

PENENTUAN UKURAN *BATCH* OPTIMAL UNTUK PELATIHAN YOLOV8 DALAM PENDETEKSIAN OBJEK PADA KENDARAAN OTONOM

DETERMINATION OF OPTIMAL BATCH SIZE FOR YOLOV8 TRAINING IN OBJECT DETECTION IN AUTONOMOUS VEHICLES

Jeri¹⁾, Zaid Syarif Hidayat²⁾

¹⁾ Program Studi Sistem Informasi, STMIK IKMI Cirebon, Cirebon

²⁾ Program Studi Teknik Informatika, STMIK IKMI Cirebon, Cirebon

^{1,2)} Perjuangan No. 10 B Majasem, Kota Cirebon, Indonesia

Email : Muhamadjeri7@gmail.com¹⁾

Abstrak

Penelitian ini bertujuan untuk menentukan ukuran *batch* optimal dalam pelatihan model YOLOv8 untuk deteksi objek pada kendaraan otonom. Dengan meningkatnya kebutuhan akan teknologi deteksi objek yang akurat dan efisien, penelitian ini mengeksplorasi pengaruh variasi ukuran *batch* terhadap kinerja model YOLOv8. Dataset yang digunakan dalam penelitian ini adalah dataset simulasi lalu lintas dari CARLA, yang diperoleh dari Roboflow universe, terdiri dari 1719 gambar yang dibagi menjadi data pelatihan, validasi, dan pengujian. Metodologi penelitian meliputi pengumpulan data, pra-pemrosesan, dan analisis data menggunakan teknik YOLOv8 dengan pengaturan *hyperparameter* yang berbeda. Hasil penelitian menunjukkan bahwa peningkatan jumlah *epoch* dan ukuran *batch* berkontribusi pada peningkatan nilai *mean Average Precision (mAP)* model. Skema pelatihan terbaik diidentifikasi dengan nilai *mAP* tertinggi sebesar 98,2%, menggunakan 100 *epoch*, ukuran *batch* 32, dan resolusi gambar 640x640. Temuan ini memberikan wawasan penting bagi pengembangan lebih lanjut dalam teknologi deteksi objek, serta memberikan panduan bagi peneliti yang ingin mengoptimalkan parameter pelatihan untuk model deteksi objek menggunakan YOLOv8 dalam konteks kendaraan otonom. Penelitian ini diharapkan dapat menjadi referensi bagi studi-studi selanjutnya dalam bidang ini.

Kata kunci: YOLOv8, deteksi objek, kendaraan otonom, ukuran *batch* optimal, CARLA dataset, *mean Average Precision (mAP)*, *hyperparameter*, pelatihan model.

Abstract

This study aims to determine the optimal batch size in training the YOLOv8 model for object detection in autonomous vehicles. With the increasing need for accurate and efficient object detection technology, this study explores the effect of batch size variation on the performance of the YOLOv8 model. The dataset used in this study is a traffic simulation dataset from CARLA, obtained from the Roboflow universe, consisting of 1719 images divided into training, validation, and testing data. The research methodology includes data collection, pre-processing, and data analysis using the YOLOv8 technique with different *hyperparameter* settings. The results showed that increasing the number of *epochs* and batch size contributed to the increase in the *mean Average Precision (mAP)* value of the model. The best training scheme was identified with the highest *mAP* value of 98.2%, using 100 *epochs*, batch size 32, and image resolution 640x640. These findings provide important insights for further development in object detection technology, as well as provide guidance for researchers who want to optimize training parameters for object detection models using YOLOv8 in the context of autonomous vehicles. This research is expected to serve as a reference for future studies in this field.

Kata kunci: YOLOv8, object detection, autonomous vehicle, optimal batch size, CARLA dataset, *mean Average Precision (mAP)*, *hyperparameters*, model training

1. PENDAHULUAN

Penelitian sebelumnya telah menunjukkan bahwa ukuran *batch* ideal mungkin bervariasi tergantung pada dataset dan model arsitektur yang digunakan. Oleh karena itu, diperkirakan bahwa dengan memahami ukuran *batch* ideal, kinerja model kerja YOLOv8 dalam deteksi objek dapat ditingkatkan secara signifikan [1]. Model pelatihan YOLOv8 dilakukan menggunakan ukuran *batch* 4, 8, 16, 32 dan 64 untuk mengamati efek pada akurasi model dan kecepatan konvergensi dengan dataset CARLA, yang banyak digunakan dalam simulasi kendaraan otonom, tersedia melalui Roboflow Universe dan menyediakan gambar-gambar jalan serta lingkungan perkotaan

yang beragam, mencakup berbagai objek seperti kendaraan, rambu lalu lintas dan pejalan kaki[2]. Hal ini menjadikannya dataset yang ideal untuk pengujian sistem deteksi objek. Setelah melakukan penyelidikan menyeluruh, hasilnya menunjukkan bahwa ukuran serangkaian 32 memberikan keseimbangan terbaik antara akurasi dan kecepatan konvergensi. Karena itu, ukuran batch optimal yang direkomendasikan yang umumnya dapat digunakan dalam pelatihan YOLOv8 adalah 32 batch[3].

Menunjukkan bahwa Penggunaan ukuran batch yang terlalu kecil seperti 16 cenderung menghasilkan mAP yang lebih rendah dan waktu pelatihan yang lebih lama dibandingkan ukuran batch yang lebih besar[4]. Namun, ukuran batch yang terlalu besar seperti 64 tidak selalu meningkatkan akurasi dan sering memerlukan lebih banyak memori, meskipun bisa mengurangi waktu pelatihan[5]. Untuk alasan ini, memilih ukuran batch yang tepat sangat penting saat melatih model YOLOv8 untuk mencapai keseimbangan antara akurasi dan kecepatan konvergensi. Bahwa dengan pilihan ukuran serangkaian yang tepat, model YOLOv8 dapat mencapai akurasi ideal tanpa memerlukan waktu pelatihan yang terlalu lama. Dengan demikian, diharapkan bahwa studi ini akan memberikan informasi yang berguna bagi para peneliti masa depan yang ingin mengoptimalkan parameter pelatihan untuk model deteksi objek menggunakan YOLOv8.

1). **Tinjauan Pustaka**

Penelitian yang dilakukan oleh Nuno Andrade, Tiago Ribeiro, Joana Coelho, Gil Lopes, dan A. Fernando Ribeiro dari University of Minho dan University of Maia, Portugal, mengembangkan integrasi YOLOv3-tiny untuk deteksi tanda jalan sementara dan algoritma Deep Deterministic Policy Gradient (DDPG) untuk pengendalian kendaraan otonom dalam skenario jalan yang dinamis. Hasilnya menunjukkan bahwa YOLOv3-tiny mencapai tingkat ketepatan (mAP) di atas 90% dalam deteksi objek, dan DDPG berhasil mengendalikan aksi kendaraan dalam lebih dari 50% dari simulasi pengujian. Studi ini menunjukkan adanya interaksi antara deteksi objek dan pembelajaran penguatan dalam sistem navigasi otonom yang kompleks[6].

