

DOI: doi.org/10.21009/03.1301.FA14

PENGHITUNG JUMLAH KENDARAAN BERBASIS DETEKSI OBJEK MENGGUNAKAN ALGORITMA YOLOV4-TINY

Rian Setiyana^{1, a)}, Hadi Nasbey^{1, b)}, Haris Suhendar^{1, c)}

¹Program Studi Fisika, FMIPA, Universitas Negeri Jakarta, Jalan Rawamangun Muka I, Jakarta 13220, Indonesia

Email: [a\)riansetiyana27@gmail.com](mailto:a)riansetiyana27@gmail.com)

Abstrak

Kemacetan merupakan situasi dimana kondisi lalu lintas terganggu atau terhenti sama sekali. Hal ini sering ditemui di wilayah perkotaan, terutama ketika terjadi ketidakseimbangan antara jumlah kendaraan dengan kapasitas jalan yang tersedia. Jika dibandingkan dengan laju pembangunan infrastruktur jalan kota, penambahan jumlah kendaraan telah meningkat secara eksponensial. Dari tahun 2013 hingga 2017, tercatat bahwa penambahan rata-rata jumlah kendaraan setiap tahunnya sekitar 8.600.000 kendaraan. Hal ini menyebabkan tingginya tingkat kepadatan lalu lintas di wilayah perkotaan. Oleh karena itu, penelitian ini bertujuan untuk memantau kepadatan lalu lintas dengan mengembangkan sebuah penghitung jumlah kendaraan berbasis deteksi objek. Penghitung ini dibuat untuk menghitung jumlah kendaraan pada rekaman video. Dalam penelitian ini, sistem penghitung jumlah kendaraan akan memanfaatkan kemampuan deteksi objek yang dimiliki oleh algoritma You Only Look Once (YOLO), khususnya YOLOv4-tiny. Algoritma YOLOv4-tiny dipilih karena memiliki ukuran model yang lebih kecil dan kemampuan deteksi yang lebih cepat. Berdasarkan hasil pelatihan, mAP pada bobot YOLOv4-tiny terbaik mencapai 95.13%. Selain itu, hasil penghitungan menunjukkan bahwa sistem berhasil menghitung jumlah kendaraan yang melewati garis bantu berdasarkan jenisnya.

Kata-kata kunci: Kemacetan lalu lintas, Penghitungan kendaraan, YOLOv4-tiny, Deteksi Objek.

Abstract

Traffic congestion occurs when vehicle flow is disrupted or completely halted. This situation is often encountered in urban areas, especially when there is an imbalance between the number of vehicles and the available road capacity. Compared to the rate of urban road infrastructure development, the increase in the number of vehicles has grown exponentially. From 2013 to 2017, the average annual increase in the number of vehicles was approximately 8.600.000. This has led to high levels of traffic congestion in urban areas. Therefore, this study aims to monitor traffic density by developing a vehicle counting system based on object detection. This system is designed to count the number of vehicles in video recordings. In this study, the vehicle counting system utilizes the object detection capabilities of the You Only Look Once (YOLO) algorithm, specifically YOLOv4-tiny. YOLOv4-tiny was chosen for its smaller model size and faster detection capabilities. Based on the training results, the mAP of the best

YOLOv4-tiny weights reached 95.13%. Furthermore, the counting results indicate that the system successfully counted vehicles passing the auxiliary line based on their types.

Keywords: Traffic congestion, Vehicle counting, YOLOv4-tiny, Object detection.

PENDAHULUAN

Mengelola lalu lintas merupakan salah satu tantangan utama dalam manajemen kota, terutama di negara berkembang. Di seluruh dunia, kemacetan lalu lintas telah menjadi masalah yang cukup serius [1]. Kemacetan lalu lintas terjadi ketika kondisi lalu lintas terganggu atau terhenti sama sekali [2]. Penyebabnya meliputi rambu lalu lintas yang tidak teratur, penggunaan ruang jalan yang tidak efektif, kapasitas jalan yang tidak memadai, serta masyarakat yang lebih memilih untuk menggunakan kendaraan pribadi [3]. Di wilayah perkotaan, kemacetan sering terjadi karena terdapat ketidakseimbangan antara jumlah kendaraan dengan kapasitas jalan yang tersedia [2]. Jika dibandingkan dengan laju pembangunan infrastruktur jalan kota, penambahan jumlah kendaraan telah meningkat secara eksponensial [4]. Dari tahun 2013 hingga 2017, rata-rata jumlah kendaraan bertambah sekitar 8.600.000 setiap tahunnya [5]. Berdasarkan data dari Badan Pusat Statistik, pada tahun 2020 jumlah kendaraan meningkat menjadi 136.137.451 unit [6]. Hal ini tentu menyebabkan tingginya tingkat kepadatan lalu lintas di wilayah perkotaan [4].

