

Deteksi Jenis Kendaraan Berdasarkan Nomor Plat Menggunakan Metode Vertical Edge Detection dan Connected Component Labelling

Aldi Pranadia¹, Ema Rachmawati², Risnandar³

^{1,2,3}Fakultas Informatika, Universitas Telkom, Bandung

¹aldipranadia@students.telkomuniversity.ac.id, ²emarachmawati@telkomuniversity.ac.id,

³risnandartelyu@telkomuniversity.ac.id

Abstrak

Plat nomor kendaraan bermotor merupakan tanda kendaraan bermotor di Indonesia. Pada plat nomor kendaraan di Indonesia terdiri dari kombinasi huruf dan angka, yang memiliki informasi kode wilayah, jenis/tipe kendaraan bermotor dan nomor registrasi dari kendaraan bermotor tersebut. Saat ini, jalan tol di Indonesia sudah menggunakan *Automatic Vehicle Classification* (AVC) untuk menentukan tarif dan golongan kendaraan bermotor di jalan tol yang diidentifikasi berdasarkan pada jenis/golongan kendaraan bermotor. Sistem AVC relatif menggunakan beberapa sensor dan tentunya membutuhkan biaya sensor yang bervariasi. Berdasarkan analisis tersebut, maka dibutuhkan sistem yang dapat mendeteksi jenis kendaraan bermotor di jalan tol. Pada penelitian ini, penulis menggunakan *Vertical Edge Detection* (VEDA) untuk mendeteksi lokasi plat nomor kendaraan bermotor, metode *Connected Component Labelling* (CCL) untuk mendeteksi dan mensegmentasi karakter pada plat dan metode *K-Nearest Neighbor* (KNN) untuk mengenali setiap karakter pada plat nomor kendaraan bermotor. Dataset yang digunakan pada pengujian ini berupa 162 dataset citra kendaraan dan 2.734 citra huruf dan angka sebanyak 36 kelas. Pada pengujian ini, sistem mampu mengenali jenis kendaraan bermotor berdasarkan nomor plat dengan akurasi sebesar 71,81%, *recall* sebesar 94,59%, *precision* sebesar 74,88% dan *F1-Score* sebesar 83,59%. Hasil tersebut menunjukkan kelas terdeteksi dengan baik oleh model yang diusulkan dalam penelitian.

Kata kunci : plat nomor kendaraan bermotor, kendaraan bermotor, *Vertical Edge Detection* (VEDA), *Connected Component Labelling* (CCL), *K-Nearest Neighbor* (KNN), jalan tol

Abstract

A license plate is a sign of a motor vehicle in Indonesia. The motor vehicle license plates in Indonesia consists of letters and number, that has information of regional code, type of motor vehicle and registration number of a motor vehicle. Currently, the highway in Indonesia has to use *Automatic Vehicle Classification* (AVC) to determine the price and class of motor vehicles on the highway that are identified based on type/class of motor vehicle. The AVC system relatively uses several sensors and of course requires varying sensor costs. Based on that analysis, a system that can detect the type of motor vehicle on the highway is needed. In this study, the authors use the *Vertical Edge Detection* (VEDA) method to detect the location of license plates, the *Connected Component Labelling* (CCL) method to detect and segment the character on license plates, and the *K-Nearest Neighbor* (KNN) method to recognize each character on the license plates. The dataset used in this test is 162 motor vehicle datasets and 2,734 characters with 36 classes. In this test, the system can identify the type of motor vehicle based on license plates with an accuracy of 71.81%, recall of 94.59%, precision of 74.88% and F1-Score of 83.59%. These results show that the class is detected properly by the model proposed in the study.

Keywords: license plate, motor vehicle, *Vertical Edge Detection* (VEDA), *Connected Component Labelling* (CCL), *K-Nearest Neighbor* (KNN), highway

1. Pendahuluan

Latar Belakang

Setiap harinya manusia selalu bepergian dari suatu tempat ke tempat lainnya untuk melakukan berbagai aktivitas, khususnya masyarakat Indonesia. Terkadang jarak yang ditempuh cukup jauh dan memakan banyak waktu, selain itu juga semakin lama perjalanan maka akan membuat volume kendaraan meningkat di waktu tersebut dan menimbulkan kepadatan lalu lintas. Berdasarkan badan pusat statistik perkembangan volume kendaraan meningkat cukup signifikan setiap tahunnya [1].

Maka dari itu untuk mempersingkat waktu perjalanan dan keteraturan lalu lintas berdasarkan PP No. 15 Tahun 2005 diselenggarakan suatu sistem manajemen transportasi yaitu jalan tol. Dalam penggunaannya pengendara akan dikenakan tarif berdasarkan golongan kendaraan bermotor di jalan tol, dimana golongan tersebut dibedakan berdasarkan jenis/tipe kendaraannya [2]. Sampai saat ini jalan tol menggunakan sistem AVC (*Automatic Vehicle Classification*) untuk menentukan tarif dan golongan kendaraan [3], dimana sistem tersebut relatif

menggunakan beberapa sensor dan tentunya membutuhkan biaya sensor yang bervariasi dalam penggunaannya. Berdasarkan Peraturan Kapolri No 5 Tahun 2012 Tentang Registrasi dan Identifikasi Kendaraan Bermotor, bahwasannya plat nomor kendaraan merupakan identitas kendaraan bermotor yang terdiri dari informasi kode wilayah, kode jenis/tipe kendaraan, dan nomor registrasi. Sistem Pengenalan Plat Nomor akan membantu menyederhanakan/menggantikan sistem AVC untuk mendeteksi jenis kendaraan berdasarkan nomor plat kendaraan untuk penentuan golongan kendaraan di jalan tol.

Riset-riset mengenai pengenalan plat nomor kendaraan bermotor sudah sering dilakukan dalam rangka memecahkan berbagai permasalahan, diantaranya membuat sistem yang akan mengenali plat nomor kendaraan dengan menggunakan kamera sebagai alat utamanya. Pada suatu penelitian sebagian besar menggunakan *edge detection* seperti metode *morphological*, *genetic algorithm*, *Bayesian* dan *canny edge detection* [4]. Terdapat kelebihan dan kelemahan dalam melakukan deteksi tepi, terkadang pada sebuah citra digital akan diperoleh hasil pendeteksian tepi yang lebih jelas menggunakan salah satu operator tertentu tergantung kondisi citra.

Beberapa sistem menggunakan metode *Vertical Edge Detection* (VEDA) untuk mendeteksi plat pada kendaraan bermotor. Metode ini membuat proses deteksi pada plat kendaraan lebih fleksibel dan lebih efisien dengan berbagai kondisi citra [5]. Oleh karena itu peneliti sedikit menyesuaikan metode *Vertical Edge Detection* (VEDA) agar dapat digunakan pada penelitian ini.

