

PENERAPAN OPENCV DALAM RANCANG BANGUN PROTOTYPE APLIKASI PENGAWASAN PELANGGARAN LALU LINTAS

Tony Darmanto¹, Thommy Willay², Adi Andoyo³

¹²³Informatika, Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak
e-mail: ¹tony.darmanto@yahoo.com, ²w.thommy@gmail.com, ³akhiang32@gmail.com

Abstract

Traffic violations are still common in the society. Traffic violation occurred due to lack of consciousness and obedience to the Society for orderly traffic so it is considered ordinary and habitual if allowed to continue. One of the traffic violation that often occurs in society is the vehicle driver breaking or crossing the traffic light when the red light is on. To overcome this problem, an application is built that can detect traffic light violation using OpenCV. In designing this traffic violation monitoring application, the author uses descriptive research design, While the data collection method used were literature study which includes scientific books, Scientific papers, scientific journals, thesis, and published sources in various media. The system analysis technique used is object oriented technique with modeling Unified Modeling Language (UML). Application design technique used Python programming language and OpenCV library, as well as database design used MySQL. This research resulted in an application of traffic violation monitoring by using OpenCV. With this application, it is expected to reduce the rate of traffic light violations. Based on research conducted, it is concluded that the built-in application can detect violations that occur at the traffic light using the OpenCV library.

Keywords: Violation, Traffic Light, OpenCV

Abstrak

Pelanggaran lalu lintas masih sering terjadi di masyarakat. Pelanggaran lalu lintas terjadi akibat kurangnya kesadaran dan ketaatan masyarakat untuk tertib berlalu lintas sehingga dianggap hal biasa dan menjadi kebiasaan jika dibiarkan berlanjut. Salah satu jenis pelanggaran lalu lintas yang sering terjadi di masyarakat adalah pengguna kendaraan melanggar atau menerobos lampu lalu lintas saat kondisi lampu merah menyala. Untuk mengatasi masalah pelanggaran tersebut, maka dibangun suatu aplikasi yang dapat mendeteksi pelanggaran lampu lalu lintas dengan menggunakan OpenCV. Dalam perancangan aplikasi pengawasan pelanggaran lalu lintas ini, penulis menggunakan desain penelitian deskriptif, sedangkan metode pengumpulan data yang digunakan adalah studi literatur yang meliputi buku-buku ilmiah, karya ilmiah, jurnal ilmiah, skripsi, serta sumber-sumber yang dipublikasikan di berbagai media. Teknik analisis sistem yang digunakan adalah teknik berorientasi objek dengan permodelan *Unified Modeling Language* (UML). Teknik perancangan aplikasi menggunakan bahasa pemrograman Python dan pustaka OpenCV, serta perancangan basis data menggunakan MySQL. Penelitian ini menghasilkan suatu aplikasi pengawasan pelanggaran lalu lintas dengan menggunakan OpenCV. Dengan aplikasi ini, diharapkan dapat mengurangi tingkat pelanggaran lampu lalu lintas. Berdasarkan dari penelitian yang dilakukan, didapat kesimpulan bahwa aplikasi yang dibangun dapat mendeteksi pelanggaran yang terjadi pada lampu lalu lintas dengan menggunakan pustaka OpenCV

Kata Kunci: Pelanggaran, Lampu Lalu Lintas, OpenCV

1. PENDAHULUAN

Perkembangan teknologi informasi dari tahun ke tahun selalu mengalami perkembangan yang sangat pesat. Hasil perkembangan teknologi informasi tersebut menghasilkan berbagai macam teknologi yang memberikan banyaknya kemudahan-kemudahan yang berdampak pada kegiatan manusia. Peranan teknologi saat ini sudah mulai merambah ke berbagai bidang, salah satunya adalah pada bidang lalu lintas. Lalu lintas merupakan pergerakan manusia dengan atau tanpa disertai kendaraan dari satu tempat ke tempat lain dengan menggunakan jalan sebagai ruang geraknya. Kendaraan yang digunakan dapat dikategorikan dalam beberapa yaitu, mobil, sepeda motor, bus, truk, *pick up*, dan lain sebagainya. Dengan adanya kendaraan, maka akan memudahkan manusia untuk mencapai tempat tujuannya.

Setiap tahunnya jumlah kendaraan yang beredar semakin bertambah. Hal ini tampak membawa pengaruh terhadap kondisi keamanan lalu lintas. Pelanggaran lalu lintas merupakan salah satu penyebab kecelakaan yang

dapat mempengaruhi kondisi keamanan lalu lintas. Pelanggaran lalu lintas terjadi akibat kurangnya kesadaran dan ketaatan masyarakat untuk tertib berlalu lintas sehingga dianggap hal biasa dan menjadi kebiasaan jika dibiarkan berlanjut. Salah satu jenis pelanggaran lalu lintas yang sering terjadi adalah para pengguna kendaraan bermotor berhenti di atas marka *zebra cross* pada saat lampu lalu lintas berada pada kondisi lampu merah. Hal ini akan mengganggu kenyamanan pejalan kaki karena *zebra cross* merupakan tempat penyeberangan di jalan yang diperuntukkan khusus bagi pejalan kaki.

