

Deteksi Plat Nomor Kendaraan dengan Hough Transform dan Harris Corner Menggunakan Akuisisi Melalui Raspberry Pi

Detection of Vehicle Plate with Hough Transform and Harris Corner using Acquisition through Raspberry Pi

Noor Akhmad Gilar Romadhon^{1*}, Koredianto Usman², Raditiana Patmasari³
^{1,2,3}Universitas Telkom

Jl. Telekomunikasi Terusan Buah Batu Bandung 40257, Telp (022) 7566456
gilaar@student.telkomuniversity.ac.id^{1*}, korediantousman@telkomuniversity.ac.id²,
raditiana@telkomuniversity.ac.id³

Abstrak – Dalam suatu tempat, seperti di mall, tempat parkir kendaraan dapat terdiri dari beberapa lantai di basement atau di bagian atas bangunan. Di samping itu juga banyak sektor yang berubah fungsi menjadi tempat parkir. Dalam situasi ini, diperlukan sistem monitoring kendaraan yang fleksibel. Pada penelitian ini dibuat sistem yang dapat mengenali jenis kendaraan berdasarkan warna dasar pada plat nomor apakah kendaraan tersebut merupakan kendaraan pribadi, umum, atau pemerintah. Data berupa citra diambil dengan menggunakan webcam melalui akuisisi Raspberry Pi. Sistem dirancang dengan menggunakan deteksi tepi dan morfologi serta menggunakan metode Hough Transform untuk memperbaiki garis tepi dan Harris Corner untuk mendeteksi sudut pada citra. Selanjutnya dilakukan proses cropping plat. Deteksi warna dasar plat nomor dilakukan dengan menggunakan ruang warna YcbC. Selain itu, dilakukan perhitungan waktu proses akuisisi dan waktu proses keseluruhan ketika Raspberry Pi dan laptop melakukan proses deteksi. Berdasarkan hasil pengujian, sistem mendapatkan hasil terbaik sebesar 100% saat deteksi plat nomor kendaraan pribadi dan hasil terburuk sebesar 70% saat deteksi plat nomor kendaraan umum. Rata-rata akurasi yang didapatkan pada sistem ini sebesar 88,9%. Akurasi tersebut diperoleh dengan 90 data uji. Rata-rata waktu komputasi adalah 0,99 detik, yang artinya proses pendeteksian dilakukan cukup cepat.

Kata Kunci: Plat Nomor, Hough Transform, Harris Corner, YCbCr, Raspberry Pi.

Abstract – In some place, such as in a mall, a vehicle parking lot can consist of several floors in the basement or at the top of a building. In addition, many sectors have also been turned into parking lots. In this situation, a flexible vehicle monitoring system is needed to detect the type of vehicle ownership through the base color on the license plate connected to the Raspberry Pi, which is expected to be able to mitigate this problem. This research develop a system that can recognize the type of vehicle based on the color of the plate number whether the vehicle is a private, public, or government vehicle. The data consists of images taken using a webcam through the acquisition of Raspberry Pi. The system was designed by using edge detection, morphology, the Hough Transform method to correct the edge, and Harris Corner to detect angles in the image, after which the plate cropping process is carried out. The license plate base color is

detected using the *YCbCr* color space, and the system will also measure the quality of the network between the *Raspberry Pi* and the laptop used during the detection process. The system gets the best results at 100% when detecting private vehicle license plates and the worst results at 70% when detecting public vehicle license plates. The average accuracy obtained in this system is 88,9% that used 90 test data. The average computation time is 0,99 seconds, which means that the detection process is quite fast.

Keywords: License Plate, Hough Transform, Harris Corner, *YCbCr*, *Raspberry Pi*.

1. Pendahuluan

Perkembangan jumlah kendaraan bermotor baik sepeda motor atau mobil mengalami peningkatan begitu cepat sehingga dapat menyebabkan pertumbuhan tempat pengelolaan kendaraan serta tempat parkir. Dalam suatu tempat, seperti di mall, tempat parkir kendaraan dapat terdiri dari beberapa lantai di *basement* atau di bagian atas bangunan. Di samping itu juga banyak sektor yang berubah fungsi menjadi tempat parkir. Dalam situasi ini, sistem monitoring kendaraan yang fleksibel diperlukan. Sistem tersebut adalah deteksi jenis kepemilikan kendaraan melalui warna dasar pada plat nomor yang terhubung dengan *Raspberry Pi* dengan harapan sistem dapat menyelesaikan permasalahan tersebut.

