

Implementasi YOLOv8 untuk Deteksi Pelat Nomor dan Validasi Pajak Kendaraan

Muhammad Wesya Pralega¹, Witta Listiya Ningrum²

¹Informatika, Fakultas Teknologi Industri, Universitas Gunadarma

²Teknologi Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, Universitas Gunadarma

wesyaega6@gmail.com, wita_listiya@staff.gunadarma.ac.id*

Abstract

The increase in the number of vehicles in Indonesia requires a more efficient administrative system for vehicle validity validation. Manual verification processes such as checking vehicle registration certificates and license plates by officers in the field are considered ineffective, prone to error, and time-consuming, especially when dealing with high volumes of vehicles. This study aims to develop a computer vision-based automated system capable of detecting vehicle license plates and independently validating tax status. The method used is the CRISP-DM method, which includes understanding requirements, data processing, modeling, evaluation, and implementation. The model used is YOLOv8 to detect the license plate area, and EasyOCR is used for alphanumeric character recognition. The research dataset consists of 587 secondary images and 15 primary images. The secondary data was divided into 70% training data, 20% validation data, and 10% test data. The YOLOv8 model was trained using the best combination of hyperparameters, namely 200 epochs, batch size 16, and learning rate 0.01, which produced a box loss value of 0.38. The tax status validation process is divided into four categories: active, expired, invalid, and no tax information available. Thus, this research can contribute to the development of an effective vehicle tax validation automation system that has the potential to be implemented in public administration services.

Keywords: yolov8, easyocr, license plate detection, CRISP-DM, streamlit.

Abstrak

Seiring dengan peningkatan jumlah kendaraan di Indonesia menuntut adanya sistem administrasi yang lebih efisien dalam proses validasi masa berlaku kendaraan. Proses verifikasi manual seperti pengecekan STNK dan pelat nomor oleh petugas di lapangan dinilai kurang efektif, rawan kesalahan, serta membutuhkan waktu yang lebih dalam terutama pada volume kendaraan yang tinggi. Penelitian ini bertujuan untuk mengembangkan sistem otomatis berbasis *computer vision* yang mampu mendeteksi pelat nomor kendaraan serta memvalidasi status pajak secara mandiri. Metode yang digunakan adalah metode *CRISP-DM* yang meliputi pemahaman kebutuhan, pengolahan data, pemodelan, evaluasi dan implementasi. Model yang digunakan adalah *YOLOv8* untuk mendeteksi area pelat nomor, dan *EasyOCR* digunakan sebagai pengenalan karakter alfanumerik. Dataset penelitian terdiri dari 587 gambar sekunder dan 15 gambar primer. Data sekunder terbagi menjadi 70% data latih, 20% data validasi, dan 10% data uji. Model *YOLOv8* dilatih dengan menggunakan kombinasi *hyperparameter* terbaik yaitu 200 *epoch*, *batch size* 16, dan *learning rate* 0.01 yang menghasilkan nilai *box loss* sebesar 0.38. Proses validasi status pajak terbagi menjadi empat kategori: aktif, kadaluwarsa, invalid, dan tidak terdapat informasi pajak. Dengan demikian, penelitian ini dapat memberikan kontribusi dalam pengembangan sistem otomatis validasi pajak kendaraan yang efektif serta berpotensi untuk diimplementasikan dalam layanan administrasi publik.

Kata kunci: *yolov8, easyocr, deteksi pelat nomor, CRISP-DM, streamlit.*

©This work is licensed under a Creative Commons Attribution -ShareAlike 4.0 International License

1. Pendahuluan

Pertumbuhan jumlah kendaraan di Indonesia terus meningkat setiap tahun seiring dengan meningkatnya mobilitas masyarakat. Pada tahun 2024, tercatat lebih dari 164 juta kendaraan terdaftar di Indonesia, dengan kenaikan sekitar 5 juta unit dalam kurun waktu delapan bulan [1]. Kondisi ini menuntut adanya sistem administrasi yang efektif, khususnya dalam melakukan proses validasi masa berlaku pajak kendaraan. Prosedur pemeriksaan yang dilakukan petugas di lapangan menyebabkan proses menjadi kurang efektif dan membutuhkan waktu yang lebih lama terutama ketika terjadinya lonjakan volume kendaraan yang tinggi. Oleh karena itu, dibutuhkan solusi berbasis teknologi untuk mendukung proses verifikasi data kendaraan agar cepat dan terotomatisasi.

Computer vision merupakan salah satu teknologi yang memungkinkan komputer mampu melihat dan mengenali objek [2]. Teknologi tersebut banyak dimanfaatkan untuk berbagai tugas visual, seperti deteksi objek dan pembacaan karakter. Salah satu pendekatan untuk pendeteksian objek secara real-time adalah menggunakan YOLO (*You Only Look Once*) [3].

