

Implementasi CLAHE dan YOLOv11 untuk Pembacaan Pelat Nomor Kendaraan Beresolusi Rendah pada Raspberry Pi 5

¹Benni Agung Nugroho, ²Afta Ramadhan Zayn, ³Ellya Nurfarida, ⁴Riswan Eko Wahyu Susanto, ⁵Hadi Rahmad

^{1*,2,3}Manajemen Informatika, Politeknik Negeri Malang, Malang, Indonesia

^{4,5}Teknik Mesin, Politeknik Negeri Malang, Malang, Indonesia

*Korespondensi: benni.agung@polinema.ac.id

Submit : 25 Juli 2026 | **Diterima** : 19 Agust 2026 | **Terbit** : 26 Agust 2026

ABSTRACT

Vehicle license plate recognition in bus fleets often faces obstacles due to small plate size, low image resolution, and poor contrast, making characters difficult to recognize. This study proposes a low-resolution bus license plate character reading system using the You Only Look Once version 11 (YOLOv11) model combined with the Contrast Limited Adaptive Histogram Equalization (CLAHE) method as a preprocessing stage in the inference process. The training dataset was developed from a combination of synthetic license plate images and low-resolution real vehicle license plate images. The model was trained using an NVIDIA A100 GPU for 500 epochs and implemented on a Raspberry Pi 5 as an edge device. Validation results show the model achieved a precision of 88.9%, a recall of 76.2%, an mAP@50 of 87.2%, and an mAP@50-95 of 53.4%. Testing on real vehicle license plate images shows that the application of CLAHE increases the success rate of character reading from 82.13% to 91.04% with a relatively small additional inference time. Furthermore, implementation on a Raspberry Pi 5 demonstrates that the model can be run efficiently with low resource usage, making it worthy of consideration as an Artificial Intelligence of Things (AIoT)-based Automatic Number Plate Recognition (ANPR) solution.

Keywords: AIoT; ANPR; CLAHE; Edge Device; YOLOv11.

ABSTRAK

Pengenalan pelat nomor kendaraan pada armada bus sering menghadapi kendala akibat ukuran pelat yang kecil, resolusi citra yang rendah, serta kontras yang kurang baik sehingga karakter sulit dikenali. Penelitian ini mengusulkan sistem pembacaan karakter pelat nomor bus beresolusi rendah menggunakan model *You Only Look Once version 11* (YOLOv11) yang dipadukan dengan metode *Contrast Limited Adaptive Histogram Equalization* (CLAHE) sebagai tahap *preprocessing* pada proses inferensi. Dataset pelatihan dikembangkan dari kombinasi citra pelat nomor sintesis dan citra pelat nomor kendaraan riil beresolusi rendah. Model dilatih menggunakan GPU NVIDIA A100 selama 500 *epoch* dan diimplementasikan pada Raspberry Pi 5 sebagai *edge device*. Hasil validasi menunjukkan model mencapai *precision* sebesar 88,9%, *recall* sebesar 76,2%, *mAP@50* sebesar 87,2%, dan *mAP@50-95* sebesar 53,4%. Pengujian pada citra pelat nomor kendaraan riil menunjukkan bahwa penerapan CLAHE meningkatkan keberhasilan pembacaan karakter dari 82,13% menjadi 91,04% dengan tambahan waktu inferensi yang relatif kecil. Selain itu, implementasi pada Raspberry Pi 5 menunjukkan bahwa model dapat dijalankan secara efisien dengan penggunaan sumber daya yang rendah sehingga layak dipertimbangkan sebagai solusi *Automatic Number Plate Recognition* (ANPR) berbasis *Artificial Intelligence of Things* (AIoT).

Kata kunci: AIoT; ANPR; CLAHE; Edge Device; YOLOv11.

PENDAHULUAN

Automatic Number Plate Recognition (ANPR) merupakan teknologi yang berperan penting dalam identifikasi kendaraan secara otomatis pada berbagai aplikasi, seperti pemantauan lalu lintas, sistem penilangan elektronik, manajemen terminal, dan pengawasan armada transportasi. Perkembangan Artificial Intelligence (AI) dan Internet of Things (IoT) turut

mendorong pemanfaatan ANPR untuk mendukung sistem transportasi cerdas yang mampu melakukan identifikasi kendaraan secara real-time (Sharma et al., 2022).

Implementasi ANPR pada armada bus masih menghadapi berbagai tantangan, terutama ketika citra pelat nomor diperoleh dari kamera CCTV dengan jarak pengambilan yang relatif jauh. Kondisi tersebut menyebabkan pelat nomor hanya menempati sebagian kecil area citra sehingga karakter tampak berukuran kecil, buram, dan memiliki kontras rendah (Li et al., 2024; Tejaswi et al., 2026). Selain dipengaruhi oleh resolusi kamera, kualitas citra juga dipengaruhi oleh pencahayaan, sudut pengambilan, serta kondisi fisik pelat nomor yang kotor atau memudar. Akibatnya, proses deteksi dan pengenalan karakter menjadi kurang optimal.