Untuk mengatasi masalah ini, diperlukan solusi teknologi yang mampu mengelola lalu lintas secara lebih efektif. Salah satu pendekatan yang menjanjikan adalah penggunaan teknologi deteksi objek berbasis *computer vision*. *Computer vision* merupakan cabang ilmu komputer yang berfokus pada pengembangan sistem yang dapat mengamati, mengenali, dan memahami gambar serta video. Tujuan utama dari *computer vision* adalah memungkinkan komputer atau mesin untuk menafsirkan dan memahami dunia visual seperti manusia [7]. Salah satu aplikasi dalam bidang *computer vision* adalah melakukan pengawasan, seperti pendeteksian objek kendaraan [8].

Sejumlah penelitian mengenai penghitungan jumlah kendaraan telah dilakukan, salah satunya dengan menggunakan algoritma *You Only Look Once* (YOLO), terutama YOLOv3 [1][9]. Penelitian tersebut memanfaatkan kemampuan deteksi objek YOLOv3 untuk menghitung jumlah kendaraan. Berbeda dengan penelitian sebelumnya, penelitian ini membuat penghitung jumlah kendaraan berbasis deteksi objek menggunakan algoritma YOLOv4-tiny. Algoritma YOLOv4-tiny dipilih karena memiliki ukuran model yang lebih kecil dan kemampuan deteksi yang lebih cepat [10]. Penghitung ini dibuat untuk menghitung jumlah kendaraan dalam rekaman video, dengan jenis kendaraan yang dideteksi dan dihitung mencakup motor, mobil, bus, dan truk.

METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini meliputi berbagai tahapan, berikut tahapan-tahapan penelitiannya.

1. Akuisisi Data

Pada tahap akuisisi data, peneliti menggunakan video kendaraan yang melintas di jalan raya yang bersumber dari internet. Jumlah video yang digunakan sebanyak 4 video, dimana masing-masing video berdurasi sekitar 1-10 menit. Video ini nantinya diubah menjadi kumpulan *frame* berformat JPG pada tahap konversi data.

2. Konversi Data

Pada tahap ini, video kendaraan dikonversi menjadi kumpulan *frame* berformat JPG. Proses konversi dilakukan menggunakan *software* Video to JPG Converter.

3. Anotasi Data

Tahap ini melibatkan penandaan atau pelabelan objek kendaraan pada setiap *frame*. Penandaan tersebut dapat berupa pembuatan *bounding box* (kotak pembatas objek) di sekitar

kendaraan, sehingga model deteksi objek dapat memahami lokasi dan keberadaan kendaraan dalam gambar. Proses anotasi data dilakukan dengan menggunakan *software* LabelImg.

4. Augmentasi Data

Tahap augmentasi data bertujuan untuk meningkatkan variasi dalam dataset tanpa perlu mengumpulkan lebih banyak gambar dan melakukan anotasi secara manual. Dalam penelitian ini, augmentasi data dilakukan menggunakan platform Roboflow, yang menyediakan berbagai teknik augmentasi untuk meningkatkan kualitas dan kuantitas dataset.

5. Memisahkan Dataset

Dataset hasil augmentasi berjumlah 1694 data, terdiri dari 847 file gambar dan 847 file anotasi. Dataset ini dibagi menjadi data latih dan data uji dengan perbandingan 80:20. Sebanyak 80% dari data yang dimiliki digunakan sebagai data latih, sedangkan 20% sisanya digunakan sebagai data uji.

6. Melatih Model YOLOv4-tiny

Pada tahap ini, model akan mempelajari pola dan karakteristik yang terdapat pada data yang diberikan. Proses pelatihan ini bertujuan agar model dapat mengenali dan memahami berbagai jenis kendaraan secara akurat.

7. Evaluasi Model YOLOv4-tiny

Evaluasi ini dilakukan menggunakan metrik *mean Average Precision* (mAP) untuk mengukur sejauh mana model dapat menghasilkan deteksi yang akurat. Metrik ini memberikan gambaran komprehensif tentang kemampuan model dalam mengidentifikasi dan menempatkan *bounding box* pada objek kendaraan.

8. Menghitung Jumlah Kendaraan

Tahap ini melibatkan penggunaan deteksi objek yang telah dilatih untuk melacak dan menghitung jumlah kendaraan yang terdapat pada video masukan.