Beberapa penelitian terdahulu telah mengembangkan sistem deteksi pelat nomor menggunakan model YOLO. Penelitian oleh [4] dalam “*Sistem Pendeteksi Pelat Nomor Polisi Kendaraan dengan Arsitektur YOLOv8*” menunjukkan bahwa model YOLOv8 mampu mendeteksi pelat nomor dengan cukup baik, ditunjukkan oleh nilai *precision* sebesar 0.871 dan *recall* sebesar 0.85. Namun, penelitian tersebut hanya

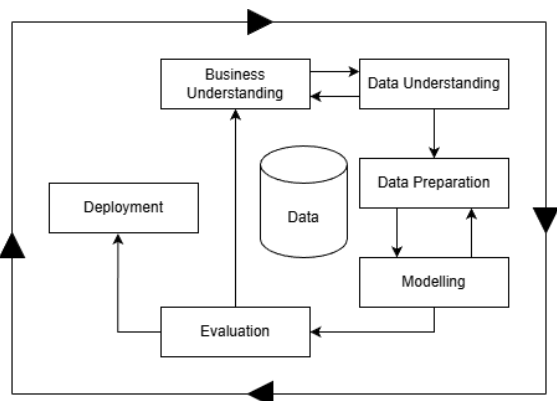
berfokus pada proses identifikasi pelat nomor tanpa melakukan analisis lebih lanjut terhadap status pajak kendaraan.

Penelitian lain oleh [5] dalam judul penelitiannya yang berjudul “Implementasi Metode YOLOv5 dan Tesseract OCR untuk Deteksi Pelat Nomor Kendaraan” menggunakan deteksi objek dengan OCR untuk membaca karakter pelat nomor. Sistem yang dikembangkan mencapai akurasi pembacaan 70%, tetapi belum mencakup proses validasi bulan dan tahun pajak kendaraan.

Permasalahan utama dalam verifikasi pajak kendaraan saat ini adalah masih bergantung pada pemeriksaan manual yang memerlukan waktu dan tidak efisien. Pendekatan yang telah ada umumnya hanya berfokus pada deteksi atau pembacaan pelat nomor tanpa melakukan validasi informasi pajak secara otomatis. Keterbatasan ini menunjukkan perlunya sistem yang mampu mengintegrasikan deteksi pelat nomor dengan proses ekstraksi dan validasi pajak secara menyeluruh. Penelitian ini menggunakan pendekatan berbasis *computer vision* dengan mengintegrasikan YOLOv8 untuk deteksi pelat nomor dan EasyOCR untuk ekstraksi karakter, yang dikembangkan menggunakan alur CRISP-DM hingga tahap *deployment* berbasis web. Kontribusi penelitian ini adalah menyediakan sistem otomatis validasi pajak kendaraan yang terukur dan aplikatif, sehingga berpotensi meningkatkan efisiensi layanan administrasi kendaraan.

2. Metode Penelitian

Penelitian ini menggunakan pendekatan *computer vision* untuk mendeteksi pelat nomor kendaraan dan melakukan validasi masa pajak secara otomatis. Model dikembangkan berdasarkan tahapan CRISP-DM yang terdiri dari enam tahapan yaitu: *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modelling*, *Evaluation* dan *Deployment* [6]. Kerangka yang digunakan pada penelitian ini dapat dilihat pada Gambar 1.



Gambar 1. Alur Metode CRISP-DM

2.1. Business Understanding

Pada tahap pertama adalah memahami permasalahan dan sasaran penelitian terkait proses validasi pajak

kendaraan. Saat ini, pengecekan pajak yang dilakukan petugas dilakukan secara manual oleh petugas sehingga menyebabkan keterlambatan dan tidak efektifitas ketika volume kendaraan melonjak tinggi. Oleh karena itu, diperlukan sistem otomatis yang mampu melakukan deteksi pelat nomor dan validasi pajak kendaraan secara cepat dan *real-time*. Tujuan dari penelitian ini adalah menghasilkan sistem yang dapat mendukung proses pemeriksaan pajak kendaraan secara lebih efisien dan terotomatisasi.

2.2. Data Understanding

Pada fase data *understanding* merupakan fase mengidentifikasi, mengumpulkan dan menganalisis kumpulan data [7]. Data yang digunakan terdiri dari 602 gambar yang terbagi menjadi 587 data sekunder dan 15 data primer. Data sekunder diperoleh dari platform *Roboflow* yang diunggah pada tahun 2024, yang kemudian digunakan sebagai data utama pelatihan dan pengujian model dalam penelitian ini. sementara data primer diperoleh langsung di lapangan pada tahun 2025 untuk merepresentasikan kondisi nyata di lingkungan.

