



Ekstraksi Karakter Citra Menggunakan Optical Character Recognition Untuk Pencetakan Nomor Kendaraan Pada Struk Parkir

Khairi Ibnuutama*, Mhd. Gilang Suryanata

Program Studi Sistem Informasi, STMIK Triguna Dharma, Medan, Indonesia

Email: ^{1,*}mr.ibnutama@gmail.com, ²suryanata.gilang@gmail.com

Email Penulis Korespondensi: mr.ibnutama@gmail.com

Abstrak—Keamanan pada sarana parkir umum dapat ditingkatkan dengan menambahkan nomor plat kendaraan pada struk parkir. Hal ini bertujuan untuk mencegah terjadinya kesalahan identifikasi saat kendaraan keluar dari sarana parkir akibat kelalaian pemilik kendaraan atau kesengajaan dari pihak yang tidak bertanggung jawab. Nomor plat kendaraan dapat dicetak pada struk parkir dengan cara mengekstrak karakter pada citra kendaraan yang umumnya diakuisisi di portal masuk sarana parkir. Proses ekstraksi karakter tersebut dapat dilakukan dengan metode Optical Character Recognition menggunakan pustaka Tesseract. Tesseract merupakan pustaka Optical Character Recognition yang paling akurat dalam pengenalnya, sehingga proses ekstraksi dapat menghasilkan teks plat nomor kendaraan yang minim kesalahan.

Kata Kunci: OCR, Tesseract, Parkir, Segmentasi, Binerisasi, Nomor Plat Kendaraan

Abstract—Security in public parking facilities can be increased by adding the vehicle plate number to the parking receipt. This aims to prevent misidentification when the vehicle exits the parking facility due to negligence of the vehicle owner or the purpose of an irresponsible party. The license plate number can be printed on the parking receipt by extracting the characters from the vehicle image which is generally acquired at the parking entrance portal. The character extraction process can be done using the Optical Character Recognition method using the Tesseract library. Tesseract is the most accurate Optical Character Recognition library in its recognition, so that the extraction process can produce vehicle license plate text with minimal errors.

Keywords: OCR, Tesseract, Parking, Segmentation, Binerization, License Plate

1. PENDAHULUAN

Saat ini hampir semua sarana parkir yang menggunakan portal masuk dan keluar hanya mengeluarkan struk parkir dengan kode batang (*barcode*) yang dicetak pada struk parkir tersebut. Kode batang pada struk parkir berguna sebagai bukti kepemilikan kendaraan yang di validasi saat pengguna akan keluar dari sarana parkir. Jika struk parkir tersebut dicuri atau tertukar dengan struk parkir milik pengguna lain maka dapat menyebabkan kerugian di salah satu pihak.

Masalah tersebut dapat diatasi dengan menerapkan pengenalan pola pada bidang ilmu pengolahan citra digital yang dapat mendeteksi serta mengekstrak karakter plat nomor kendaraan dari citra hasil akuisisi (foto) kendaraan yang ada. Karakter hasil ekstraksi tersebut dapat ikut dicetak pada struk parkir sebagai salah satu faktor validasi kepemilikan kendaraan yang diharapkan dapat meningkatkan keamanan dan kenyamanan pengguna sarana parkir yang menitipkan kendaraannya.

Salah satu bidang ilmu dalam pengolahan citra digital adalah pengenalan pola (*pattern recognition*). Pengenalan otomatis, deskripsi, dan pengelompokan objek atau data ke dalam jumlah kelas bermakna disebut sebagai pengenalan pola. Pengenalan pola dalam pengolahan citra digital umum digunakan untuk mendeteksi bentuk objek dan mencocokkan pola dari objek tersebut terhadap objek lain. Salah satu cabang dari bidang ilmu pengenalan pola dalam pengolahan citra digital adalah Optical Character Recognition (OCR)[1]. OCR merupakan metode atau konversi elektronik dari teks dalam bentuk citra, tulisan tangan, atau teks yang telah dicetak menjadi teks terkomputerisasi[2] sehingga dapat dimanfaatkan atau dimodifikasi untuk kebutuhan lain. Secara umum terdapat dua hal utama dalam proses OCR yaitu mekanisme ekstraksi ciri dan mekanisme pengenalan. Mekanisme ekstraksi ciri dilakukan untuk mendapatkan identitas dari suatu karakter yang terdapat pada citra. Sedangkan proses pengenalan bertujuan untuk mencocokkan pola karakter pada citra dengan pola yang telah disiapkan dalam basis pengetahuan (*database*).