HASIL DAN PEMBAHASAN

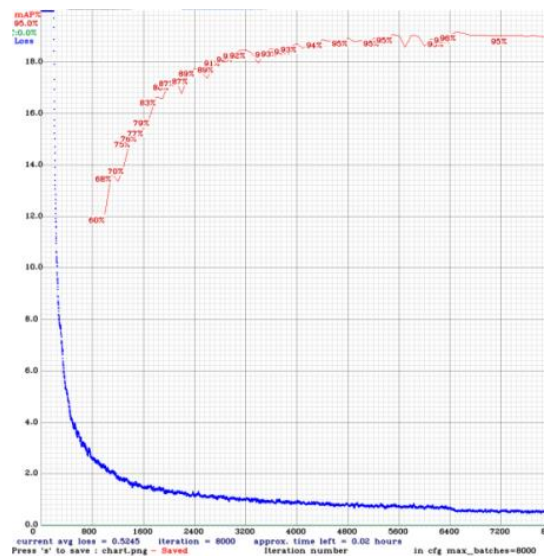
Dataset merupakan komponen utama dalam proses pelatihan model YOLOv4-tiny untuk mendeteksi objek kendaraan. Selain dataset, pengaturan parameter yang tepat pada file konfigurasi YOLOv4-tiny juga diperlukan. Berikut adalah beberapa nilai parameter yang digunakan dalam file konfigurasi tersebut.

TABEL 1. Parameter konfigurasi pelatihan.

PARAMETER	NILAI
batch	32
subdivisions	16
width	448
height	448
learning_rate	0.001
max_batches	8000
steps	6400, 7200
filters	27
classes	4

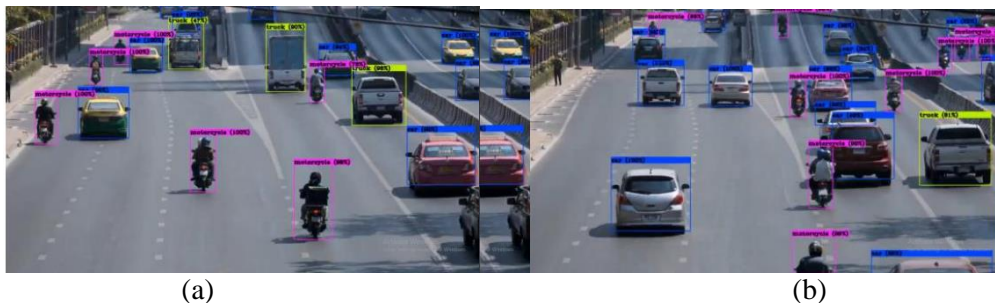
Pada proses pelatihan YOLOv4-tiny, beberapa parameter penting harus ditentukan dalam file konfigurasi. TABEL 1 menunjukkan nilai-nilai parameter yang digunakan untuk melatih model YOLOv4-tiny. Parameter-parameter tersebut memiliki peran penting dalam menentukan kinerja model selama proses pelatihan. Dengan menggunakan nilai-nilai parameter yang telah ditentukan, model YOLOv4-tiny dilatih menggunakan dataset yang telah disiapkan. Setelah proses pelatihan

selesai, maka akan didapatkan grafik yang menunjukkan nilai mAP dan *average loss* setiap iterasi selama proses pelatihan berlangsung. Berikut merupakan grafik hasil keluaran dari proses pelatihan model YOLOv4- tiny.



GAMBAR 1. Grafik yang dihasilkan setelah proses pelatihan selesai.

GAMBAR 1 memperlihatkan bahwa pada iterasi terakhir, nilai mAP mencapai 94.76% dan *average loss* sebesar 0.5245. Namun, untuk penghitungan jumlah kendaraan, peneliti akan menggunakan model dengan bobot terbaik yang memiliki nilai mAP sebesar 95.13%. Berikut ini adalah hasil deteksi kendaraan menggunakan bobot YOLOv4-tiny yang telah dilatih.



GAMBAR 2. Hasil deteksi objek menggunakan bobot YOLOv4-tiny yang sudah dilatih: (a) dan (b) merupakan salah satu frame pada video masukan.

Setelah melakukan deteksi objek menggunakan model YOLOv4-tiny yang sudah dilatih, langkah berikutnya adalah menghitung jumlah kendaraan. Sebelum melanjutkan ke proses penghitungan, bobot YOLOv4-tiny yang sudah dilatih perlu dikonversi ke dalam format TensorFlow. Konversi ini memudahkan peneliti untuk mengintegrasikan model YOLOv4-tiny yang sudah dilatih ke dalam program Python. Model tersebut akan memberikan output berupa koordinat objek yang terdeteksi, skor keyakinan (*confidence score*), ID kelas objek, dan jumlah objek yang terdeteksi dalam sebuah frame. Dengan data ini, peneliti dapat melacak keberadaan objek kendaraan yang terdeteksi. Selanjutnya, perlu dibuat garis bantu untuk memudahkan proses penghitungan jumlah kendaraan. Setiap kendaraan yang terdeteksi dan melewati garis ini akan dihitung sesuai dengan kelas objeknya (motor, mobil, bus, dan truk). Berikut merupakan tampilan saat proses penghitungan jumlah kendaraan.