Pelanggaran lalu lintas yang juga sering terjadi adalah para pengguna kendaraan bermotor melanggar atau menerobos lampu lalu lintas saat kondisi lampu merah menyala. Pelanggaran ini terjadi karena ketidaksabaran pengguna kendaraan untuk menunggu lampu hijau menyala. Hal ini dapat menyebabkan kecelakaan lalu lintas yang membahayakan dirinya sendiri dan para pengguna jalan yang lain. Berdasarkan latar belakang penelitian di atas, penulis bermaksud untuk mengembangkan suatu aplikasi untuk mendeteksi pelanggaran lalu lintas secara visual menggunakan OpenCV. OpenCV adalah sebuah pustaka perangkat lunak yang ditujukan untuk pengolahan citra dan dapat dikombinasikan sehingga memiliki berbagai fungsi dalam pemrograman yang berkaitan dengan pengolahan citra digital. Dengan aplikasi ini, diharapkan dapat membantu kepolisian dalam menindaklanjuti pelanggaran lampu lalu lintas yang terjadi dan sekaligus menertibkan para pelanggar lalu lintas dan dapat meminimalisir adanya kecelakaan lalu lintas yang terjadi.

2. METODOLOGI PENELITIAN

2.1 Rancangan Penelitian, Metode Pengumpulan Data, Teknik Analisis Sistem, Teknik Perancangan Aplikasi.

2.1.1 Rancangan Penelitian

Rancangan penelitian yang digunakan penulis dalam menyusun penelitian ini adalah penelitian deskriptif, yaitu membuat deskripsi secara sistematis, faktual, dan akurat mengenai fakta-fakta yang terjadi pada objek penelitian.

2.1.2 Metode Pengumpulan Data

Metode pengumpulan data yang dilakukan adalah dengan mengumpulkan informasi dan data dari buku-buku ilmiah, karya ilmiah, jurnal ilmiah, penelitian, dan sumber-sumber tertulis yang dipublikasikan di berbagai media. Informasi dan data dapat berupa teori-teori yang mendasari masalah dan bidang yang akan diteliti oleh penulis.

2.1.3 Teknik Analisis Sistem

Teknik analisis sistem yang digunakan penulis dalam penelitian ini adalah teknik berorientasi objek dengan alat permodelan, yaitu *Unified Modeling Language* (UML) yang bertujuan untuk memberikan gambaran mengenai prosedur-prosedur yang terdapat pada perancangan aplikasi.

2.1.4 Teknik Perancangan Aplikasi

Teknik perancangan aplikasi yang digunakan dalam membangun aplikasi ini adalah menggunakan bahasa pemrograman Python dan OpenCV untuk merancang perangkat lunak, serta perancangan basis data menggunakan MySQL.

2.2 Landasan Teori

2.2.1 Citra Digital

Citra digital dibentuk oleh kumpulan titik yang dinamakan piksel (*pixel* atau '*picture element*'). Setiap piksel digambarkan sebagai satu kotak kecil. Setiap piksel mempunyai koordinat posisi^[1]. Citra digital adalah citra yang dapat diolah oleh komputer.^[2]

2.2.2 Computer Vision

Computer vision merupakan bidang pengetahuan yang berfokus pada bidang sistem kecerdasan buatan dan berhubungan dengan akuisisi dan pemrosesan citra^[3]. *Computer vision is a field of study that associates itself with processing, analyzing, and understanding images*^[4].

2.2.3 Open Source Computer Vision (OpenCV)

OpenCV adalah program sumber terbuka berbasis C++ yang saat ini banyak digunakan sebagai program computer vision. [3] OpenCV is an open source library for developing computer vision applications that can run on multiple platforms, such as Windows, Linux, Mac, Android, and iOS^[5].

2.2.4 Webcam

Web camera atau biasa disingkat webcam adalah kamera video digital kecil yang dihubungkan ke komputer melalui port USB atau serial^[6]. Kamera web atau Webcam adalah kamera yang mampu menangkap data berupa gambar dan suara. Webcam adalah sebuah kamera video digital kecil yang dihubungkan ke komputer melalui port USB ataupun port COM^[7].

2.2.5 Background Subtraction

Background subtraction adalah proses untuk mendeteksi pergerakan atau perbedaan signifikan yang terjadi di dalam *frame video* ketika dibandingkan dengan citra referensi. Tujuan dari *background subtraction* adalah untuk memisahkan obyek dan *background* sehingga gerakan dari sebuah obyek terdeteksi^[8]. Metode *background subtraction* diterapkan untuk mendeteksi obyek bergerak dengan menggunakan kamera statis pada

citra video. Konsep awal dari metode ini memisahkan citra latar yang merepresentasikan sebagai citra yang tidak terdapat obyek bergerak dengan citra yang terdapat obyek Bergeraknya sehingga citra hasil berupa citra obyek bergerak^[9].

2.2.6 Thresholding

Thresholding merupakan cara yang digunakan dalam pengolahan citra, untuk mengkonversi citra grayscale atau derajat keabuan menjadi citra biner. Operasi ini dilakukan dengan mengelompokkan nilai derajat keabuan pada setiap pixel ke dalam 2 kelas, hitam dan putih^[10]. Thresholding (pengambilan) artinya adalah nilai piksel pada citra yang memenuhi syarat nilai ambang yang kita tentukan dirubah kenilai tertentu yang dikehendaki. Secara matematis ditulis seperti berikut:

$$f_o(x,y) = \begin{cases} T_1, & f_i(x,y) \leq T_1 \\ T_2, & T_1 < f_i(x,y) \leq T_2 \\ T_3, & T_2 < f_i(x,y) \leq T_3 \\ \dots & \dots \\ T_n, & T_{n-1} < f_i(x,y) \leq T_n \end{cases}$$

