

## PERANCANGAN SISTEM OTOMATISASI PALANG PINTU KERETA API BERBASIS *MOTION DETECTION*

### *AUTOMATIC DESIGN SYSTEM TRAIN GATE CROSSING BASED ON MOTION DETECTION*

Muhammad Fayyadh<sup>1</sup>, Unang Sunarya, ST., MT.<sup>2</sup>, Ramdhan Nugraha, S.Pd., MT.<sup>3</sup>

<sup>1,3</sup>Fakultas Teknik Elektro, Universitas Telkom

<sup>2</sup>Fakultas Ilmu Terapan, Universitas Telkom

<sup>1</sup>[fayyadh.muhammad@gmail.com](mailto:fayyadh.muhammad@gmail.com), <sup>2</sup>[unangsunarya@telkomuniversity.ac.id](mailto:unangsunarya@telkomuniversity.ac.id), <sup>3</sup>[ramdhan@telkomuniversity.ac.id](mailto:ramdhan@telkomuniversity.ac.id)

#### Abstrak

Pada tahun 2009 sampai 2011 prosentase penyebab kecelakaan kereta api yang terjadi di luar faktor eksternal, didominasi oleh *human error* atau sumber daya manusia (SDM) dari operator yakni mencapai 24%. Pada penelitian ini telah dirancang sebuah sistem teknologi pada palang pintu kereta api yang merupakan penyempurnaan dari penelitian sebelumnya. Pada penelitian sebelumnya, didapatkan waktu bahwa palang pintu kereta api menutup sempurna selama 15,92 detik dengan *delay* pengiriman informasi ke palang pintu sebesar 1,5 detik dari jarak 1,2 Km<sup>[1]</sup>. Namun, pada penelitian sebelumnya belum ada analisis keakuratan dari sensor deteksi yang digunakan.

Pada penelitian kali ini diberikan analisis dari pendeteksian kereta api dengan *video processing* dengan menggunakan metode *motion detection* sebagai deteksi objek Bergeraknya dan Transformasi Hough sebagai deteksi bentuk objek, sehingga terlihat keakuratan metode yang digunakan pada teknologi palang pintu kereta api otomatis. *Output* dari proses deteksi kereta api yang menggunakan kedua metode tersebut dikirimkan ke arduino. Arduino mengeksekusi instruksi selanjutnya sesuai dengan *output* yang diterima. Jika hasil yang terdeteksi merupakan kereta api, maka palang pintu kereta api tertutup. Namun jika bukan kereta api, palang pintu tidak tertutup.

Perbedaan antara penelitian ini dengan penelitian sebelumnya adalah dalam hal deteksi kereta api. Bila penelitian sebelumnya menggunakan sensor untuk dapat mendeteksi kereta api, pada penelitian ini digunakan *vision* untuk dapat mendeteksi kereta api. Hasil yang diperoleh dari penelitian ini adalah sistem dapat mendeteksi objek bergerak yang berupa kereta api dengan 0.03538 detik dan dengan tingkat keakuratan mencapai 100%. Hasil tersebut adalah hasil maksimal yang diperoleh oleh sistem dalam mendeteksi objek berupa kereta api.

**Kata Kunci :** Kereta api, *Motion detection*, Transformasi Hough, *Arduino*.

#### Abstract

In 2009 until 2011 percentage cause of train accidents that occur beyond external factors, dominated by human error or human resources (HR) of operators which reached 24%. In this research has designed a system of technology to train gate crossing that is a refinement of previous research. In previous research, got the time that the train gate crossing close completely for 15.92 seconds to delay the transmission of information to the gatecrossing of 1.5 seconds from a distance of 1.2 Km<sup>[1]</sup>. However, the previous research has no analysis of the accuracy of detection sensors.

In this research is given the analysis of train detection by video processing using motion detection method as moving object detection and Hough Transform as object shape detection, so it generate the accuracy of the method that used in railway technology automatic gate crossing. The output of the train detection process that uses both methods send to the Arduino. Arduino execute the next instruction according to data that has received. If the result train is detected, the train gate crossing closed. But if not detected, gate crossing not close.