GAMBAR 3. Tampilan saat proses penghitungan jumlah kendaraan yang melewati garis bantu.

GAMBAR 3 menunjukkan proses penghitungan jumlah kendaraan dalam video masukan. Setiap kendaraan akan diberi bounding box dan dihitung ketika menyentuh garis bantu. Setelah semua kendaraan dihitung, hasil penghitungannya akan ditampilkan. Berikut adalah hasil penghitungan jumlah kendaraan yang melewati garis bantu dalam video masukan.

```
Hasil akhir penghitungan jumlah kendaraan:  
{'motor': 1, 'mobil': 142, 'bus': 8, 'truk': 18}
```

GAMBAR 4. Hasil penghitungan jumlah kendaraan.

GAMBAR 4 menunjukkan bahwa proses penghitungan jumlah kendaraan berhasil menghitung kendaraan yang melewati garis bantu berdasarkan jenisnya. Namun, masih terdapat kekurangan dalam proses penghitungan ini, yaitu kendaraan yang saling menutupi satu sama lain terkadang tidak terdeteksi dan terhitung.

KESIMPULAN

Penerapan algoritma YOLOv4-tiny telah terbukti memberikan hasil yang cukup akurat dan efisien dalam mendeteksi serta menghitung jumlah kendaraan pada video masukan. Namun, masih terdapat kekurangan dalam mendeteksi kendaraan yang saling menutupi satu sama lain, menunjukkan perlunya peningkatan lebih lanjut. Meskipun demikian, secara keseluruhan, hasil ini menunjukkan bahwa metode yang digunakan dapat secara efektif mengidentifikasi dan menghitung berbagai jenis kendaraan dalam video masukan.

REFERENSI

- [1] D. A. Abdurrafi, M. T. Alawiy, and B. M. Basuki, "Deteksi klasifikasi dan menghitung kendaraan berbasis algoritma You Only Look Once (YOLO) menggunakan kamera CCTV," *Science Electro*, pp. 1-6, 2023.
- [2] S. Sahara and B. N. Nugroho, "Efektivitas penggunaan kereta listrik (KRL) Commuter Line Jabodetabek untuk mengurangi kemacetan di DKI Jakarta," *Jurnal Ekonomika45*, pp. 415-426, 2023.
- [3] M. A. Santoso, J. Raharjo, and N. Ibrahim, "Implementasi alat pemantauan kepadatan lalu lintas," *e-Proceeding of Engineering*, pp. 2981-2988, 2022.
- [4] F. Novaldi, I. Amrulloh, I. W. Wisesa, and M. C. Manullang, "Pendetections pelanggaran pada penyebrangan jalan menggunakan single-shot detector pada ESP32," *Tematik: Jurnal Teknologi Informasi Komunikasi*, pp. 119-127, 2022.
- [5] D. Agustiani, "Implementasi machine learning dan computer vision pada pengembangan sistem otomasi klasifikasi dan perhitungan kendaraan," *Seminar Nasional Dinamika Informatika 2019*, pp. 16-19, 2019.

- [6] A. A. Suradi, M. F. Rasyid, and Nasaruddin, "Sistem perhitungan jumlah kendaraan berbasis computer vision," *Prosiding Seminar Ilmiah Sistem Informasi dan Teknologi Informasi*, pp. 89- 97, 2022.
- [7] T. Sutisna, A. R. Raharja, Solihin, E. Hariyadi, and V. H. Putra, "Penggunaan computer vision untuk menghitung jumlah kendaraan dengan menggunakan metode SSD (Single Shot Detector)," *Innovative: Journal of Social Science Research*, pp. 6060-6067, 2024.
- [8] D. Indra, Herman, and F. S. Budi, "Implementasi sistem penghitung kendaraan otomatis berbasis computer vision," *Komputika: Jurnal Sistem Komputer*, pp. 53-62, 2023.
- [9] F. Rachmawati and D. Widhyaestoeti, "Deteksi jumlah kendaraan di jalur SSA Kota Bogor menggunakan algoritma deep learning YOLO," *Prosiding LPPM UIKA Bogor*, pp. 360-370, 2020.
- [10] R. M. Yusup, A. F. Anugrah, D. D. Muslimah, S. M. Permana, and S. Yuliani, "Pendeteksian objek menggunakan OpenCV dan metode YOLOv4-tiny untuk membantu tunanetra," *Journal of Computer Science and Information Technology*, pp. 59-68, 2024.