

YOLOv9 – BASED TRAFFIC SIGN DETECTION UNDER VARYING LIGHTING CONDITIONS

Akbar Pangestu^{*1}, Muhammad Pajar Kharisma Putra^{*2}

^{1,2}Informatics, Faculty of Engineering and Computer Science, Universitas Teknokrat Indonesia, Indonesia
Email: ¹akbar_pangestu@teknokrat.ac.id, ²pajarkharisma@teknokrat.ac.id

(Article received: October 09, 2024; Revision: October 21, 2024; published: February 20, 2025)

Abstract

Traffic signs are an important element that functions as a guide, regulator and safety supervisor for road users. In Indonesia, there are various types of traffic signs, including recommendation, prohibition, warning, command, and direction signs, which use numbers, letters, symbols, or a combination of the three to convey clear information to drivers. Based on data from the Indonesian National Police, 148,575 cases of traffic accidents were recorded in 2023, which continues to increase every day due to human error, poor road conditions, and lack of clarity and completeness of signs. This research aims to develop traffic sign detection technology using the YOLOv9 algorithm, starting with collecting 7,980 images from the Roboflow platform, which are then labeled and trained, and evaluated using metrics such as Accuracy, Precision, Recall, F1-Score, and Intersection over Union (IoU). Then the model was tested to detect traffic signs in various media, such as images and videos. The results of this research show that the YOLO v9 model has the best performance compared to SSD MobileNet v2 and Faster RCNN. The YOLOv9 model achieved an accuracy of 94%, while SSD MobileNet v2 only had an accuracy of 43%, and Faster RCNN had an accuracy of 57%. From the research, it can be concluded that the YOLOv9 model is optimal enough to detect traffic signs in various lighting conditions, because the model has the best performance compared to the other two models, especially in terms of accuracy and balance between precision and recall. This research is expected to support the development of safer autonomous vehicles and intelligent transportation systems through optimal traffic sign detection.

Keywords: Computer Vision, Interchange over Union (IoU), Object Detection, Traffic signs, YOLO.

DETEKSI RAMBU LALU LINTAS BERBASIS YOLOv9 DALAM BERBAGAI KONDISI PENCAHAYAAN

Abstrak

Rambu lalu lintas adalah elemen penting yang berfungsi sebagai panduan, pengatur, dan pengawas keselamatan bagi pengguna jalan. Di Indonesia, ada berbagai jenis rambu lalu lintas, termasuk rambu anjuran, larangan, peringatan, perintah, dan petunjuk, yang menggunakan angka, huruf, simbol, atau kombinasi ketiganya untuk menyampaikan informasi yang jelas kepada pengemudi. Berdasarkan data dari Kepolisian Republik Indonesia, tercatat 148.575 kasus kecelakaan lalu lintas pada tahun 2023, yang terus meningkat setiap hari akibat kesalahan manusia, kondisi jalan yang buruk, dan kurangnya kejelasan serta kelengkapan rambu. Penelitian ini bertujuan untuk mengembangkan teknologi deteksi rambu lalu lintas menggunakan algoritma YOLOv9, dimulai dengan pengumpulan 7.980 gambar dari platform Roboflow, yang kemudian dilabeli dan dilatih, serta dievaluasi menggunakan metrik seperti Akurasi, Precision, Recall, F1-Score, dan Intersection over Union (IoU). Kemudian model ditesting untuk mendeteksi rambu lalu lintas dalam berbagai media, seperti gambar dan video. Dari hasil penelitian ini menunjukkan bahwa model YOLO v9 memiliki performa terbaik dibandingkan SSD MobileNet v2 dan Faster RCNN. Model YOLOv9 mencapai akurasi sebesar 94% , sedangkan SSD MobileNet v2 hanya memiliki akurasi sebesar 43%, dan Faster RCNN memiliki akurasi 57%. Pada penelitian dapat disimpulkan bahwa Model YOLOv9 sudah cukup optimal untuk mendeteksi rambu lalu lintas dalam berbagai kondisi pencahayaan, karena model memiliki performa terbaik dibandingkan dua model lainnya, terutama dalam hal akurasi dan keseimbangan antara precision dan recall. Penelitian ini diharapkan dapat mendukung pengembangan kendaraan otonom dan sistem transportasi cerdas yang lebih aman melalui deteksi rambu lalu lintas yang optimal.

Kata kunci: Computer Vision, Deteksi Objek, Intersection over Union (IoU), Rambu Lalu Lintas; YOLO.

1. PENDAHULUAN

Rambu lalu lintas adalah elemen penting yang berfungsi sebagai panduan, pengatur, dan pengawas keselamatan pengguna jalan. Di Indonesia, terdapat ratusan jenis rambu seperti rambu anjuran, larangan, peringatan, perintah, dan petunjuk, yang menggunakan angka, huruf, simbol, atau kombinasi keduanya untuk memberikan informasi yang jelas kepada pengemudi.[1] Berdasarkan data dari Kepolisian Republik Indonesia, tercatat sebanyak 148.575 kasus kecelakaan lalu lintas pada tahun 2023. Angka ini mengalami peningkatan dibandingkan tahun 2022 yang mencatat 139.364 kasus. Kecelakaan pada tahun 2023 juga merupakan yang tertinggi dalam lima tahun terakhir.[2] Permasalahan terkait lalu lintas sering kali menjadi tantangan kompleks dalam pengelolaan kota, dikarenakan pertumbuhan populasi dan peningkatan jumlah kendaraan dengan cepat. Akibat dari situasi ini, angka kecelakaan lalu lintas terus meningkat setiap harinya.[3] Kecelakaan lalu lintas adalah salah satu dari banyak peristiwa lalu lintas yang terjadi setiap hari. Ada banyak faktor yang memengaruhi kecelakaan tersebut, salah satunya adalah kesalahan manusia.[4] Selain kesalahan manusia, faktor lingkungan seperti jalan buruk, cuaca ekstrem, dan rambu tidak memadai juga berkontribusi pada kecelakaan lalu lintas.[5] Kondisi ini menuntut adanya sistem pendeteksian rambu lalu lintas yang akurat dan dapat bekerja dalam berbagai kondisi lingkungan untuk mengurangi kecelakaan.[6]

Seiring dengan berkembangnya teknologi, algoritma deteksi objek berbasis *deep learning* seperti *You Only Look Once (YOLO)* telah terbukti efektif dalam mendeteksi objek secara real-time, bahkan pada lingkungan yang kompleks.[7] Deteksi rambu lalu lintas merupakan komponen penting yang memberikan input visual kepada sistem navigasi kendaraan. Teknologi deteksi objek diperlukan untuk mengidentifikasi rambu dengan variasi bentuk, warna, ukuran, dan posisi di lingkungan kompleks. Peningkatan dalam teknologi ini tidak hanya membantu navigasi, tetapi juga meningkatkan keselamatan dengan memperingatkan pengemudi tentang potensi bahaya.[8]

