

Rancang Bangun *Prototype Smart Traffic Light* Menggunakan Metode *Background Subtraction* dan *Find Contour*

Hartoyo¹, Suryadhi², Joko Subur³

^{1,2,3} Program Studi Teknik Elektro, Fakultas Teknik dan Ilmu Kelautan,

Universitas Hang Tuah Surabaya

E-mail: namehartoyo@gmail.com, E-mail: suryadhi@hangtuah.ac.id, E-mail: joko.subur@hangtuah.ac.id

Abstrak— Kemacetan di Indonesia sering terjadi dikota-kota besar. Hal tersebut terjadi dikarenakan banyaknya volume kendaraan yang terus bertambah dan bertambahnya volume kendaraan tersebut tidak sebanding dengan kondisi ruas jalan yang tentunya cepat atau lambat akan terjadi penumpukan sejumlah kendaraan sehingga terjadi kemacetan. Sistem pengaturan lampu lalu lintas (*traffict light*) dengan menggunakan pengaturan berbasis waktu tetap seperti sekarang ini maka banyak atau sedikit-nya jumlah kendaraan yang berada pada jalur-jalur persimpangan tidak mempengaruhi kinerja lampu lalu lintas. Disamping itu kemacetan juga dapat mempengaruhi berbagai macam kegiatan masyarakat karena disamping berdampak pada kelancaran kegiatan masyarakat dan tak kalah pentingnya pada dampak pendapatan masyarakat. Maka pada penelitian ini akan dibuat sebuah perancangan *Prototype Smart traffic light* untuk mendeteksi tingkat kepadatan kendaraan yang berada pada persimpangan ruas jalan, sehingga dapat memperkirakan ruas jalan mana yang membutuhkan waktu hijau lebih banyak (kondisi ruas jalan padat) dan memperkirakan ruas jalan yang tidak membutuhkan waktu lampu hijau lebih (kondisi jalan kosong / Sedikit kendaraan). Dalam pendeteksian tingkat kepadatan kendaraan di *traffic light*, didapatkan sebuah hasil persentase kepadatan anatar lain 100%, 90%, 70%, 50%, dan 30%. Pada persentase 100% sistem mendeteksi kepadatan jalur sehingga dilakukan penambahan waktu, Persentase 90% sistem mendeteksi kepadatan jalur sehingga dilakukan penambahan waktu. Persentase 70% sistem mendeteksi kepadatan jalur sehingga dilakukan penambahan waktu. Persentase 50% sistem tidak mendeteksi kepadatan jalur sehingga tidak dilakukan penambahan waktu. Persentase 30% sistem tidak mendeteksi kepadatan jalur sehingga tidak dilakukan penambahan waktu.

Kata kunci : Pengaturan Lampu Lalu Lintas, *Traffic Light*, *Prototype Smart traffic light*.

Abstract— Congestion in Indonesia often occurs in big cities. This happens because the volume of vehicles continues to grow and the increase in the volume of vehicles is not proportional to the condition of the road which of course sooner or later there will be a buildup of a number of vehicles causing congestion. Traffic light regulation system (*traffic light*) using a fixed time-based setting as it is today, the large or small number of vehicles that are on the crossing lanes does not affect the performance of the traffic lights. Besides that, congestion can also affect various kinds of community activities because in addition to having an impact on the smooth running of community activities and no less important to the impact of community income. So in this study, a *Smart Traffic Light Prototype* design will be made to detect the density level of vehicles at road intersections, so that it can estimate which roads require more green time (congested road conditions) and estimate which roads do not require time. more green lights (empty road conditions / Few vehicles). In detecting the density level of vehicles at traffic lights, we get a percentage of density results including 100%, 90%, 70%, 50%, and 30%. At a percentage of 100% the system detects the density of the line so

that time is added, the percentage of 90% of the system detects the density of the line so that it takes an additional. The percentage of 70% of the system detects the density of the line so that it is done additional time. The percentage of 50% of the system does not detect the density of the line so that no additional time is carried out. The percentage of 30% of the system does not detect the density of the line so that no additional time is carried out.

