

BAB IV

PENGUJIAN SISTEM

4.1 Pengujian Perangkat Lunak

Diperlukan tahap pengujian sistem guna mengetahui hasil dari simulasi.

Berikut alat dan bahan yang dibutuhkan dalam proses pengujian sistem :

1. *Microsoft Visual Studio 2008.*
2. *Personal Computer (PC) / Laptop.*

Berikut merupakan tahap-tahap pengujian sistem :

4.1.1 Pengujian Target Robot

A. Tujuan

Pengujian target robot dilakukan dengan tujuan untuk mengetahui apakah robot berjalan menuju target yang telah ditentukan oleh *user* atau tidak baik menggunakan metode *Fuzzy Logic Controller (Fuzzy)* maupun metode *Virtual Force Field (VFF)*.

B. Prosedur Pengujian

1. *Run program yang menggunakan metode Fuzzy (user / manual).*
2. *Arahkan pointer mouse pada koordinat yang diinginkan dalam koordinat arena untuk menentukan koordinat target (user / manual).*

3. Klik kanan pada *mouse* pada koordinat yang telah ditentukan (akan terlihat titik merah pada koordinat tersebut sebagai koordinat target).
4. Klik *button START* pada program (*user / manual*).
5. Robot akan membuat garis bantu kanan dan kiri menuju koordinat target sebagai acuan arah belok robot menggunakan rumus *theorema phytagoras (auto)*.
6. Ulangi langkah nomor 1 sampai langkah nomor 6, namun menggunakan program dengan metode VFF.

C. Hasil Pengujian

Dari hasil pengujian target robot, diperoleh hasil dari pembentukan garis bantu dari ujung sensor kanan dan kiri menuju target. Garis bantu tersebut digunakan untuk menentukan arah belok robot menuju target. Hasil dari pergerakan robot menuju target dengan letak koordinat target yang berbeda-beda menggunakan metode *Fuzzy* dan metode VFF dapat dilihat pada Tabel 4.1 dan Tabel 4.2.

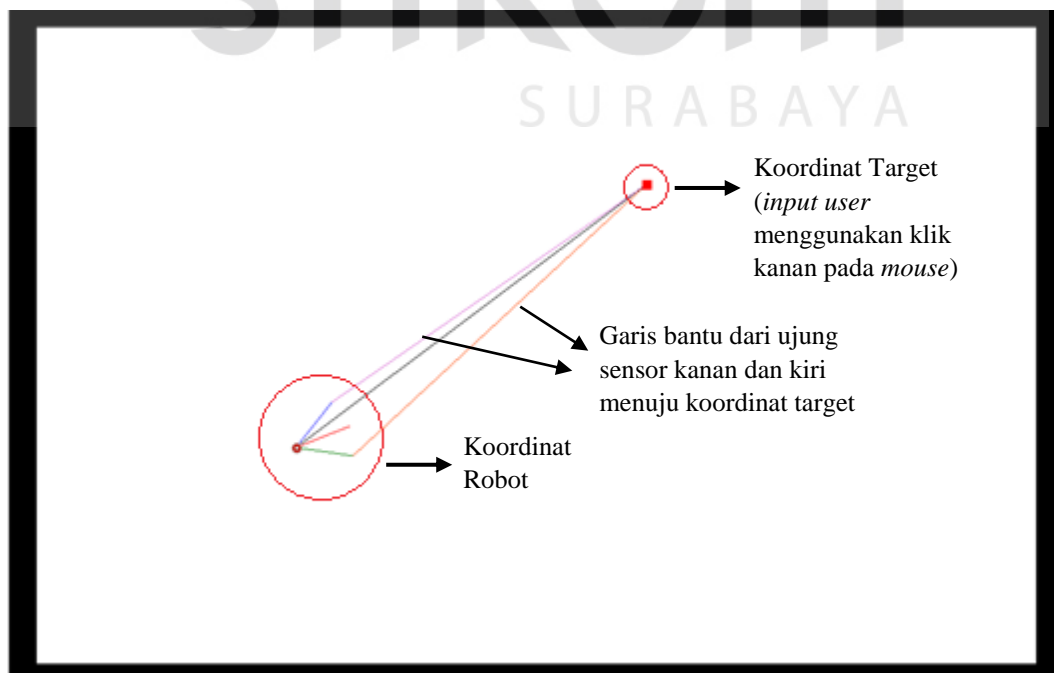
Tabel 4.1. Pengujian Target Robot Menggunakan Metode *Fuzzy*

Pengujian	Jarak Garis Bantu (Pixel)		Error Garis Bantu	Sudut Tujuan Target Terhadap Robot (Derajat)	Sudut Pergerakan (Derajat)	Error Arah (Derajat)
	Kanan	Kiri				
1	455	412	43	1	-1	2
2	334	320	14	1	-1	2
3	297	258	39	1	-1	2
4	288	235	53	1	-1	2
5	333	274	59	1	-1	2
6	235	235	0	1	0	1
7	254	279	-25	1	1	0
8	131	185	-54	1	1	0

Pengujian	Jarak Garis Bantu (Pixel)		Error Garis Bantu	Sudut Tujuan Target Terhadap Robot (Derajat)	Sudut Pergerakan (Derajat)	Error Arah (Derajat)
	Kanan	Kiri				
9	220	220	0	1	0	1
10	131	180	-49	1	1	0

Tabel 4.2. Pengujian Target Robot Menggunakan Metode VFF

Pengujian	Jarak Garis Bantu (Pixel)		Error Garis Bantu	Sudut Tujuan Target Terhadap Robot (Derajat)	Sudut Pergerakan (Derajat)	Error Arah (Derajat)
	Kanan	Kiri				
1	143	167	-24	1	1	0
2	165	115	50	0	0	0
3	300	265	35	1	0	1
4	159	159	0	1	0	1
5	525	561	-36	1	0	1
6	173	121	52	0	0	0
7	190	250	-60	0	1	-1
8	566	523	43	1	0	1
9	337	366	-29	1	0	1
10	130	89	41	0	-1	-1



Gambar 4.1. Tampilan Simulai Robot Menuju Target

D. Pembahasan

Dari hasil pengujian pada Tabel 4.1 dan Tabel 4.2, peneliti membuat garis bantu yang digambarkan dari ujung sensor kanan dan kiri menuju target menggunakan *theorema phytagoras*. Garis bantu tersebut digunakan untuk membantu robot mencapai target. Selanjutnya, dihitung selisih jarak dari ujung sensor kanan dan kiri menuju target. Nilai selisih tersebut yang akan diolah oleh *Fuzzy* dan VFF untuk mendapatkan arah belok robot menuju target.

4.1.2 Pengujian Derajat Sensor

A. Tujuan

Pengujian derajat sensor dilakukan untuk mengetahui sudut dari sensor kanan dan sensor kiri terhadap sensor tengah dalam pendeteksian *obstacle*. Sehingga robot dapat menentukan sudut belok yang tepat untuk menghindari *obstacle*.

B. Prosedur Pengujian

1. *Run* program yang menggunakan metode *Fuzzy* (*user / manual*).
2. Arahkan *pointer mouse* pada koordinat yang diinginkan dalam koordinat arena untuk menentukan koordinat target (*user / manual*).
3. Klik kanan pada *mouse* pada koordinat yang telah ditentukan (akan terlihat titik merah pada koordinat tersebut sebagai koordinat target).
4. Klik *button START* pada program (*user / manual*).
5. Arahkan *pointer mouse* pada koordinat tertentu dalam arena untuk meletakkan koordinat awal *obstacle* bergerak (*user / manual*).

6. Klik kanan pada *mouse* (akan terlihat lingkaran hitam yang disimulasikan sebagai *obstacle*).
7. Amati pergerakan robot serta sudut sensor kanan dan kiri robot terhadap sensor tengah, apakah sudah sesuai dengan yang diharapkan atau belum.
8. Peletakan *obstacle* dilakukan diulang beberapa kali untuk mengetahui hasil yang sesuai dari sudut sensor kanan dan kiri terhadap sensor tengah robot.
9. Ulangi langkah nomor 1 sampai langkah nomor 8, namun menggunakan program dengan metode VFF.

