
Penerapan Algoritma YOLOv8 Nano untuk Vehicle Tracking pada KM Tiga Putra

Fery Santria¹, Vilianty Rafida², dan Ulfah Nurfadhila³

1,2Program Studi Teknik Informatika, STMIK Widya Cipta Dharma, 3Program Studi Sistem Infromasi, STMIK Widya Cipta Dharma

Jl. M. Yamin No.25, Gn. Kelua, Kec. Samarinda Ulu, Kota Samarinda, Kalimantan Timur 75123

E-mail: 2243008@wicida.ac.id,¹, AhmadFajri@wicida.ac.id,², Presa@wicida.ac.id³ ekaarry@wicida.ac.id⁴

ABSTRACT

The load capacity management on the deck of the KM Tiga Putra ferry currently relies on manual recording, which is prone to inefficiency and the risk of overloading. This research aims to automate the monitoring by designing a Computer Vision system using the Python programming language (version 3.10+) and the Ultralytics library within the Visual Studio Code (VS Code) environment. The developed system integrates the YOLOv8 Nano architecture and the ByteTrack algorithm to precisely detect, track, and count vehicles based on a Dataset recorded by a Logitech C270 Webcam (720 pixel, 30 FPS) and pre-processed through the Roboflow platform. Considering the implementation is conducted on a low-specification perangkat keras environment a laptop running the Windows 11 operating system, an Intel Celeron processor, 8GB of RAM, and Solid State Drive (SSD) storage the computational load is optimized by implementing a frame skipping technique to prevent memory overload. Furthermore, the standard counting logic is specifically modified by eliminating the entry-exit boundary lines (In/Out counting). Instead, the system utilizes a set accumulation method along with Region of Interest (ROI) delimitation to eliminate double counting caused by random vehicle maneuvers in the parking area. The computational evaluation results demonstrate a very high Mean Average Precision (mAP) value with near-perfect classification accuracy for cars and motorcycle. Minor deviations are only found in motorcycle detection, strictly caused by visual occlusion factors where objects cover each other.

Keywords: Computer Vision, Vehicle Tracking, YOLOv8 Nano, ByteTrack, KM Tiga Putra.

Penerapan Algoritma YOLOv8 Nano untuk Vehicle Tracking pada KM Tiga Putra

ABSTRAK

Manajemen kapasitas muatan pada geladak kapal feri KM Tiga Putra saat ini masih mengandalkan pencatatan manual yang rentan terhadap inefisiensi dan berisiko menyebabkan kelebihan muatan. Penelitian ini bertujuan untuk mengotomatisasi pengawasan tersebut dengan merancang sistem *Computer Vision* menggunakan bahasa pemrograman *Python* (versi 3.10+) dan pustaka *Ultralytics* di dalam lingkungan kerja *Visual Studio Code* (VS Code). Sistem yang dibangun mengintegrasikan arsitektur *YOLOv8 Nano* dan algoritma *ByteTrack* untuk mendeteksi, melacak, serta menghitung kendaraan secara presisi berdasarkan rekaman *Dataset* dari Webcam Logitech C270 (720 piksel, 30 FPS) yang dipra-pemrosesan melalui platform *Roboflow*. Lebih lanjut, logika penghitungan standar dimodifikasi secara khusus dengan menghilangkan garis batas masuk-keluar (penghitungan *In/Out*). Sebagai gantinya, sistem menggunakan metode akumulasi himpunan (*set accumulation*) beserta pembatasan *Region of Interest* (ROI) untuk menghilangkan penghitungan ganda yang disebabkan oleh manuver kendaraan yang acak di area parkir. Hasil evaluasi komputasi menunjukkan nilai *Mean Average Precision* (mAP) yang sangat tinggi dengan akurasi klasifikasi yang nyaris sempurna untuk mobil dan sepeda motor. Penyimpangan kecil hanya ditemukan pada deteksi sepeda motor, yang semata-mata disebabkan oleh faktor oklusi visual di mana objek saling menutupi.

Kata Kunci: YOLOv8 Nano, Pelacakan Kendaraan, ByteTrack, Computer Vision, Geladak Kapal.

1. PENDAHULUAN

Transportasi sungai masih menjadi sarana penting bagi mobilitas masyarakat dan distribusi logistik di wilayah Kalimantan Timur yang memiliki keterbatasan akses darat. Pada KM Tiga Putra, pencatatan kendaraan di geladak masih mengandalkan pengamatan manual. Kondisi tersebut rentan menghasilkan ketidakonsistenan data ketika arus kendaraan meningkat dan ketika petugas harus mengawasi beberapa kendaraan secara bersamaan.