Pemilihan data tersebut dilakukan dengan mempertimbangkan keberagaman kondisi gambar, meliputi variasi jarak pengambilan gambar, sudut pandang, serta variasi jenis dan warna pelat nomor kendaraan. Pemilihan kriteria ini bertujuan agar dataset mampu merepresentasikan kondisi aktual di lapangan sehingga model dapat belajar secara optimal dan memiliki kemampuan generalisasi yang baik.

2.3. Data Preparation

Tahap data *preparation* merupakan proses untuk membangun dataset akhir dari data mentah agar lebih optimal untuk proses modeling [8]. Pemeriksaan awal dilakukan untuk memastikan anotasi *bounding box* sesuai dengan posisi pelat nomor pada gambar. Anotasi tersebut kemudian dikonversi ke format YOLO, serta melakukan *resize* gambar menjadi 640 x 640 piksel. Proses augmentasi diterapkan, meliputi rotasi, *horizontal flip*, *scaling*, dan translasi untuk meningkatkan keragaman data. Kemudian data tersebut terbagi menjadi *data training*, *data validation* dan *data test* untuk memastikan proses pelatihan dan evaluasi dilakukan secara terukur dan terstruktur.

2.4. Modeling

Model deteksi objek dibangun menggunakan arsitektur YOLOv8 untuk mendeteksi objek pelat nomor kendaraan pada gambar. Setelah pelat terdeteksi, kemudian area pelat dipotong (*cropped*) dan diproses oleh EasyOCR untuk melakukan ekstraksi karakter alfanumerik pada pelat. Informasi hasil OCR tersebut digunakan untuk memperoleh informasi bulan dan tahun masa berlaku pajak kendaraan, yang selanjutnya digunakan sebagai dasar dalam penentuan status pajak

2.5. Evaluation

Tahap *evaluation* bertujuan untuk memastikan model yang dikembangkan tepat digunakan untuk kasus penelitian ini dan sesuai dengan tujuan yang telah direncanakan [9]. Evaluasi dilakukan dengan menggunakan data *validation* dan *test* untuk memastikan model mampu melakukan generalisasi pada data baru. Kinerja model dinilai menggunakan metrik *precision*, *recall*, *F1-Score*, dan *mAP* sebagai indikator keberhasilan deteksi objek. Selain itu, *confusion matrix* juga digunakan untuk menganalisis distribusi prediksi deteksi pelat, seperti *true positive*, *false positive*, *true negative*, dan *false negative*.

2.6. Deployment

Pada tahap *deployment* merupakan tahap penerapan model yang telah dikembangkan dan dievaluasi pada sistem [10]. Pada kasus ini model diimplementasikan dengan menggunakan *YOLOv8* sebagai deteksi pelat nomor, *EasyOCR* sebagai pembaca teks, dan modul validasi pajak ke dalam antarmuka web berbasis *streamlit*. Hasil yang ditampilkan secara interaktif dan sederhana. Tujuannya agar sistem dapat digunakan secara praktis oleh masyarakat maupun instansi terkait sebagai sarana pendukung verifikasi pajak kendaraan secara cepat, efisien, dan mudah diakses.

3. Hasil dan Pembahasan

Penelitian ini menggunakan bahasa pemrograman *python* sebagai komponen utama dalam pengembangan dan implementasi model *YOLOv8* sebagai model deteksi objek, *EasyOCR* sebagai pembaca teks, dan *Streamlit* sebagai media *deployment* antarmuka web. Seluruh proses pelatihan model dilakukan secara lokal menggunakan GPU. Hasil integrasi ketiga komponen tersebut menghasilkan sistem berbasis web yang mampu mendeteksi pelat nomor kendaraan, membaca karakter teks, serta menampilkan hasil validasi pajak kendaraan secara *real-time*.

3.1. Business Understanding

Berdasarkan latar belakang penelitian, permasalahan utama yang diangkat adalah masih belum efisiennya proses pemeriksaan dan validasi pajak kendaraan yang hingga saat ini masih dilakukan secara manual oleh petugas. Proses tersebut membutuhkan waktu yang lama, terutama ketika volume kendaraan terus meningkat secara signifikan. Kondisi ini membutuhkan sistem otomatis yang mampu melakukan deteksi pelat nomor serta validasi masa berlaku pajak secara cepat dan *real-time*.

Dari permasalahan tersebut, dikembangkanlah sistem berbasis *computer vision* dengan memanfaatkan model *YOLOv8* sebagai pendeteksi pelat nomor kendaraan dan *EasyOCR* sebagai alat pembaca teks. Kedua komponen ini dikombinasikan menjadi satu alur sehingga menampilkan hasil deteksi pelat dan status pajak kendaraan secara otomatis melalui antarmuka web. Pendekatan ini dipilih karena *YOLOv8* memiliki

keunggulan dalam efisien pelatihan dibandingkan dengan versi sebelumnya, sementara *EasyOCR* dapat mengenali karakter alfanumerik tanpa perlu pelatihan tambahan. Dengan demikian, penelitian ini diharapkan dapat membantu digitalisasi pelayanan publik, agar lebih efisien dan mengurangi beban administratif petugas lapangan.

