

## Deteksi Rambu Lalu Lintas *Real-Time* di Indonesia dengan Penerapan YOLOv11: Solusi Untuk Keamanan Berkendara

Afu Ichsan Pradana<sup>1\*</sup>, Harsanto<sup>2</sup>, Wijiyanto<sup>3</sup>  
<sup>1,2,3</sup>Universitas Duta Bangsa Surakarta, Indonesia

\*Korespondensi Penulis: afu\_ichsan@udb.ac.id

---

### Info Artikel

Dikirim: 31 Oktober 2024  
Diterima: 24 November 2024  
Diterbitkan: 30 November 2024

### Kata kunci:

YOLOv11;  
*Real-time*;  
Indonesia;  
Rambu-rambu;  
Lalu lintas.

---

### ABSTRAK

Penelitian ini bertujuan untuk merumuskan dan menilai kerangka deteksi rambu lalu lintas real-time dalam konteks di negara Indonesia, menggunakan YOLOv11. Mengingat sifat heterogen rambu-rambu lalu lintas dan kondisi jalan raya di Indonesia, ada kebutuhan mendesak untuk model yang tangguh dan tepat guna meningkatkan keselamatan berkendara. Metode yang digunakan adalah YOLOv11, sebuah algoritma deep learning yang dirancang untuk deteksi objek dengan kecepatan tinggi dan akurasi tinggi. Algoritma ini memproses gambar dalam satu langkah inferensi dan memiliki arsitektur yang dioptimalkan untuk pengenalan real-time. Proses meliputi preprocessing data seperti augmentasi dan anotasi, pelatihan model menggunakan dataset rambu lalu lintas Indonesia, serta evaluasi performa model menggunakan metrik precision, recall, dan mean average precision (mAP). Temuan menunjukkan bahwa model berhasil mencapai mAP 0,99, dengan akurasi tinggi di berbagai klasifikasi rambu lalu lintas. Evaluasi menggunakan *Confusion Matrix* menunjukkan tingkat kesalahan yang dapat diabaikan, menandakan keandalan model untuk aplikasi dunia nyata. Aplikasi potensial teknologi ini sangat penting dalam memperkuat sistem keselamatan berkendara dan transportasi cerdas di Indonesia.

---

## 1. PENDAHULUAN

Meningkatnya jumlah kendaraan di jalan-jalan Indonesia telah menyebabkan peningkatan kecelakaan lalu lintas, sehingga sangat penting untuk meningkatkan langkah-langkah keselamatan jalan. Menurut data dari Kementerian Perhubungan, angka kecelakaan lalu lintas di Indonesia terus meningkat setiap tahunnya, dengan ratusan ribu kecelakaan yang melibatkan kendaraan pribadi maupun angkutan umum[1]. Pada tahun 2023, terdapat lebih dari 116.000 kasus kecelakaan lalu lintas, meningkat 6,8% dibandingkan tahun sebelumnya. Dari angka ini, sebanyak 18.357 korban meninggal dunia, sementara korban luka ringan dan luka berat masing-masing mencapai 134.811 dan 11.689 kasus. Mayoritas korban kecelakaan merupakan individu dalam usia produktif, yaitu antara 25–40 tahun[2]. Penyebab kecelakaan sering dikaitkan dengan kegagalan pengemudi untuk mengenali rambu-rambu lalu lintas pada waktu yang tepat dan ketidakmampuan sistem yang ada untuk memberikan peringatan *real-time*[3]. Dalam konteksnya, rambu lalu lintas berfungsi untuk memberi peringatan kepada pengemudi mengenai potensi bahaya. Rambu-rambu ini menyampaikan informasi semantik yang jelas, seperti batas kecepatan, arah, peringatan bahaya, dan lain sebagainya[4]. Dengan demikian, ada kebutuhan mendesak untuk solusi inovatif yang memanfaatkan teknologi canggih, seperti visi komputer dan pembelajaran mesin[5], untuk meningkatkan keselamatan pengguna jalan dan memastikan pengemudi segera diberitahu tentang rambu-rambu lalu lintas dan maknanya[6]. Beberapa penelitian telah mengeksplorasi berbagai metode untuk deteksi rambu lalu lintas seperti algoritma pembelajaran mendalam[7], teknik pemrosesan gambar[8],

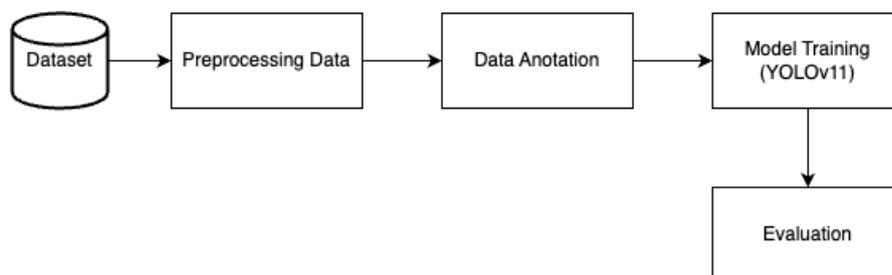
dan jaringan saraf tiruan[9], yang telah menunjukkan hasil yang menjanjikan dalam mengidentifikasi rambu-rambu lalu lintas secara akurat dalam berbagai kondisi. Berbagai macam algoritma telah dikembangkan untuk meningkatkan ketahanan sistem ini, memungkinkan analisis *real-time*[10] dan respons terhadap lingkungan mengemudi yang dinamis. Kemajuan ini tidak hanya bertujuan untuk meningkatkan kesadaran pengemudi tetapi juga berkontribusi pada pengurangan kecelakaan lalu lintas secara keseluruhan, mendorong lingkungan mengemudi yang lebih aman bagi semua pengguna jalan.