2). **Dataset CARLA dari Roboflow Universe**

Dataset CARLA yang tersedia di Roboflow Universe adalah kumpulan data yang digunakan untuk melatih dan menguji model deteksi objek, khususnya dalam pengembangan kendaraan otonom. Dataset ini mencakup berbagai gambar yang menggambarkan lingkungan jalan raya atau perkotaan, lengkap dengan anotasi objek seperti kendaraan, pejalan kaki, dan rambu lalu lintas[7]. Dengan menggunakan dataset ini, para peneliti dapat mengimplementasikan model deteksi objek seperti YOLO dalam situasi yang menyerupai kondisi nyata, sehingga memungkinkan pengujian yang lebih realistis di lingkungan simulasi untuk kendaraan otonom.

3). **YOLO (You Only Look Once)**

YOLO adalah metode deteksi dengan akurasi rata-rata (mAP) yang tinggi dan efisiensi tinggi.[8] YOLOv8 adalah iterasi terbaru dari seri YOLO, menampilkan beberapa perbaikan kinerja dan arsitektur dibandingkan dengan iterasi sebelumnya. Arsitektur YOLOv8 menggunakan model jaringan yang lebih terperinci dan teknik augmentasi data yang lebih sedikit untuk meningkatkan akurasi deteksi objek. Keuntungan utama YOLO adalah kemampuannya untuk melakukan deteksi objek secara real time[9], yang sangat penting untuk aplikasi seperti mengemudi otonom. Namun, kelemahan menggunakan YOLOv8 termasuk kompromi antara kecepatan dan akurasi serta persyaratan komputer yang tinggi untuk model pelatihan.

4). **Ukuran Batch dalam Model Pelatihan**

Ukuran batch adalah parameter penting dalam model pelatihan pembelajaran mendalam yang menentukan jumlah sampel yang digunakan untuk melatih model dalam satu iterasi. Ukuran batch yang berbeda dapat mempengaruhi stabilitas, generalisasi, dan konvergensi model selama proses pelatihan. Penelitian menunjukkan bahwa batch yang lebih kecil dapat memberikan perkiraan gradien yang lebih akurat tetapi dengan fluktuasi yang lebih tinggi. Batch yang lebih besar dapat membuat pelatihan lebih stabil tetapi mungkin membutuhkan lebih banyak iterasi untuk mencapai konvergensi[10]. Pengaruh ukuran batch pada kinerja model, terutama dalam skenario identifikasi objek menggunakan YOLO, adalah bidang penelitian yang penting.

5). *Mean Average Precision (mAP)*

Mean Average Precision (mAP) adalah metrik primer yang digunakan untuk mengevaluasi kinerja deteksi objek model. Menghitung rata-rata presisi pada berbagai tingkat mengingat, mAP mengukur akurasi deteksi objek. mAP memberikan informasi yang komprehensif tentang kemampuan model untuk mendeteksi dan mengklasifikasikan objek dengan tepat dalam berbagai kondisi. Penggunaan mAP dalam deteksi objek sangat penting karena memberikan indikator komprehensif kinerja model itu juga mengurangi positif palsu dan deteksi yang terlewatkan selain didasarkan pada deteksi dapat diandalkan[11]

6). **Studi Kasus dan Penelitian terkait**

Beberapa studi telah mengeksplorasi penggunaan YOLO dalam konteks mengemudi otonom, menunjukkan bahwa model ini efektif dalam mendeteksi objek seperti mobil lain, orang yang berjalan, dan objek yang terlalu kecil untuk cocok dengan visi seseorang.[12] Studi lain memeriksa efek ukuran lot pada kinerja model YOLO dan menemukan bahwa ukuran batch ideal tergantung pada konfigurasi dan dataset model[13]

2. **DASAR TEORI**

Model YOLO telah terbukti efektif dalam mengidentifikasi berbagai objek dalam konteks yang menyebabkan rasa sakit. Penelitian sebelumnya menunjukkan bahwa model ini dapat mengurangi deteksi palsu dan lewatkan positif, yang membuatnya cocok untuk aplikasi yang membutuhkan deteksi yang akurat dan cepat[14]. selain itu, penelitian ini menyoroti pentingnya ukuran batch optimal dalam meningkatkan produktivitas model YOLO sesuai dengan dataset dan konfigurasi yang digunakan. studi ini menunjukkan bahwa terlalu kecil dari ukuran batch dapat menyebabkan konvergensi yang buruk dan kinerja yang tidak dapat diandalkan, sementara terlalu besar dari ukuran lot dapat menyebabkan overfitting dan penurunan efisiensi pelatihan[15]. oleh karena itu, memilih ukuran batch yang ideal adalah faktor kunci dalam meningkatkan kinerja model YOLO. dengan mengingat hal ini, pengguna dapat meningkatkan akurasi dan efisiensi deteksi objek dalam aplikasi otonomi. selain itu, tingkat belajar memiliki dampak yang signifikan pada paradigma kerja YOLO. penelitian lebih lanjut juga menunjukkan bahwa penggunaan teknik augmentasi data termasuk rotasi dan modulasi kecerahan dapat meningkatkan akurasi deteksi objek[16]. dengan menggabungkan beberapa strategi ini, pengguna dapat mencapai hasil terbaik ketika menggunakan model YOLO untuk aplikasi kendaraan otonom.

Selain itu, penting untuk mempertimbangkan ukuran dataset yang digunakan dalam pelatihan YOLO. semakin besar dataset yang digunakan, semakin akurat kemampuan model untuk menggambarkan objek. Selain itu, pengguna harus menyadari komposisi dataset untuk memastikan bahwa itu secara akurat mencerminkan keadaan pikiran yang akan terjadi pada model latent. dengan mempertimbangkan semua faktor ini, pengguna dapat memaksimalkan kinerja model YOLO dalam deteksi objek dalam aplikasi otonomi. mengingat hal ini, pengguna harus memastikan bahwa dataset yang mereka gunakan menangkap berbagai kondisi ruang, waktu, dan objek sehingga model dapat belajar dan melakukan dengan baik dalam berbagai skenario[17]. selain itu, pengguna harus melakukan evaluasi menyeluruh dari model untuk memastikan bahwa kinerjanya konstan dan dapat diterapkan pada skenario dunia nyata. akibatnya, implementasi model YOLO dalam aplikasi kendaraan otonom dapat mencapai tingkat pengembalian target dan membantu meningkatkan keamanan dan efisiensi dalam kendaraan[18].