Salah satu metode deteksi objek yang efektif adalah YOLO (*You Only Look Once*). YOLO merupakan algoritma berbasis *deep learning* yang sangat populer karena kecepatannya yang tinggi dalam mendeteksi objek pada gambar atau video.[9] YOLO menggunakan pendekatan end-to-end yang efisien dengan membagi gambar menjadi grid. Setiap grid memprediksi bounding box dan probabilitas kelas objek di dalamnya, memungkinkan YOLO untuk mendeteksi banyak objek dengan sangat cepat, bahkan dalam lingkungan *real-time*. [10] Salah satu tujuan digunakannya YOLO (*You Only Look Once*) adalah untuk memenuhi kebutuhan akan metode deteksi objek dalam

computer vision yang cepat dan akurat.[11] Penelitian ini berupaya memberikan kontribusi pada pengembangan teknologi deteksi rambu lalu lintas dengan memanfaatkan algoritma YOLOv9 yang terbaru.[12] Penelitian ini diharapkan memberikan wawasan tentang keunggulan dan batasan YOLOv9 dalam deteksi rambu lalu lintas serta optimasinya untuk sistem transportasi cerdas. Temuan ini juga dapat menjadi acuan bagi peningkatan akurasi, pemrosesan di berbagai kondisi lingkungan, dan pengembangan YOLO di versi mendatang.[13]

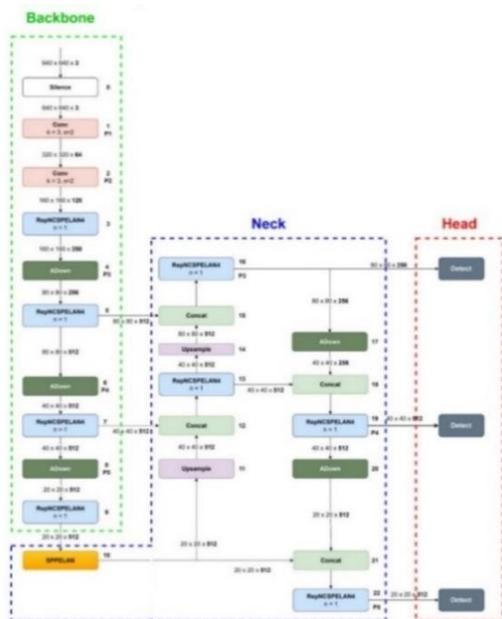
Penerapan algoritma YOLO pada deteksi rambu lalu lintas bukan tanpa tantangan. Skala dan resolusi adalah masalah utama yang dihadapi. Rambu lalu lintas, terutama yang berada jauh dari lokasi pengambilan gambar, sering kali hanya berukuran kecil pada gambar jalanan.[14] Selain itu, akurasi deteksi juga dapat dipengaruhi oleh suara dan perubahan pencahayaan yang signifikan, seperti pada malam hari atau saat cuaca buruk. Dengan meningkatkan mekanisme penanganan objek kecil dan menerapkan strategi augmentasi yang lebih baik, YOLOv9 diharapkan dapat mengatasi sebagian besar masalah ini. Agar hasilnya dapat diandalkan di dunia nyata, implementasi ini harus diuji secara menyeluruh dengan dataset yang beragam dari rambu lalu lintas.[15] Klasifikasi tipe rambu penting karena setiap rambu memiliki makna spesifik yang harus dikenali oleh kendaraan, seperti tanda peringatan, larangan, atau instruksi tertentu.[16] Kesalahan dalam mengklasifikasikan tipe rambu dapat menyebabkan kendaraan mengambil keputusan yang salah, yang berpotensi berbahaya. YOLOv9 diharapkan dapat mengenali tipe-tipe rambu dengan baik menggunakan kombinasi teknik deteksi berbasis bounding box dan klasifikasi menggunakan *deep learning* yang sudah terintegrasi dalam arsitektur jaringannya.[17]

Dalam implementasi, YOLOv9 akan melakukan deteksi dengan menggunakan bounding box untuk menentukan lokasi rambu pada gambar. Kemudian, untuk menentukan jenis rambu yang terdeteksi, hasil tersebut akan dikirim ke modul klasifikasi.[18] Metode ini memungkinkan YOLOv9 untuk mengidentifikasi keberadaan dan konteks serta kategori rambu. Selain itu, model ini dibuat untuk menangani masalah seperti variabel pencahayaan, perubahan sudut pandang, dan perubahan kondisi cuaca, yang semuanya sering memengaruhi kinerja deteksi.[19] YOLOv9 diproyeksikan dapat mencapai kinerja terbaik dalam skenario lalu lintas nyata dengan memanfaatkan keunggulan dalam kecepatan dan akurasi. Ini akan membantu mengembangkan sistem transportasi cerdas yang lebih aman dan andal.[20] Generalisasi model YOLOv9 terhadap berbagai kondisi lingkungan diupayakan untuk ditingkatkan melalui penggabungan teknik augmentasi data dan penggunaan berbagai dataset.[21] Dengan ini, YOLOv9 diharapkan dapat berkontribusi pada pengembangan sistem kendaraan

otonom dan juga dapat digunakan dalam sistem pendukung pengemudi (*Advanced Driver Assistance Systems*) juga dikenal sebagai ADAS, yang membantu orang membuat keputusan yang lebih aman saat berkendara.[22] Kemudian sistem ini dapat mengklasifikasikan rambu lalu lintas dengan benar, yang dapat mengurangi tingkat kecelakaan lalu lintas dan mendukung pembangunan sistem transportasi yang lebih efisien.[23] Penelitian ini berfokus pada pengembangan dan evaluasi YOLOv9 untuk mengatasi keterbatasan versi sebelumnya, seperti YOLOv3 dan YOLOv5, dalam mendeteksi objek pada pencahayaan rendah dan lingkungan dengan gangguan visual. Tujuannya adalah menguji kemampuan YOLOv9 dalam mendeteksi rambu lalu lintas di berbagai kondisi pencahayaan serta membandingkannya dengan SSD MobileNet v2 dan Faster RCNN. Hasilnya diharapkan berkontribusi pada pengembangan sistem transportasi cerdas di Indonesia dan mendukung upaya pengurangan kecelakaan di jalan raya.

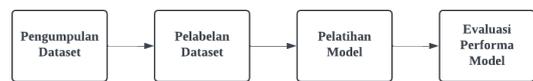
2. METODE PENELITIAN

Penelitian ini bertujuan mengembangkan dan mengevaluasi model deteksi rambu lalu lintas menggunakan algoritma YOLOv9, yang dipilih karena kecepatan dan akurasinya. Algoritma ini unggul dalam skenario lalu lintas di Indonesia karena mampu mendeteksi objek secara real-time dan menangani kondisi lingkungan dengan pencahayaan yang beragam—seperti terik siang hari, redup saat hujan, atau gelap pada malam hari. Dibandingkan dengan model seperti SSD MobileNet v2 dan Faster RCNN, YOLOv9 memiliki arsitektur yang lebih efisien dengan pemrosesan yang cepat tanpa mengurangi akurasi, sehingga ideal untuk sistem transportasi yang membutuhkan deteksi cepat dan tepat.



