

DETEKSI PENGENDARA MOTOR TANPA HELM MENGGUNAKAN YOLOv8 DENGAN NOTIFIKASI REAL-TIME MELALUI TELEGRAM

Arya Jili Pranata^{1*}, Emir Nasrullah², Anisa Ulya Darajat³
^{1,2,3}Jurusan Teknik Elektro, Fakultas Teknik, Universitas Lampung
 E-mail: aryajilipranata01@gmail.com^{1*}

ABSTRACT

Helmet violations are a major contributor to motorcycle-related traffic accidents in Indonesia. This study developed an automated helmet violation detection system using the YOLOv8 algorithm integrated with Telegram for real-time notifications. The model was trained on 6,152 images across four object classes (helmet, no helmet, motorcycle, and license plate) over 300 epochs using 640×640 pixel input and an RTX 3060 GPU. Training results achieved a precision of 85.7%, recall of 82.7%, and mAP@50 of 84.9%. The system was implemented in a real-world setting with a camera placed at 2.5 meters high and 45° angle at the entrance of Universitas Lampung. Real-time testing showed the system could detect violations and send image evidence via Telegram instantly. Field precision reached 100%, recall ranged from 36% to 85%, affected by camera angle and object visibility. These findings demonstrate that the integration of YOLOv8 and Telegram Bot API offers an effective and practical solution for computer vision-based traffic monitoring, with strong potential for real-world enforcement of road safety regulations.

Keywords: *Helmet Detection, YOLOv8, Telegram, Real-Time Notification, Computer Vision, Traffic Safety*

ABSTRAK

Pelanggaran penggunaan helm menjadi salah satu penyebab utama kecelakaan lalu lintas yang melibatkan pengendara sepeda motor di Indonesia. Penelitian ini mengembangkan sistem deteksi otomatis pelanggaran helm berbasis algoritma YOLOv8 yang terintegrasi dengan Telegram sebagai media notifikasi real-time. Model dilatih menggunakan 6.152 citra dengan empat kelas objek (helm, tanpa helm, motor, dan plat nomor) selama 300 epoch dengan input 640×640 piksel dan GPU RTX 3060. Hasil pelatihan menunjukkan precision 85,7%, recall 82,7%, dan mAP@50 sebesar 84,9%. Sistem diimplementasikan pada lingkungan nyata dengan kamera yang dipasang setinggi 2,5 meter dan sudut 45° di gerbang Universitas Lampung. Pengujian real-time menunjukkan sistem mampu mendeteksi pelanggaran dan mengirim bukti gambar ke Telegram secara instan. *Precision* lapangan mencapai 100%, nilai recall berkisar antara 36% hingga 85%, dipengaruhi oleh sudut pandang dan kualitas visual objek. Hasil ini menunjukkan bahwa integrasi YOLOv8 dan Telegram Bot API memberikan solusi yang efektif dan praktis untuk pemantauan lalu lintas berbasis visi

Article History

Received: Juli 2025
 Reviewed: Juli 2025
 Published: Juli 2025

Plagiarism Checker No
 235

Prefix DOI :
[10.8734/Kohesi.v1i2.365](https://doi.org/10.8734/Kohesi.v1i2.365)

Copyright : Author
 Publish by : Kohesi



This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)



komputer, dengan potensi penerapan nyata dalam mendukung penegakan aturan keselamatan berkendara.

Kata Kunci: *Deteksi Helm, YOLOv8, Telegram, Notifikasi Real-Time, Visi Komputer, Keselamatan Lalu Lintas*

1. PENDAHULUAN

Keselamatan lalu lintas merupakan isu krusial yang membutuhkan perhatian lintas sektor, terutama di negara berkembang seperti Indonesia yang tingkat kepemilikan kendaraan roda dua sangat tinggi. Berdasarkan data Korlantas Polri tahun 2022, tercatat sebanyak 152.275 kasus kecelakaan lalu lintas, dengan korban meninggal dunia mencapai 25.938 jiwa. Sebagian besar korban kecelakaan adalah pengguna kendaraan roda dua, yang mencapai lebih dari 70% dari total kasus kecelakaan (Korlantas Polri, 2023; Badan Pusat Statistik, 2023). Kondisi ini mengindikasikan tingkat kerentanan yang tinggi pada pengendara motor, khususnya akibat pelanggaran keselamatan berkendara seperti tidak menggunakan helm.