Dalam dekade terakhir, berbagai metode telah digunakan untuk pengenalan rambu lalu lintas, termasuk metode Machine Learning seperti Adaboost, Support Vector Machine (SVM), dan Hough Transform[11], yang mana menggunakan warna, tekstur, tepi, dan fitur tingkat rendah lainnya untuk mendeteksi area atau tepi rambu lalu lintas dalam sebuah gambar. Selain menggunakan metode *Machine Learning*, saat ini metode deep learning seperti *Convolutional Neural Network(CNN)* dan *You Look Only Once(YOLO)* lebih mendominasi dalam hal pengenalan rambu-rambu lalu lintas[12]. YOLO (*You Only Look Once*) memiliki keunggulan signifikan dibandingkan metode deteksi objek berbasis CNN tradisional[13]. Salah satu kelebihanannya adalah kecepatan inferensi, di mana YOLO dapat mendeteksi objek hanya dalam satu kali proses[14]., sementara CNN memerlukan beberapa langkah, seperti *region proposal* dan klasifikasi[15]. Algoritma YOLO telah mengalami perkembangan dari versi 1[16] hingga versi 11[17], setiap perkembangannya membawa peningkatan kecepatan dan presisi. Dalam sistem pengenalan rambu lalu lintas dataset yang digunakan berupa citra gambar rambu lalu lintas. Masing-masing negara menggunakan bentuk dan warna yang berbeda untuk identifikasi rambu lalu lintas dengan mempertimbangkan peraturan, iklim dan faktor lain seperti lingkungan[18]. Selain itu, penelitian terkait deteksi rambu lalu lintas dengan menggunakan YOLO masih tergolong baru, terutama dalam konteks di negara Indonesia[18].

Penelitian ini tidak hanya akan memberikan kontribusi pada pengembangan teknologi deteksi objek di Indonesia, tetapi juga akan mendorong pemanfaatan kecerdasan buatan dalam meningkatkan keselamatan publik. Mengingat urgensi dan potensi manfaat yang dihadirkan oleh teknologi ini, penelitian tentang deteksi rambu lalu lintas berbasis YOLOv11 di Indonesia diharapkan dapat menjadi dasar bagi pengembangan sistem yang lebih canggih, misalnya dengan integrasi ke dalam sistem navigasi[19] atau kendaraan otonom di masa depan[20]. Integrasi ini dapat secara signifikan meningkatkan proses pengambilan keputusan *real-time*, yang pada akhirnya mengarah ke jalan yang lebih aman dan sistem manajemen lalu lintas yang lebih efisien.

## 2. METODE PENELITIAN

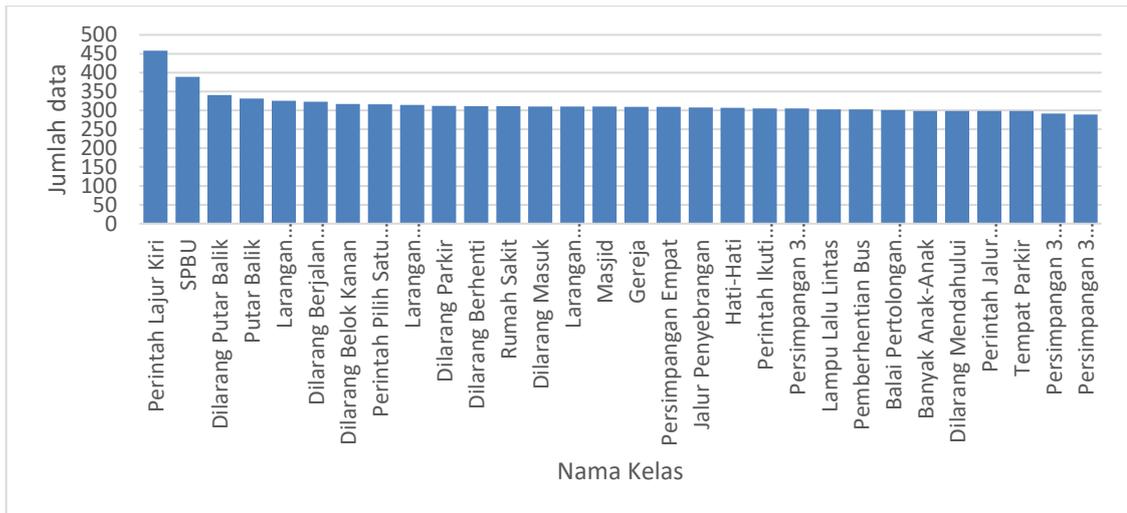
Metode yang digunakan dalam penelitian ini menggunakan pendekatan deep learning yaitu YOLOv11 karena pada versi ini memiliki keunggulan yang signifikan pada penerapan secara *real-time*[17]. Dalam penelitian ini menggunakan bahasa pemrograman Python 3 serta spesifikasi perangkat komputasi yaitu menggunakan pemrosesan berbasis GPU Nvidia Tesla T4 dengan kapasitas 15 GB dan RAM sebesar 13 GB. Penelitian ini merupakan penelitian experimental dengan tahapan Pada Gambar 1. berikut :



Gambar 1. Desain Penelitian

### 1) Dataset

*Dataset* yang digunakan dalam penelitian ini yaitu dataset *Traffic Sign In Indonesia*[21]. Pada *dataset* ini berisi data gambar rambu-rambu lalu lintas di Indonesia dengan total data 9157 gambar, dan berisi 30 kelas. Pada Gambar 2. berikut merupakan distribusi dari jumlah data setiap kelas.



Gambar 2. Distribusi data setiap kelas

## 2) *Preprocessing Data*

Sebelum memasukkan gambar ke dalam model YOLOv11, data harus melalui tahap preprocessing untuk memastikan kualitas input. Pada tahapan ini dilakukan:

### a. *Resizing*

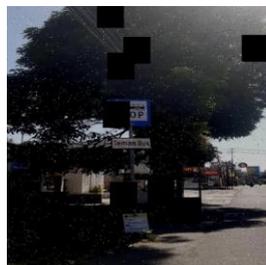
Pada tahapan ini yaitu merubah ukuran gambar input menjadi 640 x 640 pixel.