Gambar 1. Arsitektur YOLOv9[24]

Pada gambar 1 YOLOv9 memiliki arsitektur yang terdiri dari beberapa blok utama: *Backbone*, yang menggunakan Jaringan Agregasi Lapisan Efisien (ELAN) untuk menciptakan representasi fitur yang kaya dengan menyeimbangkan kedalaman dan lebar. *Neck*, yang merupakan *Path Aggregation Network* (PAN) yang meningkatkan aliran informasi dengan menggabungkan fitur dari berbagai lapisan, sehingga meningkatkan ketahanan peta fitur untuk deteksi objek pada berbagai skala. *Head*, yang berfungsi sebagai Kepala Deteksi, memanfaatkan kotak jangkar dan peta fitur multiskala untuk memprediksi kotak pembatas, probabilitas kelas, dan skor objektivitas. Arsitektur ini dirancang untuk mengatasi masalah kehilangan informasi dan meningkatkan efisiensi dalam deteksi objek.[24]



Gambar 2. Alur Penelitian

2.1. Pengumpulan Dataset

Dalam penelitian ini, pengumpulan dataset dilakukan menggunakan 7.980 gambar rambu lalu lintas, yang dikumpulkan melalui platform roboflow. Dataset ini terdiri dari 72 kelas yang masing-masing mewakili jenis rambu lalu lintas yang berbeda seperti rambu peringatan, rambu larangan, dan rambu petunjuk. Setiap kelas dirancang untuk mencerminkan variasi yang ada pada rambu-rambu tersebut, sehingga model yang dikembangkan dapat lebih efektif. Dengan adanya variasi ini, diharapkan sistem yang dibangun dapat mengenali rambu lalu lintas dan mampu beradaptasi dalam berbagai kondisi. Berikut beberapa contoh dari dataset tersebut disajikan pada Gambar 3.





Gambar 3. Dataset Rambu Lalu lintas

2.2. Pelabelan Dataset

Untuk melakukan pelabelan ini, area pada gambar yang berisi rambu lalu lintas ditandai dengan *bounding box* yang mengelilingi objek. Setiap *bounding box* diberi label yang sesuai dengan jenis rambu yang ditampilkan, sehingga model YOLOv9 dapat mengidentifikasi pola-pola dari setiap rambu. Karena proses ini harus dilakukan dengan teliti, kemampuan model untuk melakukan deteksi dan pengenalan yang ditentukan oleh akurasi anotasi. Oleh karena itu, untuk mendukung proses pelatihan yang efektif, setiap gambar harus dilabeli dengan tepat dan konsisten. Dalam prosedur ini, *bounding box* digunakan untuk menandai area yang memiliki rambu lalu lintas. Setiap box diberi label berdasarkan jenis rambu. Karena keakuratan anotasi memengaruhi kemampuan model YOLOv9 untuk mendeteksi dan mengenali objek dengan tepat, keakuratan ini sangat penting. Model dapat belajar dengan baik dan bekerja lebih baik dalam situasi dunia nyata dengan anotasi yang tepat dan konsisten. Oleh karena itu, setiap gambar harus dilabeli dengan benar, dan kualitas anotasi harus dipantau untuk memastikan dataset tetap relevan dan andal. Pada gambar 4 menunjukkan image dataset yang telah dilabeli secara otomatis menggunakan platform Roboflow.



Gambar 4. Pelabelan Dataset

2.3. Pelatihan Model

Penelitian ini mengembangkan model deteksi rambu lalu lintas dengan algoritma YOLOv9. Proses pelatihan melibatkan anotasi data yang detail dan teknik augmentasi gambar. Pengujian dilakukan dengan dataset uji yang belum pernah diolah untuk mengukur performa menggunakan metrik seperti Akurasi, Presisi, Recall, F-1 Score, dan Intersection over Union (IoU). YOLOv9 (*You Only Look Once version 9*) adalah model deteksi objek terbaru yang menawarkan peningkatan signifikan dalam akurasi dan kecepatan dibandingkan versi sebelumnya. Dengan arsitektur *Convolutional Neural Network (CNN)* yang lebih efisien dan integrasi elemen transformer, YOLOv9 mampu memproses hingga 60 frame per detik (FPS) pada resolusi tinggi, dengan akurasi *mAP (mean Average Precision)* lebih dari 50% pada benchmark *COCO*. Model YOLOv9 sangat

cocok untuk aplikasi seperti pengawasan keamanan, kendaraan otonom, dan analisis video, menjadikannya pilihan utama dalam teknologi deteksi objek modern.

Pelatihan model YOLOv9 dilakukan melalui platform Kaggle yang memanfaatkan GPU yang tersedia, yang meningkatkan efisiensi dan mempercepat proses pelatihan. Untuk memastikan kualitas data, dataset rambu lalu lintas diunggah dan diproses. Sebelum diunggah dengan Kaggle, kita dapat menggunakan library dan alat seperti Roboflow untuk otomatisasi *preprocessing*. Pada Tabel 1. Parameter pelatihan YOLOv9 mencakup pengaturan seperti ukuran batch, laju pembelajaran, jumlah epoch, dan teknik augmentasi data untuk mengoptimalkan performa deteksi objek.

Tabel 1. Parameter Pelatihan YOLOv9

| No | Parameter | Nilai |
|----|------------|-------|
| 1 | Epoch | 100 |
| 2 | Batch Size | 16 |
| 3 | Image Size | 640 |

Melalui grafik *loss* dan akurasi, Kaggle menawarkan monitoring langsung yang memungkinkan pengguna menyesuaikan parameter. Model dilatih untuk mengenali rambu lalu lintas dengan akurasi tinggi menggunakan teknik normalisasi dan augmentasi data. Evaluasi dilakukan untuk menilai kinerja model pada dataset validasi. Ini melibatkan penggunaan metrik seperti metrik seperti *Accuracy*, *Precision*, *Recall*, *F-1 Score* dan *Intersection over Union (IoU)* untuk menilai efektivitas deteksi.