Deteksi dan pengenalan nomor plat kendaraan adalah langkah awal dalam teknologi License Plate Recognition (LPR) yang membaca karakter nomor plat kendaraan secara otomatis. Sistem LPR yang beroperasi dengan baik mengandung tiga langkah utama seperti deteksi area plat, segmentasi, serta klasifikasi karakter dan digit[3]. Pengenalan nomor plat kendaraan sangat penting dalam berbagai aplikasi seperti sistem parkir kendaraan, keamanan area perbatasan antar daerah, deteksi dan pemantauan lalu lintas, serta parkir bandara[4].

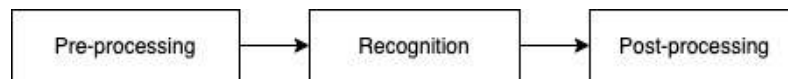
Tujuan dari pembangunan sistem ini diharapkan dapat meningkatkan keamanan dan kenyamanan pengguna sarana parkir, serta memudahkan petugas parkir untuk memvalidasi kepemilikan kendaraan pada pintu keluar sarana parkir.



2. METODOLOGI PENELITIAN

2.1 Optical Character Recognition

Pada penelitian terdahulu[5], [6] disebutkan bahwa OCR merupakan sebuah metode untuk mengubah tulisan tangan dalam bentuk digital agar dapat diperbarui, dan terdiri dari tiga proses utama diantaranya pre-processing[7], recognition[1], [8], dan post-processing[2]. OCR juga berfungsi untuk membedakan antara satu karakter dengan karakter lainnya pada sebuah citra digital[6]. Penelitian lain[4] menyebutkan bahwa OCR merupakan proses mengubah citra menjadi teks yang dapat diperbarui pada komputer, dimana teks dan angka pada citra tidak dapat diubah karena karakter tersebut terdiri dari susunan titik piksel yang membentuk tampilan citra teks dan angka. OCR diklasifikasikan dalam dua jenis, yaitu *offline recognition* dan *online recognition*. Pada jenis OCR *offline* citra uji dapat merupakan hasil pindai dari dokumen, sedangkan OCR *online* merupakan titik-titik yang berurutan direpresentasikan sebagai fungsi waktu serta terdapat urutan dari pola citra tersebut.



Gambar 1. Optical Character Recognition

- Pre-processing*: tahapan ini berfungsi untuk meningkatkan akurasi dalam pengenalan karakter[2], [6] yang meliputi binerisasi, penyusutan derau, normalisasi rasio dan skala, serta pemotongan citra.
- Character recognition*: terdapat empat jenis algoritma OCR yang utama, yaitu Template Matching, Pendekatan Statistik, Analisa Struktural, dan Jaringan Syaraf Tiruan[2].
- Post-processing*: tahapan pengolahan data hasil pengenalan karakter untuk diproses lebih lanjut[6].

2.2 Template Matching

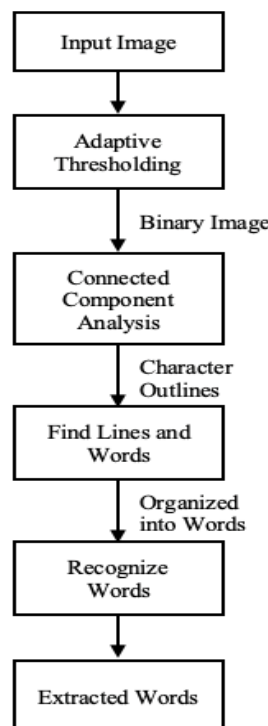
Salah satu pendekatan metode OCR dalam mendeteksi karakter[9] dapat dilakukan menggunakan Template Matching (TM)[10] dengan menghitung nilai galat terkecil[5] untuk mengetahui tingkat kecocokan karakter dari citra masukan[6].

$$\min e = \sum_{(x,y) \in W} (I_{x,y} - T_{x,y})^2 \tag{1}$$

Dimana nilai I merupakan nilai piksel citra yang akan dicocokkan dengan nilai piksel *template* (T). Kecocokan antara citra input dengan salah satu citra template dihitung berdasarkan nilai galat terkecil.