### b. *Augmentasi data*

Pada tahapan ini melakukan pengayaan dataset dengan berbagai kondisi yaitu dengan merotasi gambar  $15^{\circ}$  dengan tujuan pada saat kondisi jalan yang tidak rata kendaraan masih bisa mendeteksi rambu seperti pada Gambar 3. berikut.

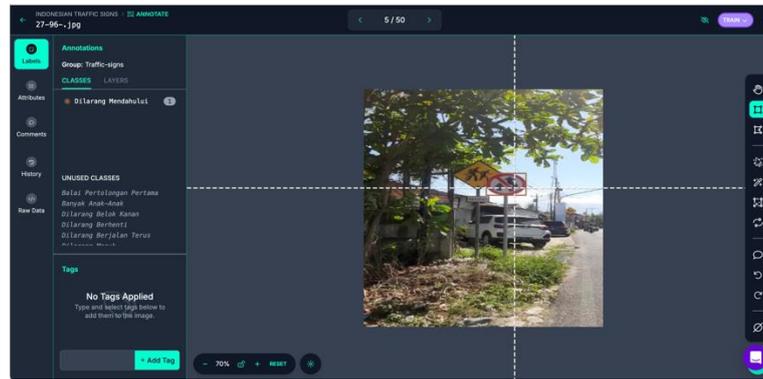
Gambar 3. Rotasi  $15^{\circ}$ 

Kemudian setelah dilakukan rotasi langkah berikutnya adalah dengan menambah efek *Cutout* dengan tujuan saat rambu tertutupi rambu masih terdeteksi seperti pada Gambar 4. berikut.

Gambar 4. *Cutout*

## 3) *Data Anotation*

Pada tahapan ini setiap rambu lalu lintas dalam dataset perlu diberi label yang sesuai untuk membantu model mengenali objek yang berbeda dari setiap kelas. Dalam konteks ini digunakan alat pelabelan yaitu *Roboflow Annotate* seperti pada Gambar 5. berikut.



Gambar 5. Data Anotation

#### 4) Model Training

Pada tahap ini, dataset yang telah dilabeli dimasukkan ke dalam model YOLOv11 untuk pelatihan. Model deteksi YOLO dapat dirumuskan sebagai berikut:

$$Pr(object) * IOU_{pred}^{truth} \quad (1)$$

di mana  $IOU$  adalah irisan atas gabungan antara kotak yang diprediksi dan kebenaran dasar.  $IOU$  adalah angka antara 0 dan 1, nilai yang mendekati 1 berarti kotak pembatas yang diprediksi lebih dekat ke kebenaran dasar. Pada irisan yang sama, bersama dengan kotak pembatas, setiap sel grid juga menentukan probabilitas kelas kondisional ( $C$ ). Probabilitas khusus kelas untuk setiap sel grid didefinisikan sebagai berikut[22] :

$$Pr(class_i|object) * Pr(object) * IOU_{pred}^{truth} = Pr(class_i) * IOU_{pred}^{truth} \quad (2)$$

Skor menunjukkan probabilitas kelas akan muncul di kotak dan seberapa baik kotak yang diprediksi sesuai dengan objek. Output akhir atau prediksi dikodekan sebagai tensor  $S \times S \times (B * 5 + C)$ . Sebelum dilakukan pelatihan maka dataset dilakukan pemisahan (*split*) data terlebih dahulu sesuai pada Tabel 1. berikut.

Tabel 1. Pemisahan data

Pemisahan ( <i>Split</i> )	Prosentase
<i>Training set</i>	80%
<i>Validation set</i>	15%
<i>Testing set</i>	5%

Kemudian tahapan berikutnya adalah pemilihan bobot YOLOv11 yang digunakan yaitu YOLOv11s dengan tuning hyperparameter sebagai berikut:

- Batch size = 32  
Pemilihan batch size didasarkan pada kebutuhan untuk mencapai keseimbangan antara penggunaan memori dan kecepatan pelatihan
- Learning rate = 0,0001  
Nilai learning rate 0,0001 dipilih setelah eksperimen dengan berbagai nilai, dan ditemukan bahwa nilai ini memberikan konvergensi yang stabil tanpa menyebabkan overfitting atau kehilangan informasi penting selama pembelajaran.
- Dropout = 0,15  
Nilai dropout sebesar 0,15 dipilih untuk memastikan bahwa model tidak terlalu mengandalkan fitur tertentu dan tetap belajar dengan baik tanpa berisiko overfitting.
- Epoch = 20 epoch  
Pelatihan dilakukan selama 20 epoch, yang dianggap cukup untuk mencapai konvergensi dengan hasil yang optimal. Setiap epoch mewakili satu siklus lengkap pelatihan melalui seluruh dataset. Pemilihan 20 epoch dipilih berdasarkan eksperimen sebelumnya, di mana model menunjukkan peningkatan yang signifikan pada akurasi

dan mAP pada epoch-epoch pertama dan mulai mencapai konvergensi stabil setelah epoch ke-20.

Sehingga menghasilkan jumlah layer layer = 319, parameter = 9,439,402, gradien = 9,439,386, dan 21.6 GFLOPs

### 5) Evaluation

Setelah pelatihan selesai, model dievaluasi untuk mengukur performanya dalam mendeteksi rambu lalu lintas. Dalam evaluasi digunakan 3 model evaluasi yaitu *precision(P)*, *recall(R)*, dan *mean average precision(mAP)*[23] dimana dapat dirumuskan sebagai berikut.