Regulasi mengenai kewajiban penggunaan helm sebenarnya telah diatur dalam Undang-Undang Republik Indonesia Nomor 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan, khususnya Pasal 106 ayat (8), yang menyebutkan bahwa setiap pengendara dan penumpang sepeda motor wajib menggunakan helm berstandar nasional. Ketentuan tersebut dipertegas kembali melalui Peraturan Pemerintah No. 55 Tahun 2012, yang mengatur standar keamanan dan kewajiban penggunaan helm selama berkendara (Undang-Undang RI, 2009; Peraturan Pemerintah RI, 2012). Namun, tingkat kepatuhan masyarakat masih rendah. Studi yang dilakukan oleh Kementerian Perhubungan mencatat bahwa sekitar 30% pengendara sepeda motor tidak menggunakan helm secara benar, bahkan ada yang tidak menggunakannya sama sekali (Kementerian Perhubungan RI, 2025).

Perkembangan teknologi digital dan kecerdasan buatan (*Artificial Intelligence/AI*) telah membuka peluang besar dalam otomatisasi sistem pengawasan lalu lintas. Salah satu cabang dari AI yang relevan adalah pengolahan citra digital (*Digital Image Processing*), yang memungkinkan ekstraksi informasi dari gambar dan video secara otomatis. Dalam konteks keselamatan lalu lintas, teknologi ini dapat dimanfaatkan untuk mendeteksi pelanggaran, termasuk penggunaan helm (Gonzalez et al., 2019; Jain, 1989).

Salah satu pendekatan deteksi visual yang paling menonjol dalam beberapa tahun terakhir adalah algoritma YOLO (*You Only Look Once*). YOLO merupakan metode deteksi objek berbasis deep learning yang mampu melakukan deteksi dan klasifikasi objek secara cepat dan akurat dalam satu tahap pemrosesan. YOLOv8, sebagai versi terbaru yang dikembangkan oleh



Ultralytics, mengadopsi arsitektur anchor-free dan decoupled head yang lebih efisien dibandingkan versi sebelumnya, sehingga sangat cocok untuk aplikasi real-time seperti pengawasan lalu lintas (Redmon et al., 2016; Jocher et al., 2025). YOLOv8 dapat mendeteksi berbagai objek sekaligus dalam satu frame, seperti helm, pengendara, sepeda motor, dan plat nomor dengan kecepatan inferensi tinggi.

Sistem deteksi saja tidak cukup. Pelanggaran yang berhasil dideteksi juga perlu segera diinformasikan kepada pihak terkait agar dapat ditindaklanjuti. Dalam konteks ini, sistem notifikasi real-time menjadi komponen penting. Salah satu platform yang dapat diintegrasikan secara mudah adalah Telegram. Telegram menyediakan Telegram Bot API yang memungkinkan pengembang untuk mengirimkan pesan berupa gambar, teks, dan metadata lainnya secara otomatis melalui jaringan internet (Telegram Messenger LLP, 2025; Singh et al., 2022). Integrasi antara YOLOv8 dan Telegram Bot membuka peluang implementasi sistem pengawasan lalu lintas yang otonom, efisien, dan responsif.

Aspek perangkat keras juga berperan besar dalam mendukung akurasi sistem deteksi. Penempatan kamera merupakan komponen penting, karena berpengaruh langsung terhadap visibilitas objek yang diamati. Parameter seperti Field of View (FOV), ketinggian, sudut kemiringan kamera, serta jarak terhadap jalur lalu lintas sangat menentukan keberhasilan sistem dalam menangkap objek penting seperti kepala pengendara, helm, dan plat nomor (Hasan et al., 2018). Oleh karena itu, perancangan sistem deteksi pelanggaran helm juga harus memperhatikan aspek teknis dari konfigurasi kamera.

Beberapa penelitian sebelumnya telah mengembangkan sistem deteksi pelanggaran helm menggunakan YOLO versi terdahulu seperti YOLOv3 atau YOLOv5, dengan hasil yang cukup menjanjikan (Setyawan et al., 2021; Batubara et al., 2024). Namun, sebagian besar belum mengintegrasikan sistem notifikasi real-time secara langsung ke platform komunikasi modern, serta belum memanfaatkan arsitektur YOLOv8 yang lebih mutakhir. Selain itu, banyak sistem yang masih terbatas pada uji coba simulasi dan belum diimplementasikan pada lingkungan nyata yang melibatkan dinamika pencahayaan, sudut kamera, dan variasi objek di jalan raya.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan untuk mengembangkan dan menguji sistem deteksi otomatis pengendara motor tanpa helm menggunakan model YOLOv8 yang terintegrasi dengan Telegram Bot sebagai media notifikasi real-time. Sistem ini dirancang untuk bekerja secara otomatis mulai dari akuisisi video melalui webcam, pendeteksian objek, validasi pelanggaran, hingga pengiriman gambar hasil deteksi ke akun Telegram pengguna atau petugas. Dengan sistem ini, diharapkan dapat tercipta mekanisme pengawasan lalu lintas yang



lebih efisien dan efektif, yang tidak hanya bersifat reaktif tetapi juga proaktif dalam mendorong kepatuhan terhadap aturan keselamatan berkendara.