3.2. Data Understanding

Pada tahap ini dilakukan proses pemahaman terhadap data yang digunakan dalam penelitian, meliputi pengumpulan data, struktur serta format anotasi, hingga konfigurasi dataset yang digunakan untuk pelatihan model deteksi pelat nomor kendaraan. Pertama adalah pengumpulan data yang terdiri dari dua jenis, yaitu data sekunder dan data primer.

Data sekunder diperoleh dari platform *roboflow*, dengan nama dataset "*Plat nomor kendaraan Indonesia Computer Vision Dataset*" yang diunggah oleh Denny. Data tersebut terdiri dari 587 gambar kendaraan baik kendaraan mobil maupun kendaraan bermotor. Dataset ini menjadi sumber utama untuk proses pelatihan dan pengujian model deteksi pelat nomor kendaraan. Dataset tersebut dapat diakses melalui tautan berikut: <https://universe.roboflow.com/institut-teknologi-sumatera-di-lxu/plat-nomor-kendaraan-indonesia/>.

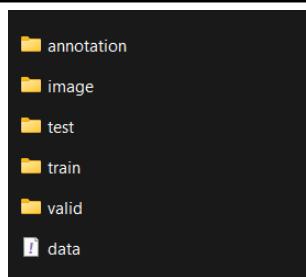
Selain data sekunder, terdapat data primer yang terdiri dari 15 gambar yang diperoleh melalui dua metode, yaitu pengambilan langsung di lapangan menggunakan kamera ponsel untuk melihat di berbagai situasi seperti pencahayaan terang, redup, dan jarak yang berbeda. Lalu pengunduhan dari *Google Images*, yang digunakan melengkapi variasi data, terutama untuk kasus yang jarang ditemukan secara langsung seperti pelat khusus atau kondisi tertentu. Penggunaan data primer bertujuan untuk melakukan uji coba lapangan agar dapat menilai kemampuan sistem dalam melakukan deteksi pelat dengan kondisi nyata di luar dataset pelatihan.

Tabel 1. Sebaran Dataset Penelitian

Jenis Dataset	Jumlah Gambar
Data Sekunder	587
Data Primer	15
Total	602

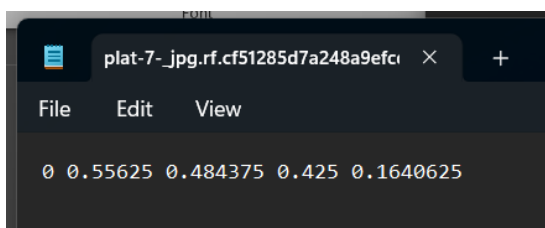
Berdasarkan Tabel 1, data sekunder mendominasi jumlah dataset, sedangkan data primer berfungsi untuk menguji kemampuan model pada kondisi nyata.

Selanjutnya adalah struktur dan format anotasi, dataset disusun sesuai dengan format pelatihan *YOLOv8* yang terdiri dari direktori *images*, *labels*, serta tiga *subset* utama yaitu *train*, *val*, dan *test*, sebagaimana yang ditunjukkan pada Gambar 2.



Gambar 2 Struktur Direktori Dataset

Pada direktori *image* menyimpan gambar pelat nomor, sedangkan, direktori *labels* menyimpan anotasi berformat *.txt* yang dapat dilihat pada Gambar 3.



Gambar 3 Format Anotasi YOLOv8

Konfigurasi dataset diatur melalui file *data.yaml*, yang mendefinisikan direktori data pelatihan, validasi, dan pengujian, serta menentukan jumlah dan nama kelas objek, dapat dilihat pada Gambar 4. Pada penelitian ini menggunakan satu kelas objek, yaitu 'plat_nomor'.

```

1 train: ../train/images
2 val: ../valid/images
3 test: ../test/images
4
5 nc: 1
6 names: ['plat_nomor']
    
```

Gambar 4 Konfigurasi data.yaml

3.3. Data Preparation

Tahap berikutnya adalah data *preparation* yang bertujuan untuk mempersiapkan dataset agar siap digunakan dalam proses pelatihan model deteksi pelat nomor menggunakan YOLOv8. Tahapan yang dilakukan mencakup melakukan verifikasi anotasi *bounding box*, pelabelan data, pembagian dataset, dan penerapan *augmentasi* data untuk memperkaya ragam gambar pada saat proses pelatihan.

