

Pengenalan Plat Nomor Kendaraan Menggunakan Deep Feature Extraction dan Faster R-CNN

AHMAD HILMY RAKHA ALCANDER¹, ELLY MATUL IMAH^{2,*}

¹Program Studi Sarjana Sains Data, Universitas Negeri Surabaya, Indonesia

²Program Studi Sarjana Kecerdasan Artifisial, Universitas Negeri Surabaya, Indonesia

Abstrak

Automatic License Plate Recognition (ALPR) merupakan teknologi berbasis visi komputer yang memungkinkan identifikasi plat nomor kendaraan secara otomatis dan telah banyak diterapkan pada sistem transportasi cerdas, manajemen parkir, serta penegakan hukum. Penelitian ini mengembangkan sistem ALPR dua tahap menggunakan Faster R-CNN dengan backbone ResNet-50-FPN untuk mendeteksi lokasi plat nomor, dilanjutkan dengan *Convolutional Neural Network* (CNN) berbasis segmentasi karakter untuk mengenali teks plat nomor. Dataset yang digunakan adalah *Indonesian License Plate Dataset* yang terdiri dari 1.000 citra kendaraan dengan anotasi *bounding box* dan 14.074 citra karakter hasil segmentasi. Model Faster R-CNN dievaluasi menggunakan ROC-AUC, Equal Error Rate (EER), dan Normalized Mean Distance (NMD) sebagai metrik pelengkap untuk mengevaluasi kemampuan diskriminasi, karakteristik kesalahan keputusan, dan ketepatan lokalisasi, sedangkan model CNN OCR dievaluasi menggunakan *Accuracy*, *Precision*, *Recall*, *F1-Score*, *Character Error Rate* (CER), *Word Accuracy*, *Mean Plate Accuracy*, dan *Overall Character Accuracy*. Hasil evaluasi menunjukkan bahwa Faster R-CNN memperoleh ROC-AUC sebesar 0,8669, EER sebesar 0,2249, dan NMD sebesar 0,1915, sedangkan CNN OCR memperoleh *accuracy* 93%, *Mean Plate Accuracy* 92,62%, *Overall Character Accuracy* 92,78%, *Word Accuracy* 69,54%, dan CER 7,22%. Temuan ini menunjukkan bahwa integrasi Faster R-CNN ResNet-50-FPN dan CNN OCR mampu membentuk sistem ALPR yang efektif untuk plat nomor kendaraan Indonesia.

Kata Kunci ALPR; CNN; Faster R-CNN; OCR; ResNet-50-FPN

Abstract

Automatic License Plate Recognition (ALPR) is a computer vision technology that enables automatic identification of vehicle license plates and has been widely deployed in intelligent transportation systems, parking management, and law enforcement applications. This study develops a two-stage ALPR system using Faster R-CNN with a ResNet-50-FPN backbone for license plate localization, followed by a segmentation-based Convolutional Neural Network (CNN) for character recognition. The dataset used is the Indonesian License Plate Dataset which consists of 1,000 vehicle images with bounding box annotations and 14,074 segmented character images. The Faster R-CNN model is evaluated using ROC-AUC, Equal Error Rate (EER), and Normalized Mean Distance (NMD) as complementary metrics to evaluate discrimination capability, decision error characteristics, and localization accuracy, respectively, while the CNN OCR model is evaluated using Accuracy, Precision, Recall, F1-Score, Character Error Rate (CER), Word

*Corresponding author. Email: ellymatul@unesa.ac.id.

Accuracy, Mean Plate Accuracy, and Overall Character Accuracy. Evaluation results show that Faster R-CNN achieved ROC-AUC of 0.8669, EER of 0.2249, and NMD of 0.1915, while the CNN OCR achieved accuracy of 93%, Mean Plate Accuracy of 92.62%, Overall Character Accuracy of 92.78%, Word Accuracy of 69.54%, and CER of 7.22%. These findings demonstrate that the integration of Faster R-CNN ResNet-50-FPN and CNN OCR constitutes an effective ALPR system for Indonesian vehicle license plates.

Keywords ALPR; CNN; Faster R-CNN; OCR; ResNet-50-FPN

1 Pendahuluan

Perkembangan teknologi informasi dan visi komputer telah membawa transformasi signifikan dalam sistem transportasi cerdas, salah satunya melalui teknologi *Automatic License Plate Recognition* (ALPR). ALPR merupakan sistem yang memungkinkan identifikasi dan ekstraksi informasi plat nomor kendaraan secara otomatis dari citra atau video tanpa intervensi manusia (Popoola dkk., 2024). Sistem ini telah banyak diterapkan pada berbagai domain seperti manajemen parkir otomatis, penegakan hukum lalu lintas, pemantauan kendaraan berbasis kamera pengawas, dan sistem gerbang tol (Sarhan dkk., 2024; Moussaoui dkk., 2024). Di Indonesia, kewajiban pemasangan plat nomor pada kendaraan diatur dalam Undang-Undang No. 22 Tahun 2009 tentang Lalu Lintas dan Angkutan Jalan, sehingga keberadaan sistem ALPR yang andal memiliki relevansi tinggi untuk berbagai keperluan administrasi dan penegakan hukum. Secara umum, sistem ALPR terdiri dari tiga komponen utama, yaitu deteksi lokasi plat nomor, segmentasi karakter, dan pengenalan karakter menggunakan *Optical Character Recognition* (OCR) (Panahi dan Gholampour, 2017).

Pendekatan konvensional dalam ALPR mengandalkan teknik *image processing* berbasis aturan (*rule-based*) seperti *edge detection*, *morphological operations*, dan *template matching*. Metode-metode ini memiliki keterbatasan fundamental dalam hal ketahanan terhadap variasi kondisi lingkungan seperti perubahan pencahayaan, sudut pandang kamera yang bervariasi, oklusi parsial, dan degradasi kualitas citra akibat cuaca atau kecepatan kendaraan (Popoola dkk., 2024; Panahi dan Gholampour, 2017). Seiring perkembangan pesat *deep learning*, paradigma dalam pengembangan sistem ALPR telah bergeser menuju pendekatan berbasis jaringan saraf tiruan yang mampu mengekstraksi fitur secara otomatis dari data (Lamichhane dkk., 2025). *Convolutional Neural Network* (CNN) telah terbukti memberikan performa yang lebih *robust* dibandingkan metode tradisional dalam berbagai tugas visi komputer, termasuk deteksi objek dan pengenalan karakter (Rofiq Mulyana dkk., 2022; Awasthi dkk., 2024). Kemampuan *deep learning* dalam mempelajari representasi fitur yang kompleks secara *end-to-end* menjadikannya kandidat kuat untuk sistem ALPR yang adaptif terhadap kondisi dunia nyata.

Dalam domain deteksi objek, Ren dkk. (2015) memperkenalkan Faster R-CNN sebagai arsitektur *two-stage detector* yang mengintegrasikan *Region Proposal Network* (RPN) untuk menghasilkan kandidat region objek secara efisien dalam satu kerangka kerja terpadu. Dibandingkan pendahulunya R-CNN (Girshick dkk., 2014) dan Fast R-CNN (Girshick, 2015), Faster R-CNN mengatasi *bottleneck* komputasi pada tahap *region proposal* sehingga mampu melakukan deteksi secara *end-to-end* dengan akurasi yang kompetitif. Studi komparatif Carranza-García dkk. (2021) menunjukkan bahwa *two-stage detector* seperti Faster R-CNN umumnya menghasilkan akurasi deteksi yang lebih tinggi dibandingkan *one-stage detector* seperti YOLO dan SSD pada objek berukuran kecil. Untuk memperkuat kemampuan deteksi multiskala, ResNet-50 dikombinasikan dengan *Feature Pyramid Network* (FPN) yang membangun hierarki fitur berhirarki

melalui mekanisme *top-down pathway* dan *lateral connections* (Lin dkk., 2017; Chen dkk., 2023). Kombinasi ResNet-50-FPN ini secara signifikan meningkatkan sensitivitas model terhadap objek berukuran kecil seperti plat nomor, yang sering muncul dengan berbagai skala dalam kondisi pengambilan gambar yang bervariasi (Abdelrahman dan Viriri, 2023; Renjun dkk., 2022).