2.3 Tesseract OCR

Tesseract OCR[11] merupakan pustaka[12] yang digunakan untuk mendeteksi karakter pada citra biner berdasarkan analisa distribusi piksel[13] dan pelatihan (training)[10]. Cara kerja Tesseract OCR dapat dilihat pada gambar berikut[14]:



Gambar 2. Cara Kerja Tesseract Engine



Tesseract mungkin merupakan *open source OCR engine* yang paling akurat yang dapat membaca banyak variasi format citra dan melakukan konversi ke dalam bentuk teks lebih dari 60 bahasa[2]. Pelatihan (training) Tesseract dilakukan dengan mempersiapkan sampel huruf (template)[6] berukuran 32x32 piksel sehingga dapat dilakukan pencocokan huruf terhadap template berdasarkan nilai rata-rata intensitas piksel[15]. Nilai rata-rata tersebut dapat dihitung menggunakan persamaan berikut ini:

$$MSE(A, B) = \frac{1}{m \cdot n} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (A[i, j] - B[i, j])^2 \tag{2}$$

Tesseract menganggap citra masukan adalah citra biner, tahapan awal adalah analisa komponen yang terkoneksi dengan *outline* komponen yang tersimpan atau *template*. Pada tahapan ini semua *outline* dikumpulkan tersusun dalam bentuk Blobs[2]. Blobs disusun dalam bentuk baris teks, dimana area dan baris tersebut dianalisa dan diperbaiki ke dalam bentuk teks yang proporsional. Baris teks dipecah menjadi kata-kata berdasarkan jenis spasi karakter menggunakan *definite spaces* dan *fuzzy spaces*[2], [14].

Tahapan pengenalan kemudian dilanjutkan dalam tahapan yang dikenal dengan *adaptive recognition* menggunakan pengenalan bentuk huruf dengan tingkat kepercayaan yang tinggi (*first pass*). Selanjutnya sisa karakter pada tahapan sebelumnya akan dikenali lebih baik pada tahapan selanjutnya (*second pass*)[14]. Tesseract menggunakan pustaka untuk meningkatkan akurasi pada tahapan segmentasi karakter[2].

2.4 Threshold dan Binerisasi

Penelitian terdahulu[2] menyebutkan bahwa teknik *threshold* merupakan salah satu metode segmentasi pada citra yang berfungsi untuk memisahkan latar depan dan latar belakang dengan memilih nilai ambang yang memadai untuk kemudian dikonversi menjadi citra biner. Citra biner tersebut harus mengandung semua informasi penting tentang posisi dan bentuk objek yang terdapat pada citra. Proses konversi dapat dilakukan menggunakan algoritma intensitas[13] sebagai berikut:

$$y' = \begin{cases} 1, & y \geq \text{threshold} \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Proses binerisasi citra dilakukan menggunakan metode Otsu dengan membagi histogram citra *gray level* menjadi dua daerah yang berbeda. Nilai ambang yang akan dicari dinyatakan dengan *k* yang berkisar antara 1 sampai 255 (L). Dimana nilai ambang *k* dapat ditentukan dengan memaksimumkan persamaan[16]:

$$\sigma_B^2(k^*) = \max_{1 \leq k \leq L} \sigma_B^2(k) \tag{4}$$

Dengan:

$$\sigma_B^2(k^*) = \frac{[\mu_T \omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]} \tag{5}$$

2.5 Segmentasi

Pada penelitian sebelumnya[8], Rakesh Kumar Mandal menggunakan teknik *Row-wise Segmentation* dalam pengenalan karakter pada tulisan tangan berbahasa Inggris. Sistem OCR yang dibuat menerapkan segmentasi karakter sebelum dikenali yang bertujuan untuk mengurangi derau, menyusun ulang karakter, dan melakukan *skew*[10] khususnya pada karakter yang tidak lurus. Setiap karakter akan dipotong-potong menjadi beberapa bagian dan dilakukan pengurangan derau (*de-noise*)[6], [10].

