

**PENGUKUR KECEPATAN KENDARAAN DI JALAN TOL DENGAN
METODE *BACKGROUND SUBTRACTION*
(Studi Kasus di Jalan Tol Jatingaleh Semarang)**

LAPORAN TUGAS AKHIR

Laporan ini disusun guna memenuhi salah satu syarat untuk menyelesaikan program studi Teknik Informatika S-1 pada Fakultas Teknologi Industri Universitas Islam Sultan Agung



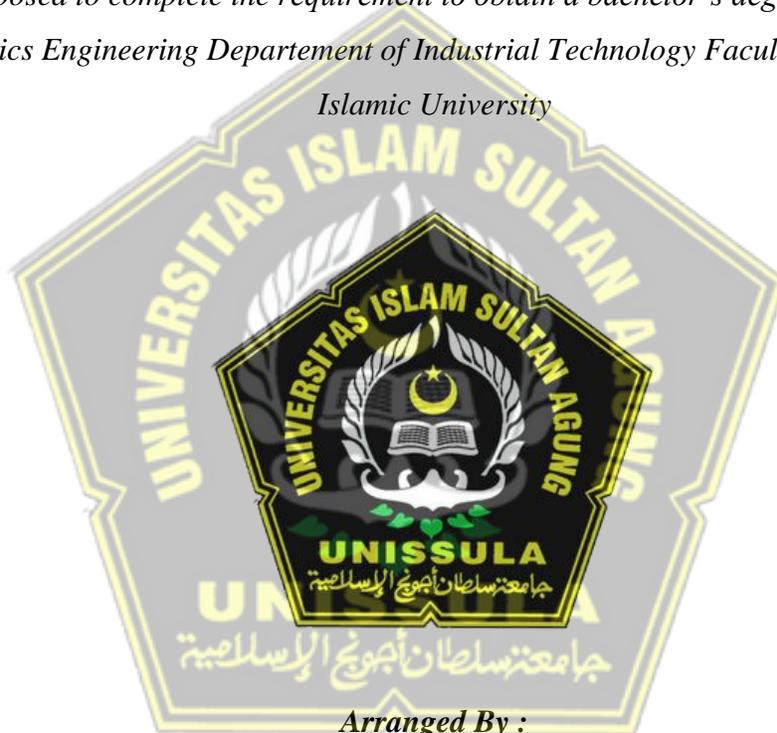
**DISUSUN OLEH :
GHOZALI TRI PUTRA WIRA UTAMA
NIM 32601601044**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM SULTAN AGUNG
SEMARANG
2023**

***MEASUREMENT OF VEHICLE SPEED ON TOLL ROADS WITH
BACKGROUND SUBTRACTION
(Case Study on Jatingaleh Semarang Toll Road)***

LAPORAN TUGAS AKHIR

*Proposed to complete the requirement to obtain a bachelor's degree (S-1) at
Informatics Engineering Departement of Industrial Technology Faculty Sultan Agung
Islamic University*



Arranged By :

GHOZALI TRI PUTRA WIRA UTAMA

NIM 32601601044

**MAJORING OF INFORMATICS ENGINEERING
INDUSTRIAL TECHNOLOGY FACULTY
SULTAN AGUNG ISLAMIC UNIVERSITY
SEMARANG**

2023

LEMBAR PENGESAHAN PEMBIMBING

LEMBAR PENGESAHAN PEMBIMBING

Laporan Tugas Akhir dengan judul “PENGUKUR KECEPATAN KENDARAAN DI JALAN TOL DENGAN METODE *BACKGROUND SUBTRACTION* (Studi Kasus di Jalan Tol Jatingaleh Semarang)” ini disusun oleh :

Nama : Ghozali Tri Putra Wira Utama

NIM : 32601601044

Program Studi : Teknik Informatika

Telah disahkan oleh dosen pembimbing pada :

Hari : Selasa

Tanggal : 8 Agustus 2023

Mengesahkan,

Pembimbing I

Ir. Sri Mulyono, M.Eng
NIDN. 0626066601

Pembimbing II

Bagus Satrio Waluyo Poetro, S.Kom., M.Cs
NIDN. 1027118801

Mengetahui,

Ketua Program Studi Teknik Informatika
Fakultas Teknologi Industri
Universitas Islam Sultan Agung

Ir. Sri Mulyono, M.Eng
NIDN. 0626066601

LEMBAR PENGESAHAN PENGUJI

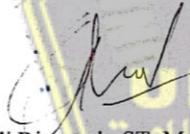
LEMBAR PENGESAHAN PENGUJI

Laporan Tugas Akhir dengan judul “PENGUKUR KECEPATAN KENDARAAN DI JALAN TOL DENGAN METODE BACKGROUND SUBTRACTION (Studi Kasus di Jalan Tol Jatingaleh Semarang)” ini telah dipertahankan di depan dosen penguji Tugas Akhir pada :

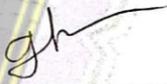
Hari : Selasa

Tanggal : 8 Agustus 2023

Ketua Penguji


Andi Riansyah, ST, M.Kom
NIDN. 0609108802

Anggota I


Ghufron, ST, M.Kom
NIDN.0602079005

SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

SURAT PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan dibawah ini :

Nama : Ghozali Tri Putra Wira Utama

NIM : 32601601044

Judul Tugas Akhir : **PENGUKUR KECEPATAN KENDARAAN DI JALAN TOL DENGAN METODE *BACKGROUND SUBTRACTION*** (Studi Kasus di Jalan Tol Jatingaleh Semarang) Dengan bahwa ini saya menyatakan bahwa judul dan isi Tugas Akhir yang saya buat dalam rangka menyelesaikan Pendidikan Strata Satu (S1) Teknik Informatika tersebut adalah asli dan belum pernah diangkat, ditulis ataupun dipublikasikan oleh siapapun baik keseluruhan maupun sebagian, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka, dan apabila di kemudian hari ternyata terbukti bahwa judul Tugas Akhir tersebut pernah diangkat, ditulis ataupun dipublikasikan, maka saya bersedia dikenakan sanksi akademis. Demikian surat pernyataan ini saya buat dengan sadar dan penuh tanggung jawab.

Semarang, 8 Agustus 2023

Yang Menyatakan,



Ghozali Tri Putra Wira Utama

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMIAH

Saya yang bertanda tangan dibawah ini :

Nama : Ghozali Tri Putra Wira Utama
NIM : 32601601044
Program Studi : Teknik Informatika
Fakultas : Teknologi industri
Alamat Asal : Jl. Lintang Trenggono V N6.67, Tlogosari Kulon,
Kec.Pedurungan, Kota Semarang

Dengan ini menyatakan Karya Ilmiah berupa Tugas akhir dengan Judul : **PENGUKUR KECEPATAN KENDARAAN DI JALAN TOL DENGAN METODE BACKGROUND SUBTRACTION (Studi Kasus di Jalan Tol Jatingaleh Semarang)** Menyetujui menjadi hak milik Universitas Islam Sultan Agung serta memberikan Hak bebas Royalti Non-Eksklusif untuk disimpan, dialihmediakan, dikelola dan pangkalan data dan dipublikasikan diinternet dan media lain untuk kepentingan akademis selama tetap menyantumkan nama penulis sebagai pemilik hak cipta. Pernyataan ini saya buat dengan sungguh-sungguh. Apabila dikemudian hari terbukti ada pelanggaran Hak Cipta/Plagiarisme dalam karya ilmiah ini, maka segala bentuk tuntutan hukum yang timbul akan saya tanggung secara pribadi tanpa melibatkan Universitas Islam Sultan agung.

Semarang, ..D...Agustus 2023

Yang menyatakan,



Ghozali Tri Putra Wira Utama

KATA PENGANTAR

Dengan mengucap syukur alhamdulillah atas kehadiran Allah SWT yang telah memberikan rahmat dan karunianya kepada penulis, sehingga dapat menyelesaikan Tugas Akhir dengan judul “Pengukur Kecepatan Kendaraan Di Jalan Tol Dengan Metode *Background Subtraction*” ini untuk memenuhi salah satu syarat menyelesaikan studi serta dalam rangka memperoleh gelar sarjana (S-1) pada Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Islam Sultan Agung Semarang. Tugas Akhir ini disusun dan dibuat dengan adanya bantuan dari berbagai pihak, materi maupun teknis, oleh karena itu saya selaku penulis mengucapkan terima kasih kepada :

1. Rektor UNISSULA Bapak Prof. Dr. H. Gunarto, SH., M.Hum yang mengizinkan penulis menimba ilmu di kampus ini.
2. Dekan Fakultas Teknologi Industri Ibu Dr. Novi Marlyana, ST., MT.
3. Dosen pembimbing I penulis Ir. Sri Mulyono M.Eng yang telah meluangkan waktu dan memberi ilmu.
4. Dosen pembimbing II penulis Bagus Satrio Waluyo Poetro, S.Kom., M.Cs yang telah memberikan banyak nasehat dan saran.
5. Orang tua penulis yang telah mengizinkan untuk menyelesaikan laporan ini.
6. Dan kepada semua pihak yang tidak dapat saya satu persatu.

Dengan segala kerendahan hati, penulis menyadari masih banyak terdapat banyak kekurangan-kekurangan dari segi kualitas atau kuantitas maupun dari ilmu pengetahuan dalam penyusunan laporan, sehingga penulis mengharapkan adanya saran dan kritikan yang bersifat membangun demi kesempurnaan laporan ini.