The difference between this research with previous research is in terms of train detection. When previous research using a sensor to detect a train, in this thesis used vision to be able to detect trains. The result is the system can detect moving objects which is train with 0.03538 seconds and with 100% accuracy rate. These result is maximum result that obtained by the system to detect an object which is a train.

**Key words :** Train, *Motion Detection*, Hough Transform, *Arduino*.

#### 1. Pendahuluan

Kereta api selalu menjadi transportasi umum untuk menghindari kemacetan. Terutama pada saat hari-hari besar seperti Hari Raya Idul Fitri. Rel kereta api yang bersinggungan dengan jalan raya

menjadi titik yang paling beresiko terhadap kecelakaan. Prosentase penyebab kecelakaan kereta api yang terjadi sepanjang 2009-2011 di luar faktor eksternal, didominasi oleh human error atau sumber daya manusia (SDM) dari operator yakni

mencapai 24%. Disusul penyebab lainnya adalah faktor sarana 21% dan prasarana 10%. Dari data yang dilampirkan di atas, penyebab kecelakaan kereta api terbesar terletak pada human error.

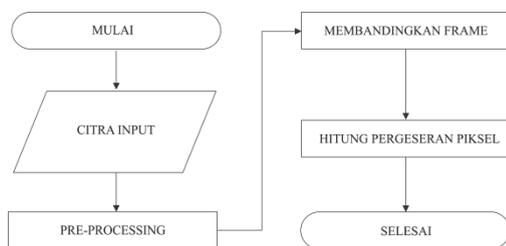
Dalam jurnal ini dibahas tentang sistem otomatisasi palang pintu kereta api dengan menggunakan beberapa algoritma. Untuk mendeteksi objek bergerak sistem otomatisasi palang pintu kereta api menggunakan metode *motion detection* dan sebagai penanda (*marking*) pada objek kereta api digunakan metode Transformasi Hough. Proses sistem dilakukan pada PC dan Mikrokontroler (*Arduino*). Hasil dari proses tersebut diteruskan ke aktuator yaitu motor DC (*Servo Std.*) untuk membuka atau menutup palang pintu kereta api sesuai dengan hasil dari proses tersebut.

Pada penelitian sebelumnya, didapatkan waktu bahwa palang pintu kereta api menutup sempurna selama 15,92 detik dengan delay pengiriman informasi ke palang pintu sebesar 1,5 detik dari jarak 1,2 Km<sup>[1]</sup>. Namun, pada penelitian sebelumnya belum ada analisis keakuratan dari sensor deteksi yang digunakan. Pada tugas akhir kali ini diberikan analisis dari pendeteksian kereta api dengan video processing, sehingga terlihat keakuratan metode yang digunakan pada teknologi palang pintu kereta api otomatis.

**2. Dasar Teori**

Ada dua metode yang digunakan dalam sistem ini yaitu *motion detection* dan transformasi Hough. Kedua metode ini merupakan metode yang digunakan untuk pengolahan citra baik *real-time* maupun non *real-time*.

Motion Detection adalah sebuah fitur video analisis yang berfungsi untuk mendeteksi adanya pergerakan (orang, mobil, dan lain-lain) dan hampir dimiliki berbagai merek IP camera atau network camera. Sistem pendeteksi gerakan ini dapat menganalisa frame video dan menentukan apakah apakah telah terjadi pergerakan di area yang telah ditentukan oleh pengguna. Sensivitas sistem pendeteksi gerakan ini dapat disesuaikan dengan kondisi yang ada<sup>[2]</sup>.