3. **METODOLOGI PENELITIAN**

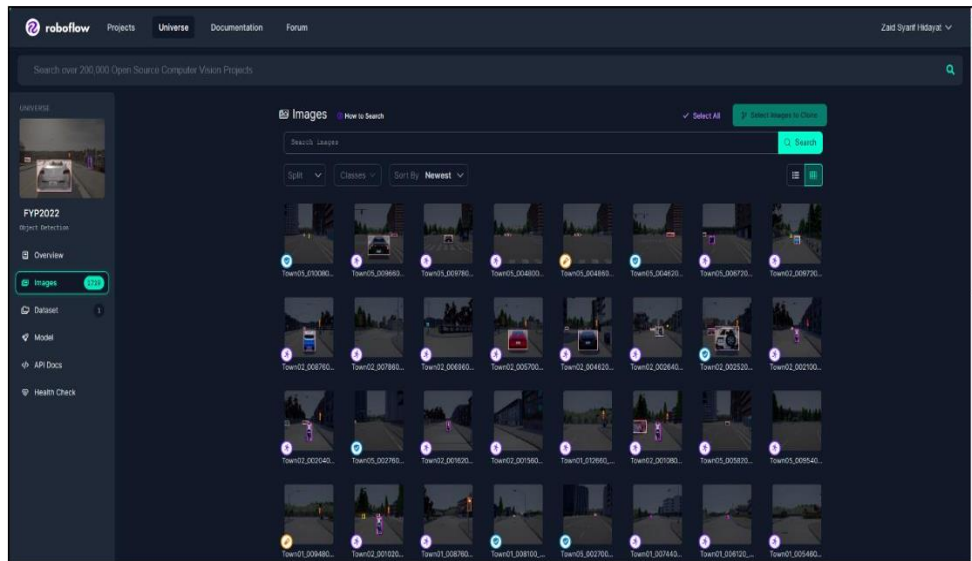
Dalam bidang teknologi yang berkembang pesat saat ini, mengemudi otonom, atau mobil mandiri, adalah salah satu topik yang paling menarik untuk dibahas. Salah satu aspek penting dari mengemudi otonom adalah kemampuannya untuk mendeteksi objek di sekitar kendaraan. Dalam studi ini, penulis akan menganalisis metode YOLOv8 untuk deteksi objek dalam mengemudi otonom[19]. Penulis akan menjelaskan metode yang digunakan dan langkah-langkah yang diambil dalam penelitian ini. Saya berharap hasil penelitian ini akan berguna dalam pengembangan teknologi autonomous otonom lebih lanjut.

Berikut ini adalah deskripsi aktivitas yang dilakukan pada tahapan metode penelitian yang terdapat pada ditulis pada Tabel 1 dibawah ini.

Tabel 1. Tahapan Metode Penelitian

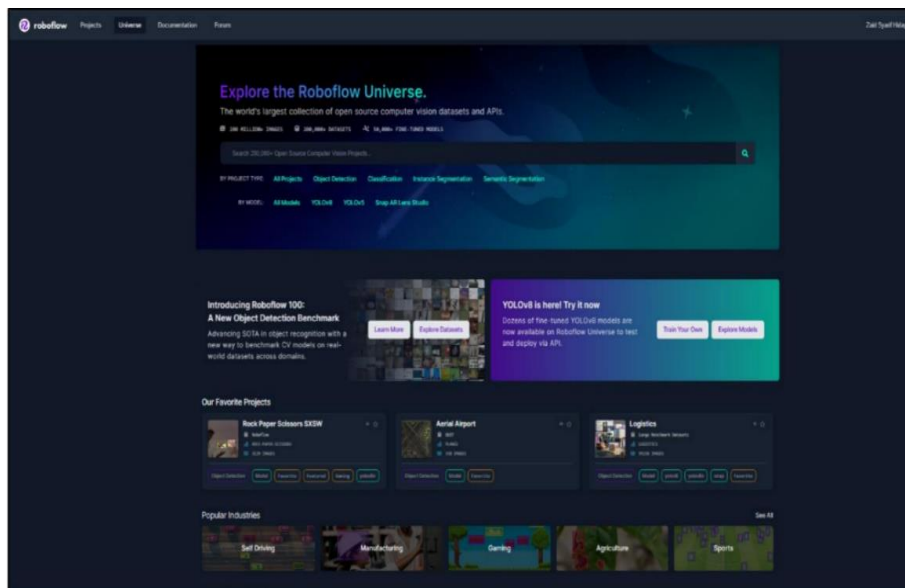
Tahapan	Aktivitas	Deskripsi Aktivitas
1. Preliminary Study	Research Topic	Mencari dan menentukan topik yang akan dilakukan penelitian serta mencari dan menentukan objek yang akan menjadi bahan penelitian. Dan untuk penelitian ini akan menggunakan topik komputasi visual dan pengolahan citra dalam menganalisis performa YOLOv8 dalam melakukan objek deteksi untuk autonomous driving
	Research Problem	Mengidentifikasi masalah penelitian yang terdapat pada penelitian sebelumnya dan berkaitan dengan analisis metode YOLOv8 untuk deteksi objek dalam autonomous driving
2. Literature Review	Identify Technique for Object Detection	Mengeksplorasi, menganalisis, dan mengidentifikasi sebuah literature atau jurnal yang berkaitan dengan objek deteksi.
	Identify Technique for Object Detection	Mengeksplorasi, menganalisis, dan mengidentifikasi sebuah literature atau jurnal yang berkaitan dengan autonomous driving dan YOLO.
3. Data Collection & Analysis Data	Data Source	Melakukan eksplorasi berbagai sumber data sampai menemukan sebuah sumber data yang sesuai dengan penelitian ini. Dan untuk penelitian ini dataset yang digunakan adalah dataset simulasi lalu lintas dari CARLA yang didapatkan dari Roboflow universe.
	Data Collection	Mengumpulkan dataset gambar simulasi lalu lintas dari CARLA yang didapatkan dari Roboflow universe yang akan digunakan pada penelitian ini.
	Data Pre-processing	Mempersiapkan dan mengolah data yang akan digunakan dalam penelitian ini seperti pembersihan data, transformasi data, konsolidasi data, pemilihan variabel, normalisasi dan transformasi, Pembuatan Dataset Pelatihan dan validation.
	Data Analysis	Melakukan analisis data menggunakan metode YOLOv8 dengan mengubah hyperparameter
4. Proposing	Proposed for Analysis Performance of YOLOv8 Object Detection for Autonomous Driving	Mengusulkan sebuah pengaturan hyperparameter yang optimal untuk YOLOv8 dalam melakukan deteksi objek dari hasil analisis performa YOLOv8.
5. Evaluation	Evaluate Analysis Performance of YOLOv8 Object Detection for Autonomous Driving	Mengevaluasi dan mengukur kinerja serta efektivitas model atau metode yang telah diusulkan dalam penelitian menggunakan metric evaluation yaitu mean average precision(mAP), atau menguji sejauh mana model yang diusulkan dapat memenuhi tujuan penelitian dan mengatasi masalah penelitian yang telah dirumuskan.

Data yang digunakan dalam penelitian ini adalah data sekunder. Jenis data yang digunakan adalah data citra skema lalu lintas dari CARLA yang diekstrak dari alam semesta melalui roboflow, seperti yang terlihat di Gambar. 1.