Semarang, 16 Agustus 2023



Ghozali Tri Putra Wira Utama

DAFTAR ISI

HALAMAN SAMPUL.....	i
HALAMAN SAMPUL.....	ii
LEMBAR PENGESAHAN PEMBIMBING	iii
LEMBAR PENGESAHAN PENGUJI.....	iv
SURAT PERNYATAAN KEASLIAN TUGAS AKHIR.....	v
KATA PENGANTAR.....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	x
DAFTAR TABEL	xi
ABSTRAK	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Pembatasan Masalah	3
1.4 Tujuan	3
1.5 Manfaat	3
1.6 Sistematika Penulisan	3
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI	5
2.1 Tinjauan Pustaka	5
2.2 Dasar Teori.....	5
2.2.1 <i>Computer Vision</i>	5
2.2.2 Object Detection.....	5
2.2.3 Open CV	6
2.2.4 PyCharm.....	7
2.2.5 Mean Square Error (MSE)	7
2.2.6 Confusion Matrix	8
BAB III METODE PENELITIAN	11
3.1 Bahan Penelitian.....	11

3.2 Alat Penelitian.....	11
3.3 Pengumpulan Data	12
3.4 Konsep Sistem.....	12
3.5 Flowchart Sistem.....	13
3.6 Metode	14
3.7 Perancangan <i>Database</i>	15
3.8 Instalasi Sistem	15
3.9 Perancangan Prototipe.....	16
BAB IV HASIL DAN ANALISIS PENELITIAN	19
4.1 Cara Kerja Sistem	19
4.1.1 Pengambilan Dataset.....	19
4.1.2 Penentuan Daerah ROI.....	19
4.1.3 Proses Pengolahan Citra.....	20
4.1.4 Proses Pendeteksian Objek	22
4.1.5 Proses Tracking Object	23
4.1.6 Proses Input Database	23
4.1.7 Pembuatan Tampilan.....	24
4.1.8 Tampilan Sistem Penghitung Kecepatan Kendaraan	24
4.1.9 Tampilan Webapp	25
4.2 Analisa Akurasi Sistem.....	26
4.2.1 Pengujian dan Analisa Pengaruh <i>Object Filter</i>	26
4.2.2 Pengujian dan Analisa Pengaruh <i>Distance</i>	29
4.2.3 Analisa Precision.....	30
4.2.4 Analisa Recall	30
4.2.5 Analisa Akurasi.....	30
4.2.6 Evaluasi Kinerja Sistem	31
BAB V KESIMPULAN DAN SARAN	32
5.1 Kesimpulan	32
5.2 Saran.....	32
DAFTAR PUSTAKA	34

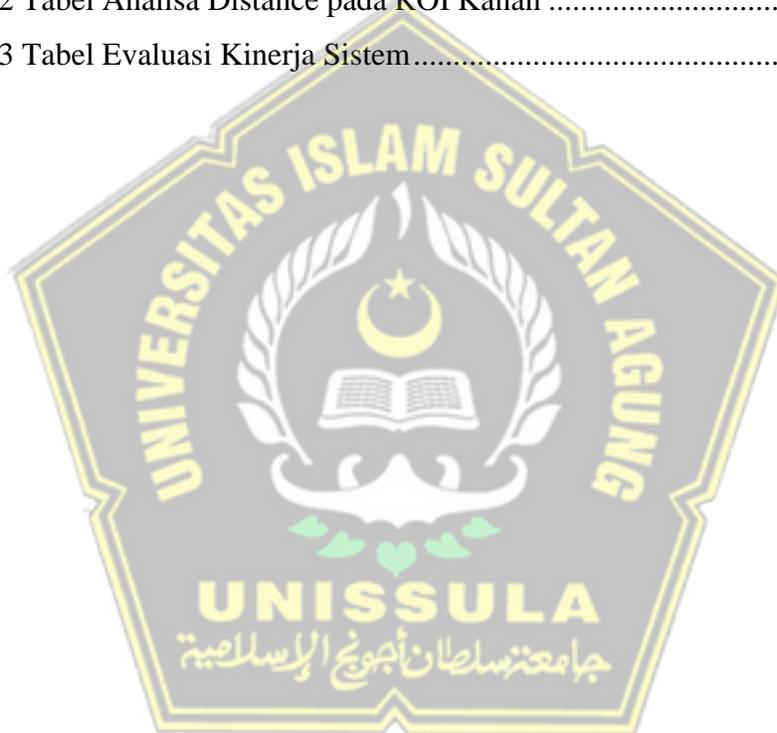
DAFTAR GAMBAR

Gambar 3. 1 Skenario Keseluruhan	12
Gambar 3. 2 Flowchart Sistem.....	13
Gambar 3. 3 Metode Sistem.....	14
Gambar 3. 4 Desain Webapp	17
Gambar 4. 1 Extract ROI	20
Gambar 4. 2 Pengolahan Citra Background Substraction.....	21
Gambar 4. 3 Pengolahan Citra Threshold.....	21
Gambar 4. 4 Pengolahan Citra Dilation.....	22
Gambar 4. 5 Output Original Frame ROI Kanan.....	24
Gambar 4. 6 Output ROI Kiri	25
Gambar 4. 7 Gambar Tampilan Webapp.....	25
Gambar 4. 8 Output Pendeteksian Benar.....	27
Gambar 4. 9 Output Pendeteksian Salah.....	28



DAFTAR TABEL

Tabel 2. 1 Tabel Mean Square Error	8
Tabel 2. 2 Tabel Confusion Matrix	8
Tabel 3. 1 Spesifikasi Komputer (Hardware)	11
Tabel 3. 2 Tabel Perancangan Database	15
Tabel 4. 1 Tabel Analisa Pengaruh Object Filter pada ROI Kanan	27
Tabel 4. 2 Tabel Analisa Distance pada ROI Kanan	29
Tabel 4. 3 Tabel Evaluasi Kinerja Sistem	31



ABSTRAK

Pertumbuhan jumlah penduduk di Indonesia berdampak pada jumlah kendaraan yang beroperasi di jalan tol sehingga menimbulkan kemacetan pada ruas-ruas jalan tol. Diperlukan adanya survey penghitungan jumlah kendaraan untuk mengetahui kepadatan pada ruas-ruas jalan tol, pelaksanaan penghitungan jumlah kendaraan masih dilakukan oleh manusia yang memungkinkan bisa terjadinya *human error*. Sistem penghitung kendaraan di jalan tol merupakan teknologi komputer untuk mendeteksi dan menghitung setiap kendaraan yang bisa diimplementasikan oleh sistem secara *real-time*. Penggunaan metode *Background Subtraction* diimplementasikan pada proses segmentasi dengan cara memisahkan antara objek dan latar belakang dengan *threshold* untuk mempertegas objek serta menghilangkan *shadow*. Dalam proses pendeteksian kendaraan dilakukan pengambilan *dataset* sebagai *input* sistem yang nantinya akan dilakukan pengujian dan menghasilkan *output* dari sistem berupa data id, arah, tanggal dan waktu yang tersimpan pada *database MySQL* dan ditampilkan pada *website*. Pengujian dilakukan dengan mendapatkan nilai *error* terendah. Evaluasi kinerja sistem mendapatkan hasil *precision* 100%, *recall* 49%, dan akurasi 53%. Metode ini kurang mendapatkan hasil yang optimal apabila diimplementasikan pada saat kondisi lalu lintas yang padat.

Kata Kunci: *Background Subtraction, Jalan Tol, Deteksi Objek, Object Filtering, Tracking*

ABSTRACT

The population growth in Indonesia is affecting the number of vehicles traveling on toll roads, causing traffic jams on toll roads. It is necessary to survey the number of vehicles to determine the density on the toll roads, the calculation of the number of vehicles is still done by humans, allowing for human error. The toll vehicle counting system is a computer technology to detect and count each vehicle that the system can do in real time. The use of background subtraction is done during segmentation by separating the subject and background by a threshold to highlight the subject and remove the shadow. During vehicle detection, data sets are taken as input to the system, which is then checked and outputs to the system in the form of stored identification, direction, and date data. in the MySQL database and displayed on the web page. Tests are performed by obtaining the lowest error value. The system performance assessment reached 100% accuracy, 49% recall and 53% accuracy. This method does not achieve optimal results when performed in heavy traffic conditions.

Keywords: *Background Subtraction, Toll roads, Object Detection, Object Filtering, Tracking*

BAB I

PEDAHULUAN

1.1 Latar Belakang

Dengan semakin bertambahnya jumlah kendaraan, maka dipastikan kondisi lalu lintas di jalan raya menjadi semakin padat. Kepadatan tersebut dapat berimbas pada kualitas jalan. Tingkat Pelayanan Jalan merupakan kualitas sebuah jalan dalam melayani arus lalu-lintas. Dalam Manual Kapasitas Jalan Raya (MKJI) dari Dirjen Bina Marga menyebutkan bahwa kecepatan dan derajat kejenuhan dapat menjadi indikator Tingkat Pelayanan Jalan tersebut. Derajat Kejenuhan sendiri merupakan perhitungan volume lalu lintas maksimum (Satuan mobil penumpang per jam) dibagi dengan kapasitas jalan (satuan mobil penumpang per jam). Menurut MKJI, derajat kejenuhan yang baik adalah kurang dari 0.75. (Harwendhani dan I.C, 2016),

Penghitungan jumlah kecepatan kendaraan yang melintas dapat dilakukan secara manual, yakni dengan mengirim petugas untuk menghitung jumlah kendaraan yang melintas. Penelitian ini melakukan analisa kualitas jalan Manado-Bitung dengan penghitungan manual selama 7 hari dengan interval 15 menit selama jam 06.00-21.00. Pencatatan manual tersebut dibantu peralatan berupa pita ukur, speed gun, stop watch, counter, dan formulir survei. Analisa kualitas jalan menggunakan pengamatan secara manual, yang mengamati jalan di pusat Kota Depok. Pengamatan dilakukan pada pagi dan sore hari yang merupakan jam padat lalu lintas. Pengamatan secara manual tentunya membutuhkan sumber daya manusia dan tidak efektif. (Adinugroho dkk, 2018)