Meskipun penelitian ALPR telah berkembang pesat secara global, terdapat beberapa *research gap* yang belum terjawab, khususnya pada konteks plat nomor Indonesia. Pertama, evaluasi sistem ALPR berbasis Faster R-CNN pada dataset plat nomor Indonesia masih sangat terbatas, sehingga tidak ada acuan performa yang jelas untuk konteks lokal. Kedua, sebagian besar penelitian ALPR menggunakan metrik evaluasi yang tidak lengkap; khususnya, penggunaan gabungan metrik ROC-AUC, EER, dan NMD untuk tahap deteksi secara bersamaan masih jarang dilakukan (Roh, 2017). Ketiga, evaluasi OCR yang komprehensif mencakup CER, *Word Accuracy*, *Mean Plate Accuracy*, dan *Overall Character Accuracy* secara simultan juga belum banyak dilaporkan pada sistem ALPR berbasis CNN untuk plat nomor Indonesia (Vesalainen dkk., 2026; Gaikwad, 2026). Keempat, mayoritas penelitian terdahulu tidak membahas secara eksplisit tantangan karakter yang memiliki kemiripan bentuk visual seperti I-1, O-0, dan Q-O dalam konteks plat nomor Indonesia. Kelima, integrasi sistem deteksi dan OCR secara *end-to-end* dengan evaluasi menyeluruh pada kedua tahap masih sangat terbatas dalam literatur yang ada.

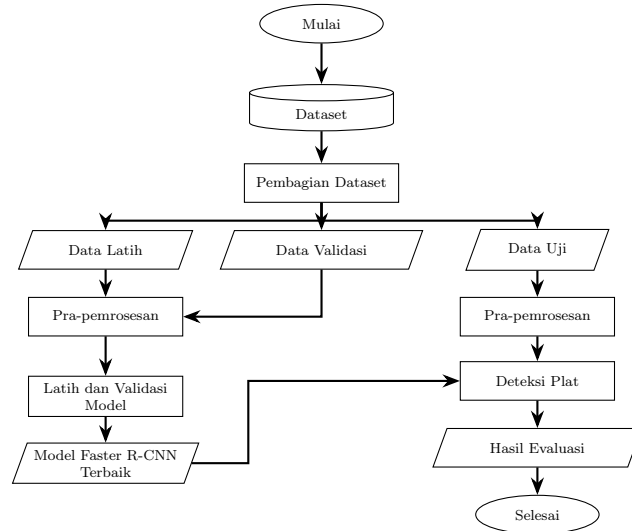
Berdasarkan *research gap* tersebut, penelitian ini memiliki tujuan sebagai berikut: (1) menganalisis performa Faster R-CNN dengan *deep feature extraction* ResNet-50-FPN dalam mendeteksi berdasarkan kemampuan diskriminasi dan karakteristik kesalahan keputusan serta ketepatan lokalisasi menggunakan metrik ROC-AUC, EER, dan NMD; (2) menganalisis performa OCR berbasis CNN dalam mengenali karakter plat nomor menggunakan metrik *Accuracy*, *Precision*, *Recall*, *F1-Score*, *Overall Character Accuracy*, *Word Accuracy*, *Mean Plate Accuracy*, dan CER; serta (3) mengevaluasi hasil integrasi kedua model secara *end-to-end* pada sistem ALPR. Penelitian ini memberikan manfaat berupa: (a) kontribusi literatur ilmiah mengenai sistem ALPR untuk plat nomor kendaraan Indonesia; (b) kerangka evaluasi komprehensif yang dapat diadopsi peneliti dan praktisi dalam mengembangkan sistem ALPR; serta (c) acuan performa sistem berbasis *deep learning* untuk aplikasi transportasi cerdas, manajemen parkir, dan penegakan hukum di Indonesia.

2 Metode

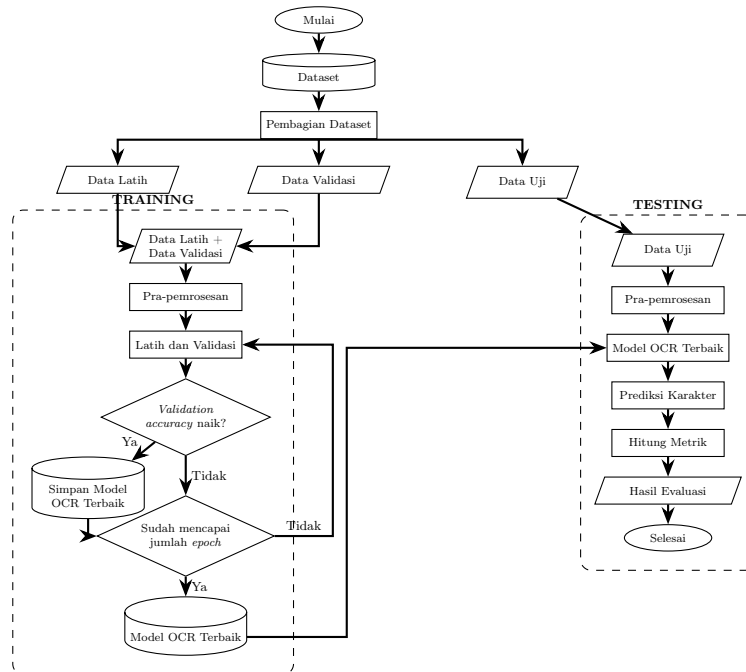
2.1 Rancangan Penelitian

Penelitian ini mengembangkan sistem ALPR dua tahap yang dirancang untuk melakukan identifikasi plat nomor kendaraan secara otomatis dari masukan citra kendaraan hingga keluaran berupa teks plat nomor. Rancangan penelitian terdiri dari dua alur utama yang ditunjukkan pada Gambar 1 dan Gambar 2. Tahap pertama merupakan proses lokalisasi plat nomor menggunakan Faster R-CNN. Alur ini dimulai dari dataset citra, kemudian dilakukan pembagian data menjadi data latih, data validasi, dan data uji. Data latih dan data validasi digunakan pada tahap pra-pemrosesan serta proses latih dan validasi model hingga diperoleh model Faster R-CNN terbaik. Selanjutnya, data uji melalui pra-pemrosesan, dideteksi menggunakan model terbaik tersebut, dan menghasilkan hasil evaluasi deteksi plat nomor. Tahap kedua merupakan proses OCR berbasis CNN. Alur ini dimulai dari pemuatan citra plat, pembentukan dataset, ekstraksi karakter, dan pembagian dataset menjadi data latih, data validasi, serta data uji. Data latih dan validasi digabungkan sebagai masukan proses pelatihan, kemudian melalui pra-pemrosesan, latih

dan validasi, pengecekan peningkatan *validation accuracy*, penyimpanan model OCR terbaik, serta pengecekan jumlah *epoch*. Setelah model OCR terbaik diperoleh, data uji dipra-pemroses, diprediksi karakternya, dihitung metrik evaluasinya, dan menghasilkan hasil evaluasi OCR. Kedua tahap tersebut saling terintegrasi dalam satu alur kerja sistem ALPR yang kohesif.



Gambar 1: Diagram alir tahap lokalisasi plat nomor kendaraan menggunakan Faster R-CNN.



Gambar 2: Diagram alir tahap OCR untuk pengenalan karakter plat nomor kendaraan.

