

SISTEM PENGENALAN PLAT NOMOR KENDARAAN INDONESIA MENGUNAKAN *K-NEAREST NEIGHBOR* BERBASIS *OPENCV PYTHON*

Ferina Putri Soedjono^{1*}

¹Teknik Informatika, UIN Maulana Malik Ibrahim Malang, Malang, Indonesia

*Penulis korespondensi: ferinaputrisoedjono@gmail.com

ABSTRAK

Plat nomor kendaraan merupakan identitas resmi kendaraan bermotor yang memiliki peran penting dalam pengawasan lalu lintas dan penegakan hukum. Pengenalan plat nomor secara manual memiliki keterbatasan dalam kecepatan dan konsistensi, sehingga dibutuhkan sistem otomatis yang mampu membaca plat nomor kendaraan secara akurat. Penelitian ini bertujuan membangun sistem *Automatic License Plate Recognition* berbasis *Computer Vision* klasik yang mampu mendeteksi dan membaca plat nomor kendaraan Indonesia secara otomatis menggunakan algoritma *K-Nearest Neighbor* (KNN) dan *adaptive thresholding* melalui library *OpenCV Python*. Sistem yang dikembangkan bekerja melalui empat tahap utama: deteksi lokasi plat nomor kendaraan menggunakan pengelompokan kontur karakter; praproses citra menggunakan ekstraksi *value channel* pada ruang warna HSV, peningkatan kontras morfologi dengan operasi Top-Hat dan Black-Hat, serta *adaptive thresholding*; segmentasi karakter melalui analisis kontur pada citra biner; dan pengenalan karakter menggunakan algoritma KNN dengan fitur vektor piksel berukuran 20×30 yang dimampatkan menjadi 600 dimensi. Sistem juga dilengkapi mekanisme deteksi kecerahan otomatis untuk membedakan plat terang dan gelap. Pengujian dilakukan terhadap 25 gambar plat nomor kendaraan dengan kondisi bervariasi. Sistem berhasil membaca seluruh karakter dengan benar pada 21 dari 25 gambar (84%). Kesalahan pengenalan terjadi pada 3 gambar akibat kemiripan bentuk karakter seperti W/V dan O/0, serta pada 1 gambar akibat *noise* pencahayaan yang terbaca sebagai karakter. Sistem bekerja optimal pada plat hitam dan putih, namun belum memberikan hasil maksimal pada plat berwarna lain seperti merah. Sistem yang dibangun terbukti mampu mengenali plat nomor kendaraan Indonesia dengan akurasi yang memadai menggunakan pendekatan ringan tanpa kebutuhan GPU.

Kata kunci: Pengenalan plat nomor kendaraan, *K-Nearest Neighbor*, OCR, *OpenCV*, *Computer Vision*

1 PENDAHULUAN

Pertumbuhan jumlah kendaraan bermotor yang pesat menimbulkan kebutuhan akan sistem identifikasi kendaraan yang efisien dan akurat. Plat nomor kendaraan merupakan identitas resmi yang dapat digunakan sebagai dasar pengawasan lalu lintas, sistem parkir otomatis, dan penegakan hukum. Proses identifikasi manual tidak lagi memadai karena keterbatasan kecepatan dan potensi kesalahan manusia. Oleh karena itu, sistem *Automatic License Plate Recognition* (ALPR) atau *Automatic Number Plate Recognition* (ANPR) menjadi komponen penting dalam sistem transportasi cerdas. Shashirangana et al. (2021) dan Lubna et al. (2021) menyatakan bahwa sistem ALPR kini dibutuhkan secara luas untuk penegakan hukum, pengawasan, hingga operasi *toll booth*, dengan spesifikasi operasional yang beragam sesuai kebutuhan aplikasi.