Dengan $f_i(x,y)$ adalah citra asli (input), $f_o(x,y)$ adalah piksel citra baru (hasil/output), T_n adalah nilai ambang yang ditentukan. Nilai piksel pada (x,y) citra output akan sama dengan T_1 jika nilai piksel (x,y) citra input tersebut $< T_1$. Nilai piksel (x,y) citra input akan sama dengan T_2 jika $T_1 < f_i(x,y) < T_2$, dan seterusnya. Sebagai contoh citra grayscale 8 bit akan dipetakan menjadi peta biner (hitam dan putih saja) dengan nilai ambang tunggal = 128 maka persamaan matematisnya

$$f_o(x,y) = \begin{cases} 0, & f_i(x,y) < 128 \\ 255, & f_i(x,y) \geq 128 \end{cases}$$

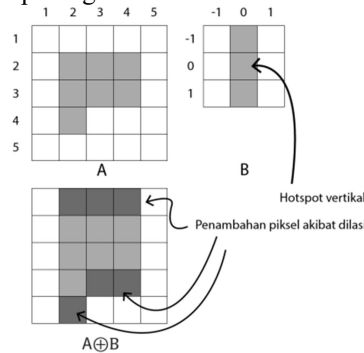
Piksel yang nilai intensitasnya dibawah 128 akan diubah menjadi hitam (nilai intensitas = 0), sedangkan piksel yang nilai intensitasnya diatas 128 akan menjadi putih (nilai intensitas = 255)^[11].

2.2.7 Dilasi

Burger & Burge mendefinisikan operasi dilasi sebagai berikut:

$$A \oplus B = \{z | z = a + b, \text{ dengan } a \in A \text{ dan } b \in B \}$$

Hasil dilasi berupa penjumlahan seluruh pasangan koordinat dari A dan B. [1]



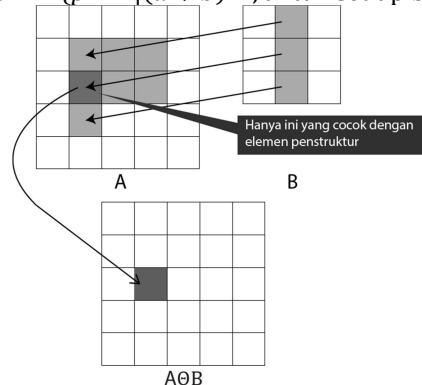
Gambar 1. Operasi Dilasi

Dilasi adalah proses penambahan piksel pada batas dari suatu obyek pada citra digital masukan^[8].

2.2.8 Erosi

Burger & Burge mendefinisikan operasi erosi sebagai berikut^[1]:

$$A \ominus B = \{p \in Z^2 | (a + b) \in 1, \text{ untuk setiap } b \in B \}$$

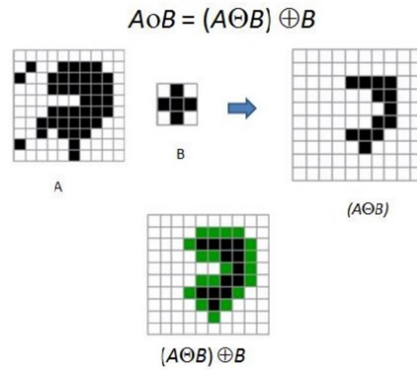


Gambar 2. Operasi Erosi

Erosi adalah proses pengurangan piksel pada batas dari suatu obyek^[8].

2.2.9 Opening

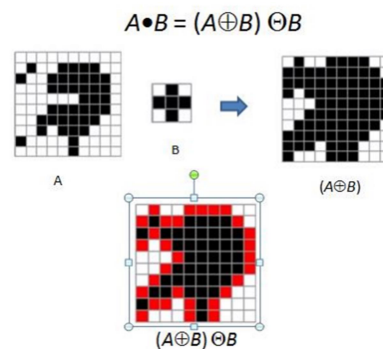
Operasi *opening* adalah operasi erosi yang diikuti dengan dilasi dengan menggunakan elemen penstruktur yang sama. Operasi ini berguna untuk menghaluskan kontur objek dan menghilangkan seluruh piksel di area yang terlalu kecil untuk ditempati oleh elemen penstruktur. Dengan perkataan lain, semua struktur latar depan yang berukuran kecil daripada elemen penstruktur akan tereliminasi oleh erosi dan kemudian penghalusan dilakukan melalui dilasi^[1]. Operasi ini memiliki fungsi untuk memberikan penghalusan permukaan obyek dan mengeliminasi semua piksel di area yang terlalu sempit dan kecil untuk diisi oleh elemen. Sehingga akhirnya seluruh area yang berukuran lebih kecil dari elemen penstruktur, akan dihilangkan menggunakan operasi erosi lalu operasi dilasi berperan untuk proses penghalusan^[8].



Gambar 3. Operasi Opening

2.2.10 Closing

Operasi *closing* dilaksanakan dengan melakukan operasi dilasi terlebih dahulu dan kemudian diikuti dengan operasi erosi^[1]. Operasi *closing* berguna untuk menghaluskan kontur dan menghilangkan lubang-lubang kecil^[8].