#### a. Precision(P)

*Precision* mengukur akurasi prediksi positif dari model, yaitu seberapa banyak prediksi yang benar-benar sesuai dengan label.

$$Precision = \frac{True\ Positives\ (TP)}{True\ Positives\ (TP) + False\ Positive\ (FP)} \quad (3)$$

#### b. Recall(R)

*Recall* mengukur sensitivitas model dalam mendeteksi semua kasus positif yang sebenarnya.

$$Recall = \frac{True\ Positives\ (TP)}{True\ Positives\ (TP) + False\ Negatives\ (FN)} \quad (4)$$

#### c. Mean Average Precision(mAP)

Average Precision dihitung dengan cara mengukur area di bawah kurva Precision-Recall (PR Curve) untuk setiap kelas objek. Untuk mendapatkan AP, kita menghitung Precision pada berbagai titik Recall. Dalam pendekatan interpolasi, AP dapat dihitung dengan:

$$AP = \int_0^1 P(R) dR \quad (5)$$

atau sebagai rata-rata dari nilai Precision di setiap titik Recall yang diukur. Setelah mendapatkan AP untuk masing-masing kelas, mAP dihitung sebagai rata-rata dari nilai AP seluruh kelas:

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \quad (6)$$

di mana N adalah jumlah kelas, dan AP<sub>i</sub> adalah Average Precision untuk kelas i.

## 3. HASIL DAN PEMBAHASAN

### 3.1 HASIL

Penelitian ini bertujuan untuk mengimplementasikan model YOLOv11 sebagai solusi deteksi rambu lalu lintas secara real-time di Indonesia. Hasil dari proses pelatihan model disajikan pada Tabel 2. berikut

Tabel 2. Hasil pelatihan dengan YOLOv11

Kelas	Precision	Recall	mAP
Balai Pertolongan Pertama	0,996	1	0,995
Banyak Anak-Anak	0,994	1	0,995
Dilarang Belok Kanan	0,996	1	0,995
Dilarang Berhenti	0,996	1	0,995
Dilarang Berjalan Terus	0,995	1	0,995
Dilarang Masuk	0,992	1	0,995
Dilarang Mendahului	1	1	0,995
Dilarang Parkir	0,995	1	0,995
Dilarang Putar Balik	0,993	1	0,995
Gereja	0,993	1	0,995

Kelas	Precision	Recall	mAP
Hati-Hati	0,99	1	0,995
Jalur Penyebrangan	0,996	1	0,995
Lampu Lalu Lintas	0,991	1	0,995
Larangan Kecepatan - 30km-jam	1	1	0,995
Larangan Kecepatan - 40km-jam	0,999	1	0,995
Larangan Kendaraan MST - 10 Ton	0,992	1	0,995
Masjid	0,993	1	0,995
Pemberhentian Bus	0,995	1	0,995
Perintah Ikuti Bundaran	0,993	1	0,995
Perintah Jalur Sepeda	0,995	1	0,995
Perintah Lajur Kiri	0,999	1	0,995
Perintah Pilih Satu Jalur	1	0,969	0,995
Persimpangan 3 Prioritas	0,994	1	0,995
Persimpangan 3 Sisi Kanan Prioritas	0,99	0,99	0,995
Persimpangan 3 Sisi Kiri Prioritas	0,993	0,993	0,995
Persimpangan Empat	0,985	1	0,995
Putar Balik	0,996	1	0,995
Rumah Sakit	0,992	1	0,995
SPBU	1	0,99	0,995
Tempat Parkir	1	0,996	0,995

Secara umum, model menunjukkan performa yang sangat baik dengan nilai *Precision* dan *Recall* yang tinggi di hampir semua kelas rambu lalu lintas, dengan sebagian besar kelas mencapai *Precision* dan *Recall* mendekati atau mencapai 1. Ini menunjukkan bahwa model mampu mendeteksi dan mengenali rambu lalu lintas dengan sangat baik, mengurangi kemungkinan kesalahan deteksi dan kelalaian objek. Nilai mAP (*Mean Average Precision*) konsisten di angka 0,995 untuk semua kelas. Ini mengindikasikan bahwa model memiliki tingkat akurasi keseluruhan yang sangat tinggi dalam mendeteksi rambu lalu lintas tanpa banyak perbedaan antar kelas. Hasil ini menunjukkan ketepatan model dalam memprediksi semua jenis rambu lalu lintas dengan tingkat presisi dan konsistensi yang hampir sempurna. Meskipun mayoritas kelas memiliki *Precision* dan *Recall* sempurna (1), terdapat beberapa kelas yang sedikit lebih rendah, seperti "Perintah Pilih Satu Jalur" (*Recall* = 0,969), "Persimpangan 3 Sisi Kanan Prioritas" (*Precision* dan *Recall* = 0,99), "SPBU" (*Recall* = 0,99), dan "Tempat Parkir" (*Recall* = 0,996). Penurunan kecil ini masih berada dalam rentang yang sangat baik, namun perlu diperhatikan dalam kasus aplikasi nyata jika kesempurnaan deteksi diinginkan.

