	Jumlah Citra	Persentase	Fokus Kegunaan
Primer	Lingkungan Kampus UNRI	152	8,05%	Mengenali konteks lokal (jilbab, pakaian mahasiswa)
Sekunder	Roboflow Universe	1.737	91,95%	Mengenali variasi cahaya, sudut, dan warna helm global
Total		1.889	100%	

Data primer dikumpulkan melalui pengambilan gambar observasi langsung pada kondisi nyata di lingkungan Universitas Riau, berlokasi di area berintensitas lalu lintas tinggi seperti gerbang utama, area parkir, dan jalan antar-fakultas. Kondisi pengambilan gambar pada data primer secara spesifik menangkap dinamika kepadatan lalu lintas kampus, jarak objek yang bervariasi, serta karakteristik lokal yang berpotensi memicu ambiguitas klasifikasi, seperti pengendara yang berboncengan rapat dan pengendara perempuan yang menggunakan berbagai model jilbab. Sementara itu, data sekunder bersumber dari repositori dataset publik Roboflow Universe ("*Helmet Detection_YOLOv8 Computer Vision Model*"). Data sekunder ini secara komprehensif melengkapi representasi variasi sudut pengambilan kamera (*camera viewpoints*) dari berbagai tingkat elevasi, keragaman kondisi pencahayaan jalan raya global (terang, redup, silau matahari), serta berbagai jenis, bentuk, dan warna helm. Terkait distribusi kelas, *dataset* ini memuat representasi anotasi objek yang komprehensif, yang secara rinci terdiri dari 2.691 sampel untuk kelas pengguna

helm (*With Helmet*) dan 2.206 sampel untuk kelas tanpa helm (*No Helmet*). Kombinasi komprehensif dari berbagai sumber data ini memastikan bahwa model dilatih untuk mengenali variasi objek, jarak, sudut, dan pencahayaan secara representatif pada lingkungan operasional sebenarnya.

C. Data Preparation

Seluruh gambar mentah yang telah dikumpulkan pada tahap sebelumnya kemudian diubah dan dirapikan agar sesuai dengan aturan yang diminta oleh sistem YOLOv8. Proses ini dimulai dengan menyamakan ukuran semua gambar. Gambar-gambar tersebut ditarik dan disesuaikan ukurannya menjadi kotak berukuran tepat 640x640 piksel, serta dilakukan perbaikan warna secara matematis. Setelah itu, semua citra melalui tahap pelabelan objek (anotasi) menggunakan format *bounding box*. Proses anotasi dilakukan secara manual dengan mendefinisikan batas kotak secara ketat pada area luar objek untuk dua kelas utama: kepala pengendara yang menggunakan helm (*With Helmet*) dan area kepala hingga batas leher bagi pengendara yang tidak menggunakan helm (*Without Helmet*).

Untuk menjaga konsistensi pelatihan model, diterapkan pedoman anotasi khusus pada skenario visual yang ambigu. Pertama, pada kasus pengendara yang berboncengan, setiap individu (pengemudi dan penumpang) dianotasi menggunakan *bounding box* yang terpisah satu sama lain, meskipun posisi mereka saling berdekatan atau bertumpuk (*occlusion*). Kedua, untuk objek yang terhalang sebagian oleh kendaraan lain, spanduk, atau anatomi tubuh pengendara di depannya, *bounding box* tetap ditarik mencakup estimasi area kepala atau helm yang terlihat (objek parsial tetap dilabeli). Ketiga, untuk mengatasi helm dengan warna yang menyerupai latar belakang (efek kamuflase) atau pantulan cahaya matahari (*glare*), penarikan *bounding box* difokuskan secara sangat presisi (ketat) pada kontur batas tepi helm. Hal ini bertujuan agar model dipaksa untuk mempelajari fitur batas struktural dan tekstur objek, bukan sekadar menghafal representasi warnanya.

Untuk memastikan bahwa sistem ini nantinya tidak hanya hafal dengan gambar-gambar awal dan menjadi bingung ketika melihat keadaan cuaca yang berbeda, gambar-gambar tersebut diperbanyak dengan cara dimanipulasi. Manipulasi ini meliputi pemutaran gambar, membalik arah gambar ke samping, mengubah warna terang menjadi gelap, dan menambahkan efek buram seolah-olah gambar diambil saat motor sedang melaju kencang. Rincian aturan manipulasi gambar ini dapat dilihat pada Tabel 2.

TABEL II
KONFIGURASI PRA-PEMROSESAN DAN AUGMENTASI DATA

Kategori	Parameter	Nilai dan Keterangan
Persiapan	Ubah Ukuran (<i>Resize</i>)	Disesuaikan (<i>Stretch</i>) hingga 640x640 piksel
Augmentasi	Pembalikan Arah (<i>Flip</i>)	Membalik gambar secara horizontal
Augmentasi	Potong (<i>Crop</i>)	Perbesaran (<i>Zoom</i>) dari 0% hingga maksimal 20%

Augmentasi	Warna Abu-abu (Grayscale)	Diterapkan secara acak pada 25% gambar
Augmentasi	Warna dan Cahaya	Hue ($\pm 25\%$), Saturation ($\pm 25\%$), Brightness ($\pm 25\%$)
Augmentasi	Efek Kabur (Blur)	Hingga tingkat kekaburan 2.5 piksel

Setelah keseluruhan proses augmentasi selesai, *dataset* final yang berjumlah 4.321 citra dipisahkan (*split*) secara proporsional ke dalam tiga *subset* independen. Rincian distribusi pembagian *dataset* ini disajikan pada Tabel 3.