C. Hasil Pengujian

Dengan ukuran *obstacle* bergerak sebesar 60x60 *pixel* dan jarak baca maksimal sensor sebesar 60 *pixel*, diperoleh hasil pengujian seperti pada Tabel 4.3 dan Tabel 4.4 berikut :

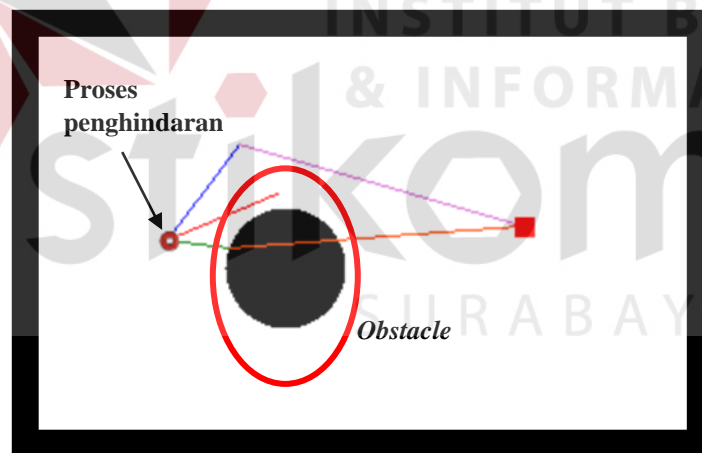
Tabel 4.3. Pengujian Sudut Sensor Kanan dan Sensor Kiri Menggunakan Metode *Fuzzy*

Pengujian	Jarak Baca Sensor Tengah	Letak Sudut Sensor (Derajat)		Robot Menyentuh <i>Obstacle</i> Bergerak	Respon Ketika <i>Obstacle</i> Berada Tepat Didepan Robot (s)
		Kanan	Kiri		
1	60	10 ⁰	-10 ⁰	Menyentuh	1,1
2	60	20 ⁰	-20 ⁰	Menyentuh	0,9
3	60	30 ⁰	-30 ⁰	Tidak	1
4	60	40 ⁰	-40 ⁰	Tidak	2,8
5	60	50 ⁰	-50 ⁰	Tidak	4,1
6	60	60 ⁰	-60 ⁰	Tidak	4,8
7	60	70 ⁰	-70 ⁰	Tidak	4,9
8	60	80 ⁰	-80 ⁰	Tidak	5,1
9	60	90 ⁰	-90 ⁰	Tidak	5,2

Tabel 4.4. Pengujian Sudut Sensor Kanan dan Sensor Kiri Menggunakan Metode VFF

Pengujian	Jarak Baca Sensor Tengah	Sudut Sensor (Derajat)		Robot Menyentuh <i>Obstacle</i> Bergerak	Respon Ketika <i>Obstacle</i> Berada Tepat Didepan Robot (s)
		Kanan	Kiri		
1	60	10^0	-10^0	Menyentuh	1
2	60	20^0	-20^0	Menyentuh	1,5
3	60	30^0	-30^0	Tidak	1,1
4	60	40^0	-40^0	Tidak	3,1
5	60	50^0	-50^0	Tidak	4,8
6	60	60^0	-60^0	Tidak	5
7	60	70^0	-70^0	Tidak	5,5
8	60	80^0	-80^0	Tidak	5,8
9	60	90^0	-90^0	Tidak	6,2

Berdasarkan hasil pengujian yang dicatat pada Tabel 4.3 dan Tabel 4.4, Gambar 4.2 adalah tampilan robot ketika merespon adanya *obstacle*.



Gambar 4.2. Respon Robot Terhadap *Obstacle*

D. Pembahasan

Dari hasil pengujian sudut sensor pada Tabel 4.3 dan Tabel 4.4, diperoleh kesimpulan bahwa peletakan sensor kiri pada -30^0 , sensor tengah pada 0^0 , dan sensor kanan pada 30^0 sudah sesuai. Karena dengan peletakan sudut sensor pada

posisi tersebut, robot dapat mendeteksi *obstacle* dan dapat meminimalisasi kegagalan robot menghindari *obstacle*.

4.1.3 Pengujian Pergerakan Kemudi

A. Tujuan

Pengujian pergerakan kemudi dilakukan dengan tujuan untuk mengetahui pergerakan robot ketika menghindari *obstacle*.

B. Prosedur Penelitian

1. *Run* program yang menggunakan metode *Fuzzy (user / manual)*.
2. Arahkan *pointer mouse* pada koordinat yang diinginkan dalam koordinat arena untuk menentukan koordinat target (*user / manual*).
3. Klik kanan *mouse* pada koordinat yang telah ditentukan (akan terlihat titik merah pada koordinat tersebut sebagai koordinat target).
4. Klik *button START* pada program (*user / manual*).
5. Robot membentuk sensor *ultrasonic* secara otomatis (*auto*).
6. Arahkan *pointer mouse* pada koordinat yang akan dilalui robot dalam arena untuk meletakkan *obstacle* bergerak (*user / manual*).
7. Klik kanan pada *mouse* (akan terlihat lingkaran hitam yang disimulasikan sebagai *obstacle*).
8. Ulangi langkah nomor 1 sampai langkah nomor 7, namun menggunakan program dengan metode VFF.

C. Hasil Pengujian

Dengan peletakan sudut sensor kanan dan kiri sebesar 30^0 terhadap sensor tengah dan sesuai dengan prosedur pengujian yang dilakukan, maka diperoleh hasil pengujian seperti pada Tabel 4.5 dan Tabel 4.6 berikut :

Tabel 4.5. Pengujian Pergerakan Kemudi Menggunakan Metode *Fuzzy*

Sensor Tengah (<i>pixel</i>)	Sensor Kanan (<i>pixel</i>)	Sensor Kiri (<i>pixel</i>)	Halangan (<i>Obstacle</i>)	Kemudi
60	60	60	Tidak ada halangan	Lurus
60	40	60	Halangan di kanan	Belok Kiri
40	40	60	Halangan depan kanan	Belok Kiri
60	60	40	Halangan di kiri	Belok kanan
40	60	40	Halangan depan kiri	Belok kanan
10	20	30	Halangan di depan kiri kanan	Belok Kiri
10	30	20	Halangan di depan kiri kanan	Belok Kanan

Tabel 4.6. Pengujian Pergerakan Kemudi Menggunakan Metode VFF

Sensor Tengah (<i>pixel</i>)	Sensor Kanan (<i>pixel</i>)	Sensor Kiri (<i>pixel</i>)	Halangan (<i>Obstacle</i>)	Kemudi
60	60	60	Tidak ada halangan	Lurus
60	40	60	Halangan di kanan	Belok Kiri
40	40	60	Halangan depan kanan	Belok Kiri
60	60	40	Halangan di kiri	Belok kanan
40	60	40	Halangan depan kiri	Belok kanan
10	20	30	Halangan di depan kiri kanan	Belok Kiri
10	30	20	Halangan di depan kiri kanan	Belok Kanan

D. Pembahasan

Dari hasil yang diperoleh dari pengujian pergerakan kemudi pada Tabel 4.5 dan Tabel 4.6, robot dengan menggunakan metode *Fuzzy* maupun menggunakan metode VFF dapat berbelok ke kanan ketika sensor kiri mengenai

obstacle. Sebaliknya, robot dapat bergerak ke kiri ketika sensor kanan mengenai *obstacle*. Apabila sensor depan mendeteksi adanya *obstacle*, robot akan membandingkan jarak sensor kanan dan kiri. Apabila lebih besar jarak sensor kanan, maka robot akan belok ke kanan. Sebaliknya, apabila lebih besar jarak sensor kiri, maka robot akan berbelok ke kiri. Sehingga, dari pengujian pergerakan kemudi, dapat disimpulkan bahwa kemudi dapat berjalan sesuai dengan yang diharapkan.

4.1.4 Pengujian Output Metode Fuzzy Logic Controller

A. Tujuan

Pengujian metode *Fuzzy Logic Controller (Fuzzy)* ini dilakukan untuk mengetahui apakah robot bisa berjalan sesuai harapan apabila arah belok robot ditentukan menggunakan metode *Fuzzy*.

B. Prosedur Penelitian

1. *Run* program (*user/manual*).
2. Arahkan *pointer mouse* pada koordinat yang diinginkan dalam koordinat arena (*user/manual*).
3. Klik kanan *mouse* pada koordinat yang telah ditentukan (akan terlihat titik merah pada koordinat tersebut sebagai koordinat target).
4. Klik *button START* pada simulasi (*user/manual*).
5. Arahkan *pointer mouse* pada koordinat yang akan dilalui robot dalam arena untuk meletakkan *obstacle* bergerak (*user / manual*).

6. Klik kanan pada *mouse* (akan terlihat lingkaran hitam yang disimulasikan sebagai *obstacle*).
7. Sensor *ultrasonic* membaca *obstacle* yang ada di depannya kemudian memberikan input data pada *Fuzzy (auto)*.
8. Amati pergerakan robot dan parameter data *Fuzzy*.
9. Lakukan pengujian beberapa kali untuk mengetahui hasil dari parameter *Fuzzy* yang sesuai.

