

Analisis Algoritma *Canny Edge Detection* dengan *Tesseract OCR* untuk Mendeteksi Pelat Nomor Kendaraan Bermotor

Ananda Sarbaini Firin dan Sriani Sriani



Volume 13, Issue 3, Pages 381–393, Dec. 2025

Diterima 13 September 2025, Direvisi 5 November 2025, Disetujui 22 November 2025, Diterbitkan 1 Desember 2025

To Cite this Article : A. S. Firin dan S. Sriani, “Analisis Algoritma *Canny Edge Detection* dengan *Tesseract OCR* untuk Mendeteksi Pelat Nomor Kendaraan Bermotor”, *Euler J. Ilm. Mat. Sains dan Teknol.*, vol. 13, no. 3, pp. 381–393, 2025, <https://doi.org/10.37905/euler.v13i3.34868>

© 2025 by author(s)

JOURNAL INFO • EULER : JURNAL ILMIAH MATEMATIKA, SAINS DAN TEKNOLOGI



🏠	Homepage	:	http://ejournal.ung.ac.id/index.php/euler/index
📄	Journal Abbreviation	:	Euler J. Ilm. Mat. Sains dan Teknol.
📅	Frequency	:	Three times a year
🗣️	Publication Language	:	English (preferable), Indonesia
🔗	DOI	:	https://doi.org/10.37905/euler
📖	Online ISSN	:	2776-3706
📄	License	:	Creative Commons Attribution-NonCommercial 4.0 International License
🏢	Publisher	:	Department of Mathematics, Universitas Negeri Gorontalo
🌍	Country	:	Indonesia
🌐	OAI Address	:	http://ejournal.ung.ac.id/index.php/euler/oai
🔍	Google Scholar ID	:	QF_r_gAAAAJ
✉️	Email	:	euler@ung.ac.id

JAMBURA JOURNAL • FIND OUR OTHER JOURNALS



Jambura Journal of Biomathematics



Jambura Journal of Mathematics



Jambura Journal of Mathematics Education



Jambura Journal of Probability and Statistics

Analisis Algoritma *Canny Edge Detection* dengan *Tesseract OCR* untuk Mendeteksi Pelat Nomor Kendaraan Bermotor

Ananda Sarbaini Firin^{1,*}, Sriani Sriani¹

¹Program Studi Ilmu Komputer, Universitas Islam Negeri Sumatera Utara, Deli Serdang 20371, Indonesia

ARTICLE HISTORY

Diterima 13 September 2025
Direvisi 5 November 2025
Disetujui 22 November 2025
Diterbitkan 1 Desember 2025

KATA KUNCI

Canny Edge Detection
Tesseract OCR
Pelat Nomor Kendaraan
Optical Character Recognition
Pengolahan Citra Digital

KEYWORDS

Canny Edge Detection
Tesseract OCR
Vehicle License Plates
Optical Character Recognition
Digital Image Processing

ABSTRAK. Pemeriksaan identitas dan kepemilikan kendaraan merupakan rutinitas penting yang dilakukan oleh pihak keamanan (*security*) di Universitas Islam Negeri Sumatera Utara (UINSU) untuk memastikan legalitas kendaraan yang masuk maupun keluar. Namun, proses pemeriksaan yang masih dilakukan secara manual dengan mencocokkan STNK dan pelat nomor kendaraan sering memakan waktu lama dan menimbulkan antrean, terutama pada jam sibuk. Penelitian ini menerapkan gabungan algoritma *Canny Edge Detection* untuk mendeteksi tepi pelat nomor dan *Tesseract OCR* (*Optical Character Recognition*) untuk mengekstraksi teks dari citra. Tujuan penelitian ini adalah mengeksplorasi efektivitas metode tersebut dalam mendeteksi dan mengenali pelat nomor kendaraan roda dua di lingkungan UINSU, serta memberikan alternatif solusi terhadap permasalahan sistem pemeriksaan manual. Dataset yang digunakan berjumlah 125 citra kendaraan roda dua yang diambil menggunakan kamera telepon genggam. Hasil penelitian menunjukkan bahwa sebelum diterapkannya *post-processing*, OCR menghasilkan rata-rata akurasi karakter sebesar 81,60% dengan CER 18,40%, sedangkan setelah *post-processing* akurasi meningkat menjadi 82,49% dan CER menurun menjadi 17,51%. Hasil ini menegaskan bahwa koreksi berbasis aturan mampu memperbaiki kesalahan pembacaan karakter, meskipun peningkatannya bersifat moderat dan belum sepenuhnya mengatasi kasus kegagalan deteksi pada beberapa citra. Temuan ini menjadi dasar penyelesaian terhadap permasalahan antrean dan keterbatasan sumber daya manusia, sekaligus menjadi pijakan dalam pengembangan sistem identifikasi kendaraan berbasis citra digital yang lebih luas di berbagai sektor.

ABSTRACT. Identity and vehicle ownership checks are an important routine carried out by security personnel at the State Islamic University of North Sumatra (UINSU) to ensure the legality of vehicles entering and exiting. However, the still manual inspection process of matching vehicle registration certificates (STNK) and license plates is often time-consuming and causes vehicle queues, especially during peak hours. This research applies a combination of the *Canny Edge Detection* algorithm to detect license plate edges and *Tesseract OCR* (*Optical Character Recognition*) to extract text from images. The purpose of this research is to explore the effectiveness of this method in detecting and recognizing two-wheeled vehicle license plates within the UINSU environment, and to provide an alternative solution to the problems of the manual inspection system. The dataset used consists of 125 images of two-wheeled vehicles taken using a mobile phone camera. The research results show that before *post-processing* was applied, OCR produced an average character accuracy of 81.60% with a CER of 18.40%, while after *post-processing*, the accuracy increased to 82.49% and the CER decreased to 17.51%. These results confirm that rule-based correction is able to improve character reading errors, although the improvement is moderate and has not completely addressed cases of detection failure in some images. This finding serves as the basis for addressing queuing issues and human resource limitations, while also providing a foundation for the broader development of digital image-based vehicle identification systems across various sectors.



This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution-NonCommercial 4.0 International License. **Editorial of EULER:** Department of Mathematics, Universitas Negeri Gorontalo, Jln. Prof. Dr. Ing. B. J. Habibie, Bone Bolango 96554, Indonesia.

1. Pendahuluan

Seiring dengan meningkatnya mobilitas civitas akademika di lingkungan Universitas Islam Negeri Sumatera Utara (UINSU), kendaraan bermotor, baik roda dua maupun roda empat, silih berganti memasuki dan meninggalkan kawasan kampus setiap harinya. Kondisi ini mencerminkan tingginya intensitas aktivitas akademik sekaligus menghadirkan kebutuhan akan sistem keamanan yang dapat menjamin bahwa setiap kendaraan yang me-

lintas merupakan milik sah dari mahasiswa, dosen, maupun tenaga kependidikan. Saat ini, sistem keamanan kendaraan di lingkungan UINSU masih mengandalkan metode manual dalam proses verifikasi, yaitu dengan mencocokkan pelat nomor kendaraan dengan dokumen kepemilikan seperti Surat Tanda Nomor Kendaraan (STNK), yang tentu saja memakan waktu dan tenaga. Apabila arus kendaraan meningkat secara signifikan, terutama pada jam pulang kuliah, maka sistem pemeriksaan manual ini kerap kali menimbulkan antrean panjang yang berpotensi mengganggu ke-

*Penulis Korespondensi.

lancaran aktivitas kampus secara keseluruhan.

Meskipun metode manual selama ini dianggap cukup untuk kebutuhan dasar pengawasan, kenyataan di lapangan menunjukkan bahwa metode tersebut memiliki banyak keterbatasan yang dapat menimbulkan permasalahan jangka panjang apabila tidak segera ditangani dengan solusi yang lebih adaptif [1, 2]. Ketergantungan terhadap observasi visual petugas keamanan tidak hanya memperlambat proses pemeriksaan kendaraan, tetapi juga membuka peluang terjadinya kesalahan identifikasi maupun kelalaian pencatatan [3, 4]. Di era digital yang terus berkembang, kebutuhan akan sistem otomatisasi yang cepat, efisien, dan minim kesalahan menjadi semakin mendesak [5, 6]. Oleh karena itu, pemanfaatan teknologi, khususnya pada bidang pengolahan citra digital, menjadi pilihan strategis untuk menggantikan proses pemeriksaan konvensional melalui penerapan sistem pengenalan pelat nomor kendaraan secara otomatis [7–9].

Untuk mendukung otomatisasi tersebut, pengembangan sistem yang bertumpu pada teknologi pengolahan citra seperti algoritma *Canny Edge Detection* dan metode *Optical Character Recognition* (OCR) seperti *Tesseract* menjadi sangat relevan dan menjanjikan, mengingat kedua metode ini mampu bekerja secara sinergis dalam mengenali identitas pelat nomor kendaraan dari citra visual [3, 7, 9, 10]. TNKB (Tanda Nomor Kendaraan Bermotor), sebagai identitas resmi yang dikeluarkan pemerintah, bersifat unik dan wajib dimiliki oleh setiap kendaraan yang beroperasi di jalan raya, sesuai dengan ketentuan Peraturan Pemerintah Nomor 60 Tahun 2016 [1, 11]. Dengan menggabungkan proses deteksi tepi untuk menemukan area pelat nomor dan proses OCR untuk mengekstraksi karakter alfanumerik di dalamnya, maka sistem pengenalan pelat nomor dapat dibangun secara lebih akurat, cepat, dan terstandarisasi sehingga memperkuat keamanan lingkungan kampus [6, 8, 12, 13].