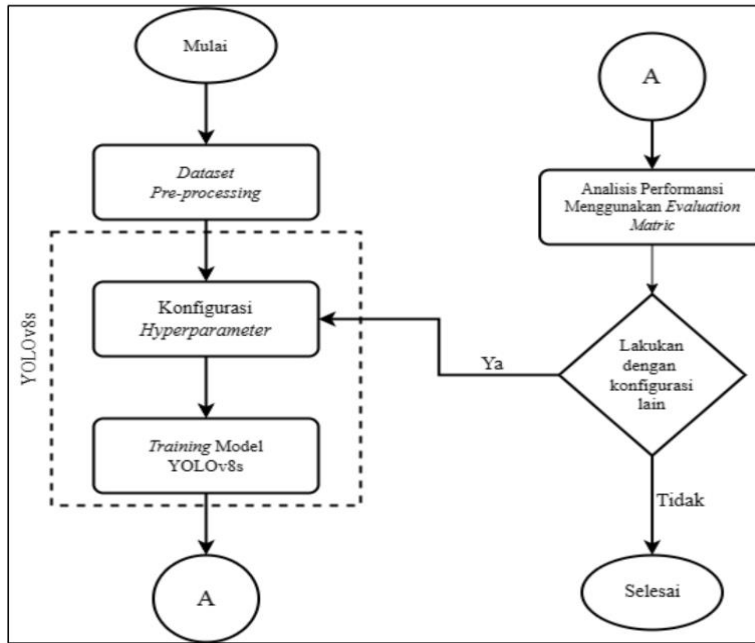
Gambar 1. Sumber dataset yang digunakan

Roboflow universe adalah sebuah open-source repository untuk computer vision dataset dan pre-trained model. Roboflow menampilkan lebih dari 200 juta gambar yang dibagi menjadi 200 ribu kumpulan data, dan 50 ribu lebih model seperti yang terlihat pada Gambar 2.



Gambar 2. *Roboflow universe (Roboflow Universe: Open Source Computer Vision Community)*

Dalam penelitian ini, Teknik Analisis Data digunakan untuk menghasilkan kesimpulan yang akurat tentang hasil penelitian dengan menginterpretasikan data. Pengumpulan data dilakukan secara sekunder dengan mengunduh data dari open-source repository Roboflow universe, memungkinkan penggunaan data yang telah ada untuk analisis. Populasi penelitian terdiri dari 1719 citra skema lalu lintas, dengan 1211 data sampel untuk training, 343 untuk validation, dan 165 untuk testing, memastikan representativitas sampel untuk menghasilkan hasil yang dapat diandalkan dan generalisasi yang tepat. Berikut ini adalah teknik analisis data yang digunakan terdapat pada Gambar 3.



Gambar 3. Teknik analisis Data

4. PENGUJIAN DAN PEMBAHASAN

Pengujian yang diperoleh mencakup penggunaan 1719 citra skema lalu lintas dengan pembagian 1211 data untuk training, 343 data untuk validation, dan 165 data untuk testing . Hasil training dilakukan dengan konfigurasi hyperparameter tertentu dan evaluasi dilakukan berdasarkan rumus mAP . Selain itu, sumber data sekunder dari Roboflow universe digunakan untuk dataset citra skema lalu lintas . Pembahasan meliputi teknik analisis data yang penting untuk menghasilkan kesimpulan yang akurat, teknik pengumpulan data sekunder dari open-source repository Roboflow universe, dan informasi tentang populasi dan sampel yang digunakan dalam penelitian . Dengan demikian, hasil dan pembahasan ini memberikan gambaran menyeluruh tentang metodologi dan temuan dalam analisis performa YOLOv8 dalam deteksi objek autonomous driving.

1). Pengujian

Penggunaan sumber data sekunder dari Roboflow universe, yang merupakan open-source repository untuk dataset dan pre-trained model dalam computer vision[20]. Data citra skema lalu lintas dari CARLA diunduh dari Roboflow universe untuk digunakan dalam penelitian. Total populasi data yang digunakan adalah 1719 citra skema lalu lintas, dengan pembagian 1211 data untuk training, 343 data untuk validation, dan 165 data untuk testing. Teknik pengumpulan data sekunder ini memungkinkan peneliti untuk menggunakan data yang telah ada secara efisien dan dapat diandalkan untuk analisis dalam penelitian[21].

Tabel 2. Daftar kelas yang terdapat pada dataset

No	Kelas	No	Kelas
1	Vehicle	6	Traffic light red
2	Moto bike	7	Traffic sign 30
3	Bike	8	Traffic sign 60
4	Traffic light green	9	Traffic sign 90
5	Traffic light orange	10	Person

Konfigurasi Hyperparameter dan Training data Sebelum melakukan training pada model YOLOv8 maka hyperparameter perlu dilakukan konfigurasi terlebih dahulu pada epoch, batch size, dan image size. Pada penelitian ini terdapat beberapa skema konfigurasi seperti yang terlihat pada Tabel 3 dibawah ini.

Tabel 3. Skema Konfigurasi *Hyperparameter*

Skema	Epoch	Batch Size	Image Size
I	25	16	416 x 416
II	50	16	416 x 416
III	100	16	416 x 416
IV	25	32	416 x 416
V	50	32	416 x 416
VI	100	32	416 x 416
VII	25	64	416 x 416
VIII	50	64	416 x 416
IX	100	64	416 x 416
X	25	16	512 x 512
XI	50	16	512 x 512
XII	100	16	512 x 512
XIII	25	32	512 x 512
XIV	50	32	512 x 512
XV	100	32	512 x 512
XVI	25	64	512 x 512
XVII	50	64	512 x 512
XVIII	100	64	512 x 512
XIX	25	16	640 x 640
XX	50	16	640 x 640
XXI	100	16	640 x 640
XXII	25	32	640 x 640
XXIII	50	32	640 x 640
XXIV	100	32	640 x 640
XXV	25	64	640 x 640
XXVI	50	64	640 x 640
XXVII	100	64	640 x 640

Setelah melakukan konfigurasi hyperparameter maka selanjutnya dapat dilakukan training menggunakan metode YOLOv8 seperti yang terlihat pada tabel 4 dibawah ini.