2.4. Evaluasi Performa YOLOv9

Evaluasi model YOLOv9 sangat penting untuk memastikan bahwa model yang telah dilatih dapat mendeteksi rambu lalu lintas dengan baik. Proses ini dilakukan dengan menguji model pada dataset validasi yang belum pernah digunakan sebelumnya untuk menilai kinerja deteksi dalam kondisi nyata. Beberapa metrik utama digunakan untuk mengevaluasi kinerja model:

- Accuracy* adalah metrik evaluasi yang menggambarkan proporsi prediksi yang benar terhadap total prediksi yang dibuat. Meskipun akurasi mudah dipahami, metrik ini kurang efektif untuk dataset dengan distribusi kelas yang tidak seimbang. Dalam penelitian, akurasi digunakan sebagai salah satu indikator kinerja dasar dalam mengukur performa model klasifikasi.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{1}$$

- Precision* adalah metrik yang mengukur ketepatan model dalam menghasilkan prediksi positif. Nilai precision diperoleh dari perbandingan antara jumlah prediksi positif yang benar (*True Positives*) dengan total

prediksi positif yang mencakup *True Positives* dan *False Positives*. Dengan kata lain, precision menunjukkan seberapa banyak prediksi positif yang benar di antara seluruh prediksi positif yang dihasilkan oleh model.

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

- c. *Recall* mengukur kemampuan model dalam mendeteksi semua instance positif dalam dataset. Nilai recall diperoleh dari rasio antara jumlah prediksi positif yang benar (*True Positives*) dan total instance positif, termasuk *True Positives* dan *False Negatives*. Jadi, recall menunjukkan seberapa baik model dalam mengidentifikasi semua instance positif yang ada.

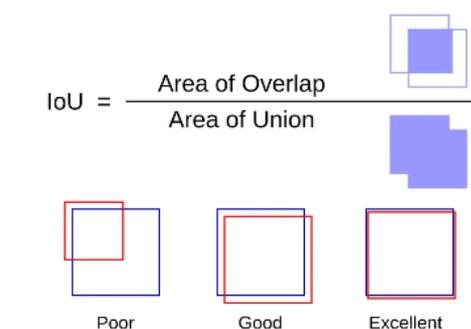
$$Recall = \frac{TP}{TP+FN} \quad (3)$$

- d. *F1-Score* berguna dalam penelitian yang melibatkan prediksi kelas dengan distribusi tidak merata, seperti deteksi anomali. Ini memberikan keseimbangan antara tingkat prediksi benar (*Precision*) dan cakupan deteksi positif (*Recall*). *F1-Score* sering digunakan dalam penelitian untuk memastikan bahwa model tidak hanya akurat, tetapi juga efisien menangkap seluruh kelas.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

- e. *Intersection over Union (IoU)* adalah metrik yang mengukur kesamaan dan perbedaan antara dua set sampel, dengan tujuan mengevaluasi area tumpang tindih antara dua bounding box, yaitu bounding box hasil prediksi dan bounding box ground truth. Gambar 5 menampilkan diagram yang menggambarkan konsep *Intersection over Union (IoU)*, menunjukkan area tumpang tindih antara bounding box hasil prediksi dan ground truth. Visualisasi ini membantu menilai akurasi deteksi objek melalui perbandingan nilai *IoU*.

$$IoU = \frac{A \cap B}{A \cup B} \quad (5)$$



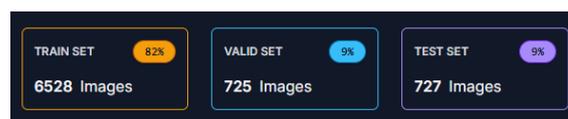
Gambar 5. Diagram yang mewakili *Intersection over Union (IoU)*[25]

IoU antara dua kotak pembatas dihitung sebagai rasio area perpotongan dengan area gabungan. Daerah perpotongan adalah daerah yang bertumpang tindih antara dua kotak pembatas, sedangkan luas gabungan adalah luas gabungan kedua kotak pembatas. Untuk menghitung *IoU* antara dua bounding box, kita harus menentukan koordinat area perpotongan dan area gabungan. Koordinat area perpotongan diperoleh dari nilai minimum dan maksimum koordinat x dan y kedua bounding box. Sementara itu, area gabungan adalah luas yang dibatasi oleh nilai minimum dan maksimum koordinat x dan y dari kedua bounding box tersebut..[25].

3. HASIL DAN PEMBAHASAN

3.1. Hasil Pelabelan Dataset

Images rambu lalu lintas yang telah dilabeli akan menerima *file.txt* yang berisi informasi tentang label dan koordinat *bounding box* dalam format YOLO. Format ini mencakup label kelas, koordinat posisi, dan ukuran *bounding box* yang disesuaikan dengan ukuran gambar. Setelah melakukan anotasi atau *labeling*, langkah selanjutnya seperti yang ditunjukkan pada gambar 6. adalah membagi 7980 image dataset menjadi tiga bagian: *Training Set* 6528 (82%), *Validation Set* 725 (9%), dan *Test Set* 727 (9%) untuk pelatihan model YOLOv9. *Training set* mengajarkan model mengenali berbagai rambu lalu lintas, *Validation Set* membantu mengatur parameter dan menghindari *overfitting*, dan *Testing set* berfungsi sebagai penilaian akhir untuk menilai seberapa efektif model menerapkan pengetahuan yang telah dipelajari dalam situasi baru.



Gambar 6. Pembagian Dataset

3.2. Hasil Pelatihan Model

Berdasarkan hasil deteksi rambu lalu lintas menggunakan model YOLOv9, secara keseluruhan model menunjukkan performa yang optimal dengan rata-rata *precision*, *recall*, dan *mAP50* yang tinggi, yaitu masing-masing 0.836, 0.853, dan 0.88. Beberapa kelas seperti *Barrier Ahead*, *Dangerous Dip*, *Horn Prohibited*, dan *Speed Limit* (rentang -100-hingga -160-) memiliki akurasi hampir sempurna dengan nilai *precision*, *recall*, dan *mAP50* mendekati 1, menandakan kemampuan model untuk mendeteksi kelas-kelas ini. Namun, terdapat beberapa kelas dengan performa rendah seperti *Eating Place*, *Speed Limit -8-*, dan *Speed Limit -65-* yang menunjukkan *precision* dan *recall* jauh lebih rendah. Meskipun demikian, performa konsisten yang ditunjukkan pada berbagai kelas rambu kecepatan dan rambu utama lainnya menunjukkan efektivitas model dalam

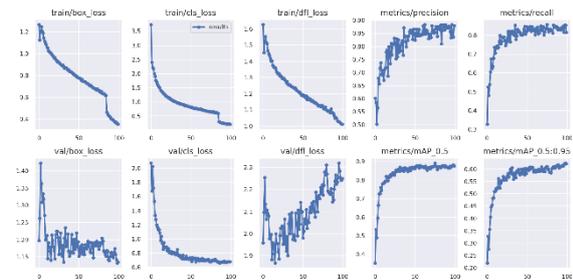
mendeteksi rambu lalu lintas dengan baik seperti yang ditunjukkan pada Tabel 2 dibawah ini.