C. Hasil Pengujian

Sesuai dengan prosedur pengujian, diperoleh hasil *output Fuzzy* untuk menentukan arah belok robot seperti pada Tabel 4.7 dan Tabel 4.8. *Input* yang digunakan untuk menentukan arah belok robot adalah delta sensor (selisih nilai jarak sensor kanan dan kiri) dan delta target (selisih nilai garis bantu dari ujung sensor kanan dan kiri menuju target).

Tabel 4.7. Pengujian *Output* Arah Belok Metode *Fuzzy Logic Controller* Berdasarkan Sensor Kanan dan Arah Target

No	Waktu	Delta Sensor	Delta Target	<i>Output</i> Pergerakan (derajat)
1	0,01	0,000	-41,207	-3,000
2	0,02	0,000	-40,338	-6,000
3	0,03	0,000	-39,448	-9,000
4	0,04	0,000	-38,537	-12,000
5	0,05	0,000	-37,603	-15,000
6	0,06	0,000	-36,647	-18,000
7	0,07	0,000	-35,669	-21,000
8	0,08	0,000	-34,669	-24,000
9	0,09	0,000	-33,646	-27,000
10	0,1	0,000	-32,600	-30,000
11	0,11	0,000	-31,531	-33,000
12	0,12	0,000	-30,440	-36,000
13	0,13	0,000	-29,324	-39,000
14	0,14	0,000	-28,186	-42,000

No	Waktu	Delta Sensor	Delta Target	Output Pergerakan (derajat)
15	0,15	0,000	-27,023	-45,000
16	0,16	0,000	-25,837	-48,000
17	0,17	0,000	-24,628	-51,000
18	0,18	0,000	-23,394	-54,000
19	0,19	0,000	-22,135	-57,000
20	0,2	0,000	-20,853	-60,000
21	0,21	0,000	-19,546	-63,000
22	0,22	-25,800	-34,638	-69,000
23	0,23	-26,200	-32,936	-75,000
24	0,24	-21,400	-27,405	-81,000
25	0,25	0,000	-9,265	-82,853
26	0,26	0,000	-8,374	-84,528
27	0,27	0,000	-7,565	-86,041
28	0,28	0,000	-6,834	-87,408
29	0,29	0,000	-6,172	-88,642
30	0,3	0,000	-5,573	-89,756
31	0,31	-23,800	-23,644	-95,756
32	0,32	-24,400	-21,657	-101,756
33	0,33	-22,800	-17,741	-107,756
34	0,34	-19,600	-12,260	-113,302
35	0,35	0,000	7,833	-111,735
36	0,36	0,000	7,074	-110,320
37	0,37	0,000	6,387	-109,043
38	0,38	-32,600	-22,879	-115,043
39	0,39	-33,600	-21,224	-121,043
40	0,4	-35,600	-20,454	-127,043
41	0,41	-36,400	-18,530	-133,043
42	0,42	-31,200	-10,752	-138,500
43	0,43	0,000	22,315	-135,500
44	0,44	0,000	21,049	-132,500
45	0,45	0,000	19,755	-129,500
46	0,46	0,000	18,432	-126,500
47	0,47	0,000	17,081	-123,500
48	0,48	0,000	15,699	-120,500
49	0,49	0,000	14,288	-117,643
50	0,5	-32,000	-16,634	-123,643

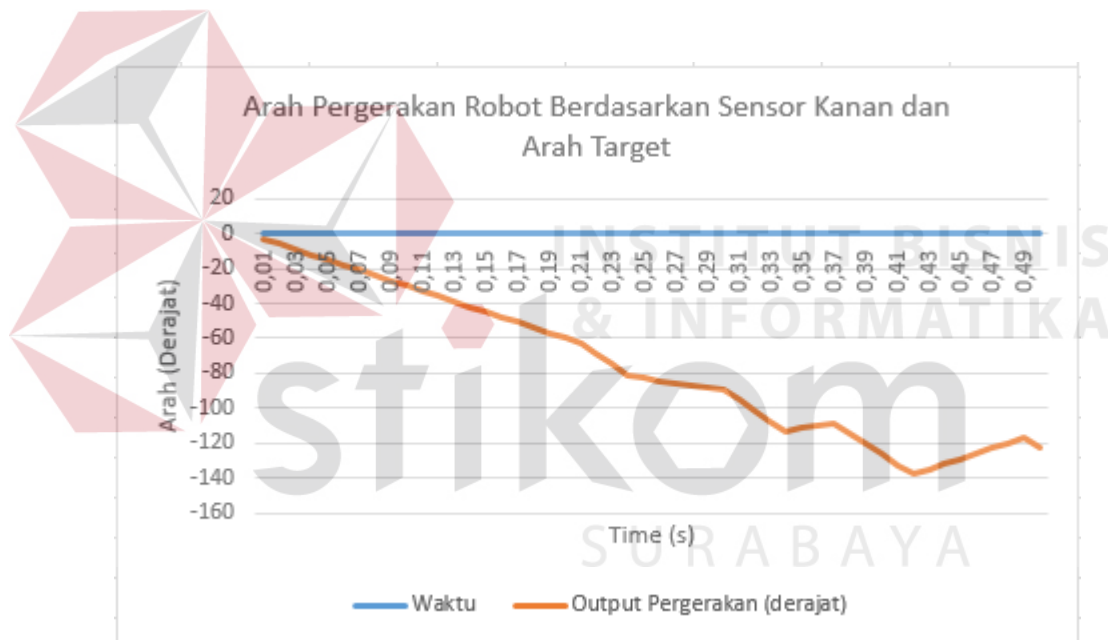
Tabel 4.8. Pengujian *Output* Arah Belok Metode *Fuzzy Logic Controller* Berdasarkan Sensor Kiri dan Arah Target

No	Waktu	Selisih Sensor	Selisih Target	Output Pergerakan (derajat)
1	0,01	0,000	24,685	3
2	0,02	0,000	23,430	6

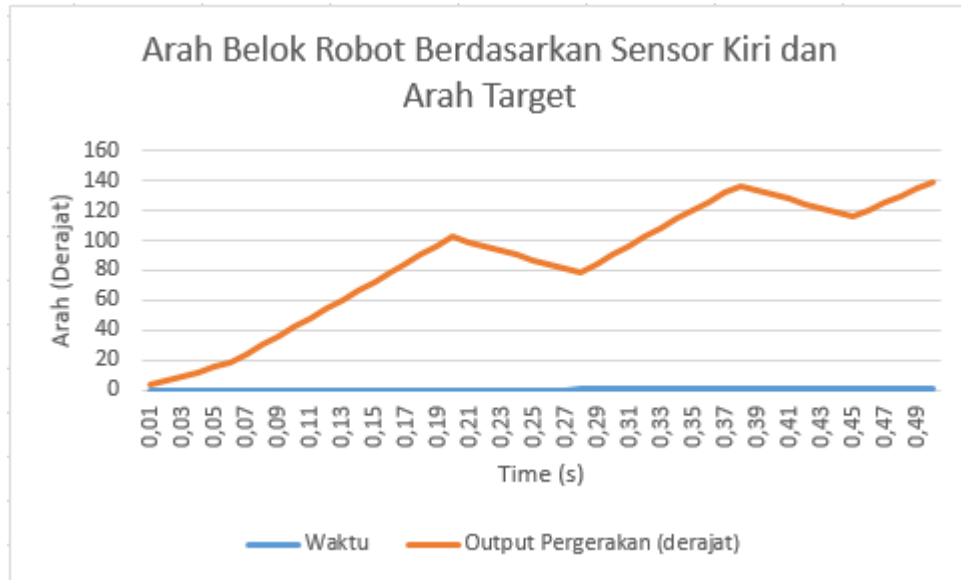
No	Waktu	Selisih Sensor	Selisih Target	Output Pergerakan (derajat)
3	0,03	0,000	22,155	9
4	0,04	0,000	20,859	12
5	0,05	0,000	19,543	15
6	0,06	0,000	18,207	18
7	0,07	45,200	47,383	24
8	0,08	44,400	45,439	30
9	0,09	44,200	43,785	36
10	0,1	43,800	41,854	42
11	0,11	43,000	39,469	48
12	0,12	43,000	37,620	54
13	0,13	42,200	34,984	60
14	0,14	44,800	35,237	66
15	0,15	46,400	34,589	72
16	0,16	47,800	33,719	78
17	0,17	49,000	32,609	84
18	0,18	49,000	30,283	90
19	0,19	48,000	26,888	96
20	0,2	40,600	17,079	102
21	0,21	0,000	-25,862	99
22	0,22	0,000	-24,632	96
23	0,23	0,000	-23,381	93
24	0,24	0,000	-22,108	90
25	0,25	0,000	-20,814	87
26	0,26	0,000	-19,499	84
27	0,27	0,000	-18,163	81
28	0,28	0,000	-16,807	78
29	0,29	47,400	29,779	84
30	0,3	46,400	26,503	90
31	0,31	46,200	23,928	96
32	0,32	46,000	21,296	102
33	0,33	44,800	17,621	108
34	0,34	45,400	15,509	114
35	0,35	47,800	15,145	120
36	0,36	49,000	13,529	125,754
37	0,37	47,800	9,579	131,227
38	0,38	42,000	1,187	136,191
39	0,39	0,000	-42,314	133,191
40	0,4	0,000	-41,457	130,191
41	0,41	0,000	-40,578	127,191
42	0,42	0,000	-39,678	124,191
43	0,43	0,000	-38,755	121,191
44	0,44	0,000	-37,810	118,191
45	0,45	0,000	-36,843	115,191
46	0,46	38,200	2,265	120,074

