

Implementasi Teknik Embossing pada Pengenalan Plat Kendaraan untuk Identifikasi Otomatis Berbasis OpenCV

Mutia Hanum^a

^a Politeknik Negeri Media Kreatif

INFORMASI ARTIKEL

Sejarah Artikel:

Diterima Redaksi: Juni 2024
Revisi Akhir: Juni 2024
Diterbitkan Online: Juni 2024

KATA KUNCI

Embossing, Plat Kendaraan, *Computer Vision*, *Edge Detection*, OpenCV

KORESPONDENSI

Mutia Hanum
Program Studi Pemeliharaan Mesin, Jurusan
Teknologi Industri, Politeknik Negeri Media
Kreatif Jl. Srengseng Sawah Raya No.17,
RT.8/RW.3, Srengseng Sawah, Kec.
Jagakarsa, Kota Jakarta Selatan, Daerah
Khusus Ibukota Jakarta 12630
Email : mutia.hanum@polimedia.ac.id

ABSTRACT

Plat kendaraan biasanya menggunakan teknik *embossing* yang dapat diklasifikasikan sebagai teknik cetak tinggi. Proses ini melibatkan pencetakan karakter dan angka yang dicetak dengan menekan plat logam menggunakan mesin cetak press sehingga karakter tersebut timbul dan bisa terbaca dengan jelas. Hal ini bertujuan untuk meningkatkan keamanan dan daya tahan terhadap pemalsuan. Metode ini sering digunakan dalam pembuatan plat kendaraan resmi oleh pemerintah atau lembaga yang berwenang. Teknologi cetak dalam pada plat kendaraan kini juga dapat digunakan dalam proses identifikasi kendaraan otomatis menggunakan teknik pengolahan citra. Seiring perkembangan teknologi, kini pemanfaatan *Artificial Intelligence* sangat digemari, salah satunya *Computer Vision* yang dapat berfungsi sebagai mata manusia. Pada penelitian ini, selain dilakukan deteksi dan identifikasi citra kendaraan plat, juga akan dilihat seberapa besar tingkat performansinya. Diawali dengan akuisisi data pengambilan citra, *preprocessing*, implementasi *edge detection*, segmentasi, hingga *Tesseract OCR*. Berdasarkan hasil pengujian diperoleh tingkat keberhasilan sebesar 92.44% yang berhasil karakternya teridentifikasi otomatis.

DOI: <https://doi.org/10.46961/jommit.v8i1>

1. PENDAHULUAN

Plat kendaraan umumnya dibuat menggunakan teknik embossing, yang merupakan salah satu teknik cetak tinggi dimana gambar atau teks dicetak dengan menonjolkan atau menekankan permukaan bahan cetak, sehingga menciptakan dimensi tambahan yang terasa dan terlihat secara visual. Dalam embossing, tekanan diterapkan pada bahan cetak, seperti lembaran logam, untuk membentuk gambar, huruf, atau angka yang menonjol atau timbul dari permukaan bahan. Pada plat kendaraan, karakter-karakter seperti nomor registrasi atau huruf-huruf yang menunjukkan wilayah atau jenis kendaraan dicetak ke dalam plat logam menggunakan mesin pres atau mesin embossing. Ini membuat karakter tersebut menjadi timbul atau menonjol, sehingga lebih mudah terbaca dan sulit dipalsukan [1].

Dengan pertumbuhan jumlah kendaraan dan perkembangan teknologi, kebutuhan akan sistem identifikasi otomatis kendaraan menjadi semakin penting, baik untuk keperluan keamanan, pengelolaan lalu lintas, atau efisiensi parkir. Proses identifikasi manual plat kendaraan dapat menjadi tugas yang melelahkan dan tidak efisien, terutama dalam skala besar atau di lingkungan yang padat. Oleh karena itu, pengembangan sistem identifikasi otomatis menjadi sangat penting untuk meningkatkan efisiensi dan akurasi proses tersebut.

Meskipun teknologi cetak tinggi pada plat kendaraan memiliki keunggulan, masih ada tantangan yang perlu diatasi dalam proses identifikasi otomatis, seperti variasi dalam jenis *font*, ukuran, dan posisi karakter plat, serta kondisi lingkungan yang berubah-ubah sehingga diperlukan pengembangan sistem yang tepat untuk mengoptimalkan proses identifikasi.