Keyword: *Traffic Light Setting*, *Traffic Light*, *Prototype Smart traffic light*

I. PENDAHULUAN

Kemacetan di Indonesia sering terjadi dikota-kota besar. Hal tersebut terjadi dikarenakan banyaknya volume kendaraan yang terus bertambah dan bertambahnya volume kendaraan tersebut tidak sebanding dengan kondisi ruas jalan. Kemacetan juga dapat menyebabkan pengaruh dalam berbagai macam kegiatan masyarakat. Karena berdampak pada kelangsungan kegiatan masyarakat dan pentingnya juga pada dampak pendapatan masyarakat. Akibat kemacetan banyak waktu yang hilang dan berkurangnya kesempatan masyarakat untuk memperoleh pendapatan. Begitu juga dilihat dari aspek lain, dengan adanya kemacetan dapat menghambat lalu lintas perjalanan kendaraan darurat seperti ambulance dan mobil pemadam api. Kemacetan di Indonesia sendiri sering terjadi di persimpangan ruas jalan. Pada persimpangan jalan dipasang sebuah alat yang digunakan untuk mengatur kepadatan kendaraan [1].

Sistem pengaturan lampu lalu lintas (*traffict light*) yang secara umum digunakan di Indonesia selama ini adalah dengan pengaturan berbasis waktu tetap. Dengan menggunakan pengaturan berbasis waktu tetap seperti sekarang ini maka banyak atau sedikit-nya jumlah kendaraan yang berada pada jalur-jalur persimpangan tidak mempengaruhi kinerja lampu lalu lintas. Kondisi ini sering menjadi tidak efektif terutama ketika dari semua jalur persimpangan sepi sehingga seorang pengendara masih harus menunggu lampu hijau hingga waktu yang cukup lama. Demikian juga ketika pada kondisi dan waktu tertentu kendaraan dari salah satu jalur sangat padat sedangkan pada jalur lain sepi maka akan terjadi antrian kendaraan yang sangat panjang dengan pemanfaatan waktu lampu lalu lintas yang sesuai waktu setting awal (tetap), hal-hal demikian tentu menyebabkan *traffict light* tidak efektif dalam membantu kelancaran lalu lintas [3]. Maka pada penelitian ini akan dibuat sebuah perancangan *Prototype Smart traffic light* untuk mendeteksi tingkat kepadatan kendaraan yang berada pada persimpangan ruas jalan, sehingga dapat memperkirakan ruas jalan mana yang membutuhkan waktu hijau lebih banyak (kondisi ruas jalan padat) dan memperkirakan ruas jalan yang tidak membutuhkan waktu lampu hijau lebih (kondisi ruas

jalan tidak padat). Dalam pendeteksian kendaraan di traffic light, digunakan sebuah metode Background Subtraction dan Find Counturs. Metode Background subtraction merupakan metode pengohalan citra digital yang memisahkan antara latar belakang dengan latar depan (Objek), sedangkan metode find contour digunakan menganalisis bentuk dan mendeteksi objek yang telah dipisahkan menggunakan metode background subtraction

II. TINJAUAN PUSTAKA

A. Traffic Light

Lampu lalu lintas menurut UU no. 22/2009 tentang Lalu Lintas dan Angkutan Jalan: alat pemberi isyarat lalu lintas adalah lampu yang mengendalikan arus lalu lintas yang terpasang di persimpangan jalan, tempat penyeberangan pejalan kaki (zebra cross), dan tempat arus lalu lintas lainnya. Lampu ini yang menandakan kapan kendaraan harus berjalan dan berhenti secara bergantian dan berbagai arah. Pengaturan lalu lintas dipersimpangan jalan dimaksudkan untuk mengatur pergerakan kendaraan pada masing-masing kelompok pergerakan kendaraan agar dapat bergerak secara bergantian sehingga tidak saling mengganggu antar arus yang ada [6].

B. Citra Digital

Citra digital merupakan representatif dari citra yang diambil oleh mesin dengan bentuk pendekatan berdasarkan sampling dan kuantisasi. Sampling menyatakan besarnya kotak-kotak yang disusun dalam baris dan kolom. Dengan kata lain, sampling pada citra menyatakan besar kecilnya ukuran pixel (titik) pada citra, dan kuantisasi menyatakan besarnya nilai tingkat kecerahan yang dinyatakan dalam nilai tingkat keabuan (grayscale) sesuai dengan jumlah bit biner yang digunakan oleh mesin, dengan kata lain kuantisasi pada citra menyatakan jumlah warna yang ada pada citra. [6].

C. Absolute Difference

Metode Absolute Difference adalah metode yang berfungsi untuk menghitung perbedaan mutlak antara dua gambar dengan mengurangi satu dari yang lain dan mengambil nilai mutlak tersebut. Gambar yang ingin diproses harus memiliki tipe data atau ukuran yang sama [5].