Perkembangan metode deep learning, khususnya keluarga You Only Look Once (YOLO) (Ruseno et al., 2026), telah menunjukkan performa yang sangat baik pada sistem ANPR. Beberapa penelitian melaporkan bahwa model berbasis YOLO mampu mencapai akurasi deteksi dan pengenalan karakter yang tinggi pada berbagai kondisi lingkungan (Moussaoui et al., 2024; S et al., 2026; Xiong et al., 2026). Di antara berbagai pengembangannya, YOLOv11 menawarkan peningkatan efisiensi dan akurasi dibandingkan generasi sebelumnya sehingga berpotensi diterapkan pada sistem pengenalan pelat nomor (Suharto et al., 2025).

Meskipun demikian, performa YOLO masih dipengaruhi oleh kualitas citra masukan, terutama pada objek berukuran kecil dan beresolusi rendah. Salah satu pendekatan yang banyak digunakan untuk meningkatkan kualitas citra adalah Contrast Limited Adaptive Histogram Equalization (CLAHE), yang mampu meningkatkan kontras lokal sehingga fitur karakter menjadi lebih jelas (Arafat et al., 2026; Dewi et al., 2026; Galahartlambang et al., 2024). Selain aspek akurasi, implementasi ANPR pada perangkat edge juga masih menjadi tantangan karena keterbatasan sumber daya komputasi.

Berdasarkan permasalahan tersebut, penelitian ini mengusulkan sistem pembacaan karakter pelat nomor bus beresolusi rendah menggunakan YOLOv11 yang dipadukan dengan CLAHE sebagai tahap preprocessing. Model kemudian diimplementasikan pada Raspberry Pi 5 untuk mengevaluasi akurasi pembacaan karakter serta performa inferensi pada edge device. Penelitian ini diharapkan dapat menghasilkan sistem ANPR yang akurat, efisien, dan layak diterapkan sebagai solusi Artificial Intelligence of Things (AIoT).

METODE PENELITIAN

Penelitian ini menerapkan metode YOLOv11 yang merupakan varian arsitektur *Convolutional Neural Network* (CNN). YOLOv11 digunakan untuk mendeteksi objek bus, pelat nomor kendaraan, serta karakter alfanumerik (alfabet dan numerik) di dalam pelat nomor. Selain YOLOv11, metode CLAHE diintegrasikan sebagai teknik pengolahan citra digital untuk meningkatkan kualitas visual pelat nomor beresolusi rendah agar fitur garis karakter dapat dikenali oleh model YOLOv11 secara presisi.

Tahapan penelitian meliputi:

- Pengumpulan *dataset* gabungan yang terdiri dari citra sintetis (dibangkitkan secara komputasi melalui kode program) dan citra riil pelat nomor bus yang diperoleh dari pemotongan otomatis (*automatic cropping*) berbasis YOLO (Nugroho et al., 2025).
- Pelatihan (*training*) model YOLOv11 menggunakan *hardware* GPU NVIDIA A100 pada platform *Google Colab*.
- Pengujian dan evaluasi model YOLOv11 dengan pengintegrasian prapemrosesan CLAHE sebelum tahap inferensi.
- Pengujian performa komputasi dan latensi pada *edge device* Raspberry Pi 5 untuk mengevaluasi kelayakan sistem pada perangkat keras bersumber daya terbatas.

Pengumpulan Dataset

Dataset yang digunakan dalam proses pelatihan YOLOv11 bersumber dari dua jenis data: (1) *dataset* sintetis berupa gambar pelat nomor kendaraan yang dibangkitkan menggunakan kode program, dan (2) *dataset* citra riil pelat nomor bus yang dipotong secara otomatis menggunakan model YOLOv11.

1. Dataset Pelat Nomor Kendaraan Sintetis

Pembangkitan *dataset* sintetis dilakukan menggunakan bahasa pemrograman Python dan pustaka *Albumentations* (Buslaev et al., 2020) dengan konfigurasi sebagai berikut:

- Pembuatan Kelas Karakter (*Class Map*):** Mendefinisikan 36 kelas objek, mencakup angka '0'–'9' dan huruf 'A'–'Z'. Setiap kelas dipetakan ke indeks numerik unik ('0' → 0, '1' → 1, ...,

- 'Z' → 35) sebagai identifikasi kelas (*class ID*) dalam anotasi YOLO.
- Penerapan Tipografi Standar:** Karakter alfanumerik dibangkitkan menggunakan *font* FE-Schrift (*Fälschungsschwerende Schrift*), yaitu jenis huruf standar keselamatan yang dirancang anti-pemalsuan (Albrecht, A., Franke, R., & Zuber, 1996)
 - Simulasi Augmentasi Geometris:**
 - Affine Transformation** (Hartley, R., & Zisserman, 2004): Meliputi skalasi (*scaling*) untuk mengubah ukuran pelat, translasi (*translation*) untuk pergeseran posisi, serta penyimpangan (*shear*) untuk memiringkan bentuk pelat guna menirukan variasi sudut pandang dan jarak kamera.
 - Perspective Transformation** (Szeliski, 2022): Mengubah sudut pandang dan kedalaman spasial untuk mensimulasikan pelat yang terekam dari ruang 3D.
 - Simulasi Kondisi Fisik Pelat (Degradation):** Menerapkan metode *Coarse Dropout* (Devries, T., & Taylor, 2017) untuk mensimulasikan efek kotoran lumpur, noda oli, cat terkelupas, debu kusam, serta pantulan cahaya (*glare*).
 - Simulasi Lensa Kamera dan Pencahayaan:** Menerapkan fungsi *Blur*, *Gaussian Noise*, *Random Brightness and Contrast*, serta variasi *Hue-Saturation-Value* (HSV) (Shorten, C., & Khoshgoftaar, 2019) untuk mensimulasikan efek derau sensor kamera dan fluktuasi pencahayaan alami.
 - Anotasi Otomatis:** Setiap karakter yang dibangkitkan langsung diberi label berupa koordinat kotak pembatas (*bounding box*) berformat YOLO.
 - Pembagian Data:** *Dataset* sintetis dibagi menjadi data *training* sebesar 80% dan data *validation* sebesar 20%.

