

DETEKSI RAMBU-RAMBU LALU LINTAS UNTUK MEMBANTU PENGGUNA JALAN RAYA DENGAN ALGORITMA CAMSHIFT

DETECTION OF TRAFFIC SIGNS TO HELP THE HIGHWAY USERS WITH CAMSHIFT ALGORITHM

Kurnia Khafidhatur Rafiah¹, Rita Magdalena, Ir., M.T.², Nur Andini, S.T., M.T.³
^{1,2,3}Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

¹kurniakhafidhatur.student@telkomuniversity.ac.id, ²ritamagdalen@telkomuniveristy.ac.id,
³nur_andini@telkomuniversity.ac.id

Abstrak

Rambu-rambu lalu lintas merupakan salah satu bagian penting yang menjadi media pemberi informasi kepada masyarakat sebagai pengguna jalan raya. Rambu lalu lintas dapat berupa simbol-simbol khusus yang memiliki makna tertentu. Beberapa hal membuktikan tidak banyak masyarakat yang benar-benar memahami maksud dari setiap rambu-rambu lalu lintas yang ada. Ketidapahaman akan makna dari rambu lalu lintas menyebabkan banyak terjadi pelanggaran lalu lintas yang berujung pada kecelakaan. Guna membantu mengatasi permasalahan tersebut dikembangkan sebuah *prototype* sistem yang mampu melakukan deteksi rambu-rambu lalu lintas dan secara otomatis memberi informasi mengenai arti setiap rambu yang sering ditemui di jalan raya.

Pada tugas akhir ini, proses deteksi rambu-rambu lalu lintas menggunakan algoritma *CamShift* dan pengenalan rambu-rambu yang telah dilacak dengan fitur yang diperoleh dari *Gabor Filter*. Terdapat database tentang simbol-simbol rambu lalu lintas yang kemudian akan diidentifikasi kesesuaiannya dengan gambar pada video yang terekam dengan menggunakan metode KNN. Sistem yang dibuat menggunakan gambar bergerak sebagai masukannya, yaitu berupa file video (.avi). Keluaran dari sistem ini adalah berupa teks yang menuliskan nama rambu hasil deteksi.

Dari pengujian yang dilakukan, sistem yang dirancang menghasilkan akurasi rata-rata 77.78% dengan waktu komputasi rata-rata sebesar 12.7 sekon. Sistem masih belum dikatakan sempurna karena masih terdapat akurasi yang rendah pada waktu dan kondisi tertentu.

Kata kunci: rambu lalu lintas, algoritma *camshift*, *gabor filter*.

Abstract

Traffic signs is one of the important part that becomes a media of information to the public as users of the highway. Traffic signs may include special symbols that have specific meaning. However, not many people really understand the purpose of each traffic signs are there. This causes a lot of traffic violations that led to the accident. To solve these problems author developed a prototype system that is able to detect traffic signs and automatically gives information about the meaning of each sign are often encountered on the highway.

In this thesis, the detection of traffic signs using *CamShift* algorithms and recognition of signs that have been tracked with features derived from *Gabor Filter*. There is a database of symbols traffic signs which will then be identified for compliance with the video images recorded using KNN method. The system is made using moving images as input, in the form of video files (.avi). Output of this system is a form of text which bears the name beacon detection results.

From the tests, the system designed was produces an average accuracy about 77.78% with an average computation time of 12.7 second. The system is still not perfect because there are said to be low accuracy at the time and under certain conditions.

Keywords : *Traffic signs, camshift algorithm, gabor filter.*

1. Pendahuluan

Rambu-rambu lalu lintas merupakan salah satu dari perlengkapan jalan raya berupa peringatan, larangan, perintah, atau petunjuk jalan yang memiliki peranan yang sangat penting dalam membantu pengguna jalan raya. Rambu-rambu lalu lintas dapat berupa huruf, angka, dan atau perpaduan antar keduanya. Perkembangan teknologi kendaraan semakin canggih dimana teknologi disisipkan pada sistem kendaraan melalui sistem komputer. Dengan mengenali rambu-rambu lalu lintas diharapkan kendaraan bisa memberikan informasi kepada pengemudi secara

otomatis dan real time sehingga diharapkan dapat mengurangi angka kecelakaan lalu lintas yang disebabkan oleh pelanggaran peraturan lalu lintas yang disampaikan melalui rambu-rambu lalu lintas. Posisi rambu lalu lintas yang berbeda-beda sangat mempengaruhi proses identifikasi objek.