Berdasarkan berbagai penelitian terdahulu, penyelesaian teknologi ini telah menunjukkan potensi yang menjanjikan, meskipun masih memiliki sejumlah keterbatasan yang perlu diperbaiki dan disempurnakan. Sebagai contoh, penelitian [3] menggunakan *Canny Edge Detection* dan berhasil mencapai tingkat akurasi tinggi dalam mendeteksi tepi pelat nomor, namun tidak melanjutkan proses hingga tahap pengenalan karakter. Penelitian lain oleh [9] membandingkan metode *Convolutional Neural Network* (CNN) dan *Tesseract OCR*, dan hasilnya menunjukkan bahwa *Tesseract* lebih unggul dalam mengenali karakter-karakter yang mirip, seperti angka “1” dan “7” atau “0” dan huruf “D”. Di sisi lain, penelitian [14] mengombinasikan deteksi tepi dengan koreksi kemiringan gambar melalui *Planar Homography* dan mengaplikasikannya pada citra video untuk meningkatkan presisi segmentasi. Namun demikian, sebagian besar penelitian ini masih terbatas pada kendaraan roda empat dan belum menerapkan segmentasi otomatis seperti *bounding box* dan *Region of Interest* (ROI) yang sangat penting untuk sistem yang sepenuhnya otomatis.

Dengan mempertimbangkan keterbatasan tersebut, penelitian ini diarahkan untuk mengembangkan model pengenalan pelat nomor kendaraan roda dua secara otomatis, yang dalam praktiknya memiliki tantangan tersendiri karena ukuran pelat yang lebih kecil, variasi posisi pengambilan gambar, dan keragaman kondisi pencahayaan di lapangan. Model yang dirancang menggabungkan algoritma *Canny Edge Detection* untuk mendeteksi batas pelat nomor dengan *Tesseract OCR* sebagai alat untuk mengeks-

trak dan mengenali karakter dalam citra. Sebelum tahap deteksi dilakukan, diterapkan serangkaian metode *preprocessing* yang mencakup *grayscale* untuk menyederhanakan informasi warna dan menurunkan kompleksitas komputasi, *noise reduction* untuk menghilangkan gangguan visual tanpa menghapus detail tepi, *contrast enhancement* untuk mempertegas perbedaan antara karakter dan latar belakang, serta *binarization* untuk memisahkan karakter dari latar belakang agar citra lebih optimal sebagai masukan bagi OCR. Untuk meningkatkan efisiensi dan akurasi proses, model ini juga dilengkapi dengan fitur otomatisasi seperti *bounding box* dan ROI untuk menyorot lokasi pelat nomor secara presisi. *Dataset* citra kendaraan yang digunakan dalam penelitian ini diperoleh langsung dari lingkungan UINSU menggunakan kamera ponsel dalam berbagai kondisi nyata di lapangan.

Dengan mengembangkan dan menguji model ini melalui simulasi dan eksperimen berbasis data, penelitian ini bertujuan untuk mengevaluasi secara menyeluruh performa penggabungan algoritma yang digunakan, serta menilai sejauh mana metode ini mampu menggantikan peran pemeriksaan manual yang selama ini diterapkan dalam sistem keamanan kendaraan UINSU. Selain memberikan solusi praktis terhadap permasalahan antrean dan keterbatasan sumber daya manusia, hasil penelitian ini diharapkan dapat menjadi pijakan awal dalam pengembangan sistem identifikasi kendaraan berbasis citra digital yang lebih luas, baik di lingkungan pendidikan maupun pada sektor-sektor lain seperti perkantoran, perumahan, atau fasilitas publik yang membutuhkan efisiensi dalam manajemen lalu lintas dan keamanan kendaraan.

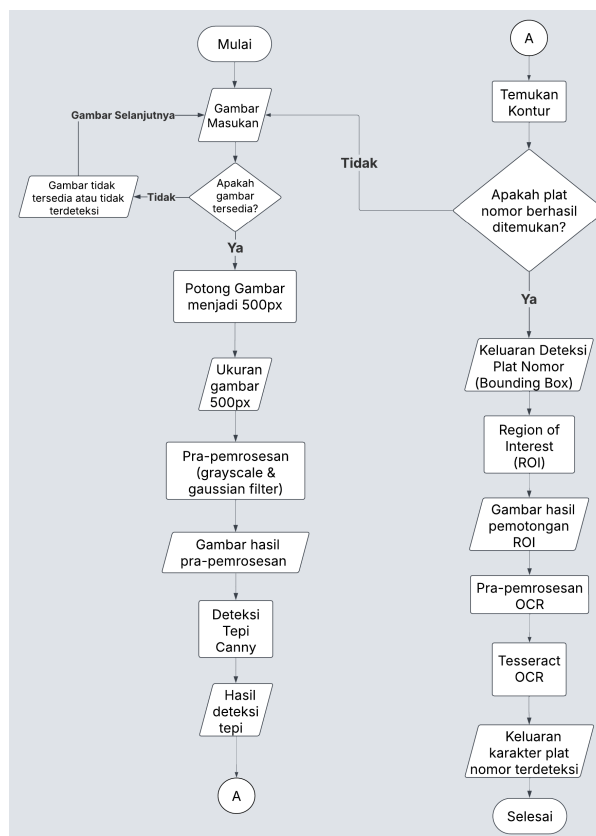
2. Metode

Penelitian ini menggunakan metode kuantitatif dan berlokasi di kawasan UINSU, yang secara sistematis menerapkan serangkaian tahapan untuk menganalisis kinerja model yang dikembangkan dalam mendeteksi dan mengenali pelat nomor kendaraan bermotor sebagai upaya mengatasi permasalahan pemeriksaan parkir manual, sebagaimana digambarkan pada Gambar 1 yang menunjukkan alur proses mulai dari akuisisi citra hingga pengenalan karakter menggunakan *Tesseract OCR* [3, 15].

Implementasi model dilakukan menggunakan *Python* sebagai bahasa pemrograman utama melalui *Visual Studio Code*, dengan lingkungan kerja (*environment*) yang dikelola oleh *Anaconda* untuk mendukung pengaturan dependensi dan replikasi eksperimen secara terstandar [16, 17]. Langkah-langkah penelitian dibagikan dalam *flowchart* pada Gambar 1.

2.1. Image Acquisition

Image acquisition, atau akuisisi citra, merupakan proses fundamental dalam pengolahan citra digital yang melibatkan penangkapan representasi digital dari objek atau pemandangan fisik menggunakan perangkat pencitraan, sehingga dunia fisik dapat diubah menjadi format digital yang siap diproses oleh komputer [6]. Dalam penelitian ini, proses akuisisi citra dilakukan dengan mengumpulkan total 125 citra yang secara spesifik merupakan gambar atau foto kendaraan roda dua, yang diperoleh langsung dari lokasi penelitian menggunakan kamera *handphone* sebagai perangkat penangkap citra [1]. Citra-citra tersebut kemudian dijadikan *dataset* utama yang akan diproses dan dianalisis oleh model yang dikembangkan [3].



Gambar 1. Flowchart langkah-langkah penelitian

2.2. Data Preprocessing

Data preprocessing adalah tahapan esensial dalam pemrosesan citra digital yang bertujuan untuk mentransformasi data mentah menjadi format yang lebih terstruktur, bersih, dan optimal untuk analisis lebih lanjut, sehingga dapat meningkatkan akurasi dan keandalan model yang akan dikembangkan. Dalam penelitian ini, pra-pemrosesan data citra yang telah diakuisisi meliputi beberapa operasi krusial yang dirancang untuk mendukung efektivitas algoritma deteksi tepi dan pengenalan karakter [13].

2.2.1. Grayscale

Konversi citra berwarna RGB menjadi format *grayscale* dilakukan untuk menyederhanakan representasi citra dengan menghilangkan informasi warna, yang secara signifikan mengurangi kompleksitas data dan membuatnya lebih efisien untuk proses deteksi tepi karena algoritma cenderung bekerja lebih efektif pada citra keabuan [18].

2.2.2. Peredaman Noise

Untuk meningkatkan kualitas citra dengan mengurangi gangguan yang tidak diinginkan tanpa mengorbankan integritas tepi objek, diterapkan teknik *noise reduction* menggunakan *Gaussian Blur* dengan kernel 3×3 dan $\sigma = 1.2$, *median blur* (untuk *pre-processing ROI*) dengan kernel 3×3 , serta *bilateral filter* (untuk *pre-processing ROI*) dengan parameter (11, 17, 17). *Median blur* efektif untuk mengurangi *salt-and-pepper noise*, sedangkan *bilateral filter* berfungsi menjaga tepi tetap tajam sambil menghaluskan tekstur di dalam area karakter [19].

2.2.3. Peningkatan Kontras

Peningkatan kontras lokal pada ROI dilakukan melalui teknik *Contrast Limited Adaptive Histogram Equalization (CLAHE)* dengan parameter *clipLimit* = 2.0 dan *tileGridSize* = (8, 8) sebagaimana digunakan oleh [20] untuk menonjolkan perbedaan antara karakter pelat nomor dan latar belakangnya. Berdasarkan penelitian tersebut, konfigurasi parameter ini efektif meningkatkan kontras tanpa menghilangkan detail penting, sehingga karakter lebih mudah dikenali oleh model OCR [6].

2.2.4. Binerisasi Citra

Metode *thresholding* Otsu digunakan untuk mengubah citra menjadi format biner (hitam–putih), yang secara efektif memisahkan objek (karakter pelat nomor) dari latar belakangnya. Dalam implementasi ini digunakan nilai ambang awal *low_thresh* = 30 dan *high_thresh* = 90, serta kondisi inversi berdasarkan jumlah piksel non-nol, yaitu jika $cv.countNonZero(thr) < thr.size/2$, maka citra diinversi menggunakan $cv.bitwise_not(thr)$ untuk memastikan karakter berwarna putih di atas latar belakang hitam. Pendekatan ini memberikan kondisi optimal bagi masukan *Tesseract OCR* [18].

2.3. Modeling

Modeling dalam penelitian ini merupakan tahapan perancangan dan implementasi sistem komputasi yang bertujuan untuk mendeteksi dan mengenali pelat nomor kendaraan bermotor secara otomatis, yang melibatkan integrasi berbagai algoritma pengolahan citra. Model yang dikembangkan memanfaatkan algoritma *Canny Edge Detection* untuk identifikasi area pelat nomor

dan *Tesseract OCR* untuk ekstraksi karakter, sebagai solusi atas tantangan pemeriksaan parkir manual [3, 7, 8].