Tabel 2. Confusion Matrix

| Class | Precision | Recall | mAP50 |
|------------------------------------|-----------|--------|-------|
| Barrier Ahead | 0.966 | 1 | 0.995 |
| Cattle | 0.851 | 1 | 0.995 |
| Caution | 0.588 | 1 | 0.995 |
| Cycle Crossing | 0.822 | 1 | 0.935 |
| Dangerous Dip | 1 | 0.902 | 0.995 |
| Eating Place | 0.435 | 0.444 | 0.544 |
| Falling Rocks | 0.907 | 1 | 0.995 |
| Ferry | 0.733 | 0.5 | 0.58 |
| First Aid Post | 0.731 | 0.923 | 0.881 |
| Give Way | 0.903 | 0.933 | 0.961 |
| Horn Prohibited | 0.901 | 1 | 0.995 |
| Hospital | 0.839 | 0.444 | 0.745 |
| Hump | 0.87 | 1 | 0.969 |
| Left Hair Pin Bend | 0.825 | 1 | 0.995 |
| Left Reverse Bend | 0.584 | 0.667 | 0.611 |
| Left hand curve | 1 | 0.81 | 0.964 |
| Light Refreshment | 0.843 | 0.615 | 0.781 |
| Men at Work | 0.97 | 1 | 0.995 |
| Narrow Bridge | 0.944 | 0.875 | 0.904 |
| Narrow road ahead | 0.921 | 1 | 0.995 |
| No Parking | 0.686 | 0.727 | 0.765 |
| No Stopping | 0.967 | 1 | 0.995 |
| No Thorough Road | 1 | 0.93 | 0.995 |
| No Thorough SideRoad | 0.736 | 1 | 0.995 |
| Parking Lot Cars | 0.693 | 0.5 | 0.677 |
| Parking Lot Cycle | 0.727 | 1 | 0.962 |
| Parking Lot Scooter and MotorCycle | 0.606 | 0.529 | 0.681 |
| Parking This side | 0.837 | 0.667 | 0.685 |
| Pedestrian Crossing | 0.912 | 0.786 | 0.947 |
| Pedestrian Prohibited | 0.924 | 1 | 0.995 |
| Petrol Pump- Gas Station | 0.91 | 0.941 | 0.946 |
| Public Telephone | 0.909 | 0.812 | 0.907 |
| Resting Place | 0.852 | 0.8 | 0.923 |
| Right Hair Pin Bend | 0.894 | 0.848 | 0.873 |
| Right Hand Curve | 0.562 | 1 | 0.995 |
| Right Reverse Bend | 0.784 | 0.933 | 0.907 |
| Road Wideness Ahead | 0.827 | 0.846 | 0.923 |
| Round About | 1 | 0.842 | 0.995 |
| School Ahead | 0.87 | 0.671 | 0.839 |
| Slippery Road | 0.912 | 0.7 | 0.927 |
| Speed Limit -10- | 0.927 | 1 | 0.995 |
| Speed Limit -100- | 0.954 | 1 | 0.995 |
| Speed Limit -110- | 0.931 | 1 | 0.995 |
| Speed Limit -120- | 0.948 | 1 | 0.995 |
| Speed Limit -130- | 0.938 | 1 | 0.995 |
| Speed Limit -140- | 1 | 0.971 | 0.995 |
| Speed Limit -150- | 0.942 | 1 | 0.995 |
| Speed Limit -160- | 1 | 1 | 0.995 |
| Speed Limit -20- | 0.938 | 0.954 | 0.988 |
| Speed Limit -25- | 0.947 | 0.9 | 0.922 |
| Speed Limit -35- | 0.944 | 1 | 0.995 |
| Speed Limit -45- | 0.885 | 1 | 0.859 |
| Speed Limit -5- | 0.793 | 0.812 | 0.715 |
| Speed Limit -50- | 0.951 | 0.941 | 0.959 |
| Speed Limit -55- | 0.932 | 1 | 0.995 |
| Speed Limit -60- | 0.817 | 0.856 | 0.874 |
| Speed Limit -65- | 0.464 | 0.667 | 0.457 |
| Speed Limit -70- | 0.899 | 0.941 | 0.975 |
| Speed Limit -75- | 0.935 | 1 | 0.995 |
| Speed Limit -8- | 0.285 | 0.333 | 0.229 |
| Speed Limit -80- | 0.824 | 0.905 | 0.855 |
| Speed Limit -90- | 0.88 | 0.917 | 0.914 |
| Speed Limit 3 | 0.721 | 1 | 0.995 |
| Speed Limit 30 | 0.93 | 0.942 | 0.905 |
| Speed limit -15- | 0.689 | 0.75 | 0.579 |
| Speed limit -40- | 1 | 0.98 | 0.995 |
| Steep Ascent | 0.946 | 0.643 | 0.773 |
| Steep Descent | 0.743 | 0.965 | 0.713 |
| Stop | 0.927 | 0.767 | 0.603 |

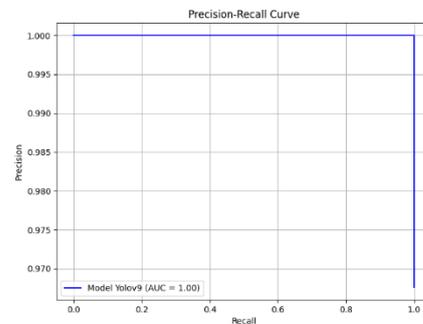
| | | | | | |
|----------------|------------|----|--------------|--------------|-------------|
| Straight Entry | Prohibitor | No | 1 | 0.682 | 0.945 |
| All | | | 0.836 | 0.853 | 0.88 |

Pada Gambar 7 menampilkan plot *training* dan *validation loss*, yang digunakan untuk memantau performa model selama proses pelatihan. Grafik ini menunjukkan bagaimana *loss* berkurang seiring bertambahnya epoch, yang mengindikasikan seberapa baik model mempelajari pola data dan apakah terjadi *overfitting* atau *underfitting*.



Gambar 7. Training dan Validation loss plots

Pada Gambar 8 menunjukan Grafik Precision-Recall Curve menunjukkan bahwa YOLOv9 memiliki performa sempurna dengan AUC =1.00, mempertahankan precision mendekati 1.0 di sepanjang rentang recall. Ini menandakan model mampu mendeteksi objek dengan sangat akurat dan konsisten, hampir tanpa kesalahan prediksi positif, dengan sedikit penurunan precision hanya pada recall tertinggi.