Penghitungan kecepatan kendaraan berdasarkan video telah dilakukan dimana keduanya menggunakan frame-frame pada video untuk menghitung lama waktu yang dicapai oleh sebuah kendaraan dari satu tempat ke tempat lain. Keduanya menggunakan koordinat pixel yang ditentukan sebelumnya untuk

dijadikan titik mula dan titik akhir penghitungan waktu. Waktu dihitung sejak titik pusat dari blob kendaraan terdeteksi mencapai koordinat mula hingga mencapai koordinat akhir. Jarak sebenarnya antara kedua pixel ditentukan dengan rasio terhadap jarak tempuh sebenarnya. (Kurniawan dan W.R, 2015)

Sistem penghitung jumlah dan kecepatan kendaraan lain juga telah dikembangkan. Sensor yang digunakan adalah 3 buah sensor ultrasonik yang dipasang secara horizontal. Jika sensor 1 terhalangi maka counter akan diaktifkan untuk menunjukkan sebuah kendaraan tengah melintasi jalan. Waktu tempuh dihitung menggunakan timer antara sensor 2 dan 3 yang aktif saat terhalang kendaraan. Kecepatan kendaraan dihitung dengan membagi jarak tempuh yang tetap antara sensor 2 dan 3 dibagi waktu tempuh antara keduanya. (Fajar Mit Cahyana dkk, 2014)

Berdasarkan kajian terhadap penelitian diatas, maka penelitian ini membangun sistem *monitoring* jumlah kendaraan yang berbasis pemrosesan citra dengan menggunakan metode *Background Subtraction*. Dengan metode tersebut, kendaraan dengan mudah dideteksi tanpa terpengaruh pencahayaan luar ruangan yang beragam dibandingkan saat menggunakan metode segmentasi objek berdasarkan intensitas. Teknik ini dapat dilakukan dengan menggunakan webcam atau kamera eksternal sebagai alat untuk mengambil gambar kendaraan yang lewat dan aplikasi web sebagai alat pemantau data..

1.2 Perumusan Masalah

Rumusan masalah dalam penelitian ini adalah :

1. Bagaimana mengimplementasikan sebuah sistem penghitung kecepatan kendaraan dengan menggunakan metode *Background Subtraction*?
2. Bagaimana sistem dapat menampilkan hasil monitoring kedalam bentuk *webapp*?

1.3 Pembatasan Masalah

Pembatasan masalah ini untuk menciptakan kondisi yang menguntungkan, menghindari kegiatan-kegiatan yang di luar sasaran, sehingga dalam pembuatan laporan ini perlu dilakukan identifikasi batasan-batasan masalah. Keterbatasan permasalahan dalam rancangan akhir ini antara lain :

1. Pendeteksian dilakukan pada pagi, siang, sore dengan tingkat kecerahan yang optimum.
2. Menggunakan metode *Background Subtraction* yang disediakan oleh *OpenCV* versi 4.2.0.
3. Sistem hanya dapat menghitung kecepatan kendaraan yang melintasi jalan tol, di Jalan Tol Jatingaleh, Semarang.

1.4 Tujuan

Tujuan dari penelitian yang akan dilakukan adalah sebagai berikut :

1. Mengimplementasikan sistem pengukur kecepatan kendaraan otomatis dengan menerapkan metode *Background Subtraction*.
2. Monitoring kecepatan kendaraan yang melintasi jalan tol Jatingaleh Semarang.

1.5 Manfaat

Manfaat dari penelitian ini adalah adanya metode baru untuk melacak dan menghitung kecepatan kendaraan yang menggunakan jalan tol khususnya jalan tol Jatingaleh Semarang, serta mengurangi human error dalam menghitung kecepatan jumlah kendaraan yang kurang efisien.

1.6 Sistematika Penulisan

Sistematika penulisan yang akan digunakan oleh penyusun serta penulis dalam pembuatan laporan tugas akhir adalah sebagai berikut :

BAB 1 : PENDAHULUAN

Pada bab ini penulis mengungkapkan latar belakang dalam penetapan judul, rumusan masalah, pembatasan masalah, tujuan, serta sistematika penulisan.

BAB 2 : TINJAUAN PUSTAKA DAN DASAR TEORI

Bab ini berisi tentang teori dasar yang menjadi sumber atau alat untuk memahami permasalahan dalam perancangan sistem penghitungan kendaraan dua arah di jalan tol dengan menggunakan Metode Background Subtraction.

BAB 3 : METODE PENELITIAN

Bab ini menjelaskan proses tahapan penelitian yang dimulai dari analisis kebutuhan sistem, dilanjutkan dengan perancangan sistem hingga aktualisasi prototipe.

BAB 4 : HASIL DAN ANALISIS PENELITIAN

Bab ini menyajikan hasil penelitian khususnya hasil implementasi prototipe sistem untuk menghitung jumlah kendaraan dua arah di jalan tol dan menganalisis kinerja sistem.

BAB 5 : KESIMPULAN DAN SARAN

Bab ini diakhiri dengan uraian keseluruhan dari bab-bab sebelumnya serta rekomendasi yang diperoleh dan berguna untuk pengembangan selanjutnya.

BAB II

TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Citra adalah suatu representasi (gambaran), kemiripan atau imitasi dari suatu objek. Citra sebagai keluaran suatu sistem perekaman data dapat bersifat optic berupa foto, bersifat analog berupa sinyal-sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan (Sutoyo dan T.D, 2009). Pengolahan citra pada pustaka OpenCV (Open Computer Vision) telah mengembangkan beberapa metode yang dapat digunakan untuk menangani pengenalan objek bergerak.

2.2 Dasar Teori

2.2.1 *Computer Vision*

Computer Vision merupakan cabang ilmu pengetahuan yang mempelajari sebuah sistem komputer yang dapat mengenali objek yang diamati. Ilmu ini mempelajari bagaimana mesin mengekstrak informasi dari gambar dan menggunakannya untuk menyelesaikan tugas tertentu. Data gambar dapat diperoleh dalam berbagai bentuk, seperti gambar video atau gambar dari berbagai sudut. Dengan cabang ilmu ini, ilmu tersebut dapat dikembangkan lebih lanjut untuk mendukung aktivitas manusia dan mungkin mengatasi permasalahan seperti berhitung kendaraan di jalan tol. (Purwanto dkk, 2015)

2.2.2 *Object Detection*

Object Detection atau deteksi objek adalah teknologi komputer yang terkait dengan visi komputer dan pemrosesan gambar yang berhubungan dengan mendeteksi *instance* objek semantic dari kelas tertentu dalam gambar dan video

digital. Bidang deteksi objek yang diteliti dengan baik meliputi deteksi wajah, kendaraan dan pejalan kaki.

Tujuan dari *Object Detection* adalah untuk mendeteksi semua objek yang diketahui, seperti orang, mobil, atau wajah dalam sebuah gambar. Biasanya hanya ada beberapa kejadian di mana objek muncul dalam gambar, namun ada banyak kemungkinan lokasi dan skala serta cara mengeksplorasinya. Setiap deteksi objek dilaporkan dalam beberapa bentuk informasi, biasanya lokasi dan skala objek ditentukan dalam kotak pembatas.

Deteksi objek bertujuan untuk memisahkan objek baik (foreground) dan latar belakang atau (background). Biasanya, untuk melakukan deteksi objek, pola latar belakang akan dihasilkan pada gambar yang tersebar berdasarkan nilai piksel yang masuk seiring waktu. (Piccardi, 2004).

2.2.3 Open CV

OpenCV (Open Source Computer Vision) adalah perpustakaan visi komputer yang diprakarsai oleh Intel pada tahun 1999. Perpustakaan ini dirancang untuk digunakan pada berbagai platform, berfokus pada pemrosesan gambar real-time dengan berbagai algoritma visi komputer gratis untuk dapat mengatur ulang secara otomatis . gratis. OpenCV dirancang untuk digunakan dalam banyak bahasa pemrograman yang berbeda seperti C, C++, Python dan Android.

OpenCV juga menyediakan beberapa *template* perhitungan Algoritma *Object Detection*. Dengan ini para pengembang dapat memanfaatkan dan mengimplementasikan dalam pembuatan proyeknya dengan pustaka ini.

2.2.4 PyCharm

Pycharm adalah lingkungan pengembangan terintegrasi (IDE) yang digunakan untuk mengembangkan proyek menggunakan bahasa pemrograman Python. Python dikembangkan oleh JetBrains dan aplikasi Pycharm dapat berjalan di berbagai sistem operasi yaitu Windows, Linux dan Mac.

2.2.5 Mean Square Error (MSE)

Mean Squared Error (MSE) adalah metode pengujian yang digunakan untuk memeriksa nilai yang dihasilkan oleh sistem yang tercemar kesalahan dan nilai yang harus dicapai sistem, dengan memberikan skor kuantitatif. atau dengan error di antaranya, MSE dihitung menggunakan Persamaan 4. (Wang, 2009).

$$MSE(x,y) = \frac{1}{N} \sum_{i=1}^N [(x_i - y_i)]^2 \quad (4)$$

Keterangan:

N : Jumlah pengujian *dataset*

(x,y) : Perbedaan antara nilai x dan y pada *dataset*

i : Jumlah percobaan

x_i : Nilai yang dihasilkan oleh sistem yang tercemar dengan kesalahan

y_i : Nilai yang seharusnya dicapai oleh *system*

Dalam pengujian MSE nilai *error* dapat dikatakan baik apabila memiliki nilai error paling rendah.

Tabel 2. 1 Tabel *Mean Square Error*

		Nilai yang sebenarnya	
		<i>TRUE</i>	<i>FALSE</i>
Hasil Prediksi	<i>POSITIVE</i>	<i>TP (True Positive)</i>	<i>FP (False Positive)</i>
	<i>NEGATIVE</i>	<i>TN (True Negative)</i>	<i>FN (False Negative)</i>

N : Jumlah pengujian dataset

(x,y) : Perbedaan antara nilai x dan y pada dataset

i : Jumlah percobaan

x_i : Nilai yang dihasilkan oleh sistem yang tercemar dengan kesalahan

y_i : Nilai yang seharusnya dicapai oleh sistem

Dalam pengujian MSE nilai error dapat dikatakan baik apabila memiliki nilai error paling rendah.