Contoh citra *dataset* sintetis beserta *bounding box* anotasi ditunjukkan pada Gambar 1.



Gambar 1. Contoh dataset sintetis dan anotasi YOLO

2. Dataset Pelat Nomor Kendaraan Riil

Dataset riil diperoleh dari mekanisme ekstraksi berjenjang memanfaatkan model pra-latih (*pre-trained*) YOLOv11 (Jocher, G., Qiu, J., 2024) dan model deteksi pelat nomor. Alur proses ekstraksi yang diilustrasikan pada Gambar 2 adalah sebagai berikut:

- Deteksi Kendaraan Bus:** Model *pre-trained* YOLOv11 mendeteksi lokasi armada bus pada citra masukan tangkapan kamera.
- Pemotongan Pelat Nomor (Plate Cropping):** Model spesifik deteksi pelat nomor melakukan inferensi pada area bus untuk mengisolasi dan memotong area pelat nomor kendaraan.
- Anotasi Manual Alfanumerik:** Karakter alfanumerik pada hasil *crop* yang masih terproyeksi dengan jelas dianotasi secara manual menggunakan perangkat *Roboflow* untuk membentuk data latih klasifikasi karakter.

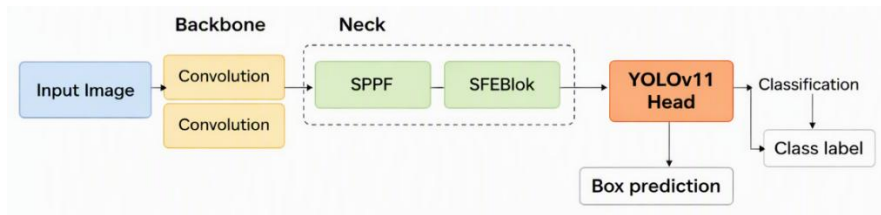


Gambar 2. Proses mendapatkan dataset pelat nomor kendaraan riil

Model YOLOv11

YOLOv11 merupakan generasi ke-11 dari seri algoritma YOLO yang dikembangkan oleh Ultralytics. YOLOv11 dirancang untuk mengoptimalkan keseimbangan antara kecepatan inferensi (*throughput*) dan akurasi deteksi. Arsitektur YOLOv11 mengintegrasikan komponen *C3k2 block*, *Spatial Pyramid Pooling – Fast* (SPPF), serta *Parallel Spatial Attention* (C2PSA) guna

meningkatkan efisiensi ekstraksi fitur spasial. Arsitektur YOLOv11 diilustrasikan pada Gambar 3.



Gambar 3. Arsitektur YOLOv11

Komponen utama arsitektur YOLOv11 terdiri dari :

- Backbone:** Bertindak sebagai ekstraktor fitur (*feature extractor*) dari citra masukan. Menggunakan blok *C3k2* untuk mempercepat konvolusi dengan beban komputasi rendah tanpa mengurangi kedalaman representasi fitur.
- Neck:** Menggabungkan fitur multi-skala menggunakan:
 - SPPF:** Mengekstrak konteks spasial pada berbagai skala resolusi secara cepat.
 - Blok SFE (*Spatial Feature Enhancement*):** Memperkuat representasi fitur spasial objek berukuran kecil (seperti karakter pelat nomor) agar tidak hilang pada lapisan konvolusi dalam.
- YOLOv11 Head:** Menghasilkan dua keluaran utama:
 - Box Prediction:** Memprediksi koordinat spasial kotak pembatas (x, y, w, h).
 - Classification:** Memprediksi distribusi probabilitas kelas alfanumerik.

Peningkatan Kontras Citra dengan Metode CLAHE

CLAHE (*Contrast Limited Adaptive Histogram Equalization*) (Sutikno et al., 2025) merupakan salah satu metode pengolahan citra digital yang masuk dalam kategori *image enhancement*, khususnya pada teknik perataan histogram (*histogram equalization*). Secara matematis, tahapan CLAHE mengacu pada perataan histogram lokal yang dibatasi batas ambang (clipping) yang dapat dijelaskan sebagai berikut :

- Perhitungan Batas Ambang (*Clip Limit*)**

Batas pemotongan dihitung berdasarkan jumlah piksel dalam ubin ($M \times N$), jumlah tingkat keabuan (L , biasanya 256), dan faktor pembatas kontras (N_{clip}):

$$T_{clip} = \frac{M \times N}{L} \times \left(1 + \frac{\alpha}{100} \times (S_{max} - 1) \right)$$

Di mana $M \times N$ adalah dimensi ubin, L adalah jumlah gray level, dan α / S_{max} mewakili tingkat pembatasan kontras yang disesuaikan.