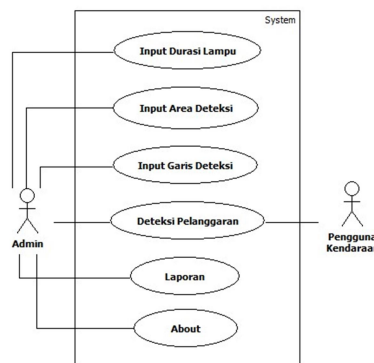


Gambar 4. Operasi Closing

3. HASIL DAN PEMBAHASAN

3.1 Perancangan Unified Modeling Language (UML)

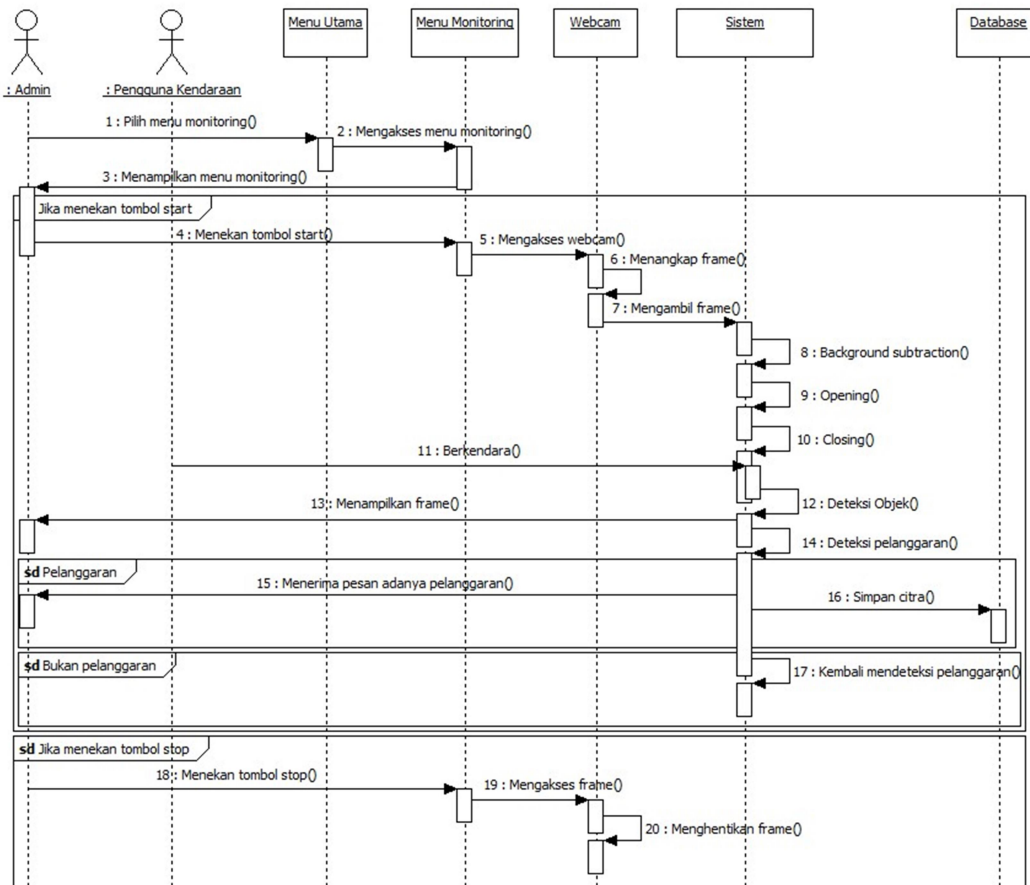
3.1.1 Diagram Use Case



Gambar 5. Diagram Use Case

Diagram *use case* pada gambar di atas terdapat dua aktor, yaitu pengguna yang bertindak sebagai admin dan pengguna kendaraan. Pengguna dapat untuk meng-*input* beberapa data, meliputi durasi lampu lalu lintas, area deteksi dan garis deteksi kendaraan. Pengguna dapat melihat dan mencetak laporan pelanggaran lalu lintas yang terdeteksi.

3.1.2 Diagram Sekuensial Deteksi Pelanggaran



Gambar 6. Diagram Sekuensial Deteksi Pelanggaran

Diagram sekuensial pada gambar di atas menggambarkan aliran prosedur yang terjadi pada proses deteksi pelanggaran lampu lalu lintas. Pada diagram tersebut, pengguna mengakses menu *monitoring*. Pada menu tersebut terdapat tombol *Start* dan tombol *Stop*. Apabila pengguna menekan tombol *Start*, maka sistem akan mengakses webcam dan menangkap *frame* yang diawasi webcam. *Frame* tersebut berupa daerah menjadi pengawasan dari webcam. Kemudian *frame* akan diproses dengan menggunakan metode *background subtraction*, operasi *opening*, dan operasi *closing* yang bertujuan untuk mendeteksi objek yang melakukan pergerakan di dalam area yang sudah ditentukan pada prosedur peng-*input*-an area deteksi. Ketika sistem mendeteksi adanya pelanggaran, maka sistem akan menampilkan pesan kepada pengguna dan mengambil citra kendaraan yang melakukan pelanggaran. Citra tersebut akan disimpan ke dalam *database*. Jika sistem tidak mendeteksi adanya pelanggaran, maka sistem akan kembali melakukan pendeteksian apakah terdapat objek yang melakukan pelanggaran lampu lalu lintas. Apabila pengguna menekan tombol *Stop*, maka sistem akan menghentikan pendeteksian objek dan pelanggaran pada *frame* webcam.