2.2.6 Confusion Matrix

Confusion Matrix adalah metode eksperimental yang digunakan untuk menggambarkan penilaian pengelompokan. Berbagai metode pengujian matriks konfusi seperti presisi, recall, dan presisi. Tabel 3.2 mengilustrasikan *Confusion Matrix* (Adinugroho dkk, 2018)

Tabel 2. 2 Tabel *Confusion Matrix*

Keterangan :

<i>TP</i>	Nilai yang seharusnya benar dan Nilai yang dihasilkan sistem benar
<i>TN</i>	Nilai yang seharusnya benar dan Nilai yang dihasilkan sistem salah
<i>FP</i>	Nilai yang seharusnya salah dan Nilai yang dihasilkan sistem benar
<i>FN</i>	Nilai yang seharusnya salah dan Nilai yang dihasilkan sistem salah

Pada penelitian ini recall digunakan untuk menghitung proporsi prediksi yang benar tanpa mempertimbangkan yang positif dan yang negative, melainkan total keseluruhan

a. *Precision*

Precision dapat ditentukan dari jumlah total nilai yang seharusnya benar dan nilai yang dihasilkan sistem benar dibagi dengan total nilai yang dihasilkan sistem benar. Nilai presisi tertinggi jika nilai yang dihasilkan 100% dan Nilai presisi terendah jika nilai yang dihasilkan 0% . Periksa akurasi yang dihitung menggunakan persamaan 5 (Pratomo dkk, 2020).

$$\text{Precision} = \frac{TP}{(TP+FP)} \times 100\% \quad (5)$$

Keterangan:

TP: Nilai yang seharusnya benar dan Nilai yang dihasilkan sistem benar

FP : Nilai yang seharusnya salah dan Nilai yang dihasilkan sistem benar

Pengujian *precision* digunakan untuk menghitung presentase nilai yang dihasilkan benar dari total keseluruhan nilai prediksi benar

b. *Recall*

Recall ditentukan dari jumlah total nilai benar yang diharapkan dan nilai benar yang dihasilkan sistem dibagi dengan positif benar ditambah negatif palsu. Nilai recovery tertinggi jika nilai hasil 100% dan Nilai recovery terendah jika nilai hasil 0%. Tes recall dihitung menggunakan persamaan 6 (Pratomo & dkk, 2020)

$$\text{Recall} = \frac{TP}{(TP+FN)} \times 100\% \quad (6)$$

Keterangan:

TP: Nilai yang seharusnya benar dan Nilai yang dihasilkan sistem benar

FN: Nilai yang seharusnya salah dan Nilai yang dihasilkan sistem salah

Pengujian recall digunakan untuk menghitung presentase dari total kendaraan terdeteksi yang memiliki nilai positif.

c. Akurasi

Akurasi ditentukan dari nilai yang harus dibagi seluruh data secara tepat. Nilai eksak tertinggi adalah bila nilai eksak yang diperoleh sama dengan 100% dan nilai eksak terendah adalah ketika nilai akurasi yang diperoleh sebesar 0%. Akurasi dihitung menggunakan persamaan 7 (Pratomo dkk., 2020)

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FN+FP} \times 100\% \quad (7)$$

Keterangan:

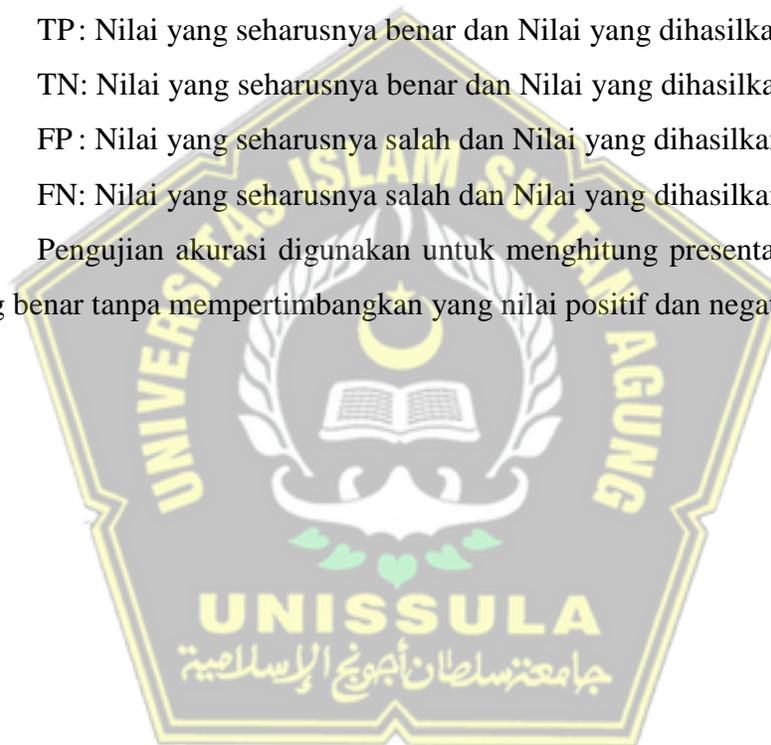
TP: Nilai yang seharusnya benar dan Nilai yang dihasilkan sistem benar

TN: Nilai yang seharusnya benar dan Nilai yang dihasilkan sistem salah

FP: Nilai yang seharusnya salah dan Nilai yang dihasilkan sistem benar

FN: Nilai yang seharusnya salah dan Nilai yang dihasilkan sistem salah

Pengujian akurasi digunakan untuk menghitung presentase nilai prediksi yang benar tanpa mempertimbangkan yang nilai positif dan negative.



BAB III METODE PENELITIAN

3.1 Bahan Penelitian

Makalah penelitian untuk penelitian ini adalah dataset video Jalan Tol Jatingaleh, Semarang. Terdapat sampel dataset video berdurasi 1 menit untuk menguji akurasi dan presisi pendeteksian, yakni dataset yang ditangkap kamera dengan dimmer, kondisi pencahayaan cukup, diambil pada pagi, siang, dan malam hari.

3.2 Alat Penelitian

Alat yang dibutuhkan dalam penelitian ini adalah berupa perangkat keras dan perangkat lunak dengan spesifikasi sebagai berikut :

1. Perangkat Keras
 - a. Komputer

Tabel 3. 1 Spesifikasi Komputer (*Hardware*)

Perangkat Keras	Spesifikasi
<i>Processor</i>	Intel® Core™ i5-9400F CPU @2.90GHz (6 CPUs) 2.90 GHz
<i>Memory</i>	RAM 16 GB
SSD	128 GB
VGA	NVIDIA GeForce GTX 1660 Ti
Monitor	LG 22 inchi

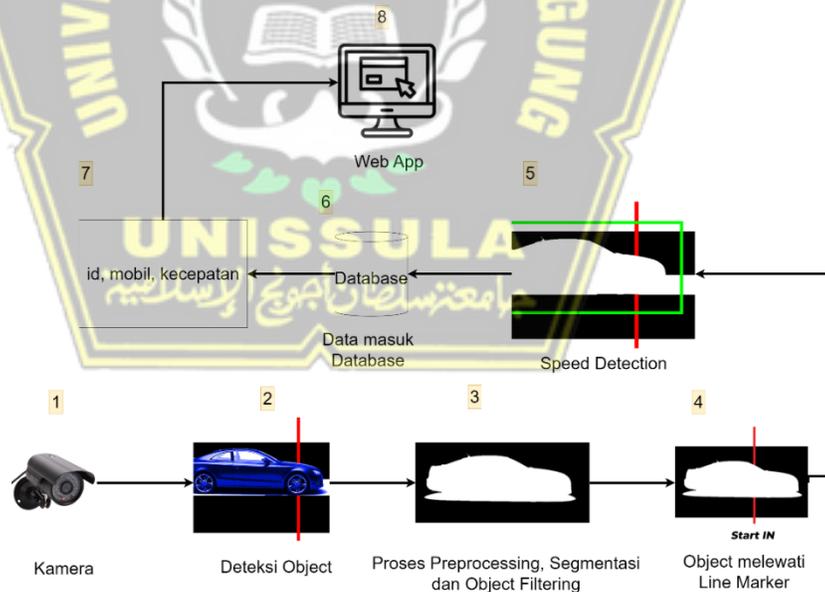
- b. Kamera / *Webcam*
2. Perangkat Lunak
 - a. Database server *MySql*
 - b. *Open Cv*

3.3 Pengumpulan Data

Pengumpulan data yang dilakukan peneliti untuk penyusunan tugas akhir adalah:

1. Video kendaraan melintasi Tol Jatingaleh Semarang. Data inilah yang akan dijadikan sampel dalam uji penghitungan dan pendeteksian kecepatan kendaraan di Tol Jatingaleh Semarang..
2. Tinjauan pustaka digunakan untuk mencari landasan teori sebagai acuan dalam penyusunan laporan penelitian yang dapat membantu peneliti membangun prototipe terkait pengurangan latar belakang. Sumber yang digunakan untuk dokumentasi adalah jurnal, artikel ilmiah, dan website untuk memudahkan peneliti dalam membuat laporan dan membuat prototipe.