2. METODE PENELITIAN

Penelitian ini dilakukan dengan pendekatan rekayasa sistem yang bertujuan untuk merancang dan mengimplementasikan sistem deteksi pelanggaran helm secara otomatis berbasis algoritma YOLOv8 yang terintegrasi dengan notifikasi *real-time* melalui Telegram. Kegiatan penelitian dilaksanakan di Laboratorium Teknik Kendali, Jurusan Teknik Elektro, Universitas Lampung, pada periode November 2024 hingga Mei 2025. Proses penelitian mencakup tahapan perancangan sistem, pelatihan model deteksi, integrasi sistem notifikasi, hingga pengujian sistem di lingkungan nyata.

Alat utama yang digunakan dalam penelitian ini adalah satu unit laptop *Acer Spin 3 Active* dengan spesifikasi prosesor Intel Core i5-1135G7, RAM 8 GB, SSD 512 GB, dan sistem operasi Windows 11. Perangkat ini digunakan untuk seluruh proses pengembangan sistem, mulai dari pelabelan data, pelatihan model, hingga implementasi sistem. Kamera *webcam* berfungsi sebagai alat akuisisi video secara *real-time*, yang kemudian diolah oleh model deteksi. Selain itu, koneksi internet diperlukan untuk mendukung pengiriman notifikasi otomatis ke Telegram melalui pemanggilan API.

Perangkat lunak utama dalam penelitian ini mencakup bahasa pemrograman Python dengan lingkungan pengembangan Google Colab dan Visual Studio Code. *Framework* yang digunakan adalah PyTorch dengan implementasi Ultralytics YOLOv8. Beberapa pustaka pendukung lain yang digunakan antara lain OpenCV untuk pemrosesan citra, NumPy dan Pandas untuk manipulasi data, serta Matplotlib untuk visualisasi hasil pelatihan. Untuk anotasi data, digunakan platform Roboflow dengan format ekspor YOLOv8. Selain itu, pengiriman notifikasi dilakukan dengan memanfaatkan Telegram Bot API yang dikonfigurasi menggunakan token dari layanan BotFather.

Dataset yang digunakan terdiri dari 6.152 gambar yang mewakili empat kelas objek yaitu helm, tanpa helm, sepeda motor, dan plat nomor. Dataset ini dibagi menjadi tiga bagian: 70% untuk pelatihan (training), 20% untuk validasi (validation), dan 10% untuk pengujian (testing). Seluruh citra telah dianotasi dalam format YOLO dengan parameter `<class_id>`, `<x_center>`, `<y_center>`, `<width>`, dan `<height>`, yang dinormalisasi terhadap ukuran gambar. Dataset kemudian dikonversi menjadi format `.yaml` yang berisi path dataset dan nama-nama kelas, untuk digunakan dalam proses pelatihan model YOLOv8.



Proses pelatihan dilakukan selama 300 epoch menggunakan YOLOv8s (small) dengan ukuran input gambar 640x640 piksel dan batch size 16. Model dilatih menggunakan GPU NVIDIA GeForce RTX 3060, memanfaatkan precision setengah (FP16) untuk efisiensi memori dan kecepatan komputasi. Hasil pelatihan model disimpan dalam bentuk file .pt dan dievaluasi menggunakan metrik evaluasi seperti precision, recall, mAP@50, dan mAP@50:95. Selama pelatihan, metrik training loss dan validation loss dipantau untuk menghindari overfitting dan memastikan generalisasi model terhadap data baru.

Setelah pelatihan selesai, sistem deteksi diintegrasikan dengan Telegram menggunakan pustaka requests untuk mengirim gambar pelanggaran secara otomatis. Telegram Bot diatur untuk menerima input berupa gambar hasil deteksi, serta informasi waktu, lokasi, dan ID pelanggar. Ketika sistem mendeteksi objek “motor” dan “tanpa helm” dalam satu frame, maka tangkapan gambar akan dikirim melalui Telegram secara real-time. Untuk menjaga kestabilan pengiriman pesan, sistem dilengkapi dengan mekanisme retry dan pemrosesan asinkron menggunakan ThreadPoolExecutor agar proses deteksi tidak terganggu oleh latency jaringan.