Langkah awal dilakukan verifikasi anotasi *bounding box* untuk memastikan bahwa setiap gambar telah memiliki anotasi yang sesuai. Proses ini menggunakan fungsi *get_bbox()* yang dapat dilihat pada Program 1.

```

Program 1. Membaca Koordinat bounding box
def get_bbox(file_path):
    info = xet.parse(file_path)
    root = info.getroot()
    member_object = root.find('object')

    labels_info = member_object.find('bndbox')
    xmin = int(labels_info.find('xmin').text)
    xmax = int(labels_info.find('xmax').text)
    
```

```

ymin = int(labels_info.find('ymin').text)
ymax = int(labels_info.find('ymax').text)

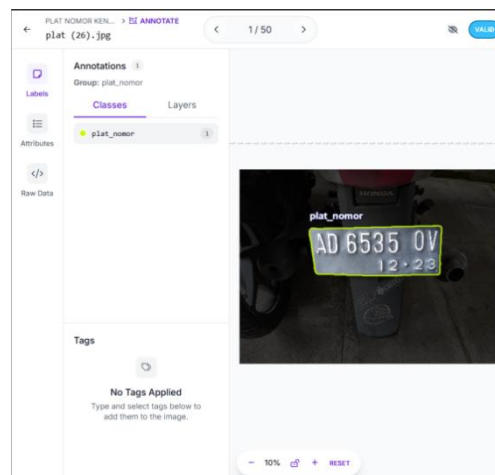
return xmin, xmax, ymin, ymax
    
```

Hasil *bounding box* dapat pada Gambar 5 menunjukkan *bounding box* sudah sesuai dengan posisi pelat nomor sebenarnya.



Gambar 5. Verifikasi dengan Bounding Box

Setelah melakukan verifikasi *bounding box* sudah selesai, proses dilanjutkan dengan melakukan pelabelan data menggunakan platform *robolow*. Setiap objek diberi label dengan nama kelas 'plat_nomor', yang dapat dilihat pada Gambar 6



Gambar 6. Pelabelan Data

Proses pelabelan dilakukan secara manual untuk memastikan bahwa setiap gambar teranotasi dengan akurat. Langkah ini bertujuan untuk mengenali pola visual pelat nomor agar konsisten selama proses pelatihan.











Dataset yang telah diberi label, selanjutnya dibagi menjadi tiga bagian utama, yaitu *training set*, *validation set*, dan *test set* dengan porsi 70:20:10. Pembagian dataset dapat dilihat pada Tabel 2.

Tabel 2. Pembagian Dataset Penelitian

Jenis Dataset	Jumlah Gambar	Persentase
Training Set	411	70%
Validation Set	117	20%
Test Set	59	10%
Total	587	100%

Agar model mampu menghadapi kondisi nyata yang bervariasi, dilakukan proses augmentasi data untuk menambahkan keragaman gambar pelatihan. Teknik augmentasi yang digunakan dapat dilihat pada Tabel 3

Tabel 3 Teknik Augmentasi yang Diterapkan

No	Deskripsi	Gambar Asli	Hasil Augmentasi
1.	Rotasi, Memutar gambar 10 derajat		
2.	Translasi, menggeser gambar hingga 8% secara horizontal atau vertikal.		
3.	Scale, mengubah skala objek hingga 10%.		
4.	Erasing, menghapus beberapa bagian gambar sebesar 20%.		
5.	Flip_lr, membalikkan gambar dari kiri ke kanan.		

3.4. Modeling

Setelah pembagian dataset, selanjutnya adalah proses pembangunan dan pelatihan model deteksi pelat nomor kendaraan dengan menggunakan YOLOv8 versi Nano (YOLOv8n). Pelatihan model dilakukan dengan menggunakan bahasa pemrograman Python dengan

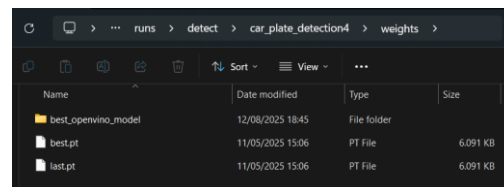
dukungan GPU lokal agar komputasi berjalan optimal. Proses pelatihan menggunakan *pre-trained* model yaitu *yolov8n.pt*, yang merupakan model dasar dari YOLOv8 dengan ukuran yang kecil dan efisien.

Penelitian ini menggunakan beberapa konfigurasi *hyperparameter* yang digunakan selama proses pelatihan. Tujuannya adalah untuk menemukan parameter yang menghasilkan nilai *loss* terendah dan waktu pelatihan paling efisien.

Tabel 4. Kombinasi *Hyperparameter*

Epoch	Batch Size	Learning Rate	Confidence Score	Box Loss	Waktu (menit)
200	16	0.01	0.9	0.38	27
150	8	0.001	0.9	0.39	20
100	4	0.0001	0.9	0.45	20
50	32	0.00001	0.9	0.63	5

Pada Tabel 4 dapat dilihat bahwa konfigurasi dengan *epoch* 200, *batch size* 16, dan *learning rate* 0.01 menghasilkan nilai *box loss* paling rendah yaitu 0.38. Seluruh kombinasi pelatihan menggunakan rasio dataset yang sama, yaitu 70% data pelatihan, 20% data validasi, dan 10% data pengujian.