#### Analisis Kelas dengan Akurasi Lebih Rendah

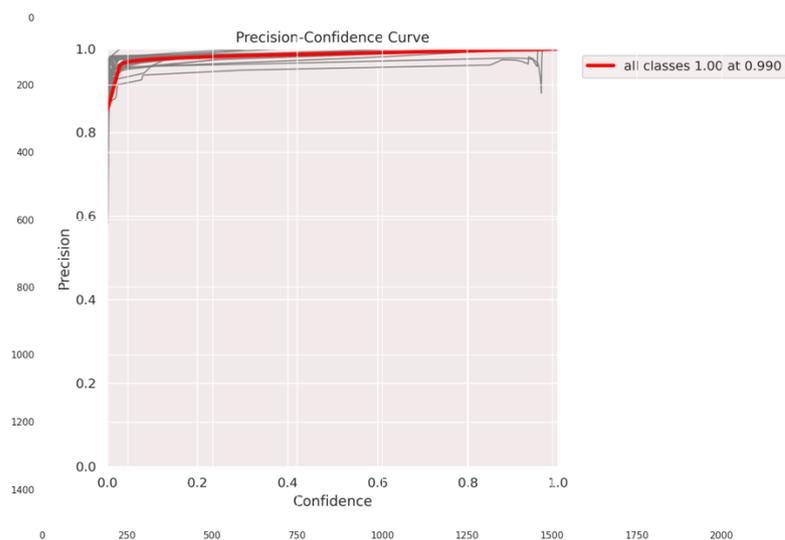
- 1) Pada kelas "Perintah Pilih Satu Jalur" menunjukkan recall yang lebih rendah dibandingkan kelas lainnya, meskipun nilai precision tetap tinggi. Hal ini mengindikasikan adanya kasus rambu yang tidak terdeteksi, meskipun prediksi yang dilakukan pada objek terdeteksi cukup akurat. Gambar rambu di kelas ini sering kali memiliki posisi yang miring atau terhalang, sehingga model kesulitan mengenalinya. Augmentasi data tambahan seperti simulasi gambar rambu dengan berbagai sudut rotasi dan kondisi pencahayaan dapat membantu meningkatkan deteksi.
- 2) Pada kelas "Persimpangan 3 Sisi Kanan Prioritas" Baik precision maupun recall berada di bawah nilai sempurna. Hal ini menunjukkan bahwa selain adanya kesalahan deteksi, terdapat pula kasus salah klasifikasi, di mana model mungkin memprediksi kelas ini sebagai "Persimpangan 3 Sisi Kiri Prioritas." Kemiripan visual antara kelas-kelas rambu yang melibatkan prioritas kanan dan kiri dapat membingungkan model. Dengan melakukan pelabelan ulang dengan memberikan anotasi tambahan

pada gambar misalnya, menandai arah prioritas secara eksplisit dan menambahkan data dari kelas ini untuk melatih model secara lebih spesifik.

- 3) Pada kelas "SPBU" Recall yang lebih rendah menunjukkan beberapa gambar rambu SPBU tidak terdeteksi. Variasi desain rambu SPBU di Indonesia, termasuk perbedaan warna atau ikon yang digunakan oleh berbagai perusahaan, dapat menyebabkan model kesulitan mengenali semua variasi. Dengan mengumpulkan dataset yang lebih representatif dengan berbagai variasi rambu SPBU di seluruh wilayah Indonesia dapat membantu meningkatkan performa.
- 4) Pada kelas "Tempat Parkir" Recall sedikit di bawah 1 menunjukkan beberapa gambar rambu tempat parkir tidak dikenali. Rambu tempat parkir sering kali terletak di lingkungan dengan banyak objek lain di sekitar, seperti kendaraan atau pepohonan, sehingga model mengalami gangguan dalam deteksi. Dengan melakukan preprocessing tambahan untuk mengurangi noise dalam gambar, serta augmentasi untuk mensimulasikan lingkungan kompleks.

### 3.2 Evaluasi

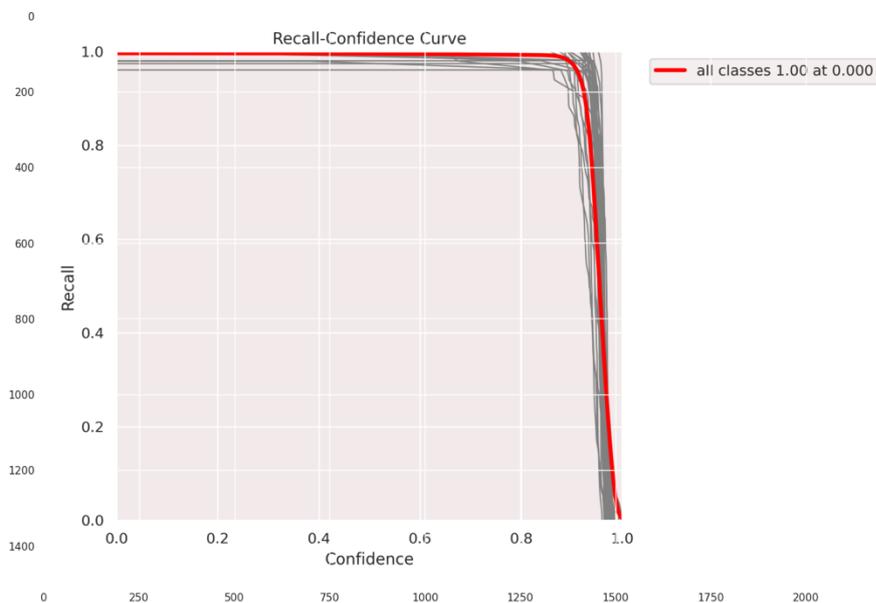
Keakuratan deteksi objek sangat dipengaruhi oleh keseimbangan antara precision dan recall, terutama dalam skenario real-time di mana model harus mampu mengidentifikasi objek dengan cepat tanpa mengorbankan akurasi tinggi grafik juga menunjukkan bahwa saat confidence mencapai nilai sangat tinggi[24]. Berikut ini adalah grafik precision, recall, dan mAP pada Gambar 6., Gambar 7., dan Gambar 8. berikut.