3.4 Konsep Sistem

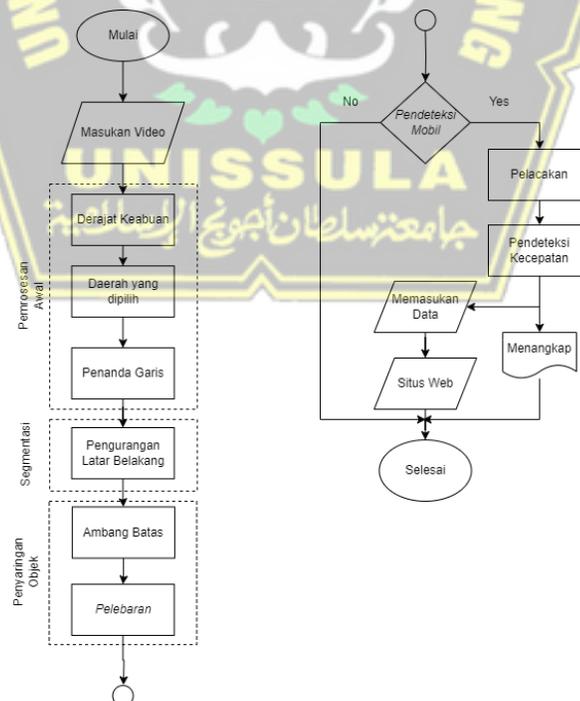


Gambar 3. 1 Skenario Keseluruhan

Gambar 3.1 menunjukkan gambaran konsep sistem keseluruhan dari sistem penghitung kecepatan kendaraan di jalan tol. Berikut keterangan Gambar 3.1 :

1. *Dataset Video* berfungsi sebagai data uji proses sistem penghitung kecepatan kendaraan.
2. Proses *preprocessing* untuk merubah citra RGB ke *Greyscale*, menentukan area *Region of Interest* dan membuat *Line Marker*.
3. Proses segmentasi dan *object filtering* untuk memisahkan objek dengan latar belakang serta mendapatkan hasil pendeteksian yang maksimal.
4. Proses *Object Tracking* untuk mengikuti setiap objek yang terdeteksi.
5. Proses *Speed Detection* untuk mengetahui kecepatan kendaraan, objek hasil pengolahan citra ditandai dengan *bounding box* atau garis kotak yang mengelilingi objek.
6. Proses penyimpanan data yang telah dihitung kecepatannya kedalam *database*.
7. Tampilan akhir webapp statistik kecepatan hasil pengolahan data dari database.

3.5 Flowchart Sistem

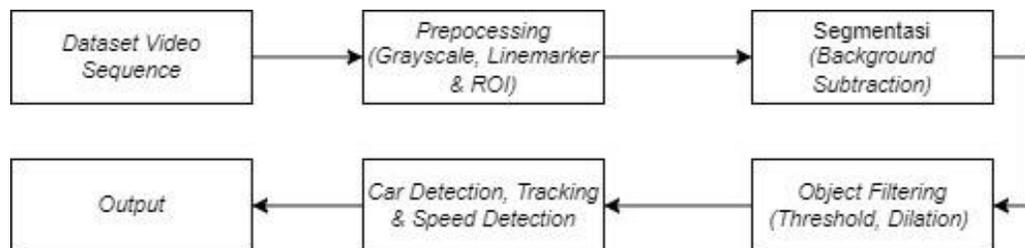


Gambar 3. 2 Flowchart Sistem

Pada Gambar 3.2 *flowchart* skema yang diuraikan di atas untuk mendapatkan bagian gambar yang diinginkan sebagai objek deteksi, mula-mula pada tahap input dataset video/sequence, kemudian diolah pada tahap preprocessing yaitu tahap awal. video gambar dengan tujuan membuat gambar skala abu-abu dan menentukan wilayah ROI serta menghasilkan sub-garis yang akan digunakan untuk menghitung laju. Kemudian tahap segmentasi adalah pemisahan antara subgambar yang mengandung atau tidak mengandung objek dengan cara pengurangan background. Kemudian jika objek yang diinginkan sudah terdeteksi, langkah selanjutnya adalah melakukan filtering objek untuk mengurangi noise dan mempertebal gambar. Proses pelacakan dan deteksi kecepatan kemudian bertujuan untuk melacak deteksi objek dan menghitung kecepatan kendaraan yang terdeteksi. Semua data tentang kendaraan yang terdeteksi kecepataannya akan dimasukkan ke dalam database untuk keperluan aplikasi web.

3.6 Metode

Metode yang digunakan dalam penelitian ini adalah pengurangan latar belakang. Dataset yang digunakan berupa video yang diambil dengan kamera/webcam. Urutan proses metode dapat dilihat pada Gambar 3.3 dibawah ini:



Gambar 3. 3 Metode Sistem

Gambar 3.3 adalah proses pengurangan latar belakang yang diterapkan dalam sistem. Langkah pertama adalah mengambil dataset video yang ditangkap kamera, kemudian melakukan greyscale untuk membantu memisahkan latar belakang dan latar depan. Kemudian lakukan pemisahan latar belakang dan latar depan menggunakan metode pengurangan latar belakang. Pada tahap pemfilteran objek melibatkan pengurangan noise dan penebalan objek sebelum melakukan deteksi dan pelacakan objek. Output dari metode ini berupa data akan diimpor ke database MySQL, yang kemudian akan diproses sesuai kebutuhan website..

3.7 Perancangan Database

Untuk membuat prototype ini, hal selanjutnya yang harus dibuat adalah database. Basis data dirancang untuk membantu menjalankan fungsi-fungsi pada sistem, yaitu menyimpan data-data yang berkaitan dengan sistem. Perancangan prototype database sistem pendeteksi kecepatan kendaraan adalah :

Tabel 3. 2 Tabel Perancangan Database

Field	Tipe	Keterangan
Id	Int(11)	Primary Key, AUTO INCREMENT
Kecepatan	Int(11)	Digunakan untuk hasil pendeteksian kecepatan

Pada Tabel 3.2 adalah tabel pendeteksian kecepatan kendaraan yang berfungsi menampilkan jumlah kendaraan, Kecepatan kendaraan dan waktu saat pendeteksian dilakukan. Tabel terdiri dari id, mobil, kecepatan.

3.8 Instalasi Sistem

Untuk membangun *prototype* agar dapat berjalan dengan lancar, maka diperlukan untuk menginstall beberapa *library python* yaitu :

1. *OpenCV* v4.1.2

OpenCV (Open Source Computer Vision) merupakan pustaka *computer vision* yang bertujuan untuk pengolahan citra. Metode yang digunakan berdasarkan penggunaan *open source* library *OpenCV*.

2. *PyMySQL* v1.0.2

PyMysql merupakan library klient *MySQL python*. *MySQL* adalah sistem manajemen *database open source* terkemuka dengan manajemen basis data *multiuser* dan *multithreaded* system. Library *PyMySQL* digunakan untuk mengkoneksikan data dapat dikirimkan kedalam *database MySQL*. (Menegez & G, 2017).

3. *Numpy* v1.18.1

Numpy merupakan *library python* yang menyediakan objek *array* multidimensi, berbagai objek turunan dan operasi cepat pada *array*. *Numpy* digunakan untuk bekerja dengan data berbentuk *array*. (Wang & Bovik, 2009)

4. *Tkinter* v.0.1.0

Tkinter merupakan *library python* yang digunakan untuk membuat *GUI* atau antarmuka pada *python*.

3.9 Perancangan Prototipe

Tujuan dari perancangan *prototype* ini adalah untuk menyampaikan rancangan sistem yang akan dibangun melalui pendeteksi kecepatan kendaraan di Jalan Tol Jatingaleh Semarang. Dalam perancangan *prototype*, mempunyai antarmuka sebagai aplikasi web, serta desain *prototype* yang akan dihasilkan:

a. Perancangan Desain *Webapp*



BAB IV

HASIL DAN ANALISIS PENELITIAN

4.1 Cara Kerja Sistem

Pada tahap cara kerja sistem akan di jelaskan bagaimana langkah-langkah kerja sistem.

4.1.1 Pengambilan Dataset

Pada tahap ini, kumpulan data dikumpulkan dari Jalan Tol Jatingaleh Semarang. Dimana dataset yang digunakan berdurasi 1 menit dengan resolusi 1920x1080 piksel. Pada saat perekaman dataset video ini diperlukan tripod agar gambar yang dihasilkan stabil. Sistem kemudian memutar ulang frame atau frame yang direkam kamera dan mengulangi atau mengulang frame terakhir.

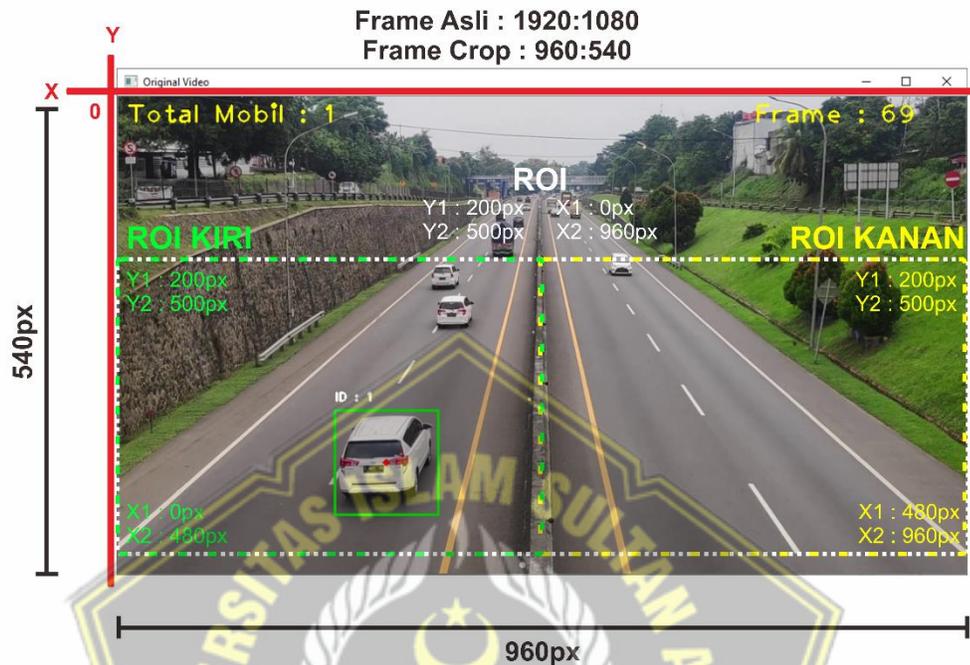
4.1.2 Penentuan Daerah ROI

Pada tahap sebelum menentukan ROI, karena dataset yang digunakan memiliki resolusi 1920 x 1080 piksel sedangkan resolusi layar kendali yang digunakan peneliti adalah 1366 x 768, maka dataset yang digunakan berskala 0,5 untuk masing-masing panjang dan lebar.