D. Background Subtraction

Background subtraction banyak digunakan pada proyek-proyek berbasis pengolahan citra. Salah satu aplikasi yang sering menggunakan fungsi dari background subtraction ini adalah aplikasi penghitung jumlah pengunjung yang memasuki maupun meninggalkan ruangan ataupun kendaraan yang melewati suatu jalur dalam sistem informasi lalu lintas. Metode ini memisahkan manusia atau kendaraan dengan cara pembedaan latar belakang (background) dan manusia atau kendaraan (foreground) yang bergerak. Jika kondisi yang akan diamati oleh background subtraction hanya berupa latar belakang dan objek bergerak yang akan diamati, hal tersebut sangat mudah. Namun, ketika terdapat objek lain yang juga berpindah dari titik satu ke titik yang lain misalnya bayangan dari objek tersebut hal ini akan diproses juga sebagai foreground atau masuk ke dalam objek yang diklasifikasikan [2].

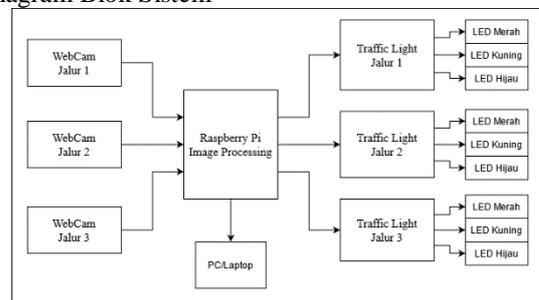
E. Find Contour

Kontur dapat dijelaskan hanya sebagai kurva yang menghubungkan semua titik kontinu (sepanjang batas), memiliki warna atau intensitas yang sama. Kontur adalah alat yang berguna untuk analisis bentuk dan deteksi dan pengenalan objek [4]. Dalam OpenCV, find contour seperti mencari objek putih dari latar belakang hitam. Maka objek yang ditemukan harus berwarna putih dan berlatar belakang hitam. Pada metode ini digunakan juga salah satu bagian dari find contour yaitu drawing contour yang digunakan untuk menggambar bentuk objek kendaraan dengan cara menghubungkan semua titik di sepanjang batas gambar yang memiliki intensitas yang sama.

III. METODE PENELITIAN

Peneliti mengambil topik penelitian permasalahan yang ada di jalan raya khususnya pada traffic light. Lampu lalu lintas juga masih menggunakan waktu yang bersifat tetap. Waktu pengaturan pada setiap persimpangan jalan tidak dapat disesuaikan berdasarkan tingkat kepadatan kendaraan. Seharusnya pengaturan waktu pada jalur yang mempunyai kepadatan kendaraan tinggi memiliki pengaturan waktu lampu hijau lebih lama dibandingkan dengan ruas jalan yang tingkat kepadatan kendaraannya rendah

A. Diagram Blok Sistem



Gambar 1 Diagram Blok Sistem

Pada gambar diatas dijelaskan bahwa pada bagian input terdapat 3 buah kamera webcam. Masing-masing kamera webcam akan ditempatkan pada setiap jalur traffic light yang akan digunakan untuk capture sebuah citra.

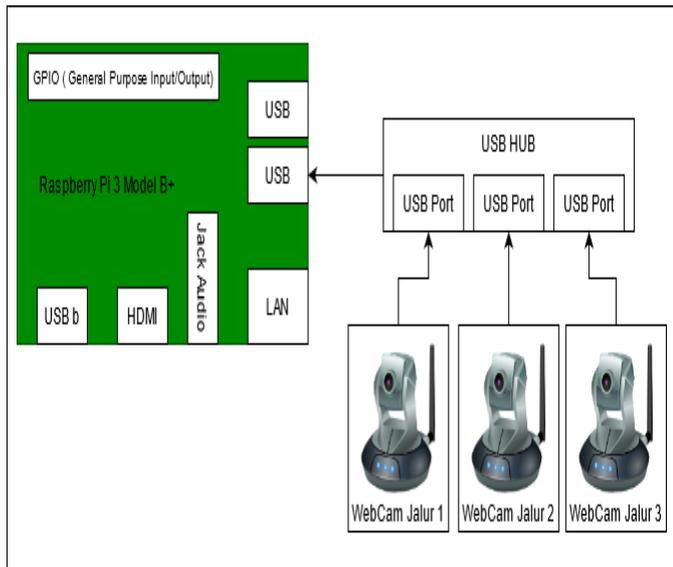
Pada controller raspberry pi akan dilakukan sebuah proses pengolahan citra digital dengan menggunakan metode background subtraction dan find contour. Setelah dilakukan proses pengolahan citra digital pada raspberry pi, dapat juga dilakukan sebuah monitoring melalui PC