Tabel 4. Ringkasan hasil training

Skema	Epoch	Batch Size	Image Size (pixel)	mAP50	mAP50-95	Time(minute)
I	25	16	416 x 416	0.945	0.708	13.45
II	50	16	416 x 416	0.964	0.734	22.19
III	100	16	416 x 416	0.965	0.746	43.44
IV	25	32	416 x 416	0.955	0.702	11.17
V	50	32	416 x 416	0.962	0.723	22.46
VI	100	32	416 x 416	0.958	0.748	40.39
VII	25	64	416 x 416	0.955	0.711	11.08
VIII	50	64	416 x 416	0.968	0.734	20.58
IX	100	64	416 x 416	0.964	0.746	36.25
X	25	16	512 x 512	0.968	0.732	15.34
XI	50	16	512 x 512	0.976	0.75	26.51

Skema	Epoch	Batch Size	Image Size (pixel)	mAP50	mAP50-95	Time(minute)
XII	100	16	512 x 512	0.973	0.76	61.03
XIII	25	32	512 x 512	0.965	0.735	14.29
XIV	50	32	512 x 512	0.973	0.745	25.07
XV	100	32	512 x 512	0.976	0.762	52.26
XVI	25	64	512 x 512	0.963	0.732	13.26
XVII	50	64	512 x 512	0.966	0.746	25.12
XVIII	100	64	512 x 512	0.973	0.755	48.28
XIX	25	16	640 x 640	0.974	0.743	20.12
XX	50	16	640 x 640	0.978	0.76	32.56
XXI	100	16	640 x 640	0.975	0.77	64.51
XXII	25	32	640 x 640	0.973	0.745	17.02
XXIII	50	32	640 x 640	0.977	0.761	32.10
XXIV	100	32	640 x 640	0.982	0.768	59.45
XXV	25	64	640 x 640	0.976	0.747	17.07
XXVI	50	64	640 x 640	0.976	0.75	33.20
XXVII	100	64	640 x 640	0.975	0.771	60.33

2). **Pembahasan**

Menganalisis performa YOLOv8 dengan menggunakan dataset citra simulasi lalu lintas dari CARLA simulator yang didapatkan dari Roboflow universe[22]. Pembahasan mencakup analisis performa terbaik dari YOLOv8 melalui perubahan pada jumlah epoch, batch size, dan image size selama training. Hasil evaluasi menunjukkan performa model berdasarkan metrik precision, recall, average precision (AP), dan mean average precision (mAP). Selain itu, confusion matrix juga digunakan untuk mengukur kinerja model dalam mengklasifikasikan data dengan benar.

Metrik Evaluasi:

Precision: Mengukur seberapa akurat prediksi positif model.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

Recall: Mengukur kemampuan model dalam mengidentifikasi semua data positif yang sebenarnya.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

Average Precision (AP) adalah metode untuk menggambarkan nilai presisi dan recall secara keseluruhan dalam satu nilai, mencerminkan rata-rata presisi secara umum atau rata-rata akurasi dari model.

$$AP = \int_0^1 p(r)dr \tag{3}$$

Mean Average Precision (mAP) merupakan nilai rata-rata Average Precision (AP) dari suatu objek pada setiap kelas.

$$mAP = \frac{1}{k} \sum_{i=1}^k AP_i \times 100\% \tag{4}$$

Hasil evaluasi menggunakan rumus pada hasil training dapat dilihat pada Tabel 5 dibawah ini.

Tabel 5. Hasil Evaluasi Berdasarkan Rumus mAP

Skema	Epoch	Batch Size	Image Size (pixel)	mAP50	mAP50-95	Time (minute)
I	25	16	416 x 416	94.5%	70.8%	13.45
II	50	16	416 x 416	96.4%	73.4%	22.19
III	100	16	416 x 416	96.5%	74.6%	43.44
IV	25	32	416 x 416	95.5%	70.2%	11.17
V	50	32	416 x 416	96.2%	72.3%	22.46
VI	100	32	416 x 416	95.8%	74.8%	40.39
VII	25	64	416 x 416	95.5%	71.1%	11.08
VIII	50	64	416 x 416	96.8%	73.4%	20.58
IX	100	64	416 x 416	96.4%	74.6%	36.25
X	25	16	512 x 512	96.8%	73.2%	15.34
XI	50	16	512 x 512	97.6%	75%	26.51
XII	100	16	512 x 512	97.3%	76%	61.03
XIII	25	32	512 x 512	96.5%	73.5%	14.29
XIV	50	32	512 x 512	97.3%	74.5%	25.07
XV	100	32	512 x 512	97.6%	76.2%	52.26
XVI	25	64	512 x 512	96.3%	73.2%	13.26
XVII	50	64	512 x 512	96.6%	74.6%	25.12
XVIII	100	64	512 x 512	97.3%	75.5%	48.28
XIX	25	16	640 x 640	97.4%	74.3%	20.12
XX	50	16	640 x 640	97.8%	76%	32.56
XXI	100	16	640 x 640	97.5%	77%	64.51
XXII	25	32	640 x 640	97.3%	74.5%	17.02
XXIII	50	32	640 x 640	97.7%	76.1%	32.10
XXIV	100	32	640 x 640	98.2%	76.8%	59.45
XXV	25	64	640 x 640	97.6%	74.7%	17.07
XXVI	50	64	640 x 640	97.6%	75%	33.20
XXVII	100	64	640 x 640	97.5%	77.1%	60.33

3). Pengaruh Jumlah Epoch Terhadap Akurasi dan Waktu Pelatihan

Peningkatan jumlah epoch dari 25 ke 50 dan 100 cenderung meningkatkan akurasi model (mAP50 dan mAP50-95), tetapi berdampak pada waktu pelatihan yang lebih lama. pada resolusi 640x640 dengan batch size 32, skema dengan 100 epoch (Skema XXIV) menghasilkan mAP50 tertinggi 98.2% dalam waktu 59.45 menit, sementara skema dengan 50 epoch (Skema XXIII) menghasilkan mAP50 sebesar 97.7% dalam 32.10 menit. Meskipun jumlah epoch yang lebih tinggi meningkatkan akurasi, waktu pelatihan yang lebih lama menjadi pertimbangan penting, terutama dalam konteks aplikasi dunia nyata. Di dunia nyata, seperti pada aplikasi kendaraan otonom atau sistem pengawasan real-time, peningkatan akurasi yang signifikan sering kali diimbangi dengan penurunan kecepatan dalam pelatihan model. oleh karena itu, untuk aplikasi yang membutuhkan deteksi objek secara cepat dan responsif, pemilihan jumlah epoch yang optimal harus disesuaikan dengan kebutuhan aplikasi tersebut.

4). Pengaruh Ukuran Batch Terhadap Akurasi dan Waktu Pelatihan

Batch size yang lebih besar, seperti 64, mempercepat waktu pelatihan tetapi sedikit menurunkan akurasi. Sebagai contoh, dengan resolusi 640x640 dan 50 epoch, skema dengan batch size 16 (Skema XX) menghasilkan mAP50 sebesar 97.8%, sementara batch size 64 (Skema

XXVI) menghasilkan mAP50 sebesar 97.6% dengan waktu pelatihan yang lebih singkat. Dalam konteks dunia nyata, keputusan untuk menggunakan ukuran batch yang lebih besar atau kecil akan sangat bergantung pada batasan sumber daya dan kebutuhan akurasi. Untuk aplikasi real-time yang mengharuskan deteksi cepat, batch size yang lebih besar dapat lebih menguntungkan meskipun sedikit mengorbankan akurasi, karena dapat mengurangi waktu pelatihan secara signifikan.