TABEL III
DISTRIBUSI PEMBAGIAN DATASET GABUNGAN

Jenis Data (Subset)	Jumlah Citra	Persentase
Train (Data Latih)	3.770	87,20%
Valid (Data Validasi)	387	9,00%
Test (Data Uji)	164	3,80%
Total Keseluruhan	4.321	100%

Untuk mencegah terjadinya *data leakage* yang dapat memicu bias performa pengujian yang semu, pemisahan *dataset* dikontrol secara ketat. Pertama, gambar-gambar primer yang memuat identitas atau objek pengendara yang sama di lokasi yang sama dipastikan hanya dialokasikan pada satu *subset* tertentu dan tidak disebar. Kedua, pemisahan data menjamin bahwa seluruh variasi citra hasil augmentasi dari satu gambar asli tidak mengalami tumpang tindih lintas *subset*, melainkan tetap terkunci di dalam *subset* yang sama dengan gambar aslinya.

D. Modeling

Terkait strategi deteksi objeknya, penelitian ini menerapkan deteksi langsung, bukan pendekatan hierarkis yang mendeteksi seluruh tubuh pengendara atau sepeda motor terlebih dahulu. Algoritma YOLOv8 secara spesifik dilatih untuk melokalisasi area di sekitar lingkaran kepala dan langsung mengklasifikasikannya ke dalam dua kelas independen: objek helm itu sendiri (kelas *With Helmet*) dan objek kepala manusia (kelas *Without Helmet*). Strategi deteksi spesifik ini mendasari desain anotasi *bounding box* yang hanya berfokus pada area kepala pada tahap sebelumnya. Pendekatan ini dipilih agar beban komputasi model lebih efisien dalam mengekstraksi fitur struktural helm, tanpa terdistraksi oleh fitur visual tubuh pengendara maupun kendaraannya.

Selanjutnya, Penelitian ini melakukan studi komparatif dengan melatih lima varian kedalaman jaringan dari algoritma YOLOv8, yang mencakup arsitektur *Nano*, *Small*, *Medium*, *Large*, dan *Extra Large*. Algoritma ini dirancang menggunakan *backbone* CSPDarknet termodifikasi dan agregasi fitur multiskala yang memisahkan klasifikasi label dari regresi koordinat secara efisien [8]. Seluruh siklus pelatihan dieksekusi di atas infrastruktur *cloud computing* Google Colab menggunakan GPU T4. Untuk memastikan bahwa eksperimen ini dapat diverifikasi dan direplikasi, konfigurasi *hyperparameter* diatur secara seragam untuk

kelima varian model. Resolusi citra masukan (*input size / imgsz*) ditetapkan pada 640x640 piksel. Pelatihan dibatasi hingga batas maksimum 100 *epochs* dengan *batch size* sebesar 16. Proses optimasi bobot jaringan didukung oleh mekanisme penghentian dini (*early stopping*) dengan parameter *patience* sebanyak 20 *epochs* guna menghentikan pelatihan secara otomatis apabila tidak ada penurunan *validation loss* yang signifikan, sehingga *overfitting* dapat dicegah. Selain itu, sistem mengoptimalkan fungsi kerugian melalui kombinasi arsitektural BCE (*Binary Cross-Entropy*) untuk klasifikasi dan CIoU (*Complete IoU*) untuk regresi *bounding box* [8].

E. Evaluation

Kinerja deteksi setiap model diukur secara ketat menggunakan instrumen *Confusion Matrix* yang merepresentasikan *True Positive*, *True Negative*, *False Positive*, dan *False Negative* [13][14]. Nilai presisi dan sensitivitas dikalkulasikan secara matematis untuk memastikan ketelitian prediksi. Metrik krusial yang dijadikan tolok ukur kesuksesan evaluasi ini adalah nilai rata-rata presisi *Mean Average Precision* (mAP) yang dikomparasikan pada rentang *Intersection over Union* tertentu [15].

F. Deployment

Tahap integrasi akhir merealisasikan rancangan model *deep learning* menjadi arsitektur perangkat lunak yang praktis dan berdaya guna. File bobot model terbaik disuntikkan ke dalam lingkungan pengembangan web menggunakan *micro-framework* Python Flask [16]. Antarmuka dasbor dirancang agar dapat mengelola aliran masukan dari kamera pengawas secara langsung, menampilkan deteksi seketika, dan mendokumentasikan log pelanggaran.