Dari hasil persentase kepadatan volume kendaraan maka dapat dilakukan perbandingan persentase kendaraan tiap jalur guna dalam menentukan jalur mana yang membutuhkan waktu penyalaan lampu hijau lebih lama dan jalur yang tidak membutuhkan waktu penyalaan lampu hijau lebih. Apabila durasi waktu penyalaan lampu hijau telah ditentukan maka raspberry pi akan menyalakan prototype traffic light dengan durasi waktu yang telah ditentukan.

B. Perancangan Webcam pada Sistem Smart Traffic Light

Pada penelitian ini terdapat perancangan sistem webcam pada raspberry pi, perancangan sistem webcam pada raspberry pi menggunakan USB Hub sebagai penghubung antar 3 kamera webcam dengan raspberry pi. Setiap webcam akan berada pada jalur traffic light yang nantinya akan digunakan untuk pendeteksi jumlah kendaraan. Pada penelitian ini webcam yang

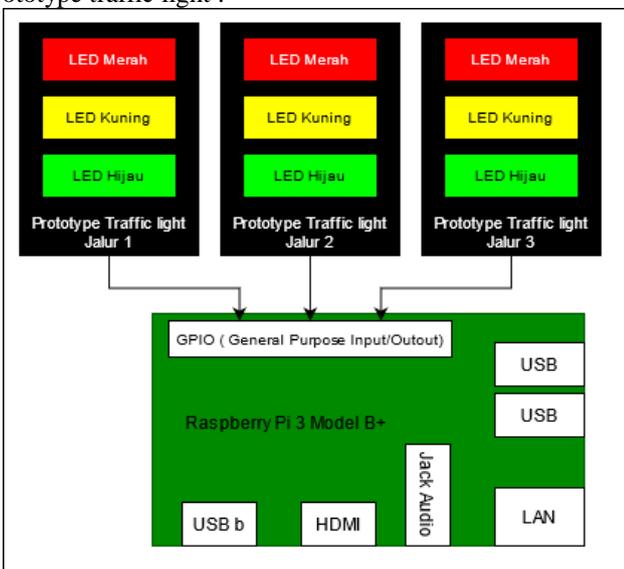
digunakan yaitu webcam Inforce TW-002 dengan resolusi 1080p . Berikut merupakan gambar perancangan sistem webcam pada raspberry pi :



Gambar 2 Perancangan Webcam pada Raspberry

C. Perancangan Prototype Traffic Light

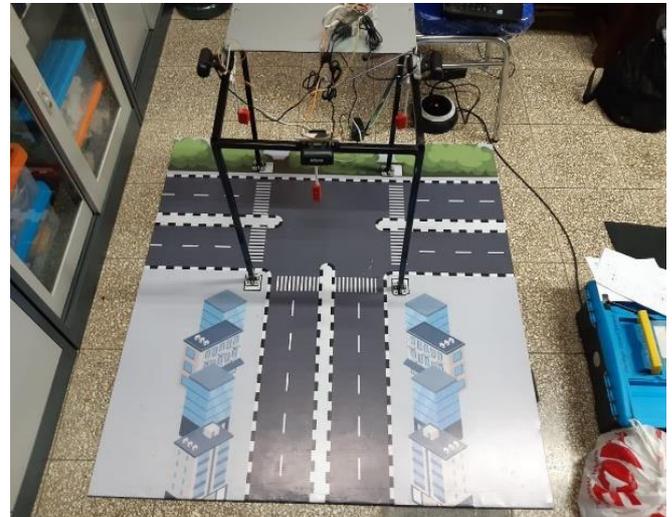
Pada penelitian ini perancangan prototype traffic light digunakan untuk mensimulasikan sebuah lampu lalu lintas seperti pada umumnya yang sering dijumpai pada setiap persimpangan jalan. Untuk traffic light sendiri digunakan sebagai alat untuk pengurai kemacetan dan mengatur jalannya lalu lintas pada persimpangan jalan. Perancangan prototype traffic light dibutuhkan beberapa part komponen seperti led dan sebuah resistor. Dibutuhkan sebuah 3 lampu led dengan warna yang berbeda seperti warna traffic light pada umumnya yaitu merah, kuning, dan hijau. Prototype traffic light akan langsung dihubungkan pada port GPIO raspberry pi, yang nantinya penyalan traffic light akan di control secara otomatis melalui raspberry pi. Berikut merupakan gambar perancangan prototype traffic light :