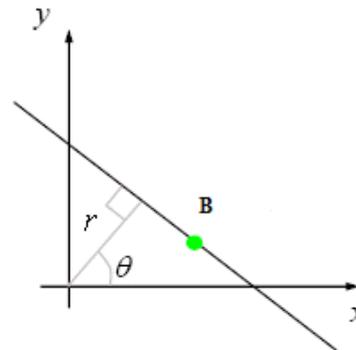
Gambar 2.1 Flow Chart Motion Detection<sup>[2]</sup>

Transformasi Hough menggunakan bentuk parametrik dan menggunakan pemungutan suara terbanyak (*voting*) untuk menentukan nilai parameter yang tepat. Apabila dalam citra

terdapat beberapa garis yang saling berpotongan pada suatu titik, maka apabila kemudian titik tersebut ditransformasi ke dalam ruang parameter didapati bahwa transformasi dalam ruang parameter adalah sebuah garis lurus dengan persamaan garis dinyatakan sebagai,

$$y_i = mx_i + c$$

*m* = Nilai Gradien  
*c* = Nilai Konstanta

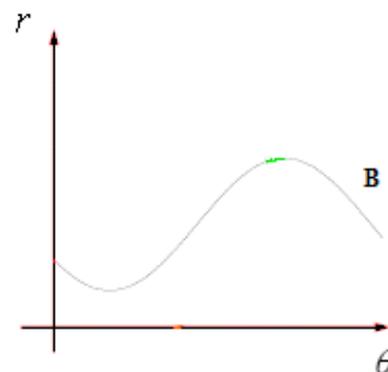


Gambar 2.2 Citra sebuah garis dalam Kartesian dengan satu titik uji<sup>[3]</sup>

Perhitungan transformasi dilakukan dengan cara menghitung tiap titik dalam citra kedalam semua variasi nilai *m - c*. Dalam kenyataannya, apabila ditemui sebuah garis vertikal, maka terjadi masalah dalam penghitungannya dikarenakan garis vertikal mempunyai nilai gradien kemiringan yang besarnya tak berhingga ∞. Sebagai alternatifnya digunakan persamaan<sup>[3]</sup>

$$r = x \cos \theta + y \sin \theta$$

*r* = Nilai Radius  
*θ* = Nilai Derajat

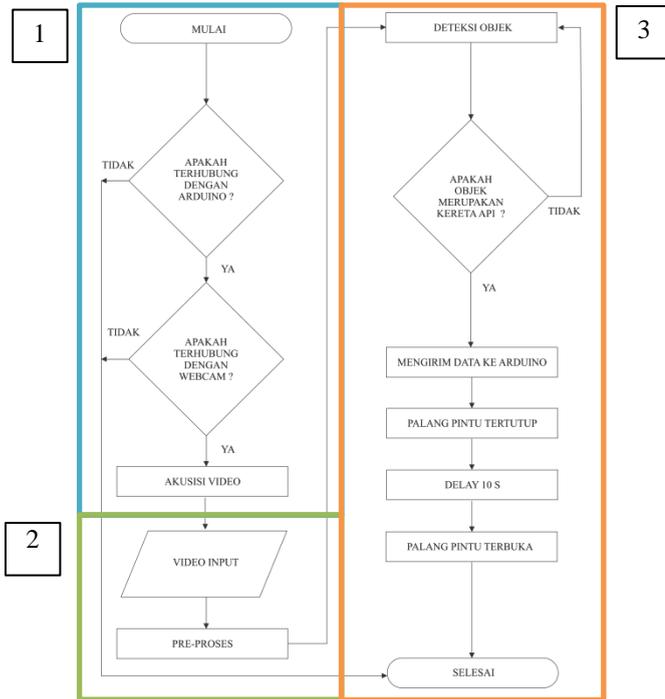


Gambar 2.3 Hasil transformasi dari satu titik uji dalam ruang parameter *r - θ*<sup>[3]</sup>

Dari dua metode tersebut proses deteksi objek berupa kereta api dilakukan. Hasil dari proses deteksi diproses kembali oleh Mikrokontroler (*Arduino*) untuk menentukan instruksi yang dieksekusi oleh aktuator<sup>[3]</sup>.

**3. Perancangan Sistem dan Implementasi**

Proses sistem otomatisasi palang pintu kereta api dijelaskan dalam diagram alir dibawah ini.