Penjejakan objek (*tracking object*) merupakan salah satu hal terpenting yang perlu diperhatikan dalam pengembangan suatu sistem deteksi pada gambar bergerak (video) baik *realtime* maupun *non-realtime*. Penjejakan objek menentukan kualitas dari suatu sistem deteksi pada gambar bergerak. Oleh karena itu, semakin baik algoritma penjejakan yang digunakan maka semakin baik pula kualitas sistem deteksi tersebut. Terdapat cukup banyak metode penjejakan objek yang berkembang saat ini. Salah satu diantaranya yaitu Algoritma Camshift (*Continuously Adaptive Mean-Shift*). Pada penelitian sebelumnya mengenai Algoritma Camshift menyatakan bahwa algoritma ini memiliki kualitas penjejakan yang cukup baik pada berbagai objek berdasarkan warna dasar dari objek tersebut^[1]. Penelitian sebelumnya melakukan penjejakan terhadap objek berupa cangkir berwarna merah, topi berwarna kuning, serta wajah manusia^[1]. Sementara itu, untuk penelitian mengenai deteksi rambu lalu lintas sebelumnya dilakukan pada file citra dimana klasifikasinya menggunakan metode *template matching*. Selain metode *template matching*, terdapat pula beberapa metode lainnya yang digunakan untuk penelitian yang serupa dengan file masukan berupa gambar statis.

Pada tugas akhir ini dilakukan analisis penjejakan objek menggunakan algoritma camshift pada aplikasi rambu lalu lintas dimana masukannya berupa gambar bergerak dengan banyak objek lain disekitar rambu lalu lintas. Selain analisis mengenai algoritma penjejakan, tugas akhir ini pun melakukan analisis mengenai ekstraksi ciri yang digunakan yaitu *gabor filter* dan metode klasifikasi yang digunakan yaitu metode KNN. Pada penelitian sebelumnya, pelacakan dan pengenalan objek dengan metode *Template Mantching* memperoleh keberhasilan sebesar 74,95 %^[2].

2. Dasar Teori

2.1 Rambu Lalu Lintas

Rambu lalu lintas merupakan alat pelengkap jalan raya yang mengandung informasi berupa peringatan, larangan, maupun petunjuk. Keberadaan rambu lalu lintas bertujuan untuk membantu pengguna jalan raya agar lebih tertib dalam menggunakan jalan raya.

Berikut dipaparkan lebih jelas mengenai jenis-jenis rambu lalu lintas yang biasa ditemui di jalan raya.

1. Rambu peringatan

Rambu peringatan memiliki fungsi untuk memberi peringatan kepada pengendara mengenai kondisi jalan atau tempat berbahaya pada jalan di depan pengendara agar pengendara dapat lebih berhati-hati. Adapun warna dasar dari rambu peringatan adalah kuning dengan warna simbol berupa hitam.

2. Rambu perintah

Rambu perintah memberikan informasi kepada pengendara tentang suatu aksi tertentu yang harus dilakukan pengguna jalan raya, serta memberi informasi mengenai keberadaan suatu tempat-tempat penting tertentu seperti tempat ibadah, rumah sakit, dan sebagainya. Warna dasar rambu perintah adalah biru dengan warna simbolnya berupa putih.

3. Rambu larangan

Rambu larangan merupakan rambu yang memiliki tiga perpaduan warna yaitu putih, merah, dan hitam. Rambu perintah adalah jenis rambu yang memberikan suatu larangan kepada pengendara agar tidak melakukan suatu aksi tertentu yang telah ditetapkan.