2.3.1. Canny Edge Detection

Canny Edge Detection adalah algoritma deteksi tepi yang dikenal karena kemampuannya dalam meminimalkan tingkat kesalahan deteksi dan menghasilkan tepi yang optimal pada citra digital. Algoritma ini dirancang berdasarkan tiga kriteria utama: mendeteksi tepi dengan baik, melokalisasi tepi dengan presisi, dan memberikan respons tepi yang jelas tanpa duplikasi [12, 19, 21]. Penerapan algoritma Canny dalam penelitian ini melibatkan beberapa tahapan sebagai berikut:

1. *Smoothing* citra menggunakan *Gaussian Derivative Kernel* untuk mereduksi *noise* [13, 19]:

$$G(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right). \quad (1)$$

Setiap piksel hasil konvolusi diperoleh melalui:

$$B(i, j) = \sum_{p=0}^{n-1} \sum_{q=0}^{n-1} G(p, q) A\left(i + p - \frac{n-1}{2}, j + q - \frac{n-1}{2}\right) \quad (2)$$

2. *Gradient calculation* untuk mengidentifikasi perubahan intensitas piksel menggunakan operator Sobel. Gradien horizontal (G_x) dan vertikal (G_y) digabungkan menjadi:

$$M = \sqrt{G_x^2 + G_y^2}. \quad (3)$$

Arah gradien dihitung dengan:

$$\theta = \arctan2(G_y, G_x) \times \frac{180}{\pi}. \quad (4)$$

3. *Non-maximum suppression* untuk menipiskan tepi dan mempertahankan hanya nilai gradien maksimum:

$$M'(i, j) = \begin{cases} M(i, j), & \text{jika } M(i, j) \geq q \text{ dan } M(i, j) \geq r \\ 0, & \text{lainnya} \end{cases}. \quad (5)$$

4. *Double thresholding* untuk memisahkan tepi kuat dan lemah, kemudian dilanjutkan dengan *edge tracking by hysteresis* untuk memastikan kontinuitas tepi [3, 21]:

$$M'(i, j) = \begin{cases} 255, & \text{jika } M(i, j) \geq T_{\text{high}} \\ 255, & \text{jika } T_{\text{low}} \leq M(i, j) < T_{\text{high}} \\ & \text{dan terhubung ke tepi kuat} \\ 0, & \text{lainnya} \end{cases}. \quad (6)$$

2.3.2. ROI Extraction

ROI Extraction adalah proses krusial setelah deteksi tepi, yang berfokus pada isolasi area pelat nomor kendaraan yang telah terdeteksi untuk mempersiapkannya pada tahap pengenalan karakter [3]. Proses ini diawali dengan penggunaan *Bounding Box* secara otomatis, yang berfungsi untuk menemukan dan menandai lokasi pelat nomor berdasarkan kontur yang teridentifikasi dari hasil *Canny Edge Detection* [22]. Setelah *bounding box* berhasil mengidentifikasi area kandidat pelat nomor, fitur ROI kemudian digunakan untuk menyurut dan memotong bagian citra

yang hanya berisikan pelat nomor tersebut [7]. Ekstraksi ROI ini bertujuan untuk meminimalisasi *noise* dan informasi yang tidak relevan di luar area pelat nomor, sehingga *Tesseract OCR* dapat bekerja dengan lebih efisien dan akurat dalam mengekstraksi karakter [4, 23].

2.3.3. Tesseract OCR

Tesseract OCR adalah sebuah *open-source library* yang dikembangkan oleh Google, dan didesain khusus untuk mengekstraksi teks dari citra dengan akurasi tinggi [7, 8]. Dalam penelitian ini, *Tesseract OCR* diimplementasikan untuk mengenali karakter alfa-numerik pada pelat nomor kendaraan setelah area pelat nomor berhasil diekstraksi sebagai ROI [9].

Arsitektur *Tesseract* melibatkan beberapa tahap, termasuk *image preprocessing* yang mengubah citra menjadi biner untuk memisahkan karakter dari latar belakang, *segmentation* untuk mengidentifikasi setiap karakter, dan *word recognition* yang menggunakan algoritma *Long Short-Term Memory* (LSTM) untuk pengenalan karakter [10]. *Tesseract* dirancang untuk mengenali teks berwarna putih dengan latar belakang hitam atau sebaliknya, sehingga hasil pra-pemrosesan ROI yang telah dibinerisasi sangat mendukung kinerjanya [8].

Konfigurasi *Tesseract* pada penelitian ini disesuaikan dengan *whitelist* karakter huruf kapital A–Z dan angka 0–9, serta menggunakan mode segmentasi halaman tunggal untuk mengoptimalkan proses pengenalan pelat nomor Indonesia [10, 24].

2.3.4. Post Processing

Post-processing berfungsi untuk memperbaiki kesalahan OCR pada pelat nomor dengan menormalkan karakter yang sering tertukar seperti $0 \leftrightarrow O$, $1 \leftrightarrow I$, dan $5 \leftrightarrow S$ berdasarkan posisi huruf dan angka. Dalam penelitian ini, proses dilakukan dengan membersihkan teks, mencocokkannya dengan pola *regex* pelat nomor Indonesia, kemudian membagi hasilnya menjadi segmen huruf–angka–huruf untuk dikoreksi sesuai konteks. Apabila hasil belum valid, sistem melakukan perbaikan tambahan seperti pemotongan karakter, substitusi alternatif, atau pengecekan variasi terdekat. Parameter yang digunakan meliputi pola *regex* format pelat, daftar pasangan karakter yang dikoreksi, serta indikator validasi hasil akhir [25].

2.4. Evaluation

Evaluation adalah proses sistematis yang digunakan untuk menilai dan menganalisis kinerja model berdasarkan data yang telah diproses [8, 26]. Dalam penelitian ini, evaluasi algoritma *Canny Edge Detection* dilakukan melalui analisis visual terhadap citra hasil pemrosesan, sedangkan kinerja *Tesseract OCR* diukur menggunakan *Character Accuracy* dan *Character Error Rate*. *Character Accuracy* digunakan untuk menghitung persentase karakter yang dikenali dengan benar oleh OCR dibandingkan dengan total karakter dalam teks asli, sedangkan *Character Error Rate* mengukur proporsi kesalahan karakter yang dilakukan OCR terhadap total karakter dalam teks referensi, dengan nilai yang lebih rendah menunjukkan kinerja yang lebih baik [10, 27].

$$\text{Character Accuracy} = \left(\frac{\text{Jumlah Karakter Benar}}{\text{Total Karakter}} \right) \times 100\%,$$

$$\text{Character Error Rate} = \frac{S + D + I}{N}$$

Keterangan:

- S = jumlah substitusi,
- D = jumlah penghapusan,
- I = jumlah penyisipan,
- N = total karakter dalam teks referensi.

3. Hasil dan Pembahasan

3.1. Hasil

Berdasarkan hasil pengujian model pendeteksian dan pengenalan pelat nomor kendaraan roda dua menggunakan algoritma *Canny Edge Detection* dan *Tesseract OCR* pada 125 citra uji, diperoleh tingkat *character accuracy* sebesar 81,56% dengan *character error rate* (CER) sebesar 18,44%. Namun, setelah menerapkan *post-processing* menggunakan *rule-based character normalization*, akurasi meningkat menjadi 82,88% dan CER menurun menjadi 17,12%.

3.1.1. Data Preprocessing

Dataset penelitian ini merepresentasikan variasi kondisi visual di lingkungan UINSU, di mana setiap citra menunjukkan perbedaan bentuk fisik pelat nomor seperti variasi *font*, intensitas pencahayaan, dan sudut pengambilan gambar yang memengaruhi hasil pengenalan karakter oleh model. Salah satu citra yang dijadikan sampel untuk tahapan pra-pemrosesan, deteksi tepi, ekstraksi ROI, hingga evaluasi akhir pengenalan karakter adalah pelat nomor BB 2430 HM, yang digunakan sebagai Citra Sampel (A). Citra Sampel (A) yang digunakan pada penelitian ini disajikan pada Gambar 2.



Gambar 2. Citra sampel (A)

Transformasi awal citra melalui konversi *grayscale* menghasilkan distribusi intensitas piksel yang menggambarkan variasi tajam antara area karakter dan latar belakang, yang sangat penting dalam tahap deteksi tepi selanjutnya. Berdasarkan Citra Sampel (A) berukuran 8×8 piksel, nilai intensitas *grayscale* yang dihasilkan berkisar dari 32 hingga 251, dengan pola distribusi yang me-

nunjukkan bahwa area karakter berada dalam rentang menengah hingga terang.

Sebagai contoh, piksel pada koordinat (0,0) dengan komposisi RGB (82, 86, 89) menghasilkan nilai *grayscale* sebesar 85, sedangkan piksel pada koordinat (0,4) dengan RGB (140, 136, 151) menghasilkan nilai 139. Proses konversi ini mengikuti rumus ITU-R BT.601 dan menggunakan pembulatan *truncation (flooring)*, sehingga menghasilkan deret nilai seperti 85, 76, 80, 99, 139, 201, hingga 251. Hasil perhitungan nilai *grayscale* secara detail untuk citra sampel ini disajikan pada Tabel 1 dan representasi visual citra setelah proses *grayscale* ditunjukkan pada Gambar 3.