- Fungsi Distribusi Kumulatif (CDF)**

Setelah histogram ubin dipotong dan didistribusikan ulang, dibuat fungsi **CDF** $C(k)$ untuk pemetaan intensitas:

$$C(k) = \sum_{j=0}^k p(j)$$

Di mana $p(j)$ adalah probabilitas kemunculan intensitas j pada ubin setelah pemotongan:

$$p(j) = \frac{n_j}{M \times N}$$

- Pemetaan Nilai Piksel Baru**

Setiap nilai piksel lama x pada ubin diubah menjadi nilai piksel baru y menggunakan hasil perkalian CDF dengan nilai keabuan maksimum ($L - 1$):

$$y = T(x) = (L - 1) \cdot C(x)$$

- Interpolasi Bilinear**

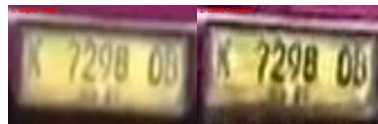
Untuk piksel P yang berada di antara empat titik pusat ubin ($C_{11}, C_{12}, C_{21}, C_{22}$), nilai piksel akhir $f(P)$ dihitung menggunakan interpolasi bilinear:

$$f(P) \approx (1 - a)(1 - b) T_{11}(x) + a(1 - b) T_{21}(x) + (1 - a)b T_{12}(x) + ab T_{22}(x)$$

Di mana a dan b adalah jarak relatif piksel ke pusat ubin terdekat ($0 \leq a, b \leq 1$), dan $T_{ij}(x)$ adalah hasil pemetaan nilai piksel dari ubin terkait.

Contoh hasil pengolahan citra menggunakan CLAHE ditunjukkan oleh Gambar 4 dimana pada gambar sebelah kiri merupakan gambar pelat nomor tanpa pengolahan citra CLAHE dan gambar sebelah kanan merupakan gambar pelat nomor setelah diterapkan metode pengolahan

citra CLAHE



Gambar 4. Contoh pelat nomor kendaraan tanpa CLAHE(kiri) dan dengan CLAHE (kanan)

HASIL DAN PEMBAHASAN

Pengujian dilakukan untuk mengukur performa model YOLOv11 berdasarkan metrik *Precision*, *Recall*, *Mean Average Precision* (mAP), kecepatan inferensi (*latency/throughput*), serta akurasi pengenalan karakter. Evaluasi dibagi menjadi tahap pelatihan, validasi, dan inferensi pada *edge device*.

Performa Pelatihan

Proses pelatihan dilakukan menggunakan GPU NVIDIA A100 pada platform *Google Colab*. *Dataset* pelatihan terdiri dari 472 citra pelat nomor (40 citra riil beresolusi rendah dengan dimensi 73×30 hingga 114×65 piksel, serta 432 citra sintetis berdimensi 640×480 piksel). Parameter utama pelatihan disajikan pada Tabel 1.

Tabel 1. Parameter pelatihan YOLOv11n

Parameter Pelatihan	Nilai
<i>Epoch</i>	500
<i>Image size</i>	640x640 pixel
<i>Learning rate</i>	0.00025
<i>Momentum</i>	0.9
<i>Optimizer</i>	AdamW
Jumlah kelas	36, berupa angka (0 – 9) dan huruf (A – Z)

Proses pelatihan berlangsung selama ≈ 30 menit 30 detik. Kurva kerugian (*loss curve*) menunjukkan tren konvergensi yang stabil hingga *epoch* ke-500:

- Box loss:** Menurun dari 2,047 (*epoch* 1) menjadi 0,473 (*epoch* 500), mengindikasikan meningkatkan presisi regresi koordinat *bounding box*.
- Class loss:** Menurun drastis dari 5,135 (*epoch* 1) menjadi 0,429 (*epoch* 500), mengindikasikan ketepatan klasifikasi karakter yang semakin membaik.
- Distribution Focal Loss (DFL):** Menyusut dari 1,566 (*epoch* 1) menjadi 0,838 (*epoch* 500), menandakan lokalisasi batas objek yang lebih konsisten.
- Training loss** menurun dari 8,748 menjadi 1,741, yang menunjukkan bahwa proses pembelajaran berlangsung secara konsisten hingga akhir pelatihan.
- Validation Loss & mAP50:** Nilai mAP50 validasi meningkat dari 0,005 (*epoch* 2) hingga mencapai 0,872 (*epoch* 500). Tren penurunan *training loss* yang sejajar dengan peningkatan performa validasi mengindikasikan bahwa model tidak mengalami *overfitting*.

Secara keseluruhan, penurunan seluruh komponen *loss* yang diikuti dengan peningkatan nilai *mAP@50* menunjukkan bahwa model berhasil mempelajari karakteristik data pelat nomor kendaraan dengan baik serta memiliki kemampuan generalisasi yang memadai pada data validasi.