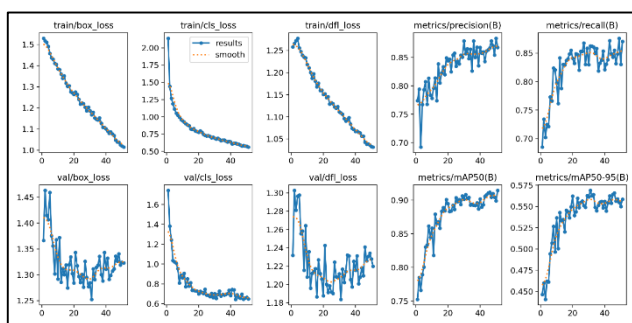
III. HASIL DAN PEMBAHASAN

Bagian ini menyajikan secara komprehensif seluruh hasil yang didapatkan dari serangkaian proses eksperimen dan penerapan langsung yang telah dilakukan. Pembahasan pada bagian ini disusun secara sistematis dan dibagi menjadi beberapa tahapan utama untuk memberikan gambaran yang jelas. Tahap pertama akan menguraikan hasil akhir dari proses *preprocessing* dan *data augmentation* yang diterapkan pada *dataset* gabungan untuk memperkaya variasi gambar. Selanjutnya, pembahasan akan difokuskan untuk melihat dinamika proses *training* dari kelima varian arsitektur YOLOv8 guna mengetahui seberapa baik setiap model dalam belajar mengenali objek. Setelah itu, performa dari masing-masing model tersebut akan dievaluasi secara mendalam menggunakan instrumen ukur *Confusion Matrix* serta nilai *Mean Average Precision* (mAP). Pada tahapan akhir, pembahasan akan menyoroti hasil *deployment* atau integrasi model yang paling optimal ke dalam sebuah antarmuka sistem berbasis web menggunakan *framework* Flask. Bagian akhir ini juga akan mengupas tuntas bagaimana keandalan sistem tersebut saat dilakukan *field testing* atau uji coba lapangan

secara *real-time* di lingkungan kampus Universitas Riau untuk menghadapi berbagai macam tantangan visual.

A. Hasil Pelatihan Model

Proses *training* dari kelima varian arsitektur YOLOv8 ini memperlihatkan grafik konvergensi (*learning curves*) yang sangat bervariasi. Mekanisme *early stopping* dengan parameter *patience* sebanyak 20 *epochs* diterapkan dalam proses *training* ini. Artinya, proses *training* akan dihentikan secara otomatis sebelum mencapai 100 *epochs* jika model tidak lagi menunjukkan penurunan nilai *validation loss* yang signifikan, guna menjaga kemampuan generalisasi model.



Gambar 2. Grafik Konvergensi Nilai Loss pada Model YOLOv8n

TABEL IV
HASIL EVALUASI KINERJA KELIMA VARIAN MODEL YOLOv8

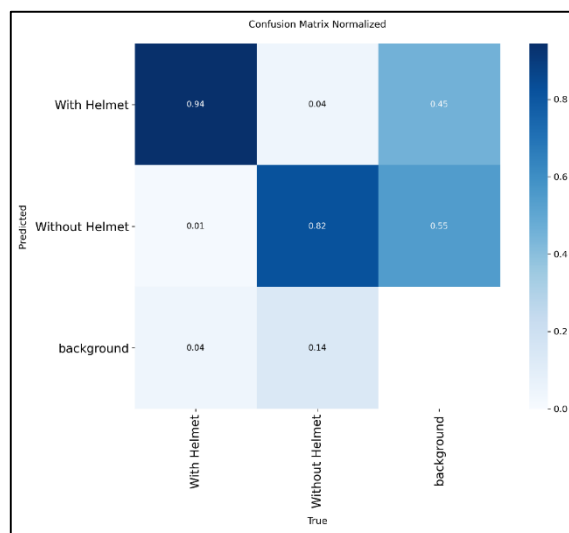
Varian Model	Epoch	Bobot Akhir (MB)	Precision (%)	Recall (%)	mAP @0.5 (%)	mAP@0.5:0.95 (%)
YOLOv8n	51	11,9	86,75	87,05	91,43	55,85
YOLOv8s	46	42,9	87,29	81,48	89,57	55,62
YOLOv8m	50	99,2	84,06	84,52	88,93	55,01
YOLOv8l	34	167,0	85,49	84,47	89,13	53,99
YOLOv8x	35	260,0	87,75	82,18	90,16	55,01

Berdasarkan metrik evaluasi yang disajikan pada Tabel 4, penetapan YOLOv8n (Nano) sebagai model paling optimal tidak hanya didasarkan pada ukuran bobotnya yang sangat ringan (11,9 MB) dan keunggulan nilai mAP@0.5 (91,43%), melainkan didukung oleh analisis performa yang komprehensif lintas metrik. Model ini mencatatkan nilai mAP@0.5:0.95 tertinggi sebesar 55,85%, yang membuktikan tingkat presisi lokalisasi *bounding box* yang sangat ketat pada berbagai ambang batas *Intersection over Union* (IoU) dibandingkan varian lainnya. Selain itu, YOLOv8n menunjukkan keseimbangan yang sangat baik antara *Precision* (86,75%) dan *Recall* (87,05%). Keseimbangan metrik ini merepresentasikan harmoni nilai *F1-Score* yang tinggi, memastikan bahwa sensitivitas model dalam menemukan objek (*Recall*) selaras dengan ketelitiannya dalam meminimalkan deteksi palsu (*Precision*).

