SISTEM KONTROL LAMPU LALU LINTAS BERDASARKAN KEPADATAN KENDARAAN MENGGUNAKAN IMAGE PROCESSING

Traffic Light Control System Based On Vehicle Density Using Image Processing

Muhammad Arfi Ario Saputra [1], Asep Mulyana [2], Suci Aulia[3]

[1,2,3] Universitas Telkom, Bandung
arfiaryo@students.telkomuniversity.ac.id[1] , asepmulyana@telkomuniversity.ac.id[2], sucia@tass.telkomuniversity.ac.id [3]

Abstrak

Pada umumnya pengaturan lamanya menyala lampu hijau pada lampu lalu-lintas (*traffic tight*) saat ini menggunakan waktu yang tetap dan sama untuk tiap arah, misalnya pada simpang empat dari masing-masing arah jalan dibuat sama selama satu menit. Dalam kondisi tidak seimbang dimana terdapat antrian panjang di satu sisi dan di sisi lain sangat pendek, maka pengaturan durasi sama rata ini menimbulkan ketidakadilan.

Dalam Proyek Akhir ini dibuat suatu alat (purwarupa) untuk mengatur durasi/lamanya waktu menyalanya lampu hijau yang sebanding dengan panjang antrian kendaraan menggunakan metoda pengolahan citra berdasarkan konversi warna kendaraan dari *Red Green Blue* (RGB) ke *grey scale* dan ke *black white* dengan menggunakan kendaraan mainan berbagai warna dengan latar belakang jalan raya berwarna putih sebagai simulasi. Karena keterbatasan jumlah kamera, muka dalam percobaan jumlah arah jalan dibatasi hanya dua arah. Dalam pengolahan citra setelah diperoleh *black and white*, maka dengan menggunakan metoda proporsi luas warna hitam dari tiap arah adalah sebanding dengan jumlah pixel pada masing-masing arah maka proporsi jumlah pixel ini digunakan untuk menentukan lamanya lampu hijau menyala di masing-masing arah,

Hasil pengujian menunjukkan dari segi fungsionl urutan menyalanya lampu (merah, kuning, hijau) urutannya sesuai dengan yang telah direncanakan. Selanjutnya dari segi akurasi lamanya lampu hijau menyala berdasarkan rasio jumlah pixel dibandingkan dengan rasio panjang antrian diperoleh akurasi 75% untuk jalan arah A dan 100% untuk jalan arah B.

Kata Kunci: Pengolahan Citra Digital, Edge Detection, Real-Time, Traffic Light, piksel putih, kepadatan kendaraan, panjang antrian kendaraan

Abstract

In general, the adjustment of the length of time the green light of traffic lights currently uses a fixed time and the same for each direction, for example at the intersection of four directions of the road of traffic light is one minute. At the certain situation where the an unbalanced condition a long queue on the one hand and a very short on the other hand, the setting of equal duration then this way is not fair.

In this final project, a prototype is made to adjust the duration of the green light which is proportional to the queue length of the vehicle. This manner is using an image processing method based on vehicle color conversion from Red Green Blue (RGB) to gray scale and to black white by using a vehicle. toys of various colors with a white highway background as a simulation. Due to the limited number of cameras, the face in the experiment in the number of directions is limited to only two directions. In image processing, after obtaining black and white, using the method of proportioning the area of black color from each direction is proportional to the number of pixels in each direction, the proportion of the number of pixels is used to determine the length of time the green light is on in each direction,

The test results show that from a functional point of view, the sequence of lights (red, yellow, green) is in accordance with what has been planned. Furthermore, in terms of accuracy of the length of time the green light is on based on the ratio of the number of pixels compared to the ratio of the queue length, an accuracy of 75% is obtained for the "A" direction-way and 100% for the "B" direction.

Keyword: Digital Image processing, Edge Detection, Real-Time, Traffic Light, White Pixel, Vehicle Density, Length Queue of Vehicle

1. Pendahuluan

Lampu lalu lintas merupakan infrastruktur penting untuk mengatur ketertiban pemakaian jalan raya khususnya di persimpangan. Ketidakteraturan pemakaian jalan raya dapat menimbulkan kecelakaan maupun memacetan. Kemacetan lalu lintas dapat menyebabkan pemborosan berupa waktu dan biaya konsumsi bahan bakar yang sia-sia. Salah satu penyebab terjadinya kemacetan antara lain karena pengaturan lamanya lampu hijau menyala tidak proporsional dengan panjang antrian. Sistem pengaturan lampu lalulintas yang ada saat ini masih belum optimal dari segi keadilan (fairness) dalam pengaturan durasi/lamanya waktu menyalanya lampu hijau dimana saat ini pengaturannya masih dibuat konstan dan sama untuk setiap arah jalan di titik persimpangan, misalnya untuk masing-masing arah dibuat tetap selama 1 menit. Hal ini tentu saja menimbulkan ketidakadilan dalam hal terjadi ketidakseimbangan panjang antrian dari tiap arah jalan di titik persimpangan. Sebagai contoh terdapat antrian kendaraan dari arah A, B, C, dan D masing-masing sejumlah 10, 2, 8, dan 20 kendaraan. Maka semestinya pengaturan durasi menyalanya lampu hijau untuk arah A, B, C, D masing-masing adalah : :(10/40)x4menit, (2/40)x4menit, (8/40)x4menit, (20/40)x4menit. Menghasilkan masing-masing : 60 detik, 12 detik, 48 detik, dan 120 detik.

Berdasarkan penelitian yang telah ada Dalam penelitiannya Sonia Dian Maniswari (2015)[8] berjudul "Smart Traffic Light Menggunakan Image Processing Dan Metode Fuzzy Logic", menggunakan kamera (pengolahan citra dijital) untuk mendeteksi jumlah kendaraan berdasarkan proses ekstraksi kontur. Jumlah objek tersebut menjadi input untuk sistem penghitungan lama waktu lampu lalu lintas menggunakan algoritma fuzzy logic. Selanjutnya Fariz Muhammad Rachmatullah (2020) [9] dalam penelitiannya berjudul "Sistem Pengaturan Lampu Lalu Lintas Menggunakan Pengolahan Citra", melakukan sistem pendeteksi antrian kendaraan berdasarkan pengolahan citra untuk mendeteksi antrian terpanjang dan terpendek dalam setiap siklusnya untuk mengatur lamanya lampu lalu lintas. Dan berikutnya Rivo Dwi Yulianto (2021)[10], dalam penelitiannya berjudul "Perancangan Sistem Informasi Untuk *Smart Traffic Light* Berdasarkan Rute Darurat", membuat sistem pendeteksian kedatangan kendaraan pemadam kebakaran (PMK) yang mendekatinya. Pada jarek tertentu yang sudah relatif dekat, dinyalakan lampu hijau dari arah datangnya mobil pemadam kebagaran agar mobil PMK tidak perlu berhenti.