5). Pengaruh Resolusi Gambar Terhadap Akurasi

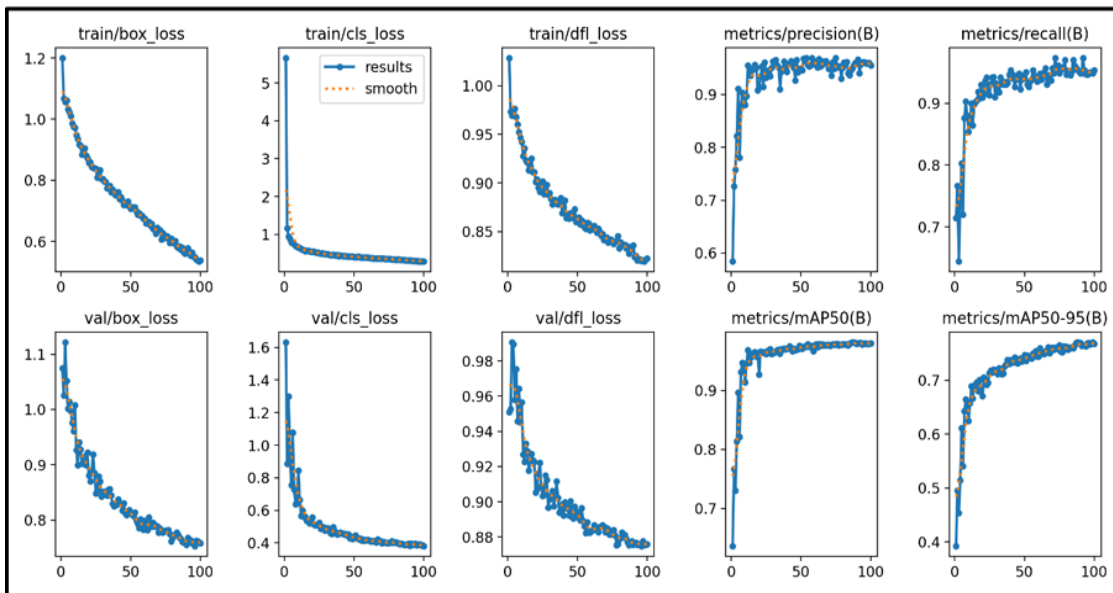
Resolusi gambar yang lebih tinggi, seperti 640x640, menghasilkan akurasi mAP50 yang lebih tinggi dibandingkan dengan resolusi yang lebih rendah, tetapi membutuhkan waktu pelatihan yang lebih panjang. Sebagai contoh, pada 100 epoch dan batch size 32, resolusi 640x640 (Skema XXIV) menghasilkan mAP50 sebesar 98.2%, sedangkan resolusi 416x416 (Skema VI) hanya mencapai 95.8%. dalam aplikasi dunia nyata, resolusi gambar yang lebih tinggi dapat meningkatkan akurasi dalam mendeteksi objek dengan lebih rinci, namun berpotensi memperlambat proses pelatihan. Oleh karena itu, pilihan resolusi harus mempertimbangkan keseimbangan antara akurasi dan kecepatan pelatihan, serta kapasitas perangkat keras yang tersedia untuk menjalankan model.

6). Optimalisasi Hyperparameter untuk mAP Terbaik

Skema dengan 100 epoch, batch size 32, dan resolusi 640x640 (Skema XXIV) menghasilkan akurasi terbaik dengan mAP50 sebesar 98.2%. namun, untuk aplikasi real-time yang mengutamakan efisiensi waktu pelatihan, skema dengan 50 epoch, batch size 32, dan resolusi 640x640 (Skema XXIII) memberikan keseimbangan ideal antara akurasi (97.7%) dan waktu pelatihan (32.10 menit). Dalam dunia nyata, aplikasi yang mengutamakan real-time processing, seperti deteksi objek dalam kendaraan otonom, lebih memilih konfigurasi yang menawarkan waktu pelatihan yang lebih efisien, meskipun dengan sedikit pengorbanan pada akurasi.

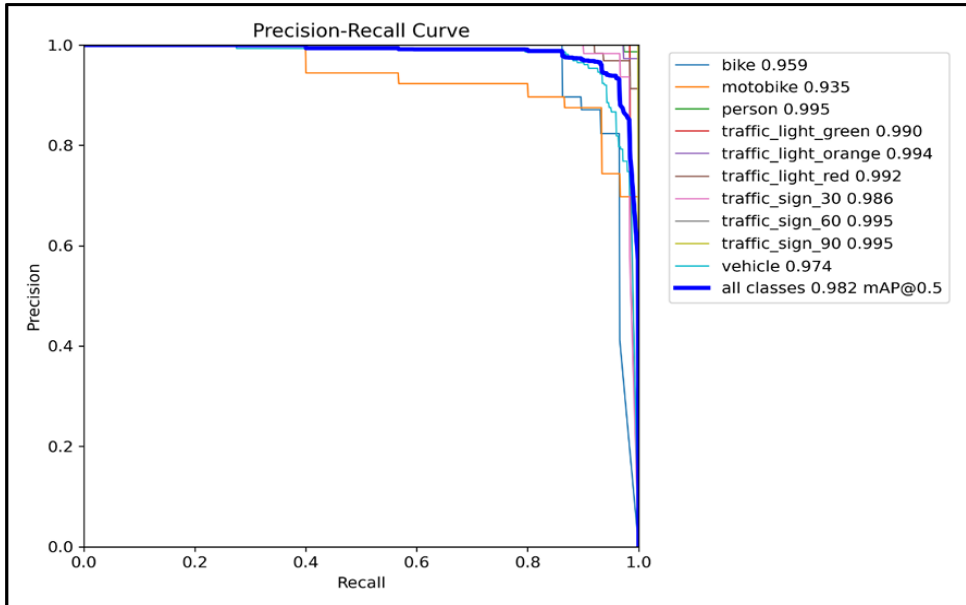
Kombinasi hyperparameter yang optimal sangat dapat disesuaikan pada prioritas aplikasi. Skema XXIV (100 epoch, batch size 32, resolusi 640x640) ideal untuk akurasi maksimal, tetapi untuk aplikasi yang membutuhkan kecepatan pelatihan dan pengolahan data secara real-time, skema XXIII (50 epoch, batch size 32, resolusi 640x640) lebih sesuai. oleh karena itu, pemilihan skema yang tepat harus disesuaikan dengan kebutuhan aplikasi, apakah lebih mengutamakan akurasi atau efisiensi waktu pelatihan.

Berikut ini grafik dari hasil training pada skema XXIV seperti yang terlihat pada Gambar 4 dibawah ini.