Sebaliknya, model yang kapasitas ekstraksi fiturnya sangat besar seperti YOLOv8l (167 MB) dan YOLOv8x (260 MB) justru mengalami *early stopping* lebih cepat, yaitu pada *epoch* ke-34 dan ke-35. Hal ini mengindikasikan bahwa arsitektur yang terlalu kompleks rentan mengalami *overfitting*

saat dilatih menggunakan *dataset* yang pola geometris objeknya relatif repetitif seperti bentuk kepala dan helm. Arsitektur yang kompleks tersebut cenderung menghafal data *training* sehingga mereka gagal melakukan generalisasi saat diuji dengan subset *testing*.

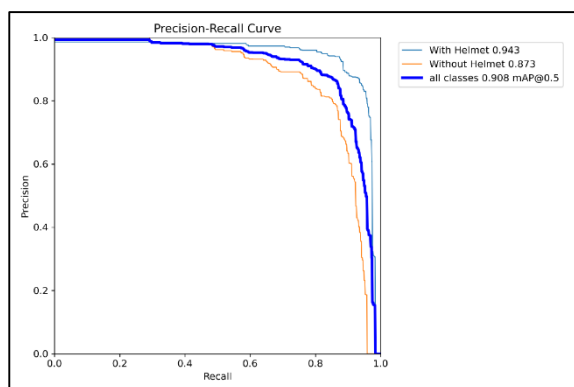
Hasil komparasi ini pada akhirnya membuktikan bahwa varian YOLOv8n memberikan *trade-off* (titik keseimbangan) yang paling optimal antara tingkat akurasi deteksi, ukuran memori model, dan kecepatan inferensi. Jika dibandingkan dengan *baseline* terdekatnya yaitu YOLOv8s, YOLOv8n tidak hanya unggul secara akurasi (mAP 91,43% berbanding 89,57%), tetapi juga memiliki ukuran bobot hampir empat kali lebih ringan (11,9 MB berbanding 42,9 MB). Ukuran parameter yang sangat ringkas ini secara langsung memangkas beban komputasi, sehingga menghasilkan kecepatan inferensi yang jauh lebih cepat. Keseimbangan metrik inilah yang menjadikan YOLOv8n sebagai arsitektur yang paling tangguh dan rasional untuk diimplementasikan pada skenario pengawasan di lingkungan kampus, di mana sistem dituntut untuk memproses aliran video secara *real-time* dengan sangat responsif, meskipun hanya menggunakan infrastruktur komputasi dan kamera yang terbatas.



Gambar 3. Evaluasi Performa Menggunakan *Confusion Matrix*

Ketangguhan model dalam membedakan kelas secara spesifik divalidasi lebih lanjut melalui instrumen *Confusion Matrix* (Gambar 3). Arsitektur YOLOv8n terbukti memiliki akurasi diferensiasi kelas yang sangat tangguh, dengan tingkat *True Positive* mencapai 94% untuk mengenali pengendara yang menggunakan helm (*With Helmet*) dan 82% untuk pengendara tanpa helm (*Without Helmet*). Tingkat kesalahan prediksi, baik berupa *False Positive* maupun *False Negative*, berhasil ditekan hingga berada di bawah 5% untuk objek kelas helm. Keunggulan performa menyeluruh pada metrik evaluasi dan klasifikasi per kelas inilah yang menjadi landasan kuat untuk mengeksplor YOLOv8n ke fase *deployment*, mengungguli varian berkapasitas besar

(YOLOv8l dan YOLOv8x) yang justru rentan mengalami *overfitting*.



Gambar 4. Grafik *Precision-Recall Curve*

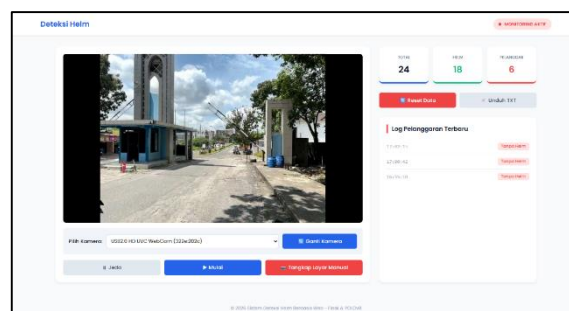
Lebih jauh, nilai $mAP@0.5$ sebesar 91,43% yang diraih oleh YOLOv8n dievaluasi menggunakan *subset* pengujian (*testing set*) yang memuat 164 citra independen. Jika dibedah berdasarkan performa masing-masing kelas visual melalui kurva *Precision-Recall* (Gambar 4), model menunjukkan dominasi akurasi pada deteksi pengguna helm dengan skor mAP spesifik mencapai 0,943 (94,3%) untuk kelas *With Helmet*. Sementara itu, performa deteksi untuk kelas target pelanggaran, yakni *Without Helmet*, mencatatkan skor mAP sebesar 0,873 (87,3%). Selisih performa ini mengindikasikan bahwa model lebih cepat mencapai konvergensi dalam mengenali pola geometris dan struktural helm yang kaku, dibandingkan mengenali variasi tak terbatas dari bentuk rambut atau anatomi kepala manusia tanpa helm. Area di bawah kurva (AUC) yang besar memastikan bahwa algoritma YOLOv8n mampu menjaga nilai *Precision* yang tinggi tanpa harus mengorbankan performa *Recall* di berbagai parameter *confidence threshold* [17].