Gambar 3.1 Skema Perancangan

Dari diagram di atas dapat diketahui ada beberapa langkah untuk dapat mendeteksi objek bergerak berupa kereta api. Langkah-langkah tersebut dijelaskan pada tabel di bawah ini.

Tabel 3.1 Langkah Proses

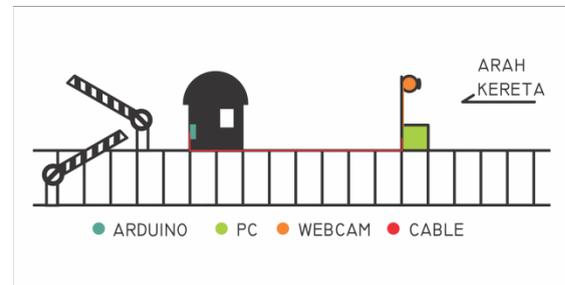
Tahap	Proses	Deskripsi
1	Inisialisasi	Inisialisasi merupakan tahap pertama yang dilakukan sistem untuk mengecek konektivitas perangkat keras (Arduino dan webcam) dengan perangkat lunak (program sistem). Apabila perangkat keras dinyatakan terkoneksi oleh perangkat lunak maka sistem melakukan akuisisi video yaitu video di resize ke dalam ukuran 640x480 untuk mempercepat waktu komputasi.
2	Pre-Proses	Setelah tahap inisialisasi dilakukan, maka webcam merekam video. Video input ini yang kemudian nanti di ubah menjadi citra HSV. Setelah diubah menjadi citra HSV, video disaring oleh HSV Filter sesuai dengan nilai acuan HSV yang telah ditentukan. Output dari HSV

		Filter diubah menjadi citra biner.
3	Proses Deteksi	Citra biner diolah oleh metode motion detection agar dapat dideteksi yangmana yang merupakan objek bergerak dan yang mana yang tidak. Selanjutnya citra biner tersebut diolah oleh metode transformasi Hough agar dapat terdeteksi objek yang merupakan kereta api dan yang bukan merupakan kereta api. Jika objek bergerak terdeteksi sebagai kereta api, maka program mengirim data ke Arduino untuk memberikan instruksi agar Arduino menutup palang pintu kereta api. Setelah 10 detik palang pintu terbuka kembali.

Untuk dapat mengetahui cara kerja sistem secara keseluruhan maka dapat dilihat dari blok diagram dibawah ini dan dari ilustrasi sistem yang telah dibuat.



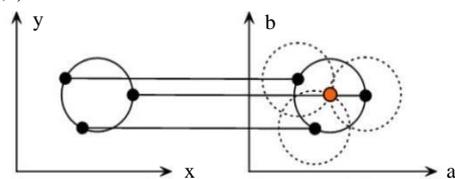
Gambar 3.2 Blok Diagram Sistem



Gambar 3.3 Ilustrasi Sistem

**3.1 Circle Hough Transform**

Dalam perkembangannya, Transformasi Hough dapat mendeteksi beberapa bentuk salah satunya lingkaran. Circle Hough Transform adalah pengembangan metode Transformasi Hough untuk mendeteksi lingkaran. Konsep dasarnya sama dengan Transformasi Hough. Namun, dalam Circle Hough Transform digunakan domain yang berbeda yaitu transformasi dari domain (x,y) ke domain (a,b,r).<sup>[4]</sup>



Gambar 3.4 Circle Hough Transform<sup>[4]</sup>

Dalam jurnal ini, *Circle Hough Transform* digunakan untuk menandai objek dari citra yang sudah melewati tahap pre-proses dan sudah diolah dengan metode *Motion Detection*.

**3.2 Pengambilan Keputusan**

Keputusan yang diambil untuk mengenali objek kereta api adalah hasil pengolahan citra terdeteksi ada objek bergerak dengan Motion Detection (ditandai dengan kotak merah) dan penandaan (ditandai dengan lingkaran biru) pada objek bergerak tersebut dengan menggunakan Circle Hough Transform. Apabila salah satu dari keduanya tidak terpenuhi maka objek tidak dapat dinyatakan kereta api.