No	Waktu	Selisih Sensor	Selisih Target	Output Pergerakan (derajat)
47	0,47	39,000	0,857	124,818
48	0,48	40,000	-0,464	129,475
49	0,49	40,800	-1,966	134,149
50	0,5	42,600	-2,520	138,904

Dari pengujian yang telah dicatat pada Tabel 4.7 dan Tabel 4.8, dapat digambarkan grafik respon sensor kanan dan sensor kiri ketika mendeteksi adanya *obstacle* menggunakan metode *Fuzzy* pada Gambar 4.3 dan Gambar 4.4.



Gambar 4.3. Grafik Arah Robot Ketika Sensor Kanan Mendeteksi *Obstacle* Menggunakan *Fuzzy*



Gambar 4.4. Grafik Arah Robot Ketika Sensor Kiri Mendeteksi *Obstacle* Menggunakan *Fuzzy*

D. Pembahasan

Dari pengujian yang telah dilakukan, robot dapat berbelok ke kanan apabila sensor kiri mendeteksi adanya *obstacle*. Sebaliknya, robot dapat berbelok ke kiri apabila sensor kanan mendeteksi adanya *obstacle*. Sehingga dapat disimpulkan bahwa robot dapat menghindari *obstacle* sesuai dengan yang diharapkan.

4.1.5 Pengujian *Output Metode Virtual Force Field (VFF)*

A. Tujuan

Pengujian metode *Virtual Force Field (VFF)* ini dilakukan untuk mengetahui apakah robot bisa berjalan sesuai harapan apabila arah belok robot ditentukan menggunakan metode VFF.

B. Prosedur Penelitian

1. *Run* program (*user/manual*).
2. Arahkan *pointer mouse* pada koordinat yang diinginkan dalam koordinat arena (*user/manual*).
3. Klik kanan *mouse* pada koordinat yang telah ditentukan (akan terlihat titik merah pada koordinat tersebut sebagai koordinat target).
4. Klik *button START* pada simulasi (*user/manual*).
5. Arahkan *pointer mouse* pada koordinat yang akan dilalui robot dalam arena untuk meletakkan *obstacle* bergerak (*user / manual*).
6. Klik kanan pada *mouse* (akan terlihat lingkaran hitam yang disimulasikan sebagai *obstacle*).
7. Sensor *ultrasonic* membaca *obstacle* yang ada di depannya kemudian memberikan input data pada VFF (*auto*).
8. Amati pergerakan robot dan parameter data VFF.
9. Lakukan pengujian beberapa kali untuk mengetahui hasil dari parameter VFF yang sesuai.

C. Hasil Pengujian

Sesuai dengan prosedur pengujian, diperoleh hasil *output* VFF untuk menentukan arah belok robot seperti pada Tabel 4.9 dan Tabel 4.10. *Input* yang digunakan untuk menentukan arah belok robot adalah selisih sensor (selisih nilai jarak sensor kanan dan kiri) dan selisih target (selisih nilai garis bantu dari ujung sensor kanan dan kiri menuju target).

Tabel 4.9. Pengujian *Output* Arah Belok Metode *Virtual Force Field* Berdasarkan Sensor Kanan dan Arah Target

No	Waktu	Selisih Sensor	Selisih Target	<i>Output</i> Pergerakan (derajat)
1	0,01	0,000	41,716	1,547
2	0,02	0,000	40,816	3,093
3	0,03	0,000	39,893	4,639
4	0,04	0,000	38,946	6,184
5	0,05	0,000	37,977	7,728
6	0,06	0,000	36,985	9,272
7	0,07	0,000	35,969	10,815
8	0,08	0,000	34,930	12,357
9	0,09	0,000	33,868	13,899
10	0,1	0,000	32,783	15,439
11	0,11	0,000	31,674	16,978
12	0,12	0,000	30,543	18,516
13	0,13	-35,200	-5,731	15,576
14	0,14	-35,600	-3,569	12,736
15	0,15	-36,000	-1,459	10,223
16	0,16	-37,600	-0,615	8,127
17	0,17	-39,200	-0,110	6,472
18	0,18	-40,800	0,037	4,962
19	0,19	-40,800	1,656	4,444
20	0,2	-42,800	0,461	3,328
21	0,21	-42,800	1,761	2,835
22	0,22	-41,000	4,278	2,630
23	0,23	-39,400	6,348	2,499
24	0,24	-34,000	12,032	2,445
25	0,25	0,000	45,177	3,994
26	0,26	0,000	44,403	5,542
27	0,27	0,000	43,608	7,090
28	0,28	0,000	42,791	8,637
29	0,29	0,000	41,951	10,184
30	0,3	0,000	41,090	11,731
31	0,31	0,000	40,207	13,277
32	0,32	0,000	39,301	14,822
33	0,33	0,000	38,373	16,367
34	0,34	0,000	37,422	17,911
35	0,35	0,000	36,448	19,454
36	0,36	0,000	35,450	20,997
37	0,37	0,000	34,429	22,539
38	0,38	0,000	33,385	24,080
39	0,39	0,000	32,317	25,619
40	0,4	0,000	31,225	27,158
41	0,41	0,000	30,109	28,696
42	0,42	0,000	28,969	30,232
43	0,43	0,000	27,804	31,767

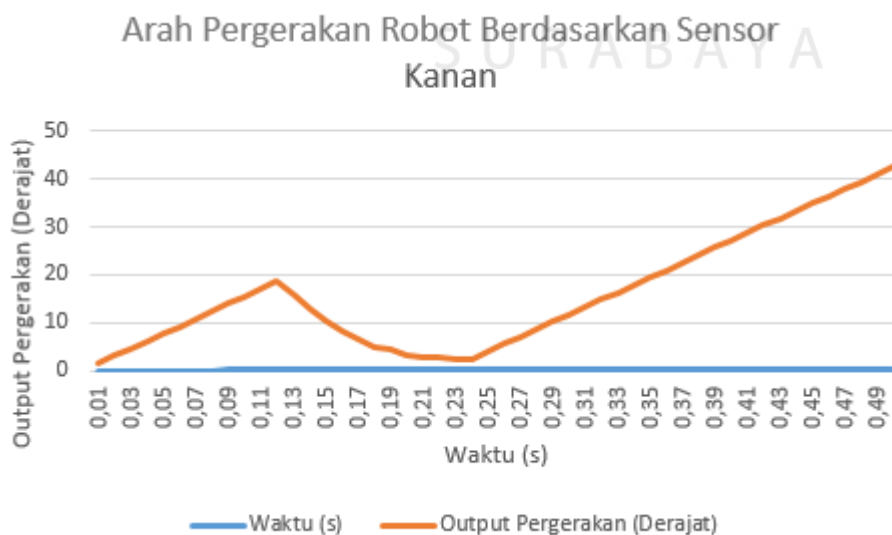
No	Waktu	Selisih Sensor	Selisih Target	Output Pergerakan (derajat)
44	0,44	0,000	26,615	33,300
45	0,45	0,000	25,401	34,832
46	0,46	0,000	24,163	36,361
47	0,47	0,000	22,900	37,888
48	0,48	0,000	21,613	39,413
49	0,49	0,000	20,301	40,934
50	0,5	0,000	18,964	42,452

Tabel 4.10. Pengujian *Output* Arah Belok Metode *Virtual Force Field* Berdasarkan Sensor Kiri dan Arah Target