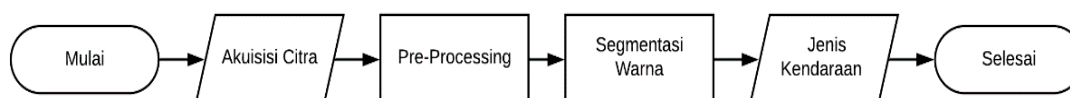
Sistem dibantu *webcam* untuk mengambil gambar berupa plat nomor pada kendaraan. Alasan menggunakan kamera untuk mendeteksi plat nomor adalah karena lebih efektif dibanding pembacaan manual oleh manusia yang mungkin dapat menimbulkan kesalahan. Setelah *webcam* mengambil gambar maka akan dilakukan proses pengolahan citra agar dapat mengenali warna dasar pada plat nomor. Proses tersebut akan dilakukan oleh *software* yang telah ditentukan dengan metode yang sesuai kebutuhan.

Pada penelitian sebelumnya [1], digunakan metode morfologi untuk membaca warna plat nomor agar dapat mendeteksi jenis kendaraan. Hasil akurasi yang didapatkan sebesar 83% dan kekurangan penelitian ini tidak dapat mengenali warna pada plat hasil *cropping* morfologi yang kurang maksimal. Penelitian lain [2] menggunakan ruang warna L^*a^*b dengan mengambil *b channel* agar dapat mendeteksi plat kendaraan warna biru dan kuning dengan mendapatkan hasil akurasi sebesar 98%. Tetapi, metode ini tidak dapat digunakan untuk mendeteksi plat kendaraan warna hitam dan merah. Pada penelitian [3], digunakan metode *Hough Transform* dan *Harris Corner* untuk proses deteksi bentuk objek 2 dimensi seperti lingkaran, segitiga, persegi dan persegi panjang. Didapatkan hasil akurasi diatas 97% dan tingkat presisi objek sebesar 100%. Penggunaan metode *Harris Corner* akan menghasilkan nilai yang konsisten dari suatu citra walaupun telah mengalami rotasi, penskalaan, variasi pencahayaan, ataupun memiliki noise seperti pada penelitian sebelumnya [4]. Kesimpulannya bahwa metode *Hough Transform* dan *Harris Corner* tersebut layak untuk digunakan dalam proses *cropping* bagian plat nomor kendaraan.

Pada penelitian ini dirancang suatu sistem yang dapat mendeteksi plat nomor kendaraan untuk mengenali kepemilikan jenis kendaraan dengan menggunakan metode *Hough Transform* dan *Harris Corner* sebagai proses deteksi dan *cropping* plat nomor. Proses deteksi warna pada plat nomor tersebut menggunakan ruang warna *YCbCr* sehingga sistem dapat mengenali kendaraan tersebut kendaraan pribadi, kendaraan umum, atau kendaraan pemerintah. *Raspberry Pi* digunakan untuk memudahkan dalam proses akuisisi citra.

2. Metode Penelitian

Tahapan dari penelitian ini ditunjukkan pada Gambar 1.



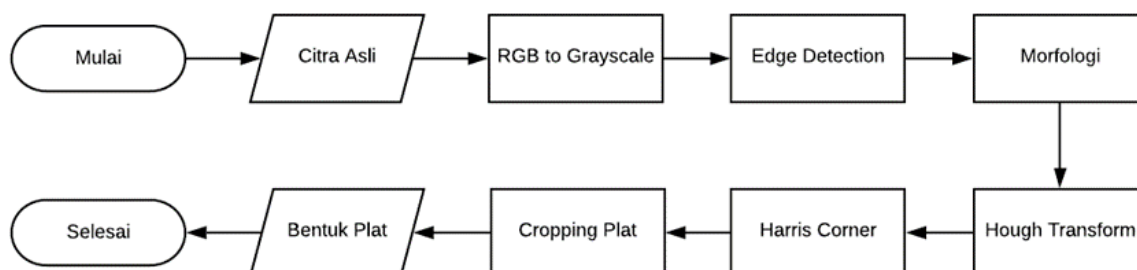
Gambar 1. Flowchart penelitian.

Webcam akan mengambil citra berupa kendaraan yang memiliki plat sesuai standar yang telah ditentukan dari arah depan kendaraan tersebut dengan jarak yang telah ditentukan untuk menjadi *input* sistem. Setelah akuisisi selesai, selanjutnya citra akan diproses sampai terdeteksi jenis kendaraannya.