2.2 Ekstraksi Ciri

2.2.1. Gabor Wavelet^{[3][4]}

Gabor Wavelet merupakan filter spasial bandpass yang optimum meminimalisasi ciri yang tidak penting dalam domain spasial dan frekuensi. Gabor dibentuk dengan memodulasikan gelombang sinusoida dengan *Gaussian envelope* dan menghasilkan kernel yang bertugas sebagai filter. Kernel ini memiliki nilai real dan imajiner yang kemudian akan dikonvolusikan dengan citra agar didapatkan ciri khusus dari citra tersebut. Bentuk representasi kernel real dan imajiner dari gabor ditunjukkan pada persamaan 1 dan 2:

$$\begin{aligned}
 \text{Real:} \quad g(x, y; \lambda, \theta, \phi, \sigma, \gamma) &= \exp\left(-\frac{x^2 + \gamma^2 y^2}{2\sigma^2}\right) \cos\left(2\pi \frac{x'}{\lambda} + \phi\right) & (1) \\
 \text{Imajiner:} \quad g(x, y; \lambda, \theta, \phi, \sigma, \gamma) &= \exp\left(-\frac{x^2 + \gamma^2 y^2}{2}\right) \sin\left(2\pi \frac{x'}{\lambda} + \phi\right) & (2)
 \end{aligned}$$

Yang mana, $x' = x \cos \theta + y \sin \theta$, $y' = -x \sin \theta + y \cos \theta$

$$\sigma = \frac{\lambda}{\pi} \sqrt{\frac{\ln 2}{2} \cdot \frac{2^b + 1}{2^{b-1}}}$$

Parameter gabor filter :

λ = panjang gelombang wavelet

θ = mengatur besarnya rotasi dari gabor filter (0° - 360°)

ψ = fasa offset, membedakan real dan imajiner

γ = *spatial aspect ratio*, menentukan elipsitas gabor filter

σ = sigma dari *Gaussian envelope*, menunjukkan seberapa luas pengaruh dari *gabor wavelet*

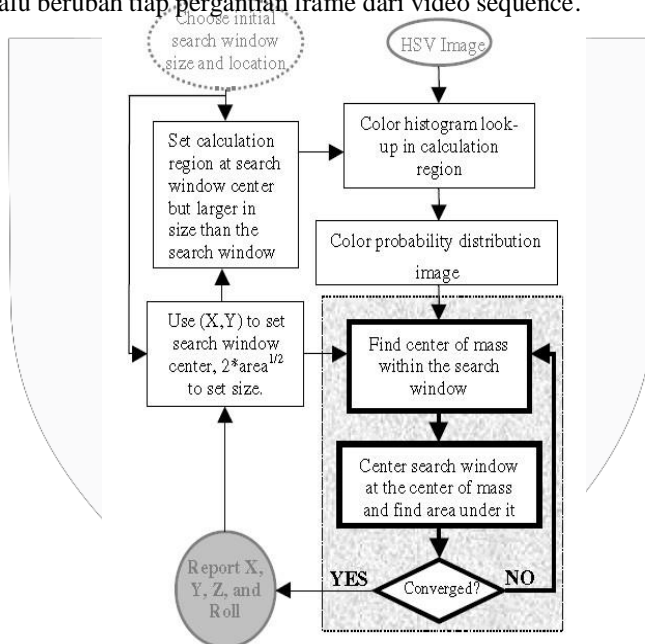
Sebagai ciri, akan diambil magnitude dari nilai kernel real dan imajiner yang sudah dikonvolusikan dengan citra menggunakan persamaan 3.

$$Output = \sqrt{R_{ave}^2 + I_{ave}^2} \tag{3}$$

Dimana R_{ave} merupakan hasil konvolusi antara kernel real dengan citra dan I_{ave} merupakan hasil konvolusi antara kernel imajiner dengan citra.