2.2 Dataset

Penelitian ini menggunakan *Indonesian License Plate Dataset* (Wijaya, 2025) yang tersedia pada platform *Kaggle* sebagai sumber data utama. Dataset ini dipilih karena secara khusus memuat plat nomor kendaraan Indonesia sehingga karakteristik bentuk, ukuran, susunan karakter, dan format plat nomor sesuai dengan kebutuhan penelitian. Dataset terdiri dari citra kendaraan dengan anotasi lokasi plat nomor dalam format YOLO yang mencakup berbagai variasi kondisi seperti perbedaan pencahayaan, sudut pengambilan citra, posisi kendaraan, jarak kamera, dan keberadaan *noise* pada citra. Pada tahap deteksi, digunakan 1.000 citra kendaraan dengan anotasi *bounding box*, sedangkan pada tahap OCR digunakan 14.074 citra karakter yang diperoleh melalui segmentasi berdasarkan anotasi karakter dari dataset yang sama. Pembagian dataset dilakukan dengan rasio 80% data latih, 10% data validasi, dan 10% data uji untuk memastikan proses pelatihan, pemantauan, dan evaluasi dilakukan pada subset data yang berbeda dan independen.

Contoh data yang digunakan ditunjukkan pada Gambar 3. Panel (a) memperlihatkan citra kendaraan pada tahap deteksi beserta lokasi plat nomor, sedangkan panel (b) memperlihatkan citra plat hasil *cropping* yang selanjutnya menjadi masukan tahap segmentasi dan pengenalan karakter. Contoh tersebut menggambarkan hubungan antara dua bentuk data dalam penelitian, yaitu citra kendaraan beranotasi untuk pelatihan Faster R-CNN dan citra plat atau karakter hasil ekstraksi untuk pelatihan serta pengujian CNN OCR.

(a) Citra kendaraan dan lokasi plat



(b) Citra plat hasil cropping



Gambar 3: Contoh dataset yang digunakan: (a) citra kendaraan beserta lokasi plat nomor dan (b) citra plat hasil *cropping* sebagai sumber ekstraksi karakter OCR.

2.3 Lokalisasi Plat Nomor Menggunakan Faster R-CNN

2.3.1 Pra-pemrosesan

Tahap pra-pemrosesan dilakukan untuk memastikan seluruh data memiliki format yang konsisten dan siap digunakan pada proses pelatihan model. Citra kendaraan dibaca menggunakan *Python Imaging Library* (PIL) dan dikonversi ke format *RGB* untuk memastikan konsistensi tiga *channel* warna pada seluruh citra, mengingat terdapat kemungkinan citra dalam format *grayscale* atau *RGBA* dalam dataset. Anotasi format *YOLO* yang berisi nilai $(x_{center}, y_{center}, w, h)$ dalam bentuk normalisasi dikonversi menjadi format $(x_{min}, y_{min}, x_{max}, y_{max})$ yang digunakan Faster R-CNN.

Seluruh citra kemudian diubah ukurannya menjadi 512×512 piksel dan dikonversi menjadi tensor PyTorch melalui `ToTensor()`, sehingga rentang nilai piksel berubah dari $[0, 255]$ menjadi $[0, 1]$ untuk menjaga stabilitas numerik selama pelatihan. Khusus pada data latih, diterapkan augmentasi data berupa *Random Horizontal Flip* dengan probabilitas 50% dan *Color Jitter* untuk meningkatkan keberagaman data dan mengurangi risiko *overfitting* (Renjun dkk., 2022), sedangkan data validasi dan uji hanya melalui proses *resize* dan konversi tensor agar evaluasi merepresentasikan kondisi data asli.

2.3.2 Deep Feature Extraction

Deep feature extraction adalah proses pembentukan representasi fitur secara otomatis melalui rangkaian lapisan konvolusi bertingkat. Berbeda dengan fitur buatan tangan seperti tepi, warna, atau tekstur yang dirancang terlebih dahulu, fitur pada jaringan dalam dipelajari langsung dari data berdasarkan fungsi objektif deteksi. Pada lapisan awal, model merespons pola visual sederhana seperti garis, sudut, kontras, dan tekstur. Semakin dalam lapisan jaringan, fitur tersebut dikombinasikan menjadi representasi yang lebih kompleks dan semantik, seperti bentuk persegi panjang plat, susunan karakter, serta perbedaan antara area plat dan latar belakang kendaraan. Secara umum, transformasi fitur pada tingkat ke- l dinyatakan sebagai

$$C_l = \phi_l(C_{l-1}; \theta_l), \quad (1)$$

dengan C_{l-1} sebagai peta fitur masukan, ϕ_l sebagai rangkaian operasi konvolusi, normalisasi, aktivasi, dan blok residual, serta θ_l sebagai parameter yang dipelajari selama pelatihan.

Dalam penelitian ini, proses *deep feature extraction* dilakukan oleh *backbone* ResNet-50 yang telah memiliki bobot awal *pretrained* COCO. Bobot tersebut menyediakan representasi visual umum sebagai titik awal, kemudian disesuaikan terhadap karakteristik plat nomor Indonesia selama pelatihan. Keluaran blok ResNet-50 pada beberapa kedalaman, yaitu C_2 – C_5 , memiliki resolusi spasial dan tingkat semantik yang berbeda. FPN menggabungkan keluaran tersebut melalui *lateral connection* dan jalur *top-down* untuk membentuk peta fitur multiskala P_2 – P_5 (Lin dkk., 2017). Proses penggabungannya dapat diringkas sebagai

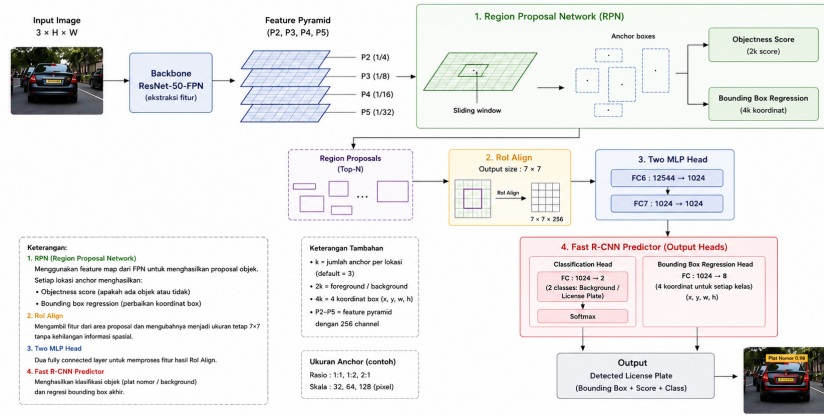
$$P_l = \text{Conv}_{3 \times 3}(\text{Conv}_{1 \times 1}(C_l) + \text{Up}(P_{l+1})), \quad (2)$$

dengan $\text{Up}(\cdot)$ sebagai operasi *upsampling*. Peta P_2 yang beresolusi lebih tinggi mempertahankan detail spasial objek kecil, sedangkan peta pada tingkat yang lebih dalam membawa informasi semantik dan konteks yang lebih kuat. Kombinasi ini penting karena ukuran plat nomor pada citra kendaraan dapat bervariasi akibat jarak dan sudut pengambilan gambar. Seluruh peta fitur multiskala kemudian digunakan oleh RPN untuk menghasilkan kandidat wilayah dan oleh *MultiScale RoI Align* untuk mengekstraksi fitur setiap kandidat sebelum dilakukan klasifikasi dan regresi *bounding box*. Dengan demikian, istilah *deep feature extraction* pada penelitian ini merujuk secara spesifik pada ekstraksi fitur hierarkis ResNet-50 dan fusi fitur multiskala FPN, bukan tahap terpisah yang menggunakan deskriptor visual manual.