2.1. Diagram Alir Sistem

2.1.1. Pre-Processing

Diagram alir *pre-processing* dapat dilihat pada Gambar 2. Proses ini dilakukan untuk mendapatkan keseragaman citra agar mudah untuk proses selanjutnya.



Gambar 2. Diagram alir *pre-processing*.

Proses ini dilakukan setelah melakukan proses akuisisi citra oleh sistem. Proses ini bertujuan untuk mendapatkan bentuk dari plat nomor pada citra tersebut. Gambar 3 merupakan contoh citra hasil dari akuisisi tersebut.



Gambar 3. Citra asli.

a. *RGB to Grayscale*

Citra asli atau citra inputan akan dilakukan proses perubahan citra *RGB* ke citra keabuan. Bertujuan untuk memudahkan proses selanjutnya yaitu deteksi tepi citra.

b. *Edge Detection*

Pada langkah ini citra yang sudah diubah ke dalam citra keabuan akan dilakukan proses pendeteksian garis tepi yang terdapat objek-objek yang ada pada citra tersebut. Jenis deteksi tepi yang digunakan pada sistem ini adalah *Canny Edge*. Pada proses ini nilai citra akan menjadi biner yaitu 0 atau 1.

c. *Morfologi*

Proses ini akan dilakukan untuk menghilangkan garis-garis *noise* atau objek yang bukan bagian dari plat kendaraan tersebut sehingga keluaran dari proses ini adalah hanya bentuk plat kendaraan tersebut. Proses morfologi dilakukan 4 langkah yaitu *bwareaopen*, dilasi, *filling*, dan erosi.

d. *Hough Transform*

Transformasi *Hough* adalah suatu teknik atau metode yang digunakan untuk mendapatkan ciri atau fitur dari suatu gambar [5], dimana metode ini dapat digunakan untuk memperbaiki kualitas objek dari citra sebelumnya yang bekerja dengan menggunakan *voting* suara terbanyak untuk menentukan nilai parameter yang tepat [6].

Transformasi *Hough* memiliki beberapa perbedaan rumus yang diterapkan. Semuanya tergantung pada jenis objek yang dicari, misalnya untuk mencari objek garis akan digunakan fungsi garis seperti berikut ini [7]

$$x \cos \theta + y \sin \theta = r. \quad (1)$$

Hough Transform dilakukan 2 kali, pertama untuk memperbaiki garis tepi objek agar terlihat lurus dan mudah untuk proses deteksi sudut selanjutnya, kedua untuk menghilangkan objek yang menempel pada objek plat tersebut. Hasil dari proses *Hough Transform* ditunjukkan pada Gambar 4.



Gambar 4. Hasil deteksi *Hough Transform*.

e. *Harris Corner*

Metode ini digunakan untuk mendeteksi sudut pada citra, dimana pada sistem ini metode *Harris Corner* bertujuan untuk mendeteksi sudut pada plat nomor kendaraan yang akan menandakan bahwa plat tersebut berbentuk persegi panjang yang sempurna dengan adanya sudut-sudut tersebut. Metode ini akan menghasilkan nilai yang konsisten dari suatu citra walaupun telah mengalami rotasi, penskalaan, variasi pencahayaan, ataupun memiliki *noise* [4]. Untuk melakukan proses ini dilakukan langkah-langkah sebagai berikut:

1) Konversi citra biner ke *grayscale*

Hasil keluaran *Hough Transform* berupa citra biner, maka dari itu harus diubah ke dalam citra *grayscale* dengan persamaan berikut

$$grayscale = 255 * citra \text{ biner}. \quad (2)$$

2) Menghitung gradien citra

Menghitung nilai gradien horizontal dan vertikal citra dengan melakukan konvolusi citra *grayscale* dengan kernel *Prewitt*. Persamaanya adalah sebagai berikut:

$$I_x = grayscale \otimes \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}, \quad (3)$$

$$I_y = grayscale \otimes \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}. \quad (4)$$