Beberapa penelitian terdahulu telah mengusulkan pendekatan ALPR khusus untuk plat nomor kendaraan Indonesia. Hidayatullah et al. (2016) mengembangkan sistem LPR untuk kendaraan Indonesia menggunakan kombinasi pemrosesan tepi horizontal-vertikal dan

template matching, dengan tingkat deteksi 78,75% dari 80 sampel. Wijaya (2022) meneliti pengenalan plat nomor Indonesia pada kendaraan bergerak menggunakan pendekatan berbasis analisis tepi vertikal. Sementara itu, sejumlah penelitian terbaru memanfaatkan *deep learning*: J. T. Wijaya & Arymurthy (2025) mengusulkan Gaussian YOLOv7 untuk deteksi dan pengenalan plat Indonesia, sedangkan Taufik et al. (2025) mengimplementasikan sistem *real-time* menggunakan *EasyOCR* dan validasi regex dengan akurasi 75%. Budianto (2018) memberikan tinjauan komprehensif terhadap perkembangan ALPR di Indonesia, mengidentifikasi bahwa faktor variasi format dan kualitas gambar menjadi tantangan utama.

Meskipun pendekatan berbasis *deep learning* menunjukkan akurasi yang tinggi, metode tersebut umumnya membutuhkan *Graphics Processing Unit* (GPU) dan dataset pelatihan yang besar, sehingga sulit diterapkan pada lingkungan dengan sumber daya terbatas. O'Mahony et al. (2020) menunjukkan bahwa pendekatan *Computer Vision* klasik masih relevan, khususnya untuk kondisi yang terkontrol dan sumber daya komputasi terbatas. Budianto et al. (2018) membuktikan bahwa KNN mampu mengungguli *Support Vector Machine* (SVM) pada tugas pengenalan karakter plat nomor kendaraan Indonesia. Putra et al. (2020) juga menunjukkan bahwa kombinasi ekstraksi fitur dan KNN efektif untuk deteksi kendaraan di jalan raya. Namun, implementasi ALPR berbasis KNN dengan *pipeline* yang sepenuhnya memanfaatkan *OpenCV Python* untuk kondisi plat nomor kendaraan Indonesia masih belum banyak dieksplorasi secara mendetail.

Berdasarkan latar belakang tersebut, penelitian ini bertujuan mengimplementasikan sistem ALPR berbasis *Computer Vision* klasik yang menggunakan KNN sebagai mesin pengenalan karakter, dengan *pipeline* pemrosesan citra berbasis *OpenCV Python*. Sistem dirancang khusus untuk menangani plat nomor kendaraan Indonesia, dilengkapi antarmuka grafis (*Graphical User Interface/GUI*) interaktif, dan dievaluasi pada 25 gambar kendaraan dengan kondisi pengambilan yang bervariasi.