Pengujian sistem dilakukan secara langsung menggunakan kamera iPhone 12 Pro yang dikonfigurasi sebagai webcam melalui koneksi jaringan internet. Kamera ditempatkan pada portal gerbang utama Universitas Lampung dengan ketinggian 2,5 meter dan sudut kemiringan 45 derajat terhadap sumbu horizontal. Penempatan kamera didesain berdasarkan perhitungan Field of View (FOV) agar objek seperti kepala pengendara, helm, dan plat nomor dapat tertangkap secara optimal. Video yang dihasilkan diproses secara real-time untuk menguji efektivitas sistem dalam mendeteksi pelanggaran helm dan mengirimkan notifikasi.

Pengujian dilakukan dalam berbagai kondisi pencahayaan dan lalu lintas, serta melibatkan beberapa pengendara motor untuk mengamati kemampuan sistem dalam mendeteksi pelanggaran aktual. Hasil pengujian kemudian dievaluasi menggunakan matriks konfusi dengan parameter *true positive*, *false positive*, *false negative*, dan *true negative*. Metrik evaluasi seperti precision, recall, dan F1-score digunakan untuk menilai performa sistem secara kuantitatif. Dengan demikian, metode yang digunakan dalam penelitian ini mencakup seluruh proses mulai dari pengumpulan dan pelabelan data, pelatihan model, integrasi sistem dengan Telegram, hingga pengujian di lingkungan nyata sebagai dasar validasi sistem yang dikembangkan.

3. HASIL DAN PEMBAHASAN

Gambaran Umum Sistem

Sistem yang dikembangkan dalam penelitian ini merupakan kombinasi antara metode deteksi objek berbasis YOLOv8 dan sistem notifikasi real-time menggunakan Telegram Bot API. Sistem ini dirancang untuk melakukan deteksi otomatis terhadap pengendara sepeda motor yang tidak menggunakan helm, kemudian mengirimkan hasil deteksi berupa tangkapan gambar disertai informasi waktu, lokasi, dan ID pelanggaran ke akun Telegram yang telah ditentukan.

Input sistem berasal dari video streaming yang diperoleh melalui kamera webcam, yang diproses secara frame-per-frame menggunakan model deteksi YOLOv8. Model ini telah dilatih untuk mengenali empat kelas objek, yaitu: motor, helm, tanpa helm, dan plat nomor. Ketika objek "tanpa helm" terdeteksi dalam satu area bounding box yang sama dengan "motor", sistem secara otomatis mencatatnya sebagai pelanggaran dan memicu fungsi notifikasi.

Antarmuka sistem dirancang untuk menampilkan deteksi secara visual melalui bounding box berwarna berbeda, hijau untuk helm, merah untuk tanpa helm, kuning untuk motor, dan putih untuk plat. Selain visualisasi langsung, sistem juga mencatat metrik evaluasi berupa TP, FP, FN, dan TN secara real-time guna keperluan analisis performa.



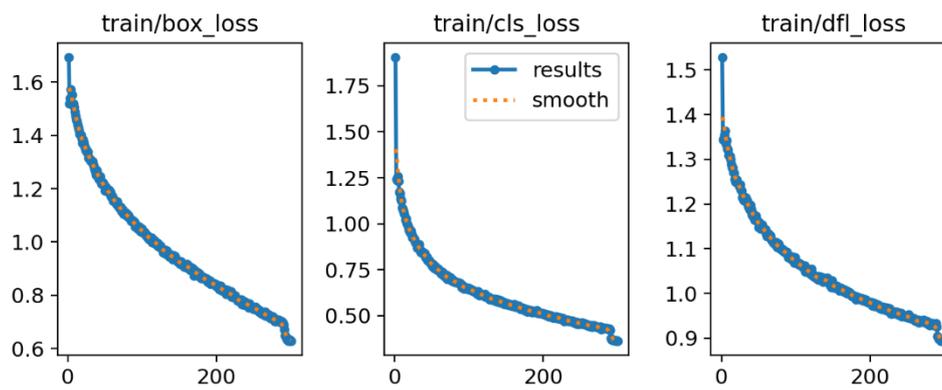
Gambar 1. Penempatan Kamera Pada Portal Gerbang UNILA