Tabel 1. Hasil perhitungan *grayscale* citra sampel (A)

85	76	80	99	139	201	244	251
39	40	44	58	88	166	235	236
32	43	56	73	141	235	251	249
41	65	71	129	212	251	250	250
64	73	118	200	248	250	250	249
87	112	193	243	250	250	250	241
121	181	234	250	250	250	243	222
176	222	250	250	250	250	222	165



Gambar 3. Hasil *grayscale*

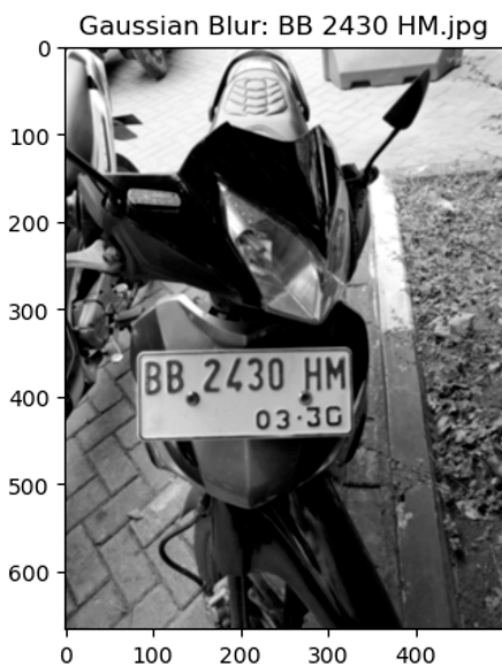
Penerapan *Gaussian blur* dengan kernel 3×3 dan $\sigma = 1.2$ berhasil mereduksi fluktuasi lokal pada citra tanpa menghilangkan struktur spasial utama dari karakter. Bobot kernel Gaussian yang digunakan telah dinormalisasi agar jumlah total elemennya bernilai 1, dengan nilai pusat kernel sebesar 0.1717 dan nilai pada keempat sudut sebesar 0.0857, sehingga memastikan bahwa efek *smoothing* tersebar secara seimbang ke segala arah.

Hasil konvolusi pada baris pertama citra, misalnya, menunjukkan transformasi nilai dari rentang 85–251 menjadi 70–250. Piksel pada koordinat (0,0) yang sebelumnya bernilai 85 berubah menjadi 70 setelah proses konvolusi, sementara piksel (0,6) meningkat menjadi 228 dari nilai aslinya 244. Perubahan ini menunjukkan bahwa *Gaussian filter* secara efektif mengurangi *local*

noise sambil tetap mempertahankan transisi tepi yang penting bagi proses deteksi karakter. Matriks hasil konvolusi *Gaussian filter* dapat dilihat pada **Tabel 2**, sedangkan hasil visual setelah penerapan *Gaussian Blur* disajikan pada **Gambar 4**.

Tabel 2. Matriks hasil *Gaussian filter* citra sampel (A)

70	68	73	93	133	186	228	246
51	53	61	83	128	187	229	244
40	48	63	95	150	206	238	245
50	62	90	138	195	234	249	248
69	89	132	186	228	247	247	247
99	130	179	223	245	249	246	238
141	175	217	242	244	247	234	216
185	213	239	249	244	239	209	177



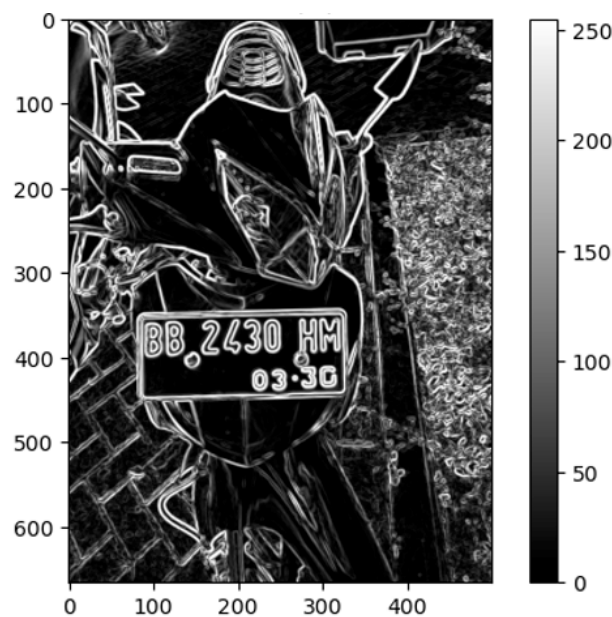
Gambar 4. Hasil *Gaussian blur*

3.1.2. Modeling

(a) Canny Edge Detection

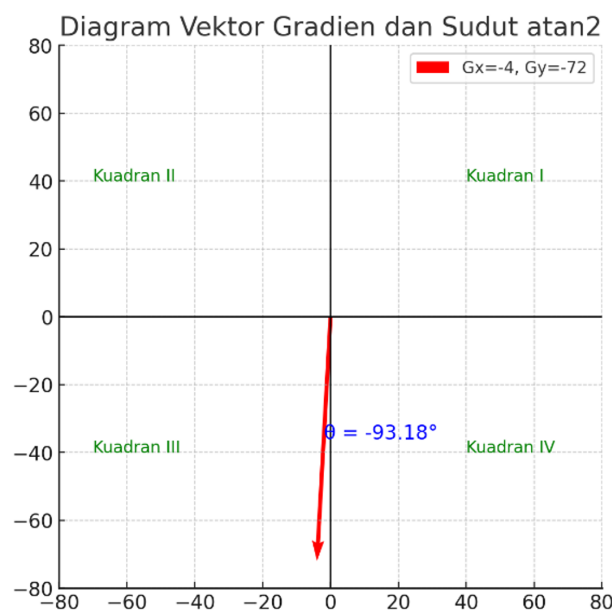
Hasil pemodelan *Canny Edge Detection* menunjukkan bahwa proses deteksi tepi pada citra kendaraan berjalan optimal dalam mengidentifikasi area pelat nomor sebagai kandidat utama untuk diekstraksi. Proses komputasi gradien menghasilkan nilai magnitudo yang merepresentasikan kekuatan tepi pada masing-masing piksel, dengan sebaran nilai intensitas gradien yang konsisten pada area berkontras tinggi, khususnya di sekitar batas fisik pelat nomor. Hasil gradien dari delapan baris dan delapan kolom piksel awal ditunjukkan pada **Gambar 5**, yang memperlihatkan bahwa area dengan tepi yang tegas menghasilkan nilai gradien yang jauh lebih tinggi dibandingkan dengan latar belakang citra.

Salah satu piksel dengan nilai gradien signifikan adalah piksel pada koordinat (2, 2), dengan magnitudo mencapai sekitar 234, yang berasal dari gradien horizontal (G_x) sebesar 200 dan gradien vertikal (G_y) sebesar 122. Informasi arah gradien dari piksel-piksel ini kemudian digunakan untuk menentukan orientasi tepi pada citra. Perhitungan arah gradien untuk piksel (1, 1),



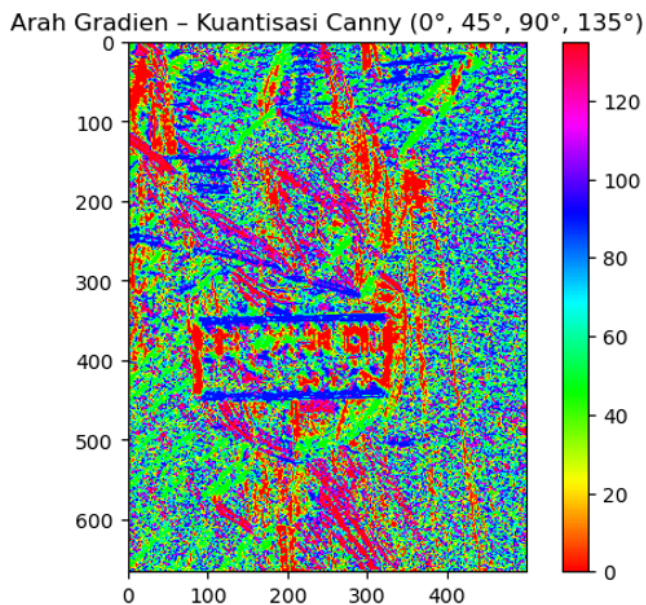
Gambar 5. Hasil proses magnitudo gradien

misalnya, menghasilkan sudut sekitar 135° , yang sesuai dengan orientasi diagonal dan dikuantisasi ke dalam salah satu dari empat arah utama. Proses penghitungan arah ini ditunjukkan pada **Gambar 6** untuk representasi piksel tunggal, dan pada **Gambar 7** untuk distribusi arah gradien pada area yang lebih luas.

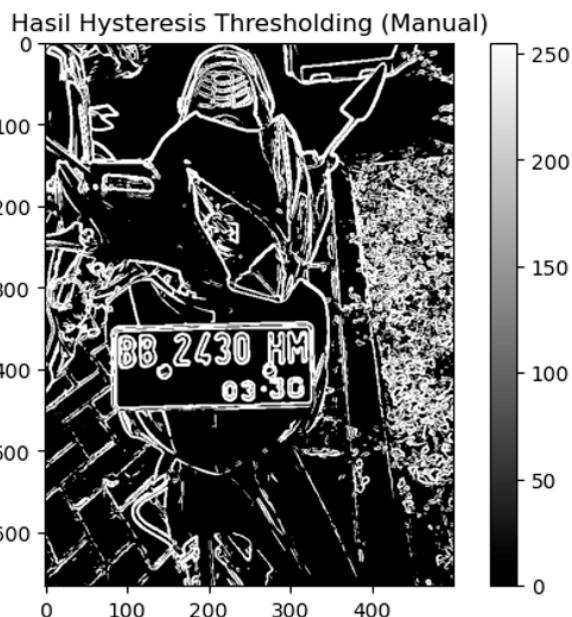


Gambar 6. Arah gradien pada piksel (1,1)

Penerapan *non-maximum suppression* berhasil menyaring piksel yang tidak relevan berdasarkan nilai lokal maksimum dalam arah tepi masing-masing. Dari hasil perbandingan nilai intensitas, hanya piksel-piksel dominan seperti pada koordinat (1, 3), (1, 4), dan (1, 5) yang dipertahankan sebagai bagian dari tepi dengan nilai magnitudo mencapai 255. Proses ini berhasil mempertipis tepi dan meningkatkan kejelasan struktur pelat nomor sebagaimana terlihat pada **Gambar 8**. Tahapan *non-maximum suppression* tersebut kemudian dilanjutkan dengan *double thresholding*