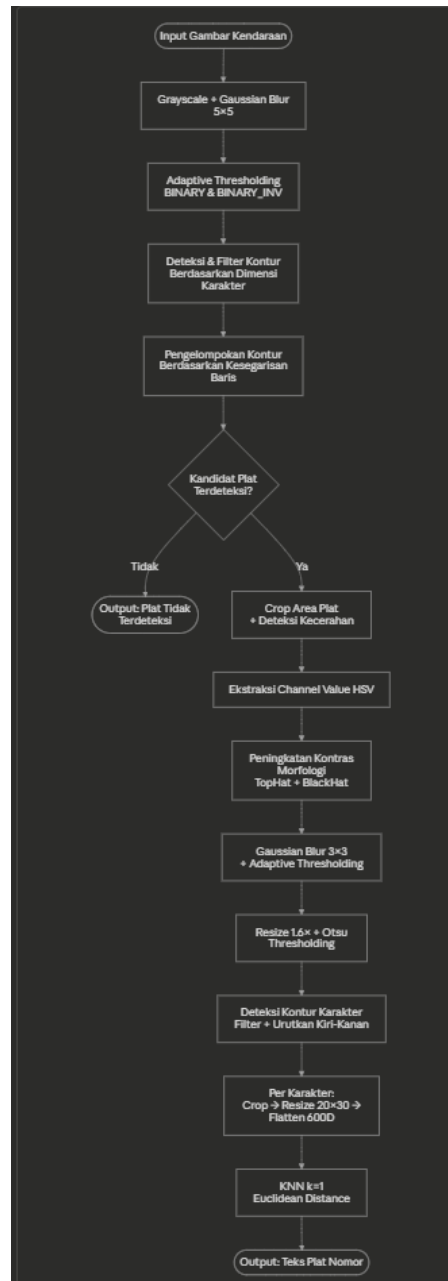
2 METODE

2.1 Dataset dan Lingkungan Penelitian

Dataset penelitian terdiri dari 25 gambar plat nomor kendaraan bermotor yang dikumpulkan secara mandiri dari kondisi nyata. Gambar diambil menggunakan kamera *smartphone* dengan variasi kondisi yang mencakup perbedaan jarak pengambilan, sudut kamera, dan kondisi pencahayaan. Plat nomor yang digunakan dalam pengujian merupakan plat hitam dengan tulisan putih (kendaraan pribadi) dan plat putih dengan tulisan hitam (format terbaru). Sistem diimplementasikan menggunakan Python 3.x dengan *library* utama *OpenCV* 4.x untuk pengolahan citra dan NumPy untuk komputasi numerik. Antarmuka pengguna dibangun menggunakan Tkinter dan Pillow.

2.2 Arsitektur Sistem

Sistem ALPR yang dikembangkan terdiri dari lima modul utama dengan tanggung jawab yang terpisah (*separation of concerns*). Modul *Preprocess.py* menangani seluruh transformasi citra dari gambar input hingga citra biner. Modul *CharDetect.py* mengelola alur deteksi dan pengenalan karakter menggunakan KNN. Modul *KemungkinanPlat.py* dan *PossibleChar.py* menyimpan representasi objek kandidat plat dan karakter. Antarmuka pengguna tersedia dalam bentuk *app.py* untuk tampilan lengkap dengan jendela detail proses. Modul *GenDataGUI.py* berfungsi sebagai alat pelatihan model KNN secara interaktif. Data model KNN disimpan dalam dua file: *classifications.txt* yang memuat label ASCII setiap karakter, dan *knn2.txt* yang memuat vektor fitur flatten 600 dimensi per sampel karakter. Alur kerja sistem secara keseluruhan ditunjukkan pada Gambar 1.



Gambar 1. Flowchart pipeline sistem ALPR

2.3 Deteksi Lokasi Plat

Deteksi lokasi plat menggunakan pendekatan berbasis kontur karakter (*character-based plate detection*), berbeda dengan pendekatan konvensional yang mencari kotak plat secara langsung. Pendekatan ini lebih robust terhadap variasi bentuk kendaraan karena mendeteksi isi plat (huruf dan angka) daripada bingkai fisiknya.

Proses deteksi dimulai dengan konversi gambar ke *grayscale* diikuti Gaussian blur berukuran kernel 5×5 untuk mengurangi *noise*. Kemudian *adaptive thresholding* diterapkan dua kali: metode `THRESH_BINARY_INV` untuk kondisi plat gelap (huruf putih pada latar hitam) dan `THRESH_BINARY` untuk plat terang (huruf hitam pada latar putih). Kounlaxay et al. (2024) menyatakan bahwa penggunaan *adaptive thresholding* secara signifikan meningkatkan performa deteksi pada kondisi pencahayaan yang tidak merata dibandingkan *global thresholding*.

Setiap kontur hasil deteksi difilter berdasarkan kriteria dimensi karakter: tinggi antara 8 piksel hingga 35% tinggi gambar, lebar antara 3 piksel hingga 15% lebar gambar, rasio aspek 0,1–1,0, dan luas area antara 50 hingga 2% area gambar. Kontur yang memenuhi syarat kemudian dikelompokkan berdasarkan kesegarisan dan kemiripan tinggi. Kelompok-kelompok yang berdekatan digabungkan (*merging*) untuk mengakomodasi spasi antar kelompok huruf pada format plat Indonesia "X NNNN XX". Setiap kandidat plat diberi skor berdasarkan jumlah kontur karakter, rasio aspek, posisi vertikal, dan lebar relatif. Kandidat dengan skor tertinggi dipilih sebagai lokasi plat.

