BAB IV

PENGUJIAN DAN ANALISIS

Pada bab ini akan diuraikan tentang proses pengujian sistem yang akan dibuat. Pengujian akan dilakukan dengan 2 (dua) cara, yaitu pengujian secara parsial dan pengujian secara keseluruhan.

4.1 Pengujian Sistem Secara Parsial

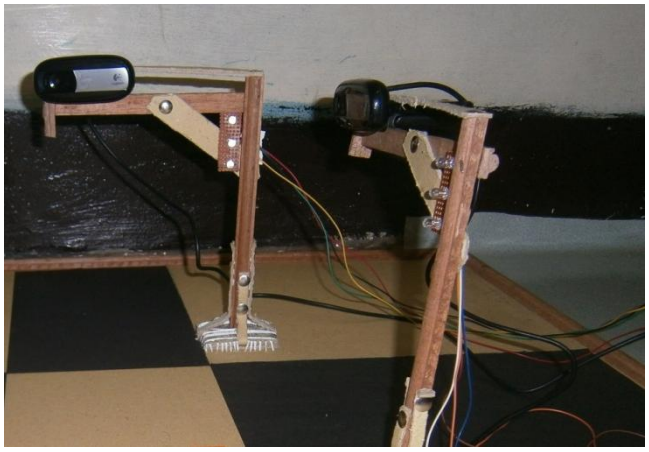
Pengujian secara parsial ini digunakan untuk mengetahui proses dari masing-masing sistem yang akan dibuat. Terdapat beberapa blok sistem yang akan dibuat, yaitu:

1. pemasangan kamera USB dan pengambilan gambar (*capture*),
2. *image processing* untuk mendeteksi jumlah kendaraan,
3. pengujian Logika *Fuzzy*,
4. penentuan kondisi-kondisi penyalaan lampu lalu lintas,
5. hubungan komunikasi serial dengan sistem minimum ATmega8535 dan lampu lalu lintas.

4.1.1 Pemasangan Kamera dan Pengambilan Gambar (*Capture*)

Tahap ini merupakan tahapan awal untuk membuat pengembangan algoritma pengolahan citra untuk mengontrol kepadatan lalu lintas pada persimpangan dua arah berbasis logika *Fuzzy*. Kamera dipasang di masing-masing jalur pada miniature yang sudah dibuat, yaitu di jalur 1 dan jalur 2. Letak kamera

berada di tengah-tengah jalur agar gambar yang ditangkap oleh kamera lebih baik dan lebih terfokus, seperti yang terlihat pada **Gambar 4.1** di bawah ini.



Gambar 4.1 Letak Kamera di Jalur 1 dan Jalur 2

Pengambilan gambar (*capture*) dilakukan agar mempermudah *image processing*. Hasil *capture* inilah yang menjadi inputan awal untuk melanjutkan ke tahap selanjutnya.



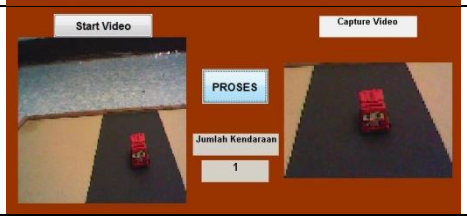
Gambar 4.2 Hasil Capture

4.1.2 *Image Processing* Untuk Mendeteksi Jumlah Kendaraan

Hasil dari pengujian *image processing* ini merupakan jumlah kendaraan yang dideteksi dari hasil *capture*. Hasil *capture* ini sudah di *crop* secara otomatis guna mengurangi *noise* yang ditangkap oleh kamera. Proses *image processing* yang dilakukan adalah perbandingan antara gambar *background* dan gambar yang sudah ada kendaraan. Untuk melakukan pengujian ini, terlebih dahulu kita mengambil gambar *background* agar mempermudah untuk mendeteksi jumlah

kendaraan. Hal ini juga dilakukan karena intensitas cahaya diberbagai tempat berbeda-beda. Berikut hasil pendeteksian kendaraan dari masing-masing hasil *capture*.

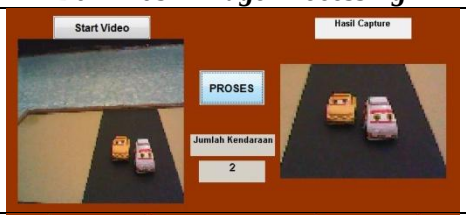
Tabel 4.1 Pendeteksian dengan Jumlah Kendaraan 1

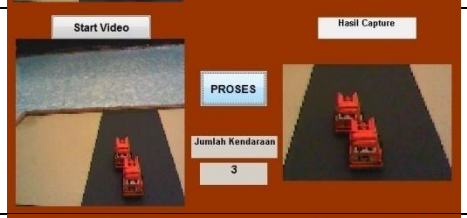
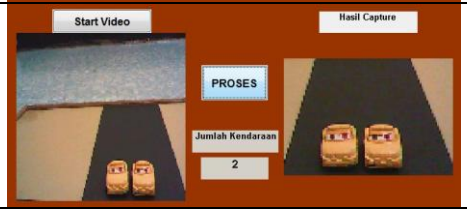
Capture	**Hasil *Capture* dan Jumlah Kendaraan Dari Hasil *Image Processing***	**Jumlah Kendaraan Yang Sebenarnya**
1		1
2		1
3		1
4		1
5		1
6		1
7		1

8			1
9			1
10			1

Hasil percobaan dari dilakukan 10 kali percobaan untuk pendeteksian kendaraan dengan jumlah kendaraan 1 memiliki tingkat keberhasilan 100%.