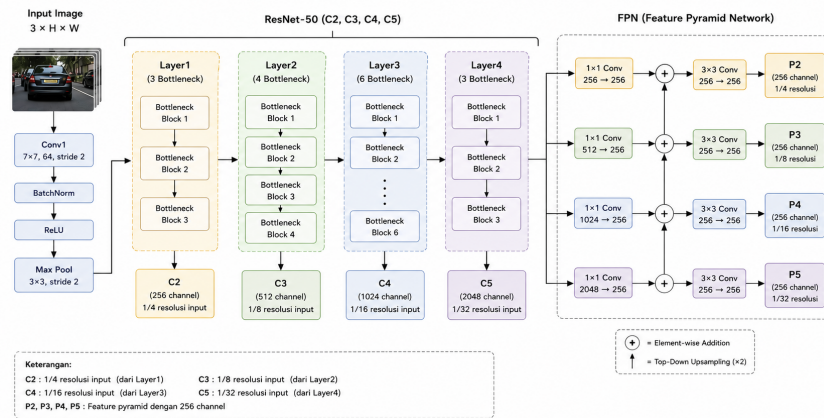
2.3.3 Arsitektur Faster R-CNN

Model Faster R-CNN yang digunakan merupakan *pretrained* dengan bobot awal dari dataset COCO untuk mempercepat konvergensi dan memberikan representasi fitur awal yang lebih baik. Arsitektur Faster R-CNN secara keseluruhan ditunjukkan pada Gambar 4, yang memperlihatkan

alur pemrosesan dari masukan citra melalui jaringan konvolusional (*Conv. Net.*) dan RPN, kemudian ke *ROI Pooling & FCs*, hingga menghasilkan prediksi posisi (*Positions*) dan skor kelas (*Scores*) melalui *Regressor* dan *Classifier* yang masing-masing merupakan *fully connected layer* (Ren dkk., 2015). Komponen *GeneralizedRCNNTransform* bertugas melakukan normalisasi citra menggunakan *mean* [0.485, 0.456, 0.406] dan *standard deviation* [0.229, 0.224, 0.225] sebagai parameter standar *pretrained COCO*. *Backbone* ResNet-50 mengekstraksi fitur visual melalui empat tahap *residual block* (layer1–layer4) yang menghasilkan peta fitur dengan tingkat representasi berbeda, mulai dari fitur tingkat rendah seperti tepi dan tekstur hingga fitur tingkat tinggi yang bersifat semantik. Selanjutnya, FPN memproses peta fitur dari *backbone* untuk menghasilkan representasi multiskala dengan 256 *channel* pada setiap level piramida (P2–P6), diikuti oleh RPN yang menghasilkan kandidat *bounding box*, *MultiScale RoI Align* berukuran 7×7 , *TwoMLPHead* dengan dua *fully connected layer* (1024, 1024 neuron), dan *FastRCNNPredictor* dengan dua kelas luaran (*background* dan plat nomor).



Gambar 4: Arsitektur Faster R-CNN dengan alur *two-stage detection* dari ekstraksi fitur konvolusional, RPN, ROI Pooling, hingga prediksi *bounding box* dan kelas.



Gambar 5: Arsitektur ResNet-50-FPN yang menggabungkan jalur *bottom-up* ResNet-50 dan jalur *top-down* FPN untuk menghasilkan representasi fitur multiskala P2–P5.

Prinsip utama ResNet-50 adalah *residual learning* (Renjun dkk., 2022; Li dan Lima, 2021), di mana jaringan tidak langsung mempelajari fungsi pemetaan target $H(x)$, melainkan mempelajari fungsi residual $F(x)$ sehingga output blok dinyatakan pada Persamaan 3:

$$H(x) = F(x) + x \quad (3)$$

dengan $H(x)$ sebagai output blok residual, $F(x)$ sebagai fungsi residual yang dipelajari, dan x sebagai input yang dilewatkan melalui *identity mapping*. Pendekatan ini secara signifikan mengurangi kesulitan optimisasi pada jaringan yang sangat dalam dan mengatasi masalah degradasi performa (Zhao dan Leong, 2025). Struktur arsitektur model Faster R-CNN yang diimplementasikan ditunjukkan pada Tabel 1.

Tabel 1: Arsitektur model Faster R-CNN yang diimplementasikan pada penelitian ini.

Komponen	Layer	Fungsi
Input	Input	Citra kendaraan $512 \times 512 \times 3$
Transform	GeneralizedRCNNTransform	Normalisasi dan <i>resize</i> citra
Backbone	ResNet-50	Ekstraksi fitur visual
FPN	Feature Pyramid Network	Representasi fitur multiskala
RPN	Region Proposal Network	Kandidat area objek
RoI Pooling	MultiScale RoI Align (7×7)	Ekstraksi fitur dari <i>region proposal</i>
Box Head	TwoMLPHead (1024, 1024)	Representasi fitur tingkat tinggi
Box Predictor	FastRCNNPredictor	Klasifikasi dan regresi <i>bounding box</i>
Output	Kotak Pembatas + <i>Confidence Score</i>	Hasil deteksi plat nomor

2.3.4 Pelatihan Model Faster R-CNN

Proses pelatihan dilakukan menggunakan optimizer Adam pada *framework* PyTorch dengan dukungan GPU NVIDIA GeForce RTX 3050. Optimizer Adam dipilih karena kemampuannya melakukan penyesuaian *learning rate* secara adaptif sehingga proses pelatihan lebih stabil dan konvergensi berlangsung lebih cepat (Ciampiconi dkk., 2024). Fungsi kehilangan multi-tugas Faster R-CNN mengombinasikan *classification loss* dan *regression loss* secara simultan sebagaimana Persamaan 4 (Ren dkk., 2015):

$$L(\{p_i\}, \{t_i\}) = \frac{1}{N_{\text{cls}}} \sum_i L_{\text{cls}}(p_i, p_i^*) + \lambda \frac{1}{N_{\text{reg}}} \sum_i p_i^* L_{\text{reg}}(t_i, t_i^*) \quad (4)$$

dengan p_i sebagai probabilitas *anchor* ke- i mengandung objek, p_i^* sebagai label *ground truth*, t_i sebagai vektor regresi *bounding box* prediksi, dan t_i^* sebagai target *ground truth*. Komponen L_{cls} menggunakan *log loss* untuk membedakan objek dan latar belakang, sedangkan L_{reg} menggunakan *Smooth L1 Loss* untuk mengukur selisih antara *bounding box* prediksi dan *ground truth*. Faktor λ mengontrol keseimbangan antara kedua komponen *loss*, sementara N_{cls} dan N_{reg} berfungsi sebagai faktor normalisasi. Model dengan performa validasi terbaik disimpan sebagai *best model* untuk digunakan pada tahap evaluasi akhir.

2.3.5 Metrik Evaluasi Faster R-CNN

Pada penelitian ini, **ROC-AUC**, **EER**, dan **NMD** digunakan sebagai metrik pelengkap untuk mengevaluasi aspek performa detektor yang berbeda. ROC-AUC dan EER digunakan untuk menganalisis karakteristik *confidence score* hasil deteksi, sedangkan NMD digunakan untuk

mengevaluasi ketepatan spasial *bounding box*. **ROC-AUC** digunakan untuk mengukur kemampuan model dalam membedakan deteksi plat nomor dan non-plat pada berbagai nilai *threshold*. **EER** digunakan untuk menganalisis keseimbangan antara *false acceptance* dan *false rejection*. Sementara itu, **NMD** digunakan untuk mengukur jarak antara *bounding box* prediksi dan *ground truth* yang dinormalisasi berdasarkan ukuran objek. Evaluasi menggunakan tiga metrik yang dipilih berdasarkan kerangka evaluasi Roh (2017). Pertama, **ROC-AUC** mengukur kemampuan model membedakan kelas positif (plat nomor) dan negatif (non-plat nomor) pada berbagai nilai *threshold* secara *threshold-independent* (Fawcett, 2006), dihitung melalui Persamaan 5:

$$\text{TPR} = \frac{TP}{TP + FN}, \quad \text{FPR} = \frac{FP}{FP + TN} \quad (5)$$

Kedua, **EER** menentukan titik keseimbangan antara *False Acceptance Rate* (FAR) dan *False Rejection Rate* (FRR) sebagaimana Persamaan 6 (Dzakwanul Fikri dkk., 2017):

$$\text{FAR} = \frac{FP}{FP + TN}, \quad \text{FRR} = \frac{FN}{TP + FN} \quad (6)$$

Ketiga, **NMD** mengukur ketepatan lokasi *bounding box* prediksi terhadap *ground truth* dengan normalisasi terhadap ukuran objek, diformulasikan pada Persamaan 7 (Roh, 2017; Zheng dkk., 2019):

$$\text{NMD} = \frac{1}{N} \sum_{i=1}^N \frac{d_i}{\max(W_i, H_i)} \quad (7)$$

dengan d_i sebagai jarak Euclidean antara pusat *bounding box* prediksi dan *ground truth*, serta $\max(W_i, H_i)$ sebagai faktor normalisasi berdasarkan ukuran objek. Normalisasi ini mencegah bias terhadap objek berukuran besar dan menjaga sensitivitas metrik terhadap kesalahan lokasi pada objek kecil (Yang dkk., 2023; Yao dkk., 2024).