Maka dalam proyek akhir ini bertujuan untuk merancang purwarupa sistem lampu lalu lintas dengan memanfaatkan teknologi pengolahan citra untuk mengolah data kepadatan kendaraan yang dideteksi oleh sensor kamera yang dipasang disetiap simpang , Sistem pengaturan lampu lalu lintas ini bekerja berdasarkan masukan berupa gambar hasil capture yang diolah menggunakan metode pengolahan citra, dimana gambar diproses dari format RGB menjadi grayscale, thresholding, hole filling, dan dilliation & errosion. Pada penelitian yang dilakukan kali ini, penulis akan menggunakan metode pengolahan citra dengan teknik pengurangan citra.

.

2. Dasar Teori

2.1. Lampu Lalu Lintas

Lampu lalu lintas adalah suatu alat kendali (kontrol) dengan menggunakan lampu yang terpasang pada persimpangan dengan tujuan untuk mengatur lalu lintas. Pengaturan lalu lintas pada persimpangan pada dasarnya dimaksudkna untuk bagaimana pergerakan kendaraan pada masing-masing kelompok pergerakan kendaraan dapat bergerak secara pergantian sehingga tidak saling menggangu antar arus yang ada[1].

2.2 Pengolahan Citra (*Image Processing*)

Pengolahan citra atau *image processing* adalah suatu sistem dimana proses dilakukan dengan masukan (*input*) berupa citra (*image*) dan hasilnya (*output*) juga berupa citra (*image*). Pada awalnya pengolahan citra ini dilakukan dengan berkembanganya dunia komputasi yang ditandai dengan semakin meningkatnya kapasitas dan kecepatan proses komputer, serta munculnya ilmu-ilmu komputer yang memungkinkan manusia dapat mengambil informasi daru suatu citra maka *image processing* tidak dapat dilepaskan dengan bidang *computer vision*[2].

2.3 Citra Digital

Citra atau image adalah angka, dari segi estetika, citra atau gambar adalah kumpulan warna yang bias terlihat indah, memiliki pola, berbentuk abstrak dan lain sebagainya. Citra dapat berupa foto udara, penampang lintang (cross section) dari suatu benda, gambar wajah, hasil tomografi otak dan lain sebagainya. Dari segi ilmiah, citra adalah gambar 3 dimensi (3D) dari suatu fungsi, biasanya intensitas warna sebagai fungsi spatial x dan y. Dikomputer warna dapat dinyatakan, misalnya sebagai angka dalam bentuk skala RGB. Karena citra adalah angka, maka citra dapat diproses secara digital[2].

2.4 RGB (*Red Green & Blue*)

Untuk citra berwarna maka digunakan model RGB (Red-Green-Blue), satu citra berwarna dinyatakan sebagai 3 buah matrik grayscale yang berupa matrik untuk Red (R-layer), matrik Green (G-layer), dan

matrik Blue (B-layer). R-layer adalah matrik yang menyatakan derajat kecerahan untuk warna merah (misalkan untuk skala keabuan 0-255, nilai 0 menyatakan gelap (hitam) dan 255 menyatakan merah. G-layer adalah matrik yang menyatakan derajat kecerahan untuk warna hijau, dan B-layer adalah matrik untuk menyatakan derajat warna biru. Dari definisi tersebut, untuk menyajikan warna tertentu dapat dengan mudah dilakukan, yaitu dengan mencampurkan ketiga warna dasar RGB[2].

2.5 Grayscale

Dalam pengolahan citra, grayscale merupakan suatu citra yang hanya memiliki satu nilai warna pada setiap pikselnya, dimana warna dari *grayscale* adalah warna keabuan yang berada diantara hitam dan putih, dengan kata lain nilai *red = green = blue.* Sistem warna *grayscale* digunakan karena pada proses threshold, citra yang dapat diproses adalah citra dengan derajat keabuan atau disebut dengan *grayscale*[3].

2.6 Black and white

Citra digital black and white setiap pikselnya mempunyai warna gradasi mulai dari putih sampai hitam. Rentang tersebut berarti bahwa setiap piksel dapat diwakili oleh 8 bit atau 1 *byte*. Rentang warna pada *black and white* sangat cocok digunakan untuk pengolahan file gambar[4].

2.7 Thresholding

Thresholding merupakan konversi citra hitam – putih ke citra biner dilakukan dengan cara mengelompokkan nilai derajat keabuan setiap piksel kedalam 2 kelas, hitam dan putih. Pada citra hitam putih terdapat 256 *level*, artinya mempunyai skala “0” sampai “255” atau [0,255], dalam hal ini nilai intensitas 0 menyatakan hitam, dan nilai intensitas 255 menyatakan putih, dan nilai antara 0 sampai 255 menyatakan warna keabuan yang terletak antara hitam dan putih[2].

2.8 Metode Otsu

Otsu thresholding merupakan metode segmentasi yang sederhana dalam teknik segmentasi, sehingga dapat lebih mudah dalam melakukan pembagian wilayah wilayah yang homogen berdasarkan kriteria keserupaan untuk mengenali objek[5].

2.9 Dillation dan errosion

Proses *dillation* dalam morphologi *image* identik dengan menambahkan piksel dalam lingkup *image* asal, dengan cara menempatkan satu demi satu pusat penataan elemen untuk masing-masing piksel *background*. Bila sembarang piksel *neighbourhood* bernilai piksel *foreground* (nilai 1) maka piksel *background* dirubah ke *foreground*. Proses *errosion* merupakan proses menghilangkan piksel dalam lingkup objek *image* dengan cara meletakkan pusat penataan elemen satu demi satu dalam piksel *foreground* (nilai 1)[6].