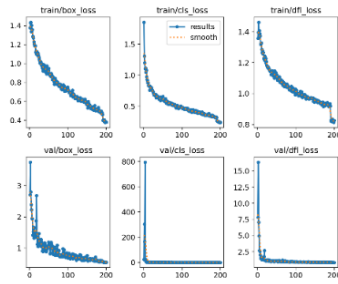
Gambar 7. File Output Model *best.pt*

Pada Gambar 7 merupakan hasil dari proses pelatihan model. File tersebut digunakan untuk mendeteksi pelat nomor kendaraan, yang diintegrasikan dengan *EasyOCR* untuk melakukan pembacaan teks dari pelat yang terdeteksi.

3.5. Evaluation

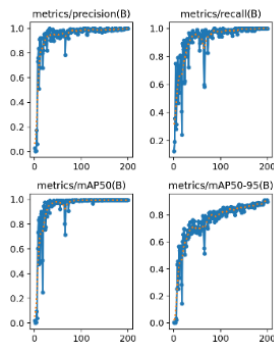
Tahap *evaluation* dilakukan untuk mengukur performa model YOLOv8n melalui pengamatan terhadap metrik-metrik pelatihan, seperti *loss function*, *recall*, *precision*, *mAP*, dan *confusion matrix*, yang dihasilkan selama 200 *epoch*.

Berdasarkan grafik pada Gambar 8, seluruh *loss function* yang terdiri dari *box loss*, *classification loss*, dan *DLF loss* menunjukkan pola penurunan yang konsisten baik pada data *training* dan data *validation*. Penurunan ini menandakan bahwa model semakin akurat dalam melakukan prediksi pada posisi *bounding box*. Penurunan tren yang stabil juga menunjukkan bahwa selama proses pelatihan berlangsung secara efektif.



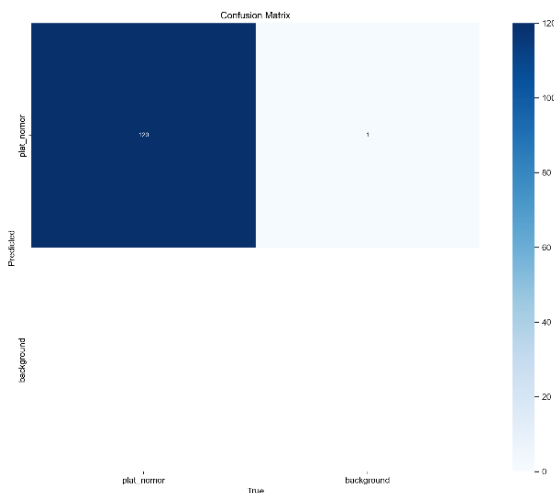
Gambar 8. Grafik Loss Function

Selanjutnya, pada grafik pada Gambar 9 menampilkan tren peningkatan pada metrik *precision*, *recall*, dan *mAP* yang meningkat tajam di awal pada pelatihan dan stabil pada nilai yang mendekati 1.0. Peningkatan yang konsisten ini menunjukkan model mampu melakukan deteksi dengan akurasi optimal. Secara keseluruhan kedua grafik tersebut memperlihatkan bahwa model YOLOv8n menunjukkan performa yang sangat baik dan stabil sepanjang proses pelatihan.



Gambar 9. Grafik Recall, Precision, dan mAP

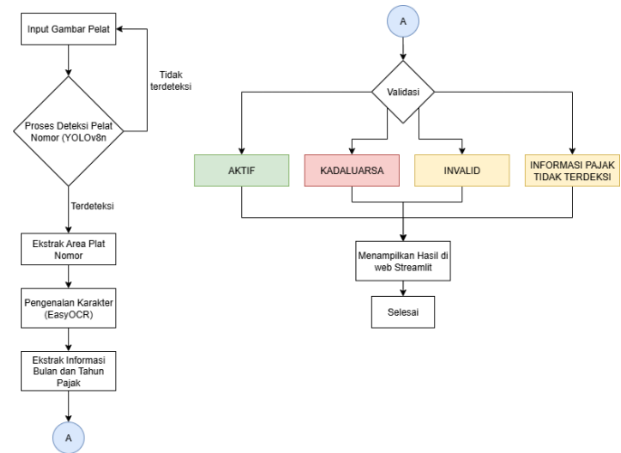
Gambar 10 menampilkan *confusion matrix* hasil prediksi model pada data *validation*. Model berhasil mendeteksi kelas 'plat_nomor' dengan sangat baik, ditunjukkan oleh 120 *True Positive* dan terdapat satu *False Positive*. Tidak ditemukan *False Negative*, sehingga model mampu mengenali seluruh objek pelat nomor yang ada.