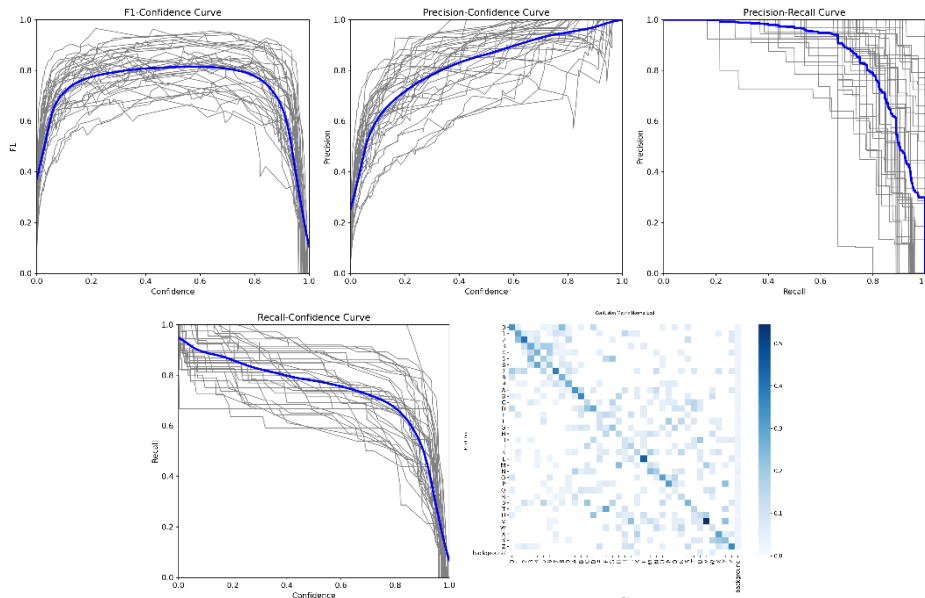
Performa Validasi

Performa validasi dievaluasi menggunakan 113 citra yang memuat 762 objek karakter, terdiri atas 5 citra pelat nomor kendaraan riil berukuran 73×29 hingga 98×43 piksel dan 108 citra pelat nomor sintetis berukuran 640×480 piksel. Hasil evaluasi disajikan pada Gambar 5, sedangkan ringkasan metrik evaluasi ditunjukkan pada Tabel 2.

Tabel 2. Ringkasan Metrik Evaluasi Model YOLOv11

Grafik Evaluasi	Metrik	Nilai	Penilaian
<i>F1-Confidence</i>	<i>Peak F1/Conf</i>	0.81@0.575	Sangat Baik

Grafik Evaluasi	Metrik	Nilai	Penilaian
<i>Precision-Confidence</i>	<i>Max Precision</i>	1.00@1.000	Sangat Baik
<i>Recall-Confidence</i>	<i>Max Recall</i>	0.95@0.000	Baik
<i>Precision-Recall</i>	<i>mAP@0.5</i>	0.872	Sangat Baik
<i>Normalized Confusion Matrix</i>	<i>Diagonal dominance</i>	Cukup pekat	Baik



Gambar 5. Grafik evaluasi dan confusion matrix

Berdasarkan Gambar 5 dan Tabel 2, model YOLOv11 menunjukkan performa validasi yang baik pada seluruh metrik evaluasi. Kurva *F1-Confidence* mencapai nilai maksimum sebesar 0,81 pada *confidence threshold* 0,575 yang menunjukkan keseimbangan yang baik antara *precision* dan *recall*. Kurva *Precision-Confidence* memperlihatkan bahwa nilai *precision* meningkat seiring bertambahnya *confidence threshold*, sedangkan kurva *Recall-Confidence* menunjukkan penurunan secara bertahap yang merupakan karakteristik umum pada model deteksi objek. Kurva *Precision-Recall* menghasilkan nilai *mAP@50* sebesar 0,872 yang mengindikasikan kemampuan model dalam mendeteksi objek dengan tingkat akurasi yang tinggi. Selain itu, *Normalized Confusion Matrix* menunjukkan bahwa sebagian besar karakter berhasil diklasifikasikan dengan benar, meskipun masih terdapat beberapa pasangan karakter yang memiliki bentuk serupa, seperti V–W, D–O, dan I–1.

Evaluasi kuantitatif menghasilkan nilai *precision* sebesar 88,9%, *recall* sebesar 76,2%, *mAP@50* sebesar 87,2%, dan *mAP@50–95* sebesar 53,4%. Nilai *precision* yang tinggi menunjukkan bahwa sebagian besar prediksi yang dihasilkan model merupakan prediksi yang benar, sedangkan nilai *recall* menunjukkan bahwa sebagian besar karakter pada dataset validasi berhasil terdeteksi. Nilai *mAP@50* dan *mAP@50–95* mengindikasikan bahwa model memiliki kemampuan deteksi yang baik, meskipun tingkat presisi lokalisasi masih dapat ditingkatkan pada ambang *Intersection over Union* (IoU) yang lebih tinggi.

Performa Pembacaan Pelat Nomor Kendaraan pada Citra Berkualitas Rendah

Model YOLOv11 hasil pelatihan selanjutnya diuji menggunakan 27 citra pelat nomor kendaraan riil beresolusi rendah (Apit et al., 2025) dengan dimensi berkisar antara 68×31 hingga 145×30 piksel. Pengujian dilakukan dengan membandingkan performa pembacaan karakter menggunakan CLAHE sebagai tahap *preprocessing* dan tanpa CLAHE. Ringkasan hasil pengujian disajikan pada Tabel 3, sedangkan contoh citra uji ditunjukkan pada Gambar 6.