Proses ROI (*Region of Interest*) dalam penelitian ini digunakan untuk menentukan area yang akan di deteksi

```
# Extract Region of interest  
roi = frame[200: 500, 0: 960]  
roiKanan = frame[200: 500, 480: 960]
```

Koding diatas berfungsi untuk menentukan letak ROI (*Region of Interest*) yang ingin diseleksi



Gambar 4. 1 *Extract ROI*

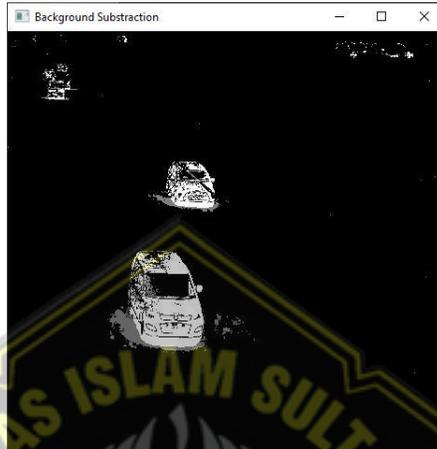
Pada gambar 4.1 merupakan proses *extract ROI* dari *frame original* untuk menghasilkan ROI Kanan dimana pada sistem monitoring kecepatan kendaraan diambil data mobil yang menghadap ke kamera.

4.1.3 Proses Pengolahan Citra

Area ROI telah ditentukan sebelumnya sehingga area pemrosesan gambar berfokus pada bingkai ROI dan bukan keseluruhan bingkai. Pengolahan citra merupakan ilmu yang mengolah citra untuk meningkatkan kualitas citra (Sayidul dkk, 2019) . Dalam implementasinya dilakukan beberapa langkah yang dikenal dengan teknik pengolahan citra (Ardhianto dkk, 2013).

Pengolahan citra yang digunakan pada peneliti terdapat 3 tahapan yaitu:

1. Tahapan Awal yaitu pemisahan objek dengan latar belakangnya dengan memanggil *library* dari *OpenCV* yaitu *Background Subtraction MOG2*.



Gambar 4. 2 Pengolahan Citra *Background Subtraction*

Pada Gambar 4.2 merupakan hasil dari pemisahan objek dengan latar belakangnya dengan menggunakan *Background Subtraction MOG2*.

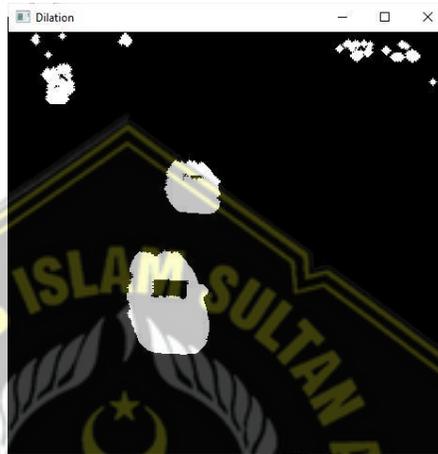
2. Tahapan selanjutnya yaitu penggunaan *Thresholding* (ambang batas), untuk memepertegas objek serta menghilangkan *shadow* atau warna keabu-abuan (hanya warna hitam dan putih) seperti pada Gambar 4.5



Gambar 4. 3 Pengolahan Citra *Threshold*

Pada gambar 4.3 merupakan hasil dari penggunaan *Thresholding* pada tahapan *object filtering*.

3. Tahap terakhir yaitu penggunaan *Dilation (Morphology)* untuk penebalan objek yang terdeteksi supaya mendapatkan *masking* yang maksimal seperti pada Gambar 4.4



Gambar 4. 4 Pengolahan Citra *Dilation*

Pada gambar 4.4 merupakan hasil dari penggunaan *dilation* pada tahapan *object filtering*.

4.1.4 Proses Pendeteksian Objek

Deteksi objek dilakukan dengan memanggil library dari OpenCV yaitu *Contouring*. Fungsi Kontur mendeteksi dan mengenali objek dengan menggabungkan titik-titik yang terbentuk pada sebuah piksel. Dalam penelitian ini, batas ditetapkan ke >7000 piksel untuk mengabaikan titik-titik kecil yang mungkin mengganggu pendeteksian objek. Objek yang terdeteksi ditandai dengan kotak pembatas dengan posisi kotak pembatas tengah, koordinat “X” dan “Y” dapat secara otomatis melacak pergerakan setiap objek.

4.1.5 Proses Tracking Object

Tracking adalah suatu teknik untuk melacak pergerakan suatu benda. Fungsi pelacakan dicapai dengan mendapatkan titik tengah objek sehingga dapat disimpan dalam array. Fungsi pelacakan diimplementasikan dengan memanggil kelas "*tracker.py*" dimana *class* ini berisi metode *update*, *catch*, dan *stop*.

Method update berguna untuk menghitung jarak / *distance* antara *center bounding box* dengan *new center bounding box* dan menambahkan id jumlah kendaraan tiap objek yang terdeteksi. Pada penelitian ini *method update* menggunakan jarak / *distance* antara *center bounding box* dengan *new center bounding box* dengan jarak ≤ 150 piksel.

```

same_object_detected = False

for id, pt in self.center_points.items():
    dist = math.hypot(cx - pt[0], cy - pt[1])
    if dist <= 150:
        self.center_points[id] = (cx, cy)
        objects_bbs_ids.append([x, y, w, h, id])
        same_object_detected = True

# New object is detected we assign the ID to that object
if same_object_detected is False:
    self.center_points[self.id_count] = (cx, cy)
    objects_bbs_ids.append([x, y, w, h, self.id_count])
    self.id_count += 1

```

4.1.6 Proses Input Database

Sampai disini objek-objek yang terdeteksi pada kotak pembatas semuanya mempunyai pengenalnya masing-masing setelah melalui proses tracking, kemudian menghitung atau menghitung jumlah kendaraan yang terdeteksi dan memasukkan datanya ke dalam database MySQL dimana query yang digunakan adalah INSERT INTO dengan Value(string) dengan perintah, jika $cx > 0$ dan

$cx \leq 480$ maka nilainya adalah "Keluar" dan jika $cx > 480$ dan $cx \leq 960$ maka nilai "Enter" lalu panggil metode kelas "mysql.py" konektor mysql di database pencarian.

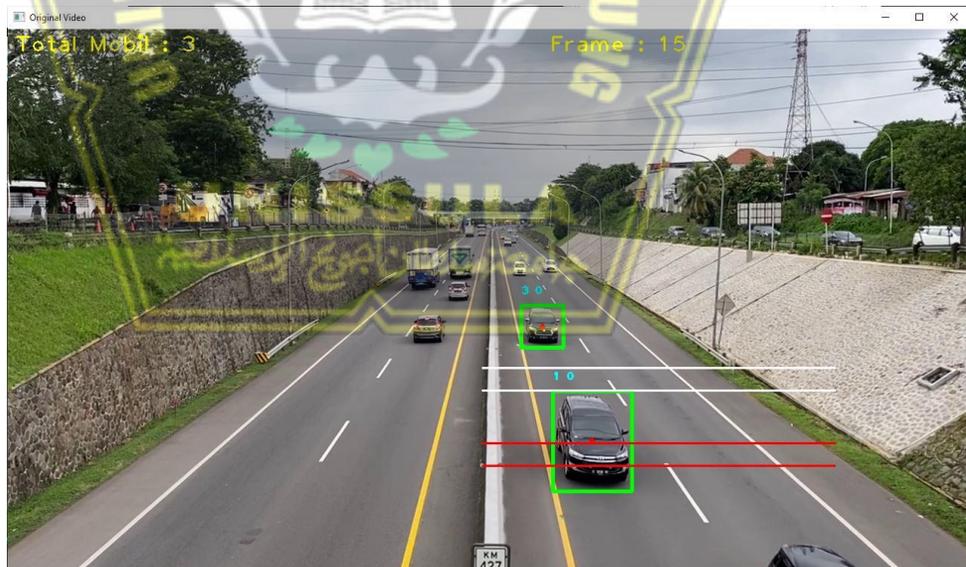
```

if (cx > 0 and cx <= 480):
    val = 'Keluar'
    sql = ("INSERT INTO cardetection (arah) VALUES (%s)")
    DB.insert(database, sql, val)
    print("Status : Keluar")
elif (cx > 480 and cx <= 960):
    val = 'Masuk'
    sql = ("INSERT INTO cardetection (arah) VALUES (%s)")
    DB.insert(database, sql, val)
    print("Status : Masuk")

```

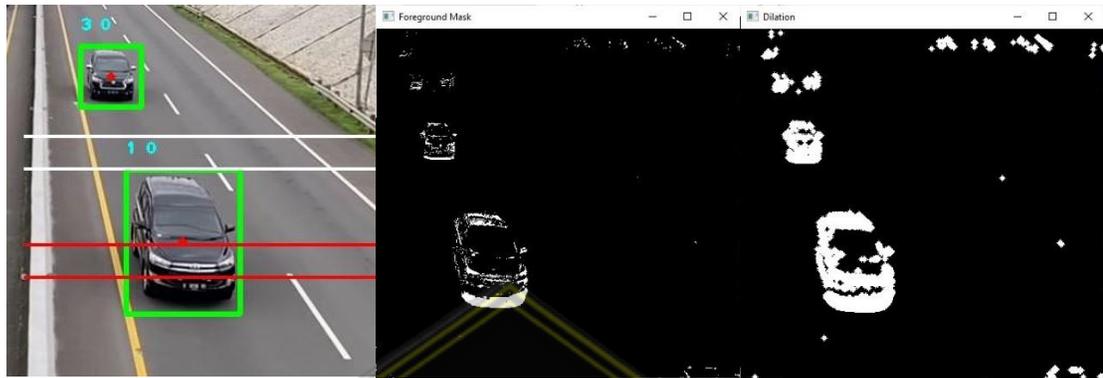
4.1.7 Pembuatan Tampilan

4.1.8 Tampilan Sistem Penghitung Kecepatan Kendaraan



Gambar 4. 5 Output Original Frame ROI Kanan

Pada Gambar 4.5 merupakan tampilan *original frame* pada sistem penghitung kendaraan di Jalan Tol Jatingaleh.