2.4 OCR Berbasis CNN

2.4.1 Pra-pemrosesan Dataset OCR

Tahap pra-pemrosesan dataset OCR dilakukan untuk menyiapkan citra karakter sebelum digunakan pada proses pelatihan model CNN. Dataset karakter diperoleh dengan melakukan *cropping* pada citra kendaraan berdasarkan anotasi koordinat karakter yang tersedia dalam dataset, sehingga setiap citra karakter sudah terisolasi dari latar belakang plat nomor. Setiap citra karakter kemudian dikonversi menjadi *grayscale* satu *channel* untuk menyederhanakan representasi dan mengurangi kompleksitas komputasi, mengingat pengenalan karakter lebih bergantung pada bentuk dan pola dibandingkan informasi warna. Citra diubah ukurannya menjadi 64×64 piksel untuk menyeragamkan dimensi input model, lalu dikonversi menjadi tensor PyTorch menggunakan *ToTensor()* dan dinormalisasi dengan *mean* 0,5 dan *standard deviation* 0,5 untuk menjaga stabilitas numerik selama pelatihan (Awasthi dkk., 2024). Pada data latih, diterapkan pula augmentasi data untuk meningkatkan keberagaman sampel dan mengurangi risiko *overfitting*, sedangkan data validasi dan uji tidak diberi augmentasi agar evaluasi merepresentasikan kondisi data asli.

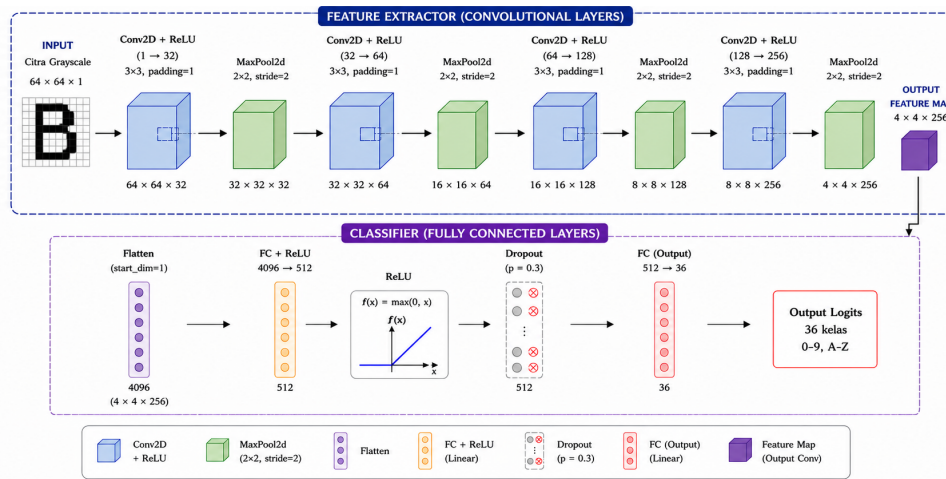
2.4.2 Segmentasi Karakter

Segmentasi karakter merupakan tahap yang dijalankan saat *inference*, yaitu ketika sistem ALPR menerima citra plat nomor baru hasil *cropping* oleh Faster R-CNN untuk kemudian dipotong per

karakter sebelum diklasifikasikan oleh CNN. Proses segmentasi dimulai dengan mengonversi citra plat ke ruang warna HSV, kemudian dibentuk *mask* area gelap untuk mengisolasi area utama plat nomor dari latar belakang. Citra selanjutnya dikonversi ke *grayscale* dan dilakukan Otsu *thresholding* pada 72% bagian atas citra agar *contour detection* berfokus pada karakter utama dan menghindari simbol tambahan seperti tanggal pada bagian bawah plat. Setiap *contour* difilter berdasarkan area, tinggi, dan rasio aspek untuk menghilangkan *noise*, kemudian diurutkan dari kiri ke kanan berdasarkan posisi horizontal agar urutan karakter sesuai dengan teks plat nomor asli. Setiap karakter hasil segmentasi selanjutnya melalui pra-pemrosesan yang sama dengan dataset training (konversi *grayscale*, *resize* 64×64 , normalisasi) sebelum diklasifikasikan oleh model CNN OCR.

2.4.3 Arsitektur CNN OCR

Model CNN OCR dirancang untuk mengklasifikasikan karakter plat nomor ke dalam 36 kelas (angka 0–9 dan huruf A–Z). Arsitektur CNN OCR secara keseluruhan ditunjukkan pada Gambar 6, yang memperlihatkan alur pemrosesan dari masukan citra karakter *grayscale* berukuran $64 \times 64 \times 1$ melalui empat blok konvolusi yang meningkat jumlah filternya secara bertahap (32, 64, 128, 256 filter), masing-masing diikuti *max pooling* 2×2 untuk mereduksi dimensi spasial. Setiap blok konvolusi menggunakan kernel 3×3 dan fungsi aktivasi ReLU, di mana jumlah filter yang meningkat bertujuan agar model dapat menangkap fitur dari tingkat sederhana (garis, tepi) hingga kompleks (bentuk, pola karakter) secara bertahap (Awasthi dkk., 2024). Setelah blok konvolusi keempat, digunakan *Adaptive Average Pooling* (1×1) yang meratakan peta fitur menjadi vektor 256 dimensi, kemudian diteruskan ke *fully connected layer* 256 neuron dengan ReLU, dan akhirnya ke *layer* output 36 neuron dengan aktivasi Softmax (Lamichhane dkk., 2025). Penggunaan *dropout* 0,3 setelah *fully connected layer* berfungsi mencegah *overfitting* selama pelatihan. Struktur lengkap arsitektur ditunjukkan pada Tabel 2.



Gambar 6: Arsitektur CNN OCR dengan empat blok konvolusi, *adaptive average pooling*, dan *fully connected layer* untuk klasifikasi 36 kelas karakter plat nomor.

Operasi konvolusi pada setiap blok dinyatakan secara matematis pada Persamaan 8 (Awasthi

dkk., 2024):

$$h_{ij}^{(k)} = f\left(\sum_m \sum_n w_{mn}^{(k)} x_{(i+m)(j+n)} + b^{(k)}\right) \quad (8)$$

Pelatihan dilakukan selama 30 *epoch* menggunakan optimizer Adam dan fungsi kehilangan *Cross Entropy Loss* yang dinyatakan pada Persamaan 9 (Janocha dan Czarnecki, 2017; Ciampiconi dkk., 2024):

$$L = - \sum_{i=1}^C y_i \log(\hat{y}_i) \quad (9)$$

dengan $C = 36$ kelas karakter, y_i sebagai label *ground truth*, dan $\hat{y}_i$ sebagai probabilitas prediksi kelas ke- i . Metrik evaluasi OCR meliputi CER pada Persamaan 10 (Vesalainen dkk., 2026), serta Word Accuracy, Mean Plate Accuracy (MPA), dan Overall Character Accuracy (OCA) pada Persamaan 11 (Gaikwad, 2026):

$$\text{CER} = \frac{S + D + I}{N} \quad (10)$$

$$\text{Word Accuracy} = \frac{N_{\text{correct}}}{N_{\text{total}}} \times 100, \quad \text{MPA} = \frac{1}{N} \sum_{i=1}^N \text{Acc}_i, \quad \text{OCA} = \frac{C_{\text{correct}}}{C_{\text{total}}} \times 100\% \quad (11)$$

Tabel 2: Arsitektur model CNN OCR dengan detail *output shape* dan jumlah parameter.