Computer Vision adalah cabang ilmu dalam bidang kecerdasan buatan (*Artificial Intelligence*) yang berfokus pada pengembangan sistem komputer yang mampu melihat, memahami, dan menganalisis gambar atau video secara mirip dengan cara manusia. *Computer Vision* bertujuan untuk membuat komputer mampu melihat dunia seperti manusia melalui pengolahan gambar dan video [2]. Dalam *Computer Vision*, komputer diprogram untuk memahami dan menafsirkan informasi visual dari gambar atau video. Ini melibatkan berbagai tugas, termasuk deteksi objek, pengenalan wajah, segmentasi citra, estimasi gerakan, dan banyak lagi. Teknik-teknik dalam *Computer Vision* menggunakan algoritma dan metode matematika untuk menganalisis gambar dan mengekstrak fitur-fitur penting dari data visual.

Aplikasi dari *Computer Vision* sangat luas dan mencakup berbagai bidang, termasuk otomasi industri, kendaraan otonom, pengenalan wajah untuk keamanan, penglihatan mesin dalam kontrol kualitas, navigasi robot, pengolahan medis, *Augmented Reality*, dan masih banyak lagi. Dengan kemajuan teknologi, *Computer Vision* menjadi semakin penting dalam menghasilkan solusi cerdas dalam berbagai industri dan aplikasi sehari-hari.

OpenCV adalah salah satu dari banyak alat yang dapat digunakan untuk menerapkan konsep dan teknik dalam *Computer Vision*. Penggunaan OpenCV dalam penelitian ini merupakan langkah yang tepat. Dengan menggunakan OpenCV, dapat mengakses berbagai fungsi dan algoritma yang berguna untuk mendeteksi objek, mengenali pola, dan melakukan analisis citra dengan efisien. Dalam konteks plat kendaraan, algoritma *Computer Vision* dapat digunakan untuk mendeteksi, mengenali, dan memahami karakter-karakter yang tercetak dalam pada plat kendaraan. Dengan menggunakan teknik pengolahan citra dan pengenalan pola, seperti *Edge Detection*, segmentasi citra, dan pengenalan karakter, sistem *Computer Vision* dapat mengidentifikasi dan menginterpretasi karakter-karakter pada plat kendaraan dengan akurasi tinggi [3].

2. TINJAUAN PUSTAKA

2.1. Pengolahan citra dengan OpenCV

Pengolahan citra adalah bidang studi yang mempelajari tentang manipulasi dan analisis gambar digital. Secara sederhana, pengolahan citra dapat diartikan sebagai proses mengubah atau meningkatkan kualitas gambar digital untuk berbagai tujuan. Secara matematis, citra digital direpresentasikan sebagai fungsi dua dimensi yang memetakan koordinat (x, y) ke nilai intensitas cahaya $I(x, y)$. Nilai intensitas ini dapat bervariasi dari 0 (hitam) hingga 255 (putih) untuk gambar *grayscale*, atau nilai RGB (merah, hijau, biru) untuk gambar berwarna. Tujuan utama pengolahan citra adalah untuk meningkatkan kualitas gambar dan mengekstraksi informasi yang berharga dari gambar tersebut. Pengolahan citra adalah bidang yang terus berkembang dengan banyak potensi untuk aplikasi baru di masa depan [4]. Beberapa

perangkat lunak yang umum digunakan adalah OpenCV, MATLAB, dan Delphi. OpenCV adalah library open-source yang berisi berbagai fungsi dan algoritma untuk pengolahan citra dan *Computer Vision*. OpenCV tersedia dalam berbagai bahasa pemrograman, termasuk C++, Python, dan Java. Pada penelitian ini digunakan OpenCV untuk melakukan berbagai operasi pengolahan citra [5].

Pada penelitian ini akan dilakukan dengan Google Colab atau Colaboratory yang merupakan sebuah layanan gratis dari Google, yang memungkinkan siapa saja untuk mengedit dan mengeksekusi kode Python melalui browser. Ini didasarkan pada konsep buku catatan Jupyter dan sangat populer di kalangan data *scientists*, peneliti, dan pendidik dalam bidang ilmu komputer, khususnya di domain kecerdasan buatan dan data science. Google colab ini mendukung banyak library pemrograman Python populer dan sering diperbaharui untuk memastikan keterjangkauan fungsionalitas terbaru [6].