2.10 Matlab

MATLAB adalah sebuah bahasa dengan (*high-performance*) kinerja tinggi untuk komputasi masalah teknik. Matlab mengintegrasikan komputasi, visualisasi, dan pemrograman dalam suatu model yang sangat mudah untuk dipakai dimana masalah-masalah dan penyelesaiannya diekspresikan dalam notasi matematika yang familiar. Penggunaan Matlab meliputi bidang–bidang:

- Matematika dan Komputasi
- Pembentukan Algorithm
- Akusisi Data
- Pemodelan, simulasi, dan pembuatan prototype
- Analisa data, explorasi, dan visualisasi
- Grafik Keilmuan dan bidang Rekayasa

2.11 Webcam

Webcam adalah perangkat yang berupa sebuah kamera digital yang dihubungkan ke komputer atau laptop. Layaknya kamera pada umumnya, sebuah webcam dapat mengirimkan gambar-gambar secara *live* dari manapun ia berada ke seluruh penjuru dunia dengan bantuan internet. Webcam merupakan sejenis kamera kompak yang dapat dipasang ke komputer atau laptop berguna menyiarkan video secara *real-time* alias live. Mirip dengan kamera digital pada umumnya, webcam bekerja dengan menangkap cahaya lewat lensa berukuran kecil di bagian depan dengan bantuan detektor cahaya mikroskopik yang terpasang pada *microchip* penerima gambar, yang umumnya berteknologi *Charge-Couple Device (CCD)* atau *CMOS image sensor*.

Gambar 2.1 Webcam Logitech C170

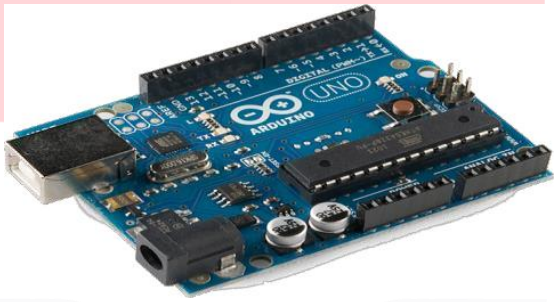
Spesifikasi dari *webcam Logitech C170* dapat dilihat dari tabel 2.4 :

Tabel 2.1 Spesifikasi Webcam

Frame Rate	30 Fps
Resolusi Maks	1024 x 768 pixels
Interface	USB 2.0
Resolusi Video	Full HD
Panjang Kabel	1.21m

2.12 Arduino Uno

Arduino Uno adalah papan sirkuit berbasis mikrokontroller ATmega328 pengendali mikro *singleboard* yang bersifat *open-source*, diturunkan dari Wiring platform, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. *Hardware*-nya memiliki prosesor Atmel AVR dan *software* nya memiliki bahasa pemrograman sendiri[7]. Memiliki 14 pin *input* dari *output* digital dimana 6 pin input tersebut dapat digunakan sebagai *output* PWM dan 6 pin *input* analog, 16 MHz osilator kristal, koneksi USB, *jack power*, ICSP header, dan tombol reset.



Gambar 2.2 Arduino Uno

Spesifikasi dari Arduino dapat dilihat dari tabel 2.10 berikut :

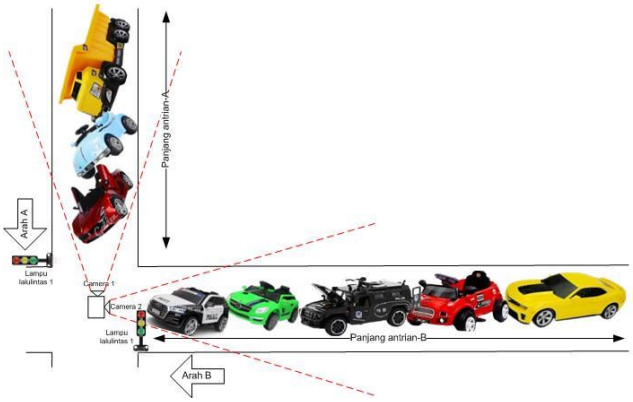
Tabel 2.2 Bagian Arduino UNO

Mikrokontroller	ATmega 328
Tegangan Pengoperasian	5 V
Tegangan Input Yang Disarankan	7 – 12 V
Batas Tegangan Input	6 – 20 V
Jumlah Pin I/O Digital	14 pin digital (6 diantaranya menyediakan keluaran PWM)
Jumlah Pin Input Analog	6 pin
Arus DC Tiap Pin I/O	40mA
Arus DC Untuk Pin 3,3 V	50mA
Memori *Flash*	32 KB (ATmega 328) sekitar 0,5 KB digunakan oleh bootloader
SRAM	2 KB (ATmega 328)
EPROM	1 KB (ATmega 328)
Clock Speed	16 MHz

3. Perancangan Sistem dan Realisasi

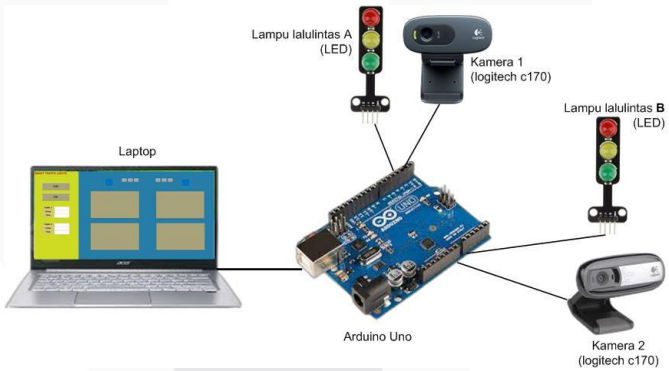
3.1. Model Percobaan (Simulasi)

Dalam model percobaan berikut ini direalisasikan dengan metoda simulasi menggunakan kendaraan mainan berbagai warna, situasi jalan raya berlatar belakang jalan berwarna putih, dengan arah jalan hanya dibatasi dari 2 arah jalan raya (karena keterbatasan jumlah kamera yang ada hanya dua buah) yaitu menggunakan dua buah kamera *webcam logitech c170* seperti terlihat pada Gambar 3.1 berikut ini.