Gambar 4. 6 *Output ROI Kiri*

Pada Gambar 4.7 merupakan hasil dari ROI kanan, *foreground mask*, *dilation* dan pada sistem penghitung kendaraan menggunakan ROI kiri.

4.1.9 Tampilan Webapp



Gambar 4. 7 Gambar Tampilan *Webapp*

Pada Gambar 4.5 merupakan tampilan *webapp* sistem penghitung kecepatan kendaraan di jalan tol menggunakan *background subtraction*. Tampilan sistem dibuat menggunakan *library OpenCv* pada *python*. Pada

tampilan ini operator dapat *memonitoring* kecepatan kendaraan yang melewati jalan tol tersebut.

4.2 Analisa Akurasi Sistem

Kemudian sampai pada tahap analisis presisi yang tujuannya adalah untuk dapat mengetahui tingkat akurasi yang dikembangkan oleh sistem. Pada penelitian ini dilakukan pengujian dengan menggunakan kumpulan data video berdurasi satu menit dan diambil langsung dengan kamera ponsel serta menggunakan tripod untuk hasil gambar/video yang stabil.

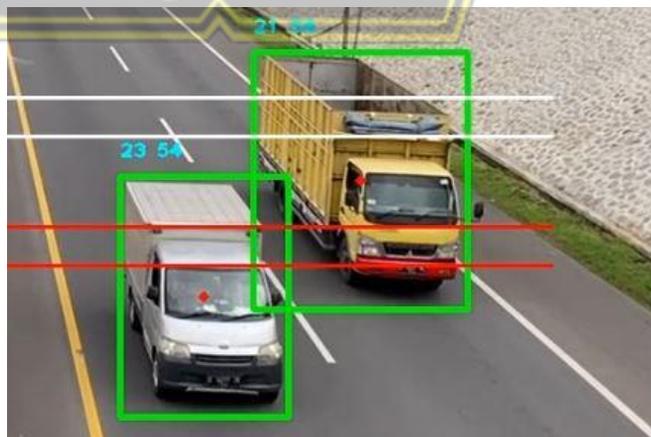
4.2.1 Pengujian dan Analisa Pengaruh *Object Filter*

Saat menguji dan menganalisis pengaruh pemfilteran objek, bahan uji yang digunakan adalah wilayah ROI yang sesuai dan dua parameter pengujian adalah inti dan loop. Parameter pengujian pertama untuk kernel menggunakan matriks kernel 3x3 dan matriks kernel 5x5, parameter pengujian kedua untuk iterasi menggunakan iterasi 1, 2 dan 3. Nilai ambang batas yang digunakan untuk pengujian ini adalah 50. Pengujian dan analisis dampak dari Filter Object bertujuan untuk menemukan matriks inti yang sesuai dengan matriks yang dapat diimplementasikan oleh sistem dan iterasi yang sesuai untuk kumpulan data yang tersedia. Pada pengujian ini perhitungan error dilakukan dengan menggunakan uji root mean square error seperti pada Persamaan 4. Untuk nilai hasil evaluasi pengujian pengaruh *object filter* dapat dilihat pada Tabel 4.1

Tabel 4. 1 Tabel Analisa Pengaruh *Object Filter* pada ROI Kanan

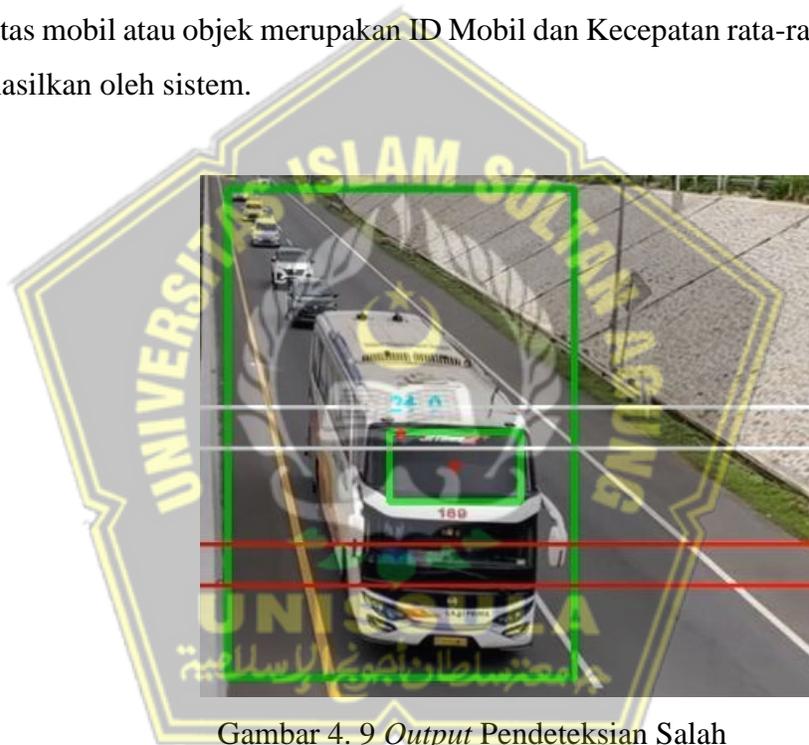
AROI Kanan / Mobil Datang					
Percobaan ke	Kernel	Dilation	Jumlah yang dihasilkan sistem	Jumlah yang sebenarnya	Mean Square Error
1	3x3	1	41	29	32,66
2	3x3	2	31	29	0,66
3	3x3	3	34	29	6
4	5x5	1	34	29	2,66
5	5x5	2	33	29	0
6	5x5	3	30	29	-0,16

Pada Tabel 4.1 merupakan hasil analisa pengaruh object filter pada ROI Kanan mendapatkan nilai error terendah pada pengujian ke-6 dengan nilai MSE=-0,16. Hasil uji impact objek filter ini didapat dari dua data uji ROI kiri dan ROI kanan, hasil error terendah adalah 0 dan -0,16 dari kedua data uji yang digunakan seperti pada baris tabel berwarna biru, maka errornya semakin besar. hasilnya adalah 20,16 dan 32,66 dari dua data uji yang digunakan seperti terlihat pada baris tabel merah, dengan nilai threshold 50.

Gambar 4. 8 *Output* Pendeteksian Benar

Pada Gambar 4.8 merupakan *output* pendeteksian benar dengan sistem yang dapat membaca objek berdekatan dengan jumlah objek yang sebenarnya. Dimana pada titik merah (*center point*) dan garis kotak hijau (*bounding box*) berfungsi sebagai kotak pembatas agar tidak melebihi objek yang akan dideteksi.

Lalu untuk garis putih berfungsi sebagai titik awal pendeteksian garis hitung kecepatan dan garis merah berfungsi sebagai titik akhir yang menghasilkan kecepatan rata-rata yang telah diproses pada sistem. Untuk angka diatas mobil atau objek merupakan ID Mobil dan Kecepatan rata-rata mobil yang dihasilkan oleh sistem.



Gambar 4. 9 *Output* Pendeteksian Salah

Pada Gambar 4.9 merupakan *output* pendeteksian salah dimana sistem dapat mampu membaca object berdekatan tetapi tidak sesuai dengan jumlah objek yang sebenarnya. Dimana pada kotak hijau (*bounding box*) dinyatakan salah karena mobil yang dideteksi melebihi satu objek.

4.2.2 Pengujian dan Analisa Pengaruh *Distance*

Pengujian dan analisa pengaruh *Distance* berguna untuk mengetahui pengaruh *distance* terhadap objek *tracking* yang terdeteksi sistem. Pada pengujian ini dilakukan evaluasi dengan jumlah *distance* 120, 130, 140, 150 dan 160 piksel. Nilai dari pengujian ini digunakan mendapatkan nilai evaluasi terkecil dan nilai evaluasi tertinggi. Dengan memanfaatkan parameter uji yang digunakan sebelumnya yang mendapatkan *error* terendah. Pengujian dan analisa pengaruh *distance* didapatkan nilai evaluasi *error* dan akurasi yang berbeda-beda. Untuk nilai hasil evaluasi pengujian pengaruh *distance* dapat dilihat pada Tabel 4.2

Tabel 4. 2 Tabel Analisa *Distance* pada ROI Kanan

Pengujian dan Analisa <i>Distance</i> Terhadap ROI Kanan				
Percobaan ke	Setting <i>Distance</i>	Jumlah yang dihasilkan sistem	Jumlah yang sebenarnya	MSE
1	120piksel	31	29	0,8
2	130piksel	31	29	0,8
3	140piksel	29	29	0
4	150piksel	28	29	-0,2
5	160piksel	28	29	-0,2

Pada Tabel 4.5 merupakan hasil Analisa *distance* pada ROI Kanan mendapatkan nilai *error* terendah pada pengujian ke 4 dengan nilai MSE=-0,2.