Penelitian deteksi kendaraan sebelumnya lebih banyak dilakukan pada CCTV jalan raya dengan YOLOv4 atau varian YOLO lainnya [1], [2], [7], [8]. Lingkungan geladak kapal memiliki karakteristik berbeda, antara lain ruang sempit, sudut kamera dari atas, kepadatan kendaraan yang tidak teratur, oklusi antarkendaraan, dan keterbatasan perangkat komputasi. Celah penelitian terletak pada kebutuhan model ringan yang tetap akurat serta mampu dipadukan dengan pelacakan identitas kendaraan pada perangkat tanpa GPU khusus.

YOLOv8 Nano dipilih karena merupakan varian paling ringan pada keluarga YOLOv8 dan mendukung deteksi objek secara anchor-free [3], [4]. Model ini diintegrasikan dengan ByteTrack untuk mempertahankan identitas kendaraan antarframe. Penelitian difokuskan pada evaluasi kualitas deteksi, kestabilan pelacakan, dan beban komputasi pada laptop Intel Celeron. ROI digunakan untuk membatasi area pengamatan yang relevan pada geladak.

2. RUANG LINGKUP

2.1 Rumusan Masalah

Berdasarkan uraian pada latar belakang di atas, maka rumusan masalah dalam penelitian ini adalah sebagai berikut: “Bagaimana penerapan YOLOv8 Nano untuk pelacakan objek dapat mendeteksi dan menghitung kendaraan jenis motor dan mobil secara otomatis pada perangkat keras terbatas di operasional kapal feri di Desa Long Beleh Modang?”

2.2 Batasan Masalah

1. **Objek.** Objek penelitian dibatasi pada kelas Mobil dan Sepeda Motor.
2. **Data.** Data berasal dari rekaman siang hari pada KM Tiga Putra dan tidak mencakup kondisi hujan atau malam.
3. **Model.** Deteksi menggunakan YOLOv8 Nano hasil fine-tuning, sedangkan pelacakan menggunakan ByteTrack.
4. **Perangkat.** Implementasi diuji pada laptop Intel Celeron dengan RAM 8 GB tanpa GPU khusus.
5. **Evaluasi.** Evaluasi mencakup mAP@50, kurva precision-recall, confusion matrix, waktu inferensi, dan pengamatan kestabilan tracking ID.

2.3 Tujuan Penelitian

Menerapkan dan mengevaluasi YOLOv8 Nano dengan pelacakan objek untuk mendeteksi kendaraan motor dan mobil secara otomatis pada perangkat keras terbatas di operasional Kapal Feri KM Tiga Putra, Desa Long Beleh Modang, guna mengatasi ketidakakuratan pendataan manual, kelemahan konvensional.

2.4 Manfaat Penelitian

1. **Bagi Akademik**
Manfaat bagi akademik adalah memberikan kontribusi ilmiah dalam pengembangan ilmu pengetahuan di bidang *computer vision*, khususnya mengenai efektivitas dan optimasi penerapan algoritma YOLOv8 Nano pada perangkat keras dengan spesifikasi terbatas (*resource-constrained perangkat keras*), serta dapat menjadi rujukan atau referensi bagi peneliti selanjutnya yang ingin mengembangkan sistem pemantauan objek otomatis di lingkungan transportasi sungai.
2. **Bagi KM Tiga Putra**
Manfaat bagi KM Tiga Putra, khususnya pengelola Kapal Feri KM Tiga Putra, adalah tersedianya solusi teknologi untuk mendigitalisasi proses pendataan *manifest* kendaraan secara otomatis yang lebih akurat dan efisien dibandingkan metode manual, sehingga pengelola dapat meminimalisir risiko *human*

error dalam penghitungan muatan guna menjaga standar keselamatan pelayaran.

3. Bagi Masyarakat

Manfaat bagi masyarakat adalah meningkatnya kualitas layanan dan keamanan transportasi penyeberangan di wilayah Desa Long Beleh Modang melalui penerapan sistem monitoring muatan yang lebih transparan dan terukur, sehingga masyarakat sebagai pengguna jasa kapal feri dapat merasa lebih aman dan nyaman dalam melakukan mobilitas serta distribusi logistik antar wilayah.