3.2 Tampilan Aplikasi Pengawasan Pelanggaran Lalu Lintas

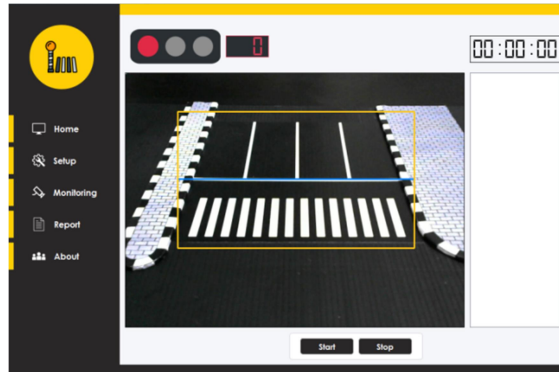
3.2.1 Tampilan Menu Monitoring

Menu *monitoring* dapat diakses melalui tombol *monitoring* pada aplikasi. Pada menu ini terdapat LCD *Number*, *image traffic light*, *frame video*, *list view*, tombol *Start*, dan tombol *Stop*. Berikut penjelasan mengenai elemen yang terdapat pada menu *monitoring*:

- Tombol *Start*, berfungsi untuk memulai proses pendeteksian pelanggaran lampu lalu lintas.
- Frame video*, berfungsi untuk menampilkan proses pendeteksian dalam bentuk video.
- List view*, berfungsi untuk menampilkan pesan apabila sistem mendeteksi adanya pelanggaran yang terjadi. Pesan tersebut berupa waktu terjadinya pelanggaran.

- d. *Image traffic light*, berfungsi untuk menampilkan lampu lalu lintas yang menyala secara bergantian sesuai dengan durasi waktu yang sudah ditentukan.
- e. *LCD number*, berfungsi untuk menampilkan durasi lampu lalu lintas secara *countdown* dan menampilkan waktu pada aplikasi.
- f. Tombol *Stop*, berfungsi untuk menghentikan proses pendeteksian.

Pengguna dapat memulai proses pendeteksian pelanggaran dengan menekan tombol *Start*. Proses pendeteksian akan ditampilkan pada *frame video*. Tampilan proses pendeteksian dapat dilihat pada gambar 8.



Gambar 7. Tampilan Menu Monitoring Saat Menjalankan Proses Pendeteksian

Ketika sistem mendeteksi adanya pelanggaran, sistem kemudian akan menampilkan pesan waktu terjadinya pelanggaran pada *list view* dan menangkap citra kendaraan yang melakukan pelanggaran lalu lintas dan menyimpannya ke dalam *database*. Pengguna dapat menghentikan proses pendeteksian pelanggaran dengan menekan tombol *Stop*.

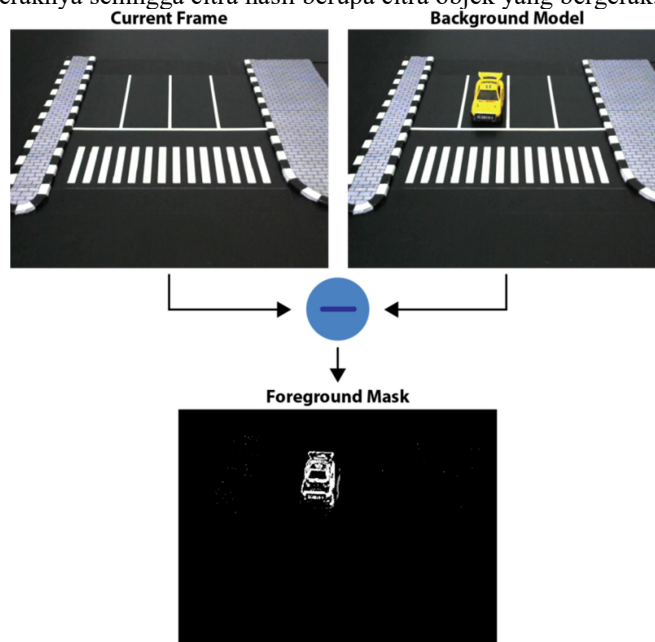
3.3 Proses Deteksi Pelanggaran

3.3.1 Pengambilan Frame Video

Pada tahap ini, sistem akan mengakses webcam dan mengambil *frame video* yang akan dilakukan proses pendeteksian. *Frame video* pada dasarnya merupakan gabungan citra gambar yang saling bergantian pada durasi waktu tertentu. Proses pergantian *frame video* saling berlangsung secara cepat sehingga membentuk suatu video yang terus berjalan

3.3.2 Background Subtraction

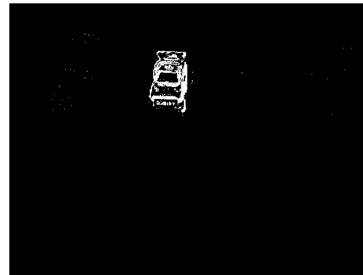
Pada tahap ini, *frame video* yang telah diambil kemudian akan diproses dengan menggunakan metode *background subtraction*. Metode *background subtraction* merupakan suatu proses untuk mendeteksi objek bergerak dengan menggunakan kamera statis pada *frame video*. Konsep dari metode ini memisahkan model latar belakang (*background*) yang merepresentasikan sebagai citra yang tidak terdapat objek bergerak dengan citra yang terdapat objek Bergeraknya sehingga hasil berupa citra objek yang bergerak.



Gambar 8. Skema Proses Background Subtraction

3.3.3 Thresholding

Pada tahap ini, *frame* hasil proses *background subtraction* kemudian akan dilakukan proses *thresholding*. Proses ini akan mengelompokkan nilai intensitas pada setiap piksel yang memenuhi syarat nilai *threshold* ke dalam dua warna, yaitu hitam dan putih. Pengelompokkan piksel mengacu pada nilai *threshold* atau nilai batas yang telah ditetapkan. Piksel dengan nilai intensitas yang lebih kecil dari nilai *threshold* akan diubah menjadi hitam (nilai intensitas = 0), sedangkan piksel yang nilai intensitasnya melebihi nilai *threshold* maka akan diubah menjadi putih (nilai intensitas = 255).