Colab mendukung berbagai library Python populer yang biasa digunakan dalam machine learning, seperti TensorFlow, PyTorch, Keras, dan OpenCV, serta mengizinkan instalasi library lain jika diperlukan. Karena berbasis cloud, pengguna tidak perlu menginstal perangkat lunak dan terintegrasi dengan Google Drive sehingga memudahkan penyimpanan, akses, dan pengelolaan notebook [7].

2.2. Edge Detection

Edge detection atau deteksi tepi adalah proses pengolahan citra yang bertujuan untuk mendeteksi perubahan tajam dalam intensitas cahaya suatu citra digital. Deteksi tepi bertujuan untuk menandai bagian yang menjadi detail citra dan memperbaiki detail citra yang kabur karena adanya kerusakan atau efek akuisisi data. Deteksi tepi sangat penting dalam penglihatan komputer karena dapat membantu dalam menentukan batas objek, orientasi permukaan, dan perubahan sifat material dalam suatu citra [8].

Beberapa metode deteksi tepi yang umum digunakan dalam pengolahan citra adalah:

2.2.1. Metode Sobel

Metode ini menggunakan operator Sobel untuk mendeteksi tepi. Operator Sobel terdiri dari dua komponen, yaitu komponen mendatar dan komponen vertikal. Keduanya digunakan untuk mendeteksi perubahan intensitas lokal dalam citra.

2.2.2. Metode Robert

Metode ini menggunakan operator Robert untuk mendeteksi tepi. Operator Robert terdiri dari tiga komponen, yaitu komponen mendatar, komponen vertikal, dan komponen diagonal. Keduanya digunakan untuk mendeteksi perubahan intensitas lokal dalam citra.

2.2.3. Metode Canny

Metode ini menggunakan operator Canny untuk mendeteksi tepi. Operator Canny terdiri dari dua

tahap, yaitu tahap pertama untuk mendeteksi tepi dan tahap kedua untuk memperbaiki hasil deteksi. Metode Canny lebih baik dalam melakukan penghitungan deteksi tepi karena outputnya memiliki batas dan tepi yang jelas [9].

Deteksi tepi sering menjadi langkah penting dalam berbagai proses pengolahan citra seperti segmentasi citra, pengenalan pola, dan registrasi citra. Deteksi tepi membantu mempertajam dan memperjelas batas obyek dalam sebuah citra, yang dapat mempermudah interpretasi dan analisis visual [10]. Teknik deteksi tepi dirancang untuk menonjolkan informasi struktural penting sambil mengabaikan variasi yang disebabkan oleh noise, dengan demikian dapat membantu mengurangi noise tanpa mengorbankan informasi penting dalam citra. Dengan menghilangkan informasi yang tidak penting dan memfokuskan hanya pada tepi, kita dapat mengurangi jumlah data yang diproses dan fokus pada informasi yang paling penting untuk analisis lebih lanjut sehingga pemrosesan lebih efisien.

2.3. Tesseract OCR

Tesseract OCR adalah sebuah perangkat lunak pengenalan karakter optik yang dapat mengonversi teks yang ada dalam gambar menjadi teks yang dapat diedit oleh manusia dan mesin. Tesseract awalnya dikembangkan oleh Hewlett-Packard pada tahun 1985 dan kemudian dijadikan sebagai proyek sumber terbuka (open source) di bawah lisensi Apache pada tahun 2005. Referensi web [11].

Tesseract difavoritkan karena akurasi yang tinggi dalam mengenali teks pada berbagai bahasa dan format dokumen. Ia juga dapat digunakan dalam berkombinasi dengan berbagai libraries dan frameworks pemrosesan gambar yang sudah ada, seperti OpenCV, untuk membentuk solusi pengenalan teks yang kuat. Tesseract memungkinkan pengguna untuk mengambil teks dari gambar, potongan-potongan gambar, atau file PDF yang telah di-scan dan mengubahnya menjadi teks yang dapat disimpan dalam berbagai format, yang kemudian bisa diedit, pencarian teks, atau diolah lebih lanjut untuk berbagai tujuan. Guna mencapai akurasi yang optimal, biasanya diperlukan langkah pra-pemrosesan terhadap gambar yang akan diolah oleh Tesseract, seperti penyetelan kontras, pemotongan, dan pembersihan noise, untuk memastikan bahwa teks dapat dikenali dengan sebaik mungkin [12].