Apabila objek bergerak yang terdeteksi merupakan kereta api, maka program mengirimkan data. Dalam Tugas Akhir ini, data yang dikirim ke Arduino adalah 121 atau huruf 'y' jika diterjemahkan ke dalam ASCII. Arduino hanya memberikan instruksi kepada Servo jika data yang diterimanya adalah 121. Ketika Arduino menerima data berupa 121, maka Arduino menggerakkan Servo dari 0° sampai 90°. Setelah Servo berada diposisi 90°, maka ada waktu tunggu (delay) selama 10 detik. Kemudian Servo kembali pada posisi 0° setelah waktu tunggu tersebut berlalu.



Gambar 3.3 Objek Kereta Api telah terdeteksi

**3.3 Akurasi Sistem**

Akurasi sistem merupakan ukuran ketepatan sistem mengenali masukan yang diberikan sehingga menghasilkan keluaran yang benar. Secara matematis dapat dituliskan sebagai berikut :

**4. Pengujian Sistem dan Analisis**

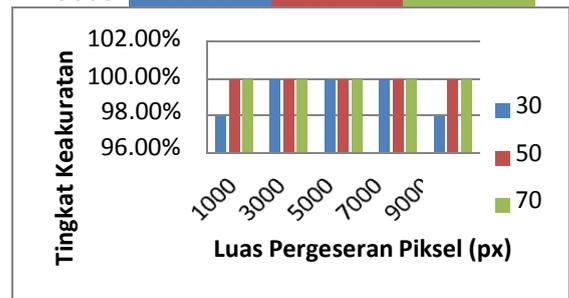
**4.1 Hasil Pengujian Indoor**

Pengujian tahap ini dilakukan untuk mengetahui parameter yang efektif agar dapat optimal pada kondisi outdoor nantinya. Parameter yang diujikan adalah luas daerah pergeseran piksel dan parameter sensitivitas deteksi *circle* dalam fungsi `cvHoughCircles`.

**A. Pengaruh Parameter Hough dan Luas Pergeseran Piksel**

Tabel 4.1 Tingkat Keakuratan Sistem pada Keadaan Indoor

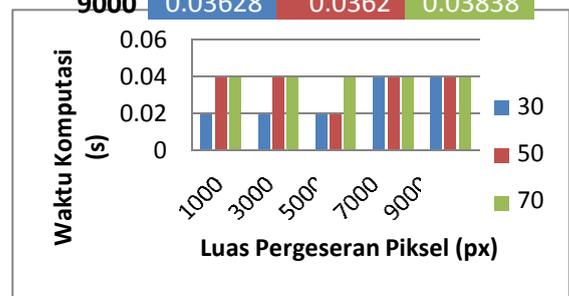
	30	50	70
1000	98.00%	100.00%	100.00%
3000	100.00%	100.00%	100.00%
5000	100.00%	100.00%	100.00%
7000	100.00%	100.00%	100.00%
9000	98.00%	100.00%	100.00%



Gambar 4.1 Grafik Tingkat Keakuratan Sistem pada Indoor

Tabel 4.2 Waktu Komputasi Sistem pada Keadaan Indoor

	30	50	70
1000	0.03042	0.03638	0.03828
3000	0.03474	0.03566	0.03898
5000	0.03258	0.03462	0.0366
7000	0.0366	0.035	0.0392
9000	0.03628	0.0362	0.03838



Gambar 4.2 Grafik Waktu Komputasi Sistem pada Indoor

Dapat dilihat dari hasil pengujian di atas bahwa parameter Hough berpengaruh besar pada tingkat keakuratan dan waktu komputasi sistem. Hal ini disebabkan oleh fungsi dari parameter Hough yang diujikan adalah sebagai parameter sensitivitas untuk mendeteksi objek.