Oleh karena itu, penulis membangun Sistem Deteksi Jenis Kendaraan Berdasarkan Nomor Plat Kendaraan menggunakan metode *Vertical Edge Detection* (VEDA) dan *Connected Component Labelling* (CCL). Metode *Vertical Edge Detection* digunakan untuk mendeteksi lokasi plat pada suatu citra kendaraan dengan memanfaatkan garis tepi vertikal pada plat kendaraan [5], kemudian metode *Connected Component Labelling* digunakan untuk mendeteksi dan mensegmentasi setiap karakter pada plat kendaraan dengan memanfaatkan teori *connectivity* antar piksel pada citra [6].

Topik dan Batasannya

Berdasarkan latar belakang masalah penelitian ini, dapat disimpulkan bahwa topik permasalahan yang diselesaikan yaitu membangun sistem yang dapat mengenali jenis kendaraan berdasarkan nomor plat menggunakan metode *Vertical Edge Detection* (VEDA) dan *Connected Component Labelling* (CCL) untuk menentukan golongan kendaraan di jalan tol. Proses deteksi jenis kendaraan terdiri dari beberapa tahap yang dilakukan yaitu, mencari dan menentukan lokasi plat pada suatu citra kendaraan dengan menggunakan *Vertical Edge Detection* (VEDA), kemudian plat tersebut akan dilakukan proses segmentasi untuk memisahkan karakter dari plat dengan menggunakan *Connected Component Labelling* (CCL). Setelah proses segmentasi selesai, ekstraksi fitur dilakukan untuk mendapatkan informasi berupa nilai piksel setiap karakter pada plat kendaraan. Hasil ekstraksi fitur tersebut digunakan untuk proses klasifikasi dengan menggunakan *K-Nearest Neighbor* (KNN) yang bertujuan untuk mengenali karakter pada plat kendaraan.

Batasan permasalahan pada penelitian ini yaitu :

1. Dataset citra kendaraan merupakan rekaman CCTV di gerbang tol yang sudah di konversi menjadi citra/frame dengan format JPG resolusi 1920x1080 sebanyak 162 data.
2. Dataset citra karakter terdiri dari 26 kelas huruf abjad (A-Z) dan 10 kelas angka (0-9) sebanyak 2.734 data.
3. Mengidentifikasi jenis kendaraan Golongan I (Sedan, MPV/Hatchback/City Car, SUV, Jip, Pick Up/Truk Kecil dan Bis) dan non Golongan I (Truk Besar).
4. Plat kendaraan bermotor roda 4 Golongan I berasal dari daerah : Sumatera, Jawa, Bali, Nusa Tenggara, Kalimantan, Gorontalo, Sulawesi, Maluku dan Papua.

Tujuan

Dengan menggunakan beberapa metode yang telah dijelaskan sebelumnya, penulis bertujuan untuk melakukan analisa terhadap tingkat akurasi sistem mampu mendeteksi jenis kendaraan berdasarkan nomor plat.

Organisasi Tulisan

Penelitian ini dibagi menjadi 5 bagian. Bagian pertama mengenai latar belakang mengapa penelitian dilakukan. Pada bagian kedua dijelaskan mengenai studi terkait dengan penelitian. Pada bagian ketiga dijelaskan mengenai sistem yang dibangun. Pada bagian keempat dijelaskan mengenai hasil dan evaluasi dari sistem yang telah dibangun. Pada bagian kelima dijelaskan hasil penelitian yang telah dilakukan.

2. Studi Terkait

Pengenalan Plat Kendaraan

Vijaya B. Ghule (2015) telah melakukan penelitian tentang pengenalan plat kendaraan menggunakan metode *Vertical Edge Detection* (VEDA), metode ini membuat proses deteksi pada plat kendaraan lebih efisien dan fleksibel karena metode ini berfokus pada kontras nilai abu-abu [5]. Dalam metode ini input citra yang diterima harus berupa citra biner yang telah diproses menggunakan metode *Adaptive Thresholding*. Kemudian metode

Unwanted Line Elimination (ULEA) diusulkan untuk membantu deteksi plat kendaraan menjadi lebih akurat dan cepat, karena metode ini bertujuan untuk menghilangkan garis-garis yang tidak digunakan pada hasil *Adaptive Thresholding*. Hasil dari deteksi tepi pada plat berdasarkan metode tersebut dalam mengidentifikasi plat kendaraan menghasilkan kinerja deteksi tepi yang akurat dan efisien [5].

Pengenalan Nomor Plat Kendaraan

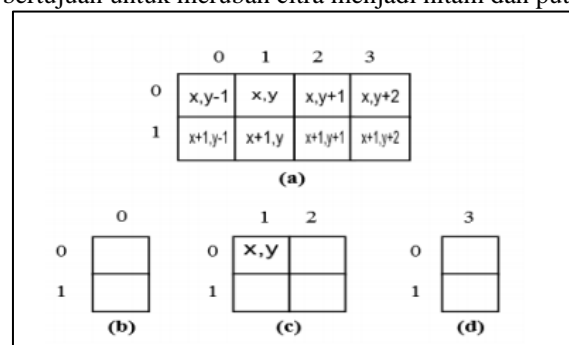
Ana Riza F. Quiros (2017) telah melakukan penelitian tentang pengenalan nomor plat kendaraan dengan menggunakan metode *KNN* dan mendapatkan akurasi 87,43%, pada penelitian ini dibangun suatu sistem yang dapat mengenali plat nomor kendaraan secara otomatis menggunakan kamera. Kamera akan mengambil citra kendaraan di jalan dan mengolah citra untuk mendapatkan plat kendaraan. Setelah plat dideteksi sistem akan melakukan proses segmentasi karakter untuk memisahkan tiap karakter pada plat. Setiap karakter yang sudah disegmentasi akan diidentifikasi menggunakan algoritma *KNN*.

Pengenalan Jenis Kendaraan

Jenis kendaraan secara umum dibagi menjadi dua yaitu kendaraan penumpang dan kendaraan barang. Kendaraan penumpang terdiri dari kendaraan pribadi (Sedan, MPV/Hatchback/City Car, SUV dan Jip), Microbus, minibus dan bis. Kendaraan barang terdiri dari pickup bak terbuka, pickup kabin ganda, pickup box, blind van dan truk[7]. Jenis kendaraan tersebut dapat dikenali dari plat nomor kendaraan, karena plat nomor kendaraan terdiri dari kombinasi angka dan huruf yang menunjukkan identitas suatu kendaraan seperti asal wilayah, jenis kendaraan, tipe kendaraan dan asal kota/kabupaten.