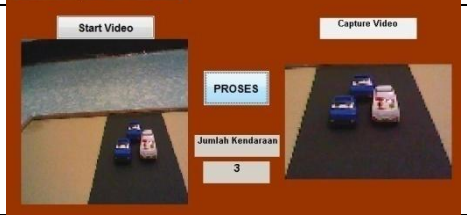
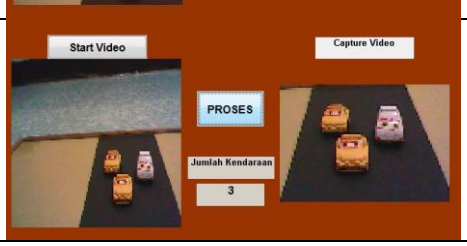
Tabel 4.2 Pendeteksian dengan Jumlah Kendaraan 2

Capture	**Hasil *Capture* dan Jumlah Kendaraan Dari Hasil *Image Processing***	**Jumlah Kendaraan Yang Sebenarnya**
1		2
2		2
3		2

4				2
5				2
6				2
7				2
8				2
9				2
10				2

Sama halnya dengan pendeteksian untuk jumlah kendaraan 1, hasil pendeteksian memiliki tingkat keberhasilan 100%

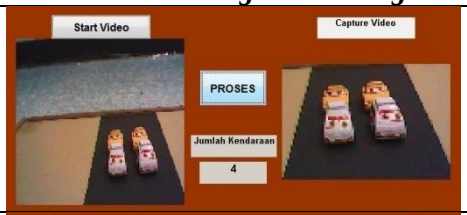
Tabel 4.3 Pendeteksian dengan Jumlah Kendaraan 3

Capture	Hasil *Capture* dan Jumlah Kendaraan Dari Hasil *Image Processing*	Jumlah Kendaraan Yang Sebenarnya
1		3
2		3
3		3
4		3
5		3
6		3
7		3

8			3
9			3
10			3

Hasil percobaan untuk jumlah kendaraan 3, memiliki tingkat keberhasilan 80% dalam 10 kali percobaan dengan *error* 20%. Ketidakakuratan pendeteksian ditunjukkan pada percobaan 1 dan percobaan 9. Hal ini dipengaruhi oleh intensitas cahaya, warna kendaraan, dan nilai *threshold* yang belum adaptif. *Error* deteksi untuk jumlah kendaraan 3 berkisar ±1.

Tabel 4.4 Pendeteksian dengan Jumlah Kendaraan 4

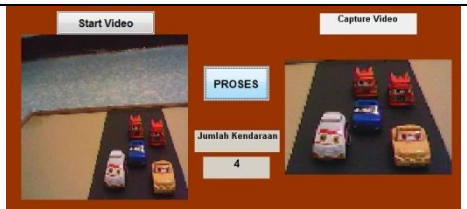
Capture	**Hasil *Capture* dan Jumlah Kendaraan Dari Hasil *Image Processing***	**Jumlah Kendaraan Yang Sebenarnya**
1		4
2		4

3				4
4				4
5				4
6				4
7				4
8				4
9				4
10				4

Sama halnya dengan jumlah kendaraan 3, untuk kendaraan jumlah 4 juga memiliki tingkat keberhasilannya 80% dari 10 kali percobaan dengan *error* 20%. Ketidakakuratan pendeteksian ditunjukkan pada percobaan 5 dan percobaan 9. Hal ini dipengaruhi juga oleh intensitas cahaya, warna kendaraan, dan nilai *threshold* yang belum adaptif. *Error* deteksi untuk jumlah kendaraan 3 berkisar ±1.

Tabel 4.5 Pendeteksian dengan Jumlah Kendaraan 5

Capture	**Hasil *Capture* dan Jumlah Kendaraan Dari Hasil *Image Processing***	**Jumlah Kendaraan Yang Sebenarnya**
1		5
2		5
3		5
4		5
5		5

6				5
7				5
8				5
9				5
10				5

Hasil percobaan untuk jumlah kendaraan 5, tingkat keberhasilannya 90% dalam 10 kali percobaan dengan *error*-nya 10%. Ketidakakuratan pendeteksian ditunjukkan pada percobaan 6. Hali ini dipengaruhi oleh oleh intensitas cahaya, warna kendaraan, dan nilai *threshold* yang belum adaptif. *Error* deteksi untuk jumlah kendaraan 3 berkisar ±1.

Tabel 4.6 Pendeteksian dengan Jumlah Kendaraan 6

Capture	Hasil *Capture* dan Jumlah Kendaraan Dari Hasil *Image Processing*	Jumlah Kendaraan Yang Sebenarnya
1		6
2		6
3		6
4		6
5		6
6		6
7		6

8		6
9		6
10		6

Hasil percobaan untuk jumlah kendaraan 6, memiliki tingkat keberhasilannya 90% dari 10 kali percobaan dengan *error* 10%. Ketidakakuratan pendeteksian ditunjukkan pada percobaan 10. Hal ini dipengaruhi oleh intensitas cahaya, warna kendaraan, dan nilai *threshold* yang belum adaptif. *Error* deteksi untuk jumlah kendaraan 3 berkisar ±1.

Tabel 4.7 Pendeteksian dengan Jumlah Kendaraan 7

Capture	**Hasil *Capture* dan Jumlah Kendaraan Dari Hasil *Image Processing***	**Jumlah Kendaraan Yang Sebenarnya**
1		7
2		7

3				7
4				7
5				7
6				7
7				7
8				7
9				7
10				7

Hasil percobaan untuk jumlah kendaraan 7, memiliki tingkat keberhasilannya 80% dari 10 kali percobaan dengan *error* 20%. Ketidakakuratan pendeteksian ditunjukkan pada percobaan 5 dan percobaan 7. Hal ini dipengaruhi oleh intensitas cahaya, warna kendaraan, dan nilai *threshold* yang belum adaptif. *Error* deteksi untuk jumlah kendaraan 3 berkisar ±1.