3) Menghitung nilai turunan

Setelah mendapatkan nilai gradien, maka selanjutnya adalah menghitung nilai turunan setiap piksel dengan persamaan sebagai berikut:

$$I_x^2 = I_x * I_x, \quad (5)$$

$$I_y^2 = I_y * I_y, \quad (6)$$

$$I_{xy} = I_x * I_y. \quad (7)$$

4) Konvolusi dengan filter *Gaussian*

Melakukan konvolusi antara matriks turunan dengan matriks *gaussian filter* untuk mencari jumlah nilai turunan produk tiap pikselnya dengan persamaan dan *gaussian filter* sebagai berikut:

$$Gaussian = \frac{1}{273} \begin{bmatrix} 1 & 4 & 7 & 4 & 1 \\ 4 & 16 & 26 & 16 & 4 \\ 7 & 26 & 41 & 26 & 7 \\ 4 & 16 & 26 & 16 & 4 \\ 1 & 4 & 7 & 4 & 1 \end{bmatrix}, \quad (8)$$

$$S_x^2 = I_x^2 \otimes G_\sigma, \quad (9)$$

$$S_y^2 = I_y^2 \otimes G_\sigma, \quad (10)$$

$$S_{xy} = I_{xy} \otimes G_\sigma. \quad (11)$$

5) Menghitung respons *Harris*

Proses ini dilakukan untuk menentukan piksel tersebut suatu sudut atau bukan. Apabila nilai respon lebih besar dari nilai *threshold* yang telah ditentukan maka piksel tersebut akan dianggap sebagai sudut, tetapi harus melihat nilai *local maxima* terlebih dahulu. Persamaan untuk menghitung respon *Harris* sebagai berikut

$$Respon\ Harris = ((S_x^2) \cdot (S_y^2) - (S_{xy}^2)) - 0.04((S_x^2) + (S_y^2))^2. \quad (12)$$

6) Menghitung nilai maksimum

Nilai maksimum atau *local maxima* (M) diambil dari nilai terbesar pada respon *Harris* (R) yang didapatkan dan dibuat matriks sesuai hasil respon *Harris*, lalu carilah nilai M dan R yang sama dan memiliki nilai lebih besar dari *threshold* yang ditentukan pada sistem.

7) Deteksi sudut

Dari perhitungan sebelumnya yang memiliki nilai yang sama maka akan diasumsikan menjadi 1 (sudut) dan lainnya menjadi 0 (bukan sudut) untuk dijadikan ke dalam matriks baru [3]. Contoh matriksnya sebagai berikut

$$Harris\ Point = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \quad (13)$$

Hasil dari proses *Harris Corner* ditunjukkan pada Gambar 5.



Gambar 5. Hasil deteksi sudut *Harris Corner*.

f. *Cropping Plat*

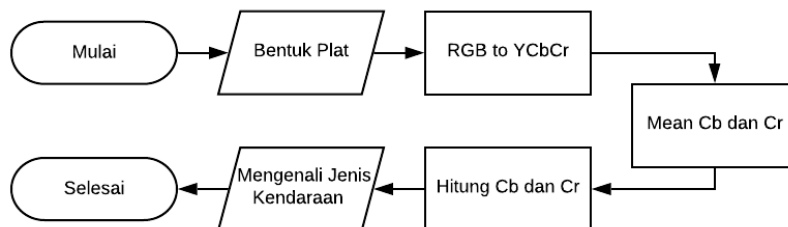
Selanjutnya adalah melakukan proses *cropping* plat dengan mengambil keluaran dari proses *Hough Transform* yang dikonvolusi dengan citra asli. Gambar 6. adalah hasil dari proses *cropping* tersebut.



Gambar 6. Hasil *cropping* plat.

2.1.2. Proses Segmentasi

Diagram alir proses segmentasi dapat dilihat pada Gambar 7.



Gambar 7. Diagram alir segmentasi.

Proses ini dilakukan setelah mendapatkan citra berupa plat nomor yang telah di-*cropping* pada proses *pre-processing*. Proses segmentasi dilakukan untuk mendeteksi warna pada citra sehingga sistem dapat mengenali jenis kendaraan berdasarkan warna yang didapatkan.

a. *RGB to YCbCr*

Sebelumnya bentuk plat sudah didapat pada proses *pre-processing* dan masih dalam ruang warna *RGB*. Sistem ini menggunakan ruang warna *YCbCr* untuk mendeteksi warna pada plat tersebut karena lebih mudah untuk membatasi atau men-*threshold* nilai warna pada citra bentuk plat tersebut. Ruang warna *YCbCr* terdiri dari dua komponen, yaitu *Y* sebagai komponen *luminance* yang digunakan mempresentasikan warna *RGB* sedangkan *Cb* dan *Cr* sebagai komponen *chrominance* yang digunakan untuk mempresentasikan corak warna dan saturasi [8]. Warna *YCbCr* dapat diperoleh menggunakan warna *RGB* dengan persamaan sebagai berikut

$$\begin{pmatrix} Y \\ C_B \\ C_R \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.168 & -0.331 & 0.5 \\ 0.5 & -0.418 & -0.081 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{14}$$

Hasil proses perubahan ke ruang warna *YCbCr* ditunjukkan pada Gambar 8.