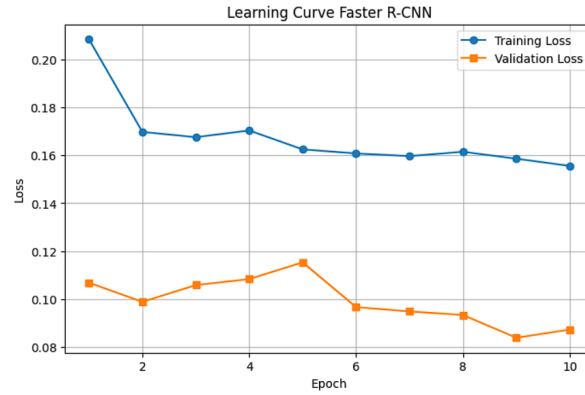
Komponen	Layer	Output Shape	Parameter
Input	Input	$64 \times 64 \times 1$	–
Conv Block 1	Conv2D (32) + ReLU + MaxPool	$32 \times 32 \times 32$	320
Conv Block 2	Conv2D (64) + ReLU + MaxPool	$16 \times 16 \times 64$	18.496
Conv Block 3	Conv2D (128) + ReLU + MaxPool	$8 \times 8 \times 128$	73.856
Conv Block 4	Conv2D (256) + ReLU + MaxPool	$4 \times 4 \times 256$	295.168
Adaptive Avg Pool	AvgPool (1×1)	$1 \times 1 \times 256$	0
Flatten	Flatten	256	0
Fully Connected	Linear (256 $\rightarrow$ 256) + ReLU	256	65.792
Dropout	Dropout (0,3)	256	0
Output	Linear (256 $\rightarrow$ 36)	36	9.252
Total Parameter			462.884

3 Hasil dan Pembahasan

3.1 Hasil Pelatihan Model Faster R-CNN

Proses pelatihan Faster R-CNN dilakukan selama 10 *epoch* menggunakan 800 citra data latih dan 100 citra data validasi dengan bobot awal *pretrained* COCO pada GPU NVIDIA GeForce RTX 3050. Gambar 7 menunjukkan *learning curve* model Faster R-CNN selama proses pelatihan, yang memvisualisasikan perubahan nilai *training loss* (biru) dan *validation loss* (oranye) pada setiap *epoch*. Nilai *training loss* menunjukkan tren penurunan yang konsisten dari sekitar 0,21 pada *epoch* pertama menjadi sekitar 0,16 pada *epoch* ke-10, yang mengindikasikan bahwa

model secara bertahap berhasil mempelajari karakteristik visual plat nomor kendaraan. *Validation loss* juga cenderung menurun dan mencapai nilai terendah sekitar 0,084 pada *epoch* ke-9, menunjukkan performa generalisasi terbaik model terhadap data yang tidak digunakan selama pelatihan. Jarak yang relatif kecil antara kurva *training loss* dan *validation loss* sepanjang proses pelatihan mengindikasikan bahwa model tidak mengalami *overfitting*, sehingga bobot model pada *epoch* ke-9 dipilih sebagai *best model* untuk digunakan pada tahap evaluasi.



Gambar 7: *Learning curve* Faster R-CNN selama 10 *epoch* yang menunjukkan penurunan konsisten *training loss* dan *validation loss* tanpa indikasi *overfitting*.

3.2 Kinerja Model Faster R-CNN

Evaluasi model Faster R-CNN dilakukan menggunakan data uji yang tidak pernah digunakan selama pelatihan, dengan hasil yang ditunjukkan pada Tabel 3. Nilai ROC-AUC sebesar 0,8669 menunjukkan bahwa model memiliki kemampuan yang baik dalam membedakan area plat nomor dan non-plat nomor pada berbagai nilai *threshold* (Fawcett, 2006); nilai ini jauh di atas 0,5 yang merepresentasikan klasifikasi acak, sehingga mengindikasikan diskriminasi model yang bermakna secara statistik. Nilai EER sebesar 0,2249 berarti nilai $(1 - \text{EER})$ sebesar 0,7751, yang secara substansial lebih baik dibandingkan Faster R-CNN standar dengan $(1 - \text{EER}) = 0,61$ yang dilaporkan Roh (2017), menunjukkan bahwa penambahan ResNet-50-FPN sebagai *backbone* memberikan peningkatan performa deteksi yang nyata. Dari sisi akurasi lokalisasi, nilai NMD sebesar 0,1915 menunjukkan bahwa pusat *bounding box* prediksi berada pada jarak rata-rata sekitar 19% dari ukuran objek terhadap *ground truth*, yang juga lebih baik dibandingkan NMD 0,40 pada *baseline* Faster R-CNN (Roh, 2017) dan NMD 0,46 khusus untuk deteksi plat nomor yang dilaporkan dalam studi yang sama. Secara keseluruhan, ketiga metrik memberikan informasi yang saling melengkapi. ROC-AUC dan EER menggambarkan karakteristik diskriminasi serta tingkat kesalahan keputusan berdasarkan *confidence score*, sedangkan NMD memberikan informasi mengenai ketepatan spasial *bounding box* terhadap *ground truth*. Dengan demikian, evaluasi tidak hanya mempertimbangkan kemampuan model dalam memberikan keputusan deteksi, tetapi juga kualitas lokalisasi plat nomor.

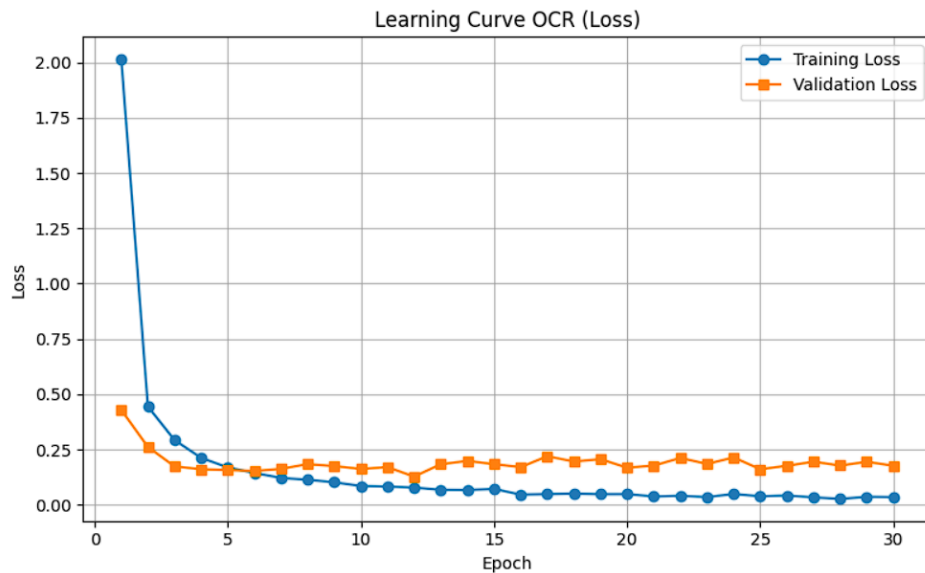
3.3 Hasil Pelatihan Model CNN OCR

Proses pelatihan model CNN OCR dilakukan selama 30 *epoch* menggunakan optimizer Adam dan *Cross Entropy Loss*. Gambar 8 menunjukkan *learning curve* OCR dari sisi *loss*, di mana

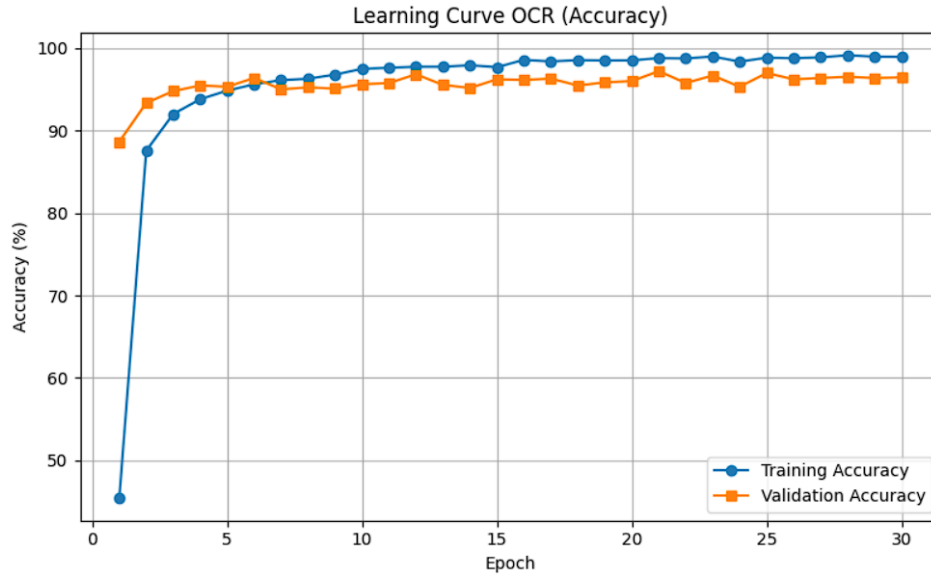
Tabel 3: Hasil metrik evaluasi model Faster R-CNN pada data uji.