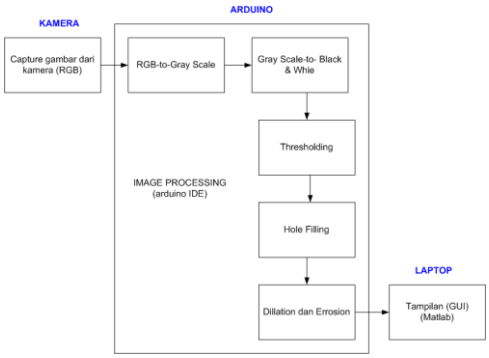
Gambar 3.1 Model Percobaan

Dalam simpang jalan dua arah tersebut terpasang 2 buah kamera, masing-masing kamera 1 memonitor panjang antrian kendaraan dari arah jalan A, dan kamera 2 memonitor panjang antrian kendaraan dari arah jalan B, Kedua kamera tersebut terkoneksi arduino sebagi pemroses hasil pengambilan gambar dari kamera dan mengontrol lampu lalulintas yang ditampilkan berupa *Graphical User Interface* (GUI) pada laptop. Saat pertama kali dihidupkan arduino mengambil gambar kamera 1, kemudian mengolahnya secara citra digital untuk mendapatkan panjang antrian kendaraan pada jalan arah A. Dilanjutkan kemudian mengambil gambar kamera 2 untuk mendapatkan panjang antrian kendaraan pada jalan arah B Selanjutnya kedua panjang antrian dari kedua arah jalan tersebut dibandingkan dan hasil perbandingan tersebut digunakan untuk menentukan lamanya menyala lampu hijau di masing-masing arah jalan (A dan B). Misalkan hasil perbandingan panjang antrian dari kedua arah jalan tersebut adalah a/b. Dan jika seandainya a=b maka masing-masing arah jalan akan diberi lama waktu 1 menit, maka total lamanya waktu lampu hijau menyala adalah 2 menit. Maka secara umum lama waktu lampu hijau untuk arah A adalah = {a/(a+b)} x 2 menit, misal= ta, untuk arah jalan B, lamanya menyala lampu hijau adalah {a/(a+b)} x 2 menit, misal= tb. Maka selanjutnya petama akan dinyalakan lampu hijau untuk arah A selama ta menit, dilanjutkan lampu kuning selama 3 detik dilanjutkan lampu merah untuk arah A bersamaan dengan menyalanya lampu hijau arah B selama tb menit dilanjutkan lampu kuning arah B selama 3 detik dan seterusnya dilakukan secara berulang ulang. Adapun gambar diagram realisasi dari sistem *traffic light* tersebut diperlihatkan pada Gambar 3.2 berikut ini :



Gambar 3.2.Blok diagram realisasi

Dan secara diagram blok sistem, proses yang terjadi adalah seperti digambarkan pada Gambar 3.3 berikut :



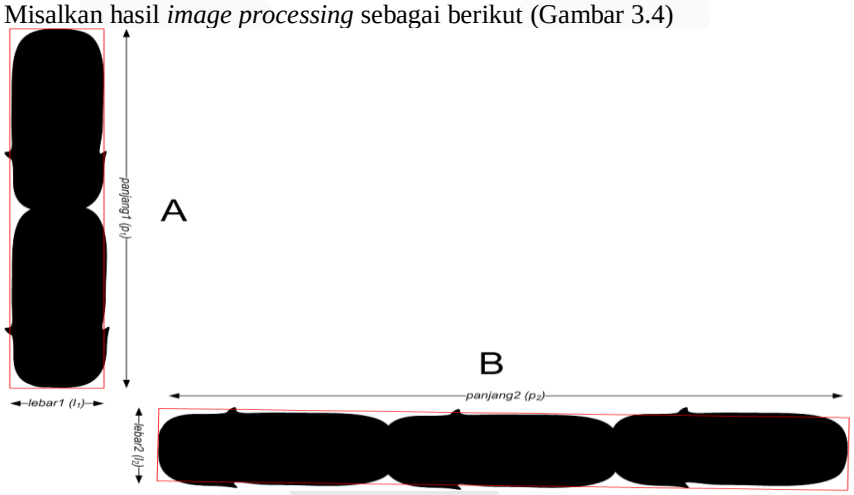
Gambar 3.3.Blok diagram sistem

1. *Capture Image*
   Akusisi citra merupakan langkah pertama dari proses identifikasi objek dan warna. Untuk mendapatkan citra dilakukan pengambilan gambar menggunakan *webcam logitech c170* dengan resolusi maksimal 1024x768 *pixels*, dengan pengaruh antara jarak dan cahaya saat mengambil objek berupa mobil.
2. Melakukan *Image Processing*
   Setelah melakukan pengambilan gambar kemudian dilakukan *image processing* dengan menggunakan beberapa tahap *preprocessing* yaitu akan melewati beberapa tahap mulai dari *input* data citra, konversi *RGB to Grayscale* adalah untuk memudahkan proses selanjutnya karena jika *RGB* proses lebih rumit dibandingkan dengan *Grayscale* karena *RGB* terdiri dari 3 *layer* dan *Grayscale* terdiri dari 1 *layer* yang merupakan hasil penyederhanaan dari 3 *layer* tersebut, kemudian di konversi dari *grayscale to BW* (*black and white*) citra yang berskala keabuan ini dikonversi ke citra biner cara yang digunakan yaitu dengan menerapkan suatu nilai yang dikenal sebagai nilai *thresholding*, *hole filling* adalah untuk mengisi lubang di input gambar biner *Black And White*, *dillation* biasa dipakai untuk mempertebal piksel bernilai 1 dan *errosion* biasa dipakai untuk menghilangkan piksel pada batas objek.
3. Melakukan Perhitungan Perbandingan *Pixel*
   Setelah dilakukannya *image processing* keemudian langkah selanjutnya adalah melakukan perhitungan jumlah *pixel* warna *black* pada tiap arah jalan A dan B yang selanjutnya perbandingan jumlah piksel tersebut digunakan sebagai pendekatan perbandingan panjang antrian kendaraan dimana perbandingan panjang antrian ini dijadikan sebagai dasar untuk menentukan lamanya menyala lampu hijau secara berbanding lurus seperti yang telah diuraikan pada Sub Bab 3.1.