Gambar 8. Hasil *RGB to YCbCr*.

b. *Mean Cb dan Cr*

Pada proses ini citra yang sudah diubah warnanya akan dilakukan rata-rata pada komponen nilai *Cb* dan *Cr*. Dalam proses ini rata-rata dilakukan 2 kali, yaitu yang pertama untuk mengambil nilai rata-rata dari masing-masing kolom pada matriks dan yang kedua untuk mengambil rata-rata nilai dari seluruh kolom tersebut. Nilai dari *Cb* dan *Cr* yang diperoleh tersebut merupakan bilangan *real*.

$$\text{Nilai } C_b = \begin{pmatrix} 131 & 131 & 131 & 133 & 131 & 133 \\ 132 & 131 & 133 & 133 & 133 & 133 \\ 129 & 132 & 132 & 132 & 132 & 132 \\ 130 & 131 & 131 & 131 & 131 & 131 \\ 129 & 128 & 129 & 129 & 129 & 129 \\ 129 & 129 & 129 & 129 & 129 & 129 \end{pmatrix} = (130 \ 131 \ 131 \ 131 \ 131 \ 131) = 131 \tag{15}$$

$$\text{Nilai Cr} = \begin{pmatrix} 123 & 122 & 122 & 122 & 122 & 122 \\ 124 & 123 & 123 & 123 & 123 & 123 \\ 125 & 124 & 124 & 124 & 124 & 124 \\ 127 & 126 & 127 & 127 & 127 & 127 \\ 127 & 127 & 126 & 126 & 126 & 126 \\ 127 & 127 & 126 & 127 & 126 & 127 \end{pmatrix} = (125 \ 124 \ 124 \ 125 \ 124 \ 125) = 124 \quad (16)$$

c. Hitung Cb dan Cr

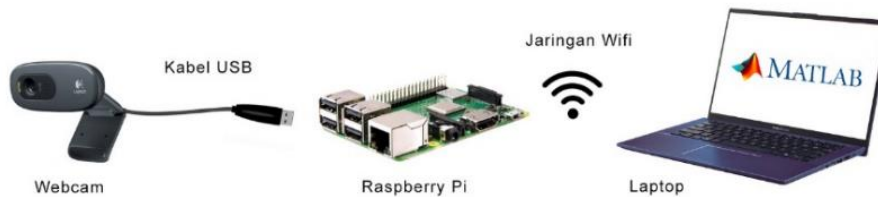
Pada proses sebelumnya didapat nilai Cb = 131 dan Cr = 124, maka dari nilai tersebut dapat diketahui bahwa nilai tersebut berada pada rentang kondisi nilai Cb dan Cr warna hitam, sehingga plat tersebut dapat dikenali oleh sistem sebagai kendaraan pribadi. Proses ini dilakukan pada kondisi rentang nilai Cb dan Cr yang telah dibuat sesuai perhitungan sistem yang ditunjukkan pada Tabel 1.

Tabel 1. Rentang nilai Cb dan Cr.

Nilai	Warna Kuning / Kendaraan Umum	Warna Merah / Kendaraan Pemerintah	Warna Hitam / Kendaraan Pribadi
Cb	50 – 105	105 – 125	115 – 140
Cr	130 – 150	135 – 200	115 – 135

2.2. Desain Perangkat Keras

Desain perangkat keras ditunjukkan pada Gambar 9. menjelaskan perangkat keras yang akan digunakan.



Gambar 9. Desain perangkat keras.

Terlihat pada gambar di atas bahwa media komunikasi yang digunakan ada 2, yaitu kabel USB untuk menghubungkan antara webcam dengan Raspberry Pi dan jaringan wifi untuk menghubungkan antara Raspberry Pi dengan laptop yang berfungsi sebagai pengolah citra menggunakan software Matlab R2018a. Komunikasi pada jaringan wifi ini akan diukur kualitasnya saat proses pendeteksian plat nomor kendaraan.

Untuk spesifikasi dari perangkat Raspberry Pi dan laptop yang digunakan dapat ditunjukkan pada Tabel 2.