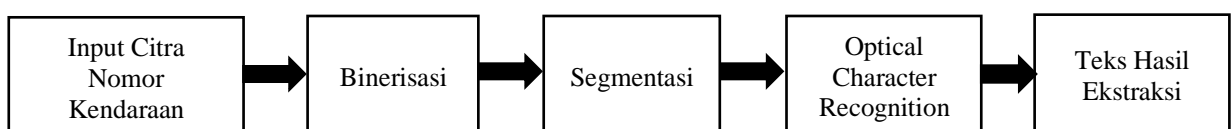
Proses segmentasi dilakukan dengan metode Profil Proyeksi dengan menentukan profil vertikal dan horizontal dari citra, dimana profil vertikal merupakan banyaknya piksel hitam yang tegak lurus terhadap sumbu *y*, sedangkan profil horizontal adalah banyaknya piksel hitam tegak lurus terhadap sumbu *x*[17]. Profil vertikal (*P_v*) berukuran *N* pada baris ke-*i* dan horizontal (*P_h*) berukuran *M* pada kolom ke-*j* didefinisikan sebagai berikut:

$$P_v[i] = \sum_{j=1}^M S[i, j] \tag{6}$$

$$P_h[j] = \sum_{i=1}^N S[i, j] \tag{7}$$

3. HASIL DAN PEMBAHASAN

Pada sistem parkir yang dibuat bekerja dengan mengikuti tahapan-tahapan yang telah disebutkan sebelumnya pada bagian metode penelitian, sehingga dapat disimpulkan alur kerja sistem seperti pada gambar berikut ini:



Gambar 3. Alur Kerja Sistem



Hasil ekstraksi karakter berupa teks yang didapat dari citra plat nomor kendaraan hasil akusisi dan akan dicetak pada struk parkir, beserta data-data lainnya (nomor struk, tanggal, jam masuk, dan lain-lain). Sistem parkir dibuat berbasis website sehingga mendukung pustaka Tesseract dalam bentuk JavaScript.

3.1 Citra Plat Nomor Kendaraan

Citra plat nomor kendaraan diakusisi menggunakan kamera yang terpasang pada portal masuk sarana parkir. Pada sistem yang dibuat, akusisi citra dilakukan dengan menggunakan citra sampel plat nomor kendaraan seperti pada gambar berikut ini.



Gambar 4. Citra Plat Nomor Kendaraan

3.2 Binerisasi

Citra hasil akusisi akan dikonversi menjadi citra biner setelah sebelumnya dilakukan penghapusan derau dan objek lain yang bukan karakter teks plat nomor kendaraan. Proses konversi ini dilakukan untuk memisahkan karakter dari objek lain yang terdapat pada citra. Proses binerisasi dilakukan menggunakan persamaan (3) sehingga mendapatkan hasil konversi yang dapat dilihat pada gambar berikut ini:



Gambar 5. Binerisasi Citra

3.3 Segmentasi

Proses segmentasi pada citra biner hasil konversi dilakukan untuk memilah tiap karakter plat nomor kendaraan sehingga proses pengenalan karakter dilakukan satu persatu. Hasil segmentasi citra biner dapat dilihat pada gambar berikut ini:



Gambar 6. Segmentasi Karakter

3.4 Pengenalan Karakter

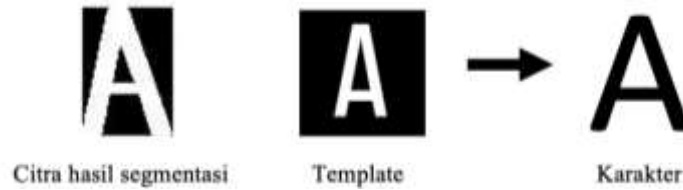
Sistem parkir yang dibuat menggunakan pustaka Javascript Tesseract untuk pengenalan karakter dari citra biner plat nomor kendaraan hasil segmentasi. Tesseract JS akan mengenali tiap-tiap karakter hasil segmentasi pada citra setelah sebelumnya melakukan *training* terhadap *template* karakter. Proses pengenalan karakter citra plat nomor kendaraan (A) terhadap *template* (B) dapat dihitung melalui *Mean Squared Error* antar kedua citra $n \times m$ sehingga didapat nilai kemiripan melalui persamaan berikut[15]:

$$MSE(A, B) = \frac{1}{m \cdot n} \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (A[i, j] - B[i, j])^2 \quad (8)$$

Dimana Tesseract menggunakan empat parameter saat diinisialisasi, yaitu:

1. *Data path*, merupakan lokasi pustaka data yang digunakan sebagai *template* saat mengenali karakter.
2. *Language*, bahasa yang digunakan.
3. *Mode*, mode OCR yang digunakan.
4. *Whitelist*, daftar karakter yang akan diidentifikasi.