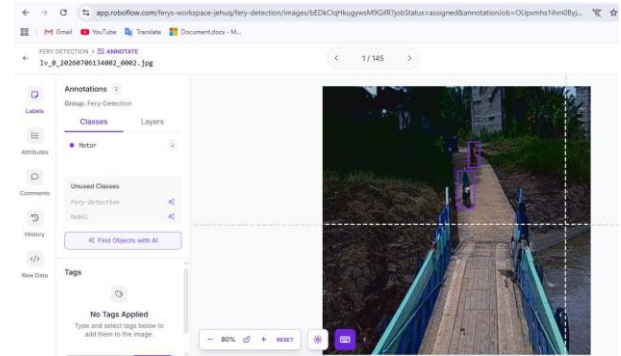
3. BAHAN DAN METODE

3.1 Bahan dan Perangkat

Data direkam menggunakan webcam Logitech C270 pada resolusi 1280×720 piksel dan 30 FPS selama dua menit. Perangkat komputasi menggunakan laptop dengan prosesor *Intel Celeron*, RAM 8 GB, sistem operasi *Windows 11*, dan penyimpanan SSD. Perangkat lunak yang digunakan meliputi *Python 3.10+*, *Ultralytics YOLO*, *PyTorch*, *OpenCV*, *Roboflow*, dan *Visual Studio Code*.

3.2 Persiapan Dataset

Video geladak diekstraksi menjadi frame, kemudian gambar yang jelas dan memuat kendaraan dipilih untuk anotasi. Setiap mobil dan sepeda motor diberi bounding box pada Roboflow. Dataset akhir berjumlah 166 gambar dan seluruh citra diseragamkan menjadi 640×640 piksel.



Gambar 1. Anotasi objek kendaraan pada platform Roboflow.

Tabel 1. Pembagian dataset kendaraan

Subset	Jumlah
Training 70%	117
Validation 20%	33
Test 10%	16

3.3 Pelatihan YOLOv8 Nano

Model awal *yolov8n.pt* dilatih selama 50 epoch menggunakan ukuran masukan 640×640 piksel, batch size 2, device CPU, dan workers 0. Training set digunakan untuk memperbarui bobot, sedangkan validation set memantau nilai box loss, classification loss, precision, recall, dan mAP. Bobot dengan kinerja validasi terbaik disimpan sebagai *best.pt*.

3.4 Pelacakan dan Skenario Pengujian

Bobot *best.pt* diterapkan pada rekaman operasional menggunakan fungsi *model.track()* dengan ByteTrack dan *persist=True*. Setiap deteksi menghasilkan kelas, confidence, bounding box, dan tracking ID. ROI membatasi area geladak yang dianalisis, sedangkan frame skipping digunakan untuk mengurangi antrean frame pada perangkat CPU terbatas.

Evaluasi deteksi menggunakan precision, recall, AP tiap kelas, mAP@50, dan confusion matrix. Evaluasi komputasi menggunakan waktu preprocess, inference, dan postprocess per frame. Kestabilan pelacakan ditinjau secara visual dari kemampuan ByteTrack mempertahankan ID serta kemunculan flickering ID ketika objek mengalami oklusi.

Analisis data dilakukan dengan membaca nilai AP setiap kelas, mAP@50, distribusi kesalahan pada confusion matrix, dan waktu komputasi per frame. Waktu inferensi dikonversi menjadi

perkiraan FPS untuk menilai kesesuaian model terhadap sumber video 30 FPS. Hasil tracking dianalisis secara visual berdasarkan kontinuitas tracking ID pada kondisi normal dan saat terjadi oklusi.

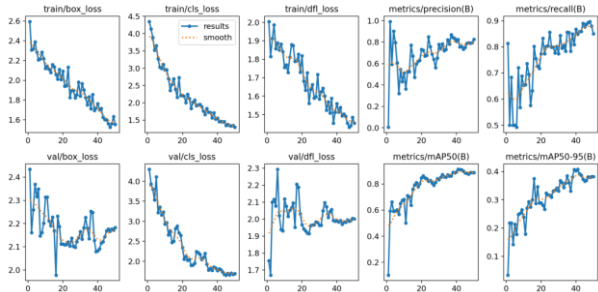


Gambar 2. Alur penerapan dan evaluasi YOLOv8 Nano.

4. HASIL DAN PEMBAHASAN

4.1 Hasil Pelatihan Model

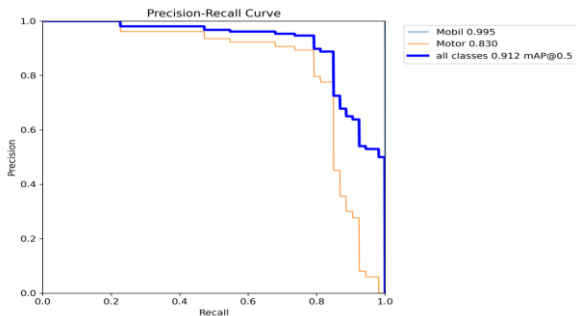
Pelatihan selama 50 epoch menghasilkan penurunan box loss dan classification loss pada data pelatihan. Kurva validasi berfluktuasi lebih besar karena jumlah validation set hanya 33 gambar, tetapi mAP@50 meningkat dan stabil mendekati 0,9 pada akhir pelatihan. Pola tersebut menunjukkan model telah mempelajari dua kelas kendaraan, namun ukuran dataset yang terbatas tetap membatasi generalisasi terhadap kondisi visual yang belum terwakili.