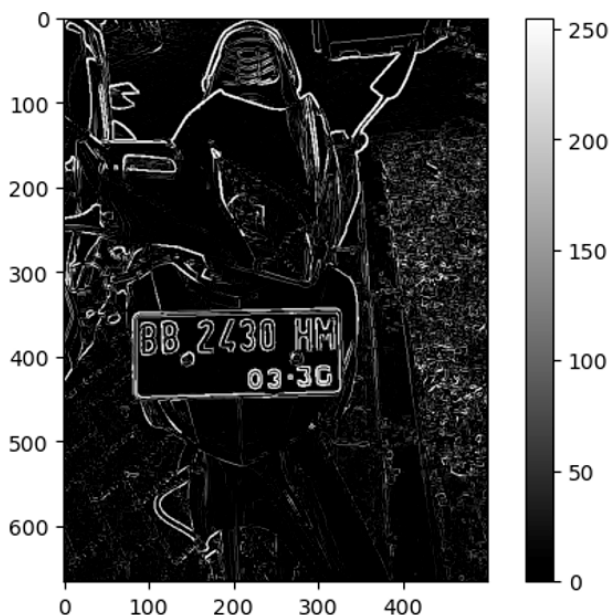
Gambar 7. Arah gradien



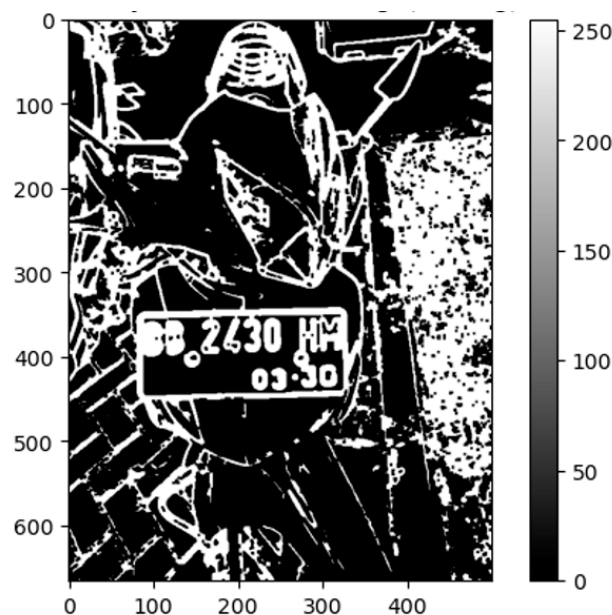
Gambar 9. Hasil proses hysteresis

dan *edge tracking by hysteresis*, yang secara efektif mengklasifikasikan tepi menjadi tepi kuat dan tepi lemah berdasarkan nilai ambang batas bawah 30 dan ambang batas atas 90. Sebagai contoh, piksel pada koordinat (1, 1) hingga (1, 5) diklasifikasikan sebagai tepi kuat karena memiliki nilai di atas 90 dan berkontribusi terhadap struktur tepi akhir. Hasil dari tahap ini ditunjukkan pada Gambar 9, yang memperlihatkan kontinuitas tepi pelat nomor secara utuh dan bersih, dengan penghilangan *noise* yang cukup efektif.

bagian-bagian pelat nomor yang terputus tanpa merusak struktur utamanya. Output dari transformasi morfologi ini ditampilkan pada Gambar 10, yang memperlihatkan bentuk pelat nomor yang lebih utuh dan siap untuk diekstraksi sebagai ROI.



Gambar 8. Hasil proses non-maximum suppression



Gambar 10. Hasil morfologi

Untuk meningkatkan integritas bentuk area pelat yang telah terdeteksi, diterapkan operasi morfologi dengan pendekatan *closing*, yaitu kombinasi *dilation* dan *erosion* menggunakan *structuring element* berukuran 3×3 . Proses ini memperhalus hasil deteksi tepi dengan menutup celah-celah kecil dan menyambungkan

(b) Tesseract OCR

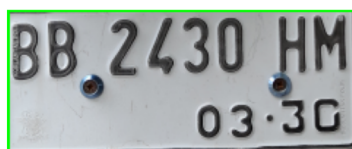
Hasil pemodelan *Tesseract OCR* menunjukkan bahwa area pelat nomor berhasil dideteksi dan ditandai secara otomatis menggunakan pendekatan *bounding box* berdasarkan hasil kontur dari proses sebelumnya. Area tersebut diklasifikasikan sebagai kandidat ROI berdasarkan kriteria rasio dimensi dan ukuran minimum yang telah ditentukan. Area pelat yang terdeteksi ditunjukkan pada Gambar 11, yang memperlihatkan letak *bounding box* berwarna hijau mengelilingi area pelat nomor secara presisi

pada beberapa sampel citra. Hal ini menandakan bahwa pemisahan area teks dari latar belakang berjalan optimal pada sebagian besar data uji.

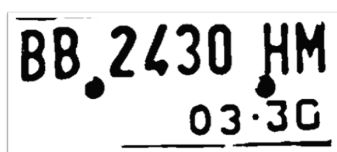


Gambar 11. Hasil boundingbox area plat nomor ditemukan

Setelah area plat ditandai, hasil pemotongan citra (ROI) menunjukkan isolasi visual yang bersih terhadap area plat nomor. ROI yang dihasilkan memenuhi syarat untuk diproses lebih lanjut tanpa menyertakan bagian latar belakang kendaraan yang tidak relevan. Pada Gambar 12 diperlihatkan contoh ROI hasil pemodelan yang menunjukkan batas teks plat nomor secara jelas dalam satu baris horizontal. Proses preprocessing yang diterapkan pada ROI menunjukkan peningkatan kualitas citra untuk kebutuhan ekstraksi karakter. Hasil preprocessing ditunjukkan pada Gambar 13, menunjukkan bahwa karakter pada plat nomor berhasil dipisahkan secara kontras terhadap latar belakang. Citra hitam-putih hasil thresholding menampilkan karakter berwarna putih pada latar belakang gelap, sesuai dengan preferensi input yang optimal bagi Tesseract OCR. Keberhasilan preprocessing ini menjadi faktor kunci dalam meningkatkan akurasi pembacaan karakter oleh model OCR.



Gambar 12. ROI plat nomor



Gambar 13. Hasil preprocessing pada ROI

3.1.3. Post Processing

Untuk gambar "BB 2430 HM", plat nomor aslinya adalah "BB2430HM". Hasil OCR mentah yang didapatkan adalah "BB2430HM9", namun setelah melalui proses post-processing rule-based correction, teks OCR diperbaiki menjadi "BB2430HM", dan hasil post-processing ini dianggap valid sesuai dengan pola format plat nomor yang diharapkan.

3.1.4. Evaluation

Untuk mengukur performa model dalam mengenali karakter pelat nomor kendaraan, evaluasi dilakukan menggunakan dua metrik utama, yaitu *Character Accuracy (CA)* dan *Character Error Rate (CER)*. Proses pengujian dijalankan menggunakan konfigurasi Tesseract OCR yang telah disesuaikan dengan format standar plat nomor Indonesia, sehingga hasil pengenalan dapat lebih relevan dengan karakteristik data sebenarnya. Seluruh keluaran pengenalan karakter dan nilai metrik disimpan dalam struktur data terorganisasi untuk memudahkan analisis hasil secara terukur dan replikatif pada tahap selanjutnya. Perbandingan antara hasil pembacaan karakter dari model OCR terhadap nilai referensi atau *Ground Truth (GT)* menjadi dasar penilaian terhadap kinerja model. Sebagai ilustrasi, pelat dengan nilai referensi "BB2430HM" dikenali oleh model OCR sebagai "BB2430HM9", yang menunjukkan adanya satu penyisipan karakter tambahan di bagian akhir hasil pembacaan. Hasil evaluasi ekstraksi karakter pada contoh citra ini disajikan pada Tabel 3.

Tabel 3. Evaluasi ekstraksi karakter pada contoh citra

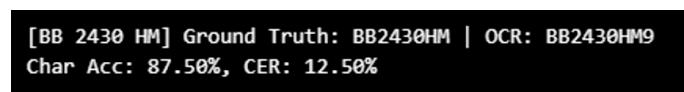
Ground Truth (GT)	OCR	Char Acc	CER
BB2430HM	BB2430HM9	87,50%	12,50%

Panjang karakter dari teks referensi (GT) adalah $N = 8$, dan model menghasilkan satu penyisipan karakter tanpa adanya substitusi atau penghapusan. Maka, jumlah karakter yang dikenali dengan benar adalah: Jumlah Karakter Benar = $N - (S + D + I) = 8 - (0 + 0 + 1) = 7$, sehingga diperoleh perhitungan sebagai berikut:

$$CA = \frac{\text{Jumlah Karakter Benar}}{N} \times 100\% = \frac{7}{8} \times 100\% = 87,50\%$$

$$CER = \frac{S + D + I}{N} \times 100\% = \frac{1}{8} \times 100\% = 12,50\%$$

Hasil evaluasi ini menunjukkan bahwa model OCR mengenali tujuh dari delapan karakter dengan benar, dengan tingkat ketepatan pembacaan sebesar 87,50% dan tingkat kesalahan sebesar 12,50%. Evaluasi ini menegaskan bahwa meskipun performa OCR cukup akurat, tetap terdapat potensi kesalahan, terutama pada kasus penyisipan karakter yang tidak terdapat dalam teks referensi aslinya. Hasil evaluasi contoh citra disajikan pada Gambar 14.