Gambar 4. Grafik hasil training

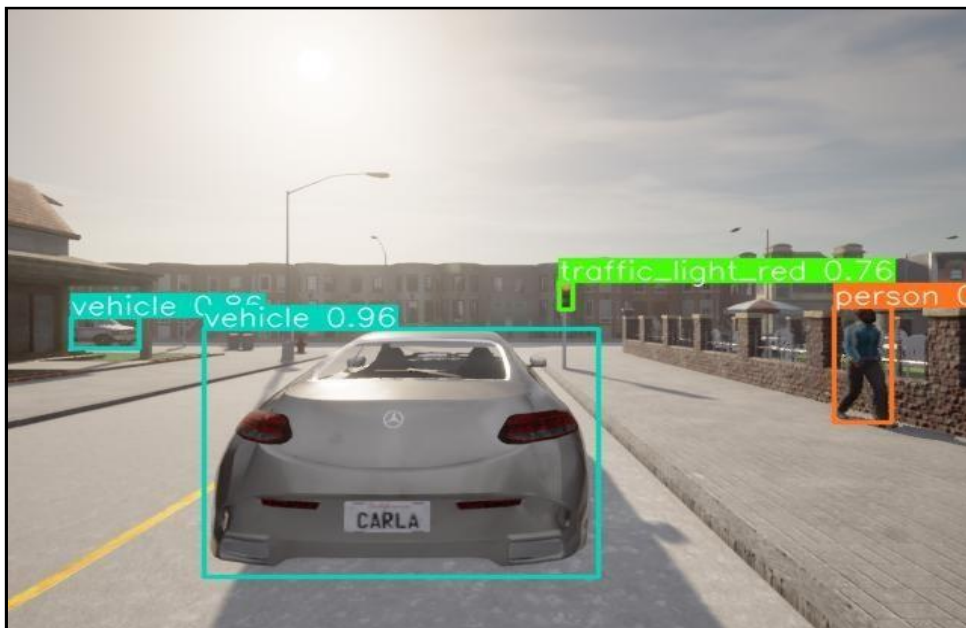
Selain grafik, pada skema XXIV juga terdapat Kurva Precision - Recall seperti yang terlihat pada Gambar 4.64 di bawah ini. terlihat pada Gambar 5 di bawah ini.



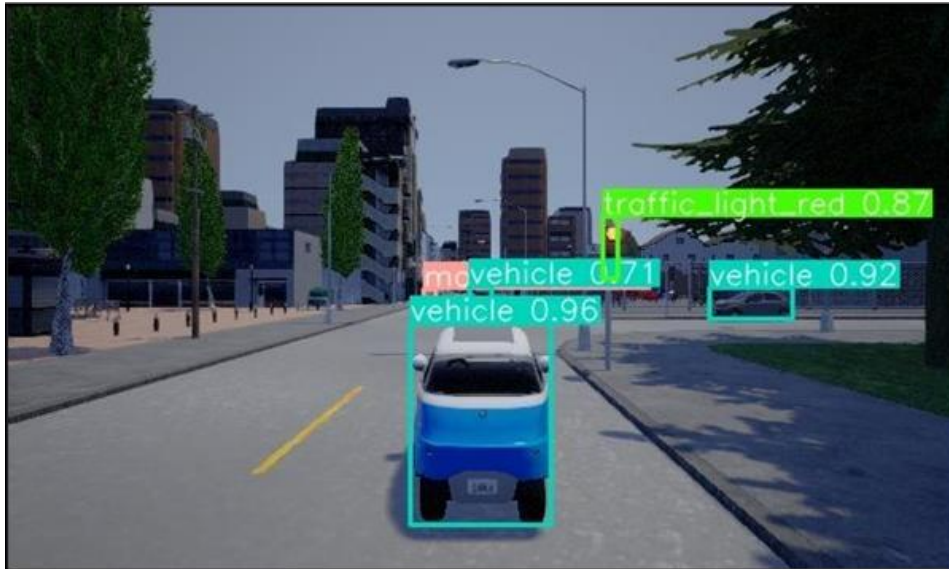
Gambar 4. Grafik *Precision-Recall Curve* skema XXIV

Seperti yang terlihat pada Precision-Recall Curve diatas maka dapat disimpulkan performa model terlihat baik pada setiap kelas. pada setiap kelas mendapatkan nilai rata - rata AP diatas 90% dan untuk semua kelas memiliki nilai mAP sebesar 98.2%.

Pada skema XXIV juga dilakukan testing dengan menggunakan data testing yang berbeda dengan data training. Berikut ini beberapa hasil dari testing menggunakan model skema XXIV. Skema XXIV dipilih karena memiliki nilai mAP yang paling tinggi dari semua skema training. untuk hasil testing dapat dilihat pada Gambar 6 dan 7 di bawah ini.



Gambar 5. Hasil Testing I



Gambar 6. Hasil Testing II

5. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat ditarik beberapa kesimpulan sebagai berikut:

Dataset: Dalam penelitian ini, 1719 gambar yang berisi skema lalu lintas digunakan, yang dibagi menjadi tiga bagian, yaitu 1211 gambar untuk pelatihan (training), 343 gambar untuk validasi, dan 165 gambar untuk pengujian (testing). **Model:** Model YOLOv8 dipilih sebagai model deteksi objek dalam penelitian ini. Beberapa hyperparameter dimodifikasi, termasuk jumlah epoch, ukuran batch, ukuran gambar, serta waktu yang dibutuhkan untuk pelatihan.

Hasil Deteksi: Kinerja model YOLOv8 menunjukkan hasil yang sangat baik, dengan nilai mean Average Precision (mAP) yang beragam tergantung pada skema pelatihan yang diterapkan. **Skema Terbaik:** Dari semua skema yang diuji, skema pelatihan XXIV memberikan hasil paling optimal dengan nilai mAP mencapai 98,2%. Skema ini menggunakan 100 epoch, ukuran batch 32, dan ukuran gambar 640. **Skema Tercepat:** Skema VII tercatat sebagai yang tercepat dalam pelatihan dengan durasi hanya 11,08 menit. Skema ini menghasilkan mAP 95,5% menggunakan 25 epoch, ukuran batch 64, dan ukuran gambar 416.

DAFTAR PUSTAKA

- [1] R. Steven Immanuel Sihombing, W. Abadi Harahap, and W. Kurnia Rahman, "Implementasi Yolo V8 Untuk Mendeteksi Mata Uang Rupiah Emisi Tahun 2022 Ber-Output Audio," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 4, pp. 5900–5905, 2024, doi: 10.36040/jati.v8i4.10099.
- [2] M. Alqarqaz, M. Bani Younes, and R. Qaddoura, "An Object Classification Approach for Autonomous Vehicles Using Machine Learning Techniques," *World Electr. Veh. J.*, vol. 14, no. 2, pp. 1–17, 2023, doi: 10.3390/wevj14020041.
- [3] K. Salma and S. Hidayat, "Deteksi Antusiasme Siswa dengan Algoritma Yolov8 pada Proses Pembelajaran Daring," *J. Indones. Manaj. Inform. dan Komun.*, vol. 5, no. 2, pp. 1611–1618, 2024, doi: 10.35870/jimik.v5i2.716.
- [4] M. S. Hawibowo, "Aplikasi Pengklasifikasi Kematangan Pepaya Menggunakan Metode CNN Berbasis Android," *Fak. Teknol. Ind. Univ. Islam Indones.*, 2024, [Online]. Available: <https://dspace.uii.ac.id/handle/123456789/50071>
- [5] F. A. Hariz, I. N. Yulita, and I. Suryana, "Human Activity Recognition Berdasarkan Tangkapan Webcam Menggunakan Metode Convolutional Neural Network (CNN) Dengan Arsitektur MobileNet," *JITSI J. Ilm. Teknol. Sist. Inf.*, vol. 3, no. 4, pp. 103–115, 2022, doi: 10.30630/jitsi.3.4.97.