Gambar 3. Kurva metrik pelatihan YOLOv8 Nano selama 50 epoch.

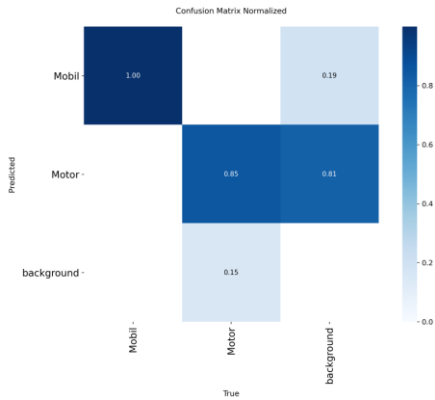
4.2 Evaluasi Akurasi Deteksi

Kurva precision-recall menghasilkan AP sebesar 0,995 untuk Mobil dan 0,830 untuk Sepeda Motor. Rata-rata kedua kelas menghasilkan mAP@50 sebesar 0,912. Selisih 16,5 poin persentase menunjukkan bahwa kelas sepeda motor merupakan sumber kesalahan utama. Ukuran objek yang lebih kecil, bentuk yang lebih bervariasi, dan posisi yang saling berhimpitan menurunkan kemampuan model mempertahankan precision ketika recall ditingkatkan.



Gambar 4. Kurva precision-recall model YOLOv8 Nano.

Confusion matrix ternormalisasi menunjukkan tingkat pengenalan 1,00 pada kelas Mobil dan 0,85 pada kelas Sepeda Motor. Tidak terlihat pertukaran kelas yang berarti antara mobil dan sepeda motor; kesalahan utama berupa objek sepeda motor yang tidak dikenali dan masuk ke kategori background sebesar 0,15. Dengan demikian, masalah dominan bukan salah kelas, melainkan false negative akibat oklusi atau keterbatasan fitur visual.



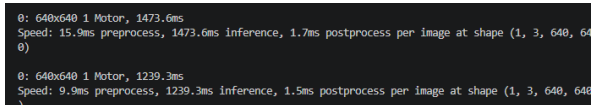
Gambar 5. Confusion matrix ternormalisasi hasil klasifikasi.

Tabel 2. Ringkasan hasil evaluasi model

Parameter	Nilai
AP Mobil	0,995
AP Sepeda Motor	0,830
mAP@50	0,912
Recall Mobil	1,00
Recall Motor	0,85
Waktu inferensi	1.239,3-1.473,6 ms
Kestabilan tracking	ID cukup stabil

4.3 Performa Komputasi

Waktu preprocess tercatat 9,9–15,9 ms dan postprocess 1,5–1,7 ms per frame, sedangkan inference membutuhkan 1.239,3–1.473,6 ms. Berdasarkan waktu inferensi tersebut, laju pemrosesan efektif hanya sekitar 0,68–0,81 FPS. Nilai ini menunjukkan bahwa sistem belum memenuhi karakteristik real-time pada sumber video 30 FPS. Frame skipping mencegah penumpukan memori dan membuat proses tetap berjalan, tetapi tidak meningkatkan kapasitas inferensi model.



Gambar 6. Waktu preprocess, inference, dan postprocess pada perangkat uji.

4.4 Evaluasi Pelacakan Kendaraan

Integrasi ByteTrack memungkinkan setiap kendaraan menerima tracking ID dan mempertahankan identitas selama objek masih terlihat. Pada kondisi kendaraan tidak saling menutupi, identitas relatif stabil dan bounding box mengikuti pergerakan objek. Ketika sepeda motor tertutup badan mobil atau berimpit dengan sepeda motor lain, ID dapat hilang sementara atau berubah saat objek muncul kembali. Fenomena flickering tersebut konsisten dengan penurunan recall kelas sepeda motor pada confusion matrix.



Gambar 7. Hasil deteksi dan pelacakan kendaraan pada geladak KM Tiga Putra.

Hasil ini memperlihatkan bahwa akurasi deteksi tinggi tidak otomatis menjamin pelacakan yang stabil. Tracking bergantung pada kontinuitas deteksi antarframe; satu false negative dapat memutus lintasan ID. Oleh karena itu, peningkatan kualitas dataset pada skenario oklusi lebih relevan dibanding hanya menaikkan confidence threshold, karena threshold yang terlalu tinggi justru berpotensi meningkatkan objek yang terlewat.