Gambar 3 Perancangan Prototype Smart Traffic Light

D. Perancangan Prototype Secara Keseluruhan

Pada penelitian ini dilakukan perancangan prototype secara keseluruhan, perancangan prototype secara keseluruhan ini merupakan gabungan dari beberapa perancangan yang telah dijelaskan sebelumnya , seperti perancangan prototype pertigaan jalan raya, perancangan webcam pada sistem smart traffic light, perancangan prototype traffic light, dan perancangan tiang yang digunakan untuk tempat webcam beserta tempat prototype traffic light



Gambar 4 Hasil Perancangan Prototype Secara Keseluruhan

IV. HASIL DAN PEMBAHASAN

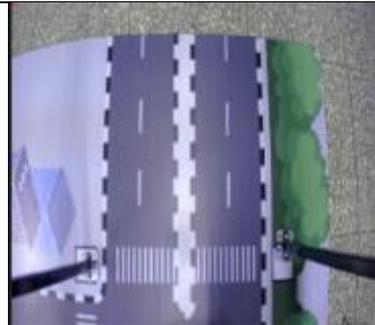
Pengujian sistem perangkat lunak (software). Pada bab ini dilakukan pengujian sistem untuk mengetahui dan menguji kehandalan dari sistem yang telah dibuat pada rancang bangun prototype ini. Adapun beberapa hasil pengujian yang dilakukan adalah sebagai berikut :

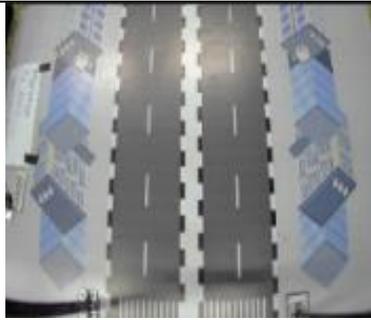
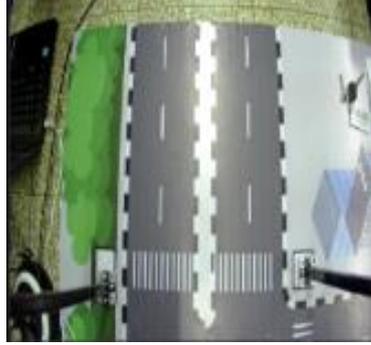
A. Pengujian Akses Kamera

Pada pengujian ini kamera menangkap gambar berupa citra RGB (citra berwarna) dengan menggunakan kamera webcam USB Inforce tipe TW-002 dengan memiliki resolusi gambar 1080p. Berikut pengujian akses kamera tiap jalur pada smart traffic light :

TABEL 1

DATA HASIL UJICOBA AKSES WEBCAM

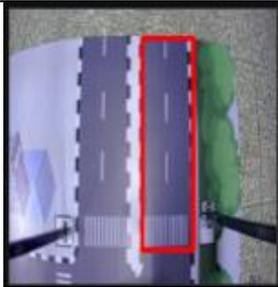
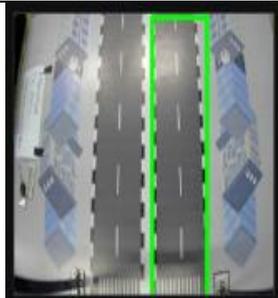
| No | Jalur | Citra RGB | Port Webcam |
|----|---------|--|-------------|
| 1 | Jalur 1 |  | 4 |

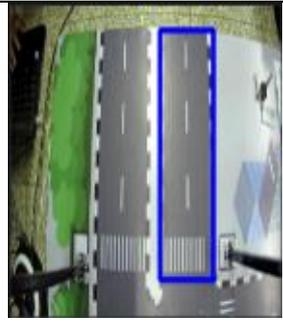
| | | | |
|---|---------|---|---|
| 2 | Jalur 2 |  | 2 |
| 3 | Jalur 3 |  | 0 |

Dari data tabel 1 merupakan hasil dari pengujian akses kamera webcam yang dilakukan pada setiap jalurnya. Dari pengujian tersebut bisa diketahui bahwa pengujian akses kamera webcam dapat dilakukan dengan baik