Semakin kecil nilai parameter Hough, maka semakin sensitif dalam mendeteksi objek. Oleh karena itu pada Gambar 4.1 terlihat bahwa parameter Hough yang bernilai 30 kurang akurat

dalam mendeteksi objek yang merupakan kereta api. Hal ini dikarenakan ada noise yang tertangkap oleh parameter tersebut sehingga sistem menganggap noise tersebut merupakan kereta api.

Sebaliknya, pada Gambar 4.2 terlihat bahwa semakin kecil parameter Hough maka semakin kecil pula waktu komputasinya. Hal ini disebabkan oleh tingkat selektif dari setiap parameter Hough untuk mendeteksi objek. Semakin cepat pendeteksian objek dilakukan maka semakin cepat pula waktu komputasi yang dihasilkan. Korelasi antara tingkat akurasi dengan waktu komputasi adalah semakin akurat sistem dalam mendeteksi

objek maka semakin lama waktu komputasi yang dibutuhkan.

Parameter luas pergeseran piksel terbukti tidak memberikan pengaruh yang cukup besar pada sistem. Hal ini dikarenakan eksekusi dari luas pergeseran piksel menggunakan perbedaan antara citra yang pertama kali diambil sebagai titik acuan, dibandingkan dengan citra-citra setelahnya. Sehingga pengambilan keputusan dalam mendeteksi objek adalah dengan mengukur luas pergeseran yang terjadi pada video input dengan membandingkan setiap citra pada video input tersebut. Dalam percobaan ini dilakukan pengujian parameter luas pergeseran piksel yang diatur agar tidak mudah untuk mendeteksi noise berukuran kecil. Sehingga sistem tidak mudah untuk dapat mendeteksi pergerakan yang diakibatkan noise tersebut.

Hasil terbaik adalah nilai tingkat keakuratan 100% dengan waktu komputasi 0.03462 detik diperoleh oleh parameter Hough bernilai 50 dengan nilai luas pergeseran piksel sebesar 5000. Nilai rata-rata tingkat keakuratan yang dihasilkan pada pengujian indoor ini adalah 99,73% dengan waktu komputasi sebesar 0.035994667 detik.

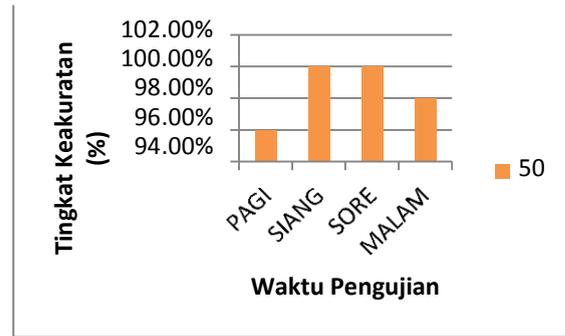
**4.2 Hasil Pengujian Outdoor**

Dari data hasil pengujian indoor, maka selanjutnya dilakukan pengujian outdoor agar dapat mengetahui kinerja sistem pada keadaan sebenarnya. Dari pengujian indoor didapatkan hasil terbaik adalah nilai parameter Hough 50 dengan luas pergeseran piksel 5000. Pengujian outdoor menganalisis seberapa besar pengaruh cahaya dalam mempengaruhi tingkat keakuratan dan waktu komputasi sistem.

**A. Pengujian dan Analisis Pengaruh Cahaya**

Tabel 4.3 Tingkat Keakuratan Sistem pada keadaan Outdoor

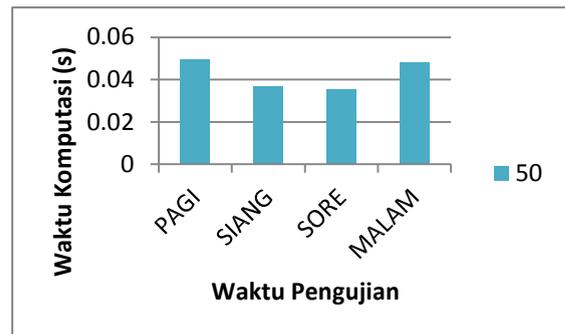
PAGI	96.00%
SIANG	100.00%
SORE	100.00%
MALAM	98.00%