2.4 Preprocessing Citra

Setelah area plat di-*crop*, sistem mendeteksi tingkat kecerahan rata-rata piksel *grayscale* area plat. Jika nilai rata-rata di bawah 130 (plat gelap), gambar diinversi sehingga karakter menjadi hitam di atas latar putih, memungkinkan satu *preprocessing pipeline* yang seragam untuk berbagai kondisi plat.

Preprocessing citra dilakukan oleh modul Preprocess.py dalam empat langkah berurutan. Pertama, ekstraksi *channel value* (V) dari ruang warna HSV dilakukan menggunakan `cv2.cvtColor(img, COLOR_BGR2HSV)`. Giuliani (2022) menunjukkan bahwa ruang warna HSV memisahkan informasi kecerahan dari saturasi dan *hue*, sehingga *channel V* lebih representatif untuk pengolahan citra daripada konversi *grayscale* biasa. Kedua, peningkatan kontras morfologi dilakukan dengan kombinasi operasi Top-Hat dan Black-Hat menggunakan *structuring element* persegi 3×3:

$$C_out = C_in + TopHat(C_in) - BlackHat(C_in)$$

Widyantera et al. (2016) dan Kushol et al. (2019) menjelaskan bahwa Top-Hat mengekstrak objek terang pada latar gelap, Black-Hat mengekstrak objek gelap pada latar terang, dan kombinasi keduanya menghasilkan peningkatan kontras yang efektif pada citra dengan pencahayaan tidak seragam. Ketiga, Gaussian blur berkernel 3×3 diterapkan untuk menghaluskan *noise* sebelum *thresholding*. Keempat, Adaptive Mean Thresholding dengan *block size* 19 dan konstanta $C = 9$ menghasilkan citra biner hitam-putih.

2.5 Segmentasi Karakter

Segmentasi karakter dilakukan pada citra *threshold* hasil preprocessing. Citra diperbesar 1,6× terlebih dahulu menggunakan interpolasi bilinear untuk meningkatkan resolusi karakter. Kemudian *otsu thresholding* diterapkan sebagai langkah pembersihan tambahan yang menentukan nilai ambang batas secara adaptif berdasarkan distribusi histogram citra (Islam et al., 2023). Semua kontur pada citra dideteksi menggunakan `cv2.findContours` dengan mode `RETR_LIST` dan difilter berdasarkan kriteria yang sama dengan tahap deteksi plat. Kontur yang saling tumpang tindih lebih dari 50% luasnya dihapus dengan mempertahankan kontur yang lebih besar. Kontur-kontur yang tersisa dikelompokkan berdasarkan kesegarisan baris, kemudian diurutkan dari kiri ke kanan berdasarkan koordinat X untuk menghasilkan urutan pembacaan yang benar.

2.6 Pengenalan Karakter Menggunakan KNN

K-Nearest Neighbor (KNN) adalah algoritma pembelajaran mesin berbasis *instance* yang mengklasifikasikan data baru berdasarkan kelas mayoritas dari k tetangga terdekatnya dalam ruang fitur, dengan jarak dihitung menggunakan Euclidean distance (Halder et al., 2024). Budianto et al. (2018) membuktikan bahwa KNN efektif untuk pengenalan karakter plat nomor kendaraan Indonesia, bahkan dibandingkan dengan SVM pada kondisi dataset yang terbatas.