B. Implementasi Antarmuka Sistem Berbasis Web

Sebagai sarana pendukung untuk menguji kesiapan operasional model di lapangan, implementasi sistem pemantauan direalisasikan menggunakan *framework* Flask. Penting untuk dicatat bahwa penggunaan Flask pada penelitian ini diposisikan murni sebagai jembatan integrasi (*deployment tool*), bukan sebagai kontribusi utama penelitian; kontribusi utamanya tetap bertumpu pada optimasi arsitektur pendeteksi YOLOv8n itu sendiri.

Dalam mekanisme kerjanya, model YOLOv8n diintegrasikan dengan sistem *video surveillance* melalui penangkapan aliran video (*video stream*) dari kamera pengawas secara langsung. Setiap *frame* dari video tersebut diproses seketika (*real-time*) oleh model untuk melakukan deteksi area lingkaran kepala dan mengklasifikasikannya ke dalam kelas *With Helmet* atau *Without Helmet* beserta skor keyakinannya. Proses pelacakan (*tracking*) objek bergantung pada konsistensi penempatan *bounding box* antar-*frame* secara kontinu, meskipun mekanisme pelacakan ini masih rentan mengalami kehilangan jejak (*lost tracking*) apabila

terjadi *motion blur* akibat keterbatasan FPS kamera pada pergerakan kendaraan berkecepatan tinggi.



Gambar 5. Dasbor Web Deteksi Helm YOLOv8 Berbasis Flask

Terkait mekanisme penyimpanan dan identifikasi pelanggaran, setiap kali algoritma mendeteksi objek kelas *Without Helmet*, sistem *backend* akan merespons dengan secara otonom memotong gambar (*cropping*) pada koordinat target dan menyimpannya ke dalam direktori lokal server sebagai bukti otentik. Identifikasi pelanggaran saat ini disajikan melalui pencatatan riwayat (*log*) waktu kejadian secara kronologis pada tabel dasbor. Adapun fitur tingkat lanjut seperti pemberian notifikasi aktif (misalnya alarm peringatan) atau identifikasi identitas spesifik pelanggar (melalui integrasi ke basis data nomor induk mahasiswa) belum diimplementasikan pada versi sistem saat ini, dan direkomendasikan sebagai fokus pengembangan lanjutan di masa mendatang.

Sistem ini secara ketat hanya berfokus pada klasifikasi penggunaan helm pada objek kepala, dan tidak dirancang untuk mengidentifikasi profil personal pengendara (seperti pengenalan wajah atau plat nomor). Untuk memastikan akurasi data statistik pada dasbor dan mencegah terjadinya duplikasi penghitungan, di mana satu pengendara yang melintas terekam dalam puluhan *frame* berturut-turut, sistem dilengkapi dengan mekanisme *object tracking* (pelacakan objek).

Mekanisme ini bekerja dengan memberikan identitas pelacakan unik (*unique tracking ID*) secara sekuensial pada setiap objek yang masuk ke dalam jangkauan kamera, yang secara visual direpresentasikan pada label *bounding box* (sebagai contoh, pelabelan 'id:326'). Dengan adanya penetapan ID unik ini, saat sebuah objek diklasifikasikan sebagai *Without Helmet*, sistem *backend* hanya akan memicu proses pencatatan *log* pelanggaran dan ekstraksi gambar (*cropping*) sebanyak satu kali untuk ID tersebut. Algoritma pelacakan ini menjamin bahwa keberadaan objek pengendara yang sama di berbagai *frame* yang saling berurutan tidak akan dihitung sebagai beberapa insiden pelanggaran yang berbeda, sehingga akumulasi data pelanggaran harian yang disajikan pada sistem tetap valid dan akurat.

Untuk membuktikan kelayakan sistem dalam skenario pemantauan waktu nyata (*real-time monitoring*), evaluasi tidak hanya bertumpu pada efisiensi ukuran model (11,9 MB), melainkan divalidasi secara kuantitatif melalui pengukuran latensi inferensi dan utilisasi perangkat keras. Spesifikasi

lingkungan komputasi yang digunakan selama pengujian operasional di lapangan dirinci pada Tabel 5.

TABEL V
SPESIFIKASI PERANGKAT KERAS PENGUJIAN LAPANGAN

Komponen	Spesifikasi
Model Perangkat	ASUS TUF Gaming FX506HM
Prosesor (CPU)	Intel® Core™ i7-11800H (8 Cores)
Memori (RAM)	16 GB
Kartu Grafis (GPU)	NVIDIA GeForce RTX 3060 Laptop GPU (6 GB VRAM)
Sensor Visual	Built-in HD Webcam

Pada lingkungan komputasi tersebut, model YOLOv8n mencatatkan waktu inferensi aktual yang sangat efisien, yakni rata-rata di kisaran 65 hingga 80 milidetik per *frame*. Berbekal latensi inferensi yang rendah ini, sistem *backend* secara keseluruhan mampu memproses aliran video pada kecepatan rata-rata 5 hingga 5,5 *Frames Per Second* (FPS). Selama beroperasi, tingkat utilisasi memori RAM berjalan stabil di kisaran 87,7% hingga 89,7%, sedangkan beban kerja CPU berfluktuasi secara dinamis dari kondisi siaga hingga mencapai beban puncak (100%) pada *frame* tertentu. Menariknya, arsitektur YOLOv8n terbukti sangat ringan bagi pemrosesan grafis; tingkat utilisasi GPU konsisten di bawah 15% dengan konsumsi VRAM statis di angka 678 MB.