Gambar 6. Precision-Confidence Curve

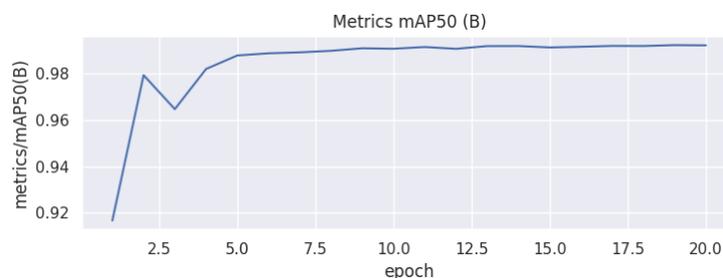
Grafik *Precision-Confidence Curve* ini menunjukkan hubungan antara tingkat kepercayaan dan presisi model YOLOv11 dalam mendeteksi rambu lalu lintas. Sumbu horizontal (X) merepresentasikan tingkat kepercayaan prediksi model yang berkisar antara 0 hingga 1, di mana semakin tinggi nilai confidence, semakin yakin model akan akurasi prediksinya. Sementara itu, sumbu vertikal (Y) menunjukkan nilai presisi, atau proporsi prediksi yang benar terhadap seluruh prediksi pada tingkat confidence tertentu. Kurva ini memperlihatkan bahwa model mampu mempertahankan nilai presisi yang sangat tinggi hampir di seluruh rentang *confidence*, terutama pada *confidence* mendekati 1,0. Pada confidence sekitar 0,990 precision mencapai nilai maksimal, yaitu 1,0 yang menunjukkan bahwa model sangat efektif dalam mendeteksi rambu lalu lintas dengan sedikit sekali kesalahan pada tingkat kepercayaan tinggi. Grafik ini juga menunjukkan bahwa model memiliki keseragaman presisi yang stabil di berbagai *confidence*, dengan kurva merah yang mendatar di tingkat presisi tinggi. Artinya, model ini konsisten dalam menghasilkan prediksi yang akurat dan dapat diandalkan dalam berbagai tingkat confidence. Hal ini mengindikasikan bahwa model yang digunakan memiliki kinerja deteksi yang sangat baik, menjadikannya ideal untuk diterapkan dalam sistem deteksi rambu lalu lintas yang membutuhkan akurasi tinggi di dunia nyata. Presisi tinggi pada tingkat *confidence* yang luas juga memastikan bahwa sistem dapat bekerja

dengan efisien dalam aplikasi real-time, seperti pada kendaraan otonom atau sistem navigasi pintar, di mana keandalan dalam mendeteksi rambu lalu lintas sangat penting[25].



Gambar 7. Recall-Confidence Curve

Grafik *Recall-Confidence Curve* ini menunjukkan hubungan antara tingkat kepercayaan model dan nilai recall untuk setiap kelas rambu lalu lintas yang dideteksi. Sumbu horizontal (X) merepresentasikan confidence model, yang berkisar antara 0 hingga 1, sedangkan sumbu vertikal (Y) menunjukkan nilai recall, yang merupakan rasio dari prediksi benar terhadap total kejadian aktual dari objek yang dideteksi. Dari grafik ini terlihat bahwa pada hampir seluruh rentang *confidence* hingga mendekati 1, nilai recall model berada pada tingkat maksimum, yaitu 1,0. Hal ini menunjukkan bahwa model memiliki kemampuan tinggi dalam mendeteksi seluruh kejadian dari rambu yang ada, dengan sedikit atau bahkan tanpa kelalaian pada prediksi. Namun, grafik juga menunjukkan bahwa saat *confidence* mencapai nilai sangat tinggi (mendekati 1), terjadi sedikit penurunan pada nilai recall. Ini mengindikasikan bahwa pada tingkat kepercayaan yang terlalu tinggi, model menjadi lebih selektif dalam prediksi, yang kadang mengorbankan beberapa prediksi benar demi memastikan akurasi. Secara keseluruhan, hasil ini menegaskan bahwa model YOLOv11 yang digunakan dalam penelitian memiliki performa deteksi yang sangat baik dan dapat diandalkan untuk aplikasi real-time dalam mendeteksi rambu lalu lintas di Indonesia. Model ini mampu mempertahankan tingkat recall yang maksimal dalam rentang *confidence* yang luas, yang berarti model efektif dalam mendeteksi seluruh rambu tanpa banyak kehilangan informasi penting, sehingga ideal untuk aplikasi yang membutuhkan sensitivitas tinggi seperti sistem keselamatan berkendara[26].



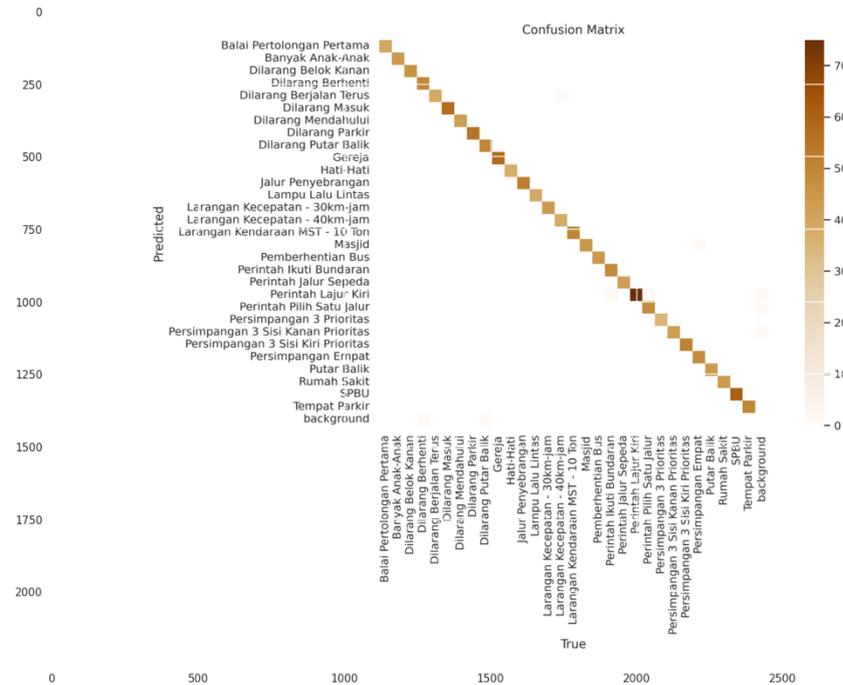
Gambar 8. Metrics mAP50

Grafik ini menunjukkan perkembangan mAP50 (Mean Average Precision pada IoU threshold 0.5) selama 20 epoch pelatihan model. Pada epoch awal (1-5), mAP50 meningkat tajam dari sekitar 0,92 ke 0,99, menunjukkan bahwa model belajar cepat dan meningkatkan akurasi deteksi objek secara signifikan. Setelah epoch ke-5,

mAP50 cenderung stabil di sekitar 0,98 hingga 0,99, mengindikasikan bahwa model telah mencapai konvergensi dan mempertahankan performa optimal. Artinya, pelatihan tambahan di atas 20 epoch kemungkinan tidak akan meningkatkan akurasi secara signifikan, sehingga model ini sudah cukup akurat dan stabil dalam mendeteksi objek pada tahap ini.