Tabel 3. Hasil pengujian pembacaan karakter di dalam pelat nomor kendaraan

	Terbaca Manusia	Terbaca CLAHE + YOLOv11	Terbaca YOLOv11
Jumlah total karakter di dalam 27 pelat nomor kendaraan	204	201	207
Jumlah karakter False Positive (FP)	0	18	37
Jumlah karakter True Positive (TP)	204	183	170
Presentase keberhasilan mengenali karakter	100%	91,04%	82,13%

Berdasarkan Tabel 3, penerapan CLAHE meningkatkan persentase keberhasilan pembacaan karakter dari 82,13% menjadi 91,04%. Peningkatan ini menunjukkan bahwa CLAHE mampu memperbaiki kontras lokal sehingga fitur tepi karakter menjadi lebih jelas dan lebih mudah dikenali oleh model YOLOv11. Selain meningkatkan jumlah karakter yang berhasil dikenali, penggunaan CLAHE juga mengurangi kesalahan deteksi dibandingkan pengujian tanpa tahap *preprocessing*. Hasil tersebut menunjukkan bahwa peningkatan kualitas citra sebelum proses inferensi memberikan kontribusi yang signifikan terhadap pembacaan karakter pada citra pelat nomor beresolusi rendah.

Persentase keberhasilan pembacaan karakter dihitung menggunakan persamaan berikut.

$$\text{Presentase pengenalan karakter} = \frac{\text{total karakter TP}}{\text{total karakter terbaca}} \times 100\%$$



Gambar 6. Contoh citra pelat nomor kendaraan berkualitas rendah

Performa Inferensi YOLOv11 pada Edge Device

Pengujian performa inferensi dilakukan untuk mengevaluasi kelayakan implementasi model YOLOv11 pada Raspberry Pi 5 sebagai *edge device* (Moussaoui et al., 2024). Pengujian menggunakan dua kelompok data, yaitu 113 citra validasi berukuran 640×480 piksel dan 27 citra pelat nomor kendaraan berukuran 68×31 hingga 145×30 piksel. Spesifikasi perangkat keras pengujian disajikan pada Tabel 4, sedangkan hasil pengujian ditunjukkan pada Tabel 5.

Tabel 4. Spesifikasi hardware pengujian performa pada edge device

Perangkat Keras Spesifikasi	
Raspberry Pi 5	CPU: Broadcom BCM2712 (Quad-core 64-bit Arm Cortex-A76 @2.4GHz) RAM : 8 GB Storage : Micro-SD 32 GB
Power supply	27 Watt

Tabel 5. Hasil uji performa inferensi pada edge device (Raspberry Pi 5)

Data uji	Metode	$\bar{x}$ CLAHE Time (ms)	$\bar{x}$ Inference (ms)	$\bar{x}$ Total Latency (ms)	$\bar{x}$ Throughput (FPS)	$\bar{x}$ CPU Load (%)	Peak RAM (MB)
113 Data validasi (640x480)	CLAHE + YOLOv11	9.24	300.31	309.55	3.23	51.46%	408.70
	YOLOv11	0	300.84	300.98	3.32	51.78%	405.06

Data uji	Metode	$\bar{x}$ CLAHE Time (ms)	$\bar{x}$ Inference (ms)	$\bar{x}$ Total Latency (ms)	$\bar{x}$ Throughput (FPS)	$\bar{x}$ CPU Load (%)	Peak RAM (MB)
pixel)							
27 Data pelat	CLAHE + YOLOv11	5.34	193.56	198.90	5.03	52.18%	402.72
kendaraan (68x31 s/d 145x30 pixel)	YOLOv11	0	191.58	191.59	5.22	52.83	398.31

Berdasarkan Tabel 5, penerapan CLAHE hanya menambah waktu *preprocessing* sebesar 5–9 ms sehingga peningkatan *total latency* relatif kecil dibandingkan pengujian tanpa CLAHE. Hal ini menunjukkan bahwa penggunaan CLAHE tidak memberikan beban komputasi yang signifikan pada Raspberry Pi 5. Sebaliknya, ukuran citra memberikan pengaruh yang lebih besar terhadap waktu inferensi. Citra berukuran 640×480 piksel menghasilkan *latency* sekitar 300 ms, sedangkan citra berukuran lebih kecil menghasilkan *latency* sekitar 190 ms.

Nilai *throughput* yang berada pada kisaran 3–5 FPS menunjukkan bahwa model masih belum memenuhi kebutuhan aplikasi yang memerlukan pemrosesan *real-time* berkecepatan tinggi. Meskipun demikian, performa tersebut masih memadai untuk aplikasi pemantauan kendaraan dengan laju pergerakan rendah, seperti sistem pendataan kendaraan pada gerbang parkir, terminal, atau area akses terbatas.

Selama pengujian, penggunaan CPU berada pada kisaran 51–53%, sedangkan kebutuhan memori hanya sekitar 400 MB. Hasil tersebut menunjukkan bahwa model dapat dijalankan secara stabil dengan penggunaan sumber daya yang relatif efisien sehingga layak diimplementasikan pada perangkat *edge* dengan kapasitas komputasi terbatas.