4.5 Pembahasan Kritis

Kontribusi utama penelitian terletak pada penerapan model ringan di lingkungan feri sungai yang berbeda dari skenario CCTV jalan raya pada penelitian terdahulu [7], [8]. YOLOv8 Nano terbukti mampu mempertahankan mAP@50 di atas 0,9 dengan hanya 166 gambar, tetapi kesenjangan AP antar kelas dan kecepatan di bawah 1 FPS menunjukkan bahwa istilah real-time harus digunakan secara hati-hati. Sistem lebih tepat diposisikan sebagai prototipe pemantauan berbasis video yang berjalan berkelanjutan pada perangkat terbatas, bukan sebagai pelacakan real-time berkecepatan penuh.

4.6 Keterbatasan Penelitian

Keterbatasan pertama berasal dari cakupan data. Dataset 166 gambar dan test set 16 gambar belum mewakili seluruh variasi pencahayaan, cuaca, kepadatan, serta sudut kendaraan pada operasi KM Tiga Putra. Nilai akurasi yang tinggi karena itu perlu dibaca sebagai hasil pada skenario pengujian yang tersedia, bukan sebagai jaminan performa pada seluruh kondisi operasional.

Keterbatasan kedua berkaitan dengan evaluasi tracking. Penelitian belum menghitung metrik IDF1, HOTA, MOTA, atau jumlah ID switch, sehingga kualitas ByteTrack belum dapat dibandingkan secara kuantitatif dengan algoritma pelacak lain. Keterbatasan ketiga adalah perangkat uji tanpa GPU, yang menyebabkan throughput di bawah 1 FPS dan membatasi pengamatan terhadap respons sistem pada arus video berkecepatan penuh.

5. KESIMPULAN

YOLOv8 Nano yang dilatih menggunakan 166 gambar berhasil mendeteksi mobil dan sepeda motor pada geladak KM Tiga Putra dengan mAP@50 sebesar 0,912. Kinerja kelas Mobil mencapai AP 0,995 dan tingkat pengenalan 1,00, sedangkan kelas Sepeda Motor memperoleh AP 0,830 dan tingkat pengenalan 0,85. Kesalahan dominan berasal dari sepeda motor yang tidak terdeteksi akibat ukuran objek dan oklusi, bukan pertukaran kelas.

ByteTrack dapat mempertahankan identitas kendaraan ketika deteksi berlangsung kontinu, tetapi flickering ID muncul saat objek tertutup. Pada Intel Celeron dan RAM 8 GB, waktu inferensi 1.239,3–1.473,6 ms per frame menghasilkan sekitar 0,68–0,81 FPS. Dengan demikian, model layak sebagai dasar sistem pemantauan kendaraan pada geladak, tetapi belum memenuhi pemrosesan real-time untuk video 30 FPS.

6. SARAN

1. **Dataset.** Tambahkan rekaman malam, variasi cuaca, kepadatan tinggi, dan anotasi oklusi parsial untuk meningkatkan recall sepeda motor.
2. **Perangkat keras.** Gunakan GPU khusus atau perangkat edge berakselerasi agar inferensi mendekati laju video sumber tanpa bergantung pada frame skipping.
3. **Pelacakan.** Gunakan metrik IDF1, HOTA, MOTA, dan jumlah ID switch agar performa ByteTrack dapat dinilai secara kuantitatif.
4. **Akuisisi visual.** Pertimbangkan kamera multisudut untuk mengurangi kehilangan objek akibat kendaraan saling menutupi.

7. REFERENSI

1. M. R. Amanullah, R. M. Akbar, dan Y. D. Rosita, "Simulasi Deteksi dan Hitung Jumlah Kendaraan Menggunakan YOLOv4 pada CCTV Persimpangan Jalan Raya," Semastek, 2023.
2. F. Oklilas, D. Dwinta, G. Shofi, N. P. Mariza, S. A. Kinanti, dan Y. A. Sari, "Akurasi Pengujian Model Hasil Training Menggunakan YOLOv4 untuk Pengenalan Kendaraan di Jalan Raya," Repository Universitas Sriwijaya, 2023.

3. R. C. Gonzalez dan R. E. Woods, Digital Image Processing, 4th ed. Pearson, 2018.
4. Y. Heryadi dan M. Irwansyah, "Implementasi Deep Learning untuk Deteksi Objek," Jurnal Informatika Indonesia, vol. 10, no. 2, pp. 78–89, 2020.