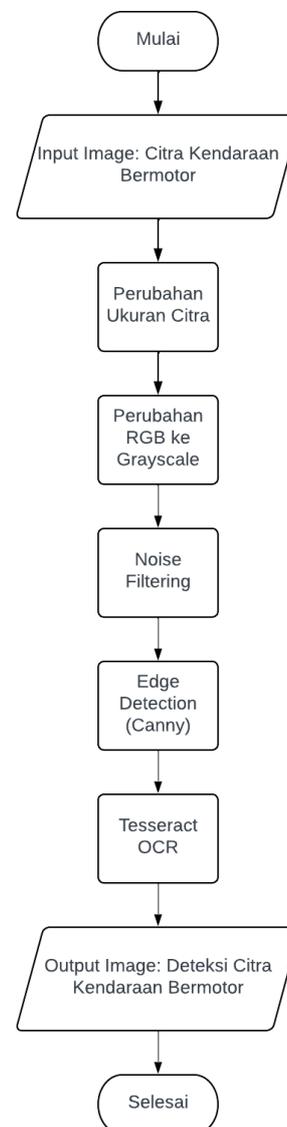
Terdapat beberapa penelitian sebelumnya yang telah dilakukan. Penelitian pertama berjudul “Deteksi Nomor Kendaraan Dengan Metode Connected Component Dan Svm” menunjukkan bahwa pelat kendaraan beroda dua dan beroda empat terdeteksi sebanyak 78% [13]. Kemudian penelitian yang berjudul “pengenalan plat kendaraan bermotor dengan menggunakan metode template matching dan deep belief network”. Studi tersebut menunjukkan presentase keberhasilan mengenali pelat nomor kendaraan sebesar 80% dengan dua puluh kali pengujian [14].

Selanjutnya, penelitian yang berjudul “Penelitian Deteksi Pelat Nomor Kendaraan : Kajian Pustaka”. Meskipun ada satu penelitian yang berhasil mencapai akurasi di atas 90%, metode

pemrosesan gambar dan deep learning masih kurang memuaskan dalam penelitian ini [15].

Berikutnya, penelitian yang berjudul “Deteksi Plat Nomor Kendaraan Bermotor dengan Segmentasi Gambar” Penelitian ini masih memiliki banyak kekurangan dan tidak memiliki bukti yang cukup untuk pendekatan pendeteksian objek plat kendaraan. Faktor-faktor seperti blur gambar, jarak pengambilan gambar, ukuran gambar, dan konversi tipe data gambar adalah beberapa faktor penentu yang belum dilakukan. Sekurang-kurangnya, penelitian ini diharapkan menjadi sumber referensi untuk penelitian tambahan yang bertujuan untuk meningkatkan kemampuan dan efektivitas aplikasi pengolahan gambar ini [16]

3. KONSEP PERANCANGAN



Gambar 1. Diagram Alir Proses Penelitian

Pada penelitian ini, implementasi teknik *embossing* pada plat kendaraan untuk identifikasi otomatis berbasis OpenCV dapat dilakukan dengan langkah-langkah berikut :

Tahap pertama dikumpulkan dataset plat kendaraan sebagai referensi untuk proses identifikasi. Lalu menginstal OpenCV menggunakan pip install opencv-python jika belum terinstal.

Selanjutnya *Import library* yang dibutuhkan untuk pengolahan citra.

Tahap selanjutnya dilakukan pra-pemrosesan data dengan menggunakan metode pengolahan citra seperti *resize image*, mengubah RGB to Grayscale, Noise Filtering dan segmentasi karakter plat kendaraan. Edge detection diterapkan pada citra uji dan kontur untuk mengekstraksi fitur gambar plat kendaraan yang akan dikenali.

Tahap berikutnya diterapkan metode pengenalan karakter seperti *Optical Character Recognition* untuk mengidentifikasi karakter-karakter plat kendaraan. Kemudian dilakukan verifikasi dan validasi hasil identifikasi plat kendaraan dengan membandingkan hasil yang diperoleh dengan database plat kendaraan. Terakhir data identifikasi plat kendaraan yang valid disimpan ke dalam

sistem database untuk keperluan nantinya seperti untuk sistem parkir, tilang elektronik dan sebagainya.