Vertical Edge Detection

Vertical Edge Detection adalah sebuah algoritma yang digunakan untuk mencari atau membedakan garis-garis pada daerah plat kendaraan [5]. *Vertical Edge Detection* hanya berfokus pada daerah hitam dan putih, maka dari itu sebelum masuk proses *Vertical Edge Detection*, citra harus diproses terlebih dahulu menggunakan metode *Adaptive Thresholding* yang bertujuan untuk merubah citra menjadi hitam dan putih.

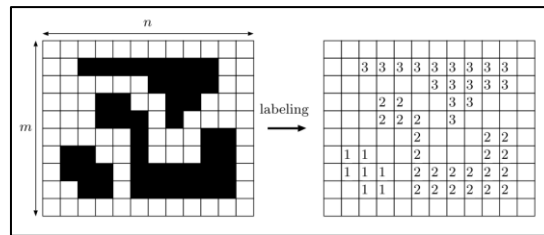


Gambar 1. Desain mask yang diusulkan pada VEDA [5]

Metode *Vertical Edge Detection* membutuhkan mask berukuran 2x4 untuk memeriksa piksel yang berwarna hitam. Pada Gambar 1, kita dapat melihat bahwa mask dibagi menjadi tiga bagian. Gambar 1(b) merupakan mask kiri, Gambar 1(c) merupakan mask tengah, dan Gambar 1(d) merupakan mask kanan. Piksel (0,1) dan (1,1) pada bagian mask tengah merupakan nilai pusat dari mask tersebut. Pada Gambar 1, kotak atau mask 2x4 akan berpindah dari kiri ke kanan dan dari atas ke bawah. Jika 2 piksel pada mask kiri (0,0) dan (0,1) berwarna hitam, maka semua nilai piksel pada mask kiri adalah 0. Jika 4 piksel pada mask tengah (0,1), (1,1), (0,2), (1,2) berwarna hitam, maka semua nilai piksel pada mask tengah adalah 0. Jika 2 piksel pada mask kanan (0,3) dan (1,3) berwarna hitam, maka semua nilai piksel pada mask kanan adalah 0. Jika tiga bagian selain nilai pusat pada mask (0,0), (1,0), (0,2), (1,2), (0,3), (1,3) berwarna hitam maka mask pusat (0,1) dan (1,1) akan diubah menjadi warna putih [5].

Connected Component Labelling

Connected Component Labelling merupakan algoritma yang diterapkan berdasarkan teori *connectivity* dan dapat digunakan sebagai metode segmentasi elemen pada citra [6]. *Connected Component Labelling* biasa juga dikenal sebagai *Connected Component Analysis* karena setelah proses pelabelan, objek akan dianalisis untuk mendapatkan objek yang sesuai dengan fitur yang telah ditentukan. Proses pelabelan biasanya dilakukan untuk menandai objek yang terhubung dengan citra biner. Proses pelabelan citra biner dilakukan dengan menempelkan label serupa pada piksel yang saling berdampingan. Hasil pelabelan menggunakan metode *Connected Component Labelling* ditentukan pula oleh jumlah *connectivity* yang digunakan pada saat melakukan pengecekan label tetangga terdekat. Pengecekan label tetangga terdekat dapat dilakukan dengan dua acara, yaitu 4-konektivitas atau 8-konektivitas [6].



Gambar 2. Ilustrasi Connected Component Labelling [8]

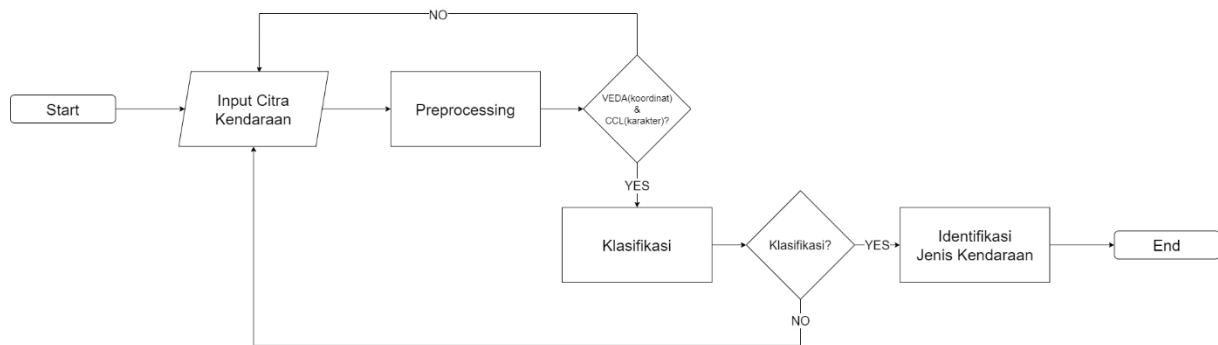
K-Nearest Neighbor

K-Nearest Neighbor adalah salah satu metode *machine learning* yang dapat menyimpan semua data pelatihan untuk proses klasifikasi dengan menggunakan teori ketetanggaan [9]. *KNN* merupakan algoritma yang menyimpan semua data yang ada dan kemudian dijadikan model untuk proses klasifikasi. Kelebihan dari metode ini adalah dapat digunakan pada data latih yang memiliki noise dan sangat efektif digunakan pada data latih yang sangat besar [9].

Sistem kerja dari metode *K-Nearest Neighbor* adalah pertama kita tentukan parameter nilai *K* terlebih dahulu untuk melihat jarak terdekat dari sebuah data baru/data test dengan data latih, kemudian *Euclidean* adalah salah satu rumus yang digunakan untuk menghitung jarak antara data baru/data test dengan data latih yang sudah memiliki label.