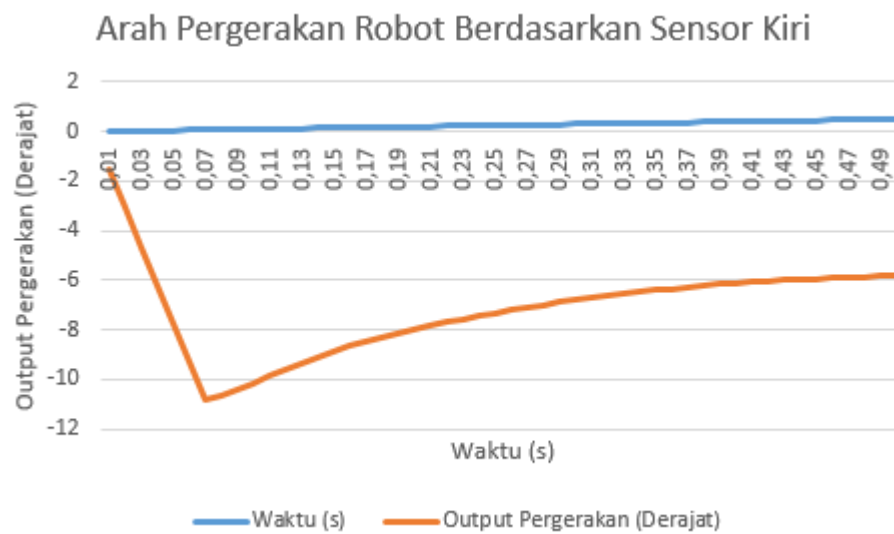
No	Waktu	Selisih Sensor	Selisih Target	Output Pergerakan (derajat)
1	0,01	0,000	-46,604	-1,549
2	0,02	0,000	-45,839	-3,098
3	0,03	0,000	-45,051	-4,647
4	0,04	0,000	-44,240	-6,195
5	0,05	0,000	-43,407	-7,743
6	0,06	0,000	-42,551	-9,290
7	0,07	0,000	-41,672	-10,837
8	0,08	36,800	-4,510	-10,646
9	0,09	37,400	-4,175	-10,438
10	0,1	38,800	-3,061	-10,148
11	0,11	38,800	-3,339	-9,882
12	0,12	38,800	-3,580	-9,636
13	0,13	39,000	-3,607	-9,391
14	0,14	39,200	-3,630	-9,148
15	0,15	39,400	-3,642	-8,905
16	0,16	39,400	-3,849	-8,676
17	0,17	39,400	-4,045	-8,459
18	0,18	38,200	-5,404	-8,302
19	0,19	38,200	-5,538	-8,150
20	0,2	38,400	-5,472	-7,995
21	0,21	38,400	-5,603	-7,845
22	0,22	38,400	-5,731	-7,698
23	0,23	38,000	-6,246	-7,566
24	0,24	37,800	-6,554	-7,441
25	0,25	37,800	-6,659	-7,318
26	0,26	37,200	-7,348	-7,210
27	0,27	37,200	-7,439	-7,103
28	0,28	37,400	-7,335	-6,994
29	0,29	37,400	-7,426	-6,887
30	0,3	36,200	-8,684	-6,800
31	0,31	36,200	-8,757	-6,714
32	0,32	36,200	-8,829	-6,629

No	Waktu	Selisih Sensor	Selisih Target	Output Pergerakan (derajat)
33	0,33	36,200	-8,900	-6,544
34	0,34	35,000	-10,136	-6,475
35	0,35	35,000	-10,195	-6,405
36	0,36	35,000	-10,252	-6,337
37	0,37	35,000	-10,309	-6,269
38	0,38	33,800	-11,530	-6,212
39	0,39	32,600	-12,741	-6,164
40	0,4	32,600	-12,780	-6,117
41	0,41	32,600	-12,819	-6,069
42	0,42	31,400	-14,020	-6,030
43	0,43	30,200	-15,215	-5,997
44	0,44	30,200	-15,241	-5,965
45	0,45	30,200	-15,267	-5,933
46	0,46	29,000	-16,455	-5,907
47	0,47	27,800	-17,637	-5,886
48	0,48	27,800	-17,654	-5,865
49	0,49	27,800	-17,670	-5,845
50	0,5	26,600	-18,848	-5,829

Dari pengujian yang telah dicatat pada Tabel 4.9 dan Tabel 4.10, dapat digambarkan grafik respon sensor kanan dan sensor kiri ketika mendeteksi adanya *obstacle* menggunakan metode *Fuzzy* pada Gambar 4.5 dan Gambar 4.6.



Gambar 4.5. Grafik Arah Robot Ketika Sensor Kanan Mendeteksi *Obstacle* Menggunakan VFF



Gambar 4.6. Grafik Arah Robot Ketika Sensor Kiri Mendeteksi *Obstacle* Menggunakan VFF

D. Pembahasan

Dari pengujian yang telah dilakukan, robot dapat berbelok ke kanan apabila sensor kiri mendeteksi adanya *obstacle*. Sebaliknya, robot dapat berbelok ke kiri apabila sensor kanan mendeteksi adanya *obstacle*. Sehingga dapat disimpulkan bahwa robot dapat menghindari *obstacle* sesuai dengan yang diharapkan.

4.2 Pengujian Keseluruhan Sistem

Pengujian keseluruhan sistem dilakukan untuk mengetahui apakah keseluruhan sistem sudah berjalan sesuai dengan yang diharapkan atau belum. Pengujian ini dilakukan dengan cara pengambilan data ketika robot menuju target mulai dari tanpa *obstacle*, satu *obstacle*, dan *multiple obstacle* menggunakan metode *Fuzzy* dan VFF. Berikut merupakan hasil dari pengujian sistem secara keseluruhan.

4.2.1. Gerak Robot Terhadap Koordinat Target Menggunakan Metode *Fuzzy Logic Controller (Fuzzy)*

A. Hasil Pengujian

Pengujian ini mengambil beberapa data pergerakan robot menuju target menggunakan metode *Fuzzy* ketika tidak ada *obstacle* sama sekali, ketika terdapat satu *obstacle*, serta ketika terdapat beberapa *obstacle*. Data yang dicatat adalah *error* koordinat robot terhadap koordinat target yang telah ditentukan, baik dalam bentuk *pixel* maupun persentase. Berikut adalah hasil pengujian gerak robot terhadap koordinat target menggunakan metode *Fuzzy*.

Tabel 4.11. Pengujian Gerak Robot Menuju Target Tanpa *Obstacle* Menggunakan Metode *Fuzzy*

Time (s)	Koordinat Target		Koordinat Robot		Kesalahan (<i>pixel</i>)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
0,2	443	505	224	517	219	12	49,436	2,376
0,4	443	505	244	516	199	11	44,921	2,178
0,6	443	505	264	515	179	10	40,406	1,980
0,8	443	505	284	514	159	9	35,892	1,782
1	443	505	304	513	139	8	31,377	1,584
1,2	443	505	324	512	119	7	26,862	1,386
1,4	443	505	344	511	99	6	22,348	1,188
1,6	443	505	364	510	79	5	17,833	0,990
1,8	443	505	384	508	59	3	13,318	0,594
1,9	443	505	394	508	49	3	11,061	0,594
2	443	505	404	507	39	2	8,804	0,396
2,10	443	505	414	507	29	2	6,546	0,396
2,19	443	505	422	506	21	1	4,740	0,198
2,25	443	505	428	506	15	1	3,386	0,198
2,29	443	505	433	506	10	1	2,257	0,198
2,32	443	505	436	505	7	0	1,580	0,000
2,34	443	505	438	505	5	0	1,129	0,000
2,36	443	505	439	505	4	0	0,903	0,000
2,37	443	505	440	505	3	0	0,677	0,000
2,38	443	505	441	505	2	0	0,451	0,000
2,38	443	505	442	505	1	0	0,226	0,000
2,38	443	505	442	505	1	0	0,226	0,000

Time (s)	Koordinat Target		Koordinat Robot		Kesalahan (pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
2,39	443	505	442	505	1	0	0,226	0,000
2,39	443	505	443	505	0	0	0,000	0,000
2,39	443	505	443	505	0	0	0,000	0,000
Rata-Rata					57,52	3,24	12,984	0,642

Tabel 4.12. Pengujian 1 Gerak Robot Menuju Target dengan Satu *Obstacle* Menggunakan Metode *Fuzzy*