Hasil Pelatihan Model YOLOv8

Model YOLOv8 dilatih selama 300 epoch menggunakan dataset sebanyak 6.152 gambar yang terbagi menjadi 70% data latih, 20% data validasi, dan 10% data uji. Model menggunakan ukuran input 640x640 piksel dengan batch size sebesar 16. Pelatihan dilakukan menggunakan GPU NVIDIA RTX 3060 dengan pengaturan precision setengah (FP16), untuk efisiensi waktu dan penggunaan memori. File bobot terbaik hasil pelatihan disimpan dalam format .pt untuk digunakan pada tahap inferensi dan pengujian sistem secara langsung.

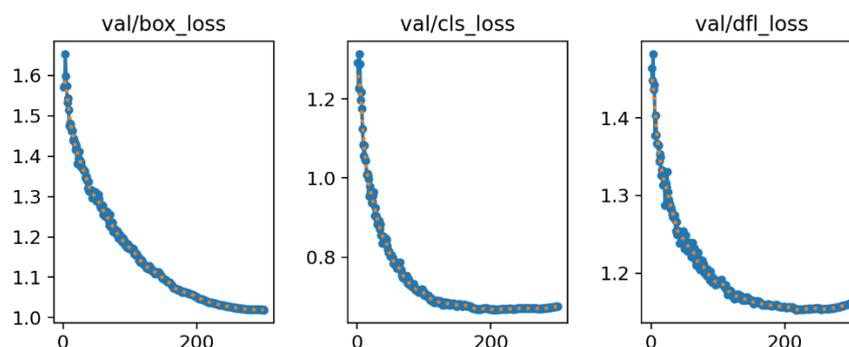
Selama proses pelatihan, kinerja model dipantau menggunakan grafik *training loss* yang terdiri dari tiga komponen utama, yaitu *box_loss*, *cls_loss*, dan *dfl_loss*. Grafik training loss

menunjukkan tren penurunan yang konsisten sejak awal hingga akhir epoch. Pada awal pelatihan, nilai `box_loss` berada di kisaran 1.6 dan menurun hingga mencapai 0.65, yang mengindikasikan bahwa model semakin akurat dalam memprediksi posisi bounding box dari objek. Nilai `cls_loss` (kesalahan klasifikasi) turun dari 1.8 menjadi 0.45, menandakan kemampuan model dalam membedakan antara kelas helm, tanpa helm, motor, dan plat menjadi lebih baik. Nilai `df_l_loss` (distribution focal loss), yang berperan dalam meningkatkan akurasi prediksi bounding box, juga menurun dari 1.5 menjadi 0.9. Penurunan nilai ketiga loss tersebut mengonfirmasi bahwa model berhasil belajar secara progresif dan stabil tanpa mengalami overfitting pada data latih.



Gambar 2. Grafik Training Loss

Grafik *validation loss* dievaluasi untuk melihat kemampuan generalisasi model terhadap data baru. Hasil grafik menunjukkan bahwa nilai `val_box_loss` turun dari 1.6 menjadi sekitar 1.02, sedangkan `val_cls_loss` menurun dari 1.3 menjadi sekitar 0.68. Penurunan ini relatif stabil, meskipun menjelang akhir pelatihan terdapat sedikit fluktuasi. Hal tersebut masih dalam batas wajar dan tidak mengindikasikan gejala overfitting. `Val_dfl_loss` juga turun dari 1.5 ke 1.15, menguatkan bukti bahwa model tidak hanya menghafal pola pada data pelatihan, tetapi juga mampu memproyeksikan performa yang baik terhadap data validasi.



Gambar 3. Grafik Validation Loss

Evaluasi model selanjutnya dilakukan menggunakan dataset validasi sebanyak 1.218 gambar dengan total 7.423 instance objek dari empat kelas target: helm, tanpa helm, motor, dan plat nomor. Evaluasi dilakukan dengan metrik klasifikasi standar yaitu precision, recall, mAP@50, dan mAP@50:95. Model diuji menggunakan file bobot terbaik (best.pt) dengan framework YOLOv8 versi 2.58 di lingkungan GPU.