Metrik Evaluasi	Hasil	Baseline (Roh, 2017)
ROC-AUC	0,8669	—
EER	0,2249	0,39
NMD	0,1915	0,40

training loss (biru) turun drastis dari 2,0155 pada *epoch* pertama menjadi 0,0349 pada *epoch* ke-30, mengindikasikan bahwa model berhasil mempelajari distribusi fitur karakter secara efektif. *Validation loss* (oranye) juga turun dari 0,4292 dan menstabilkan diri pada kisaran 0,15–0,20 setelah *epoch* ke-5, menunjukkan bahwa model mencapai kondisi konvergen secara relatif cepat berkat penggunaan *pretrained weights* dan arsitektur CNN yang sesuai. Gambar 9 menunjukkan *learning curve* dari sisi akurasi, di mana *training accuracy* meningkat secara signifikan dari 45,41% menjadi 98,93% dan *validation accuracy* meningkat dari 88,64% menjadi 96,44% pada akhir pelatihan. *Validation accuracy* tertinggi diperoleh pada *epoch* ke-21 sebesar 97,20%, sehingga bobot model pada *epoch* tersebut dipilih sebagai *best model* untuk tahap evaluasi. Jarak yang kecil antara kurva *training* dan *validation* pada kedua grafik mengindikasikan kemampuan generalisasi model yang baik terhadap data yang belum pernah dilihat sebelumnya.



Gambar 8: *Learning curve* CNN OCR selama 30 *epoch* yang memperlihatkan penurunan *training loss* dari 2,0155 menjadi 0,0349 dan stabilisasi *validation loss* pada kisaran 0,15–0,20.



Gambar 9: *Learning curve* akurasi CNN OCR selama 30 *epoch* dengan *validation accuracy* terbaik 97,20% pada *epoch* ke-21 sebagai *best model*.

3.4 Kinerja Model OCR

Hasil evaluasi OCR pada data uji ditunjukkan pada Tabel 4. Model CNN OCR memperoleh *accuracy* klasifikasi karakter sebesar 93%, *precision* (*weighted average*) sebesar 0,93, *recall* (*weighted average*) sebesar 0,93, dan *F1-score* (*weighted average*) sebesar 0,93. Selain itu, model menghasilkan *Overall Character Accuracy* sebesar 92,78%, *Mean Plate Accuracy* sebesar 92,62%, *Word Accuracy* sebesar 69,54%, dan *Character Error Rate* (CER) sebesar 7,22%. Nilai *accuracy*, *precision*, *recall*, dan *F1-score* yang sama-sama tinggi menunjukkan bahwa model memiliki performa klasifikasi karakter yang baik serta mampu mempertahankan keseimbangan antara ketepatan prediksi, kemampuan mengenali karakter yang benar, dan konsistensi kinerja pada seluruh kelas karakter. Nilai *Overall Character Accuracy* dan *Mean Plate Accuracy* yang juga tinggi menunjukkan bahwa model mampu mengenali sebagian besar karakter maupun keseluruhan plat nomor dengan baik. Nilai CER yang relatif rendah menunjukkan bahwa jumlah operasi *edit* yang diperlukan untuk mengubah hasil prediksi menjadi *ground truth* juga relatif kecil (Vesalainen dkk., 2026). Nilai *Word Accuracy* yang lebih rendah menunjukkan bahwa satu kesalahan karakter saja dapat menyebabkan keseluruhan teks plat nomor dikategorikan sebagai prediksi yang salah (Gaikwad, 2026). Meskipun demikian, nilai *Word Accuracy* sebesar 69,54% menunjukkan bahwa sebagian besar citra plat nomor masih berhasil dikenali secara utuh.

3.5 Hasil Pengenalan Plat Nomor secara *End-to-End*

Contoh hasil pengenalan plat nomor secara *end-to-end* ditunjukkan pada Gambar 10. Pada tahap pertama, Faster R-CNN berhasil melokalisasi plat nomor pada citra kendaraan dengan *confidence score* sebesar 0,72. Area tersebut kemudian dipotong dan diproses melalui pembentukan masker warna hitam pada ruang warna HSV, pemilihan area utama plat, konversi *grayscale*, pembesaran citra, *Gaussian blur*, serta segmentasi menggunakan ambang Otsu. Tahap segmentasi menghasilkan delapan kandidat karakter yang selanjutnya diklasifikasikan oleh CNN OCR.

Tabel 4: Hasil metrik evaluasi model CNN OCR pada data uji.

Metrik	Nilai
Accuracy	93%
Precision (<i>weighted avg</i>)	0,93
Recall (<i>weighted avg</i>)	0,93
F1-Score (<i>weighted avg</i>)	0,93
Mean Plate Accuracy	92,62%
Overall Character Accuracy	92,78%
Word Accuracy	69,54%
Character Error Rate (CER)	7,22%
Karakter benar / total	1.350 / 1.455

Sistem menghasilkan teks “B1308RF0”, sesuai dengan susunan karakter yang terlihat pada citra plat. Visualisasi ini menunjukkan bahwa keluaran detektor dapat diteruskan hingga tahap pengenalan karakter, sekaligus memperlihatkan peran pra-pemrosesan dalam memisahkan karakter dari latar belakang plat. Contoh kualitatif tersebut melengkapi hasil kuantitatif pada Tabel 3 dan Tabel 4; performa keseluruhan sistem tetap mengacu pada evaluasi seluruh data uji, bukan hanya pada satu contoh citra.



Gambar 10: Contoh hasil pengenalan plat nomor secara *end-to-end*, mulai dari lokalisasi plat oleh Faster R-CNN, pra-pemrosesan dan segmentasi karakter, hingga keluaran CNN OCR “B1308RF0”.