Time (s)	Koordinat Target		Koordinat Robot		Kesalahan (pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
0,2	468	523	224	518	244	5	52,137	0,956
0,37	468	523	241	516	227	7	48,504	1,338
0,57	468	523	260	508	208	15	44,444	2,868
0,77	468	523	279	503	189	20	40,385	3,824
0,97	468	523	299	502	169	21	36,111	4,015
1,17	468	523	319	503	149	20	31,838	3,824
1,37	468	523	339	506	129	17	27,564	3,250
1,57	468	523	358	508	110	15	23,504	2,868
1,77	468	523	378	511	90	12	19,231	2,294
1,97	468	523	398	514	70	9	14,957	1,721
2,13	468	523	414	516	54	7	11,538	1,338
2,24	468	523	424	517	44	6	9,402	1,147
2,34	468	523	434	518	34	5	7,265	0,956
2,43	468	523	443	520	25	3	5,342	0,574
2,51	468	523	451	521	17	2	3,632	0,382
2,56	468	523	456	521	12	2	2,564	0,382
2,59	468	523	459	522	9	1	1,923	0,191
2,62	468	523	462	522	6	1	1,282	0,191
2,64	468	523	464	522	4	1	0,855	0,191
2,65	468	523	465	523	3	0	0,641	0,000
2,66	468	523	466	523	2	0	0,427	0,000
2,67	468	523	466	523	2	0	0,427	0,000
2,67	468	523	467	523	1	0	0,214	0,000
2,67	468	523	467	523	1	0	0,214	0,000
2,67	468	523	467	523	1	0	0,214	0,000
2,67	468	523	468	523	0	0	0,000	0,000
Rata-Rata					69,23	6,5	14,793	1,243

Tabel 4.13. Pengujian 2 Gerak Robot Menuju Target dengan Satu *Obstacle* Menggunakan Metode *Fuzzy*

Time (s)	Koordinat Target		Koordinat Robot		Kesalahan (pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
0,2	189	413	215	225	26	188	13,757	45,521
0,4	189	413	233	232	44	181	23,280	43,826
0,49	189	413	241	237	52	176	27,513	42,615
0,68	189	413	249	254	60	159	31,746	38,499
0,88	189	413	251	274	62	139	32,804	33,656
1,08	189	413	249	294	60	119	31,746	28,814
1,28	189	413	242	313	53	100	28,042	24,213
1,48	189	413	233	330	44	83	23,280	20,097
1,68	189	413	223	348	34	65	17,989	15,738
1,86	189	413	215	364	26	49	13,757	11,864
1,96	189	413	210	373	21	40	11,111	9,685
2,06	189	413	206	382	17	31	8,995	7,506
2,16	189	413	201	390	12	23	6,349	5,569
2,23	189	413	198	397	9	16	4,762	3,874
2,29	189	413	195	402	6	11	3,175	2,663
2,32	189	413	193	405	4	8	2,116	1,937
2,35	189	413	192	407	3	6	1,587	1,453
2,37	189	413	191	409	2	4	1,058	0,969
2,38	189	413	191	410	2	3	1,058	0,726
2,39	189	413	190	411	1	2	0,529	0,484
2,40	189	413	190	412	1	1	0,529	0,242
2,40	189	413	190	412	1	1	0,529	0,242
2,41	189	413	189	412	0	1	0,000	0,242
2,41	189	413	189	412	0	1	0,000	0,242
2,41	189	413	189	413	0	0	0,000	0,000
Rata-Rata					21,6	56,28	11,429	13,627

Tabel 4.14. Pengujian 1 Gerak Robot Menuju Target dengan *Multiple Obstacle* Menggunakan Metode *Fuzzy*

Time (s)	Koordinat Target		Koordinat Robot		Kesalahan (pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
0,2	186	477	215	225	29	252	15,591	52,830
0,4	186	477	233	232	47	245	25,269	51,363
0,56	186	477	246	241	60	236	32,258	49,476
0,76	186	477	255	259	69	218	37,097	45,702
0,96	186	477	258	278	72	199	38,710	41,719
1,15	186	477	255	297	69	180	37,097	37,736
1,35	186	477	247	315	61	162	32,796	33,962
1,55	186	477	240	334	54	143	29,032	29,979
1,75	186	477	233	353	47	124	25,269	25,996

Time (s)	Koordinat Target		Koordinat Robot		Kesalahan (pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
1,95	186	477	226	371	40	106	21,505	22,222
2,15	186	477	219	390	33	87	17,742	18,239
2,35	186	477	212	409	26	68	13,978	14,256
2,53	186	477	205	426	19	51	10,215	10,692
2,63	186	477	202	435	16	42	8,602	8,805
2,73	186	477	198	444	12	33	6,452	6,918
2,82	186	477	195	453	9	24	4,839	5,031
2,90	186	477	192	460	6	17	3,226	3,564
2,95	186	477	190	465	4	12	2,151	2,516
2,99	186	477	189	469	3	8	1,613	1,677
3,01	186	477	188	471	2	6	1,075	1,258
3,03	186	477	188	473	2	4	1,075	0,839
3,04	186	477	187	474	1	3	0,538	0,629
3,05	186	477	187	475	1	2	0,538	0,419
3,06	186	477	187	476	1	1	0,538	0,210
3,06	186	477	186	476	0	1	0,000	0,210
3,07	186	477	186	476	0	1	0,000	0,210
3,07	186	477	186	476	0	1	0,000	0,210
3,07	186	477	186	477	0	0	0,000	0,000
Rata-Rata					24,393	79,5	13,114	16,667

Tabel 4.15. Pengujian 2 Gerak Robot Menuju Target dengan *Multiple Obstacle* Menggunakan Metode *Fuzzy*

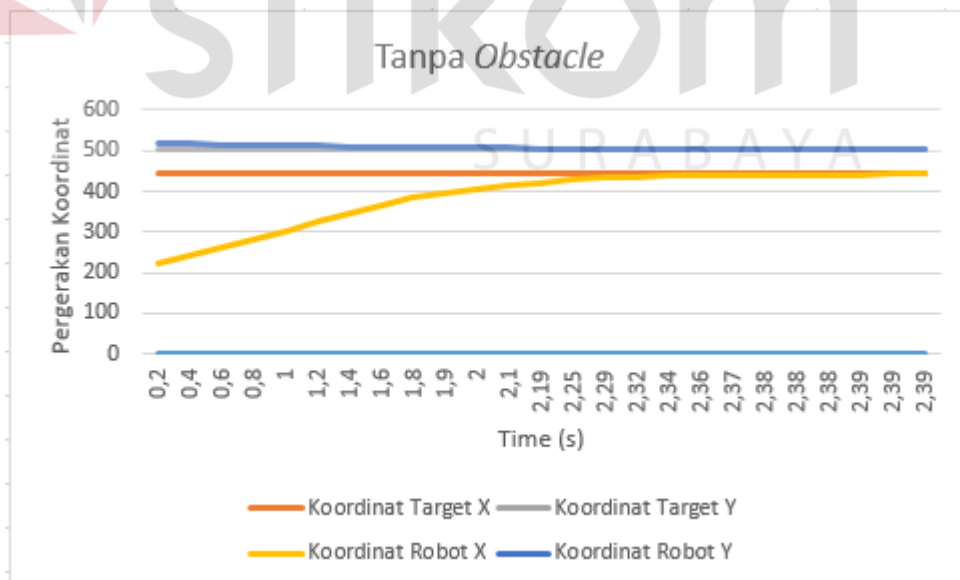
Time (s)	Koordinat Target		Koordinat Robot		Kesalahan (pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
0,2	140	198	174	459	34	261	24,286	131,818
0,31	140	198	182	453	42	255	30,000	128,788
0,51	140	198	194	437	54	239	38,571	120,707
0,71	140	198	201	418	61	220	43,571	111,111
0,91	140	198	201	398	61	200	43,571	101,010
1,11	140	198	197	379	57	181	40,714	91,414
1,31	140	198	192	360	52	162	37,143	81,818
1,51	140	198	186	340	46	142	32,857	71,717
1,71	140	198	180	321	40	123	28,571	62,121
1,91	140	198	174	302	34	104	24,286	52,525
2,11	140	198	167	283	27	85	19,286	42,929
2,31	140	198	161	264	21	66	15,000	33,333
2,47	140	198	156	249	16	51	11,429	25,758
2,57	140	198	153	240	13	42	9,286	21,212
2,67	140	198	150	230	10	32	7,143	16,162
2,76	140	198	147	221	7	23	5,000	11,616
2,83	140	198	145	214	5	16	3,571	8,081

Time (s)	Koordinat Target		Koordinat Robot		Kesalahan (pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
2,88	140	198	144	210	4	12	2,857	6,061
2,92	140	198	143	206	3	8	2,143	4,040
2,94	140	198	142	204	2	6	1,429	3,030
2,96	140	198	141	202	1	4	0,714	2,020
2,97	140	198	141	201	1	3	0,714	1,515
2,98	140	198	141	200	1	2	0,714	1,010
2,99	140	198	140	199	0	1	0,000	0,505
2,99	140	198	140	199	0	1	0,000	0,505
3,00	140	198	140	199	0	1	0,000	0,505
3,00	140	198	140	199	0	1	0,000	0,505
3,00	140	198	140	198	0	0	0,000	0,000
Rata-Rata					21,143	80,036	15,102	40,422

B. Pembahasan

Dari Tabel 4.11 sampai 4.15 didapatkan hasil pergerakan robot menuju target menggunakan metode *Fuzzy* seperti berikut.