3.2. Metoda pendekatan perbandingan jumlah pixel sebagai representasi perbandingan panjang antrian

Adapun yang menjadi dasar pemikiran bahwa jumlah piksel sebagai pendekatan panjang antrian dapat dijelaskan sebagai berikut :

- Misalkan hasil *image processing* sebagai berikut (Gambar 3.4)

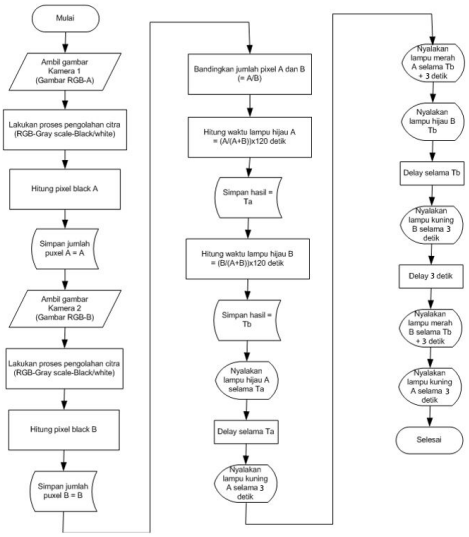


Gambar 3.4.Hasil pengolahan citra (black)

- Bentuk kurva hasil pengolahan citra (*black*) dari masing-masing arah A dan B pada Gambar 3.4 dapat didekati masing-masing dengan bentuk (ideal) empat persegi panjang A dan B.
- Artinya perbandingan jumlah piksel hasil pengolahan citra (*black*) masing masing A dan B dapat didekati dengan perbandingan luas empat persegi panjang A dan B.
- Luas A = (*p1xl1*) dan luas B = (*p1xl1*) .
- Maka perbandingan luas A dengan luas B = (*p1xl1*)/ (*p2xl2*).
- Dengan asumsi semua kendaraan mempunyai lebar yang sama atau *l1=l2*, maka perbandingan luas A dan B menjadi *p1*/*p2*.
- Dengan demikian maka perbandingan panjang antrian kendaraan dari masing- masing arah (misal A, B) dapat didekati dengan jumlah piksel hasil pengolahan citra (*black*) dari masing-masing arah lalulintas (A, B).

3.3. Perancangan program lampu lalu lintas

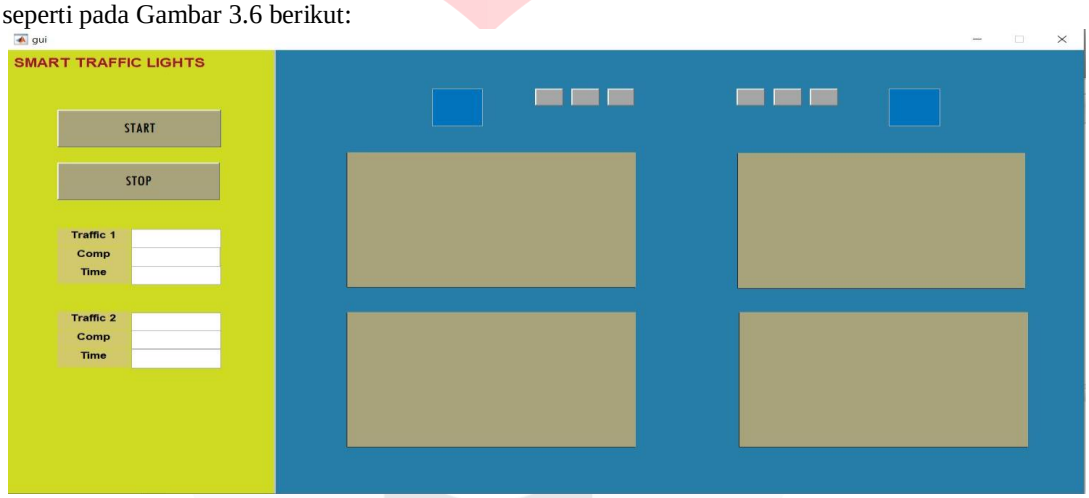
Gambar 3.5 berikut adalah diagram alir pemrograman pada Arduino



Gambar 3.5.Diagram alir pemrograman pada Arduino

3.4. Pembuatan GUI

Untuk menampilkan tahapan-tahapan yang terjadi pada sistem khususnya pada saat melakukan peragaan maupun saat pengujian maka dibuat suatu rancangan tampilan atau *Graphical User Interface* (GUI) seperti pada Gambar 3.6 berikut:



Gambar 3.6.Tampilan GUI

Masukan (*input*) sistem *traffic light* berupa hasil citra yang dihasilkan dari kamera berwarna dengan format RGB pada setiap objeknya, dengan tujuan pada identifikasi citra akan lebih mudah pada proses pencocokan *database*, pada panel 1 dan panel 2 adalah proses pengambilan suatu citra dan panel 3 dan panel 4 adalah hasil dari *image processing*, kemudian tombol *start* berfungsi untuk menjalankan video secara *real time* untuk proses pengambilan citra.

4. Hasil Pengujian dan Pembahasan

4.1. Pengujian fungsional

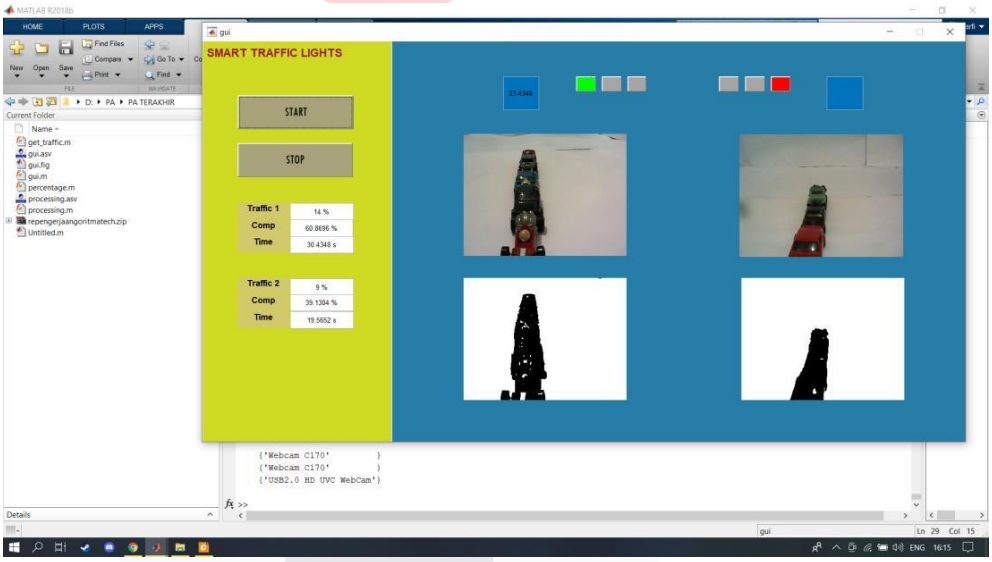
Dalam pengujian fungsional berikut dimaksudkan untuk memverifikasi tahapan-tahapan bekerjanya sistem apakah sesuai dengan tahapan yang digambarkan pada diagram alir Gambar 3.5 yang kondisinya dimonitor melalui urutan menyalanya lampu hijau kuning merah pada GUI yang juga dilakukan pencatatan selain urutan menyalanya lampu sesuai warnanya juga dicatat lamanya menyala.