B. Pengujian koordinat ROI (Region Of Interest)

TABEL 2
DATA HASIL UJICOBA KOORDINAT ROI

| No | X1 | Y1 | X2 | Y2 | Hasil Implementasi pada Citra |
|----|-----|----|-----|-----|---|
| 1 | 148 | 27 | 205 | 218 |  |
| 2 | 158 | 4 | 221 | 251 |  |

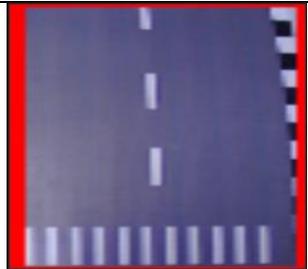
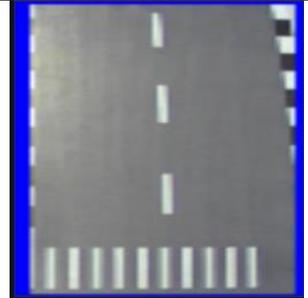
| | | | | | |
|---|-----|----|-----|-----|---|
| 3 | 164 | 20 | 223 | 219 |  |
|---|-----|----|-----|-----|---|

Tabel 2 merupakan data hasil implementasi koordinat ROI pada citra yang dihasilkan dari proses drag and drop. Dimana y1 dan x2 merupakan titik koordinat dimulainya drag area ROI, sedangkan untuk x1 dan y2 merupakan Panjang dan lebar dari area ROI. Setelah dilakukan penentuan koordinat ROI akan dilakukan set ROI guna untuk mengimplementasikan koordinat roi pada citra hasil capture dari webcam

C. Pengujian Capture Citra Background

TABEL 3

DATA HASIL PENGUJIAN CAPTURE CITRA BACKGROUND

| No | Jalur | Citra Background |
|----|---------|---|
| 1 | Jalur 1 |  |
| 2 | Jalur 2 |  |
| 3 | Jalur 3 |  |

Tabel 3 merupakan data hasil dari proses capture citra background. Pengambilan citra background dilakukan sebelum area pemrosesan terdapat sebuah objek sehingga hasil dari pengambilan citra background murni sebuah background tanpa terdapat objek. Hasil capture citra background berupa citra

jalan raya yang dimana hasil citra ini sesuai dengan ukuran area yang telah dilakukan proses ROI (Region Of Interest).

D. Pengujian Capture Citra Foreground

TABEL 4
DATA HASIL PENGUJIAN CAPTURE CITRA FOREGROUND

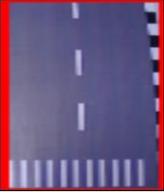
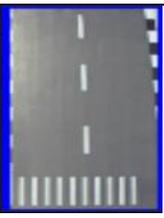
| No | Jalur | Citra Foreground |
|----|---------|---|
| 1 | Jalur 1 |  |
| 2 | Jalur 2 |  |
| 3 | Jalur 3 |  |

Tabel 4 merupakan hasil dari pengambilan citra foreground pada setiap jalur, dimana hasil dari pengambilan citra foreground sesuai dengan ukuran area pemrosesan ROI dan pada hasil pengambilan citra foreground terdapat sejumlah objek kendaraan yang akan dideteksi kamera pada proses selanjutnya

E. Pengujian Pengurangan pixel Citra background dengan Citra Foreground

TABEL 5
DATA HASIL PENGUJIAN PENGURANGAN PIXEL CITRA BACKGROUND DENGAN CITRA FOREGROUND

| Jalur | Citra background | Citra Foreground | Hasil |
|-------|------------------|------------------|-------|
| | | | |

| | | | |
|---|--|---|---|
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |

Tabel 5 merupakan hasil pengujian pengurangan antar pixel citra background dan citra foreground pada setiap jalur. Dari gambar diatas dapat dilihat bahwa citra background tidak terdapat objek dan citra foreground terdapat objek. Dari kedua citra tersebut akan dilakukan pengurangan antar pixel citra yang akan menghasilkan sebuah pemisahan antar objek kendaraan dengan background. Hasil pemisahan objek dan background berupa bentuk objek kendaraan yang nanti akan dilanjutkan pada tahap selanjutnya dengan proses Analisa bentuk objek.