Gambar 8. Grafik Precision-Recall Curve

Berdasarkan Tabel 3 yang menunjukkan hasil kinerja model pada kondisi terang dan kondisi gelap, model deteksi rambu lalu lintas mampu memberikan prediksi dengan benar di kondisi terang pada kelas *Right Hand Curve-0.47*, *No Parking-0.97*, dan *Narrow Bridge-0.95*, ditunjukkan dengan kotak prediksi yang tepat sesuai dengan kelas rambu dan keterangan "*Benar*". Demikian juga pada kondisi gelap model deteksi rambu lalu lintas mampu memberikan prediksi dengan benar di kondisi terang pada kelas *Right Hand Curve-0.38*, *No Parking-0.94*, dan *Narrow Bridge-0.95* Meskipun gambar memiliki pencahayaan yang rendah, model masih mampu mendeteksi ketiga rambu tersebut dengan akurasi yang stabil. Hal ini menunjukkan bahwa model cukup kokoh terhadap variasi pencahayaan, baik pada

gambar terang maupun gelap. Namun, tingkat kepercayaan (*confidence score*) pada kondisi gelap tampak sedikit lebih rendah dibandingkan kondisi terang, yang menandakan potensi penurunan akurasi dalam deteksi di lingkungan dengan pencahayaan terbatas.

Tabel 3. Hasil Kinerja Model

| Class | Kondisi | Prediksi | Keterangan |
|------------------|---|---|------------|
| Right Hand Curve |  |  | Benar |
| No Parking |  |  | Benar |
| Narrow Bridge |  |  | Benar |
| Right Hand Curve |  |  | Benar |
| No Parking |  |  | Benar |
| Narrow Bridge |  |  | Benar |

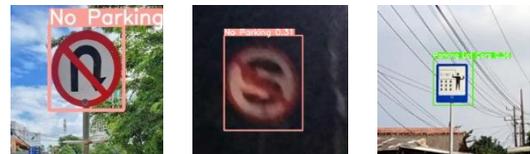
3.3. Hasil Evaluasi Performa YOLOv9

Gambar 9 yang menunjukkan hasil deteksi rambu lalu lintas menggunakan video dapat disimpulkan bahwa model mampu mengenali berbagai jenis rambu dengan *tingkat kepercayaan* (*confidence score*) yang bervariasi. Pada kondisi pencahayaan normal, model dapat mendeteksi dengan optimal rambu *School Ahead*, *Narrow Bridge*, *Left Hand Curve*, dan *Speed Limit 40 km/h* dengan *confidence score* yang cukup tinggi (0.73–0.80), menunjukkan kinerja yang cukup optimal pada lingkungan terang. Namun, pada rambu *No Parking* di kondisi yang lebih gelap atau minim cahaya, nilai *confidence*-nya lebih rendah. Seperti terlihat pada deteksi *Pedestrian Crossing* dengan nilai *confidence* hanya 0.50 walaupun dengan kondisi pencahayaan terang. Secara keseluruhan, model masih cukup andal untuk mendeteksi rambu lalu lintas di berbagai kondisi.



Gambar 9. Deteksi Rambu Lalu lintas menggunakan video

Pada gambar 10, terlihat kesalahan deteksi pada rambu lalu lintas. Hal ini menunjukkan bahwa model memiliki kesulitan dalam membedakan rambu-rambu dengan simbol yang memiliki kemiripan, terutama pada rambu dengan tanda larangan berbentuk lingkaran merah. Kesalahan ini dapat disebabkan oleh kemiripan visual antar simbol pada kelas-kelas yang memiliki karakteristik mirip. Akurasi deteksi yang tidak tepat seperti ini dapat menimbulkan kesalahpahaman dalam memahami informasi dari rambu tersebut, sehingga perlu dilakukan peningkatan pada model.



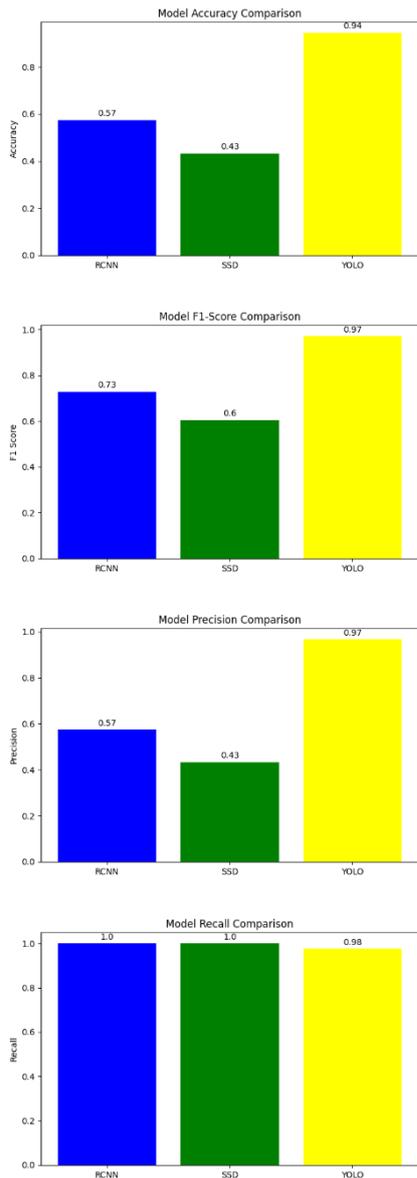
Gambar 10. Deteksi Error.

3.4. Eksperimen Perbandingan

Tujuan dari eksperimen ini adalah untuk mengevaluasi kinerja masing-masing model untuk menentukan model yang paling efektif dalam deteksi objek. Dilakukan eksperimen dengan membandingkan model YOLO dengan algoritma lain yang ada di Tabel 4, seperti *Faster RCNN* dan *SSD MobileNet v2*. Ketiga model ini diuji pada dataset yang sama untuk mendapatkan hasil yang lebih relevan. Pertama digunakan *Faster RCNN*, yang dikenal memiliki akurasi tinggi tetapi memerlukan waktu untuk inferensi. Meskipun *Faster RCNN* memiliki hasil yang baik dalam mendeteksi objek dengan detail yang tinggi, kecepatan prosesnya menjadi kendala dalam aplikasi real-time. Selanjutnya, digunakan *SSD MobileNet v2*, yang menyeimbangkan kecepatan dan akurasi, dan hasilnya menunjukkan bahwa *SSD MobileNet v2* memiliki kemampuan mendeteksi objek dalam waktu.

Tabel.4 Hasil Perbandingan Metrik Kinerja Model

| No | Models | Steps | Accuracy (%) | Precision (%) | Recall (%) | F-1 Score (%) |
|----|------------------|-------|--------------|---------------|------------|---------------|
| 1 | YOLOv9 | 100 | 0.94 | 0.97 | 0.98 | 0.97 |
| 2 | SSD MobileNet v2 | 5000 | 0.43 | 0.93 | 1.0 | 0.6 |
| 3 | Faster RCNN | 5000 | 0.57 | 0.57 | 1.0 | 0.73 |



Gambar 11. Diagram Perbandingan Metrik Kinerja Model

Dalam evaluasi kinerja model deteksi objek, *Intersection over Union* (IoU) digunakan sebagai metrik utama. Proses ini diawali dengan menetapkan *bounding box* berdasarkan *ground truth*, yang berfungsi sebagai referensi untuk lokasi objek dalam gambar. Selanjutnya, model yang telah dilatih digunakan untuk memprediksi *bounding box*, yang mencakup identifikasi dan penempatan kotak di sekitar objek yang terdeteksi. Setelah dihasilkan dua

bounding box, satu dari *ground truth* dan satu dari hasil prediksi model, tahap berikutnya adalah menghitung nilai IoU. Perhitungan IoU dilakukan dengan membandingkan area perpotongan antara *kotak prediksi* dan *ground truth*, kemudian membaginya dengan total area gabungan dari kedua kotak tersebut. Nilai ini menunjukkan sejauh mana model dapat mendeteksi objek. Seperti pada gambar 12 berikut yang merupakan hasil nilai IoU.