Setiap karakter tersegmentasi diproses melalui lima langkah: (1) karakter di-*crop* dari citra *threshold* sesuai *bounding box*-nya; (2) di-*resize* ke ukuran standar 20×30 piksel

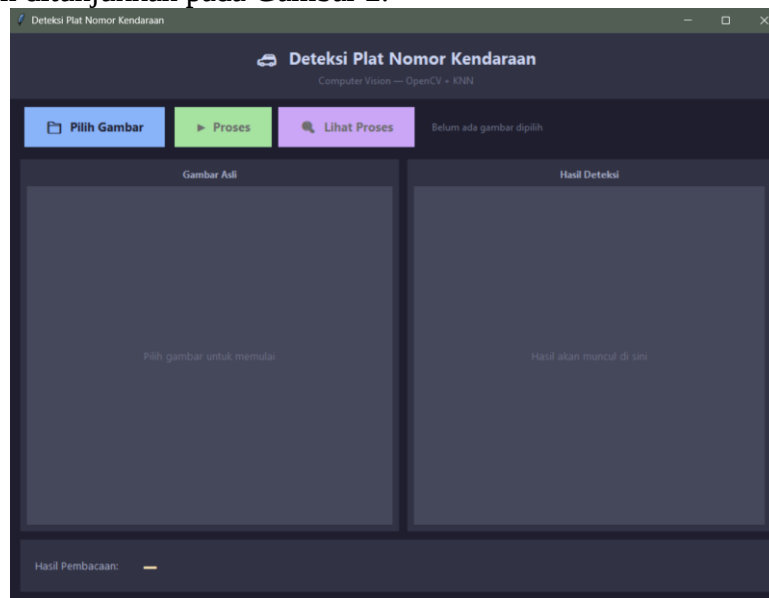
menggunakan interpolasi bilinear; (3) di-flatten menjadi vektor fitur berdimensi 600 ($20 \times 30 = 600$ nilai piksel); (4) vektor fitur dibandingkan dengan seluruh data training menggunakan KNN dengan nilai $k=1$ berbasis *Euclidean distance*; (5) karakter dengan jarak terkecil dari data training dipilih dan kode ASCII-nya dikonversi menjadi karakter teks. Seluruh karakter yang dikenali kemudian digabungkan membentuk string plat nomor akhir. Implementasi KNN menggunakan `cv2.ml.KNearest_create()` dari modul machine learning *OpenCV*.

Pelatihan model dilakukan secara semi-otomatis melalui modul *GenDataGUI.py*. Pengguna dapat memilih gambar kendaraan sebagai sumber pelatihan; sistem kemudian mendeteksi plat, melakukan preprocessing, dan menampilkan karakter satu per satu untuk dilabeli. Data baru ditambahkan (*append*) ke file training yang sudah ada sehingga model dapat ditingkatkan secara bertahap.