F. Pengujian Nilai threshold, Dilasi dan Erosi

Pada pengujian ini dilakukan penentuan nilai threshold, dilasi dan erosi yang digunakan dalam pendeteksian objek kendaraan. Pada pengujian ini akan dilakukan pencarian nilai yang optimal dalam pendeteksian objek kendaraan untuk digunakan pada setiap jalurnya. Berikut pengujian yang akan disajikan dalam bentuk data tabel pada setiap jalurnya :

TABEL 6
DATA HASIL PENGUJIAN NILAI THRESHOLD, DILASI, DAN EROSI

| No | Jalur | Nilai Threshold | Nilai Dilasi | Nilai Erosi |
|----|-------|-----------------|--------------|-------------|
| 1 | 1 | 18 | 4 | 2 |
| 2 | 2 | 27 | 2 | 2 |
| 3 | 3 | 26 | 3 | 3 |

Dari data Tabel 6 dapat diketahui bahwa dari hasil pengujian dalam menentukan nilai threshold, dilasi, dan erosi didapatkan sebuah nilai yang akan digunakan pada setiap jalur. Dari data tabel tersebut dapat dilihat bahwa nilai yang dihasilkan dari pengujian tiap jalur tidak memiliki kesamaan, jadi pada setiap jalur memiliki konfigurasi nilai yang berbeda beda dalam melakukan proses pendeteksian

G. Pengujian Deteksi Objek dengan Tingkat Pencahayaan yang Berbeda

Pada pengujian ini dilakukan uji coba pendeteksian objek dengan kondisi pencahayaan yang berbeda-beda. Uji coba yang dilakukan menggunakan 3 parameter yaitu redup, terang, dan sangat terang. Berikut pengujian deteksi objek dengan tingkat pencahayaan yang berbeda beda pada setiap jalurnya :

TABEL 7

DATA HASIL PENGUJIAN DENGAN PENCAHAYAAN YANG REDUP

| No | Lux | Jalur | Objek | Terdeteksi |
|----|-----|-------|-------|------------|
| 1 | 10 | 1 | 4 | 3 |
| 2 | 10 | 2 | 4 | 3 |
| 3 | 10 | 3 | 4 | 1 |

Pada table 7 merupakan data hasil percobaan pengujian deteksi objek yang dilakukan dengan kondisi pencahayaan redup dengan nilai lux 10. Pada hasil pengujian tersebut dapat dilihat bahwa pendeteksian objek tidak akurat dikarenakan intensitas cahaya yang kurang sehingga hasil objek yang terdeteksi tidak sama dengan objek yang diujikan

TABEL 8

DATA HASIL PENGUJIAN DENGAN PENCAHAYAAN YANG TERANG

| No | Lux | Jalur | Objek | Terdeteksi |
|----|-----|-------|-------|------------|
| 1 | 27 | 1 | 4 | 4 |
| 2 | 27 | 2 | 4 | 4 |
| 3 | 27 | 3 | 4 | 4 |

Pada table 8 merupakan data hasil percobaan pengujian deteksi objek yang dilakukan dengan kondisi pencahayaan terang dengan nilai lux 27. Pada hasil pengujian tersebut dapat dilihat bahwa pendeteksian objek akurat dikarenakan intensitas cahaya yang cukup sehingga hasil objek yang terdeteksi sesuai dengan objek yang diujikan

TABEL 9

DATA HASIL PENGUJIAN DENGAN PENCAHAYAAN YANG SANGAT TERANG

| No | Lux | Jalur | Objek | Terdeteksi |
|----|-----|-------|-------|------------|
| 1 | 41 | 1 | 4 | 4 |
| 2 | 41 | 2 | 4 | 4 |
| 3 | 41 | 3 | 4 | 4 |

Pada tabel 9 merupakan data hasil percobaan pengujian deteksi objek yang dilakukan dengan kondisi pencahayaan sangat terang dengan nilai lux 41. Pada hasil pengujian tersebut dapat dilihat bahwa pendeteksian objek akurat dikarenakan intensitas cahaya yang teran sehingga hasil objek yang terdeteksi sesuai dengan objek yang diujikan.

Dari percobaan yang dilakukan diatas dapat disimpulkan bahwa dalam pendeteksian objek kendaraan dibutuhkan pencahayaan yang cukup. Dikarenakan apabila

dalam pendeteksian objek kendaraan dengan kondisi cahaya yang redup dapat menyebabkan perhitunga jumlah kendaraan yang tidak akurat

H. Pengujian Penentuan Delay Waktu Traffic Light

Pada pengujian ini dilakukan sebuah ujicoba data dari hasil pendeteksian jumlah kendaraan yang dimana nanti jumlah kendaraan akan dikonversikan dalam bentuk persentase. Pengujian ini dilakukan untuk membuktikan apakah sistem dapat menentukan jalur yang membutuhkan delay waktu tambahan atau tidak. Berikut pengujian penentuan delay waktu traffic light pada tiap tiap jalurnya :