Metrik operasional ini membuktikan bahwa efisiensi model mampu mendukung operasional *real-time* tanpa memicu *bottleneck* pada memori grafis. Adapun keterbatasan kecepatan akhir (5 FPS) dan munculnya efek *motion blur* pada pengujian murni bersumber dari limitasi *shutter speed* kamera web bawaan serta tingginya beban pemrosesan *streaming* video pada CPU oleh *framework* antarmuka web. Kendala perangkat keras tersebut memberikan ruang lingkup perbaikan untuk implementasi masa depan menggunakan infrastruktur kamera pengawas (CCTV) beresolusi tinggi.

C. Evaluasi Uji Coba Lapangan (Field Testing)

Guna membuktikan kesiapan sistem *monitoring* otomatis secara komprehensif dan tidak sekadar bergantung pada hasil evaluasi *dataset* statis, pengujian *deployment* di dunia nyata dilaksanakan di tiga pintu gerbang utama Universitas Riau, yaitu Gerbang Binawidya, Gerbang Muchtar Lutfi, dan Gerbang Bangau Sakti. Pengujian ini mengevaluasi ketahanan (*robustness*) sistem terhadap dinamika kendaraan yang bergerak bebas, berbagai sudut pengambilan kamera pengawas, dan tingkat kepadatan lalu lintas yang berbeda.

1) **Evaluasi Terhadap Karakteristik Fisik Pengendara:** Ketika sistem diuji untuk memproses objek dengan berbagai macam ciri fisik anatomi, mulai dari pria dengan postur yang berbeda hingga perempuan yang memakai berbagai gaya jilbab, algoritma terbukti sangat stabil dan tidak mengalami bias klasifikasi. Model berhasil membedakan batas tekstur jilbab tanpa menyalahartikannya sebagai geometri helm, sehingga angka *False Positive* tetap terkendali.



Gambar 6. Hasil Deteksi Pengendara Laki-Laki



Gambar 7. Hasil Deteksi Pengendara Perempuan Dengan Jilbab

2) **Evaluasi Terhadap Variasi Atribut Helm:** Berkat keragaman *feature base* dari *dataset* Roboflow, model sukses mengenali karakteristik struktural berbagai jenis helm. *Field testing* membuktikan bahwa model mampu mengklasifikasikan helm jenis *half-face*, *full-face*, maupun helm bogo dengan sama baiknya. Sistem ini juga memiliki ketahanan yang tinggi terhadap efek silau (*glare*) dari *visor*, kamufase warna helm dengan *background*, hingga helm yang memiliki corak grafis kompleks.



Gambar 8. Hasil Deteksi Terhadap Variasi Bentuk dan Warna Helm

3) **Evaluasi Terhadap Dinamika Kecepatan (Speed):** Skenario pengujian kecepatan pergerakan kendaraan mengungkap batas fisik operasional dari infrastruktur *hardware* kamera yang digunakan. Pada saat melakukan *inference* terhadap motor yang melaju dengan kecepatan rendah, model mampu menempatkan *bounding box* dengan presisi tinggi, namun *confidence score* rata-rata tertahan di angka 70% hingga 80% karena faktor resolusi spasial kamera yang rendah.



Gambar 9. Hasil Deteksi pada Kecepatan Rendah

Hambatan komputasi yang sangat nyata baru terekspos ketika motor melaju dengan kecepatan sedang hingga tinggi. Spesifikasi *Frame Per Second* (FPS) yang sangat terbatas pada lensa kamera memicu distorsi visual berupa *motion blur* yang parah pada frame video. Degradasi kualitas piksel akibat *motion blur* ini menyebabkan model sesekali mengalami *lost tracking*. *Bounding box* sempat terlepas dari objek target selama beberapa detik sebelum algoritma berhasil memulihkan pelacakan.



Gambar 10. Hasil Deteksi pada Kecepatan Sedang-Tinggi

4) *Evaluasi Terhadap Karakteristik Lokasi Pengujian:* Tantangan iluminasi luar ruangan di setiap gerbang memberikan metrik penilaian tersendiri. Di Gerbang Binawidya yang lajur kendaraannya renggang, *inference* berjalan sempurna tanpa kendala oklusi (objek yang tumpang tindih). Di Gerbang Muchtar Lutfi, bayangan tajam dari rimbunan pohon menciptakan variasi kontras cahaya, namun karena sistem sudah dilatih dengan teknik augmentasi *Hue* dan *Brightness*, model tetap mampu melakukan klasifikasi dengan akurat. Sementara itu, pengujian terhadap kepadatan lalu lintas yang tinggi diuji di Gerbang Bangau Sakti. Pada kondisi kendaraan yang padat dan bergerak bersamaan, model sukses memecahkan masalah *occlusion* (objek yang saling menutupi). Algoritma terbukti tangguh membedakan *bounding box* antara pengemudi di depan dan penumpang di belakangnya yang posisinya bertumpuk dari sudut pandang kamera. Secara keseluruhan, implementasi aplikasi berbasis web ini telah siap secara fungsional untuk menutupi celah pengawasan manual di Universitas Riau.