Gambar 12. Menentukan Nilai IoU

Menentukan batas IoU menggunakan *threshold* dengan nilai 0.8. *Threshold* ini dipilih karena mampu mencapai keseimbangan antara mendeteksi objek yang tepat dan mengurangi terjadinya deteksi palsu. Dengan menetapkan ambang ini, prediksi diklasifikasikan sebagai benar positif (*TP*) jika nilai IoU mencapai 0.8 atau lebih. Pendekatan ini memastikan bahwa model yang digunakan memiliki akurasi yang cukup dalam mendeteksi objek tanpa mengorbankan kecepatan. Dalam proses evaluasi, *threshold IoU* berfungsi sebagai parameter penting untuk menilai kinerja maksimal model dalam aplikasi tertentu. Jika nilai IoU di bawah 0,8, prediksi tersebut akan dianggap sebagai false positive (FP) atau false negative (FN).

4. DISKUSI

Berdasarkan hasil penelitian yang dilakukan, menunjukkan bahwa YOLOv9 lebih handal dalam mendeteksi rambu lalu lintas, khususnya dalam kondisi pencahayaan yang beragam, sehingga sangat potensial untuk diaplikasikan dalam sistem transportasi cerdas dan otonom. YOLOv9 unggul dalam kecepatan real-time hingga 60 FPS, cocok untuk aplikasi seperti kendaraan otonom dan ADAS, dengan akurasi tinggi (precision 0,97, recall 0,98) pada pencahayaan normal. Arsitekturnya yang end-to-end membuatnya lebih efisien dibanding Faster RCNN. Namun, performanya menurun pada pencahayaan rendah, dengan confidence score hanya 0,50 untuk beberapa objek, dan mengalami kesulitan mendeteksi objek berukuran kecil atau jauh dari kamera.

Untuk meningkatkan performa YOLOv9 pada pencahayaan rendah, dapat diterapkan strategi seperti augmentasi data (*brightness adjustment*, *gamma correction*), transfer learning dengan dataset besar seperti COCO, dan penggunaan sensor tambahan seperti inframerah atau LiDAR. Penyesuaian *threshold IoU* juga membantu deteksi lebih akurat, sementara arsitektur hibrid dengan fitur lokal seperti SIFT atau HOG meningkatkan akurasi untuk objek kecil atau samar.

Jika dibandingkan dengan penelitian terdahulu, YOLOv9 menunjukkan peningkatan yang signifikan dalam hal kecepatan dan akurasi. Misalnya, penelitian yang menggunakan YOLOv3 atau YOLOv5 pada deteksi rambu lalu lintas menunjukkan akurasi yang lebih rendah, terutama dalam kondisi pencahayaan yang buruk.[12] Hal ini membuktikan bahwa perbaikan dalam arsitektur YOLO khususnya pada versi 9, berkontribusi pada peningkatan performa yang signifikan dalam mendeteksi objek di lingkungan yang lebih kompleks.

Penelitian ini menegaskan pentingnya penerapan algoritma deteksi objek yang cepat dan akurat, seperti YOLOv9, dalam konteks dunia nyata. Hasilnya memiliki dampak signifikan terhadap perkembangan sistem transportasi masa depan, terutama dalam mendukung teknologi kendaraan otonom dan sistem bantuan pengemudi. Dengan kemampuan YOLOv9 mendeteksi rambu lalu lintas secara akurat, sistem transportasi dapat menjadi lebih aman dan efisien.

5. KESIMPULAN

Dari hasil penelitian ini menunjukkan bahwa model YOLOv9 memiliki performa tertinggi dibandingkan dengan *SSD MobileNet v2* dan *Faster RCNN*. Model YOLOv9 mencapai akurasi sebesar 94% , sedangkan *SSD MobileNet v2* hanya memiliki akurasi sebesar 43%. Hal serupa terjadi pada *Faster RCNN*, yang memiliki akurasi 57%. Kemampuan model ini untuk mendeteksi rambu lalu lintas dalam berbagai media, seperti gambar dan video, menunjukkan potensi aplikasinya dalam situasi dunia nyata yang kompleks. Evaluasi yang dilakukan dengan menggunakan Intersection over Union (IoU) dan ambang batas 0,8 menjamin keseimbangan antara deteksi objek yang relevan serta mengurangi deteksi palsu, sehingga dapat mempertahankan tingkat presisi dan recall yang tinggi.

Pada penelitian dapat disimpulkan bahwa Model YOLOv9 sudah cukup optimal untuk mendeteksi rambu lalu lintas dalam berbagai kondisi pencahayaan, karena model memiliki performa tertinggi dibandingkan dua model lainnya, terutama dalam hal akurasi dan keseimbangan antara precision dan recall. Selain itu, performa YOLOv9 juga tetap konsisten meskipun pada kondisi pencahayaan rendah, menunjukkan bahwa model ini mampu mendeteksi rambu lalu lintas dengan baik dalam berbagai kondisi pencahayaan.

Dianjurkan untuk menerapkan YOLOv9 pada sistem transportasi cerdas guna mengurangi risiko kecelakaan di persimpangan dan area padat. Penelitian lebih lanjut bisa difokuskan pada integrasi YOLOv9 dengan sistem ADAS untuk kendaraan otonom.

DAFTAR PUSTAKA

[1] M. Akbar, A. S. Purnomo, and S. Supatman,

“Multi-Scale Convolutional Networks untuk Pengenalan Rambu Lalu Lintas di Indonesia,” *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 11, no. 3, pp. 310–315, 2022, doi: 10.32736/sisfokom.v11i3.1452.