3 HASIL DAN PEMBAHASAN

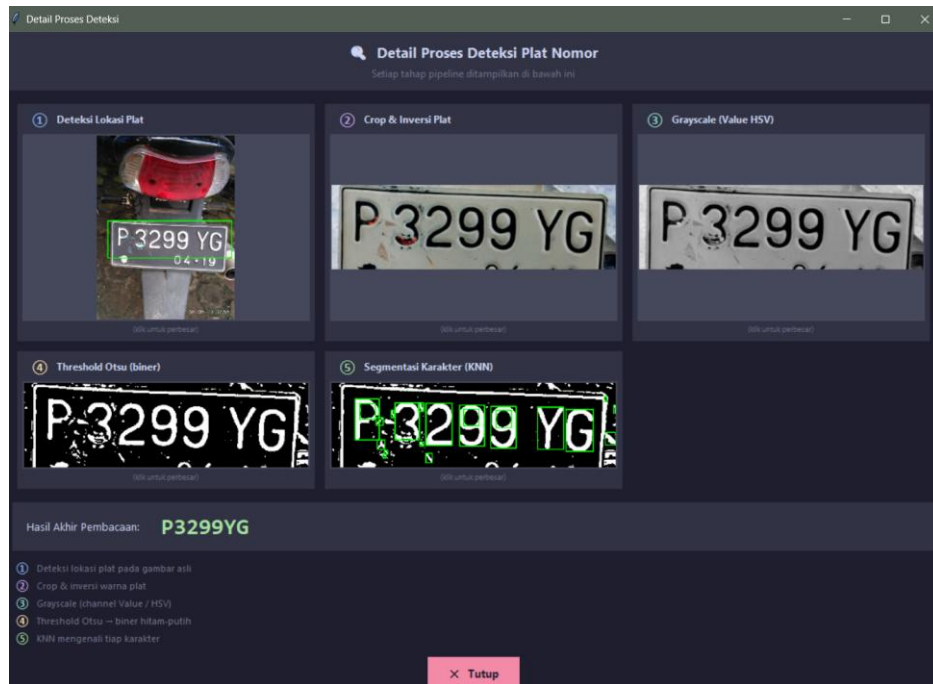
3.1 Implementasi Sistem

Sistem berhasil diimplementasikan sebagai aplikasi berbasis GUI menggunakan Tkinter. Antarmuka utama *app.py* menampilkan dua panel gambar berdampingan: panel kiri untuk gambar asli kendaraan dan panel kanan untuk hasil deteksi. Tombol "Lihat Proses" membuka jendela terpisah yang menampilkan lima tahap *pipeline* secara visual: deteksi lokasi plat, crop dan inversi, *grayscale* HSV, *otsu thresholding*, dan segmentasi karakter. Antarmuka *MainGUI.py* menampilkan proses serupa dalam format *scrollable step-by-step*. Tampilan antarmuka sistem ditunjukkan pada Gambar 2.



Gambar 2. Screenshot antarmuka utama *app.py*

Contoh visual hasil pembacaan sistem terhadap sampel-sampel tersebut ditunjukkan pada Gambar 3.



Gambar 3. Screenshot hasil pembacaan plat

3.2 Hasil Pengujian

Pengujian dilakukan terhadap 25 gambar plat nomor kendaraan yang dikumpulkan dari kondisi nyata dengan variasi jarak, sudut, dan pencahayaan. Dari 25 gambar tersebut, sistem berhasil membaca seluruh karakter dengan benar pada 21 gambar (84%). Ringkasan performa sistem secara keseluruhan dari 25 gambar uji disajikan pada Tabel 1.

Tabel 1. Ringkasan performa sistem pada 25 gambar uji

Metrik	Jumlah	Persentase
Total gambar diuji	25	100%
Pembacaan seluruh karakter benar	21	84%
Kesalahan karakter mirip	3	12%
Karakter noise terbaca sebagai karakter	1	4%

3.3 Analisis Kesalahan dan Pembahasan

Berdasarkan hasil pengujian, terdapat dua kategori kesalahan yang teridentifikasi. Kategori pertama adalah kebingungan karakter yang secara visual mirip (*character confusion*), yang terjadi pada 3 gambar (12%). Pada sampel nomor 2, karakter "W" dibaca sebagai "V" karena keduanya memiliki struktur bentuk yang serupa, dua garis diagonal yang bertemu di tengah, sehingga vektor fitur pixel flatten 600 dimensi menghasilkan jarak Euclidean yang sangat kecil antar keduanya. Pada sampel nomor 8, karakter "Q" dibaca sebagai "0" (nol) karena ekor kecil pada karakter "Q" seringkali tidak terdeteksi dengan jelas pada citra threshold, sehingga karakter tersebut tampak seperti lingkaran penuh layaknya "0". Kelemahan ini merupakan keterbatasan inheren dari pendekatan berbasis *raw pixel flatten* yang tidak mengekstrak fitur struktural karakter secara eksplisit (Putra et al., 2020).