Tabel 4.8 Pendeteksian dengan Jumlah Kendaraan 8

Capture	**Hasil *Capture* dan Jumlah Kendaraan Dari Hasil *Image Processing***	**Jumlah Kendaraan Yang Sebenarnya**
1		8
2		8
3		8
4		8
5		8

6				8
7				8
8				8
9				8
10				8

Hasil percobaan untuk jumlah kendaraan 8, memiliki tingkat keberhasilannya 80% dari 10 kali percobaan dengan *error* 20%. Ketidakakuratan pendeteksian ditunjukkan pada percobaan 3 dan percobaan 8. Hal ini dipengaruhi oleh intensitas cahaya, warna kendaraan, dan nilai *threshold* yang belum adaptif. *Error* deteksi untuk jumlah kendaraan 3 berkisar ±1.

Tabel 4.9 Pendeteksian dengan Jumlah Kendaraan 9

Capture	Hasil *Capture* dan Jumlah Kendaraan Dari Hasil *Image Processing*	Jumlah Kendaraan Yang Sebenarnya
1		9
2		9
3		9
4		9
5		9
6		9
7		9

8		9
9		9
10		9

Hasil percobaan untuk jumlah kendaraan 9, memiliki tingkat keberhasilannya 80% dari 10 kali percobaan dengan *error* 20%. Ketidakakuratan pendeteksian ditunjukkan pada percobaan 2 dan percobaan 8. Hal ini dipengaruhi oleh intensitas cahaya, warna kendaraan, dan nilai *threshold* yang belum adaptif. *Error* deteksi untuk jumlah kendaraan 3 berkisar ±1.

Tabel 4.10 Pendeteksian dengan Jumlah Kendaraan 10

Capture	**Hasil *Capture* dan Jumlah Kendaraan Dari Hasil *Image Processing***	**Jumlah Kendaraan Yang Sebenarnya**
1		10
2		10

3				10
4				10
5				10
6				10
7				10
8				10
9				10
10				10

Hasil percobaan untuk jumlah kendaraan 10, memiliki tingkat keberhasilannya 70% dari 10 kali percobaan dengan *error* 30%. Ketidakakuratan pendeteksian ditunjukkan pada percobaan 4, 6, dan percobaan 9. Hal ini dipengaruhi oleh intensitas cahaya, warna kendaraan, dan nilai *threshold* yang belum adaptif. *Error* deteksi untuk jumlah kendaraan 3 berkisar ±1.

Dari pengujian *image processing* di atas dapat dilihat seberapa bagus atau seberapa akuratnya pendeteksian kendaraan yang dihasilkan. Dengan melakukan 10 kali percobaan dari masing-masing jumlah kendaraan yang berbeda-beda, hasil pendeteksian yang didapat juga berbeda-beda. Berikut tingkat keberhasilan dan *error* dari hasil deteksi kendaraan.

Tabel 4.11 Tingkat Keberhasilan Deteksi Kendaraan

Jumlah Kendaraan	**Tingkat Keberhasilan (%)**	***Error* (%)**
1	100	0
2	100	0
3	80	20
4	80	20
5	90	10
6	90	10
7	80	20
8	80	20
9	80	20
10	70	30
Rata-rata Tingkat Keberhasilan = 85 %		
***Error* = 15 %**		

Dari tabel hasil pendeteksian di atas terlihat bahwa tingkat keberhasilan belum semuanya 100 %. Ada yang 100%, 90%, 80%, bahkan 70%. Dimana semakin banyak jumlah kendaraan, *error* dalam pendeteksian semakin besar. Hal ini dipengaruhi oleh beberapa hal, antara lain: intensitas cahaya yang berubah-ubah, warna kendaraan, posisi kendaraan yang tidak teratur, nilai *threshold* yang

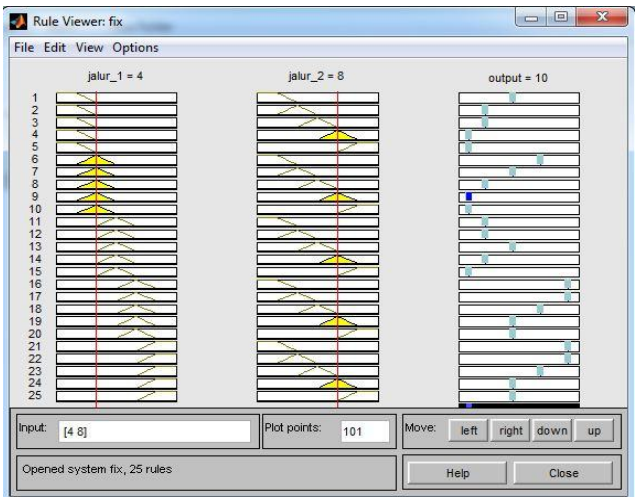
belum adaptif (masih manual). Rata-rata tingkat keberhasilan dari 10 kali percobaan dengan jumlah kendaraan berbeda-beda memiliki tingkat keberhasilan 85% dan *error*-nya 15% dengan *error detection* berkisar ±1.

4.1.3 Pengujian Logika *Fuzzy*

Pengujian logika *Fuzzy* ini bertujuan untuk mengetahui bagaimana proses logika *Fuzzy* untuk mendeteksi kondisi-kondisi kepadatan berdasarkan jumlah kendaraan yang terdeteksi. Dan output dari proses logika *Fuzzy* ini digunakan sebagai penentuan kondisi-kondisi penyalaan lampu lalu lintas yang akan di buat. Pengujian dari logika *Fuzzy* ini dilakukan dengan membandingkan *output* dari 3 cara yang digunakan yaitu dengan menggunakan simulasi MATLAB, dengan program yang dibuat, dan pengujian perhitungan manual. Setiap *input*-an akan diproses secara bertahap mulai dari mendeteksi *membership function* hingga menghasilkan *output*.