Gambar 14. Hasil evaluasi contoh citra

Berdasarkan seluruh hasil pengujian terhadap dataset, sejumlah citra menunjukkan performa maksimal dengan akurasi pengenalan karakter mencapai 100%, yang mana lima contoh citra

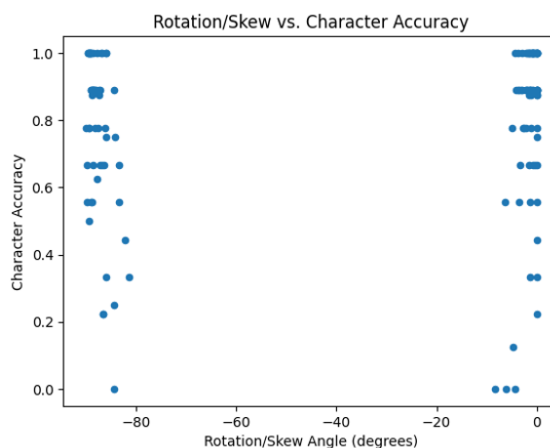
Tabel 4. Contoh 5 citra hasil pemodelan terbaik

Field	BK 2177 ACS	BK 2200 WAI	BK 2417 AKY	BK 2543 ZAK	BK 2546 RBM
Citra Resize					
Hasil Canny (Morfologi)					
Bounding Box					
ROI					
Ground Truth (GT)	BK2177ACS	BK2200WAI	BK2417AKY	BK2543ZAK	BK2546RBM
OCR Text	BK2177ACS	BK2200WAI	BK2417AKY	BK2543ZAK	BK2546RBM
Character Accuracy	1.000 (100%)	1.000 (100%)	1.000 (100%)	1.000 (100%)	1.000 (100%)
CER (Error Rate)	0.000 (0%)	0.000 (0%)	0.000 (0%)	0.000 (0%)	0.000 (0%)
Rotasi / Kemiringan	-86.855042	-0.000000	-0.763898	-89.301308	-1.562225
Kejelasan Tepi (StdDev)	44.019802	46.028233	62.238363	44.022111	64.722877
Perbandingan Jarak Relatif	0.083904	0.088865	0.064865	0.058261	0.085060
Pencahayaan (Mean Intensity)	138.622405	160.576237	212.998194	175.902582	179.168332

dengan hasil terbaik ditampilkan pada **Tabel 4** sebagai pembandingan terhadap citra yang gagal dikenali. Tabel ini juga berfungsi sebagai dasar analisis untuk mengkorelasikan metrik kinerja OCR, seperti CER, dengan faktor-faktor kondisi citra yang terukur, yaitu Rotasi Kemiringan, Kejelasan Tepi, Perbandingan Jarak Relatif, dan Pencahayaan.

Sebaliknya, model juga menghasilkan pengenalan karakter yang sangat rendah pada beberapa citra, termasuk kasus kegagalan total, sehingga lima citra dengan performa terendah ditampilkan pada **Tabel 5** sebagai representasi dari tantangan yang masih dihadapi dalam proses pengenalan karakter otomatis. **Tabel 5** berfungsi sebagai dasar untuk menganalisis penyebab kesalahan model dengan mengkorelasikan metrik kinerja OCR, seperti CER, terhadap faktor-faktor kondisi citra yang terukur, yaitu Rotasi Kemiringan, Kejelasan Tepi, Perbandingan Jarak Relatif, dan Pencahayaan.

Hasil rekapitulasi pada **Tabel 6** menunjukkan bahwa pada kategori Rotasi/Kemiringan, sub-kategori *Low Skew* memiliki akurasi karakter sebesar 0.809193, *Medium Skew* sebesar 0.698509, dan *High Skew* sebesar 0.831680. Pada kategori Kejelasan Tepi, *Low Clarity* memperoleh akurasi 0.839286, *Medium Clarity* sebesar 0.726965, dan *High Clarity* sebesar 0.773810. Untuk kategori Jarak Relatif, sub-kategori *Close* mencapai akurasi 0.798611, *Medium Distance* sebesar 0.768970, dan *Far* sebesar 0.773479. Sementara pada kategori Pencahayaan, *Low Lighting* menunjukkan



Gambar 15. Scatter plots metrik kuantitatif vs. character accuracy

akurasi 0.687500, *Medium Lighting* sebesar 0.861789, dan *High Lighting* sebesar 0.793981.

Untuk memahami hubungan antara kondisi citra dan kinerja OCR, **Gambar 15** menunjukkan hubungan antara sudut kemiringan plat nomor dan akurasi karakter hasil OCR. Setiap titik mewakili satu citra, dengan sumbu horizontal menunjukkan derajat rotasi yang berkisar antara sekitar -90° hingga 0° , sedangkan

Tabel 5. Contoh 5 citra hasil pemodelan terburuk

Field	BK 5593 AGM	BK 4195 AHU	BK 2150 AJF	BK 5111 IDY	BL 4325 IL
Citra Resize					
Hasil Canny (Morfologi)					
Bounding Box					
ROI					
Ground Truth (GT)	BK5593AGM	BK4195AHU	BK2150AJF	BK5111IDY	BL4325IL
OCR Text	493AM2083	JTBABK419	25UIF023	BKOITDY11	BL
Character Accuracy	0.000000 (0.0%)	0.111111 (11.1%)	0.222222 (22.2%)	0.222222 (22.2%)	0.250000 (25.0%)
CER (Error Rate)	1.000000 (100.0%)	0.888889 (88.9%)	0.777778 (77.8%)	0.777778 (77.8%)	0.750000 (75.0%)
Rotasi / Kemiringan	-4.370797	-8.386148	-86.520081	-86.478546	-84.359444
Kejelasan Tepi (StdDev)	62.141532	59.088167	56.960408	80.797544	69.474491
Perbandingan Jarak Relatif	0.070670	0.369369	0.086919	0.059189	0.083138
Pencahayaan (Mean Intensity)	115.599584	82.618683	146.371614	182.833739	166.161315

Tabel 6. Rekapitulasi akurasi per kategori (rotasi/kemiringan, kejelasan tepi, jarak relatif, pencahayaan)

Kategori	Sub-Kategori	Character Accuracy	CER	Interpretasi
Rotasi / Kemiringan	Low Skew	0.809193	0.190807	Sedang
	Medium Skew	0.698509	0.301491	Sedang
	High Skew	0.831680	0.168320	Sedang
Kejelasan Tepi	Low Clarity	0.839286	0.160714	Sedang
	Medium Clarity	0.726965	0.273035	Sedang
	High Clarity	0.773810	0.226190	Sedang
Jarak Relatif	Close	0.798611	0.201389	Sedang
	Medium Distance	0.768970	0.231030	Sedang
	Far	0.773479	0.226521	Sedang
Pencahayaan	Low Lighting	0.687500	0.312500	Sedang
	Medium Lighting	0.861789	0.138211	Sedang
	High Lighting	0.793981	0.206019	Sedang

Keterangan: Akurasi Bagus jika ≥ 0.88 , Jelek jika ≤ 0.68 , Sedang jika di antaranya.

sumbu vertikal menunjukkan nilai akurasi karakter yang berada dalam rentang 0.0 hingga 1.0.

3.2. Pembahasan

Hasil penelitian menunjukkan bahwa penerapan algoritma Canny Edge Detection dan Tesseract OCR memiliki kemampuan yang cukup baik dalam mengenali karakter pada pelat nomor kendaraan roda dua, dengan character accuracy rata-rata sebesar 81,6% dan character error rate sebesar 18,4%. Nilai tersebut membuktikan bahwa metode yang digunakan mampu melakukan identifikasi

si karakter secara efektif pada sebagian besar citra uji, meskipun masih ditemukan sejumlah kesalahan pengenalan pada kondisi tertentu seperti pencahayaan yang tidak seragam, kemiringan pelat, atau permukaan pelat yang kotor maupun reflektif. Ketidaktercapaian akurasi ini bersumber dari beberapa faktor, antara lain kegagalan segmentasi tepi pada area dengan iluminasi tidak merata—misalnya akibat refleksi kuat atau bayangan—yang menyebabkan terbentuknya tepi terputus atau tepi semu. Selain itu, distorsi geometrik akibat kemiringan pelat mengubah bentuk serta jarak antar karakter, sehingga batas antar-glyph menjadi

sulit dipisahkan. Variabilitas tipografi dan kondisi fisik pelat (aus atau kotor) juga menimbulkan kemiripan morfologis antar karakter (misalnya O–0, S–5, B–8), yang pada akhirnya memicu terjadinya substitusi maupun penyisipan karakter saat proses OCR.

Setelah dilakukan *post-processing* berbasis aturan, akurasi karakter mengalami peningkatan menjadi 82,49% dengan CER menurun menjadi 17,51%. Peningkatan ini menunjukkan efektivitas aturan koreksi, terutama dalam menangani kesalahan substitusi karakter yang sering terjadi (seperti O–0, I–1, S–5) berdasarkan pola format pelat nomor Indonesia. Meskipun *post-processing* berhasil mengurangi beberapa kesalahan, dampaknya lebih terasa pada perbaikan kesalahan OCR tingkat karakter daripada mengatasi masalah segmentasi atau distorsi yang muncul pada tahap deteksi tepi dan ekstraksi ROI, yang umumnya disebabkan oleh kondisi citra yang buruk. Analisis lebih lanjut per kategori kondisi gambar (rotasi, kejelasan tepi, jarak, pencahayaan) mengonfirmasi bahwa meskipun *post-processing* memberikan peningkatan, akurasi tetap cenderung lebih rendah pada gambar dengan kualitas tepi yang buruk (*Low Clarity*), jarak yang terlalu jauh (*Far*), atau pencahayaan ekstrem (*Low Lighting*). Temuan ini menunjukkan bahwa kualitas citra input merupakan faktor pembatas utama terhadap performa sistem secara keseluruhan.