$$d_{Euclidean}(x, u) = \sqrt{\sum_{i=1}^n (x^{(i)} - u^{(i)})^2} \tag{1}$$

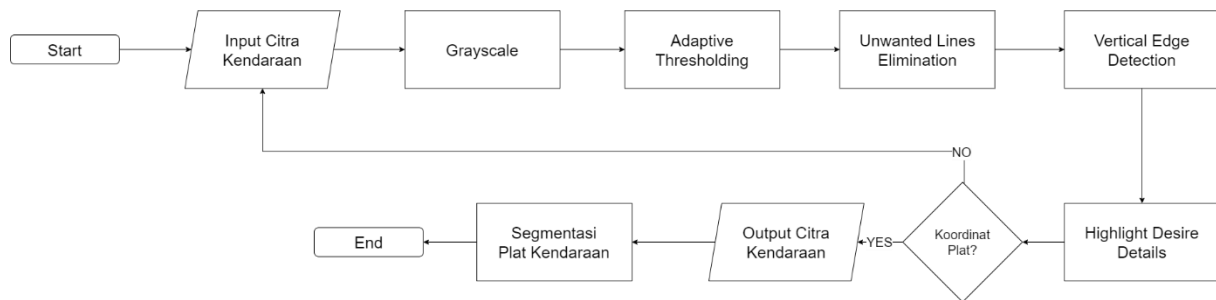
3. Sistem yang Dibangun



Gambar 3. Skema umum proses identifikasi jenis kendaraan

Pada penelitian tugas akhir ini, penulis bertujuan untuk membuat sistem yang bisa mengenali nomor plat pada sebuah plat kendaraan. Objek yang diterima oleh sistem berupa citra kendaraan yang memiliki plat nomor. Citra kendaraan merupakan hasil rekaman *CCTV* di gerbang tol yang sudah dikonversi menjadi citra/frame, rekaman *CCTV* dari gerbang tol diharuskan untuk menjamin bahwa citra kendaraan merupakan kendaraan bermotor roda 4 atau lebih bukan kendaraan bermotor roda dua. Citra tersebut akan diproses untuk mempermudah sistem dalam mencari lokasi area plat kendaraan dan juga mensegmentasi setiap karakter dari plat kendaraan. Apabila lokasi plat kendaraan berhasil terdeteksi dan karakter berhasil disegmentasi dengan sempurna maka akan masuk ke proses klasifikasi untuk mengenali setiap huruf dan angka dengan menggunakan model klasifikasi *KNN*, namun bila gagal akan kembali menginputkan citra kendaraan lain. Kemudian apabila klasifikasi berhasil akan dilanjutkan ke proses identifikasi jenis kendaraan, namun bila gagal akan kembali menginputkan citra kendaraan yang lain.

3.1 Preprocessing



Gambar 4. Skema proses preprocessing (VEDA)

3.1.1 Grayscale

Citra kendaraan yang diterima merupakan citra *RGB* (Red, Green, Blue) maka dari itu citra perlu diubah kedalam bentuk *grayscale* atau skala abu-abu agar citra dapat diproses di tahap selanjutnya. Ada beberapa rumus yang dapat digunakan untuk mengkonversi citra *RGB* menjadi *grayscale* [10].

$$Y = 0.299 R + 0.587 G + 0.114 B \quad (2)$$

3.1.2 Adaptive Thresholding

Citra yang diterima pada proses *adaptive threshold* adalah citra *grayscale* yang memiliki nilai antara 0-255. Maka selanjutnya akan dilakukan proses *threshold* pada citra [11], setiap nilai pada piksel akan diubah dengan batas tertentu yang nantinya akan merubah nilai setiap piksel menjadi 0 atau 255 yang menghasilkan citra hitam dan putih.



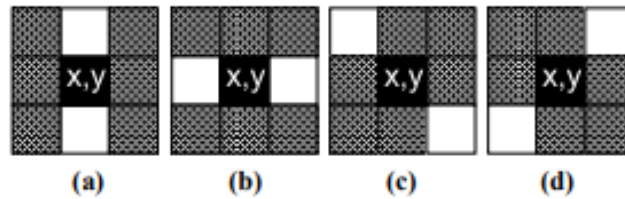
Gambar 5. Input citra Grayscale



Gambar 6. Output citra Adaptive Threshold

3.1.3 Unwanted Lines Elimination Process

Unwanted Lines Elimination Algorithm (ULEA) adalah suatu algoritma yang bertujuan untuk menghilangkan garis-garis atau piksel yang tidak sesuai/tidak diperlukan pada suatu citra. Proses thresholding pada umumnya akan menghasilkan banyak garis tipis yang tidak termasuk kedalam daerah plat kendaraan, maka dari itu proses *ULEA* ini akan berkontribusi dalam hal akurasi dan waktu pemrosesan.



Gambar 7. (a) untuk kondisi garis vertikal (90°), (b) untuk kondisi garis horizontal (0°), (c) untuk kondisi garis diagonal (45°), (d) untuk kondisi garis diagonal (135°)

Untuk menjaga detail plat kendaraan, *mask* 3x3 digunakan di seluruh piksel citra. Hanya nilai piksel hitam pada gambar ambang yang diuji, diasumsikan bahwa $g(x,y)$ adalah nilai untuk citra ambang batas. Apabila salah satu kondisi pada Gambar 7 terpenuhi, maka citra ambang batas $g(x,y)$ akan diubah menjadi warna putih [11].



Gambar 8. Input citra *Adaptive Threshold*



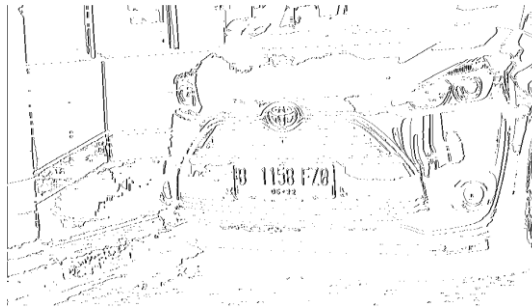
Gambar 9. Output citra *ULEA*

3.1.4 Vertical Edge Detection Process

Vertical Edge Detection Algorithm (VEDA) digunakan untuk mendeteksi garis tepi pada plat kendaraan. Algoritma ini memanfaatkan *mask* 2x4 dan bergerak dari atas ke bawah dan dari kiri ke kanan [5]. Pada proses ini garis tepi dimunculkan dengan ketentuan *VEDA* (x,y) dan *VEDA* $(x+1,y) = 255$ apabila $img(x,y)$ dan $img(x+1,y) = 0$, $img(x,y+1)$ dan $img(x+1,y+1) = 0$, $img(x,y+2)$ dan $img(x+1,y+2) = 0$, $img(x,y-1)$ dan $img(x+1,y-1) = 0$. Dengan nilai x menunjukkan tinggi citra dan y menunjukkan lebar citra.