Hasil evaluasi menunjukkan performa yang sangat baik. Untuk kelas helm, precision mencapai 87,4% dan recall 80,1%. Kelas motor memperoleh precision 88,3% dan recall 84,6%. Kelas tanpa helm yang menjadi fokus utama penelitian menunjukkan precision sebesar 83,0% dan recall 79,9%, dengan mAP@50 sebesar 81,5%. Sementara itu, untuk kelas plat nomor, precision sebesar 84,0% dan recall 86,1%. Secara keseluruhan, model mencapai rata-rata precision sebesar 85,7%, recall 82,7%, mAP@50 sebesar 84,9%, dan mAP@50:95 sebesar 59,8%.

Tabel 1. Evaluasi Hasil Pelatihan YOLOv8

Kelas	Precision	Recall	mAP@50	mAP@50:95
Helm	87.4%	80.1%	82.7%	55.7%
Motor	88.3%	84.6%	87.7%	68.8%
Tanpa Helm	83.0%	79.9%	81.5%	65.2%
Plat Nomor	84.0%	86.1%	87.7%	49.7%

Hasil ini menunjukkan bahwa model tidak hanya unggul dari sisi kecepatan (karakteristik YOLOv8), tetapi juga memiliki akurasi yang tinggi dan dapat digunakan pada aplikasi nyata. Seluruh metrik evaluasi membuktikan bahwa model tidak mengalami degradasi performa saat diuji terhadap data baru. Performa stabil ini menjadi fondasi kuat untuk integrasi sistem lebih lanjut dengan Telegram sebagai alat notifikasi real-time.

```

300 epochs completed in 6.658 hours.
Optimizer stripped from runs\detect\train3\weights\last.pt, 22.5MB
Optimizer stripped from runs\detect\train3\weights\best.pt, 22.5MB

Validating runs\detect\train3\weights\best.pt...
Ultralytics YOLOv8.2.58 Python-3.9.13 torch-2.5.1+cu121 CUDA:0 (NVIDIA GeForce RTX 3060 Laptop GPU, 6144MiB)
Model summary (fused): 168 layers, 11,127,132 parameters, 0 gradients, 28.4 GFLOPs
Class      Images  Instances  Box(P)   R        mAP50  mAP50-95: 100% ██████████ 39/39 [00:11<00:00, 3.37it/s]
  all         1218     7423      0.857    0.827    0.849    0.598
  helm        848     2383      0.874    0.801    0.827    0.557
  motor      1123     2463      0.883    0.846    0.877    0.688
  tanpa helm  478     817       0.83     0.799    0.815    0.652
  plat       703     1760      0.84     0.861    0.877    0.497
Speed: 0.2ms preprocess, 3.7ms inference, 0.0ms loss, 1.0ms postprocess per image
Results saved to [1]runs\detect\train3[0]m

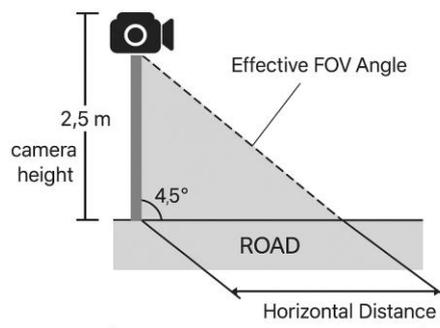
```

Gambar 4. Hasil Validasi YOLOv8

Penempatan Kamera dan Perhitungan Field of View (FOV)

Penempatan kamera dilakukan pada portal gerbang Universitas Lampung, dengan ketinggian 2,5 meter dan sudut kemiringan 45° . Kamera diarahkan ke arah jalur kendaraan masuk agar dapat menangkap bagian atas tubuh pengendara, termasuk kepala, serta posisi helm dan plat nomor. Perhitungan FOV dilakukan menggunakan panjang fokus lensa 5,1 mm dan ukuran sensor 5,76 mm x 4,29 mm.

Hasil perhitungan menunjukkan bahwa lebar cakupan horizontal mencapai ± 6 meter, cukup untuk mencakup dua jalur kendaraan. Penempatan ini terbukti optimal berdasarkan hasil visual yang menunjukkan posisi kepala pengendara terekam dengan jelas pada berbagai kondisi. Konfigurasi ini mendukung efektivitas sistem dalam mendeteksi helm maupun tanpa helm.



Gambar 5. Skema Visual Penempatan Kamera

Pengujian Sistem Deteksi Pengendara Motor Tanpa Helm dengan Notifikasi *Real-Time* melalui Telegram

Pengujian sistem dilakukan secara langsung menggunakan kamera iPhone 12 Pro yang dikonversi menjadi webcam. Sistem diuji pada siang dan sore hari dalam lima kali sesi pengujian. Setiap pelanggaran yang terdeteksi akan menghasilkan tangkapan gambar dan dikirim ke akun Telegram secara otomatis. Notifikasi mencakup ID pelanggaran, waktu kejadian, dan lokasi. Gambar 6 memperlihatkan salah satu hasil notifikasi pelanggaran yang dikirim sistem.