Kategori kedua adalah pembacaan elemen fisik non-karakter sebagai karakter, yang terjadi pada 1 gambar (4%). Pada sampel nomor 6, baut atau elemen mekanis pada permukaan plat yang sudah aus memiliki dimensi dan rasio aspek yang memenuhi kriteria filter kontur karakter, sehingga ikut tersegmentasi dan dikenali sebagai karakter "A". Islam et al. (2023) dan Kounlaxay et al. (2024) mengidentifikasi permasalahan serupa dalam sistem LPR berbasis contour, di mana elemen fisik plat yang kotor atau rusak dapat mempersulit proses segmentasi.

Analisis lebih lanjut menunjukkan bahwa sistem bekerja konsisten pada plat hitam dengan tulisan putih maupun plat putih dengan tulisan hitam, berkat mekanisme deteksi kecerahan otomatis yang membalikkan citra plat gelap sebelum *preprocessing*. Kegagalan masih terjadi pada gambar dengan elemen fisik *noise* pada permukaan plat dan pada karakter yang memiliki kemiripan visual tinggi, yang merupakan batasan dari pendekatan ekstraksi fitur berbasis nilai piksel mentah.

4 KESIMPULAN

Penelitian ini berhasil mengimplementasikan sistem ALPR berbasis *Computer Vision* klasik untuk plat nomor kendaraan Indonesia menggunakan KNN dan *OpenCV* Python. Berdasarkan pengembangan, implementasi, dan pengujian yang dilakukan, dapat disimpulkan bahwa sistem yang dibangun terbukti mampu bekerja tanpa kebutuhan GPU dan dapat berjalan di lingkungan dengan sumber daya komputasi terbatas. *Pipeline* deteksi lokasi plat berbasis pengelompokan kontur karakter berhasil mengidentifikasi area plat pada gambar kendaraan nyata, dan *preprocessing* menggunakan ekstraksi channel value HSV serta peningkatan kontras morfologi memberikan citra biner yang cukup baik untuk proses segmentasi. Sistem mencapai akurasi pembacaan karakter penuh sebesar 84% pada 25 gambar uji, mengungguli beberapa sistem sejenis berbasis *Computer Vision* klasik yang dipublikasikan sebelumnya. Kelemahan utama yang teridentifikasi adalah kebingungan pada pasangan karakter yang secara visual mirip (W/V, Q/O) sebesar 12%, serta pembacaan elemen fisik *noise* sebagai karakter sebesar 4%, yang keduanya merupakan keterbatasan dari pendekatan fitur berbasis *raw pixel flatten*.

Sebagai pengembangan lebih lanjut, penelitian mendatang dapat mengeksplorasi penggunaan fitur yang lebih deskriptif seperti *Histogram of Oriented Gradients* (HOG) sebagai pengganti *raw pixel flatten* untuk meningkatkan kemampuan diskriminasi KNN pada karakter yang mirip. Penambahan lapisan validasi format plat nomor kendaraan Indonesia menggunakan *regular expression* (regex) juga berpotensi mengurangi kesalahan akibat elemen *noise*. Selain itu, perluasan dataset pengujian yang lebih beragam dan seimbang antar kelas karakter akan memberikan evaluasi yang lebih representatif terhadap performa sistem.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Dosen Pengampu Mata Kuliah Komputer Vision dan Manajemen Proyek yang telah memberikan bimbingan dan arahan selama proses penelitian dan pengembangan sistem ini berlangsung.

DAFTAR PUSTAKA

- Budianto, A. (2018). Automatic License Plate Recognition: A Review with Indonesian Case Study. *Scientific Journal of Informatics*, 5(2), 2407–7658. <https://doi.org/10.15294/sji.v5i2.15804>
- Budianto, A., Maryono, D., & Ariyuana, R. (2018). Perbandingan *K-Nearest Neighbor* (KNN) dan Support Vector Machine (SVM) dalam Pengenalan Karakter Plat Kendaraan Bermotor. *Jurnal Ilmiah Pendidikan Teknik dan Kejuruan*, 11(1), 27. <https://doi.org/10.20961/jiptek.v11i1.18018>
- Giuliani, D. (2022). Metaheuristic Algorithms Applied to Color Image Segmentation on HSV Space. *Journal of Imaging*, 8(1). <https://doi.org/10.3390/jimaging8010006>
- Halder, R. K., Uddin, M. N., Uddin, M. A., Aryal, S., & Khraisat, A. (2024). Enhancing *K-Nearest Neighbor* algorithm: a comprehensive review and performance analysis of modifications. *Journal of Big Data*, 11(1). <https://doi.org/10.1186/s40537-024-00973-y>