Gambar 4.1 berikut adalah foto perangkat dalam percobaan sistem saat dilakukan pengujian fungsi. Skenario yang dilakukan adalah :

- Pertama menyiapkan antrian kendaraan pada arah jalan A sebanyak 1 kendaraan dan di arah jalan B sebanyak 7 kendaraan lalu sistem distart.
- Indikator warna lampu diamati pada tampilan di laptop (Gambar 4.2) baik urutan warna lampu yang menyala untuk arah jalan A dan B.

- Percobaan tersebut diulang-ulang dengan mengganti-ganti komposisi jumlah kendaraan pada arah A dan B : berturut-turut masing-masing sebagai berikut : 2 dan 6, 3 dan 5 dan seterusnya seperti pada Tabel 4.1.

Gambar 4.1 Foto pengujian fungsi



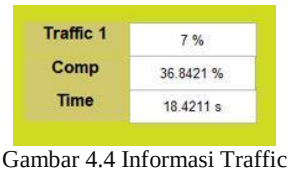
Gambar 4.2 Pengujian Fungsi

Dari gambar 4.2 terlihat hasil pengujian simulasi sistem kontrol *traffic light* dengan masukan berupa citra yang diperoleh dari hasil foto menggunakan *camera webcam* dan diproses oleh sistem. Pada hasil keluaran yang ditampilkan yaitu berupa panjang antrian dan lamanya waktu antrian kendaraan.

Gambar 4.3 Hasil Identifikasi Objek

Dari gambar 4.3 menunjukan gambar hasil identifikasi objek dan warna. Pada proses ini yaitu adanya proses *RGB to Gray* dan *Gray to Black and White* yang kemudian di *thresholding* untuk menentukan level

otomatis dengan metode *otsu* kemudian di *hole filling* dan terakhir dilakukan metode *dillation* dan *errosion* agar identifikasi objek berfungsi dengan baik.



Gambar 4.4 Informasi Traffic

Dari gambar 4.4 merupakan hasil *output* berupa informasi dari *traffic*, informasi traffic dikeluarkan menjadi 3 jenis traffic, Comp dan yang terakhir time atau waktu lamanya yang di dapat dari hasil satu antrian. Proses pengambilan informasi *traffic* dibuat dengan mengidentifikasi suatu objek warna selain warna putih kemudian akan di dapatkan hasil informasi *traffic*.

Data hasil pengujian dengan panjang antrian tetap, yaitu dengan mengganti-ganti komposisi kendaraan pada arah A dan B yang diperlihatkan pada Tabel 4.1 berikut ini :

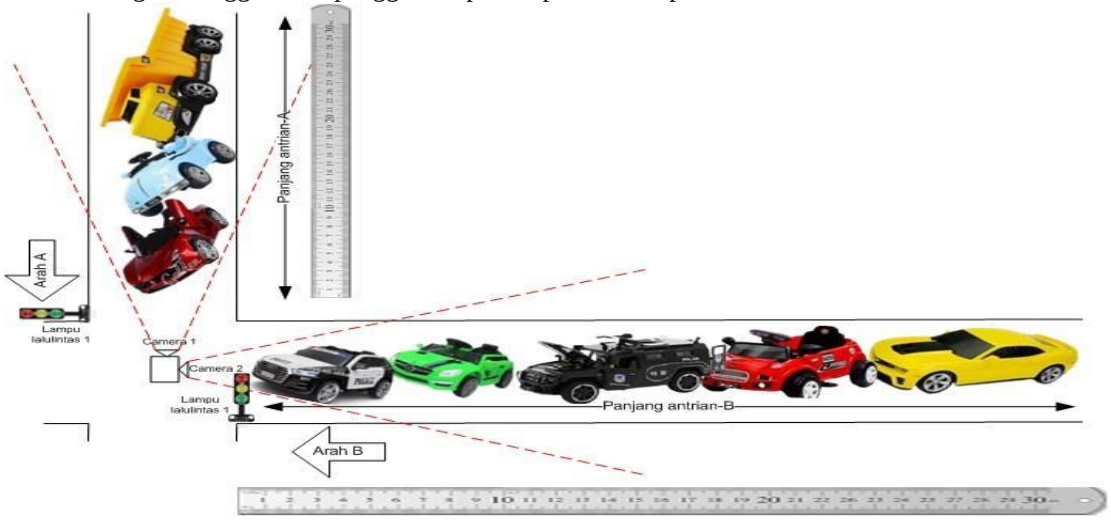
Tabel 4.1 Hasil pengujian fungsional

Urutan ke	Bnyak kendaraan (antrian)		Lampu yang menyala					
			Arah jalan A			Arah jalan B		
	A	B	Hijau	Kuning	Merah	Hijau	Kuning	Merah
1	1	7	8 s					
				3s				
					41 s	41 s		
							3s	
								8 s
2	2	6	14 s					
				3s				
					35 s	35 s		
							3s	
								14 s
3	3	5	16 s					
				3 s				
					33 s	33 s		
							3 s	
								16 s
4	4	4	30 s					
				3s				
					32 s	30 s		
							3s	
								30 s
5	5	3	31 s					
				3 s				
					19 s	19 s		
							3 s	
								31 s
6	6	2	29 s					
				3 s				
					20 s	20 s		
							3 s	
								29 s
7	7	1	41 s					
				3 s				
					8 s	8 s		
							3 s	
								41 s

Dari Tabel 4.1 di atas urutan waktu menyalanya lampu (hijau, kuning, merah) telah sesuai dengan yang telah direncanakan. Adapun akurasi lamanya waktu menyala lampu hijau akan dianalisis dalam pengujian berikut :