Pembahasan

Hasil penelitian menunjukkan bahwa penerapan *Contrast Limited Adaptive Histogram Equalization* (CLAHE) sebagai tahap *preprocessing* mampu meningkatkan keberhasilan pembacaan karakter pada citra pelat nomor kendaraan beresolusi rendah dari 82,13% menjadi 91,04%. Peningkatan tersebut menunjukkan bahwa CLAHE efektif dalam memperbaiki kualitas citra sebelum proses inferensi sehingga karakter lebih mudah dikenali oleh model YOLOv11.

Peningkatan performa tersebut dipengaruhi oleh kemampuan CLAHE dalam meningkatkan kontras lokal tanpa memperbesar derau (*noise*) secara berlebihan. Perbaikan kontras menghasilkan fitur tepi karakter yang lebih jelas sehingga proses ekstraksi fitur dan deteksi karakter oleh YOLOv11 menjadi lebih optimal, terutama pada citra pelat nomor dengan kontras rendah dan ukuran karakter yang kecil. Selain itu, hasil pengujian menunjukkan bahwa penerapan CLAHE hanya menambah waktu *preprocessing* sebesar beberapa milidetik sehingga tidak memberikan pengaruh yang signifikan terhadap waktu inferensi pada Raspberry Pi 5.

Temuan penelitian ini sejalan dengan beberapa penelitian terdahulu yang melaporkan bahwa peningkatan kualitas citra sebelum proses deteksi mampu meningkatkan performa sistem *Automatic Number Plate Recognition* (ANPR). Penelitian oleh (Arafat et al., 2026) menunjukkan bahwa CLAHE efektif meningkatkan kontras lokal sehingga objek dengan intensitas rendah menjadi lebih mudah terdeteksi. Hasil serupa juga dilaporkan oleh (Galahartlambang et al., 2024) dan (Dewi et al., 2026) yang menyatakan bahwa penerapan CLAHE mampu memperjelas fitur citra dan meningkatkan akurasi proses pengenalan karakter pada berbagai aplikasi *computer vision*. Hasil penelitian ini memperkuat temuan-temuan tersebut dengan menunjukkan bahwa manfaat CLAHE juga tetap diperoleh pada sistem pembacaan karakter pelat nomor kendaraan beresolusi rendah yang diimplementasikan pada Raspberry Pi 5 sebagai *edge device*.

Implementasi model pada Raspberry Pi 5 juga menunjukkan bahwa sistem dapat dijalankan dengan penggunaan CPU dan memori yang relatif efisien. Meskipun *throughput* yang diperoleh masih berada pada kisaran 3–5 FPS, performa tersebut masih memadai untuk aplikasi ANPR pada lingkungan dengan laju kendaraan rendah hingga sedang, seperti gerbang parkir, terminal, atau area akses terbatas. Hasil ini menunjukkan bahwa kombinasi CLAHE dan YOLOv11 memiliki potensi untuk diterapkan pada sistem ANPR berbasis *edge device*.

Meskipun demikian, penelitian ini masih memiliki beberapa keterbatasan. Pertama, jumlah citra pelat nomor kendaraan riil yang digunakan sebagai dataset pelatihan dan pengujian masih relatif terbatas sehingga kemampuan generalisasi model pada kondisi lapangan yang lebih beragam belum dapat dievaluasi secara menyeluruh. Kedua, implementasi sistem hanya diuji pada Raspberry Pi 5 sehingga performanya pada platform *edge device* lain belum diketahui. Ketiga, penelitian ini hanya mengevaluasi CLAHE sebagai metode peningkatan kualitas citra tanpa membandingkannya dengan metode *image enhancement* lainnya. Oleh karena itu, penelitian selanjutnya dapat difokuskan pada penambahan dataset riil yang lebih beragam, evaluasi pada berbagai platform *edge device*, serta perbandingan beberapa metode *image enhancement* untuk memperoleh kombinasi yang paling efektif.

KESIMPULAN

Penelitian ini berhasil mengembangkan sistem pembacaan karakter pelat nomor kendaraan bus beresolusi rendah menggunakan model YOLOv11 yang dipadukan dengan metode prapemrosesan CLAHE pada *edge device* Raspberry Pi 5. Hasil pengujian menunjukkan bahwa penerapan CLAHE mampu meningkatkan keberhasilan pembacaan karakter dari 82,13% menjadi 91,04% tanpa menimbulkan penurunan performa inferensi yang signifikan. Model yang dihasilkan juga menunjukkan performa validasi yang baik dengan nilai *precision*, *recall*, dan *mAP* yang tinggi, sehingga layak diterapkan sebagai solusi *Automatic Number Plate Recognition* (ANPR) berbasis *Artificial Intelligence of Things* (AIoT). Meskipun demikian, peningkatan jumlah dataset riil serta optimasi model masih diperlukan untuk meningkatkan kemampuan generalisasi dan kecepatan inferensi pada implementasi di lingkungan nyata.