1. Tanpa *Obstacle*



Gambar 4.7. Grafik Pengujian Gerak Robot Menuju Target Tanpa *Obstacle* Menggunakan Metode *Fuzzy*

Grafik dari pencatatan pergerakan robot menuju target tanpa melewati *obstacle* dapat dilihat pada Gambar 4.7. Berdasarkan data dari Tabel 4.11, dapat diketahui rata-rata dari kesalahan koordinat robot terhadap target sebagai berikut :

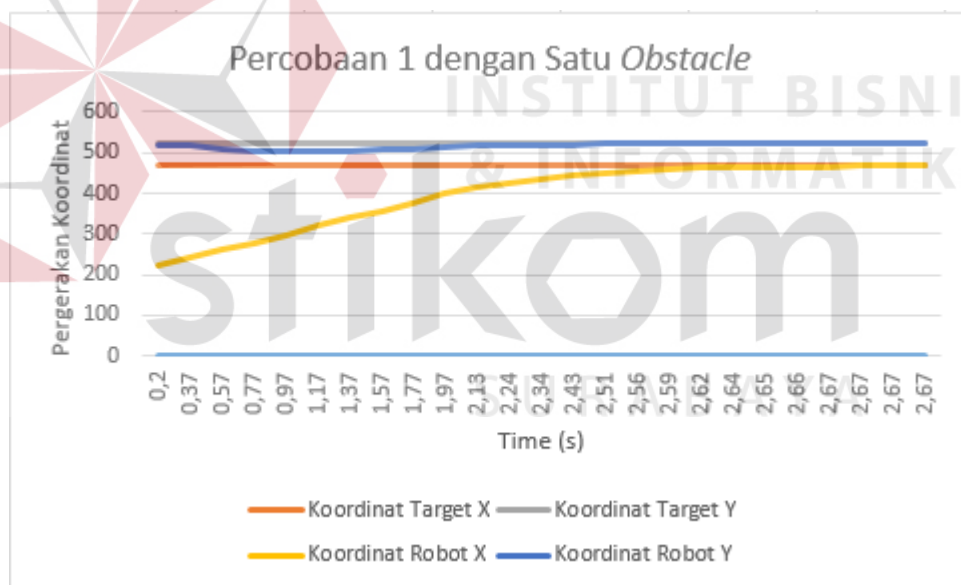
$$\text{Rata-rata kesalahan sumbu X} = \frac{1438}{25} = 57,52 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{81}{25} = 3,24 \text{ pixel}$$

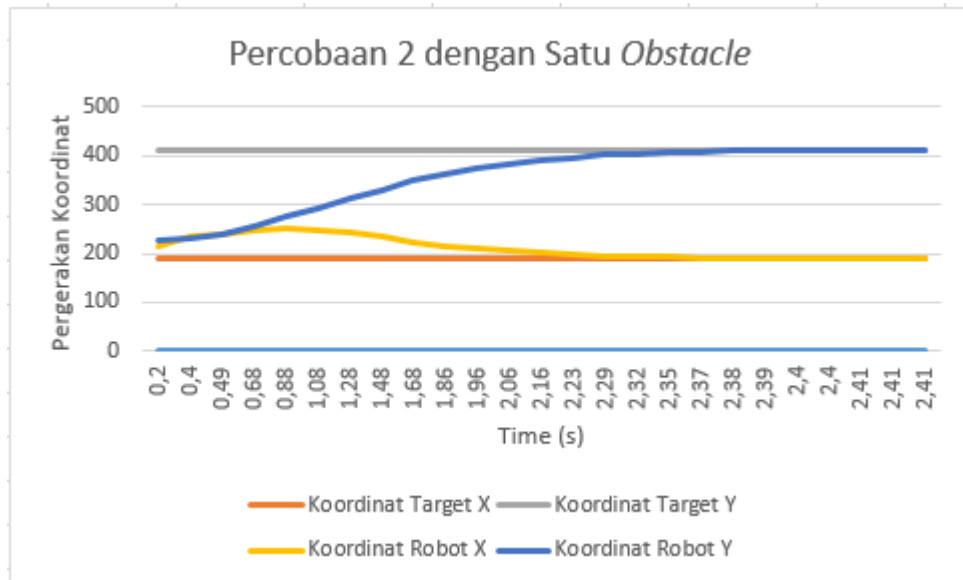
$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{1438}{25} = 12,984\%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{81}{25} = 0,642 \%$$

2. Melewati Satu *Obstacle*



Gambar 4.8. Grafik Percobaan 1 Gerak Robot Menuju Target dengan Satu *Obstacle* Menggunakan Metode *Fuzzy*



Gambar 4.9. Grafik Percobaan 2 Gerak Robot Menuju Target dengan Satu *Obstacle* Menggunakan Metode *Fuzzy*

Grafik dari pencatatan dua kali percobaan pergerakan robot menuju target dengan melewati satu *obstacle* dapat dilihat pada Gambar 4.8 dan Gambar 4.9. Berdasarkan data dari Tabel 4.12 dan Tabel 4.13 , dapat diketahui rata-rata dari kesalahan koordinat robot terhadap target sebagai berikut :

Percobaan 1:

$$\text{Rata-rata kesalahan sumbu X} = \frac{1800}{26} = 69,23 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{169}{26} = 6,5 \text{ pixel}$$

$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{1800}{26} = 14,793 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{169}{26} = 1,243 \%$$

Percobaan 2 :

$$\text{Rata-rata kesalahan sumbu X} = \frac{540}{25} = 21,6 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{1407}{25} = 56,28 \text{ pixel}$$

$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{540}{25} = 11,429 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{1407}{25} = 13,627 \%$$

Rata-rata kesalahan dari percobaan 1 dan percobaan 2 :

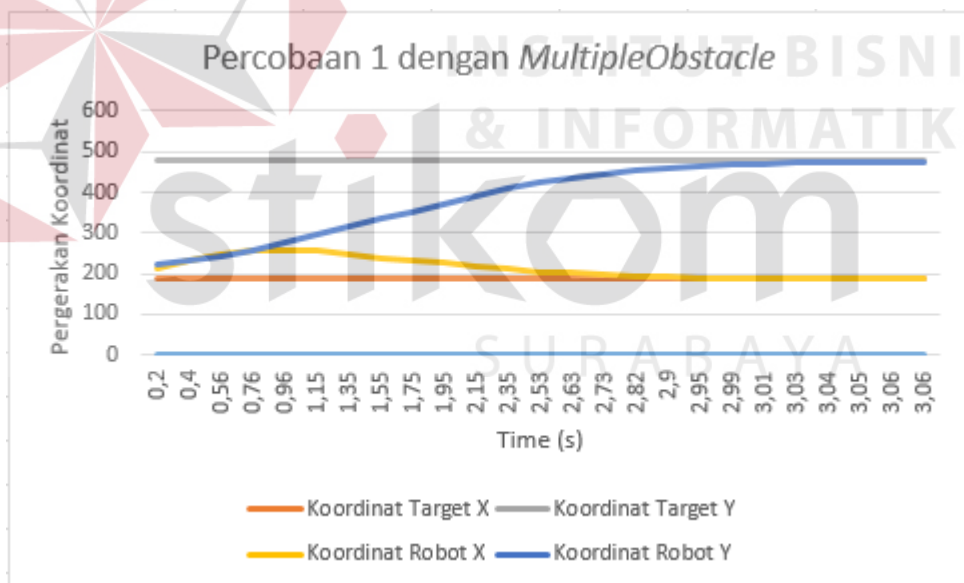
$$\text{Rata-rata kesalahan sumbu X} = \frac{69,23 + 21,6}{2} = 45,415 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{6,5 + 56,28}{2} = 31,39 \text{ pixel}$$