Gambar 6. Notifikasi Pelanggaran

Evaluasi dilakukan menggunakan matriks konfusi (TP, FP, FN, TN) dan dihitung metrik precision, recall, serta F1-score. Hasilnya disajikan pada Tabel 2 berikut.

Tabel 2. Evaluasi Hasil Pelatihan YOLOv8 (Per Kelas)

Uji ke	TP	FP	FN	TN	Precision	Recall	F1-Score
1	26	0	46	1265	1.00	0.3611	0.5306
2	9	0	10	681	1.00	0.4737	0.6429
3	6	0	4	430	1.00	0.6000	0.7500
4	12	0	2	433	1.00	0.8571	0.9231
5	11	0	6	665	1.00	0.6471	0.7857
Rata-rata					1.00	0.5878	0.7265

Sistem menunjukkan *precision* 100% pada seluruh pengujian, menandakan tidak ada pelanggaran palsu (false positive). Namun, nilai recall masih bervariasi, berkisar antara 36% hingga 85%, menandakan adanya pelanggaran yang luput terdeteksi karena faktor teknis seperti posisi kepala yang tertutup, sudut kamera, atau pencahayaan.

Analisa dan Pembahasan

Hasil penelitian menunjukkan bahwa sistem deteksi pelanggaran helm berbasis YOLOv8 yang dikembangkan mampu bekerja secara real-time dan memberikan notifikasi secara otomatis dengan tingkat precision yang sangat tinggi. Ini membuktikan efektivitas penggunaan arsitektur YOLOv8 dalam aplikasi dunia nyata, sebagaimana disampaikan oleh Redmon et al. (2016) dan Jocher et al. (2025) bahwa YOLO sangat sesuai untuk skenario pemrosesan cepat seperti pengawasan lalu lintas.



Penerapan notifikasi melalui Telegram memberikan nilai tambah signifikan. Telegram Bot API terbukti responsif dan dapat diintegrasikan secara efisien, mendukung sistem otonom berbasis AI seperti yang dibahas oleh Singh et al. (2022). Keunggulan utama dari sistem ini adalah mampu bekerja tanpa intervensi manusia, dari proses akuisisi video, deteksi objek, hingga pengiriman notifikasi.

Nilai *recall* yang tidak selalu tinggi menunjukkan masih ada ruang untuk peningkatan. Hal ini terutama disebabkan oleh faktor sudut pandang kamera, jarak, atau gangguan visual seperti bayangan atau pengendara yang tertutupi. Untuk meningkatkan *recall*, sistem dapat ditingkatkan dengan menambah jumlah kamera dari berbagai sudut atau menggunakan kamera resolusi tinggi dengan frame rate lebih tinggi.

Dibandingkan dengan penelitian sebelumnya seperti Batubara et al. (2024) yang menggunakan YOLOv5 dan hanya menguji sistem dalam lingkungan simulasi, penelitian ini telah mengimplementasikan sistem secara langsung di lapangan dengan integrasi sistem notifikasi, menjadikannya lebih siap untuk diadopsi dalam sistem pengawasan nyata.

4. KESIMPULAN

Sistem Penelitian ini berhasil mengembangkan sistem deteksi otomatis pelanggaran helm berbasis algoritma YOLOv8 yang terintegrasi dengan sistem notifikasi real-time melalui Telegram. Berdasarkan hasil pelatihan selama 300 epoch dan validasi terhadap 1.218 citra uji, model YOLOv8 menunjukkan performa deteksi yang baik dengan rata-rata precision sebesar 85,7% dan mAP@50 sebesar 84,9%. Nilai loss pada grafik pelatihan mengalami penurunan yang konsisten tanpa gejala overfitting, menandakan kemampuan generalisasi yang baik dari model terhadap data baru.