- [6] N. Andrade, T. Ribeiro, J. Coelho, G. Lopes, and A. F. Ribeiro, "Combining YOLO and Deep Reinforcement Learning for Autonomous Driving in Public Roadworks Scenarios," *Int. Conf. Agents Artif. Intell.*, vol. 3, no. Icaart, pp. 793–800, 2022, doi: 10.5220/0010913600003116.
- [7] D. Bogdoll, J. Imhof, T. Joseph, and J. M. Zöllner, "Hybrid Video Anomaly Detection for Anomalous Scenarios in Autonomous Driving," *arXiv*, vol. 2, 2024, doi: <https://doi.org/10.48550/arXiv.2406.06423>.
- [8] G. I. Andaru, "Pengembangan Model Deteksi Untuk On-Shelf Availability Produk Menggunakan YOLOv8 Pada Aplikasi Bergerak," *Undergrad. Thesis, progr. Stud. Inform. Fak. Teknol. Ind. Univ. Islam Indones.*, 2024, [Online]. Available: <https://dspace.uui.ac.id/handle/123456789/51069>
- [9] Y. P. Iswoyo, R. Wulanningrum, and A. Bagus, "Identifikasi Jenis Burung Menggunakan Yolo8 Berbasis Web Streamlit," *INOTEK, Univ. Nusant. PGRI Kediri*, vol. 8, pp. 8–15, 2024, doi: <https://doi.org/10.29407/inotek.v8i1.4902>.
- [10] D. Anggara, N. Suarna, and Y. Arie Wijaya, "Performance Comparison Analysis Of Optimizer Adam, SGD, and RMSPROP on The H5 Model," *J. Ilm. NERO*, vol. 8, no. 1, p. 2023, 2023, [Online]. Available: <https://www.kaggle.com/datasets/jonathanoheix/face-expression-recognition-dataset>
- [11] A. D. Lestari, "Aplikasi Deteksi Hama dan Penyakit pada Buah Kakao Menggunakan YOLOv8 Berbasis Website," *Undergrad. Thesis, Progr. Stud. Inform. Fak. Tek. dan Inform. Univ. PGRI Semarang*, 2024, vol. 15, no. 1, pp. 37–48, 2024, [Online]. Available: [https://eprints3.upgris.ac.id/id/eprint/4337/1/Aldhita Dwi Lestari_20670133_Skripsi.pdf](https://eprints3.upgris.ac.id/id/eprint/4337/1/Aldhita%20Dwi%20Lestari_20670133_Skripsi.pdf)
- [12] M. F. Arif, A. Nurkholis, S. Laia, and P. Rosyani, "Deteksi Kendaraan Dengan Metode YOLO," *J. Artif. Intel. dan Sist. Penunjang Keputusan*, vol. 2, no. 1, pp. 20–27, 2023, [Online]. Available: <https://jurnalmahasiswa.com/index.php/aidanspk>
- [13] R. Fajri and F. Fitria, "Pengembangan Real-Time Object Detection System pada Perangkat Single-Board Computer," *Pengemb. Real-Time Object Detect. Syst. pada Perangkat Single-Board Comput.*, vol. 4, no. 2, pp. 1154–1162, 2023, doi: 10.30865/klik.v4i2.1224.
- [14] N. M. HARIS, "Implementasi Algoritma YOLOv7 Pada Sistem Penghitug Benih Kelapa Sawit Secara Realtime," *Dep. Mat. Fak. Mat. dan Ilmu Pengetah. Alam Univ. Hasanuddin*, no. 0, pp. 1–23, 2016.
- [15] J. Tang, C. Ye, X. Zhou, and L. Xu, "YOLO-Fusion and Internet of Things: Advancing object detection in smart transportation," *Alexandria Eng. J.*, vol. 107, no. August, pp. 1–12, 2024, doi: 10.1016/j.aej.2024.09.012.
- [16] S. Wei, Z. Sun, Z. Wang, F. Liao, Z. Li, and H. Mi, "An Efficient Data Augmentation Method for Automatic Modulation Recognition from Low-Data Imbalanced-Class Regime," *Appl. Sci.*, vol. 13, no. 5, 2023, doi: 10.3390/app13053177.
- [17] R. Ramadhan and R. M. Falah, "Pengatur Kepadatan Traffic Light Berbasis Image Recognition," *Undergrad. Proj. Progr. Elektro, Tek. Ind. Fak. Teknol. Indones. Univ. Islam*, no. 20524143, 2024, [Online]. Available: <https://dspace.uui.ac.id/handle/123456789/51616>
- [18] İ. Özcan, Y. Altun, and C. Parlak, "Improving YOLO Detection Performance of Autonomous Vehicles in Adverse Weather Conditions Using Metaheuristic Algorithms," *Appl. Sci.*, vol. 14, no. 13, 2024, doi: 10.3390/app14135841.
- [19] E. U. Armin, A. Purnama Edra, F. I. Alifin, I. Sadidan, I. P. Sary, and U. Latifa, "Performa Model YOLOv8 untuk Deteksi Kondisi Mengantuk pada pengendara mobil," *BRAHMANA J. Penerapan Kecerdasan Buatan*, vol. 5, no. 1, pp. 67–76, 2023, [Online]. Available: <https://doi.org/10.30645/brahmana.v5i1.279>
- [20] M. H. A. Jihaannuriy, "Pengenaln Pelat Nomor Otomatis Menggunakan Metode Inception-ResNet v2 Skripsi Disusun Oleh," *Undergrad. Thesis, Progr. Stud. Mat. Fak. Sains dan Teknol. Univ. Negeri Syarif Hidayatullah.*, 2022, [Online]. Available: <https://repository.uinjkt.ac.id/dspace/handle/123456789/74283>
- [21] A. D. M. HM and S. A. Junianti, "Penerapan Teknologi Blockchain Dalam Sistem Informasi Akuntansi," *Jawara Sist. Inf.*, vol. 1, no. 1, pp. 1–12, 2023, [Online]. Available:

- <https://ejournal.universitاسmandiri.ac.id/index.php/jsi/article/view/21>
- [22] E. W. Hjelle and S. M. Kanstad, “An Approach to First-Level Situation Awareness for Autonomous Surface Vehicles,” *M.S. thesis, Dept. Comput. Sci. Nor. Univ. Sci. Technol. Trondheim, Norw.*, no. June, 2023.