- [2] Ajeng Dwita Ayuningtyas, “2023 Capai Angka Tertinggi Kecelakaan Lalu Lintas 5 Tahun Terakhir,” GoodStats v2.3.2. [Online]. Available: <https://goodstats.id/article/2023-capai-angka-tertinggi-kecelakaan-lalu-lintas-5-tahun-terakhir-z67bk>
- [3] 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,” *Sci. Electro*, no. 1, pp. 1–9, 2023, [Online]. Available: <https://jim.unisma.ac.id/index.php/jte/article/viewFile/21551/16069>
- [4] R. Pauziah and A. Herliana, “Implementasi Deteksi Rambu Lalu Lintas Menggunakan Metode Extreme Learning Machine,” *E-Prosiding Sist. Inf.*, vol. x, no. x, pp. 36–41, 2021.
- [5] R. A. Kusuma, Y. Sholva, and R. D. Nyoto, “39255-75676626433-2-Pb,” vol. 08, no. 3, pp. 230–238, 2020, doi: 10.26418/justin.v8i1.39255.
- [6] N. C. Kuncoro, S. A. Wibowo, and K. Usman, “Analisis Kinerja Prototipe Traffic Sign Recognition Untuk Sistem Autonomous Car Menggunakan You Only Look Once Performance Analysis of Prototyping Traffic Sign Recognition for Autonomous Car System By Using You Only Look Once,” vol. 7, no. 3, pp. 8872–8878, 2020.
- [7] M. Iqbal and R. A. Saputra, “Pengenalan Rambu Lalu Lintas Menggunakan Metode Yolov8,” *JIKA (Jurnal Inform. Univ. Muhammadiyah Tangerang)*, vol. 8, no. 2, pp. 204–212, 2024.
- [8] M. Zulfikri, K. Abd Latif, R. Hammad, M. Syahrir, and P. Studi, “Deteksi dan Estimasi Kecepatan Kendaraan dalam Sistem Pengawasan Lalu Lintas Menggunakan Pengolahan Citra Detection and Estimation of Vehicle Speed in Traffic Control Systems Using Image Processing,” *Agustus*, vol. 20, no. 3, pp. 455–467, 2021.
- [9] Y. Lai, “Traffic Sign Recognition Based on Deep Learning Technique,” *ACM Int. Conf. Proceeding Ser.*, no. July 2021, pp. 62–68, 2022, doi: 10.1145/3546157.3546167.
- [10] P. Jiang, D. Ergu, F. Liu, Y. Cai, and B. Ma, “A Review of Yolo Algorithm Developments,” *Procedia Comput. Sci.*, vol. 199, pp. 1066–1073, 2021, doi: 10.1016/j.procs.2022.01.135.

- [11] D. Nafis Alfarizi, R. Agung Pangestu, D. Aditya, M. Adi Setiawan, and P. Rosyani, "Penggunaan Metode YOLO Pada Deteksi Objek: Sebuah Tinjauan Literatur Sistematis," *J. Artif. Intel. dan Sist. Penunjang Keputusan*, vol. 1, no. 1, pp. 54–63, 2023, [Online]. Available: <https://jurnalmahasiswa.com/index.php/aidanpspk>
- [12] P. Saputra, B.M., Ilman, M.Z., Audina, M., Jepri, M., & Rosyani, "Sistem Pengenalan Tanda Lalu Lintas Menggunakan Algoritma YOLO," *JURIHUM J. Inov. dan Hum.*, vol. Volume 1, no. 1, pp. 161–164, 2023.
- [13] M. Yaseen, "WHAT IS YOLOV9: AN IN-DEPTH EXPLORATION OF THE INTERNAL FEATURES OF THE NEXT-GENERATION OBJECT DETECTOR," vol. 9, 2024.
- [14] M. Li, L. Zhang, L. Li, and W. Song, "Yolo-Based Traffic Sign Recognition Algorithm," *Comput. Intell. Neurosci.*, vol. 2022, 2022, doi: 10.1155/2022/2682921.
- [15] S. Qu, X. Yang, H. Zhou, and Y. Xie, "Improved YOLOv5-based for small traffic sign detection under complex weather," *Sci. Rep.*, vol. 13, no. 1, pp. 1–12, 2023, doi: 10.1038/s41598-023-42753-3.
- [16] Y. Sinha, B. Hadassa, G. Krishna, and C. V. Ravi Kumar, "Traffic sign recognition using convolutional neural networks," *Int. J. Electr. Eng. Technol.*, vol. 11, no. 3, pp. 210–217, 2020, doi: 10.14710/jtsiskom.2021.13959.
- [17] Barokah Asmarahman Takarob, "Deteksidan Identifikasi Rambu-Rambu Lalu Lintasberbasis Algoritma You Only Look Once (YOLO)," *Univ. Muhammadiyah Jakarta*, pp. 1–46, 2022, [Online]. Available: <http://repository.umj.ac.id/11987/>
- [18] A. A. G. Bagus Janapriya, "Pengenalan Jenis Rambu Lalu Lintas menggunakan Metode YOLO V5," *JELIKU (Jurnal Elektron. Ilmu Komput. Udayana)*, vol. 11, no. 4, p. 1011, 2023, doi: 10.24843/jlk.2023.v11.i04.p32.
- [19] A. W. Sudjana and H. Supeno, "Implementasi Deep Learning untuk Object Detection Menggunakan Algoritma YOLO (You Only Look Once) pada Rambu Lalu Lintas di Indonesia," *Univ. Pas.*, pp. 1–8, 2021.
- [20] A. S. Kusuma, A. I. Pradana, and B. W. Pamekas, "3201-Naskah Publikasi-14728-1-10-20240725," vol. 7, pp. 166–179, 2024.
- [21] I. Septiana, F. Rizki, A. Niken, A. Cahya, and A. F. Syifa, "Deteksi Orang dengan Menggunakan Algoritma YOLOv3," vol. 2, no. 1, pp. 54–57, 2024.
- [22] M. Fauzan Ridho, Fransiskus Panca, Welly Yandi, and Almeera Amsana Rachmani, "Drowsiness Detection in the Advanced Driver-Assistance System using YOLO V5 Detection Model," *ELECTRON J. Ilm. Tek. Elektro*, vol. 5, no. 1, pp. 130–137, 2024, doi: 10.33019/electron.v5i1.136.
- [23] Muhammad Agus Syaputra, Josua Pinem, Afiq Alghazali Lubis, and Yuva Denia, "Implementasi Algoritma YOLO Dalam Pengklasifikasian Objek Transportasi pada Lalu Lintas Kota Medan," *Pop. J. Penelit. Mhs.*, vol. 3, no. 1, pp. 13–23, 2023, doi: 10.58192/populer.v3i1.1641.
- [24] Y. Fadhillah *et al.*, "ADVANCING VEHICLE DETECTION WITH YOLOV9: INTEGRATING PROGRAMMABLE GRADIENT INFORMATION," vol. 5, no. 2, pp. 90–100, 2024.
- [25] P. Hallaj, "What is Intersection over Union (IoU) in Object Detection?," Medium.Com. [Online]. Available: <https://medium.com/@pouyahallaj/what-is-intersection-over-union-iou-in-object-detection-648d147ed5eb#:~:text=The IoU formula is relatively straightforward%3A IoU %3D,represents the combined area of both bounding boxes.>