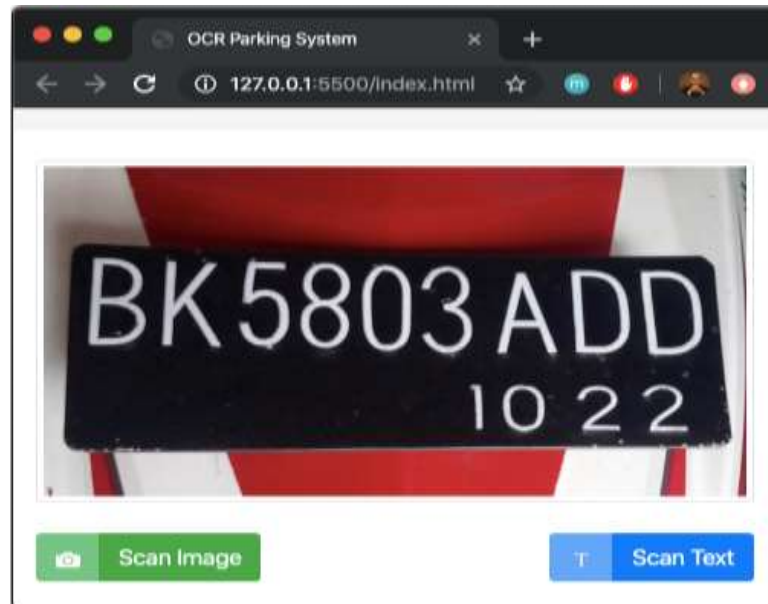
Sehingga didapat hasil deteksi seperti pada gambar berikut:



Gambar 7. Implementasi Tesseract OCR

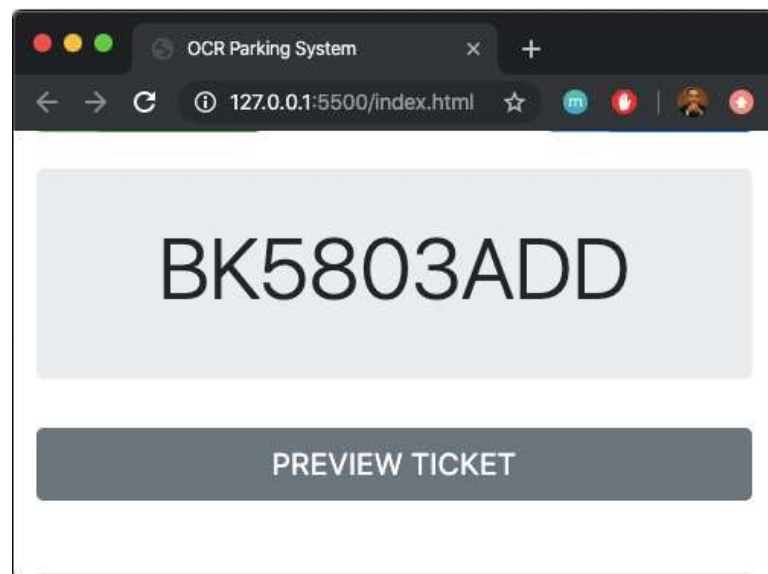
3.5 Implementasi

Implementasi Tesseract JS pada sistem parkir berbasis *website* dilakukan dengan melakukan akuisisi citra plat nomor kendaraan menggunakan kamera.



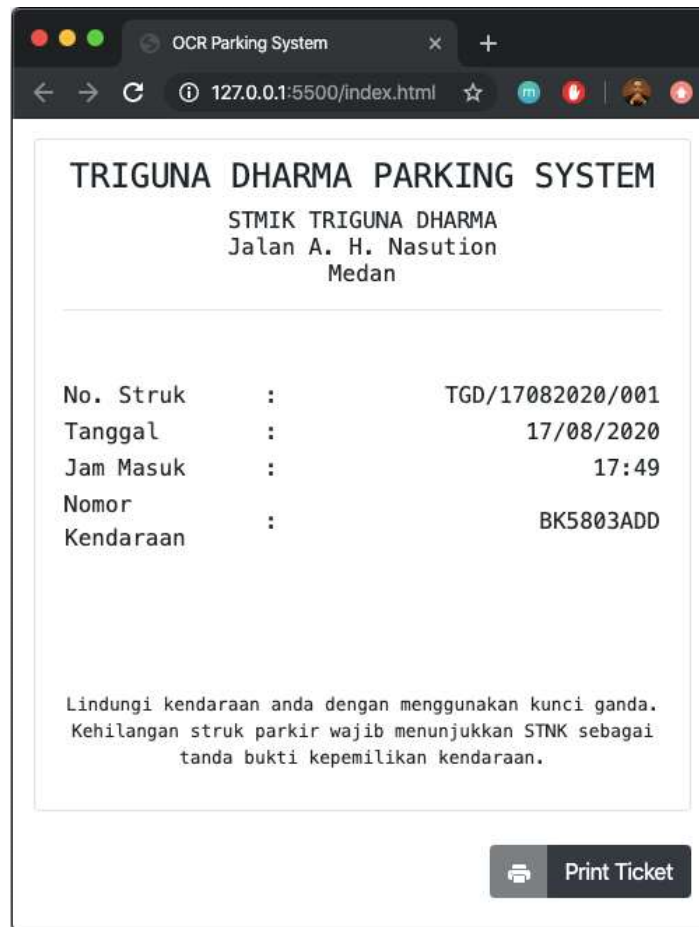
Gambar 8. Akuisisi Citra Plat Nomor Kendaraan