2.3 Algoritma Camshift^[5]

CamShift adalah singkatan dari *Continuously Adaptive Mean Shift* yang merupakan pengembangan dari algoritma Mean Shift yang dilakukan secara terus menerus untuk melakukan penyesuaian terhadap distribusi probabilitas warna yang selalu berubah tiap pergantian frame dari video sequence.



Gambar 2 Alur Algoritma Camshift^[5]

Alur dari algoritma CamShift dapat dilihat pada Gambar 2. Tahap pertama adalah melakukan penghitungan terhadap nilai *hue* dari objek target. Nilai histogram *hue* hasil perhitungan tersebut selanjutnya akan disimpan untuk digunakan sebagai *lookup table* (LUT) distribusi probabilitas citra. Tahap kedua adalah penginisiasian ukuran *search window* awal yaitu seukuran dengan *frame* pertama yang datang dari kamera. Tahap kedua atau *preprocessing* adalah pengekstraksian fitur dari setiap *frame* yang diterima. Tahap ketiga merupakan inti dari algoritma *CAMShift*. Tidak sama dengan algoritma *MeanShift* yang didesain untuk gambar statis, algoritma *CAMShift* didesain untuk distribusi gambar yang bersifat dinamis. Hal ini terjadi karena objek yang dideteksi sebagai video bergerak yang akan menyebabkan lokasi dan ukuran *search window* berubah terhadap waktu. Oleh karena itu algoritma *CAMShift* akan selalu mengeset ukuran *search window* pada setiap pengolahan *frame* dengan kata lain *search window* akan lebih adaptif.

2.4 K-Nearest Neighbor

K-Nearest Neighbor (K-NN) adalah salah satu metode klasifikasi objek berdasarkan nilai jarak terdekat antara data latih dengan data uji^[6]. Data latih memiliki titik titik latih yang disebut sebagai kelas (k). Ketika diberikan sebuah sampel uji maka sistem akan melakukan perhitungan sedemikian hingga diperoleh sejumlah titik latih yang memiliki jarak terdekat dengan sampel uji tersebut.

Komponen lainnya yang mempengaruhi keakurasian dari klasifikasi ini adalah *distance metric*, yaitu metode yang dilakukan untuk menghitung jarak ketetanggaan. Secara default K-NN menggunakan metode *euclidian distance* untuk melakukan perhitungan jarak berdasarkan persamaan 4.^[6].

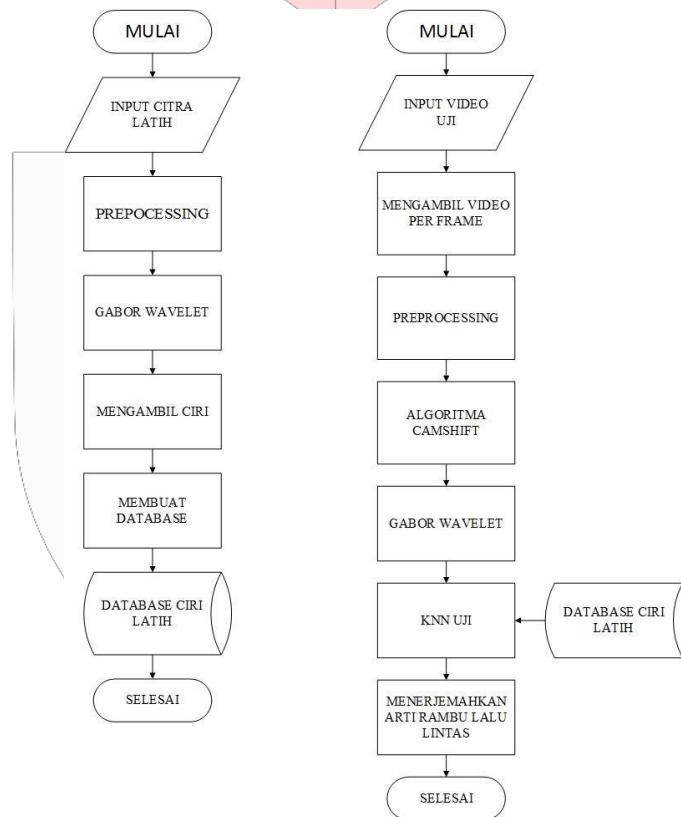
$$d_e = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{4}$$