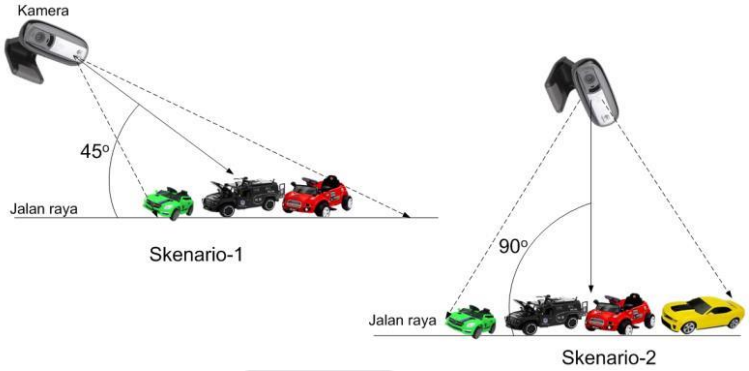
4.2. Pengujian akurasi perbandingan panjang antrian

Seperti dikemukakan pada Sub Bab 3.3 bahwa perbandingan jumlah piksel dijadikan sebagai pendekatan untuk merepresentasikan perbandingan panjang antrian. Maka berikut dilakukan pengujian akurasi antara perbandingan jumlah piksel arah A dan B dibandingkan hasil perbandingan panjang antrian arah A dan B dengan menggunakan penggaris seperti diperlihatkan pada Gambar 3.7 berikut:



Gambar 4.5 Pengujian Akurasi perbandingan Panjang Dengan Piksel

Dengan mengubah-ubah komposisi jumlah antrian pada arah A dan B kemudian program dijalankan untuk setiap komposisi dicatat perbandingan jumlah piksel (hasil pengolahan citra) dibandingkan dengan perbandingan panjang kendaraan yang diukur dengan menggunakan penggaris. Untuk mengetahui pengaruh sudut pengambilan gambar terhadap akurasi maka dilakukan dua skenario percobaan yaitu yang pertama dengan sudut arah kamera terhadap bidang horizontal sebesar 45 derajat dan skenario kedua adalah dengan menempatkan kamera tepat di atas kendaraan dengan posisi arah kamera tegak lurus (90 derajat) terhadap bidang jalan raya seperti diilustrasikan pada Gambar 4.3 berikut :



Gambar 4.6 Skenario pengujian akurasi jumlah pixel

Dibawah ini adalah hasil pengujian 2 skenario yang diperlihatkan pada 2 tabel sebagai berikut :

Tabel 4.2 Hasil pengujian akurasi Skenario 1

PERCOBAAN KE	JUMLAH KENDARAAN		JUMLAH PIXEL		PANJANG ANTRIAN		LAMA WAKTU LAMPU HIJAU				PENYIMPANGAN (DEVIASI) LAMA WAKTU NYALA LAMPU HIJAU	
							BERDASAR IMAGE PROCESSING		BERDASARKAN PENGUKURAN PANJANG ANTRIAN			
	A	B	A (pix)	B (pix)	A (cm)	B (cm)	A (detik)	B (detik)	A (detik)	B (detik)	A (detik)	B (detik)
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)
1	1	7	36140	103442	6	52	31.07	88.93	12.41	88.93	18.66	0
2	2	6	70380	88131	13	45	53.28	66.72	26.90	66.72	26.38	0
3	3	5	69056	101386	21	39	48.62	71.38	42.00	71.38	6.62	0
4	4	4	97746	67082	29	29	71.16	48.84	60.00	48.84	11.16	0
5	5	3	100618	68920	35	21	71.22	48.78	75.00	48.78	3.78	0
6	6	2	101952	63316	45	13	74.03	45.97	93.10	45.97	19.08	0
7	7	1	103442	36140	52	6	88.93	31.07	107.59	31.07	18.66	0
										RATA-RATA	**14.91**	**0**

Dari Tabel 4.2 di atas, akurasi lama waktu menyalanya lampu hijau berdasarkan metoda jumlah piksel dibandingkan dengan metoda pengukuran panjang antrian dapat dihitung sebagai berikut :

a. Kolom 8 adalah lama waktu lampu hijau menyala pada arah jalan A berdasarkan jumlah piksel di A = {(kolom 4)/(kolom 4 + kolom 5)} x 120 detik = 31,07 detik
b. Kolom 10 adalah lama waktu lampu hijau menyala pada arah jalan A berdasarkan panjang antrian di A = {(kolom 6)/(kolom 6 + kolom 7)} x 120 detik = 12,41 detik

Maka akurasi dari hasil a dibandingkan dengan nilai yang seharusnya yaitu b, terjadi penyimpangan (deviasi) sebesar 18,66 detik. Jika merujuk pada standar lama waktu lampu hijau selama 1 menit (yang umum) maka presentase penyimpangannya adalah (18,66 det/120 det) x 100% = 15,55%, Ini berarti akurasinya = 100% -15,55% = 84,45%

Dan seterusnya untuk arah jalan A dengan mengubah-ubah komposisi jumlah kendaraan pada arah jalan A (seperti pada Tabel 4.2) diperoleh deviasi rata-rata menyalanya lampu hijau untuk A = 14,91 detik aau (14,91/60)x 100% = 25%, atau akurasinya = 75% dan Dengan cara yang sama untuk arah jalan B diperoleh akurasi 100%.

Ditinjau dari deviasi (penyimpangan) lama waktu, ternyata untuk jalan arah A cukup besar penyimpangannya, yakni rata-rata 14,91 detik. Jika lama waktu 1 menit dianggapsebagai acuan, maka deviasi ini berarti sebesar 25% atau akurasi sebesar 75%. Sementara untuk jalan arah B deviasi/penyimpangan sebesar nol berarti akurasi 100%. Hal ini dapat disebabkan karena perbedaan kemiringan posisi kamera 1 dan 2 sehingga kendaraan nampak bertumpuk yang menghasilkan jumlah piksel lebih sedikit dari semestinya. Adapun idealnya adalah posisi tegak lurus obyek.