3.6 Analisis dan Sintesis

Secara keseluruhan, hasil penelitian ini menunjukkan bahwa integrasi Faster R-CNN ResNet-50-FPN dan CNN OCR merupakan pendekatan yang efektif untuk sistem ALPR pada plat nomor kendaraan Indonesia. Perbandingan dengan baseline Roh (2017) menunjukkan bahwa model memperoleh nilai EER sebesar 0,2249, lebih rendah dibandingkan baseline sebesar 0,39, serta nilai NMD sebesar 0,1915, lebih rendah dibandingkan baseline sebesar 0,40. Hasil tersebut menunjukkan adanya peningkatan pada karakteristik kesalahan keputusan dan ketepatan lokalisasi

dibandingkan baseline. Nilai ROC-AUC sebesar 0,8669 juga menunjukkan kemampuan diskriminasi yang baik berdasarkan *confidence score* deteksi, meskipun masih terdapat ruang peningkatan menuju nilai ideal 1,0. Dengan demikian, ketiga metrik tersebut memberikan informasi yang saling melengkapi, yaitu ROC-AUC untuk mengevaluasi kemampuan diskriminasi *confidence score* pada berbagai *threshold*, EER untuk mengevaluasi keseimbangan kesalahan *false acceptance* dan *false rejection*, serta NMD untuk mengevaluasi ketepatan spasial *bounding box* terhadap *ground truth*. Pada tahap OCR, perbandingan dengan Kaur dkk. (2023) yang melaporkan akurasi keseluruhan 90

4 Kesimpulan

Penelitian ini mengembangkan dan mengevaluasi sistem *Automatic License Plate Recognition* (ALPR) dua tahap yang mengintegrasikan Faster R-CNN dengan *backbone* ResNet-50-FPN untuk lokalisasi plat nomor kendaraan dan CNN berbasis segmentasi karakter untuk pengenalan teks plat nomor. Evaluasi komprehensif menggunakan kerangka metrik yang mencakup ROC-AUC, EER, NMD, CER, *Word Accuracy*, *Mean Plate Accuracy*, *Overall Character Accuracy*, *Precision*, *Recall*, dan *F1-Score* menghasilkan temuan utama berikut. Pertama, Faster R-CNN dengan *backbone* ResNet-50-FPN mencapai ROC-AUC 0,8669, EER 0,2249, dan NMD 0,1915, melampaui *baseline* Faster R-CNN standar yang dilaporkan Roh (2017) secara signifikan. Kedua, model CNN OCR mencapai *accuracy* sebesar 93%, *precision (weighted average)* sebesar 0,93, *recall (weighted average)* sebesar 0,93, dan *F1-score (weighted average)* sebesar 0,93. Selain itu, model menghasilkan Overall Character Accuracy sebesar 92,78%, Mean Plate Accuracy sebesar 92,62%, dan Character Error Rate (CER) sebesar 7,22%, yang menunjukkan kemampuan pengenalan karakter yang baik dan konsisten pada data uji. Ketiga, pengujian *end-to-end* menunjukkan *Word Accuracy* sebesar 69,54%, yang mengindikasikan bahwa kesalahan pada satu atau lebih karakter masih dapat memengaruhi keberhasilan pengenalan teks plat nomor secara utuh. Hasil ini menunjukkan bahwa peningkatan pada tahap OCR masih diperlukan untuk meningkatkan akurasi pengenalan plat nomor secara keseluruhan. Penelitian ini memberikan kontribusi berupa kerangka evaluasi komprehensif dan acuan performa sistem ALPR berbasis *deep learning* untuk plat nomor kendaraan Indonesia. Untuk pengembangan selanjutnya, disarankan eksplorasi metode OCR berbasis *sequence* seperti CRNN, penerapan *post-processing* berbasis aturan format plat Indonesia, perbandingan dengan detektor modern seperti YOLOv8, serta implementasi *real-time* pada video atau CCTV untuk aplikasi sistem transportasi cerdas.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih kepada Program Studi Sarjana Sains Data Universitas Negeri Surabaya serta seluruh pihak yang telah memberikan dukungan dan masukan selama pelaksanaan penelitian ini.

Daftar Pustaka

- Abdelrahman, A. dan Viriri, S. (2023). FPN-SE-ResNet model for accurate diagnosis of kidney tumors using CT images. *Applied Sciences*, 13(17).
- Awasthi, A., Rao, S. B., dan Acharjya, K. (2024). Efficient image feature extraction using convolutional neural networks. pages 1–5.

- Carranza-García, M., Torres-Mateo, J., Lara-Benítez, P., dan García-Gutiérrez, J. (2021). On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data. *Remote Sensing*, 13(1):1–23.
- Chen, Y., Zhu, X., Li, Y., Wei, Y., dan Ye, L. (2023). Enhanced semantic feature pyramid network for small object detection. *Signal Processing: Image Communication*, 113.
- Ciampiconi, L., Elwood, A., Leonardi, M., Mohamed, A., dan Rozza, A. (2024). A survey and taxonomy of loss functions in machine learning. *arXiv preprint arXiv:2301.05579*.
- Dzakwanul Fikri, A., Utaminingrum, F., dan Setyawan, E. G. E. (2017). Sistem pendeteksi kekerasan di ruang publik menggunakan metode 3d convolutional neural network. In *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, volume 1.
- Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874.
- Gaikwad, V. (2026). Benchmarking OCR pipelines with adaptive enhancement for multi-domain retail bill digitization. *arXiv preprint arXiv:2604.25176*.
- Girshick, R. (2015). Fast R-CNN. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448.
- Girshick, R., Donahue, J., Darrell, T., dan Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587.
- Janocha, K. dan Czarnecki, W. M. (2017). On loss functions for deep neural networks in classification. *arXiv preprint arXiv:1702.05659*.
- Kaur, G., Jaiswal, A. K., Kumar, R., dan Thakur, K. (2023). Automatic license plate recognition system. In *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pages 1–6.
- Lamichhane, B. R., Srijuntongsiri, G., dan Horanont, T. (2025). CNN based 2d object detection techniques: A review. *Frontiers in Computer Science*, 7.
- Li, B. dan Lima, D. (2021). Facial expression recognition via ResNet-50. *International Journal of Cognitive Computing in Engineering*, 2:57–64.
- Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., dan Belongie, S. (2017). Feature pyramid networks for object detection.
- Moussaoui, H., Akkad, N. E., Benslimane, M., El-Shafai, W., Baihan, A., Hewage, C., dan Rathore, R. S. (2024). Enhancing automated vehicle identification by integrating YOLO v8 and OCR techniques for high-precision license plate detection and recognition. *Scientific Reports*, 14(1).
- Panahi, R. dan Gholampour, I. (2017). Accurate detection and recognition of dirty vehicle plate numbers for high-speed applications. *IEEE Transactions on Intelligent Transportation Systems*, 18(4):767–779.
- Popoola, S., Wolfram, D., Paulus, R., Horneegger, J., Tabak, E., dan Shamo, P. (2024). How to use computer vision to create a system for automatic license plate recognition. *ResearchGate Preprint*.
- Ren, S., He, K., Girshick, R., dan Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In *Proceedings of the 29th International Conference on Neural Information Processing Systems*, pages 91–99.
- Renjun, X., Junliang, Y., Yi, W., dan Mengcheng, S. (2022). Fault detection method based on improved faster R-CNN: Take ResNet-50 as an example. *Geofluids*.
- Rofiq Mulyana, A., Bejo, A., dan Nugroho, H. A. (2022). The improvement of character recognition on ANPR algorithm using CNN method with batch normalization. In *2021 International*

- Seminar on Machine Learning, Optimization, and Data Science (ISMODE)*, pages 168–172.
- Roh, M.-C. (2017). Refining Faster-RCNN for accurate object detection. *Kakao Corp Technical Report*.
- Sarhan, A., Abdel-Rahem, R., Darwish, B., Abou-Attia, A., Sneed, A., Hatem, S., Badran, A., dan Ramadan, M. (2024). Egyptian car plate recognition based on YOLOv8, Easy-OCR, and CNN. *Journal of Electrical Systems and Information Technology*, 11(1).
- Vesalainen, A., Mäkelä, E., Ruotsalainen, L., dan Tolonen, M. (2026). Error patterns in historical OCR: A comparative analysis of TrOCR and a vision-language model. *arXiv preprint arXiv:2602.14524*.
- Wijaya, J. T. (2025). Indonesian license plate dataset. Kaggle.
- Yang, X., Wu, J., He, L., Ma, S., Hou, Z., dan Sun, W. (2023). CPSSFAT: A consistent positive sample selection for object detection with full adaptive threshold. *Pattern Recognition*, 141.
- Yao, G., Zhu, S., Zhang, L., dan Qi, M. (2024). HP-YOLOv8: High-precision small object detection algorithm for remote sensing images. *Sensors*, 24(15).
- Zhao, C. H. dan Leong, W. Y. (2025). Optimisation solutions and simple innovative solution research on ResNet50 model. *ASM Science Journal*, 20(1).
- Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., dan Ren, D. (2019). Distance-IoU loss: Faster and better learning for bounding box regression. *arXiv preprint arXiv:1911.08287*.