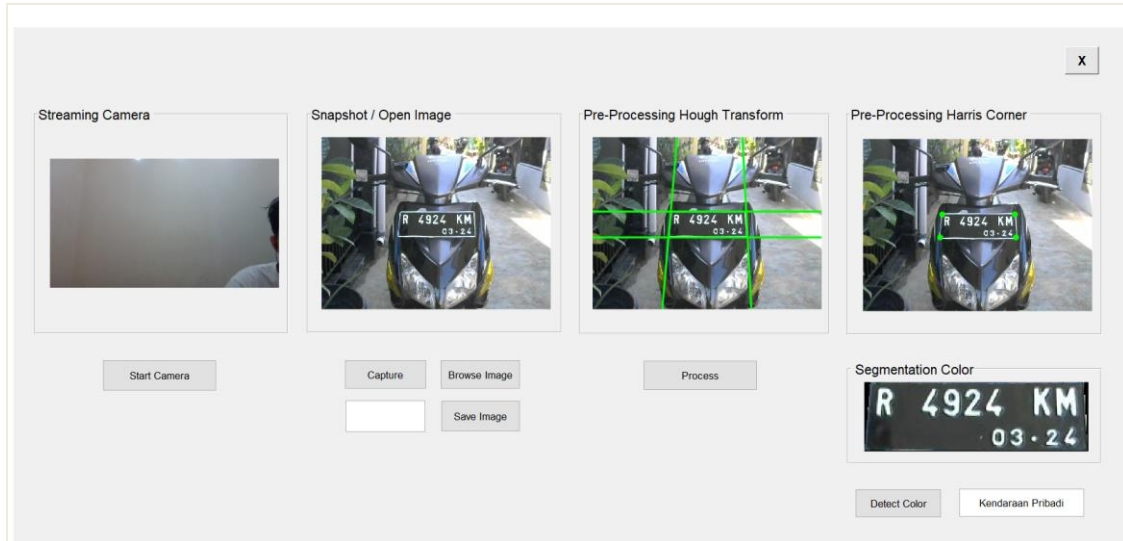
Tabel 2. Spesifikasi perangkat keras.

Raspberry Pi 3 model B	Laptop Asus a412da
CPU: 1.2 GHZ quad-core ARM Cortex A53	Prosesor: AMD Ryzen 3 3200U dual-core 2,5GHz - 3,5GHz
GPU: Broadcom VideoCore IV @ 400 MHz	Grafis: AMD Radeon RX Vega 3 iGPU
Memori: 1 GB LPDDR2-900 SDRAM	Memori: RAM 4GB DDR4-2400MHz
Network: 10/100 MBPS Ethernet, 802.11n Wireless LAN, Bluetooth 4.0	Network: Intel® Dual Band Wireless-AC 8265
SoC: Broadcom BCM2837	Sistem Operasi: Microsoft Windows

3. Hasil dan Pembahasan

3.1. Tampilan GUI

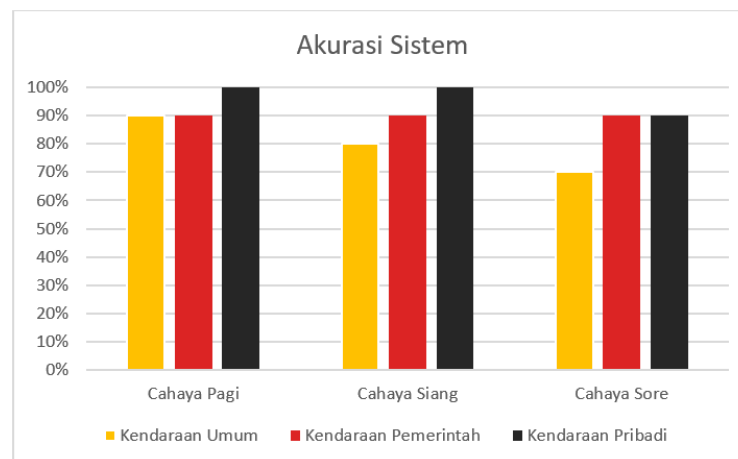
Sistem ini dibuat dengan memiliki tampilan *Graphic User Interface (GUI)* yang telah dirancang seperti pada Gambar 10.



Gambar 10. Tampilan GUI.

3.2. Hasil Pengujian Akurasi

Dari pengujian yang telah dilakukan, maka didapatkan hasil akurasi yang ditunjukkan pada Gambar 11. yang merupakan grafik perbandingan antara masing – masing plat nomor dan skenario pengujiannya. Pengujian dilakukan pada masing-masing skenario sebanyak 10 sampel.