Penerapan *post-processing* pada hasil keluaran Tesseract OCR menunjukkan bahwa kesalahan pembacaan karakter tidak sepenuhnya disebabkan oleh kegagalan algoritma dalam mendeteksi bentuk huruf secara visual, melainkan oleh ambiguitas morfologis antar karakter yang memiliki kemiripan struktur. Peningkatan akurasi dari 81,56% menjadi 82,88% menandakan bahwa sebagian *error* yang semula dianggap sebagai kekeliruan model OCR dapat diperbaiki tanpa mengubah proses ekstraksi citra, melalui pemanfaatan pola substitusi yang sering berulang, seperti O–0 atau S–5. *Post-processing* tidak menambah kemampuan deteksi visual, tetapi berfungsi sebagai mekanisme koreksi logis yang “menyaring” hasil OCR berdasarkan konteks pelat nomor. Penurunan CER sebesar 1,32% juga mengindikasikan bahwa kesalahan yang berhasil dikoreksi bersifat konsisten dan tidak acak, sehingga metode berbasis aturan masih relevan untuk domain dengan ruang kode alfanumerik yang terbatas. Temuan ini menunjukkan bahwa meskipun peningkatan akurasi tidak drastis, *post-processing* berperan sebagai sistem stabilisasi yang membantu mempertahankan kinerja pada citra yang secara visual sulit, serta mengurangi ketergantungan penuh pada kualitas citra atau performa OCR.

Nilai akurasi yang tidak mencapai 100% menunjukkan bahwa proses deteksi dan pengenalan karakter masih dipengaruhi oleh kondisi visual citra masukan. Dalam penelitian ini ditemukan beberapa citra dengan *overexposure* yang melemahkan gradien kontur karakter sehingga tepi menjadi kurang stabil, serta citra dengan refleksi atau tekstur latar yang memicu terbentuknya kontur palsu. Hasil citra biner dari tahap deteksi kemudian menjadi masukan bagi Tesseract berbasis LSTM, yang menuntut *glyph* dengan kontras tinggi dan batas yang tegas. Apabila ROI masih memuat artefak berupa garis atau bercak, serta batas antar karakter yang tidak tajam, modul segmentasi cenderung menghasilkan *token* tambahan atau kehilangan *token* dalam proses pengenalan.

Pada contoh citra sampel (A) dengan pelat nomor BB 2430 HM, model mengenali hasil keluaran OCR sebagai “BB2430HM9”, dengan *character accuracy* sebesar 87,5% dan *character error rate* sebesar 12,5%. Kesalahan berupa penyisipan karakter “9” diseb-

babkan oleh residu bercak berkontras tinggi pada area bawah pelat yang lolos dari tahap *binarization* dan *non-maximum suppression*, kemudian tersegmentasi sebagai komponen terhubung dengan tinggi, rasio aspek, dan ketebalan gores yang berada dalam rentang *glyph* valid. Kondisi ini menyebabkan modul tokenisasi LSTM pada Tesseract mengenalinya sebagai karakter mandiri, sehingga terjadi *over-segmentation* dan memunculkan *token* tambahan. Temuan ini menunjukkan bahwa Tesseract OCR sensitif terhadap pola visual berkontras tinggi yang tidak sepenuhnya merepresentasikan karakter sebenarnya, sehingga area non-teks masih dapat terbaca sebagai bagian teks valid. Secara teknis, kesalahan ini membuktikan adanya keterbatasan segmentasi karakter akibat ambang *thresholding* dan pembersihan morfologi yang kurang ketat, yang memungkinkan artefak tepi semu lolos validasi dan dikenali sebagai karakter sah oleh model.

Analisis Tabel 4, Tabel 5, dan Tabel 6 menunjukkan bahwa performa rekapitulasi hasil model OCR dipengaruhi oleh satu faktor tunggal sekaligus merupakan konsekuensi gabungan dari beberapa variabel kualitas citra yang saling berinteraksi. Ketika satu faktor mengalami degradasi ekstrem, misalnya rotasi mendekati -90° atau pencahayaan yang terlalu rendah, akurasi tidak hanya menurun sebagian, tetapi dapat runtuh sepenuhnya hingga menghasilkan CER 100%, meskipun faktor lain berada dalam kondisi baik. Hal ini menunjukkan bahwa model tidak bekerja secara linear, tetapi bersifat *threshold-dependent*, yakni kinerja masih stabil selama kualitas citra berada di atas ambang tertentu, dan langsung gagal apabila ambang tersebut dilanggar. Kondisi ini mendukung asumsi dasar *Image Quality Assessment Theory* bahwa model pengenalan pola berbasis OCR tidak gagal karena kesalahan klasifikasi karakter secara bertahap, melainkan karena hilangnya struktur visual yang menjadi dasar proses segmentasi dan ekstraksi fitur. Perbedaan akurasi antar kategori yang relatif sempit (berkisar 0.68–0.86) menunjukkan bahwa tidak ada satu variabel yang benar-benar dominan dalam menentukan keberhasilan; rotasi, kejelasan tepi, jarak, dan pencahayaan semuanya memiliki kontribusi yang sebanding terhadap kestabilan pengenalan karakter. Artinya, peningkatan kinerja tidak dapat dicapai hanya dengan memperbaiki satu aspek citra—misalnya perataan sudut saja—tanpa meningkatkan kualitas tepi atau intensitas cahaya. Hal ini pula yang menjelaskan mengapa tingkat *error* secara keseluruhan mencapai 18,4%, karena kesalahan tidak terjadi akibat satu faktor tunggal, melainkan merupakan akumulasi deviasi kecil pada beberapa faktor yang secara statistik saling memperlemah proses ekstraksi fitur, meskipun tidak selalu menghasilkan kegagalan total seperti pada kasus CER 100%. Dengan kata lain, OCR *error* yang tinggi bukan semata-mata disebabkan oleh lemahnya algoritma, melainkan oleh sifat ketergantungan multimodal model terhadap input visual yang terstruktur. Kesimpulan ini juga menunjukkan bahwa model yang digunakan belum memiliki *robustness* yang memadai terhadap variasi kondisi citra di dunia nyata, sehingga peningkatan model perlu diarahkan pada *adaptive preprocessing* atau *noise-invariant feature extraction* agar tidak terjadi *collapse performance* seperti yang ditunjukkan pada kasus akurasi 0%.

Apabila dibandingkan dengan penelitian terdahulu, hasil ini menunjukkan relevansi dan konsistensi dengan teori yang telah dikemukakan. Penelitian [3] membuktikan bahwa algoritma Canny Edge Detection efektif dalam menyoroti batas tepi objek de-

ngan presisi tinggi, dan hasil serupa juga tercermin dalam penelitian ini. Perbedaannya terletak pada tahap implementasi, di mana penelitian ini melanjutkan proses hingga tahap pengenalan karakter menggunakan *Tesseract OCR*. Hasil ini memperkuat bukti bahwa penggunaan metode deteksi tepi dan pengenalan karakter mampu menghasilkan model pengenalan yang lebih komprehensif. Kemudian penelitian [9] menunjukkan bahwa *Tesseract OCR* memiliki keunggulan dalam membedakan karakter yang memiliki kemiripan bentuk, seperti angka “0” dengan huruf “D” atau angka “1” dengan “7”. Hasil penelitian ini mendukung temuan tersebut karena model mampu mengenali sebagian besar karakter dengan tepat pada citra dengan kualitas tinggi. Meskipun demikian, hasil yang diperoleh juga menunjukkan bahwa *Tesseract OCR* masih rentan terhadap kesalahan pengenalan akibat *noise* visual dan ketidakteraturan bentuk fisik pelat nomor, terutama pada kondisi pelat yang telah aus atau kotor. Sementara itu, penelitian [14] mengenai peningkatan ketepatan segmentasi melalui koreksi kemiringan citra memberikan dasar teoritis yang menjelaskan mengapa beberapa kesalahan pembacaan masih terjadi dalam penelitian ini. Tanpa adanya koreksi geometrik, sudut pengambilan gambar yang tidak sejajar dengan bidang pelat menyebabkan distorsi karakter yang menurunkan keakuratan pengenalan. Kondisi ini mengindikasikan bahwa kualitas hasil OCR tidak hanya bergantung pada performa algoritma, tetapi juga pada keseragaman orientasi dan kestabilan citra masukan.

Dengan demikian, hasil penelitian ini memperlihatkan bahwa algoritma *Canny Edge Detection* efektif dalam menghasilkan batas pelat nomor yang jelas, dan *Tesseract OCR* mampu mengenali karakter dengan tingkat ketepatan yang memadai. Akan tetapi, kinerja model masih dipengaruhi oleh variasi kondisi visual di lapangan. Hal ini menunjukkan bahwa keberhasilan model pengenalan pelat nomor ditentukan oleh efektivitas algoritma yang digunakan, konsistensi kualitas citra, dan ketepatan tahap segmentasi sebelum proses OCR dijalankan.

4. Kesimpulan

Penelitian ini menunjukkan bahwa penerapan algoritma *Canny Edge Detection* dan *Tesseract OCR* efektif dalam mendeteksi dan mengenali pelat nomor kendaraan roda dua secara otomatis. Hasil pengujian terhadap 125 citra kendaraan menunjukkan rata-rata *Character Accuracy* sebesar 81,6% dan *Character Error Rate* sebesar 18,4%, yang membuktikan bahwa model mampu melakukan identifikasi karakter dengan tingkat ketepatan yang cukup tinggi pada sebagian besar citra uji. Setelah diterapkan *post-processing* berbasis aturan, akurasi karakter meningkat menjadi 82,49% dengan penurunan *CER* menjadi 17,51%, yang menegaskan efektivitas mekanisme koreksi dalam memperbaiki kesalahan pengenalan karakter. Temuan ini menunjukkan bahwa kombinasi metode deteksi tepi dan pengenalan karakter dapat berfungsi sebagai alternatif solusi terhadap sistem pemeriksaan manual yang selama ini digunakan di UINSU. Selain itu, penelitian ini juga mengungkap bahwa kualitas citra masukan dan ketepatan tahap segmentasi memiliki pengaruh langsung terhadap performa OCR. Untuk penelitian selanjutnya, disarankan pengembangan model dengan menambahkan algoritma koreksi kemiringan (*deskewing*), menerapkan pendekatan *deep learning*-based OCR seperti *Convolutional Neural Network (CNN)* atau *Recurrent Neural Network (RNN)*, serta memperluas cakupan dataset agar model mampu beradaptasi

terhadap variasi kondisi visual yang lebih kompleks.