Hasil pengujian *distance* ini diambil dari dua data uji yaitu ROI Kiri dan ROI Kanan, hasil dari error terendah 0 dan -0,2 dari kedua data uji yang digunakan seperti pada baris tabel berwarna hijau dan hasil *error* terbesar 0,8 dari kedua data uji yang digunakan seperti pada baris tabel berwarna merah.

4.2.3 Analisa Precision

Perhitungan precision sistem penghitung jumlah kendaraan dua arah menggunakan metode *background subtraction*:

Precision ROI Kanan

$$\text{Precision ROI Kanan} = \frac{22}{22+0} \times 100\%$$

$$\text{Precision ROI Kanan} = 100\%$$

Dari hasil perhitungan Precision ROI Kanan menggunakan dataset yang ada didapatkan hasil 100%.

4.2.4 Analisa Recall

Perhitungan *recall* sistem penghitung jumlah kendaraan dua arah menggunakan metode *background subtraction*:

Recall ROI Kanan

$$\text{Recall ROI Kanan} = \frac{22}{22+23} \times 100\%$$

$$\text{Recall ROI Kanan} = 49\%$$

Dari hasil perhitungan *Recall* ROI Kanan menggunakan dataset yang ada didapatkan hasil 49%.

4.2.5 Analisa Akurasi

Perhitungan Akurasi sistem penghitung jumlah kendaraan dua arah menggunakan metode *background subtraction*:

Akurasi ROI Kanan

$$\text{Akurasi ROI Kanan} = \frac{22+4}{22+4+0+23} \times 100\%$$

$$\text{Akurasi ROI Kanan} = 53\%$$

Dari hasil perhitungan Akurasi ROI Kanan menggunakan dataset yang ada didapatkan hasil 53%.

4.2.6 Evaluasi Kinerja Sistem

Pengujian evaluasi kinerja sistem digunakan untuk mengevaluasi kinerja *precision*, *recall* dan akurasi pada sistem penghitung jumlah kendaraan dua arah menggunakan *Background Substraction* yang dihitung berdasarkan empat nilai yaitu *True Positive* (TP), *False Positive* (FP), *True Negative* (TN) dan *False Negative* (FN).

Dari hasil pengujian dengan dataset video berdurasi 1 menit diperoleh tingkat probabilitas seperti pada Tabel 4.3.

Tabel 4. 3 Tabel Evaluasi Kinerja Sistem

Jenis Kinjera	ROI	Presentase
<i>Precision</i>	ROI Kanan	100%
<i>Recall</i>	ROI Kanan	49%
Akurasi	ROI Kanan	53%

Pada Tabel 4.3 merupakan hasil pengujian evaluasi kinerja sistem diketahui bahwa *precision* sebesar 100%, *recall* sebesar 49% dan akurasi sebesar 53%. Nilai *precision*, *recall* dan akurasi dapat dimaksimalkan lagi dengan cara menambahkan pengenalan terhadap objek yang akan dideteksi dan parameter uji.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan perancangan sistem penghitung kecepatan kendaraan dengan metode background subtraction di jalan tol Jatingaleh Semarang, penulis dapat menarik kesimpulan sebagai berikut:

1. Pendeteksi dan perhitungan kecepatan kendaraan dapat dilakukan menggunakan metode algoritma *Background Subtraction* dengan memanfaatkan objek bergerak yang terdapat pada citra dengan beberapa tahapan untuk penyempurnaan sebuah citra dengan hasil yang maksimal. Kinerja sistem dengan menggunakan metode *Background Subtraction* pada kondisi lenggang lebih baik daripada kondisi padat atau ramai sebesar 53%. Kesalahan ini pada saat kondisi padat banyak disebabkan karena posisi kendaraan yang berdekatan, sehingga sistem membaca banyak kendaraan sebagai satu objek kendaraan dengan dimensi besar.
2. Sistem dapat memonitoring jumlah kendaraan yang melintasi Jalan Tol Jatingaleh Semarang dan menampilkan data berupa id dan kecepatan rata-rata yang ditampilkan di *Webapp*.

5.2 Saran

Dalam proses penerapan metode Background Subtraction pada sistem penghitungan kendaraan j, *prototype* ini masih mempunyai banyak kekurangan. Karena keterbatasan penulis dalam perancangan sistem. Jadi saya menemukan beberapa saran berguna untuk mengembangkan prototipe ini guna memperbaikinya.

1. Untuk pencarian lebih dalam, data pendekatan atau pelatihan akan ditambahkan pada objek yang akan dideteksi sehingga sistem dapat

mengetahui objek mana yang harus dideteksi dan menambahkan parameter untuk akurasi yang maksimal.

2. Sistem dapat diimplementasikan sebagai video dan digunakan secara real time dengan menambahkan webcam di titik-titik tol tertentu agar dapat dideteksi.



DAFTAR PUSTAKA

- Adinugroho, S., dan Sari, dan Y.A. (2018). Implementasi Data Mining Menggunakan Weka. Universitas Brawijaya Press.
- Aldhiyatika Amwin. (2021). *Algoritma You Only Look Once (Yolo)*.
- Ardhianto, dkk. (2013). Implementasi Metode *Image Subtracting* dan Metode *Regionprops* untuk Mendeteksi Jumlah Objek Berwarna RGB pada File Video. *Jurnal Teknologi Informasi Dinamik*, 18(2), 91–100.
- Budiyanto dan Elvira Sukma Wahyuni. (2020). Kecepatan Kendaraan Berbasis Pemrosesan Citra.
- Catur Rakhmad Handoko. (2018). Mesin Perekam Kecepatan Kendaraan Di Jalan Tol Berbasis *Raspberry Pi 3*.
- Daniel Nugraha. (2016). *Prototype* Sistem Pengawasan Pada Lampu Lalu Lintas di Persimpangan Jalan Menggunakan *Raspberry Pi* Untuk Mencatat Pelanggaran Lampu Merah.
- Fajar Mit Cahyana, dkk. (2014). Perancangan Program Penghitung Jumlah Kendaraan Di Lintasan Jalan Raya Satu Arah Menggunakan Bahasa Pemrograman C++ Dengan Pustaka *Opencv* Publikasi Jurnal Skripsi.
- Farah Fajriyah. (2016). Pengembangan Perangkat Lunak deteksi Kecepatan Kendaraanbergerak Berbasis Pengolahan Citradigital.
- Harwendhani, dan i.C. (2016). Sistem Pendeteksi Jumlah Mobil Dalam *Intelligent Transport System (ITS)* Menggunakan Metode Viola-Jones. 2(1), 279–288.
- Jendro Yuwono, M., Purnama, B., & Sthevanie, F. (2017). Penghitungan Kepadatan Kendaraan Di Jalan Tol Menggunakan Metode *Gaussian Mixture Model* dan Kalman Filter *Counting Density of Vehicle Based on Gaussian Mixture Model and Kalman Filter in Highway*.

- Kurniawan, dan W.R. (2015). Purwarupa sistem klasifikasi dan penghitung jumlah kendaraan bermotor menggunakan kamerawebcam berbasis citra digital. <http://etd.repository.ugm.ac.id/>
- Menegez, dan G. (2017). *Morphological Image Processing Introduction*.
- Piccardi, M. (2004). *Background subtraction techniques: a review* *.
- Prakoso, S. B., Pembimbing, D., Zaini, A., dan Kurniawan, S. A. (2018). Tugas Akhir-Tj 141502 Pengembangan Sistem Penghitung Jumlah Kendaraan Memanfaatkan Citra Aerial yang Diambil dengan Pesawat Tanpa Awak.
- Pratomo, A. H., Kaswidjanti, W., dan Mu'arifah, S. (2020). *Implementasi Algoritma Region Of Interest (Roi) Untuk Meningkatkan Performa Algoritma Deteksi Dan Klasifikasi Kendaraan Implementation Of Region Of Interest (ROI) Algorithm To Improve Car Detection and Classification Algorithm*. 7(1), 155–162. <https://doi.org/10.25126/jtiik.202071718>
- Purwanto, P., Dirgantoro Ir, B., dan Nugroho Jati, A. S. (2015). Implementasi *Face Identification Dan Face Recognition Pada Kamera Pengawas Sebagai Pendeteksi Bahaya Implementation Of Face Identification And Face Recognition On Security Camera As Threat Detector*.
- Sayidul, S., Augusta, A., Sari, Y. A., dan Adikara, P. P. (2019). Penentuan Jumlah Kendaraan Menggunakan *Blob Detection* dan *Background Subtraction* (Vol. 3, Issue 1). <http://j-ptiik.ub.ac.id>
- Sugandi, B., dan Lifitri, S. (2022). Deteksi Pelanggaran Lampu Lalu Lintas Berdasarkan *Sensor Visual*. JST (Jurnal Sains Dan Teknologi), 11(2), 315–323. <https://doi.org/10.23887/jstundiksha.v11i2.50287>
- Sutoyo, dan T.D. (2009). Teori Pengolahan Citra Digital. Andi.
- Taufiq, R. M., Sunanto, Rizki, Y., & Pratama, M. R. A. (2022). Simulasi Deteksi Golongan Kendaraan pada Gerbang Tol Menggunakan YOLOv4. *Jurnal CoSciTech (Computer Science and Information Technology)*, 3(2), 199–206. <https://doi.org/10.37859/coscitech.v3i2.3928>

- Tika Devi Widayanti. (2020). Identifikasi Objek Bawah Permukaan Untuk Fondasi Jalan Tol Di Jakarta Menggunakan Metode Ground Penetrating Radar (Gpr) Pada Segmen Area Y.
- Wang, Z., dan Bovik, A. C. (2009). *Mean squared error: Lot it or leave it? A new look at signal fidelity measures. IEEE Signal Processing Magazine, 26(1), 98–117. <https://doi.org/10.1109/MSP.2008.930649>*