Gambar 10. Input citra *ULEA*



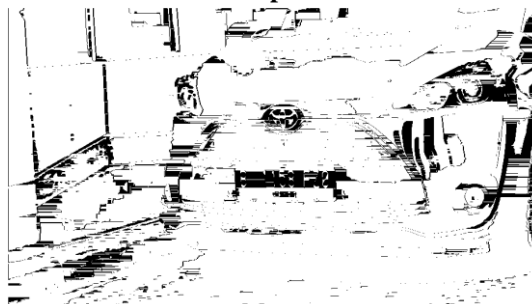
Gambar 11. Output citra VEDA

3.1.5 Highlight Desired Details

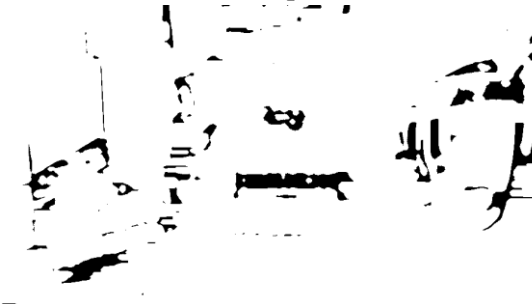
Setelah proses *VEDA* selesai, maka selanjutnya akan dilakukan pengisian piksel hitam pada daerah yang terletak diantara dua tepi vertikal [11]. Sebelum melakukan pengisian piksel hitam di antara 2 garis vertikal akan dilakukan perhitungan terlebih dahulu pada jarak antara 2 garis vertikal tersebut, apabila dua garis vertikal tersebut memiliki jarak horizontal (hd) yaitu $hd < 40$ maka daerah diantara 2 garis vertikal tersebut dapat dihitamkan [12]. Gambar 13 merupakan hasil dari penghitaman daerah diantara 2 garis vertikal yang telah memenuhi kondisi yang telah ditentukan sebelumnya, kemudian pada tahap ini *median blur* diusulkan untuk menghilangkan elemen yang tidak diperlukan dan juga menggunakan fitur *cv2.findContours()* dari OpenCV untuk mengambil objek yang telah dihitamkan sebelumnya. Apabila lokasi plat kendaraan berhasil terdeteksi akan dilanjutkan ke proses segmentasi plat dan apabila gagal akan kembali menginputkan kembali citra kendaraan lain.



Gambar 12. Input citra VEDA



Gambar 13. Output citra HDD

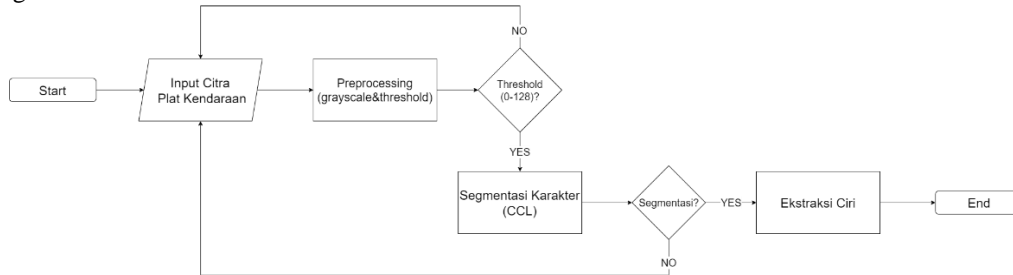


Gambar 14. Output citra median blur



Gambar 15. Output lokasi plat kendaraan

3.1.6 Segmentasi Karakter



Gambar 16. Skema proses *preprocessing* (CCL)

Pada segmentasi karakter digunakan metode *Connected Component Labelling (CCL)*, sebelum dilakukan proses pelabelan pada setiap objek karakter akan dilakukan *preprocessing* terlebih dahulu untuk menghilangkan *noise* atau objek selain karakter pada plat kendaraan. Pada tahap ini citra yang diterima adalah citra plat kendaraan yang telah berhasil dideteksi dan disegmentasi pada tahap *preprocessing* sebelumnya [13].



Gambar 17. Input citra plat kendaraan



Gambar 18. Output citra *grayscale* plat kendaraan



Gambar 19. Output citra *threshold* plat kendaraan



Gambar 20. Output citra plat kendaraan hasil *preprocessing*

Setelah citra tersebut melalui tahap *preprocessing* akan menghasilkan seperti yang ada pada Gambar 20 yaitu citra yang bebas dari *noise* atau tidak ada objek selain objek karakter pada plat kendaraan. Selanjutnya akan dilakukan proses *labelling* menggunakan *CCL* dan juga setiap karakter nya akan langsung disegmentasi [6].

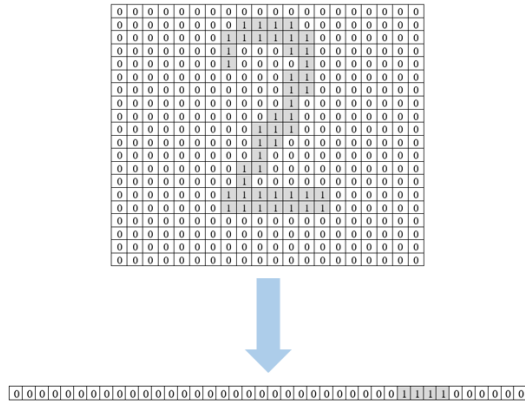


Gambar 21. Output citra segmentasi karakter menggunakan *CCL*

Apabila setiap karakter berhasil tersegmentasi dengan sempurna maka akan dilanjutkan ke proses ekstraksi ciri, namun apabila gagal tersegmentasi dengan sempurna maka akan kembali menginputkan citra plat kendaraan yang lain.

3.1.7 Ekstraksi Ciri

Pada penelitian ini ekstraksi ciri yang digunakan adalah dengan mengambil nilai informasi setiap piksel pada citra karakter sebagai fitur untuk membangun model klasifikasi. Sebelum proses ekstraksi ciri dilakukan, proses *resize* pada setiap citra karakter penting untuk dilakukan pada tahap ini.

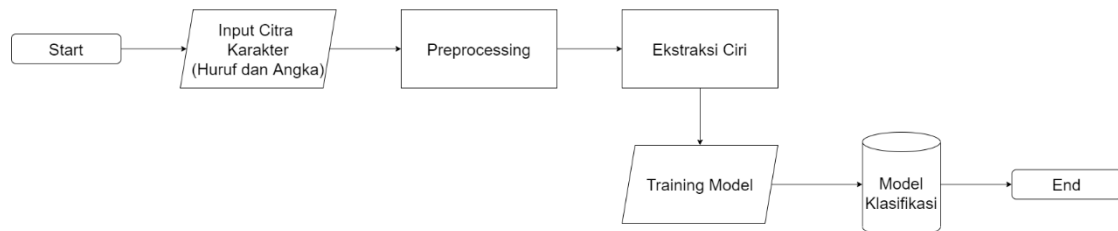


Gambar 22. Proses ekstraksi ciri

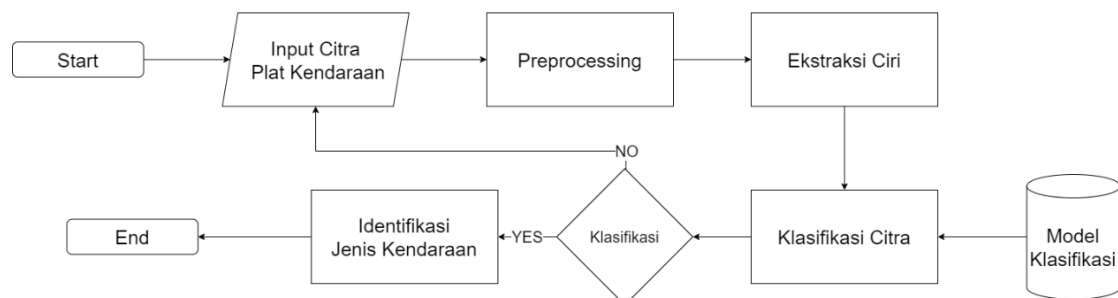
Pada Gambar 22 merupakan ilustrasi dari cara mengambil informasi nilai piksel pada citra karakter yang berukuran 20x20.