REFERENSI

- Albrecht, A., Franke, R., & Zuber, W. (1996). *FE-Schrift: Die redefinition der deutschen kraftfahrzeugkennzeichen*. Bundesanstalt für Straßenwesen (BASt).
- Apit, L. A. T., Avelino, J. D. S., & Yumang, A. N. (2025). Enhanced License Plate Image Resolution Using Super Resolution Generative Adversarial Networks. *Advances in Transdisciplinary Engineering*, 78, 185–194. <https://doi.org/10.3233/ATDE251144>
- Arafat, Andono, P. N., Syukur, A., & Affandy. (2026). Multi-Variant Enhancement and OCR Selection for Robust License Plate Recognition in Hazy Conditions. *Statistics, Optimization and Information Computing*, 16(August), 1086–1109. <https://doi.org/10.19139/soic-2310-5070-3818>
- Buslaev, A., Iglovikov, V. I., Khvedchenya, E., Parinov, A., Druzhinin, M., & Kalinin, A. A. (2020). Albumentations: Fast and flexible image augmentations. *Information (Switzerland)*, 11(2), 1–20. <https://doi.org/10.3390/info11020125>
- Devries, T., & Taylor, G. W. (2017). *Improved regularization of convolutional neural networks with Cutout*.
- Dewi, C., Chernovita, H. P., Santoso, M. V. V., Mailoa, E., Philemon, S. A., Sutresno, S. A., & Chen, A. P. S. (2026). Improving Early Fire and Smoke Detection with YOLOv12N via Histogram-Based Contrast Enhancement. *International Journal of Safety and Security Engineering*, 16(2), 431–445. <https://doi.org/https://doi.org/10.18280/ijssse.160217>
- Galahartlambang, Y., Khotiah, T., K, I. B., Anwar, M., & Abdillah, D. F. (2024). Optimalisasi Preprocessing untuk Peningkatan Akurasi Pengenalan Plat Nomor pada Citra Tidak Ideal. *Nucleus Journal Jurnal Sains Dan Teknologi*, 3(2), 109–116. <https://doi.org/https://doi.org/10.32492/nucleus.v3i2.3206>
- Hartley, R., & Zisserman, A. (2004). *Multiple view geometry in computer vision (2nd ed.)*. Cambridge University Press.
- Jocher, G., Qiu, J., & C. A. (2024). *Ultralytics YOLOv11 (Version 11.0.0) [Computer software]*. Ultralytics. <https://github.com/ultralytics/ultralytics>
- Li, X., Luo, Y., Yu, Z., Wang, J., & Ni, L. (2024). Research and Implementation of License Plate Recognition Based on CNN Algorithm Under Low Illumination. *Advances in Transdisciplinary Engineering*, 55, 46–56. <https://doi.org/10.3233/ATDE240374>
- Moussaoui, H., Akkad, N. El, Benslimane, M., El-Shafai, W., Baihan, A., Hewage, C., & 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(14389). <https://doi.org/10.1038/s41598-024-65272-1>
- Nugroho, B. A., Cinderatama, T. A., Izzah, A., Nurfarida, E., Studi, P., Informatika, M., Kediri, P.

- P., Lingkar, J., & No, M. (2025). *Deteksi Plat Nomor Kendaraan Angkutan Bus Menggunakan YOLOv11*. 17(2), 112–120.
- Ruseno, N., Ernawan, F., Salamat, M. A., Arshad, M. S., & Amnur, H. (2026). An Enhanced YOLOv8 Model Integrated with CNN for Real-Time Vehicle License Plate Recognition. *International Journal on Informatics Visualization*, 10(1), 343–349. <https://doi.org/10.62527/joiv.10.1.5145>
- S, T., Babu, T., Naira, R. R., & Rajamani, K. (2026). ScienceDirect Two-Stage Cascaded Detection of Vehicles and License Plates under Adverse Conditions. *Procedia Computer Science*, 282, 124–134. <https://doi.org/10.1016/j.procs.2026.05.061>
- Sharma, P., Gupta, S., Singh, P., Shejul, K., & Reddy, D. (2022). Automatic Number Plate Recognition and Parking Management. *International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, 1–8. <https://doi.org/10.1109/ACCAI53970.2022.9752632>
- Shorten, C., & Khoshgoftaar, T. M. (2019). (2019). A survey on image data augmentation for deep learning. *Journal of Big Data*, 6(1), 1–48. <https://doi.org/10.1186/s40537-019-0197-0>
- Suharto, R. N., Faroby, M. H. Z. Al, & Setiawan, Y. (2025). Comparative Evaluation of YOLO-based Models for Indonesian License Plate Detection and Recognition with EASY OCR. *The 10th International Conference on Computer Science and Computational Intelligence*, 269, 943–952. <https://doi.org/10.1016/j.procs.2025.09.037>
- Sutikno, Sugiharto, A., & Kusumaningrum, R. (2025). Enhanced Automatic License Plate Detection and Recognition using CLAHE and YOLOv11 for Seat Belt Compliance Detection. *Engineering, Technology and Applied Science Research*, 15(1), 20271–20278. <https://doi.org/10.48084/etasr.9629>
- Szeliski, R. (2022). *Computer vision: Algorithms and applications (2nd ed.)*. Springer Nature.
- Tejaswi, S., Babu, T., Nair, R. R., & Rajamani, K. (2026). Two-Stage Cascaded Detection of Vehicles and License Plates under Adverse Conditions. *Procedia Computer Science*, 282, 124–134. <https://doi.org/10.1016/j.procs.2026.05.061>
- Xiong, W., Cao, L., Yan, D., Jiang, Y., Zhang, G., Wang, Y., Huang, X., & Liu, L. (2026). License plate recognition methodology in complex scenarios based on CSCM-YOLOv8 and CSM-LPRNet. *PLoS One*, 21(1), 1–28. <https://doi.org/10.1371/journal.pone.0339649>