- d = jarak terdekat
- (x_1, y_1) = koordinat data latih
- (x_2, y_2) = koordinat data uji

K-NN memiliki tingkat keakurasian yang tinggi. Pada matlab, untuk melakukan klasifikasi dengan K-NN dapat dilakukan dengan memanggil fungsi *knnclassification*.

3. Pembahasan

3.1 Perancangan Sistem



Gambar 3 Diagram Alir Pelatihan Sistem(kiri) dan Pengujian Sistem(kanan)

3.2 Analisis Hasil Simulasi Sistem

3.2.1. Analisis Parameter *Initial Search Window* Terhadap Waktu Komputasi Sistem

Pengujian pertama dilakukan untuk melihat pengaruh perubahan nilai inisialisasi *pixel* citra yang akan ditandai sebagai hasil *tracking* terhadap waktu komputasi sistem. Pada skenario ini pengujian dilakukan terhadap 74

data video yang diambil rata-rata waktu komputasinya. Data lengkap tabel percobaan dapat dilihat pada lampiran B buku ini.

Tabel 1 Pengujian Parameter P (Initial Search Window)

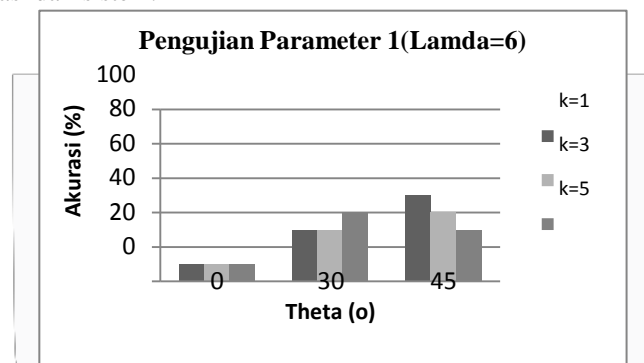
Skenario Pertama	
Parameter Camshift(P)	Waktu Komputasi Rata-rata
1	5,161
10	8,811
100	123,276

Berdasarkan tabel 1 kita dapat melihat bahwa semakin besar nilai *initial search window* maka semakin besar pula waktu komputasi yang diperlukan. Hal tersebut disebabkan karena apabila nilai *initial search window* semakin besar maka sistem akan melakukan pelacakan dengan luas daerah yang lebih besar sehingga dibutuhkan waktu yang lebih lama untuk menemukan daerah yang dimaksud sebagai rambu-rambu lalu lintas. Sebaliknya ketika nilai *initial search window* kecil maka luas daerah yang akan dideteksi semakin kecil sehingga sistem akan langsung melakukan penandaan pada daerah yang telah lolos dari proses *preprocessing*. Dari pengujian ini maka diambil nilai *initial search window* terkecil untuk digunakan pada pengujian selanjutnya agar diperoleh pengujian dengan waktu komputasi yang kecil.

3.2.2. Analisis Kombinasi Parameter Filter Gabor dan Klasifikasi KNN

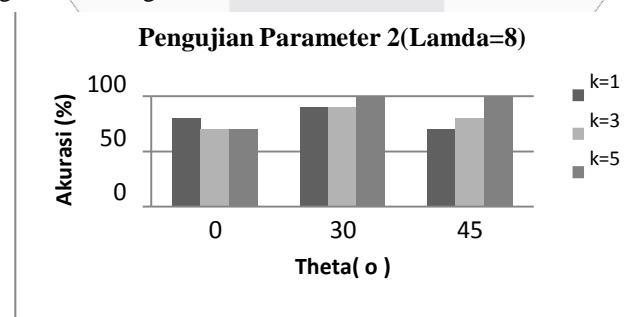
Pengujian ini dilakukan dengan menggunakan 10 video yang berbeda. Pengujian dilakukan dengan mengubah 2 parameter untuk proses ekstraksi ciri dan 1 parameter untuk proses klasifikasi.