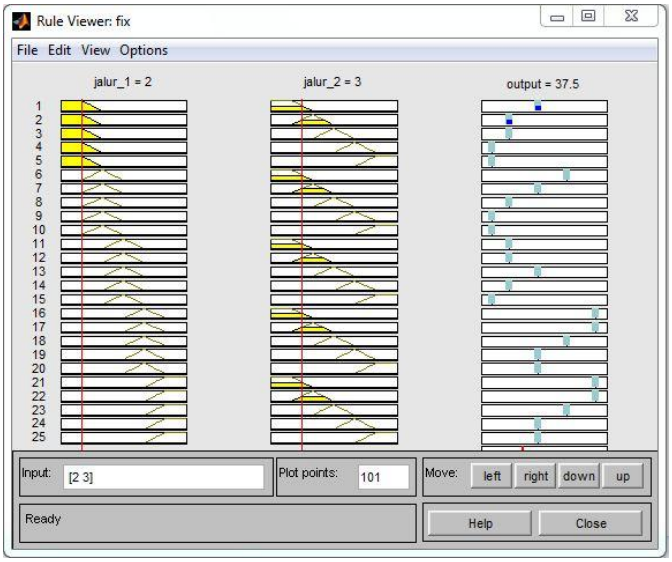
a. Hasil pengujian logika *Fuzzy* menggunakan simulasi MATLAB:

- Jika jalur_1 = 4 dan jalur_2 = 8



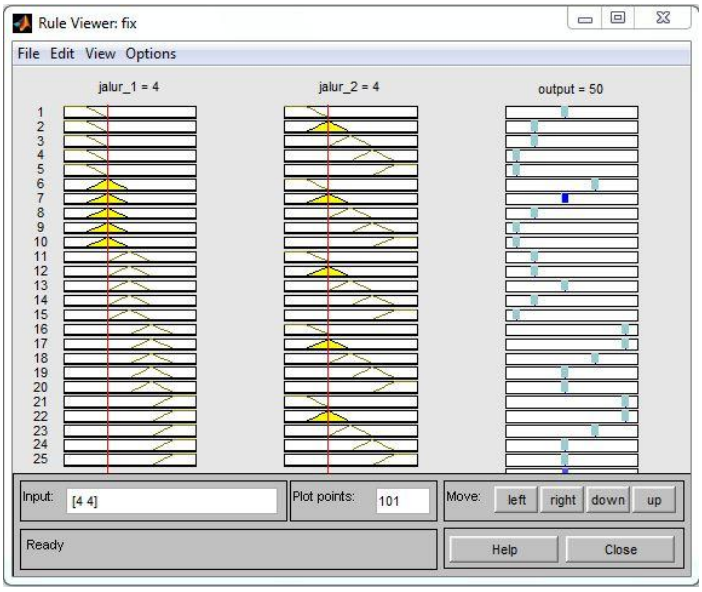
Gambar 4.3 Hasil Simulasi MATLAB pada Pengujian Logika *Fuzzy* ke-1

- Jika jalur_1 = 2 dan jalur_2 = 3



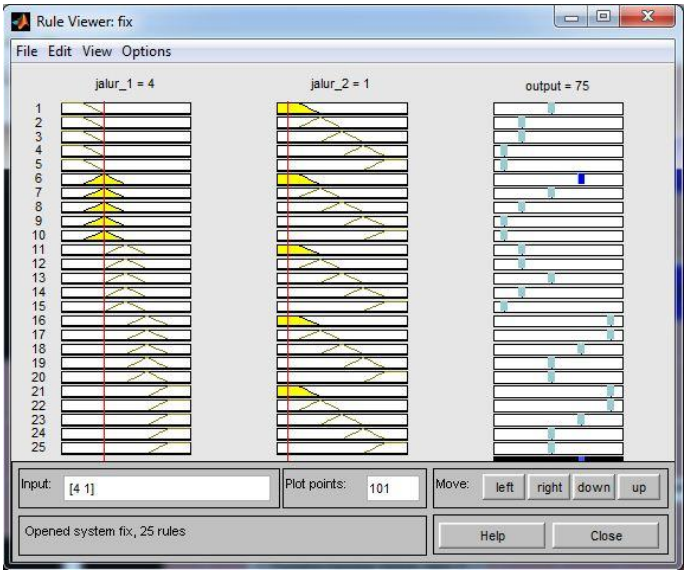
Gambar 4.4 Hasil Simulasi MATLAB pada Pengujian Logika *Fuzzy* ke-2

- Jika jalur_1 = 4 dan jalur_2 = 4



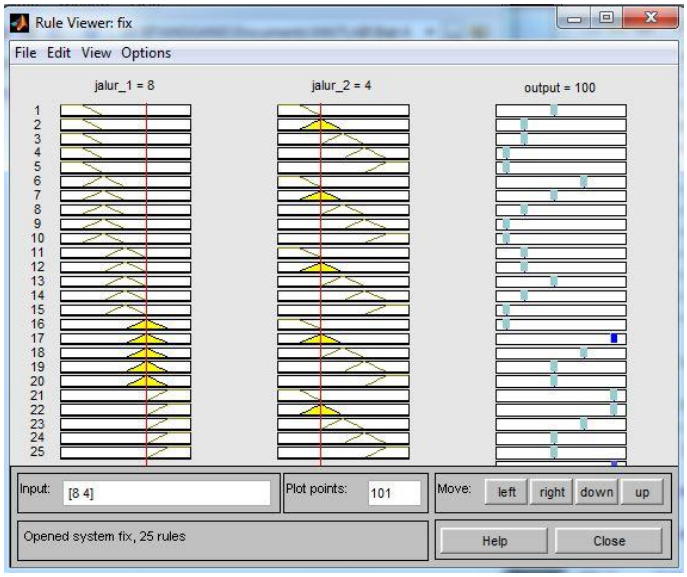
Gambar 4.5 Hasil Simulasi MATLAB pada Pengujian Logika *Fuzzy* ke-3