3.2 Klasifikasi

Proses klasifikasi merupakan proses untuk mengenali setiap huruf dan angka yang terdapat pada plat kendaraan. Model klasifikasi yang digunakan pada penelitian ini adalah model klasifikasi *KNN* untuk mengenali setiap huruf dan angka pada area plat kendaraan.



Gambar 23. Skema pembangunan model klasifikasi



Gambar 24. Skema proses klasifikasi

Pada Gambar 23 ditunjukkan skema dari proses pembangunan model klasifikasi dari data latih. Data latih yang dipakai merupakan citra karakter huruf dan angka, sehingga diperlukan proses *preprocessing* terlebih dahulu, proses *preprocessing* pada model klasifikasi ini yaitu *resize* semua ukuran citra karakter huruf dan angka, kemudian ekstraksi ciri dilakukan untuk mendapatkan nilai informasi dari setiap piksel pada citra. Hasil ekstraksi ciri tersebut akan disimpan dalam bentuk model klasifikasi dan akan digunakan pada proses pengujian.

Setelah model klasifikasi dibangun, maka proses klasifikasi dapat dilakukan seperti pada Gambar 24 yang merupakan skema dari sistem pengujian yang dibangun. Data citra kendaraan tersebut sudah melewati tahap *preprocessing* dan ekstraksi ciri, sehingga proses klasifikasi dapat dilakukan dengan memanfaatkan nilai *Euclidean* antara setiap citra kendaraan dengan model klasifikasi. Apabila klasifikasi berhasil dilakukan maka akan masuk ke proses identifikasi jenis kendaraan, namun apabila gagal akan kembali menginputkan citra plat kendaraan yang lain.

3.3 Pengukuran Kinerja

Untuk mengukur kinerja sistem digunakan *confusion matrix* [14]. Pada *confusion matrix*, pengukuran kinerja sistem berupa akurasi, *recall* dan *precision* dengan menggunakan 4 parameter sebagai representasi dari proses klasifikasi yaitu : *True Positive* (TP), yang merupakan data terdeteksi positif dan teridentifikasi benar, *True Negative* (TN), yang merupakan data terdeteksi positif dan teridentifikasi salah, *False Positive* (FP), yang merupakan data terdeteksi negatif dan teridentifikasi benar, *False Negative* (FN), yang merupakan data terdeteksi negatif dan teridentifikasi salah.

$$\text{Akurasi} = \frac{TP+TN}{TP+FP+FN+TN} \times 100\% \quad (3)$$

$$\text{recall} = \text{True Positive Ratio (TPR)} = \text{sensitivity} = \frac{TP}{TP+FP} \quad (4)$$

$$\text{precision} = \text{positive predictive value (PPV)} = \frac{TP}{FP+FN} \quad (5)$$

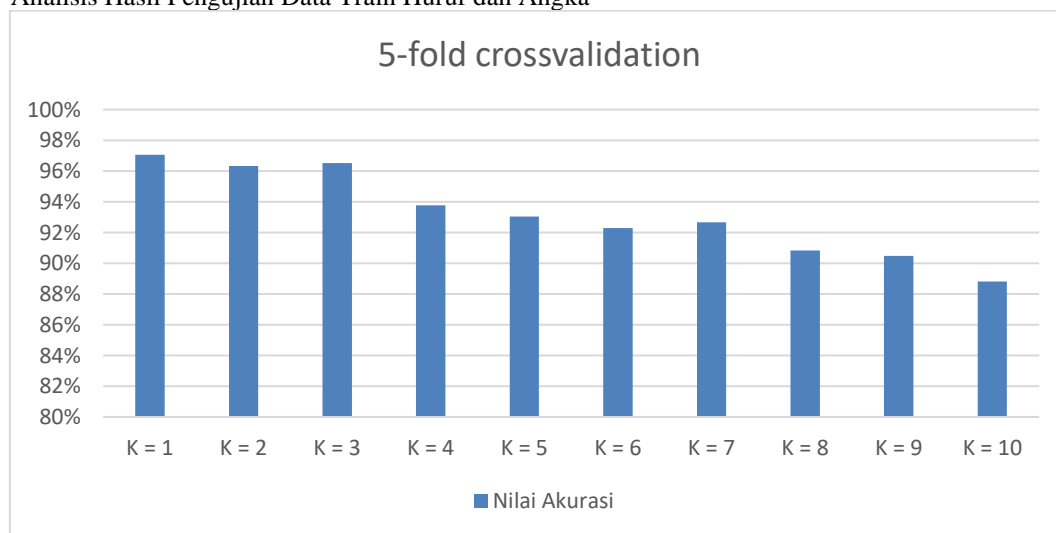
$$\text{F1 - Score} = \frac{2TP}{2TP+FP+FN} \quad (6)$$

Persamaan 3 hingga Persamaan 6 menunjukkan cara untuk menghitung nilai akurasi, *recall*, *precision* dan *F1-Score* yang dihasilkan dalam proses pengujian. Dalam penelitian ini, *recall* merupakan perbandingan antara jumlah fitur relevan yang berhasil diidentifikasi dengan jumlah fitur yang relevan pada plat nomor kendaraan. *Precision* merupakan perbandingan antara jumlah relevan fitur yang berhasil diidentifikasi dengan jumlah fitur yang relevan dan tidak relevan pada plat nomor kendaraan.