Tabel 10. Penentuan Delay Traffic lisght

| No | Percobaan | Hasil Persentase | Delay Waktu |
|----|-----------|------------------|-------------|
| 1 | 100 % | 100% | 20 detik |
| 2 | 90 % | 90 % | 20 detik |
| 3 | 70 % | 70 % | 20 detik |
| 4 | 50 % | 50 % | 10 detik |
| 5 | 30 % | 30 % | 10 detik |

Tabel 10 merupakan data hasil pengujian penentuan waktu delay *traffic light* berdasarkan persentase kepadatan pada jalur persimpangan. Dari data tabel diatas dapat dilihat bahwa sistem telah berhasil menentukan delay waktu penyalaaan *traffic light* yang dibutuhkan pada jalur *traffic light*. Hasil tersebut didapatkan berdasarkan tingkat kepadatan kendaraan yang dilakukan dengan 5 kali percobaan menggunakan nilai parameter persentase tingkat kepadatan kendaraan yang berbeda-beda, sesuai yang telah ditentukan.

V. KESIMPULAN

Dari hasil pengujian dan pengambilan data yang telah dilakukan, maka dapat diambil sebuah kesimpulan sebagai berikut :

1. Telah terbangunnya sebuah *software user interface Smart Traffic Light* yang digunakan untuk menjalankan dan memonitoring sistem pengontrolan *traffic light* otomatis berdasarkan tingkat kepadatan kendaraan dengan menggunakan pengolahan citra digital.
2. Sistem dengan pencahayaan ruangan yang cukup terang, mampu mendeteksi objek kendaraan berdasarkan hasil pengurangan frame citra dengan menerapkan *background subtraction* yang lebih baik dalam mendeteksi objek kendaraan.
3. Sistem *traffic light* yang dibuat bersifat dinamis menentukan adanya penambahan waktu atau tidak pada setiap jalur dalam penyalaaan lampu hijau.
4. Pada persentase 100% sistem mendeteksi kepadatan jalur sehingga dilakukan penambahan waktu, Persentase 90% sistem mendeteksi kepadatan jalur sehingga dilakukan penambahan waktu. Persentase 70% sistem mendeteksi kepadatan jalur sehingga dilakukan penambahan waktu. Persentase 50% sistem tidak mendeteksi kepadatan jalur sehingga tidak dilakukan

penambahan waktu. Persentase 30% sistem tidak mendeteksi kepadatan jalur sehingga tidak dilakukan penambahan waktu.

REFERENSI

- [1] Daryanto, T. and Pratomo, P. T. (2010) 'Control System Berdasarkan Waktu Kantor (Studi Kasus Perempatan Tomang)', II(2), p. 2.
- [2] Lugianti, I., Sari, J. Y. and Ningrum, I. P. (2012) 'Deteksi Kecepatan Kendaraan Bergerak Berbasis Video Menggunakan Metode Frame Difference', Seminar Nasional, 1, pp. 324–332
- [3] Nugroho, E. A. (2017) 'Sistem Pengendali Lampu Lalulintas Berbasis Logika Fuzzy', Jurnal SIMETRIS, 8(1), pp. 75–84.
- [4] Rizal, A. M. (2020) 'Implementation of a Ball Tracking System In Image Processing Using the Find Contour Method Implementasi Sistem Tracking Bola Pada Pengolahan Citra Menggunakan Metode Find Contour', Journal of Electrical and Electronic Engineering-UMSIDA, 4(2), pp. 105–114.
- [5] Singla, N. (2014) 'Motion Detection Based on Frame Difference Method', International Journal of Information & Computation Technology, 4(15), pp. 1559–1565. Available at: http://www.ripublication.com/irph/ijict_spl/ijictv4n15spl_10.pdf.
- [6] Utama, R. P., Hariyani, Y. S. and Aulia, S. (2017) 'Rancang Bangun Lampu Lalu Lintas Otomatis Berdasarkan Panjang Antrian Kendaraan Berbasis Pengolahan Citra Digital'. & Computation Technology, 4(15), pp. 1559–1565. Available at: http://www.ripublication.com/irph/ijict_spl/ijictv4n15spl_10.pdf.
- [7] Utama, R. P., Hariyani, Y. S. and Aulia, S. (2017) 'Rancang Bangun Lampu Lalu Lintas Otomatis Berdasarkan Panjang Antrian Kendaraan Berbasis Pengolahan Citra Digital'.