Gambar 11. Hasil pengujian sistem.

Grafik diatas menunjukkan bahwa plat nomor kendaraan pribadi mendapatkan hasil akurasi terbesar yaitu 100% dan plat nomor kendaraan umum mendapatkan hasil akurasi terendah yaitu 70%. Dapat disimpulkan bahwa kondisi cahaya mempengaruhi hasil pengujian akurasi. Dari keseluruhan total pengujian akurasi diatas maka didapatkan rata-rata akurasi sistem ini sebesar 88,9%.

Pada penelitian ini dilakukan proses *Hough Transform* sebanyak 2 kali yang bertujuan agar mendapatkan bentuk plat yang sempurna. Gambar 12 menunjukkan perubahan bentuk plat yang diproses menggunakan metode *Hough Transform*.



Gambar 12. Hasil sebelum dan sesudah *Hough Transform*.

Dilihat dari Gambar 12, sebelum dilakukan metode *Hough Transform*, bentuk plat masih memiliki objek lain atau *noise* yang menempel pada plat tersebut dan setelah dilakukan *Hough Transform* maka *noise* tersebut hilang sehingga bentuk plat dapat terdeteksi sempurna.



Gambar 13. Hasil deteksi sudut *Harris Corner*.

Gambar 13 sebelah kiri menunjukkan deteksi sudut sebelum dilakukan proses *Hough Transform* sehingga ada objek lain yang terdeteksi sudutnya sehingga hasilnya kurang sempurna. Gambar sebelah kanan telah dilakukan proses *Hough Transform*, objek memiliki 4 sudut yang terdeteksi yang artinya bahwa bentuk plat terlihat sempurna seperti persegi panjang pada umumnya. Kesimpulannya bahwa proses *Hough Transform* sebelumnya mempengaruhi hasil dari proses *Harris Corner* dan kondisi plat nomor mempengaruhi hasil pengujian akurasi, karena kondisi plat nomor yang masih dalam keadaan belum rusak akan mudah untuk terdeteksi oleh sistem.

3.3. Hasil Pengujian Waktu Komputasi

Pengujian ini dilakukan melihat seberapa lama waktu proses yang dilakukan oleh sistem ketika melakukan proses pendeteksian plat nomor kendaraan dari proses deteksi tepi, morfologi, *Hough Transform*, *Harris Corner*, dan segmentasi warna. Parameter ini dilakukan untuk melihat bahwa sistem memiliki kinerja yang baik atau tidak. Pengujian waktu komputasi dilakukan menggunakan laptop dengan spesifikasi seperti tercantum pada Tabel 2. Hasil pengujian waktu komputasi ditunjukkan pada Tabel 3.

Tabel 3. Hasil pengujian waktu komputasi.

Jenis Plat Nomor	Cahaya			Rata – rata waktu komputasi
	Pagi	Siang	Sore	
Plat Kuning	0,88 s	0,83 s	0,53 s	0,74 s
Plat Merah	0,93 s	1,61 s	1,79 s	1,44 s
Plat Hitam	0,70 s	0,88 s	0,84 s	0,80 s
Rata – rata waktu komputasi keseluruhan				0,99 s

3.4. Hasil Pengujian Komunikasi *Raspberry Pi*

Pengujian ini dilakukan untuk mengetahui kinerja dari komunikasi *Raspberry Pi* dengan sistem, dimana komunikasi dilakukan dengan jaringan *wireless mobile hotspot* dari laptop. Pengujian ini bertujuan untuk mengetahui seberapa lama waktu akuisisi citra dan waktu lamanya

pemrosesan sistem secara keseluruhan. Untuk lebih jelasnya bisa dilihat hasil pengujian pada Tabel 3 dibawah ini.

Tabel 3. Pengujian komunikasi *Raspberry Pi*.