- Jika jalur_1 = 4 dan jalur_2 = 1



Gambar 4.6 Hasil Simulasi MATLAB pada Pengujian Logika *Fuzzy* ke-4

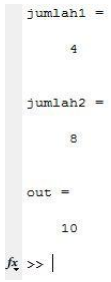
- Jika jalur_1 = 8 dan jalur_2 = 4



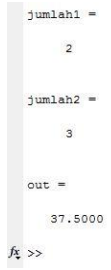
Gambar 4.7 Hasil Simulasi MATLAB pada Pengujian Logika *Fuzzy* ke-5

b. Hasil pengujian Logika *Fuzzy* yang dibuat menggunakan software MATLAB:

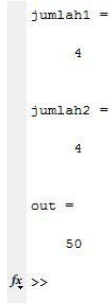
- Pengujian ke-1, jika jalur_1 = 4 dan jalur_2 = 8



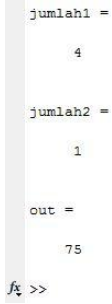
- Pengujian ke-2, jika jalur_1 = 2 dan jalur_2 = 3



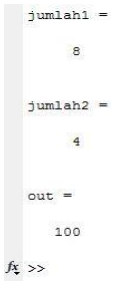
- Pengujian ke-3, jika jalur_1 = 4 dan jalur_2 = 4



- Pengujian ke-4, jika jalur_1 = 4 dan jalur_2 = 1



- Pengujian ke-5, jika jalur_1 = 8 dan jalur_2 = 4



c. Hasil pengujian dengan perhitungan manual:

- Pengujian ke-1, jika *input* jalur_1 = 4 dan jalur_2 = 8

Jalur_1	**Jalur_2**
Input = 4	*Input* = 8
SS = 0	SS2 = 0
S = $\frac{(4-2)}{(4-2)} = \frac{2}{2} = 1$	S2 = 0
SD = 0	SD2 = 0
P = 0	P2 = $\frac{(8-6)}{(8-6)} = \frac{2}{2} = 1$
SP = 0	SP2 = 0

Rule Fuzzy

Tabel 4.12 Rule Pengujian Manual Logika *Fuzzy* ke-1

Jalur 1 / **Jalur 2**	**SS2**	**S2**	**SD2**	**P2**	**SP2**
SS	0	0	0	0	0
S	0	0	0	1	0
SD	0	0	0	0	0
P	0	0	0	0	0
SP	0	0	0	0	0

Defuzzification

$$out = \big((0*50)+(0*25)+(0*25)+(0*10)+(0*10)+(0*75)+(0*50)+(0*25)+(1*10)+(0*10)+(0*25)+(0*25)+(0*50)+(0*25)+(0*10)+(0*10)+(0*100)+(0*75)+(0*50)+(0*50)+(0*100)+(0*100)+(0*75)+(0*50)+(0*50)\big)/(0+0+0+0+0+0+0+1+0+0+0+0+0+0+0+0+0+0+0+0+0+0+0+0)$$

$$= 10$$

- Pengujian ke-2, jika *input* jalur_1 = 2 dan jalur_2 = 3

Jalur_1	**Jalur_2**
Input = 2	*Input* = 3
SS = 1	SS2 = $\frac{(4-3)}{(4-2)} = \frac{1}{2} = 0{,}5$
S = 0	S2 = $\frac{(4-3)}{(4-2)} = \frac{1}{2} = 0{,}5$
SD = 0	SD2 = 0
P = 0	P2 = 0
SP = 0	SP2 = 0

Rule Fuzzy

Tabel 4.13 Rule Pengujian Manual Logika *Fuzzy* ke-2

Jalur 1 / **Jalur 2**	**SS2**	**S2**	**SD2**	**P2**	**SP2**
SS	0,5	0,5	0	0	0
S	0	0	0	0	0
SD	0	0	0	0	0
P	0	0	0	0	0
SP	0	0	0	0	0

Defuzzification

$$out = \big((0{,}5 * 50) + (0{,}5 * 25) + (0 * 25) + (0 * 10) + (0 * 10) + (0 * 75) + (0 * 50) + (0 * 25) + (0 * 25) + (0 * 10) + (0 * 10) + (0 * 25) + (0 * 25) + (0 * 50) + (0 * 25) + (0 * 10) + (0 * 10) + (0 * 100) + (0 * 75) + (0 * 50) + (0 * 50) + (0 * 100) + (0 * 100) + (0 * 75) + (0 * 50) + (0 * 50)\big)/(0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 1 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0 + 0) = 37{,}5$$

- Pengujian ke-3, jika *input* jalur_1 = 4 dan jalur_2 = 4

Jalur_1	**Jalur_2**
Input = 4	*Input* = 4
SS = 0	SS2 = 0
S = $\frac{(6-4)}{(6-4)} = \frac{2}{2} = 1$	S2 = $\frac{(6-4)}{(6-4)} = \frac{2}{2} = 1$
SD = 0	SD2 = 0
P = 0	P2 = 0
SP = 0	SP2 = 0

Rule Fuzzy

Tabel 4.14 Rule Pengujian Manual Logika *Fuzzy* ke-3

Jalur 1 / **Jalur 2**	**SS2**	**S2**	**SD2**	**P2**	**SP2**
SS	0	0	0	0	0
S	0	1	0	0	0
SD	0	0	0	0	0
P	0	0	0	0	0
SP	0	0	0	0	0

Defuzzification

$$out = \big((0*50)+(0*25)+(0*25)+(0*10)+(0*10)+(0*75)+(1*50)+(0*25)+(0*10)+(0*10)+(0*25)+(0*25)+(0*50)+(0*25)+(0*10)+(0*10)+(0*100)+(0*75)+(0*50)+(0*50)+(0*100)+(0*100)+(0*75)+(0*50)+(0*50)\big)/(0+0+0+0+0+0+0+1+0+0+0+0+0+0+0+0+0+0+0+0+0+0+0+0) = 50$$