4. HASIL DAN PEMBAHASAN

4.1. Input Citra

Pada tahap ini dikumpulkan citra uji yang berupa plat kendaraan dari berbagai sumber seperti yang terlihat pada Gambar 2. Pada penelitian ini dataset bersumber dari Kaggle dengan berbagai kondisi pencahayaan dan sudut pengambilan gambar. Fungsi `cv2.imread` dalam library OpenCV digunakan untuk membaca gambar dari file. Fungsi ini mengembalikan sebuah objek yang merepresentasikan gambar yang dibaca. Sedangkan `cv2.imshow` adalah sebuah fungsi dalam pustaka OpenCV yang digunakan untuk menampilkan gambar.



Gambar 2. *Input image* untuk citra uji, [<https://www.kaggle.com/datasets/andrewmvd/car-plate-detection>]

4.2. Penerapan Grayscale

Selanjutnya, tahap *grayscale* dalam pengolahan citra yaitu mengkonversi citra berwarna RGB menjadi citra dengan skala abu-abu, yang sangat berguna dalam mengurangi kompleksitas data dan memperjelas karakter yang terdapat pada plat kendaraan. Penggunaan OpenCV memungkinkan kita untuk melakukan pengolahan ini dengan efisien dan menerapkan berbagai teknik deteksi dan pengenalan karakter. Sebelum mengkonversi citra dari format RGB ke *grayscale* yang merupakan langkah dasar dalam pengolahan citra yang dapat menyederhanakan data dan

fokus pada informasi penting, terlebih dahulu dilakukan perubahan ukuran dengan fungsi *resize*. Fungsi ini pada citra sangat berguna dalam berbagai aplikasi pengolahan citra, seperti menyesuaikan ukuran gambar untuk analisis lebih lanjut, meningkatkan kecepatan pemrosesan, atau menyesuaikan gambar ke dalam format yang diinginkan.

OpenCV menyediakan fungsi `cv2.resize` untuk mengubah ukuran citra. Dengan mengubah ukuran gambar pada tahap ini, dapat dipastikan bahwa semua citra yang diproses memiliki dimensi yang konsisten, yang dapat membantu meningkatkan kinerja dan akurasi dalam aplikasi pengolahan citra berikutnya.

Mengkonversi citra RGB ke Grayscale dengan menggunakan `cv2.COLOR_BGR2GRAY` untuk mengubah citra dari warna RGB (standar OpenCV) menjadi *Grayscale*.



Gambar 3. Citra *Grayscale*

4.3. Penerapan Noise Filtering

Pada tahap ini diterapkan *noise filtering* pada citra uji untuk menghilangkan atau mengurangi *noise* yang mungkin terjadi selama pengambilan citra uji atau pada saat pengolahan citra. Ada beberapa metode yang dapat digunakan untuk melakukan *noise filtering*, pada penelitian ini digunakan *filter Bilateral*. Pemilihan filter ini memiliki kemampuan unik untuk mempertahankan tepi (*edges*) sambil menghaluskan area yang tidak mengandung tepi. Ini sangat penting dalam pengolahan gambar pelat kendaraan di mana tepi karakter perlu dipertahankan untuk deteksi dan pembacaan karakter yang akurat. Filter bilateral mengurangi noise tanpa mengaburkan tepi, yang sangat berharga untuk pengenalan karakter optik (OCR). Hasil citra setelah diterapkan fungsi `cv2.bilateralFilter` terlihat pada Gambar 4.



Gambar 4. Citra Noise Filtering

4.4. Penerapan Edge Detection

Berikutnya diterapkan *edge detection* untuk mendeteksi tepi karakter pada citra uji. Pemilihan algoritma Canny karena mampu memberikan deteksi tepi yang akurat dan andal dengan pengurangan *noise* yang efektif. Kombinasi teknik gradien ganda dan histeresis membuatnya sangat efisien dalam menemukan tepi yang signifikan dalam sebuah citra. Hal ini menjadikannya pilihan yang populer dalam berbagai aplikasi pengolahan citra Tabel 1. Hasil Uji Keberhasilan Identifikasi Plat Otomatis

dan *Computer Vision* sehingga langkah selanjutnya diterapkan fungsi algoritma Canny yaitu `cv2.Canny` untuk mendeteksi tepi.