Gambar 9. Hasil Proses Thresholding

Hasil dari proses ini masih belum cukup, karena objek masih terdapat banyak lubang dan *noise*. Maka dari itu, perlu dilakukan beberapa tahap selanjutnya untuk memperbaiki objek dan menghilangkan *noise* pada citra.

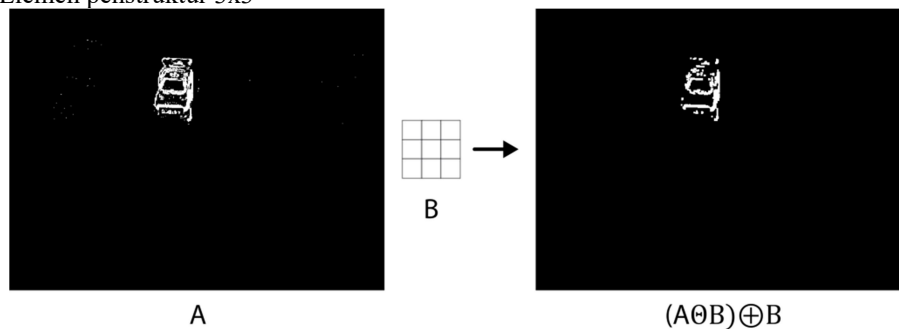
3.3.4 Opening

Pada tahap ini, *frame* hasil *thresholding* akan diolah dengan operasi *opening*. Operasi *opening* adalah operasi erosi yang kemudian diikuti dengan operasi dilasi. Operasi ini berfungsi untuk mengeliminasi seluruh piksel yang terlalu kecil untuk ditempati oleh elemen penstruktur. Sehingga akhirnya seluruh area yang berukuran lebih kecil dari elemen penstruktur, akan dihilangkan menggunakan operasi erosi lalu operasi dilasi berperan untuk proses penghalusan.

Keterangan:

A = Citra hasil *thresholding*

B = Elemen penstruktur 3x3



Gambar 10. Tampilan Visual Operasi Opening

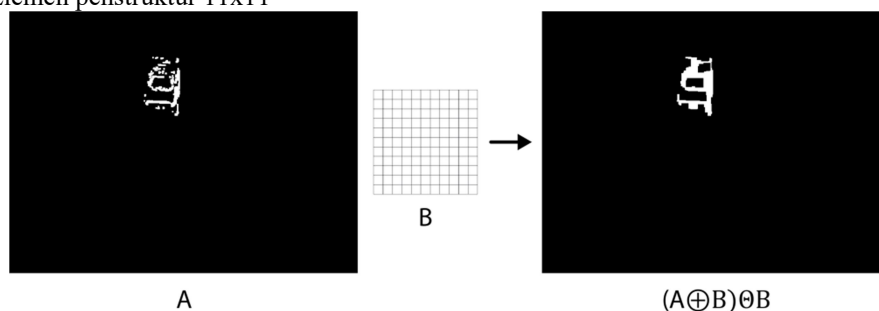
3.3.5 Closing

Pada tahap ini, *frame* hasil operasi *opening* akan diolah dengan proses *closing*. Operasi *closing* adalah operasi dilasi yang kemudian diikuti dengan operasi erosi. Operasi ini bertujuan untuk menghaluskan area objek dan mengisi lubang-lubang kecil pada objek.

Keterangan:

A = Citra hasil operasi *opening*

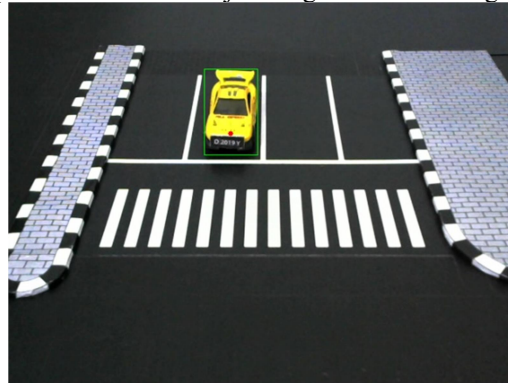
B = Elemen penstruktur 11x11



Gambar 11. Tampilan Visual Operasi Closing

3.3.6 Deteksi Objek

Setelah melalui beberapa proses sebelumnya, akan dilakukan pendeteksian objek pada citra hasil *closing*. Pendeteksian objek dilakukan dengan mencari *contour* dari objek yang bergerak. *Contour* (kontur) merupakan suatu garis kurva yang menggabungkan semua titik sepanjang batas objek yang memiliki warna atau intensitas yang sama. Garis tersebut merupakan batas antara objek dengan latar belakang.



Gambar 12. Penandaan Pada Objek

3.3.7 Deteksi Pelanggaran

Pelanggaran akan terjadi jika objek kendaraan melewati batas marka *zebra cross* saat lampu lalu lintas merah menyala. Untuk dapat mendeteksi adanya pelanggaran, maka perlu dilakukan penentuan suatu area deteksi. Penentuan area deteksi ini bertujuan untuk mengawasi area yang diperlukan untuk mendeteksi objek kendaraan. Setelah area deteksi telah ditentukan, maka selanjutnya adalah menentukan suatu garis bantu yang melintang secara horizontal sebagai batas dari marka *zebra cross*. Posisi dari garis batas ini tidak bergerak dan diposisikan sebelum marka *zebra cross*.