Tujuan dari pengujian ini adalah memperoleh kombinasi parameter dengan hasil terbaik. Pengujian ini akan sangat mempengaruhi akurasi dari sistem.



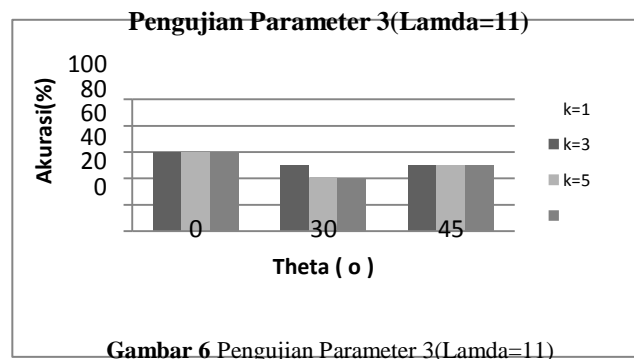
Gambar 4 Pengujian Parameter 1 (Lamda=6)

Pada gambar 4 terlihat bahwa kombinasi parameter terbaik untuk nilai lamda=6 adalah kombinasi dengan nilai theta=45⁰ dan k=1 dengan akurasi 50%. Berdasarkan hasil ini belum dapat disimpulkan bahwasanya kombinasi parameter tersebut dapat digunakan sebagai kombinasi terbaik untuk sistem sebab akurasinya masih cukup kecil.



Gambar 5 Pengujian Parameter 2 (Lamda=8)

Pada gambar 5 terlihat bahwa terdapat dua kombinasi parameter dengan nilai terbaik untuk nilai lamda=8, yaitu kombinasi dengan nilai theta=30⁰ dan k=5 dan kombinasi parameter theta=45⁰ dan k=5 dengan akurasi yang sama yaitu 100%.



Pada gambar 6 terlihat bahwa terdapat tiga kombinasi parameter dengan nilai terbaik untuk nilai lamda=11, yaitu kombinasi dengan nilai theta=0⁰ dan k=1, kombinasi parameter theta=0⁰ dan k=3, dan kombinasi parameter theta=0⁰ dan k=5 dengan akurasi yang sama yaitu 60%.

Berdasarkan ketiga kombinasi parameter terbaik dari masing-masing nilai lamda dapat diperoleh kesimpulan parameter-parameter terbaik dari setiap lamda yang ditunjukkan pada tabel 2.

Tabel 2 Kombinasi Parameter Terbaik Dari Setiap Nilai Lamda

NO	Kombinasi Parameter	Akurasi (%)
1	P=1,Lamda=6,Theta=45,k=1	50
2	P=1,Lamda=8,Theta=30,k=5	100
3	P=1,Lamda=8,Theta=45,k=5	100
4	P=1,Lamda=11,Theta=0,k=1	60
5	P=1,Lamda=11,Theta=0,k=3	60
6	P=1,Lamda=11,Theta=0,k=5	60

Berdasarkan tabel 2 diperoleh kombinasi parameter terbaik secara keseluruhan adalah kombinasi dengan nilai P=1, Lamda=8, Theta= 30^o, k=5, dan kombinasi dengan nilai P=1, Lamda=8, Theta= 45^o, k=5.

3.2.3. Analisis Hasil Pengujian Sistem Secara Keseluruhan

Tujuan dari pengujian ini adalah untuk menguji ketahanan dan akurasi sistem secara keseluruhan. Pengujian ini dilakukan pada 45 video berbeda dengan menggunakan dua kombinasi parameter terbaik yang telah diperoleh dari pengujian sebelumnya.