- Pengujian ke-4, jika *input* jalur_1 = 4 dan jalur_2 = 1

Jalur_1	**Jalur_2**
Input = 4	*Input* = 1
SS = 0	SS2 = 1
S = $\frac{(6-4)}{(6-4)} = \frac{2}{2} = 1$	S2 = 0
SD = 0	SD2 = 0
P = 0	P2 = 0
SP = 0	SP2 = 0

Rule Fuzzy

Tabel 4.15 Rule Pengujian Manual Logika *Fuzzy* ke-4

Jalur 1 / **Jalur 2**	**SS2**	**S2**	**SD2**	**P2**	**SP2**
SS	0	0	0	0	0
S	1	0	0	0	0
SD	0	0	0	0	0
P	0	0	0	0	0
SP	0	0	0	0	0

Defuzzification

$$out = \big((0*50)+(0*25)+(0*25)+(0*10)+(0*10)+(1*75)+(0*50)+(0*25)+(0*10)+(0*10)+(0*25)+(0*25)+(0*50)+(0*25)+(0*10)+(0*10)+(0*100)+(0*75)+(0*50)+(0*50)+(0*100)+(0*100)+(0*75)+(0*50)+(0*50)\big)/(0+0+0+0+0+0+0+1+0+0+0+0+0+0+0+0+0+0+0+0+0+0+0+0)$$

$$= 75$$

- Pengujian ke-5, jika *input* jalur_1 = 8 dan jalur_2 = 4

Jalur_1	**Jalur_2**
Input = 8	*Input* = 4
SS = 0	SS2 = 0
S = 0	S2 = $\frac{(6-4)}{(6-4)} = \frac{2}{2} = 1$
SD = 0	SD2 = 0
P = $\frac{(8-6)}{(8-6)} = \frac{2}{2} = 1$	P2 = 0
SP = 0	SP2 = 0

Rule Fuzzy

Tabel 4.16 Rule Pengujian Manual Logika *Fuzzy* ke-5

Jalur 1 / **Jalur 2**	**SS2**	**S2**	**SD2**	**P2**	**SP2**
SS	0	0	0	0	0
S	0	0	0	0	0
SD	0	0	0	0	0
P	0	1	0	0	0
SP	0	0	0	0	0

Defuzzification

$$out = \big((0*50)+(0*25)+(0*25)+(0*10)+(0*10)+(0*75)+(0*50)+(0*25)+(0*10)+(0*10)+(0*25)+(0*25)+(0*50)+(0*25)+(0*10)+(0*10)+(1*100)+(0*75)+(0*50)+(0*50)+(0*100)+(0*100)+(0*75)+(0*50)+(0*50)\big)/(0+0+0+0+0+0+0+1+0+0+0+0+0+0+0+0+0+0+0+0+0+0+0+0)$$

$$= 100$$

Dari ketiga hasil pengujian logika *Fuzzy* di atas dapat dilihat hasil perbandingan dari setiap *output* yang dihasilkan. Berikut perbandingan hasil pengujian dari ketiga cara yang dilakukan:

Tabel 4.17 Hasil Pebandingan Pengujian Logika *Fuzzy*

No	Input		Output		
	Jalur_1	Jalur_2	Simulasi MATLAB	MATLAB	Perhitungan Manual
1	4	8	10	10	10
2	2	3	37,5	37,5	37,5
3	4	4	50	50	50
4	4	1	75	75	75
5	8	4	100	100	100

Dari tabel hasil pengujian logika *Fuzzy* di atas terlihat bahwa *output* dari program yang dibuat menggunakan MATLAB bernilai sama dengan simulasi MATLAB maupun perhitungan manual.

4.1.4 Penentuan Kondisi-Kondisi Penyalaan Lampu Lalu Lintas

Setelah proses pengujian logika *Fuzzy* selesai, *output* dari proses *defuzzyfikasi* akan digunakan untuk penentuan nyala lampu lalu lintas. Lama nyala

lampu lalu lintas tergantung dari kondisi yang didapat. Berikut kondisi-kondisi nyala lampu lalu lintas dan waktu penyalaannya berdasarkan *output defuzzyfikasi*.

- Kondisi 1 apabila, ***output defuzzifikasi* ≥ 5 dan *output defuzzifikasi* < 21**



Waktu yang sudah ditentukan:

```
if status == 'kondisi_1'
     M_1 = 13;          % Lampu Merah 1
     H_2 = 10;          % Lampu Hijau 2
     K_1 = 2;           % Lampu Kuning 1
     K_2 = 2;           % Lampu Kuning 2
     M_2 = 73;          % Lampu Merah 2
     H_1 = 70;          % Lampu Hijau 1
```

- Kondisi 2 apabila, ***output defuzzifikasi* ≥ 21 dan *output defuzzifikasi* < 41**



Waktu yang sudah ditentukan:

```
if status == 'kondisi_2'
     M_1 = 23;          % Lampu Merah 1
     H_2 = 20;          % Lampu Hijau 2
     K_1 = 2;           % Lampu Kuning 1
     K_2 = 2;           % Lampu Kuning 2
     M_2 = 53;          % Lampu Merah 2
     H_1 = 50;          % Lampu Hijau 1
```

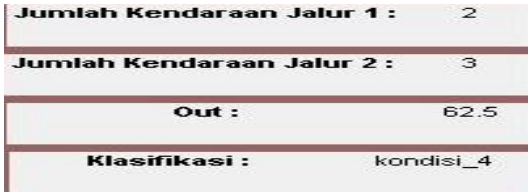
- Kondisi 3 apabila, ***output defuzzifikasi* ≥ 41 dan *output defuzzifikasi* < 61**



Waktu yang sudah ditentukan:

```
if status == 'kondisi_3'
     M_1 = 43;          % Lampu Merah 1
     H_2 = 40;          % Lampu Hijau 2
     K_1 = 2;           % Lampu Kuning 1
     K_2 = 2;           % Lampu Kuning 2
     M_2 = 43;          % Lampu Merah 2
     H_1 = 40;          % Lampu Hijau 1
```