3.4 Perancangan Diorama Jalan Raya

Rancangan diorama jalan raya dibuat untuk menggambarkan dan memudahkan pengujian sistem dengan webcam secara langsung. Rancangan diorama yang dibuat hanya menggunakan satu ruas jalan yang terdapat marka *zebra cross* dan ditempatkan pada papan yang berukuran ± 61 x 50 cm.

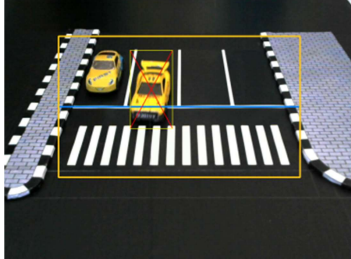
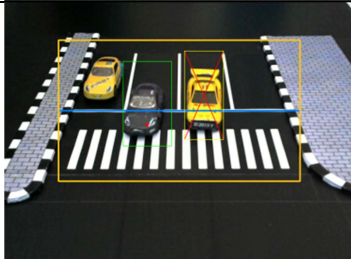
3.5 Pengujian Sistem

Pengujian sistem merupakan suatu proses uji coba sistem yang dilakukan untuk mengidentifikasi tingkat keberhasilan dari sistem yang telah dibangun. Pengujian pendeteksian pelanggaran lampu lalu lintas ini menggunakan metode *black box testing* untuk menguji fungsionalitas dari sistem. Pengujian sistem dilakukan dengan beberapa mobil mainan yang ditarik melewati marka *zebra cross* pada saat kondisi lampu lalu lintas merah menyala. Pengujian dilakukan dalam beberapa kondisi. Adapun pengujian sistem yang dilakukan sebagai berikut:

3.5.1 Pengujian Pagi Hari

Pengujian pagi hari dilakukan di luar ruangan pada pukul 09.00 WIB. Kondisi pada pagi hari memiliki intensitas cahaya yang terang.

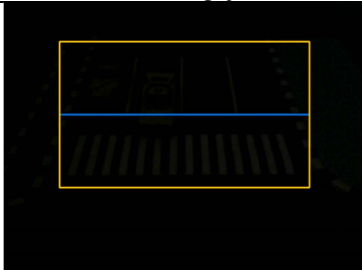
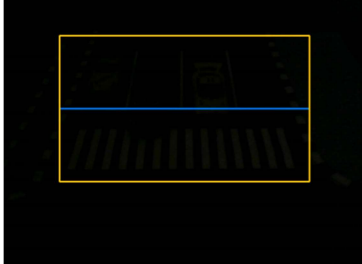
Tabel 1. Pengujian Pagi Hari

No	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Satu mobil melintas dan satu mobil berhenti	Mendeteksi pelanggaran		Valid
2	Dua mobil melintas dan satu mobil berhenti	Mendeteksi pelanggaran		Valid

3.5.2 Pengujian Malam Hari Tanpa Pencahayaan

Pengujian dilakukan di dalam ruangan tanpa ada pencahayaan dari lampu.

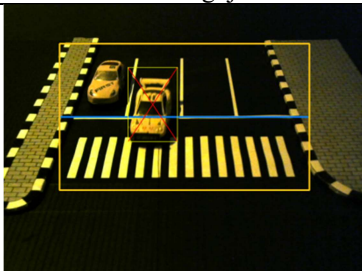
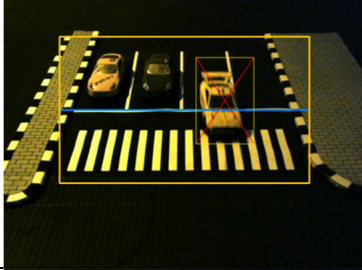
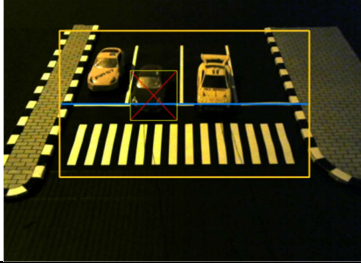
Tabel 2. Pengujian Malam Hari Tanpa Pencahayaan Lampu

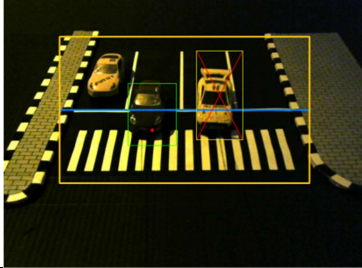
No	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Satu mobil melintas dan satu mobil berhenti	Mendeteksi pelanggaran		Tidak Valid
2	Dua mobil melintas dan satu mobil berhenti	Mendeteksi pelanggaran		Tidak Valid

3.5.3 Pengujian Malam Hari Dengan Pencahayaan Lampu

Pengujian dilakukan di dalam ruangan yang gelap dengan bantuan pencahayaan dari lampu untuk menerangi diorama jalan raya.

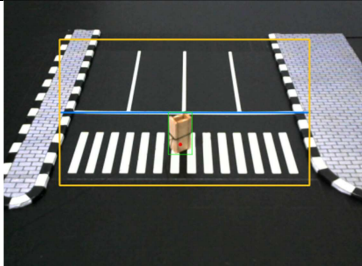
Tabel 3. Pengujian Malam Hari Dengan Pencahayaan Lampu

No	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Satu mobil melintas dan satu mobil berhenti	Mendeteksi pelanggaran		Valid
2	Satu mobil melintas dan dua mobil berhenti	Mendeteksi pelanggaran		Valid
3	Dua mobil melintas dan satu mobil berhenti	Mendeteksi pelanggaran		Valid

				
--	--	--	--	--

3.5.4 Pengujian Objek Menyebrang di Zebra Cross

Pengujian dilakukan dengan suatu objek yang ditarik menyebrangi marka zebra cross.