Gambar 10. Confusion Matrix

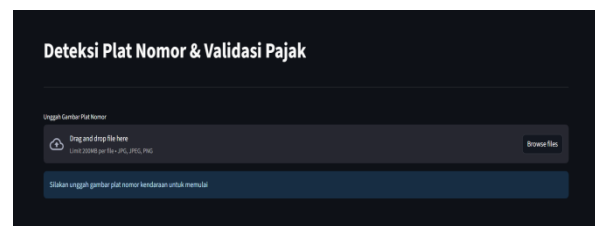
3.6. Deployment

Tahap *deployment* menjelaskan proses implementasi model YOLOv8 dan EasyOCR ke dalam aplikasi web berbasis Streamlit. Gambar 11 menunjukkan alur kerja sistem dimulai dari *input* gambar hingga penentuan status pajak.



Gambar 11. Alur Kerja Sistem

Diagram pada Gambar 11 menggambarkan alur kerja sistem pada tahap *deployment*. Proses dimulai dari *input* gambar kendaraan yang kemudian diproses oleh model YOLOv8 untuk mendeteksi area pelat nomor. Hasil deteksi otomatis dipotong dan dikirim ke modul EasyOCR untuk dilakukan pengenalan karakter. Teks hasil pembacaan kemudian diekstraksi untuk memperoleh informasi bulan dan tahun pajak. Berdasarkan informasi tersebut dilakukan proses validasi untuk menentukan status pajak, status pajak terbagi menjadi empat kategori: aktif, kadaluarsa, invalid, dan informasi pajak tidak terdeteksi. Hasil akhir ditampilkan melalui antarmuka web Streamlit.



Gambar 12. Tampilan web Streamlit

Gambar 12 merupakan tampilan awal web *streamlit* yang menyediakan fitur unggah gambar pelat nomor sebagai langkah awal proses deteksi dan validasi pajak.



Gambar 13. Tampilan Hasil Deteksi Pelat Nomor dan Validasi Pajak Pada Aplikasi Streamlit

Gambar 13 merupakan hasil deteksi sistem berupa area pelat nomor, hasil pembacaan OCR, serta informasi bulan, tahun, dan status pajak kendaraan. Contoh pada Gambar 13, sistem berhasil mendeteksi pelat nomor dan menentukan bahwa pajak kendaraan berstatus aktif.

3.7. Hasil Pengujian Sistem

Pada tahap ini dilakukan pengujian sistem secara langsung untuk memastikan bahwa proses deteksi pelat nomor, pembacaan teks, dan validasi status pajak berjalan sesuai rancangan. Pengujian dilakukan dengan menggunakan beberapa data primer. Hasil pengujian dilakukan dengan menampilkan antarmuka *Streamlit* dan menunjukkan bahwa sistem mampu menghasilkan deteksi pelat nomor yang akurat. Hasil pengujian dapat dilihat pada Tabel 5.

Tabel 5. Hasil Uji Coba Pada Antarmuka Streamlit

Hasil Deteksi dan Validasi	Keterangan
	<p>Pengujian pertama dengan menggunakan pelat berwarna hijau. Model berhasil mendeteksi area pelat dan OCR mampu membaca teks dengan baik, meskipun terjadi kesalahan pada huruf 'O'. Informasi bulan dan tahun pajak dapat diekstrak dengan benar, sehingga sistem mengklasifikasi status pajak aktif.</p>
	<p>Pengujian dilakukan pada malam hari, sistem berhasil mendeteksi area pelat nomor, namun OCR tidak mampu mengenali karakter dengan dan menghasilkan teks acak. Informasi pajak tidak dapat diekstrak sehingga sistem gagal</p>

melakukan proses validasi



Uji coba dilakukan di atas jembatan penyebrangan dengan model berhasil mendeteksi area pelat nomor, namun OCR tidak berhasil membaca karakter dengan benar karena ukuran pelat yang kecil pada pelat. Sehingga, informasi bulan dan tahun pajak tidak dapat diekstraksi.



Pada uji coba ini, sistem diberikan gambar kucing yang termasuk non-pelat. Sistem tidak melakukan deteksi dan menampilkan pesan 'Tidak terdapat pelat nomor terdeteksi pada gambar'.



Uji coba dilakukan dengan menggunakan pelat khusus TNI, sistem masih dapat mendeteksi area pelat, sementara bulan dan tahun tidak dikenali. Karena pelat TNI tidak membuat informasi bulan dan tahun, sistem menampilkan *output* 'Informasi pajak tidak terdeteksi pada pelat nomor'.