No	Jarak <i>Raspberry Pi</i> dengan Laptop	Tanpa Penghalang		Dengan Penghalang (tembok)	
		Waktu Akuisisi (detik)	Proses Keseluruhan (detik)	Waktu Akuisisi (detik)	Proses Keseluruhan (detik)
1	Jarak 1 meter	0,34	0,94	0,44	1,25
2		0,37	0,95	0,37	1,10
3		0,39	0,97	0,38	1,38
1	Jarak 3 meter	0,35	0,96	0,40	1,17
2		0,35	1,16	0,43	1,20
3		0,39	1,02	0,51	1,36
1	Jarak 5 meter	0,36	1,26	0,85	1,75
2		0,44	1,15	0,89	1,57
3		0,36	1,19	1,05	2,10
1	Jarak 10 meter	0,38	1,44	1,87	2,39
2		0,36	1,19	1,51	1,75
3		0,35	1,02	1,52	2,25

Dari tabel diatas dapat disimpulkan bahwa semakin jauh jaraknya ketika tanpa adanya penghalang seperti tembok maka waktu akuisisi masih tetap sekitar 0,3 – 0,4 detik dan lamanya proses deteksi keseluruhan sekitar 0,9 – 1,4 detik. Sedangkan ketika diberikan penghalang seperti tembok, waktu akuisisi sedikit bertambah lama ketika jarak 5 meter dan 10 meter walaupun masih di sekitar 1 detik dan lamanya proses deteksi keseluruhan sekitar 1 – 1,4 detik ketika jarak 1 meter dan 3 meter, lalu sekitar 1,5 – 2 detik ketika jarak 5 meter dan sekitar 1,7 – 2,3 detik ketika jarak 10 meter. Hal itu terjadi dikarenakan jarak komunikasi *Raspberry Pi* ke jaringan hotspot semakin jauh dan adanya penghalang seperti tembok saat melakukan komunikasi.

4. Kesimpulan

Berdasarkan dari hasil pengujian dan analisis deteksi jenis plat nomor kendaraan dengan metode *Hough Transform* dan *Harris Corner*, didapat akurasi sistem tertinggi sebesar 100 % pada saat mendeteksi plat kendaraan pribadi ketika kondisi cahaya pagi hari dan siang hari, dan akurasi sistem terendah sebesar 70 % pada saat mendeteksi plat kendaraan umum ketika kondisi cahaya sore hari. Rata-rata akurasi sistem adalah 88,9 %. Sementara itu, rata-rata waktu komputasi sistem adalah sebesar 0,99 detik. Proses *Harris Corner* akan maksimal dan bergantung pada keluaran proses *Hough Transform*. Berdasarkan pengujian, dapat disimpulkan pula bahwa kondisi cahaya dan kondisi plat nomor berpengaruh terhadap hasil deteksi. Selain itu, komunikasi jaringan antara *Raspberry Pi* dengan laptop dipengaruhi oleh jarak dan penghalang seperti tembok.

Referensi

- [1] D. A. Priandini, J. Nangi, M. Muchtar, and J. Y. Sari, "Deteksi area plat mobil menggunakan operasi morfologi citra," *Semin. Nas. Teknol. Terap. Berbas. Kearifan Lokal*, pp. 294–302, 2018.
- [2] L. Xu, "A new method for license plate detection based on color and edge information of Lab space," *Proc. - 2011 Int. Conf. Multimed. Signal Process. C. 2011*, vol. 1, pp. 99–102, 2011, doi: 10.1109/CMSP.2011.26.
- [3] R. Ginting, R. Patmasari, and S. Aulia, "Sistem Orientasi Objek Dengan Metode Stereo Vision Berbasis Raspberry Pi," *It J. Res. Dev.*, vol. 4, no. 1, pp. 72–85, 2019, doi:

- 10.25299/itjrd.2019.vol4(1).3562.
- [4] J. Sánchez, N. Monzón, and A. Salgado, “An analysis and implementation of the harris corner detector,” *Image Process. Line*, vol. 8, pp. 305–328, 2018, doi: 10.5201/ipol.2018.229.
 - [5] R. O. Duda and P. E. Hart, “Use of the Hough Transformation to Detect Lines and Curves in Pictures,” *Commun. ACM*, vol. 15, no. 1, pp. 11–15, 1972, doi: 10.1145/361237.361242.
 - [6] H. Sa’diyah, R. R. Isnanto, and A. Hidayanto, “Aplikasi Transformasi Hough Untuk Deteksi Garis Lurus,” pp. 1–7.
 - [7] S. Muharom, “Penerapan Metode Hough Line Transform Untuk Mendeteksi Pintu Ruangan Menggunakan Kamera,” *J. IPTEK*, vol. 21, no. 1, p. 79, 2017, doi: 10.31284/j.ipitek.2017.v21i1.108.
 - [8] E. R. Swedia and M. Cahyanti, “Algoritma Transformasi Ruang Warna,” *Depok Univ. Gunadarma*, 2010.