4. Evaluasi

4.1 Hasil Pengujian

4.1.1 Analisis Hasil Pengujian Data Train Huruf dan Angka



Gambar 25. Grafik Hasil Validasi Data Train Huruf dan Angka

Data train yang digunakan pada penelitian ini adalah sebanyak 2.734 citra karakter, pada Gambar 25 ditunjukkan hasil pengujian yang dilakukan pada data train yang bertujuan untuk mendapatkan nilai K optimum. Nilai K menunjukkan jarak atau nilai tetangga terdekat yang akan diambil dari *euclidean distance* antara data tes dan data train pada saat klasifikasi. Dataset yang dipakai merupakan data latih yang terdiri dari citra huruf dan angka. Pengujian dilakukan sebanyak 10 kali iterasi dengan parameter nilai *K-fold* = 5 dan parameter K yang berbeda setiap iterasinya untuk model klasifikasi *KNN*. Pada Gambar 25 dapat dilihat bahwasannya semakin besar nilai K maka akan nilai akurasi yang diperoleh akan semakin kecil. Dari 10 kali iterasi dapat disimpulkan bahwa

nilai akurasi terbaik didapat pada iterasi ke 1 dengan menggunakan parameter nilai $K = 1$ dengan tingkat akurasi yang diperoleh sebesar 97,06%.

4.1.2 Analisis Hasil Pengujian Deteksi Plat Kendaraan

Data test yang digunakan pada penelitian ini adalah sebanyak 162 citra kendaraan, dari sebanyak 162 dataset citra kendaraan yang digunakan pada tahap pengujian, sistem dapat mengenali lokasi plat kendaraan sebanyak 160 dataset citra kendaraan dengan akurasi 98,76%.



Gambar 26. Plat Kendaraan Gagal Dikenali



Gambar 27. Output Adaptive Threshold Plat Kendaraan Gagal Dikenali



Gambar 28. Plat Kendaraan Berhasil Dikenali





Gambar 29. Output Adaptive Threshold Plat Kendaraan Berhasil Dikenali

Pada Gambar 26 ditunjukkan citra *grayscale* dari salah satu citra kendaraan yang gagal dikenali lokasi plat kendaraannya. Gambar 27 merupakan hasil *adaptive threshold* dari Gambar 26, dapat dilihat bahwasannya nilai keabuan pada daerah plat kendaraan tidak sepenuhnya hitam maka ketika proses *adaptive threshold* menghasilkan plat kendaraan yang kurang sempurna untuk dideteksi hal ini biasanya disebabkan karena plat kendaraan terkena pantulan cahaya matahari atau semacamnya yang membuat daerah sekitar plat kendaraan memiliki tingkat nilai keabuan yang tinggi dan tidak berhasil dideteksi oleh sistem, berbeda dengan hasil pada Gambar 28 dan Gambar 29 yang bersih dari noise dan berhasil terdeteksi oleh sistem. Dari pengujian ini dapat disimpulkan bahwa area plat kendaraan yang berwarna selain hitam tidak dapat diproses menggunakan *adaptive threshold*.

4.1.3 Analisis Hasil Pengujian Segmentasi Karakter

Pada pengujian ini data yang dipakai merupakan hasil pengujian deteksi plat kendaraan yang berhasil dikenali oleh sistem. Dari sebanyak 160 data citra plat kendaraan yang dipakai, sistem hanya mampu melakukan segmentasi karakter sebanyak 118 data citra plat kendaraan dengan akurasi 73,75%. Dari hasil pengujian, terdapat beberapa kategori plat kendaraan yang tidak dapat dilakukan proses segmentasi yang ditampilkan pada Tabel 1.

Tabel 1. Sampel data plat kendaraan gagal pada proses segmentasi karakter

No	Plat Kendaraan	Hasil Segmentasi
1		

2		
3		
4		
5		

Pada Tabel 1 menunjukkan contoh dari beberapa kategori plat kendaraan yang gagal saat proses segmentasi karakter, hal ini disebabkan banyak faktor seperti ditunjukkan pada Tabel 1 bagian nomor 1 menunjukkan bahwa sistem tidak dapat melakukan segmentasi karakter karena karakter saling terhubung satu sama lain dengan karakter lainnya maupun dengan objek selain karakter. Pada Tabel 1 bagian nomor 2 menunjukkan bahwa sistem tidak dapat melakukan segmentasi terhadap plat yang masih menyisakan noise pada wilayah sekitar plat kendaraan. Pada Tabel 1 nomor 3 menunjukkan bahwa sistem tidak dapat melakukan segmentasi karakter pada plat yang segmentasi pada proses sebelumnya kurang sempurna. Pada Tabel 1 nomor 4 menunjukkan bahwa sistem tidak dapat melakukan segmentasi karakter pada plat yang terdapat sejenis besi yang menutupi karakter pada plat kendaraan tersebut. Pada Tabel 1 nomor 5 menunjukkan bahwa sistem tidak dapat melakukan segmentasi karakter pada plat yang kondisinya kurang baik. Pada pengujian ini proses segmentasi karakter menggunakan *connected component labelling* namun sebelum melakukan proses *labelling* diperlukan *preprocessing* berupa penghilangan objek-objek selain karakter pada plat kendaraan.

4.1.4 Analisis Hasil Pengujian Pengenalan Nomor Plat

Dari sebanyak 118 data citra plat kendaraan yang karakternya berhasil tersegmentasi sempurna, sistem mampu mengenali nomor plat sebanyak 85 dari 118 citra plat kendaraan. Pada pengujian ini semua karakter huruf dan angka diklasifikasikan berdasarkan model klasifikasi yang telah dibangun sebelumnya dan untuk proses klasifikasinya sendiri memanfaatkan nilai *Euclidean* antara karakter yang diuji dan data latih pada model klasifikasi. Dari hasil pengujian pengenalan nomor plat terdapat 85 plat nomor kendaraan yang berhasil dikenali dan 33 plat nomor kendaraan yang gagal dikenali.

Tabel 2. Rincian komponen *confusion matrix* pada citra kendaraan

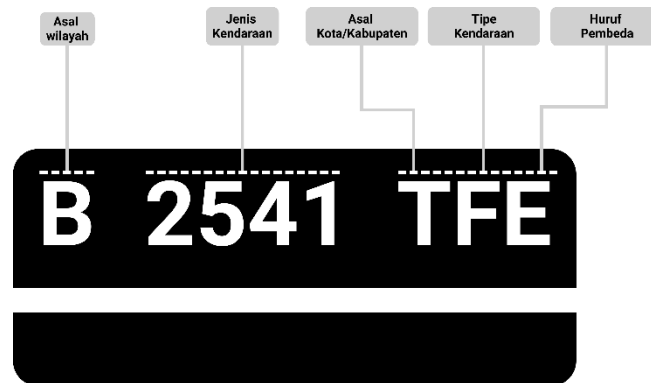
Hasil Identifikasi Karakter		Komponen <i>confusion matrix</i> pada citra karakter			
Terdeteksi	Gagal Terdeteksi	TP	FP	TN	FN
665	261	665	223	0	38

Tabel 2 menunjukkan rincian komponen *confusion matrix*, yaitu sebanyak 665 karakter yang berhasil terdeteksi dan 261 karakter yang gagal terdeteksi.