Gambar 5. Citra Deteksi Canny

Untuk menghasilkan citra kontur yang menyoroti karakter pada plat kendaraan, dapat digunakan fungsi `findContours` seperti yang terlihat pada Gambar 6. Untuk ekstraksi dan pengenalan karakter digunakan metode Tesseract OCR (*Optical Character Recognition*) supaya karakter pada plat kendaraan dapat teridentifikasi, hasilnya dapat dilihat pada Gambar 7.



Gambar 6. Citra Deteksi Canny



Gambar 7. Output berupa deteksi citra plat kendaraan dan karakter yang teridentifikasi

Untuk menghitung persentase keberhasilan deteksi plat kendaraan bermotor digunakan rumus seperti yang dituliskan pada persamaan 2.

$$\% = \frac{\text{Jumlah Bagian}}{\text{Jumlah total}} \times 100\% \quad (2)$$

No	Hasil Deteksi Citra Keluaran	Hasil Deteksi Karakter	Persentase Keberhasilan
1		 KEG10A2555	90%
2		 D217 YXR	85.7%
3		 PUB Bes,	71.4%
4		 -eEE KL.01CC 50	100%
5		 B 505 WLG	100%
6		 B 2156 TOR	100%
7		 B 123 CD	100%

Berdasarkan pengujian yang dilakukan berjumlah 7 citra uji dengan berbagai jenis plat kendaraan bermotor diperoleh rata-rata untuk metode tesseract OCR dalam mendeteksi plat nomor untuk diubah menjadi teks adalah sebesar 92,44%. Setelah melakukan pengujian deteksi plat kendaraan bermotor, persentase keberhasilan ditunjukkan seperti pada tabel 1.

Berdasarkan dari data uji diperoleh tingkat keberhasilan 92.44%. Angka 92.44% menunjukkan bahwa sistem deteksi plat kendaraan memiliki performa yang cukup baik dan dapat diandalkan dalam kebanyakan situasi. Namun, masih ada ruang untuk perbaikan untuk mencapai tingkat keberhasilan yang lebih tinggi. Secara keseluruhan, tingkat keberhasilan deteksi sebesar 92.44% menunjukkan sistem yang cukup andal, tetapi masih memiliki beberapa area untuk peningkatan. Tingkat keberhasilan ini bisa dipengaruhi oleh berbagai faktor seperti kondisi pencahayaan, kualitas gambar, kecepatan kendaraan, sudut kamera, dan kejelasan plat nomor. Sistem mungkin bekerja lebih

baik dalam kondisi optimal dan bisa mengalami penurunan kinerja dalam kondisi yang kurang ideal.

Kesalahan pada deteksi plat kendaraan bermotor dari pengujian terjadi bukan saja karena cara pengambilan citra akan tetapi juga karena gagal pada proses segmentasi, sehingga terdapat beberapa angka dan simbol tidak muncul bahkan terjadi kesalahan. Kesalahan deteksi plat kendaraan bermotor ini dapat dilihat pada Tabel 1.

5. KESIMPULAN DAN SARAN

Teknik *embossing* pada plat kendaraan merupakan salah satu cara untuk meningkatkan kejelasan karakter yang terdapat pada plat tersebut sehingga mempermudah proses identifikasi otomatis menggunakan metode pengolahan citra. Setelah dilakukan pengujian, maka dapat disimpulkan bahwa telah terealisasinya penelitian tentang implementasi teknik *embossing* pada plat

kendaraan untuk identifikasi otomatis berdasarkan openCV. Di sini menggunakan berbagai teknik pengolahan citra hingga diperoleh tingkat keberhasilan sebesar 92.44% yang berhasil karakternya teridentifikasi otomatis. Untuk meningkatkan tingkat keberhasilan tersebut ke tingkat yang lebih tinggi, dapat dilakukan beberapa cara diantaranya menggunakan kamera dari sudut pandang yang berbeda untuk mendapatkan citra plat nomor yang lebih lengkap, menambah data uji, menggunakan teknologi AI dan *Machine Learning*, serta melakukan pelatihan ulang model secara berkala dengan data terbaru untuk menangani perubahan dalam pola plat nomor dan kondisi lingkungan. Penelitian ini merupakan langkah awal untuk mendeteksi dan mengidentifikasi plat kendaraan secara otomatis yang nantinya memiliki potensi besar untuk diterapkan dalam berbagai aplikasi praktis. Dengan akurasi yang tinggi dalam mengenali karakter pada plat kendaraan, teknologi ini bisa menjadi dasar untuk sistem parkir otomatis, tilang elektronik, manajemen lalu lintas, pengawasan keamanan, dan analisis data lalu lintas. Implementasi lebih lanjut dan pengujian dalam kondisi nyata akan memperkuat efektivitas dan keandalannya, membuka peluang untuk pengembangan lebih lanjut sesuai kebutuhan.