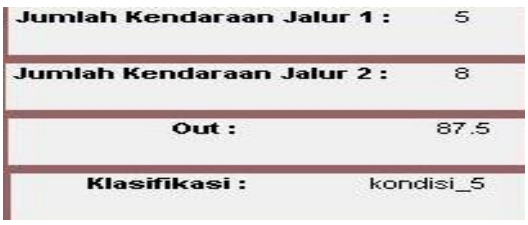
- Kondisi 4 apabila, ***output defuzzifikasi* ≥ 61 dan *output defuzzifikasi* < 81**



Waktu yang sudah ditentukan:

```
if status == 'kondisi_4'
     M_1 = 53;          % Lampu Merah 1
     H_2 = 50;          % Lampu Hijau 2
     K_1 = 2;           % Lampu Kuning 1
     K_2 = 2;           % Lampu Kuning 2
     M_2 = 23;          % Lampu Merah 2
     H_1 = 20;          % Lampu Hijau 1
```

- Kondisi 5 apabila, ***output defuzzifikasi* ≥ 81 dan *output defuzzifikasi* < 101**



Waktu yang sudah ditentukan:

```
if status == 'kondisi_5'
```

```
M_1 = 73;          % Lampu Merah 1
H_2 = 70;          % Lampu Hijau 2
K_1 = 2;           % Lampu Kuning 1
K_2 = 2;           % Lampu Kuning 2
M_2 = 13;          % Lampu Merah 2
H_1 = 10;          % Lampu Hijau 1
```

Dari penjelasan yang sebelumnya dijelaskan bahwa kondisi-kondisi penyalaan lampu lalu lintas tersebut dipilih berdasarkan *output defuzzyfikasi* yang dilakukan saat proses *Fuzzy*. Dan *output defuzzyfikasi* sendiri didapat dari jumlah kendaraan yang terdeteksi di masing-masing jalur dari hasil *image processing*. Kondisi-kondisi inilah yang nantinya dikirim ke sistem minimum ATmega8535 menggunakan komunikasi serial RS232 untuk menyalakan lampu lalu lintas.

4.1.5 Hubungan Komunikasi Serial dengan Sistem Minimum ATmega8535 dan Lampu Lalu Lintas

Tahapan ini menjadi tahapan terakhir dalam pembuatan sistem pengembangan algoritma pengolahan citra untuk mengontrol kepadatan lalu lintas pada persimpangan dua arah berbasis logika *Fuzzy* ini. Program MATLAB yang dikirim ke sistem minimum ATmega8535 adalah berupa data dari kondisi-kondisi penyalaan lampu lalu lintas. Berikut data yang akan dikirim ke sistem minimum ATmega8535 dari kondisi-kondisi penyalaan lampu lalu lintas

```
coba_serial = serial(PORT);
data1='a';
data2='b';
data3='c';
data4='d';
data5='e';

      if status == 'kondisi_1'
         fprintf(coba_serial, '%s', data1);

      elseif status == 'kondisi_2'
```

```
        fprintf(coba_serial, '%s', data2);

    elseif status == 'kondisi_3'
        fprintf(coba_serial, '%s', data3);

    elseif status == 'kondisi_4'
        fprintf(coba_serial, '%s', data4);

    elseif status == 'kondisi_5'
        fprintf(coba_serial, '%s', data5);
    end
```

Data untuk penyalaan lampu lalu lintas yang diterima oleh sistem minimum ATmega8535 akan diolah sesuai dengan data yang masuk. Komunikasi serial RS232 dengan sistem minimum ATmega8535 memiliki hubungan yang saling berkaitan. Hal ini bertujuan agar nyala lampu lalu lintas di *GUI* (*Guide User Interface*) MATLAB dengan nyala lampu lalu lintas di *hardware* menyala secara bersamaan. Berikut keterkaitan penyalaan lampu lalu lintas di MATLAB dengan penyalaan lampu lalu lintas di sistem minimum ATmega8535.

PORTC.0 = merah 1 PORTC.5 = merah 2
PORTC.1 = kuning 1 PORTC.6 = kuning 2
PORTC.2 = hijau 1 PORTC.7 = hijau 2

```
case 'a':
PORTC.0=0;PORTC.7=0;delay_ms(10000);
PORTC.7=1;PORTC.6=0;delay_ms(2000);
PORTC.6=1;PORTC.5=0;delay_ms(1000);
PORTC.0=1;

PORTC.2=0;delay_ms(60000);delay_ms(10
000);PORTC.2=1;
PORTC.1=0;delay_ms(2000);PORTC.1=1;
PORTC.0=0;delay_ms(1000);
        break;

case 'b':
PORTC.0=0;PORTC.7=0;delay_ms(20000);
PORTC.7=1;PORTC.6=0;delay_ms(2000);
PORTC.6=1;PORTC.5=0;delay_ms(1000);
PORTC.0=1;

PORTC.2=0;delay_ms(50000);PORTC.2=1;
PORTC.1=0;delay_ms(2000);PORTC.1=1;
PORTC.0=0;delay_ms(1000);
        break;
```

```
case 'c':
PORTC.0=0;PORTC.7=0;delay_ms(40000);
PORTC.7=1;PORTC.6=0;delay_ms(2000);
PORTC.6=1;PORTC.5=0;delay_ms(1000);
PORTC.0=1;

PORTC.2=0;delay_ms(40000);PORTC.2=1;
PORTC.1=0;delay_ms(2000);PORTC.1=1;
PORTC.0=0;delay_ms(1000);
        break;

case 'd':
PORTC.0=0;PORTC.7=0;delay_ms(50000);
PORTC.7=1;PORTC.6=0;delay_ms(2000);
PORTC.6=1;PORTC.5=0;delay_ms(1000);
PORTC.0=1;

PORTC.2=0;delay_ms(20000);PORTC.2=1;
PORTC.1=0;delay_ms(2000);PORTC.1=1;
PORTC.0=0;delay_ms(1000);
        break;
```

```
case 'e':
PORTC.0=0;PORTC.7=0;delay_ms(60000);        PORTC.2=0;delay_ms(10000);PORTC.2=1;
delay_ms(10000);                            PORTC.1=0;delay_ms(2000);PORTC.1=1;
PORTC.7=1;PORTC.6=0;delay_ms(2000);         PORTC.0=0;delay_ms(1000);
PORTC.6=1;PORTC.5=0;delay_ms(1000);             break;
PORTC.0=1;
```