Tabel 3 Hasil Pengujian Sistem Secara Keseluruhan

NO.	Parameter Pengujian	Akurasi Total	Waktu komputasi rata-rata
1	P=1,Lamda=8,Theta=30,k=5	75.56 %	12.395 sekon
2	P=1,Lamda=8,Theta=45,k=5	77.78 %	12.395 sekon

Berdasarkan tabel 3 diperoleh akurasi terbaik dari sistem pendeteksian rambu lalu lintas yaitu 77.78% dengan kombinasi parameter yang digunakan adalah P=1, Lamda=8, Theta=45^o dan k=5. Nilai parameter P untuk algoritma Camshift mempengaruhi waktu komputasi dari sistem. Untuk pengujian ini digunakan parameter P=1 karena berdasarkan pengujian pada skenario pertama terbukti bahwa semakin kecil nilai dari parameter P maka waktu komputasi sistem akan semakin kecil. Nilai lamda untuk parameter gabor berpengaruh pada besar frekuensi yang akan mempengaruhi kerapatan sinyal sinusoidal sebagai sinyal modulasi dari filter gabor. Nilai lamda=8 memiliki hasil yang terbaik untuk sistem ini karena pada nilai lamda=8 ciri untuk masing-masing citra latihan lebih jelas perbedaannya. Sementara itu nilai theta akan mempengaruhi sudut pengambilan ciri dari gabor wavelet, hal tersebut menyebabkan pengambilan ciri dari citra latihan akan lebih rinci karena sudut 45^o dapat direfleksikan merata disetiap bagian citra. Sementara itu, pada nilai k=5 dapat diperoleh hasil terbaik karena klasifikasi KNN merupakan klasifikasi yang dilakukan dengan mencocokkan ciri antara tetangga terdekatnya. Nilai k=5 semakin baik untuk

sistem membuktikan bahwa semakin banyak sampel tetangga yg digunakan dalam membandingkan ciri maka hasil akan semakin baik.

4. Kesimpulan

Kesimpulan yang dapat diambil dari penelitian ini adalah:

1. Pada pelacakan rambu lalu lintas menggunakan algoritma *Camshift* pengujian dilakukan dengan mengubah parameter P, akurasi dengan Algoritma *Camshift* yang paling baik diperoleh dari nilai parameter $P=1$ pada pelacakan rambu dilarang berhenti dengan akurasi sebesar 73.33% dan waktu komputasi rata-rata 5.161 detik. Semakin besar nilai P maka semakin lama waktu komputasi yang diperlukan dan semakin kecil akurasi pelacakan.
2. Pengenalan rambu-rambu lalu lintas dengan ekstraksi ciri berupa filter gabor dan klasifikasi KNN diperoleh akurasi terbaik pada nilai parameter $\lambda=8$, $\theta=45$, dan $k=5$ sebesar 100%.
3. Performansi sistem secara keseluruhan diperoleh akurasi sebesar 77.78% dengan waktu komputasi rata-rata sebesar 12.395 detik.

Daftar Pustaka

- [1] C. Zhang, Y. Qiao, E. Fallon, and C. Xu. An improved camshift algorithm for target tracking in video surveillance. In Proceedings of 9th. IT and T Conference, number 12, 2009.
- [2] Indra Pramana, M Zén, Setiawardhana. 2011. "Tracking Objects Menggunakan Metode Template Matching Berbasis Stereo Vision". Tugas Akhir. Surabaya: ITS.
- [3] Chunming Li, Chenyang Xu, Changfeng Gui, and Martin D. Fox. "Distance Regularized Level Set Evolution And Its Application To Image Segmentation". IEEE Trans. Image Process. Vol. 19, No. 12. 2010.
- [4] Purnomo, Mauridhi Hery dan Muntasari, Arif. (2010). *Konsep Pengolahan Citra Digital dan Ekstraksi Fitur*. Surabaya: Graha Ilmu.
- [5] Intel Corporation. 1999. *Open Source Computer Vision Library Reference Manual*. USA: Intel Corporation
- [6] Putra, Darma. 2010. *Pengolahan Citra Digital*, Yogyakarta: Andi