No	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Objek menyebrangi zebra cross	Tidak Mendeteksi pelanggaran		Valid

4. KESIMPULAN

Berdasarkan perancangan prototipe aplikasi pengawasan pelanggaran lalu lintas menggunakan OpenCV yang telah dilakukan, maka penulis dapat menarik beberapa kesimpulan sebagai berikut:

- Dari hasil pengujian yang telah dilakukan maka dapat diketahui bahwa aplikasi yang dirancang dapat mendeteksi adanya pelanggaran lampu lalu lintas saat kondisi lampu merah. Ketika terdapat pelanggaran maka sistem akan mengambil citra kendaraan tersebut.
- OpenCV mempunyai banyak fitur yang dapat dimanfaatkan dalam pengembangan sistem di bidang *computer vision* dan *image processing*.
- Berdasarkan hasil pengujian, objek kendaraan dapat terdeteksi dengan baik dalam kondisi pencahayaan yang baik. Namun, terdapat kondisi bayangan objek kendaraan juga ikut terdeteksi apabila intensitas bayangan objek melebihi nilai batas *thresholding*.
- Pada pengujian dengan kondisi pencahayaan yang minim, sistem tidak dapat mendeteksi objek kendaraan dengan baik.
- Metode *background subtraction* yang digunakan tidak dapat membedakan objek yang bukan kendaraan karena metode ini akan mendeteksi setiap objek yang bergerak dalam suatu area.

5. SARAN

Setelah merancang prototipe aplikasi pengawasan pelanggaran lalu lintas menggunakan OpenCV, penulis menyadari bahwa aplikasi yang dihasilkan mempunyai kekurangan. Berdasarkan kekurangan tersebut, terdapat beberapa saran agar aplikasi ini dapat dikembangkan dan digunakan sebagai acuan untuk melakukan penelitian yang akan datang adalah sebagai berikut:

- Menggunakan kamera yang memiliki penyesuaian fokus yang baik dan kemampuan untuk merekam objek yang bergerak cepat sehingga dapat meningkatkan kualitas *frame* video dan hasil penangkapan citra kendaraan lebih jelas.
- Menambahkan sistem pengenalan objek (*object recognition*) untuk mengklasifikasikan setiap objek yang terdeteksi.
- Menambahkan sistem untuk mendeteksi objek plat nomor kendaraan. Sistem secara otomatis akan mencari posisi plat nomor kendaraan dalam suatu citra kendaraan.
- Menambahkan sistem *Optical Character Recognition* (OCR) untuk mengenali karakter yang ada pada plat nomor sehingga dapat dilakukan proses pencatatan plat nomor kendaraan ke sistem secara otomatis.

UCAPAN TERIMA KASIH

Dalam penulisan ini, penulis telah banyak mendapatkan bantuan berupa bimbingan, petunjuk, data, saran, maupun dorongan moral dari berbagai pihak, maka pada kesempatan ini penulis mengucapkan terima

kasih kepada civitas akademika Fakultas Teknologi Informasi Universitas Widya Dharma Pontianak, serta kepada keluarga tercinta yang telah banyak memberikan kasih sayang, bantuan, dan dukungan serta doa kepada penulis selama menjalani studi dari awal perkuliahan hingga selesainya penelitian ini.

DAFTAR PUSTAKA

- [1] Kadir, Abdul dan Adhi Susanto. (2013). *Teori dan Aplikasi Pengolahan Citra*. Andi. Yogyakarta.
- [2] Andono, Pulung Nurtantio, T. Sutojo, dan Muljono. (2018). *Pengolahan Citra Digital*. Andi. Yogyakarta.
- [3] Budiharto, Widodo. (2012). *Robot Vision*. Andi. Yogyakarta.
- [4] Datta, Samyak. (2016). *Learning OpenCV 3 Application Development*. Packt Publishing. Birmingham.
- [5] Laganiere, Robert. (2017). *OpenCV 3 Computer Vision Application Programming Cookbook Third Edition*. Packt Publishing. Birmingham.
- [6] Jubilee Enterprise. (2013). *Teknik Mengendalikan PC dari Jarak Jauh*. PT Elex Media Komputindo. Jakarta.
- [7] Cahyadi, Budi. (2014). *Home Security*. Andi. Yogyakarta.
- [8] Umam, Khairul dan Benny Sukma Negara. (Desember 2016). "Deteksi Obyek Manusia Pada Basis Data Video Menggunakan Metode Background Subtraction Dan Operasi Morfologi." *Jurnal CoreIT*. Vol. 2, no. 2:hal. 33-34.
- [9] Prabowo, Cipto dan Zurnawita. (November 2018). "Penerapan Background Subtraction dengan Menggunakan Kandidat Sampling Background Untuk Deteksi Kemacetan." *Jurnal Teknologi Informasi dan Ilmu Komputer*. Vol. 5, no. 6:hal. 732.
- [10] Christianto, Rizky Natanael dan Yulius Denny Prabowo. (Agustus 2017). "Aplikasi Perekam Citra Berdasarkan Pergerakan Objek Yang Nampak." *Jurnal Sains dan Teknologi*. Vol. 4, no. 2:hal. 187.
- [11] Perkasa, Therzian Richard, Helmy Widyantara dan Pauladie Susanto. (September 2013). "Rancang Bangun Pendeteksi Gerak Menggunakan Metode Image Subtraction pada Single Board Computer (SBC)." *Journal of Control and Network Systems*. Vol. 3, no. 2:hal. 92-93.