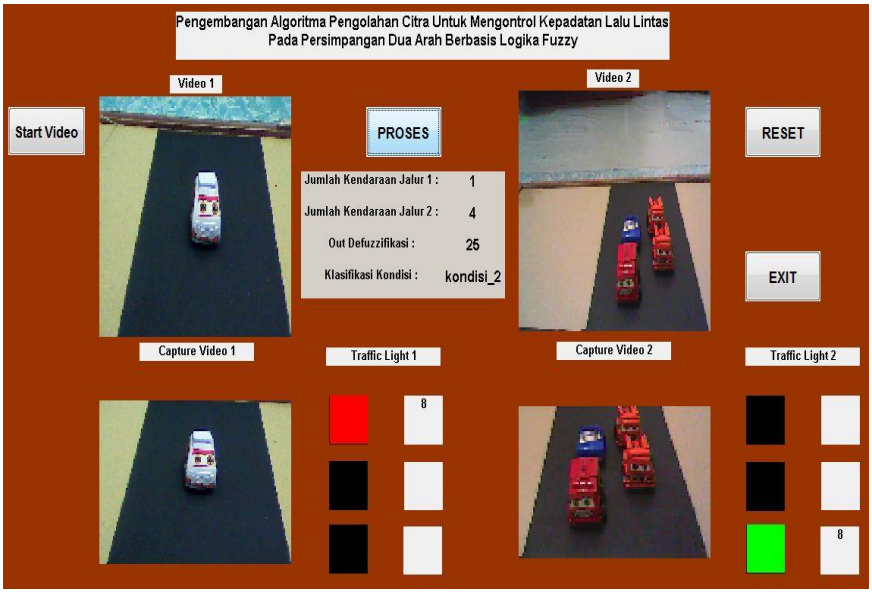
Gambar 4.8 Penyalaan Lampu Lalu Lintas di Jalur 1 dan di Jalur 2

Gambar diatas merupakan nyala lampu lalu lintas dari sistem yang dibuat. Dimana sudah dijelaskan sebelumnya bahwa hasil dari proses *defuzzifikasi* digunakan untuk menentukan kondisi-kondisi penyalaan lampu lalu lintas. Lamanya waktu penyalaan lampu lalu lintas ini tergantung dari kondisi yang dikirim ke sistem minimum ATmega8535 melalui komunikasi serial RS232.

4.2 Pengujian Sistem Secara Keseluruhan

Pengujian sistem secara keseluruhan ini dilakukan untuk mengetahui kemampuan kerja sistem secara umum. Pengujian dilakukan dengan cara menyamakan hasil keluaran lampu lalu lintas yang ada di MATLAB dengan hasil keluaran lampu lalu lintas di *hardware* .

a. Hasil lampu lalu lintas di MATLAB dan di miniature saat jalur 1 merah dan jalur 2 hijau.

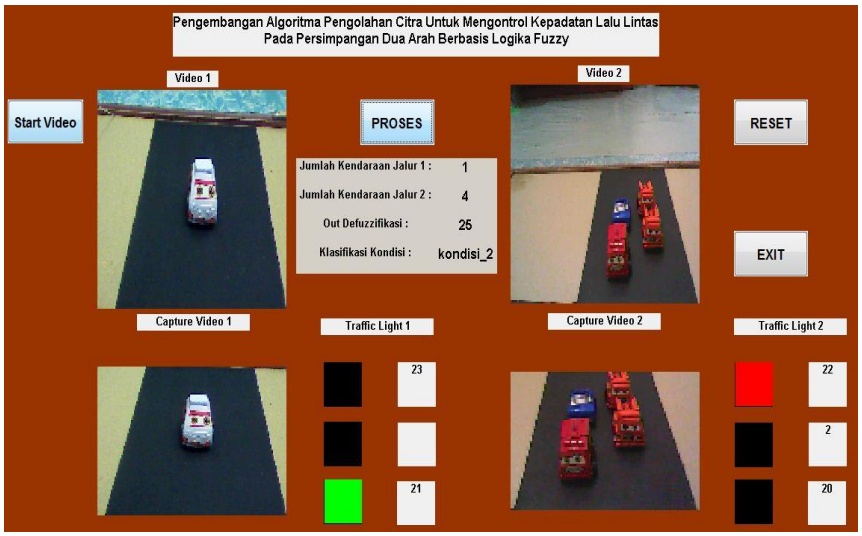


Gambar 4.9 Hasil Lampu Lalu Lintas di MATLAB



Gambar 4.10 Hasil Lampu Lalu Lintas di Miniature

b. Hasil lampu lalu lintas di MATLAB dan di miniature saat jalur 1 hijau dan jalur 2 merah.



Gambar 4.11 Hasil Lampu Lalu Lintas di MATLAB



Gambar 4.12 Hasil Lampu Lalu Lintas di Miniature

Dari pengujian keseluruhan di atas dapat dilihat bahwa nyala lampu lalu lintas di MATLAB dan di miniature sudah sesuai. Saat di MATLAB jalur 1 lampu merah dan jalur 2 lampu hijau, maka di miniature jalur 1 lampu merah dan jalur 2 lampu hijau juga. Dan sebaliknya.