Citra hasil akuisisi akan diekstrak sehingga menghasilkan teks berisi nomor plat kendaraan dengan menekan tombol *Scan Text* yang terdapat pada sistem.



Gambar 9. Akuisisi Citra Plat Nomor Kendaraan

Teks plat nomor kendaraan hasil ekstraksi akan dicantumkan pada struk parkir untuk dicetak dan diserahkan kepada pengguna sarana parkir dengan menekan tombol *Print Ticket*.

**Gambar 10.** Pratampil Struk Parkir Kendaraan

Hasil pengujian terhadap 10 sampel plat nomor kendaraan menunjukkan tingkat akurasi pengenalan karakter menggunakan Tesseract sangat baik. Kesalahan pengenalan karakter yang terjadi disebabkan oleh banyaknya derau pada citra plat nomor kendaraan hasil akuisisi.

Tabel 1. Hasil Uji Coba Sistem

Nomor Plat	Hasil Ekstraksi	Akurasi
BK 5083 ADD	BK5083ADD	100%
BK 2320 MAU	BK2320HAU	88,9%
BK 2807 VBA	BK2807VBA	100%
BK 2995 AAR	BK2995AAR	100%
BK 3956 ADA	BK3956ADA	100%
BK 5157 LQ	BK5157LQ	100%
BK 3471 NY	BK3471NY	100%
BK 2635 RAK	BK2635RAK	100%
BK 2862 CL	BKZ862CL	75%
BK 4219 AFK	BK4219AFK	100%

4. KESIMPULAN

Hasil implementasi sistem pada 10 sampel nomor plat kendaraan menghasilkan rata-rata persentase akurasi hasil ekstraksi sebanyak 96,39%. Rata-rata persentase tersebut menunjukkan bahwa Tesseract dapat melakukan pengenalan karakter teks pada citra dengan sangat baik. Penyebab utama kesalahan pengenalan yang terjadi adalah banyaknya derau dan objek lain pada citra plat nomor kendaraan hasil akuisisi sehingga menyebabkan proses segmentasi ikut memisahkan objek yang bukan karakter. Derau yang terdapat pada citra hasil akuisisi berasal dari debu, kotoran, *sticker*, atau bias dan refleksi cahaya. Proses pengenalan karakter akan lebih akurat jika plat nomor kendaraan memiliki warna latar putih dengan teks hitam sehingga meminimalisir bias atau refleksi cahaya yang tertangkap pada saat akuisisi. Hal ini dibuktikan pada penelitian sebelumnya yang mendeteksi karakter pada plat nomor kendaraan dengan warna latar belakang putih[4], [10].



UCAPAN TERIMAKASIH

Puji dan syukur dipanjatkan kepada Tuhan Yang Maha Esa sehingga penelitian ini dapat berjalan lancar sebagaimana yang diharapkan. Ucapan terima kasih yang sebesar-besarnya kepada Pemerintah dan Kementerian Riset dan Teknologi melalui LLDIKTI Wilayah 1 sebagai sumber dana penelitian ini. Ucapan terima kasih juga disampaikan peneliti kepada STMIK Triguna Dharma sebagai homebase peneliti yang telah memberikan dukungan serta akses sebagai tempat untuk melakukan penelitian, bapak Mhd. Gilang Suryanata yang telah memberikan kontribusi pada penelitian, serta keluarga dan rekan sejawat sesama dosen yang telah memberikan dukungan pengetahuan dan moril.