Tabel 4.3 Hasil pengujian akurasi Skenario 2

PERCOBAAN KE	JUMLAH KENDARAAN		JUMLAH PIXEL		PANJANG ANTRIAN		LAMA WAKTU LAMPU HIJAU				PENYIMPANGAN (DEVIASI) LAMA WAKTU NYALA LAMPU HIJAU	
							BERDASAR IMAGE PROCESSING		BERDASARKAN PENGUKURAN PANJANG ANTRIAN			
	A	B	A (pix)	B (pix)	A (cm)	B (cm)	A (detik)	B (detik)	A (detik)	B (detik)	A (detik)	B (detik)
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)
1	1	7	95194	256538	6	52	32,48	87,52	12.41	88.93	20,06	0
2	2	6	176970	301248	13	45	44,41	75,59	26.90	66.72	17,51	0
3	3	5	227361	291103	21	39	52,62	67,38	42.00	71.38	10,62	0
4	4	4	301169	340152	29	29	56,35	63,65	60.00	48.84	3,65	0
5	5	3	375411	274061	35	21	69,36	50,64	75.00	48.78	5,64	0
6	6	2	434486	247306	45	13	76,47	43,53	93.10	45.97	16,63	0
7	7	1	401615	217193	52	6	77,88	42,12	107.59	31.07	29,70	0
										RATA-RATA	**14.83**	**0**

Dari Tabel 4.2 di atas, akurasi lama waktu menyalanya lampu hijau berdasarkan metoda jumlah piksel dibandingkan dengan metoda pengukuran panjang antrian dapat dihitung sebagai berikut :

a. Kolom 8 adalah lama waktu lampu hijau menyala pada arah jalan A berdasarkan jumlah piksel di A = {(kolom 4)/(kolom 4 + kolom 5)} x 120 detik = 32,48 detik
b. Kolom 10 adalah lama waktu lampu hijau menyala pada arah jalan A berdasarkan panjang antrian di A = {(kolom 6)/(kolom 6 + kolom 7)} x 120 detik = 12,41 detik

Maka akurasi dari hasil a dibandingkan dengan nilai yang seharusnya yaitu b, terjadi penyimpangan (deviasi) sebesar 20,06 detik. Jika merujuk pada standar lama waktu lampu hijau selama 1 menit (yang umum) maka presentase penyimpangannya adalah (20,66 det/120 det) x 100% = 16,66%, Ini berarti akurasinya = 100%-16,66% = 83,34%

Dan seterusnya untuk arah jalan A dengan mengubah-ubah komposisi jumlah kendaraan pada arah jalan A (seperti pada Tabel 4.3) diperoleh deviasi rata-rata menyalanya lampu hijau untuk A = 14,83 detik aau (14,83/60)x 100% = 25%, atau akurasinya = 75% dan Dengan cara yang sama untuk arah jalan B diperoleh akurasi 100%.

Ditinjau dari deviasi (penyimpangan) lama waktu, ternyata untuk jalan arah A cukup besar penyimpangannya, yakni rata-rata 14,83 detik. Jika lama waktu 1 menit dianggapsebagai acuan, maka deviasi ini berarti sebesar 25% atau akurasi sebesar 75%. Sementara untuk jalan arah B deviasi/penyimpangan sebesar nol berarti akurasi 100%. Hal ini dapat disebabkan karena perbedaan kemiringan posisi kamera 1 dan 2 sehingga kendaraan nampak bertumpuk yang menghasilkan jumlah piksel lebih sedikit dari semestinya. Adapun idealnya adalah posisi tegak lurus obyek.

5. Kesimpulan dan Saran

Kesimpulan

Kesimpulan pada pengerjaan Proyek Akhir ini adalah:

1. Alat sistem pengaturan waktu lampu lalu lintas dapat berjalan sesuai dengan fungsi dan tahapannya baik urutan waktu menyalanya lampu maupun durasi/lamanya menyala lampu.
2. Pendekatan perbandingan jumlah piksel kendaraan untuk merepresentasikan perbandingan panjang antrian sebagai pendekatan untuk menentukan rasio lamanya lampu hijau menyala terbukti dapat digunakan dengan akurasi 75% untuk arah A dan 100% untuk arah B.

Saran

Saran berdasarkan pengerjaan Proyek Akhir ini adalah:

1. Untuk penelitian lebih lanjut agar dilakukan dengan kondisi *real* (melakukan *image processing* terhadap warna jalan raya yang asli dan kendaraan asli) dan dengan jumlah kamera 4 buah sesuai kebanyakan simpang jalan raya merupakan simpang empat.
2. Pengaruh sudut kemiringan kamera harus menjadi skenario pengujian.

REFERENSI

[1] J. Ilmiah *et al.*, “SIMULASI PENGATURAN LAMPU LALU LINTAS BERDASARKAN DATA IMAGE PROCESSING KEPADATAN KENDARAAN BERBASIS MIKROKONTROLER ATMEGA16 Sekolah Tinggi Manajemen Informatika dan Komputer ASIA Malang,” vol. 8, no. 2, pp. 81–96, 2014.

[2] H. Mulyawan, M. Z. H. Samsono, and Setiawardhana, “Identifikasi dan Tracking Objek Berbasis Image Processing Secara Real Time,” 2011.

[3] G. L. Djavendra, S. Aisyah, and E. R. Jamzuri, “Desain sistem pengatur lampu lalu lintas dengan identifikasi kepadatan kendaraan menggunakan metode subtraction,” vol. 7, no. 2, 2018.

[4] O. N. Shpakov and G. V. Bogomolov, “Technogenic activity of man and local sources of environmental pollution,” *Stud. Environ. Sci.*, vol. 17, no. C, pp. 329–332, 1981.

[5] M. Thresholding, D. Otsu, and S. Bhahri, “Transformasi Citra Biner Menggunakan,” vol. 7, no. 2, pp. 195–203.

[6] T. Edition and J. Wiley, “Morphological Image Processing Reading:,” *Image* (Rochester, N.Y.), vol. 1, pp. 1–4, 2001.

[7] H. K. Ramadhan, D. I. Savitri, and Y. Y. Santoso, “Prototipe Lampu Lalu Lintas dengan Pewaktuan Adaptif,” pp. 3–7, 1920.

[8] Sonia Dian Maniswari, Angga Rusdinar, ST., MT., Ph.d, Bedy Purnama, S.Si., MT, *Smart Traffic Light Menggunakan Image Processing Dan Metode Fuzzy Logic*, e-Proceeding of Engineering : Vol.2, No.2 Agustus 2015

[9] Fariz Muhammad Rachmatullah1, Gita Indah Hapsari2, Lisda Meisaroh3 *Sistem Pengaturan Lampu Lalu Lintas Menggunakan Pengolahan Citra Digital,* e-Proceeding of Applied Science : Vol.6, No.2 Desember 2020

[10] Rivo Dwi Yulianto1, Ir. Agus Ganda Permana, M.T.2, Dadan Nur Ramadan, S.Pd., M.T.3 *Perancangan Sistem Informasi Untuk Smart Traffic Light Berdasarkan Rute Darurat,* e-Proceeding of Applied Science : Vol.7, No.1 Februari 2021