### 3.3 Confusion Matrix

*Confusion Matrix* adalah alat evaluasi yang membantu untuk melihat lebih detail performa model dalam mendeteksi setiap kelas[27]. Hasil confusion matrix disajikan pada Gambar 9. berikut.



Gambar 9. *Confusion Matrix*

Pada Gambar 9. menunjukkan model deteksi terhadap berbagai kelas rambu lalu lintas. Sumbu horizontal (*True*) menunjukkan kelas asli rambu, sementara sumbu vertikal (*Predicted*) menunjukkan kelas prediksi model. Warna yang lebih gelap di sepanjang diagonal matrix menunjukkan jumlah prediksi yang benar untuk setiap kelas, dengan semakin gelap berarti semakin banyak prediksi yang benar. Mayoritas prediksi berada di sepanjang diagonal, yang menunjukkan bahwa model berhasil mendeteksi sebagian besar kelas dengan akurat. Terdapat sedikit area berwarna di luar diagonal, yang menunjukkan kesalahan prediksi (*misclassifications*) untuk beberapa kelas tertentu, namun jumlahnya relatif kecil. Secara keseluruhan, *Confusion Matrix* ini menunjukkan bahwa model memiliki akurasi tinggi dalam mendeteksi berbagai jenis rambu lalu lintas dengan kesalahan yang minimal.

## 4. KESIMPULAN

Penelitian ini berhasil mengimplementasikan model YOLOv11 sebagai solusi deteksi rambu lalu lintas real-time dengan performa yang sangat baik. Model mencapai akurasi tinggi, dengan nilai mAP50 yang stabil di sekitar 0,99 setelah beberapa epoch pelatihan, menunjukkan bahwa model mampu mendeteksi berbagai jenis rambu lalu lintas dengan tingkat akurasi dan keandalan yang memadai untuk aplikasi *real-time*. Evaluasi melalui *Confusion Matrix* menunjukkan bahwa sebagian besar prediksi model berada di sepanjang diagonal, yang mengindikasikan tingkat kesalahan yang minimal dalam klasifikasi rambu. Hasil ini menandakan bahwa YOLOv11 mampu mendeteksi rambu lalu lintas dengan presisi tinggi dan kesalahan minimal. Namun, beberapa kelas rambu tertentu masih menunjukkan sedikit kesalahan yang dapat diperbaiki melalui pelatihan lebih lanjut atau peningkatan dataset yang lebih representatif. Secara keseluruhan, penelitian ini menunjukkan bahwa model YOLOv11 memiliki potensi besar untuk diterapkan dalam sistem keselamatan berkendara dan

aplikasi transportasi cerdas di Indonesia. Teknologi ini dapat diintegrasikan dalam kendaraan atau sistem pemantauan lalu lintas untuk mendukung pengemudi dalam mengenali rambu-rambu dengan cepat dan akurat, sehingga berpotensi meningkatkan keselamatan jalan. Rekomendasi untuk penelitian lanjutan adalah penting untuk mengumpulkan dataset yang lebih beragam dan representatif, terutama untuk kelas-kelas dengan performa lebih rendah. Dataset yang mencakup berbagai kondisi pencahayaan, cuaca, dan lingkungan akan membantu model lebih adaptif terhadap variasi yang ditemukan di dunia nyata. Eksplorasi model deteksi objek lainnya, seperti model berbasis Transformer atau kombinasi CNN dan Transformer, juga dapat dijajaki untuk meningkatkan kemampuan model dalam mendeteksi rambu dengan kemiripan visual yang tinggi, yang sering kali menjadi tantangan bagi model YOLO.

## UCAPAN TERIMA KASIH

Kami ucapkan terima kasih kepada Direktorat Jenderal Pendidikan Vokasi, Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi, atas dukungan pendanaan yang diberikan untuk pelaksanaan penelitian dosen pemula pada tahun 2024.