- Hidayatullah, P., Feirizal, F., Permana, H., Mauluddiah, Q., & Dwitama, A. (2016). License plate detection and recognition for Indonesian cars. *International Journal on Electrical Engineering and Informatics*, 8(2), 331–346. <https://doi.org/10.15676/ijeei.2016.8.2.7>
- Islam, D., Mahmud, T., & Chowdhury, T. (2023). An efficient automated vehicle license plate recognition system under image processing. *Indonesian Journal of Electrical Engineering and Computer Science*, 29(2), 1055–1062. <https://doi.org/10.11591/ijeecs.v29.i2.pp1055-1062>
- Kounlaxay, K., Yoon, Y. C., & Kim, S. K. (2024). *Vehicle license plate detection and recognition using OpenCV and Tesseract OCR*. 14(4). <https://doi.org/10.18517/ijaseit.14.4.18137>
- Kushol, R., Raihan, Md. N., Rahman, A. B. M. A., & Salekin, M. S. (2019). *contrast enhancement of medical X-Ray image using morphological operators with optimal structuring element*. <http://arxiv.org/abs/1905.08545>
- Lubna, Mufti, N., & Shah, S. A. A. (2021). Automatic number plate recognition:A detailed survey of relevant algorithms. In *Sensors* (Vol. 21, Number 9). MDPI AG. <https://doi.org/10.3390/s21093028>
- O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., Riordan, D., & Walsh, J. (2020). Deep Learning vs. Traditional *Computer Vision. Advances in Intelligent Systems and Computing*, 943, 128–144. https://doi.org/10.1007/978-3-030-17795-9_10
- Putra, F. A. I. A., Utaminingrum, F., & Mahmudy, W. F. (2020). HOG feature extraction and KNN classification for detecting vehicle in the highway. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, 14(3), 231. <https://doi.org/10.22146/ijccs.54050>
- Shashirangana, J., Padmasiri, H., Meedeniya, D., & Perera, C. (2021). Automated license plate recognition: A survey on methods and techniques. In *IEEE Access* (Vol. 9, pp. 11203–11225). Institute of Electrical and Electronics Engineers Inc. <https://doi.org/10.1109/ACCESS.2020.3047929>
- Taufik, M., Hernandi, A., Anaam, M. W. S., & Riansyah, A. (2025). *Implementation of an Indonesian vehicle license plate recognition system in real-time using easyocr and regex pattern validation*. <https://doi.org/10.20527/jtiulm.v10i2.473>
- Widyantara, I. M. O., Wirastuti, N. M. A. E. D., Asana, I. M. D. P., & Adnyana, I. B. P. (2016). Image Enhancement Using Morphological Contrast Enhancement for Video Based Image Analysis. In *2017 1st International Conference on Informatics and Computational Sciences (ICICoS)*. <https://doi.org/10.1109/ICODSE.2016.7936115>
- Wijaya, J. T., & Arymurthy, A. M. (2025). Indonesian license plate detection and recognition system using Gaussian YOLOv7. *Jurnal Ilmu Komputer Dan Informasi*, 18(2), 141–153. <https://doi.org/10.21609/jiki.v18i2.1320>
- Wijaya, M. C. (2022). Research of indonesian license plates recognition on moving vehicles. *Eureka, Physics and Engineering*, 2022(6), 185–198. <https://doi.org/10.21303/2461-4262.2022.002424>