Gambar 4.3 Grafik Tingkat Keakuratan pada Keadaan Outdoor

Tabel 4.4 Waktu Komputasi pada Keadaan Outdoor

PAGI	0.04966
SIANG	0.03672
SORE	0.03538
MALAM	0.04798



Gambar 4.4 Grafik Waktu Komputasi pada Keadaan Outdoor

Hasil pengujian outdoor menunjukkan bahwa tingkat keakuratan berbanding terbalik dengan waktu komputasi. Semakin tinggi tingkat keakuratannya, maka semakin kecil waktu komputasinya. Hal ini disebabkan oleh kondisi pencahayaan pada masing-masing waktu pengujian. Semakin baik pencahayaannya (objek terlihat dengan sangat jelas), maka semakin akurat sistem dalam mengenali kereta api.

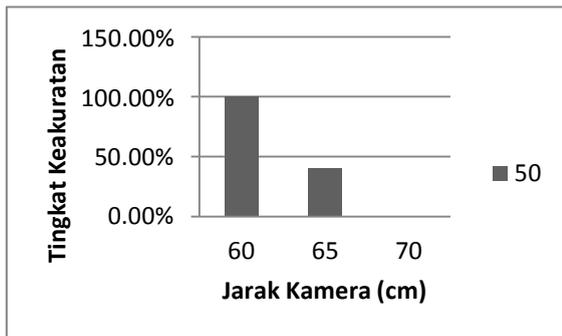
Dapat terlihat pada grafik bahwa pagi hari memberikan hasil tingkat keakuratan yang rendah dan waktu komputasi yang lama. Hal ini disebabkan oleh pencahayaan pada pagi hari yang tidak terlalu baik, sehingga dapat membuat sistem mendeteksi objek yang bukan kereta api. Sementara sebaliknya, pada siang, sore, dan malam hari memberikan hasil yang sangat baik. Tingginya tingkat keakuratan dan kecilnya waktu komputasi

ini disebabkan oleh pencahayaan yang baik dari masing-masing waktu pengujian.

**B. Pengaruh Jarak Pengambilan Data**

Tabel 4.5 Tingkat Akurasi Berdasarkan Jarak Pengambilan Data

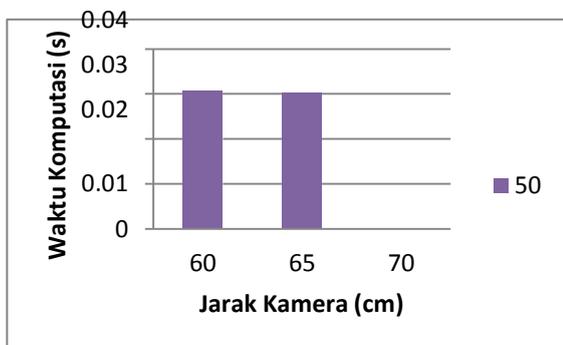
60 cm	100.00%
65 cm	40.00%
70 cm	0.00%



Gambar 4.5 Grafik Akurasi Sistem Berdasarkan Jarak Pengambilan Data

Tabel 4.6 Waktu Komputasi Berdasarkan Jarak Pengambilan Data

60 cm	0.03062
65 cm	0.03015
70 cm	0.0



Gambar 4.6 Grafik Waktu Komputasi Berdasarkan Jarak Pengambilan Data

Hasil pengujian menunjukkan bahwa jarak pengambilan data berpengaruh terhadap waktu komputasi. Hal ini disebabkan oleh pengaruh pergeseran piksel yang diolah oleh sistem. Semakin dekat jaraknya, maka pergeseran piksel semakin luas perubahannya yang membuat sistem lebih lama untuk mendeteksi objek bergerak. Sebaliknya, jika semakin jauh jaraknya maka semakin kecil luas pergeseran pikselnya sehingga dapat membuat proses deteksi lebih singkat. Namun ada batasan dalam jarak pengambilan data. Jika jarak terlalu

jauh maka objek tidak terdeteksi. Pada awal pengujian ini telah dilakukan pengambilan data dari jarak 70 cm dan sistem tidak dapat mendeteksi objek bergerak tersebut. Hal ini disebabkan oleh batas minimal dari luas pergeseran piksel yang tidak tercapai yaitu 5000 piksel.