DAFTAR PUSTAKA

- [1] Brown, K. (2020) 'Modern embossing techniques in industrial applications'. *Industrial Embossing Journal*. Available at: <https://www.industrialembossingjournal.com> (Accessed: 7 May 2024).
- [2] Darnell, R. (2019) 'Introduction to computer vision and image processing'. Available at: <https://www.coursera.org/learn/computer-vision> (Accessed: 7 May 2024).
- [3] Kaehler, A. & Bradski, G. (2016) *Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library*. 1st edn. Sebastopol, CA: O'Reilly Media.
- [4] Aris Budi S, Suma'inna, Hata Maulana, 2016. Pengenalan Citra Wajah Sebagai Identifier Menggunakan Metode. *JURNAL TEKNIK INFORMATIKA*, Volume VOL 9 NO. 2, pp. 166-175.
- [5] Community AlgoStudio. (2024). OpenCV untuk Pengolahan Citra dan Penglihatan Komputer. Available at: <https://community.algostudio.net/open-cv-untuk-pengolahan-citra-dan-penglihatan-komputer/> (Accessed: 8 May 2024).
- [6] Brian Kim, Graham Henke, 2021. Easy-to-Use Cloud Computing for Teaching Data. *Journal of Statistics and Data Science Education*, Volume VOL. 29, NO. S1, p. S103–S111.
- [7] Mark J. Nelson, Amy K. Hoover, 2020. Notes on Using Google Colaboratory in AI Education. Trondheim, Norway, ITiCSE, pp. 533-534.
- [8] Putu Teguh Krisna Putra, N. K. A. W., 2014. Pengolahan Citra Digital Deteksi Tepi Untuk. *MERPATI*, Volume VOL. 2, NO. 2, pp. 253-261.
- [9] Amri, Egi Fransisco Saputra, M. Farel Antonio, Sinyo Kasanova, 2022. Deteksi Tepi pada Pengolahan Citra Digital Menggunakan Matlab. *MDP STUDENT CONFERENCE (MSC)*, pp. 224-230.
- [10] Wang, B., Wang, Y., Sui, X. et al. Gradient domain weighted guided image filtering. *SIViP* 17, 4097–4105 (2023). <https://doi.org/10.1007/s11760-023-02641-9>
- [11] Tesseract Community. (2024). Tesseract OCR. Available at: <https://github.com/tesseract-ocr/tesseract> (Accessed: 8 May 2024).
- [12] Prabowo Budi Utomo, Ibnu Mas'ud Luthfi, M. Nur Fu'ad, M. Mujiono, 2024. Penerapan Optical Character Recognition(OCR) Dengan Text-To-Speech(TTS) dalam Konversi Gambar ke Suara. *Jurnal Vocational Teknik Elektronika dan Informatika*, Volume Vol 11, No 4 , pp. 415-423.
- [13] Budianto, A., Adji, T. B., & Hartanto, R. (2015). Deteksi Nomor Kendaraan Dengan Metode Connected Component Dan Svm. *Jurnal Teknologi Informasi Magister Darmajaya*, 1(01), 106–117.
- [14] M. Michael, F. Tanoto, E. Wibowo, F. Lutan, and A. Dharma, "Pengenalan Plat Kendaraan Bermotor dengan Menggunakan Metode Template Matching dan Deep Belief Network," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 19, no. 1, pp. 27–36, 2019, doi: 10.30812/matrik.v19i1.475.
- [15] D. S. Ramadhansyah and A. Kurniawardhan, "Penelitian Deteksi Pelat Nomor Kendaraan : Kajian Pustaka," *J. Autom.*, vol. 2, no. 1, p. 5, 2019
- [16] Y. O. L. Rema, "Deteksi Plat Nomor Kendaraan Bermotor dengan Segmentasi Gambar," *Jurnal Saintek Lahan Kering*, pp. 20-23, 2019.