Untuk memvalidasi efektivitas sistem secara kuantitatif, pengujian lapangan (*field testing*) mencatat performa deteksi terhadap total 179 individu yang melintas di ketiga gerbang utama kampus. Penghitungan ini didasarkan pada objek kepala yang terdeteksi, sehingga pengendara yang berboncengan dihitung sebagai dua entitas yang dievaluasi secara terpisah.

TABEL VI
EVALUASI KUANTITATIF PENGUJIAN LAPANGAN (*FIELD TESTING*)

Lokasi Pengujian	Total Individu	TP	TN	FP	FN	Akurasi (%)
Gerbang Binawidya	36	19	11	5	1	83,3%
Gerbang Muchtar Lutfi	23	20	2	1	0	95,6%
Gerbang Bangau Sakti	120	25	83	12	0	90,0%
Total Keseluruhan	179	64	96	18	1	89,4%

Berdasarkan Tabel 6, sistem menunjukkan performa paling stabil di Gerbang Muchtar Lutfi dengan akurasi mencapai 95,6%, tingkat kesalahan yang sangat minim pada kondisi ini menunjukkan ketahanan model terhadap variasi pencahayaan dari rimbunan pohon. Sebaliknya, Gerbang Bangau Sakti mencatatkan volume lalu lintas tertinggi dengan 120 individu yang diuji, yang mayoritas didominasi oleh kelas pelanggaran (83 *True Negative*). Kondisi padat ini menyumbang angka *False Positive* terbanyak (12 kasus), sejalan dengan temuan observasi di mana masalah *occlusion* (objek bertumpuk) sering mengecoh model. Di Gerbang Binawidya, meskipun akurasi berada pada 83,3%, tercatat 1 kasus *False Negative* di mana sistem gagal mengenali helm akibat hilangnya pelacakan (*lost tracking*) dari efek *motion blur*. Meskipun dihadapkan pada berbagai kendala lapangan, secara keseluruhan sistem berhasil mengklasifikasikan 160 kejadian (64 TP dan 96 TN) dengan benar dari 179 sampel, menghasilkan akurasi operasional rata-rata sebesar 89,4%. Hal ini membuktikan bahwa keandalan klasifikasi YOLOv8n pada kondisi nyata terbukti sejalan dan tidak berbeda jauh dari hasil evaluasi mAP (91,43%) pada *dataset* latihan.

IV. KESIMPULAN

Berdasarkan analisis komprehensif dari proses eksperimen dan pengujian operasional lapangan, arsitektur *Deep Learning* YOLOv8n (Nano) terbukti menjadi model yang paling optimal untuk tugas deteksi objek penggunaan helm di wilayah kampus. Model ini menunjukkan efisiensi komputasi yang tinggi dengan ukuran *weights* yang hanya sebesar 11,9 MB, sembari berhasil mencapai nilai mAP@0.5 yang sangat memuaskan, yakni 91,43%. Selain itu, algoritma visi komputer ini memiliki kemampuan generalisasi fitur yang sangat baik. Hal ini dibuktikan dengan ketangguhan model yang tidak mengalami bias saat membedakan properti fisik seperti jilbab dengan helm, serta konsistensinya dalam mengklasifikasikan berbagai macam bentuk, warna, dan corak helm. Ketangguhan teoritis ini berhasil dibuktikan secara nyata pada pengujian lapangan (*field testing*), di mana sistem mencatatkan tingkat akurasi operasional sebesar 89,4% dalam mendeteksi 179 individu yang diuji di tiga gerbang utama kampus.

Dari sisi implementasi, fase *deployment* dari model menjadi sistem perangkat lunak fungsional berbasis antarmuka web Flask telah berhasil direalisasikan untuk memfasilitasi proses pemantauan secara *real-time*. Namun demikian, kesimpulan mengenai efektivitas sistem ini harus

dibatasi secara objektif pada ruang lingkup eksperimen. Pengujian operasional pada penelitian ini hanya divalidasi pada skala terbatas di tiga titik akses Universitas Riau. Meskipun sistem mampu beroperasi secara fungsional, masih ditemukan kendala berupa kegagalan pelacakan objek (*lost tracking*) saat mendeteksi kendaraan yang melaju dengan kecepatan tinggi. Kegagalan ini dipicu oleh efek *motion blur* pada rekaman video yang bersumber dari keterbatasan *shutter speed* kamera web bawaan serta limitasi pemrosesan sistem di angka 5 FPS.

Oleh karena batasan uji coba dan masih adanya kendala perangkat keras tersebut, performa sistem ini belum dapat digeneralisasi sebagai solusi pengawasan lalu lintas yang sepenuhnya efektif untuk seluruh kondisi lingkungan eksternal tanpa adanya validasi yang lebih luas. Untuk pengembangan infrastruktur keamanan kampus di masa depan, sangat direkomendasikan untuk melakukan pembaruan menggunakan kamera CCTV dengan spesifikasi resolusi dan FPS yang lebih memadai, serta mengintegrasikan perangkat keras tambahan seperti alarm peringatan suara yang terhubung langsung dengan *output* prediksi model.