Sementara itu pada grafik tingkat akurasi mengalami perubahan yang signifikan. Hal ini dikarenakan ada jarak pandang batas untuk dapat mendeteksi kereta api. Jika jarak semakin jauh maka objek kereta mempunyai peluang yang lebih kecil untuk terdeteksi disebabkan oleh semakin kecil objek dalam pandangan webcam untuk dapat terdeteksi.

**5. Kesimpulan**

Setelah melakukan pengujian dan analisis pada sistem, maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Hasil terbaik yang didapatkan pada pengujian *indoor* adalah tingkat akurasi mencapai 100% dan waktu komputasi 0.03462 detik dengan nilai parameter Hough 50 dan luas pergeseran piksel 5000. Pengujian *indoor* ini menggunakan 750 citra sebagai data uji.
2. Hasil terbaik yang didapatkan pada pengujian *outdoor* adalah tingkat akurasi mencapai 100% dan waktu komputasi 0.03538 detik dengan nilai parameter Hough 50 dan luas pergeseran piksel 5000. Pengujian *indoor* ini menggunakan 200 citra sebagai data uji.
3. Sistem dapat mendeteksi objek kereta api dengan tingkat akurasi maksimal dari jarak 60 m dengan parameter Hough 50 dan luas pergeseran piksel 5000.

**6. Daftar Pustaka**

[1] Nusantara, Puja Aqsony. 2011. IMPLEMENTASI SITEMPALANG PINTU KERETA API OTOMATIS : PENDETEKSI KERETA API DENGAN SENSOR INFRAMERAH. Bandung: IT Telkom.

[2] Sosa, Ibrahim. 2013. DETEKSI JARAK AMAN MENGENDARAI MOBIL DENGAN MEMPERHATIKAN PERUBAHAN LAJU MENGGUNAKAN METODE MOTION DETECTION. Bandung: IT Telkom.

[3] Sa'diyatul, Halimatus, R.Rizal Isnanto, dan Achmad Hidayatno. APLIKASI TRANSFORMASI HOUGH UNTUK DETEKSI GARIS LURUS. Semarang: Universitas Diponegoro.

[4] Riwinoto. *Penggunaan Algoritma Hough Tranforms Untuk Deteksi Bentuk Lingkaran pada Ruang 2D*. Batam: Politeknik Batam.

[5] Munir, Renaldi. 2004. Pengolahan Citra Digital dengan Pendekatan Algoritmik. Bandung: Penerbit Informatika.

- [6] Imadudin M., Adiwijaya, dan Fazmah A. Y. PERBAIKAN CITRA VIA HSV HISTOGRAM MENGGUNAKAN INTENSIFIKASI FUZZY. Bandung: STT Telkom.
- [7] Nugraha, Ramdhan, dkk. 2014. *A Research on Automated Railway Crossing Alert System using Arduino and Android*. Bandung: Universitas Telkom.
- [8] Saragih, Gustaf W. 2012. SISTEM PERINGATAN DAN PENGATURAN PINTU PERLINTASAN KERETA API BERDASARKAN DETEKSI GETARAN. Bandung: IT Telkom.
- [9] Septiana, Nanda. 2014. IMPLEMENTASI DAN ANALISIS MOTION IDENTIFICATION BERUPA GAYA BERJALAN MANUSIA MENGGUNAKAN SUPPORT VECTOR MACHINE. Bandung: Universitas Telkom.
- [10] Sukardjo. MM, Ir. Hartono Atmo. 2008. Lokomotif dan Kereta Rel Diesel di Indonesia. Depok: PT. Ilalang Sakti Komunikasi.