Implementasi sistem di lingkungan nyata, dengan input berupa video streaming dari webcam yang dipasang pada portal gerbang kampus, menunjukkan bahwa sistem mampu mendeteksi pengendara tanpa helm dan mengirimkan tangkapan gambar pelanggaran secara otomatis melalui Telegram. Evaluasi di lapangan menghasilkan nilai precision sebesar 100% dalam seluruh sesi pengujian, yang berarti seluruh pelanggaran yang terdeteksi benar-benar valid. Namun, nilai *recall* yang bervariasi antara 36% hingga 85% menunjukkan bahwa masih terdapat pelanggaran yang tidak terdeteksi, yang kemungkinan disebabkan oleh keterbatasan sudut kamera, kecepatan objek, atau kualitas pencahayaan. Secara keseluruhan, sistem ini terbukti layak diterapkan sebagai solusi teknologi dalam mendukung pemantauan lalu lintas dan penegakan aturan penggunaan helm secara efisien dan otonom. Integrasi antara deteksi



visual dan sistem notifikasi menjadikan sistem ini tidak hanya mendeteksi pelanggaran, tetapi juga langsung menyampaikan informasi tersebut kepada pihak terkait secara real-time.

Perlu dilakukan optimasi terhadap sudut dan posisi kamera untuk memperjelas visual bagian kepala pengendara, serta peningkatan resolusi, frame rate, dan pencahayaan guna mengurangi kesalahan deteksi akibat kualitas gambar. Penggunaan algoritma pelacakan adaptif seperti ByteTrack juga disarankan untuk menghindari deteksi ganda. Ke depan, sistem ini dapat dikembangkan lebih lanjut dengan menambahkan dashboard monitoring real-time dan penyimpanan data pelanggaran ke dalam basis data guna mendukung analisis dan pelaporan. Selain itu, perluasan fungsi sistem ke jenis pelanggaran lalu lintas lain serta pengujian di berbagai lokasi dan kondisi cuaca akan memperkuat validitas dan ketahanan sistem dalam skenario dunia nyata.

DAFTAR PUSTAKA

- Badan Pusat Statistik. (2023). *Statistik transportasi darat 2023*. BPS.
- Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. In *Proceedings of IEEE Computer Vision and Pattern Recognition (CVPR)* (pp. 886-893).
- Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 580-587).
- Gonzalez, R. C., Woods, R. E., & Eddins, S. L. (2019). *Digital image processing using MATLAB* (3rd ed.). McGraw-Hill.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
- Hasan, M. M., Islam, M. S., & Rahman, M. A. (2018). Helmet detection on motorcyclists using image processing and machine learning techniques. *IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*.
- Jain, A. K. (1989). *Fundamentals of digital image processing*. Prentice-Hall.
- Jocher, G., Chaurasia, A., Qiu, J., & Shah, N. (2025). YOLO by Ultralytics. Retrieved May 22, 2025, from <https://github.com/Ultralytics/Ultralytics>
- Kementerian Perhubungan Republik Indonesia. (2025). Tekan angka kecelakaan lalu lintas, Kemenhub ajak masyarakat beralih ke transportasi umum dan utamakan keselamatan berkendara. Retrieved May 22, 2025, from <https://dephub.go.id/>
- Korlantas Polri. (2023). *Data kecelakaan lalu lintas tahun 2022*. Jakarta, Indonesia.
- Korlantas Polri. (2024). *Jumlah data kendaraan per pulau*. Retrieved August 2, 2024, from <http://rc.korlantas.polri.go.id:8900/eri2017/laprekappulau.php>



- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems* (Vol. 25, pp. 1097-1105).
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436-444.
- Liu, L., et al. (2020). Deep learning for generic object detection: A survey. *International Journal of Computer Vision*, 128, 261-318.
- Peraturan Pemerintah Republik Indonesia Nomor 55 Tahun 2012 tentang Kendaraan. (2012). Indonesia.
- Powers, D. M. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2(1), 37-63.
- Qadri, M. T., & Naghmi, M. Y. (2009). Automatic number plate recognition system for vehicle identification using optical character recognition. In *2009 International Conference on Education Technology and Computer* (pp. 335-338). Singapore.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 779-788).
- Singh, M., Wang, L., Nguyen, H. D., & Ahamed, S. R. (2022). Real-time monitoring and alert system using Telegram Bot API integrated with Python for IoT and AI applications. *International Journal of Emerging Technology and Advanced Engineering*, 12(5), 34-40.
- Szeliski, R. (2010). *Computer vision: Algorithms and applications*. Springer.
- Telegram Messenger LLP. (2025). Telegram Bot API. Retrieved May 22, 2025, from <https://core.telegram.org/bots/api>
- Ultralytics. (2025). YOLOv8. Retrieved May 23, 2025, from <https://docs.ultralytics.com>
- Undang-Undang Republik Indonesia Nomor 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan. (2009). Indonesia.