DAFTAR PUSTAKA

- [1] J. R. Lepard *et al.*, "Differences in Outcomes of Mandatory Motorcycle Helmet Legislation by Country Income Level: A Systematic Review and Meta-analysis," *PLoS Med.*, vol. 18, no. 9, Sep. 2021, doi: 10.1371/journal.pmed.1003795.
- [2] B. Rachmat *et al.*, "Prevalence and Factors Associated with the Use of Standard Helmets among Motorcyclists among workers in Indonesia," in *E3S Web of Conferences*, EDP Sciences, Nov. 2023, doi: 10.1051/e3sconf/202344802023.
- [3] A. Purnama, J. Indra, S. A. P. Lestari, and S. Faisal, "Deteksi Pelanggaran Penggunaan Helm Dengan Metode SSD Dan Arsitektur MobileNetV2," *Journal of Information System Management (JOISM)*, vol. 7, no. 1, pp. 7–14, 2025.
- [4] A. R. Haerudin, S. A. Hudjimartu, and E. Hermawan, "Deteksi Pelanggaran Lalu Lintas Kendaraan Bermotor Roda Dua Menggunakan Metode Convolutional Neural Network Wilayah Kota Bogor," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 3, pp. 4304–4309, Jun. 2025.
- [5] A. W. Suryanto and A. R. Kardian, "Deteksi Pelanggaran Lalu Lintas Tidak Menggunakan Helm Dengan YOLO V4 Pada Sistem ETLE," *Jurnal Teknik Komputer AMIK BSI*, vol. 9, no. 2, pp. 129–134, Jul. 2023, doi: 10.31294/jtk.v4i2.
- [6] M. P. Anugrah, B. Fatkhurrozi, and H. T. Setiawan, "Deteksi Helm Pengendara dan Plat Nomor Kendaraan pada CCTV Lampu Lalu Lintas Menggunakan Algoritma YOLO," *Jurnal Vocational Teknik Elektronika dan Informatika*, vol. 12, no. 1, p. 1, Mar. 2024, [Online]. Available: <http://ejournal.unp.ac.id/index.php/voteknika/index>
- [7] Z. H. Batubara, Y. H. Nainggolan, M. Arfan, and A. Hidayatno, "Perancangan Sistem Deteksi Pelanggaran Penggunaan Helm Dengan Metode Deep Learning Menggunakan YOLOv5 Ultralytic," *TRANSIENT*, vol. 13, no. 1, pp. 11–20, Mar. 2024, [Online]. Available: <http://ejournal3.undip.ac.id/index.php/transient>
- [8] M. Sohan, T. SaiRam, and Ch. V. RamiReddy, "A Review on YOLOv8 and Its Advancements," in *International Conference on Data Intelligence and Cognitive Informatics*, Jan. 2024, pp. 529–545. doi: 10.1007/978-981-99-7962-2_39.
- [9] A. Sharma, V. Kumar, and L. Longchamps, "Comparative performance of YOLOv8, YOLOv9, YOLOv10, YOLOv11 and Faster R-CNN models for detection of multiple weed species," *Smart Agricultural Technology*, vol. 9, Dec. 2024, doi: 10.1016/j.atech.2024.100648.
- [10] K. Kim, J. Lee, I. Jeong, J. Jung, and J. Cho, "Real-Time Parking Space Management System Based on a Low-Power Embedded Platform," *Sensors*, vol. 25, no. 22, Nov. 2025, doi: 10.3390/s25227009.
- [11] M. A. Rahman, H. Setiawan, and M. A. Rosid, "Deteksi Penggunaan Helm Dan Plat Nomor Menggunakan YOLOv8 Di Area Parkir," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 9, no. 5, pp. 7652–7658, Oct. 2025.
- [12] R. Wirth and J. Hipp, "CRISP-DM: Towards a Standard Process Model for Data Mining," in *4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, Apr. 2000, pp. 29–40.
- [13] E. Y. Wijaya, M. H. D. Prasetyo, F. A. Kartika, and Q. A. T. Effendy, "Pengembangan Sistem Informasi Rekomendasi Wisata Kalimantan Timur Berbasis Web Menggunakan Metode Item-Based Collaborative Filtering," *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 6, no. 4, pp. 562–572, Oct. 2023, doi: 10.32493/jtsi.v6i3.32962.
- [14] T. Y. Hadiwandra, "Perbandingan Kinerja Model Klasifikasi Decision Tree, Bayesian Classifier, Instance Base, Linear Function Base, Rule Base pada 4 Dataset Berbeda," *SATIN - Sains dan Teknologi Informasi*, vol. 5, no. 1, pp. 70–78, Jun. 2019, [Online]. Available: <http://jurnal.stmik-amik-riau.ac.id>
- [15] N. Jegham, C. Y. Koh, M. Abdelatti, and A. Hendawi, "YOLO Evolution: A Comprehensive Benchmark and Architectural Review of YOLOv12, YOLO11, and Their Previous Versions," Mar. 2025, [Online]. Available: <http://arxiv.org/abs/2411.00201>
- [16] M. Grinberg, *Flask Web Development*, 1st ed. O'Reilly Media, 2014.
- [17] M. Trigka and E. Dritsas, "A Comprehensive Survey of Machine Learning Techniques and Models for Object Detection," *Sensors*, vol. 25, no. 1, pp. 1–32, Jan. 2025, doi: 10.3390/s25010214.