REFERENCES

- [1] S. S. Patil and A. S. Bhalchandra, "Pattern Recognition Using Genetic Algorithm," in *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 2017, no. x, pp. 310–314.
- [2] S. Tangwannawit and W. Saetang, "Recognition of Lottery Digits Using OCR Technology," in *2016 12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, 2016, pp. 632–636.
- [3] A. C. Roy, M. K. Hossen, and D. Nag, "License Plate Detection and Character Recognition System for Commercial Vehicles Based on Morphological Approach and Template Matching," *Electr. Eng. Inf. Commun. Technol. (ICEEICT), 2016 3rd Int. Conf.*, pp. 1–6, 2016.
- [4] S. S. Omran and J. A. Jarallah, "Iraqi Car License Plate Recognition Using OCR," *2017 Annu. Conf. New Trends Inf. Commun. Technol. Appl. NTICT 2017*, no. March, pp. 298–303, 2017.
- [5] P. Hidayatullah, F. Feirizal, H. Permama, Q. Mauluddiah, and A. Dwitama, "License Plate Detection and Recognition for Indonesian Cars," *Int. J. Electr. Eng. Informatics*, vol. 8, no. 2, pp. 331–346, Jun. 2016.
- [6] K. Ibtnutama, Z. Panjaitan, and E. F. Ginting, "Modifikasi Metode Template Matching pada OCR Untuk Meningkatkan Akurasi Deteksi Plat Nomor Kendaraan," *J. Teknol. Sist. Inf. dan Sist. Komput. TGD*, vol. 2, no. 2, pp. 21–29, 2019.
- [7] J. B. L. Bernardo and L. J. M. Raboy, "Vehicle Plate Monitoring and Information System Using Optical Character Recognition (OC) Technique," *SSRN Electron. J.*, vol. 1, no. December, pp. 1–6, 2015.
- [8] S. K. Henge and B. Rama, "Comparative Study With Analysis of OCR Algorithms and Invention Analysis of Character Recognition Approached Methodologies," in *2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES)*, 2016, pp. 1–6.
- [9] S. Lukas, P. Yugopustito, and D. Krisnadi, "Identification of Indonesian Vehicle Registration Plate by Adaptive Thresholding and Region Labeling Algorithm," *Int. Conf. ICT Knowl. Eng.*, pp. 1–4, 2013.
- [10] A. S., J. Yankey, and E. O., "An Automatic Number Plate Recognition System using OpenCV and Tesseract OCR Engine," *Int. J. Comput. Appl.*, vol. 180, no. 43, pp. 1–5, 2018.
- [11] R. Smith, "An Overview of the Tesseract OCR Engine," *Lect. Google Code. Google Inc.*, pp. 629–633, 2007.
- [12] M. Koistinen, J. Kervinen, and K. Kettunen, "How to Improve Optical Character Recognition of Historical Finnish Newspapers Using Open Source Tesseract OCR Engine," *8th Lang. Technol. Conf. Hum. Lang. Technol. as a Chall. Comput. Sci. Linguist.*, no. November, pp. 279–283, 2018.
- [13] S. W. Utama and A. Kusumawardhani, "Aplikasi Pendeteksi Plat Nomor Negara Indonesia Menggunakan OpenCV dan Tesseract OCR pada Android Studio," no. December, 2017.
- [14] M. K. Audichya, "A Study to Recognize Printed Gujarati Characters Using Tesseract OCR," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. V, no. IX, pp. 1505–1510, 2017.
- [15] B. Nunamaker, S. S. Bukhari, D. Borth, and A. Dengel, "A Tesseract-based OCR Framework for Historical Documents Lacking Ground-truth Text," in *2016 IEEE International Conference on Image Processing (ICIP)*, 2016, pp. 3269–3273.
- [16] I. K. G. D. Putra and I. G. Suarjana, "Segmentasi Citra Retina Digital Retinopati Diabetes Untuk Membantu Pendeteksian Mikroaneurisma," *J. Tek. Elektro*, vol. 9, no. 1, pp. 44–49, 2010.
- [17] A. Septiarini, "Segmentasi Karakter Menggunakan Profil Proyeksi," *J. Inform. Mulawarman*, vol. 7, no. 2, pp. 66–69, 2012.