4. Kesimpulan

Penelitian ini telah berhasil mewujudkan sistem deteksi pelat nomor kendaraan dan validasi pajak berbasis web dengan mengintegrasikan YOLOv8 sebagai model pendeteksi dan EasyOCR sebagai sistem pengenalan karakter. Sistem mampu mengidentifikasi pelat nomor secara otomatis, mengekstraksi informasi bulan dan tahun masa berlaku pajak, serta menyajikan hasil validasi melalui antarmuka Streamlit. Dengan konfigurasi pelatihan terbaik berupa 200 epoch, batch size 16, dan learning rate 0,01, model menghasilkan performa yang stabil dengan nilai *box loss* sebesar 0,38.

Kinerja OCR masih dipengaruhi kualitas citra pelat nomor, khususnya pada kondisi pencahayaan yang kurang baik, gambar buram, atau karakter yang berdekatan. Selain itu, mekanisme validasi pajak masih bersifat visual dan belum terhubung dengan basis data resmi kendaraan seperti SAMSAT, sehingga tingkat keandalannya masih terbatas.

Penelitian selanjutnya dapat difokuskan pada penerapan OCR per karakter untuk meningkatkan akurasi pembacaan, serta pengembangan fitur deteksi berbasis video secara *real-time* pada antarmuka web

agar sistem lebih aplikatif di lapangan.

Daftar Rujukan

- [1] A. Novelino, Jumlah Kendaraan di Indonesia Tembus 164 Juta Unit, 83 Persen Motor. Accessed: Nov. 25, 2025. [Online]. Available: <https://www.cnnindonesia.com/otomotif/20241004133318-579-1151516/jumlah-kendaraan-di-indonesia-tembus-164-juta-unit-83-persen-motor>
- [2] J. Subur, Suryadhi, M. Taufiqurrohmah, and N. Reza Al Hafizh, 2024. Pemanfaatan Teknologi Computer Vision untuk Deteksi Ukuran Ikan Bandeng dalam Membantu Proses Sortir Ikan. *CYCLOTRON*, vol. 7, no. 01, pp. 52–60, Jan. 2024, doi: 10.30651/cl.v7i01.21239.
- [3] P. Y. Putra, A. S. Arifianto, Z. E. Fitri, and T. D. Puspitasari, 2023. Deteksi Kendaraan Truk pada Video Menggunakan Metode Tiny-YOLO v4. *Jurnal Informatika Polinema*, vol. 9, no. 2, pp. 215–222. doi: 10.33795/jip.v9i2.1243.
- [4] L. Satya, M. R. D. Septian, M. W. Sarjono, M. Cahyanti, and E. R. Swedia, 2023. SISTEM PENDETEKSI PLAT NOMOR POLISI KENDARAAN DENGAN ARSITEKTUR YOLOV8. *Sebatik*, vol. 27, no. 2, pp. 753–761. doi: 10.46984/sebatik.v27i2.2374.
- [5] A. Meirza and N. R. Puteri, 2024. Implementasi Metode YOLOV5 dan Tesseract OCR untuk Deteksi Plat Nomor Kendaraan. *Jurnal Ilmu Komputer dan Desain Komunikasi Visual*, vol. 9, no. 1. Available: <https://journal.unusida.ac.id/index.php/jik/en/article/view/1288>
- [6] I. Budiman *et al.*, 2011. Data Clustering Menggunakan Metodologi CRISP-DM Untuk Pengenalan Pola Proporsi Pelaksanaan Tridharma. Available: <https://journal.uui.ac.id/Snati/article/view/2901>
- [7] M. N. Alwi, 2024. CRISP-DM: Tahapan, Studi Kasus, Kelebihan, dan Kekurangan. Accessed: Nov. 25, 2025. [Online]. Available: <https://www.dicoding.com/blog/crisp-dm-tahapan-studi-kasus-kelebihan-dan-kekurangan/>
- [8] M. A. Hasanah, S. Soim, and A. S. Handayani, 2021. Implementasi CRISP-DM Model Menggunakan Metode Decision Tree dengan Algoritma CART untuk Prediksi Curah Hujan Berpotensi Banjir. *Journal of Applied Informatics and Computing*, vol. 5, no. 2, pp. 103–108. doi: 10.30871/jaic.v5i2.3200.
- [9] S. S. Hartinah and Sugiyono, 2024. Pemodelan Data Mining Transaksi Penjualan Menggunakan Algoritma Apriori (Studi Kasus: Kedai Ngodeng & Smoothies). *Jurnal Indonesia : Manajemen Informatika dan Komunikasi*, vol. 5, no. 3, pp. 3080–3098. doi: 10.35870/jimik.v5i3.992.
- [10] F. Abdusyukur, 2023. PENERAPAN ALGORITMA SUPPORT VECTOR MACHINE (SVM) UNTUK KLASIFIKASI PENCEMARAN NAMA BAIK DI MEDIA SOSIAL TWITTER. *Komputa : Jurnal Ilmiah Komputer dan Informatika*, vol. 12, no. 1, pp. 73–82. doi: 10.34010/komputa.v12i1.9418.