Tabel 3. Hasil pengujian pengenalan nomor plat

Akurasi	<i>Recall</i>	<i>Precision</i>	<i>F1-Score</i>
71,81%	94,59%	74,88%	83,59%

Tabel 3 menunjukkan nilai Akurasi, *recall* dan *precision* dari hasil pengujian pengenalan nomor plat dengan nilai akurasi sebesar 71,81%, nilai *recall* sebesar 94,59%, nilai *precision* sebesar 74,88% dan nilai *F1-Score* sebesar 83,59%.



Gambar 30. Penjelasan mengenai makna nomor plat kendaraan bermotor

Dari 85 citra plat kendaraan yang berhasil dikenali, kemudian sistem mengidentifikasi makna dari nomor plat kendaraan. Terdapat 5 bagian yaitu kode huruf pertama menunjukkan asal wilayah kendaraan, kode angka menunjukkan jenis kendaraan, kode huruf kedua menunjukkan asal kota/kabupaten, kode huruf ketiga menunjukkan tipe kendaraan dan kode huruf keempat menunjukkan huruf pembeda [7]. Kemudian untuk menentukan golongan di jalan tol, sistem menentukan berdasarkan kode angka dan kode huruf kedua pada nomor plat kendaraan.

Tabel 4. Dataset plat kendaraan gagal dikenali

No	Plat Kendaraan	Hasil Klasifikasi	Hasil Identifikasi Asal Daerah	Hasil Identifikasi Jenis Kendaraan	Hasil Identifikasi Asal Kota/Kabupaten	Hasil Identifikasi Tipe Kendaraan	Hasil Identifikasi Golongan Kendaraan
1	B100PIP	B100B1P	-	-	-	-	-
2	B1532RFR	B1532R6R	-	-	-	-	-
3	B1055NKU	B1Q55NKU	-	-	-	-	-
4	B9770JZ	BQ770JZ	-	-	-	-	-
5	B1337KYC	B1397KYC	-	-	-	-	-
6	B2096TFE	B2Q96TFE	-	-	-	-	-
7	B9520WT	B952BWT	-	-	-	-	-
8	B1790KZD	B1790KZB	-	-	-	-	-
9	B1564TRT	B156ATR	-	-	-	-	-
10	B2355TON	B2355TOV	-	-	-	-	-
11	B1794URQ	B1794IR0	-	-	-	-	-
12	F8109GB	B81D9GB	-	-	-	-	-
13	B1092ZFFH	B1092AFH	-	-	-	-	-
14	B2678SKX	B2678AKX	-	-	-	-	-
15	B9623SCB	B3623SCB	-	-	-	-	-
16	B9429TAT	B3429TAT	-	-	-	-	-
17	B9342PCL	B9B42PCL	-	-	-	-	-
18	B9001FUD	B9B01FUB	-	-	-	-	-
19	B2386BFO	B2B86BFO	-	-	-	-	-
20	B2792TOI	B2732TOI	-	-	-	-	-
21	B1679EKE	81679EKE	-	-	-	-	-
22	B9298WRU	B3298WRU	-	-	-	-	-
23	B8035KL	B8B35KL	-	-	-	-	-
24	B1084ZKG	B1B84ZKG	-	-	-	-	-
25	B1004TJN	B1B04TJN	-	-	-	-	-
26	B9674QH	B9644QH	-	-	-	-	-
27	B1824RFZ	B1824RFA	-	-	-	-	-
28	B2147TOQ	B2147IOQ	-	-	-	-	-
29	B1419FG	T1413FG	-	-	-	-	-
30	B848MBN	BB48MBN	-	-	-	-	-
31	B9720BDC	B972QBDC	-	-	-	-	-
32	B1341KKX	B1B41KKX	-	-	-	-	-
33	B1739PIA	B1733PIA	-	-	-	-	-

Pada Tabel 4 menunjukkan masih banyak plat kendaraan yang masih gagal dikenali nomor platnya. Pada pengujian ini banyak kesalahan pengenalan karakter terutama dikarenakan adanya karakter yang bentuknya hampir sama dengan karakter lain, kesalahan pengenalan sering terjadi pada karakter-karakter berikut :

- a. B dengan 8, P, dan D
- b. T dengan I
- c. 0 dengan Q dan B
- d. 3 dengan B dan 9
- e. 4 dengan A dan 7
- f. 9 dengan Q dan 3

5. Kesimpulan

Pada penelitian ini, telah dibangun sistem pengenalan plat nomor kendaraan untuk menentukan jenis kendaraan di jalan tol dengan menggunakan metode *Vertical Edge Detection* (VEDA) dan *Connected Component Labelling* (CCL). Berdasarkan penelitian yang telah dilakukan dapat disimpulkan bahwa pada saat pengenalan plat kendaraan menggunakan VEDA, sistem dapat mengenali plat kendaraan dengan baik, karena VEDA mampu mengenali 160 plat kendaraan dari 162 dataset citra kendaraan. Pada proses segmentasi menggunakan CCL, sistem dapat mengenali 118 data plat kendaraan dari 160 data plat kendaraan. Kemudian saat mengenali nomor plat, sistem hanya mampu mengenali sebanyak 85 plat dari 118 plat. Dengan tingkat akurasi sebesar 71,81%, nilai *recall* sebesar 94,59%, nilai *precision* sebesar 74,88% dan nilai *F1-Score* sebesar 83,59%, dilihat dari *precision* dan *recall* menunjukkan kelas terdeteksi dengan baik oleh model yang diusulkan dalam penelitian ini.

Dengan begitu dapat disimpulkan bahwa sistem secara umum menghasilkan kemampuan identifikasi dan deteksi yang baik walau masih diperlukan penyempurnaan lebih lanjut untuk mengenali plat nomor kendaraan, akan lebih baik menggunakan metode lain yang bisa menghapus noise pada plat kendaraan untuk mendapatkan performansi yang lebih baik ketika proses segmentasi karakter dan disarankan menambahkan metode tertentu untuk ekstraksi ciri agar lebih bisa mengenali setiap karakter yang bentuknya hampir sama dengan karakter lain.

Pada penelitian ini, sistem hanya mampu mendeteksi jenis kendaraan dari hasil rekaman video yang sudah dikonversi menjadi citra/frame. Diharapkan pada penelitian selanjutnya dapat ditambahkan sistem yang dapat mendeteksi jenis kendaraan dari hasil video dan juga dilakukan pengujian dengan variasi citra yang lebih banyak agar dapat mengetahui hasil dari berbagai kondisi citra.