## REFERENSI

- [1] Kementerian Perhubungan Indonesia, "Laporan Statistik Kecelakaan Lalu Lintas Tahun 2021." 2022.
- [2] Polri, "Korban Meninggal Kecelakaan Lalu Lintas Mayoritas Usia Produktif," *Korlantas Polri*, Oktober 2024.
- [3] Muh. Ikbal and R. A. Saputra, "PENGENALAN RAMBU LALU LINTAS MENGGUNAKAN METODE YOLOV8," *JIKA J. Inform.*, vol. 8, no. 2, p. 204, Apr. 2024, doi: 10.31000/jika.v8i2.10609.
- [4] E. Hrustic, Z. Xu, and D. Vivet, "Deep Learning Based Traffic Signs Boundary Estimation," *IEEE Intell. Veh. Symp. Proc.*, no. Iv, pp. 451–456, 2020, doi: 10.1109/IV47402.2020.9304590.
- [5] J. Cao, J. Zhang, and X. Jin, "A Traffic-Sign Detection Algorithm Based on Improved Sparse R-cnn," *IEEE Access*, vol. 9, pp. 122774–122788, 2021, doi: 10.1109/ACCESS.2021.3109606.
- [6] R. Mahadshetti, J. Kim, and T.-W. Um, "Sign-YOLO: Traffic Sign Detection Using Attention-Based YOLOv7," *IEEE Access*, vol. 12, pp. 132689–132700, 2024, doi: 10.1109/ACCESS.2024.3417023.
- [7] R. K. Megalingam, K. Thanigundala, S. R. Musani, H. Nidamanuru, and L. Gadde, "Indian traffic sign detection and recognition using deep learning," *Int. J. Transp. Sci. Technol.*, vol. 12, no. 3, pp. 683–699, Sep. 2023, doi: 10.1016/j.ijst.2022.06.002.
- [8] N. Triki, M. Karray, and M. Ksantini, "A Real-Time Traffic Sign Recognition Method Using a New Attention-Based Deep Convolutional Neural Network for Smart Vehicles," *Appl. Sci.*, vol. 13, no. 8, p. 4793, Apr. 2023, doi: 10.3390/app13084793.
- [9] I. J. Thira, D. Riana, A. N. Ilhami, B. R. S. Dwinanda, and H. Choerunisyah, "Pengenalan Alfabet Sistem Isyarat Bahasa Indonesia (SIBI) Menggunakan Convolutional Neural Network," *J. Algoritma*, vol. 20, no. 2, pp. 421–432, Oct. 2023, doi: 10.33364/algoritma/v.20-2.1480.
- [10] X. Cao, Y. Xu, J. He, J. Liu, and Y. Wang, "A Lightweight Traffic Sign Detection Method With Improved YOLOv7-Tiny," *IEEE Access*, vol. 12, pp. 105131–105147, 2024, doi: 10.1109/ACCESS.2024.3435384.
- [11] Y. Tian, J. Gelernter, X. Wang, J. Li, and Y. Yu, "Traffic Sign Detection Using a Multi-Scale Recurrent Attention Network," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 12, pp. 4466–4475, 2019, doi: 10.1109/TITS.2018.2886283.
- [12] W. Farag, "Recognition of traffic signs by convolutional neural nets for self-driving vehicles," *Int. J. Knowl.-Based Intell. Eng. Syst.*, vol. 22, no. 3, pp. 205–214, 2018, doi: 10.3233/KES-180385.
- [13] A. Wang *et al.*, "YOLOv10: Real-Time End-to-End Object Detection," May 23, 2024, *arXiv: arXiv:2405.14458*. Accessed: Sep. 16, 2024. [Online]. Available: <http://arxiv.org/abs/2405.14458>
- [14] M. Beyersdorff and *et al.*, "YOLOv8: Towards the Next Generation of Real-Time Object Detection," 2023.
- [15] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *ArXiv Prepr. ArXiv180402767*, vol. 1, pp. 1–6, 2016.
- [16] M. Hussain, "YOLOv1 to v8: Unveiling Each Variant—A Comprehensive Review of YOLO," *IEEE Access*, vol. 12, pp. 42816–42833, 2024, doi: 10.1109/ACCESS.2024.3378568.
- [17] R. Khanam and M. Hussain, "YOLOv11: An Overview of the Key Architectural Enhancements," Oct. 23, 2024, *arXiv: arXiv:2410.17725*. Accessed: Oct. 30, 2024. [Online]. Available: <http://arxiv.org/abs/2410.17725>

- [18] A. Mulyanto, R. I. Borman, P. Prasetyawan, W. Jatmiko, P. Mursanto, and A. Sinaga, "Indonesian Traffic Sign Recognition for Advanced Driver Assistent (ADAS) Using YOLOv4," *2020 3rd Int. Semin. Res. Inf. Technol. Intell. Syst. ISRITI 2020*, pp. 520–524, 2020, doi: 10.1109/ISRITI51436.2020.9315368.
- [19] A. A. Khalifa, W. M. Alayed, H. M. Elbadawy, and R. A. Sadek, "Real-Time Navigation Roads: Lightweight and Efficient Convolutional Neural Network (LE-CNN) for Arabic Traffic Sign Recognition in Intelligent Transportation Systems (ITS)," *Appl. Sci.*, vol. 14, no. 9, p. 3903, May 2024, doi: 10.3390/app14093903.
- [20] C. Han, G. Gao, and Y. Zhang, "Real-time small traffic sign detection with revised faster-RCNN," *Multimed. Tools Appl.*, vol. 78, no. 10, pp. 13263–13278, 2019, doi: 10.1007/s11042-018-6428-0.
- [21] Muh. Iqbal, "Traffic sign in indonesia." 2024. [Online]. Available: <https://www.kaggle.com/datasets/ikbal12082004/traffic-sign-in-indonesia/data>
- [22] G. Oreski, "YOLO\*C — Adding context improves YOLO performance," *Neurocomputing*, vol. 555, p. 126655, Oct. 2023, doi: 10.1016/j.neucom.2023.126655.
- [23] Y. Cui, D. Guo, H. Yuan, H. Gu, and H. Tang, "Enhanced YOLO Network for Improving the Efficiency of Traffic Sign Detection," *Appl. Sci.*, vol. 14, no. 2, p. 555, Jan. 2024, doi: 10.3390/app14020555.
- [24] J. Redmon and A. Farhadi, "Advancements in YOLO Object Detection for Real-World Applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 6, pp. 1254–1265, 2023.
- [25] M. Satria and H. Permadi, "Kecerdasan Buatan dan Masa Depan Transportasi di Indonesia," *J. Teknol. Transp.*, vol. 9, no. 4, pp. 298–310, 2021.
- [26] L. Purwanto and W. Riyanto, "Tantangan dan Peluang dalam Implementasi Sistem Deteksi Rambu di Indonesia," *J. Inov. Transp.*, vol. 5, no. 1, pp. 23–35, 2022.
- [27] D. Chicco and G. Jurman, "The Advantages of the Matthews Correlation Coefficient (MCC) over F1 Score and Accuracy in Binary Classification Evaluation," *BMC Genomics*, vol. 21, no. 1, p. 6, 2020.