Kontribusi Penulis. Ananda Sarbaini Firin: Metodologi, perangkat lunak, penulisan–persiapan draf asli. Sriani: Penulisan–tinjauan dan penyuntingan, konseptualisasi, supervisi. Semua penulis telah membaca dan menyetujui versi manuskrip yang diterbitkan.

Ucapan Terima Kasih. Para penulis mengucapkan terima kasih kepada semua pihak yang telah berkontribusi dalam penelitian ini dan dalam penyusunan manuskrip. Kami sangat menghargai editor dan reviewer atas masukan serta dukungannya dalam menyempurnakan karya ini.

Pembiayaan. Penelitian ini tidak menerima pendanaan eksternal.

Konflik Kepentingan. Para penulis menyatakan tidak ada konflik kepentingan yang terkait dengan artikel ini.

Ketersediaan Data. Tidak tersedia.

Referensi

- [1] O. B. Kharisma and I. Junaedi, “Sistem identifikasi plat nomor kendaraan dalam penerapan regulasi pajak berbasis citra digital,” *Jurnal Sains dan Teknologi*, vol. 10, no. 1, pp. 117–127, 2021, doi: 10.23887/jstundiksha.v10i1.32975.
- [2] M. A. Khulan and R. Pebrianti, “Rancang bangun sistem keamanan parkir kendaraan roda dua dengan teknologi plate recognition,” pp. 1–88, 2023.
- [3] S. N. Aini and R. Kurniawan, “Application of Canny method to detect vehicle license plate in Tanjung Balai City Government Mess Area,” *Indonesian Journal of Artificial Intelligence and Data Mining*, vol. 7, no. 2, pp. 530–540, 2024, doi: 10.24014/ijaidm.v7i2.32426.
- [4] S. Setiawan, D. N. Sulistyowati, and N. Machmud, “Implementation of image processing in the recognition of official vehicle license plates,” *JITK (Jurnal Ilmu Pengetahuan dan Teknologi Komputer)*, vol. 9, no. 1, pp. 23–29, 2023, doi: 10.33480/jitk.v9i1.4181.
- [5] D. N. Alfarizi, R. A. Pangestu, D. Aditya, M. A. Setiawan, and P. Rosyani, “Penggunaan metode YOLO pada deteksi objek: Sebuah tinjauan literatur sistematis,” *Jurnal Artificial Intelligent dan Sistem Penunjang Keputusan*, vol. 1, no. 1, pp. 54–63, 2023. [Online]. Available: <https://jurnalmahasiswa.com/index.php/aidanspk>
- [6] M. Hanum, “Implementasi teknik embossing pada pengenalan plat kendaraan untuk identifikasi otomatis berbasis OpenCV,” *JoMMiE: Jurnal Multi Media dan IT*, vol. 8, no. 1, pp. 62–68, 2024, doi: 10.46961/jommit.v8i1.1361.
- [7] A. R. Hanif, E. Nasrullah, and F. X. A. Setyawan, “Deteksi karakter plat nomor kendaraan dengan menggunakan metode optical character recognition (OCR),” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 11, no. 1, pp. 109–117, 2023, doi: 10.23960/jitet.v11i1.2897.
- [8] R. M. Ridwan, S. Sumaryo, and A. Zamhuri, “Implementasi teknologi OCR untuk mendeteksi nomor plat mobil,” *e-Proceeding of Engineering*, vol. 11, no. 5, pp. 5429–5432, 2024.
- [9] T. I. Cahyani, M. Zakiamani, D. Riana, and S. Hardianti, “Perbandingan akurasi pengenalan karakter plat nomor menggunakan Tesseract dan data latih EMNIST,” *Journal of Information Technology and Computer Science (INTECOMS)*, vol. 5, no. 2, pp. 18–25, 2022.
- [10] A. F. I. Setyadi and Y. A. Susetyo, “Implementasi algoritma LSTM pada aplikasi optical character recognition berbasis website menggunakan Tesseract OCR,” *Jurnal Teknologi Sistem Informasi dan Aplikasi*, vol. 6, no. 2, pp. 63–71, 2023, doi: 10.32493/jtsi.v6i2.29235.
- [11] B. Setyawan and G. M. Sabrie, “Kajian potensi kendaraan bermotor menjadi barang kena cukai,” *Jurnal Perspektif Bea dan Cukai*, vol. 6, no. 2, pp. 365–385, 2022.
- [12] R. Devita and S. Sumijan, “Canny edge detection and image segmentation for precision face recognition system,” *JURTEKSI (Jurnal Teknologi dan Sistem Informasi)*, vol. 10, no. 2, pp. 347–354, 2024, doi: 10.33330/jurteksiv10i2.3059.
- [13] D. Wicaksono, D. P. Almeyda, I. M. M. Putra, and L. Malihatuningrum, “Analisis perbandingan metode pra pemrosesan citra untuk deteksi tepi Canny pada citra berbagai kondisi jalan menggunakan bahasa pemrograman Python,” *Jurnal Teknologi dan Ilmu Komputer Prima (JUTIKOMP)*, vol. 7, no. 1, pp. 17–31, 2024.

- [14] A. E. Raharja, D. A. Navastara, and S. C. Hidayati, "Koreksi kemiringan citra menggunakan planar homography untuk pengenalan pelat nomor kendaraan," *Jurnal Teknik ITS*, vol. 10, no. 2, pp. 360–365, 2021.
- [15] B. N. Situmorang, K. A. A'la, A. Arvianti, F. I. Yusuf, and E. W. Handamari, "Implementasi metode Bayesian untuk menghitung premi produk asuransi kendaraan bermotor dengan pendekatan Monte Carlo Markov Chain," *Euler: Jurnal Ilmiah Matematika, Sains dan Teknologi*, vol. 13, no. 2, pp. 264–270, 2025.
- [16] R. D. Putra and U. Suwardoyo, "Identifikasi plat kendaraan berbasis aplikasi Python," *Jurnal Sintaks Logika*, vol. 4, no. 2, pp. 2412–2775, 2024. [Online]. Available: <https://jurnal.umpar.ac.id/index.php/sylog>
- [17] R. Muhammad and S. Yulianto, "Penerapan pemrograman Python dalam menentukan waktu overhaul kondensor turbin uap," *Jurnal Konversi Energi dan Manufaktur*, vol. 8, no. 1, pp. 49–57, 2023.
- [18] A. Z. S. Azizah, F. N. T. Yalis, I. S. Narayana, K. A. Syalom, and P. Rosyani, "Pengolahan citra digital dengan penerapan teknik ambang batas: Studi kasus menggunakan OpenCV," *Jurnal Artificial Intelligent dan Sistem Penunjang Keputusan*, vol. 1, no. 4, pp. 283–287, 2024. [Online]. Available: <https://jurnalmahasiswa.com/index.php/aidanspk>
- [19] J. Ulfah and Nurdin, "Implementasi metode deteksi tepi Canny untuk menghitung jumlah uang koin dalam gambar menggunakan OpenCV," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 11, no. 3, pp. 420–426, 2023, doi: [10.23960/jitet.v11i3.3147](https://doi.org/10.23960/jitet.v11i3.3147).
- [20] B. Hatipoğlu, İ. Karagöz, and M. İnal, "Threshold based image enhancement method for low contrast X-ray images using CLAHE," *Uluslararası Muhendislik Arastırma ve Gelistirme Dergisi*, Dec. 2022, doi: [10.29137/umagd.1203617](https://doi.org/10.29137/umagd.1203617).
- [21] Saluky and Y. Marine, "Penerapan algoritma Canny untuk deteksi tepi menggunakan Python dan OpenCV," *Smart Techno (Smart Technology, Informatic, and Technopreneurship)*, vol. 5, no. 1, pp. 1–7, 2023.
- [22] A. Pranadia, E. Rachmawati, and Risnandar, "Deteksi jenis kendaraan berdasarkan nomor plat menggunakan metode vertical edge detection dan connected component labelling," *Jurnal Tugas Akhir Fakultas Informatika*, vol. 8, no. 2, pp. 3014–3027, 2021.
- [23] T. Octaviani, H. Setiawan, and O. H. Kelana, "Pytesseract dan template matching untuk otomatisasi input data KTP," *Jurnal Buana Informatika*, vol. 14, no. 2, pp. 147–156, 2023.
- [24] R. Winanjaya, A. D. GS, and F. Anggraini, "Penerapan kombinasi algoritma Sobel dan Canny (SoCan) dalam identifikasi citra inversi albatros Laysan," *Building of Informatics, Technology and Science (BITS)*, vol. 4, no. 1, 2022, doi: [10.47065/bits.v4i1.1660](https://doi.org/10.47065/bits.v4i1.1660).
- [25] M. Hajjali, "OCR post-processing using large language models," *UNLV Theses, Dissertations, Professional Papers, and Capstones*, p. 4811, 2023, doi: [10.34917/36910880](https://doi.org/10.34917/36910880).
- [26] L. Maximillian, Y. F. Riti, M. Anugraha, and Y. J. Palis, "Perbandingan algoritma Sobel dan Canny untuk deteksi tepi citra daun lidah buaya," *Komputa: Jurnal Ilmiah Komputer dan Informatika*, vol. 12, no. 2, pp. 69–79, 2023.
- [27] R. A. Saputra, M. A. Rayadin, and W. O. I. Febryanti, "Perbandingan efisiensi deteksi tepi Roberts, Prewitt, dan Canny untuk identifikasi kartu mahasiswa," *Jurnal Informatika*, vol. 10, no. 2, pp. 136–142, 2023, doi: [10.31294/inf.v10i2.16726](https://doi.org/10.31294/inf.v10i2.16726).