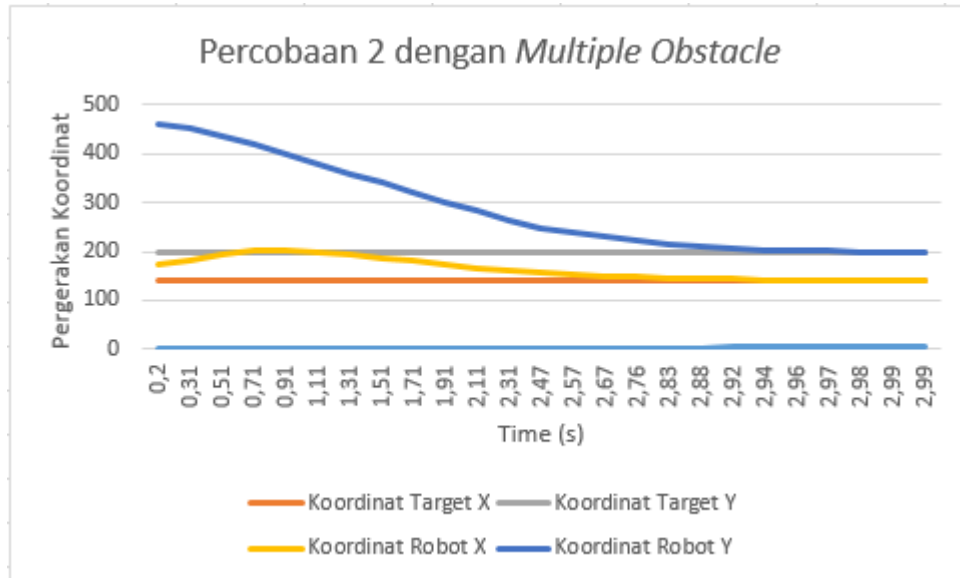
$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{69,23 + 21,6}{2} = 9,704 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{6,5 + 56,28}{2} = 6,002 \%$$

3. Melewati *Multiple Obstacle*



Gambar 4.10. Grafik Percobaan 1 Gerak Robot Menuju Target dengan *Multiple Obstacle* Menggunakan Metode *Fuzzy*



Gambar 4.11. Grafik Percobaan 2 Gerak Robot Menuju Target dengan *Multiple Obstacle* Menggunakan Metode *Fuzzy*

Grafik dari pencatatan dua kali percobaan pergerakan robot menuju target dengan melewati *multiple obstacle* dapat dilihat pada Gambar 4.10 dan Gambar 4.11. Berdasarkan data dari Tabel 4.14 dan Tabel 4.15, dapat diketahui rata-rata dari kesalahan koordinat robot terhadap target sebagai berikut :

Percobaan 1 :

$$\text{Rata-rata kesalahan sumbu X} = \frac{683}{28} = 24,393 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{2226}{28} = 79,5 \text{ pixel}$$

$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{683}{28} = 13,114 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{2226}{28} = 16,667 \%$$

Percobaan 2 :

$$\text{Rata-rata kesalahan sumbu X} = \frac{592}{28} = 21,143 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{2241}{28} = 80,036 \text{ pixel}$$

$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{592}{28} = 15,102 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{2241}{28} = 40,422 \%$$

Rata-rata kesalahan dari percobaan 1 dan percobaan 2 :

$$\text{Rata-rata kesalahan sumbu X} = \frac{24,393 + 21,143}{2} = 22,768 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{79,5 + 80,036}{2} = 79,768 \text{ pixel}$$

$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{24,393 + 21,143}{2} = 14,108 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{79,5 + 80,036}{2} = 28,545 \%$$

4.2.2. Gerak Robot Terhadap Koordinat Target Menggunakan Metode

Virtual Force Field (VFF)

A. Hasil Pengujian

Pengujian ini mengambil beberapa data pergerakan robot menuju target menggunakan metode VFF ketika tidak ada *obstacle* sama sekali, ketika terdapat satu *obstacle*, serta ketika terdapat beberapa *obstacle*. Data yang dicatat adalah *error* koordinat robot terhadap koordinat target yang telah ditentukan, baik dalam bentuk *pixel* maupun persentase. Berikut adalah hasil pengujian gerak robot terhadap koordinat target menggunakan metode VFF.

Tabel 4.16. Pengujian Gerak Robot Menuju Target Tanpa *Obstacle* Menggunakan Metode VFF

Time (s)	Koordinat Target		Koordinat Robot		Kesalahan (Pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
0,2	281	525	129	535	152	10	54,103	1,901
0,4	281	525	149	534	132	9	47,001	1,651
0,6	281	525	169	532	112	7	39,899	1,402
0,8	281	525	189	531	92	6	32,796	1,152

Time (s)	Koordinat Target		Koordinat Robot		Kesalahan (Pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
1	281	525	209	530	72	5	25,694	0,903
1,18	281	525	227	529	54	4	19,302	0,678
1,28	281	525	237	528	44	3	15,751	0,553
1,38	281	525	247	527	34	2	12,200	0,429
1,48	281	525	256	527	25	2	8,772	0,308
1,55	281	525	264	526	17	1	6,212	0,218
1,60	281	525	269	526	12	1	4,400	0,155
1,64	281	525	272	526	9	1	3,116	0,109
1,66	281	525	275	525	6	0	2,207	0,078
1,68	281	525	277	525	4	0	1,563	0,055
1,69	281	525	278	525	3	0	1,107	0,039
1,70	281	525	279	525	2	0	0,784	0,028
1,71	281	525	279	525	2	0	0,555	0,020
1,71	281	525	280	525	1	0	0,393	0,014
1,72	281	525	280	525	1	0	0,279	0,010
1,72	281	525	280	525	1	0	0,197	0,007
Rata-Rata					38,825	2,548	13,817	0,485

Tabel 4.17. Pengujian 1 Gerak Robot Menuju Target dengan Satu *Obstacle* Menggunakan Metode VFF

Waktu (s)	Koordinat Target		Koordinat Robot		Kesalahan (Pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
0,2	392	558	129	537	263	21	67,101	3,740
0,25	392	558	134	538	258	20	65,921	3,536
0,43	392	558	149	546	243	12	61,866	2,081
0,63	392	558	166	558	226	0	57,756	0,042
0,83	392	558	182	569	210	11	53,476	1,990
1,03	392	558	200	578	192	20	48,892	3,559
1,23	392	558	219	584	173	26	44,072	4,727
1,43	392	558	239	589	153	31	39,086	5,477
1,63	392	558	259	590	133	32	34,008	5,801
1,83	392	558	279	590	113	32	28,911	5,694
2,03	392	558	298	587	94	29	23,868	5,163
2,23	392	558	318	582	74	24	18,944	4,231
2,43	392	558	337	576	55	18	14,083	3,144
2,53	392	558	346	573	46	15	11,651	2,601
2,63	392	558	356	569	36	11	9,220	2,058
2,73	392	558	365	566	27	8	6,790	1,516
2,81	392	558	373	564	19	6	4,809	1,074
2,87	392	558	379	562	13	4	3,406	0,760
2,91	392	558	383	561	9	3	2,412	0,539
2,94	392	558	385	560	7	2	1,708	0,381

Waktu (s)	Koordinat Target		Koordinat Robot		Kesalahan (Pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
2,96	392	558	387	560	5	2	1,210	0,270
2,97	392	558	389	559	3	1	0,857	0,191
2,98	392	558	390	559	2	1	0,607	0,136
2,99	392	558	390	559	2	1	0,430	0,096
2,99	392	558	391	558	1	0	0,304	0,068
3,00	392	558	391	558	1	0	0,216	0,048
3,00	392	558	391	558	1	0	0,153	0,034
3,00	392	558	392	558	0	0	0,116	0,026
Rata-Rata					84,263	11,755	21,496	2,107

Tabel 4.18. Pengujian 2 Gerak Robot Menuju Target dengan Satu *Obstacle* Menggunakan Metode VFF

Waktu (s)	Koordinat Target		Koordinat Robot		Kesalahan (Pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
0,2	172	486	200	255	28	231	16,241	47,436
0,32	172	486	212	258	40	228	23,141	46,980
0,38	172	486	218	260	46	226	26,662	46,574
0,44	172	486	223	262	51	224	29,371	46,068
0,51	172	486	228	267	56	219	32,471	45,078
0,70	172	486	239	283	67	203	38,677	41,857
0,90	172	486	246	301	74	185	42,936	38,034
1,10	172	486	251	321	79	165	45,691	34,039
1,30	172	486	253	340	81	146	47,056	29,954
1,50	172	486	253	360	81	126	47,012	25,842
1,70	172	486	250	380	78	106	45,560	21,761
1,90	172	486	245	400	73	86	42,723	17,773
2,10	172	486	238	418	66	68	38,544	13,935
2,30	172	486	229	436	57	50	33,085	10,304
2,50	172	486	217	452	45	34	26,425	6,934
2,60	172	486	211	460	39	26	22,547	5,402
2,70	172	486	203	466	31	20	18,183	4,045
2,80	172	486	195	472	23	14	13,455	2,873
2,88	172	486	188	476	16	10	9,542	2,028
2,93	172	486	184	479	12	7	6,758	1,436
2,97	172	486	180	481	8	5	4,787	1,017
3,00	172	486	178	483	6	3	3,390	0,720
3,02	172	486	176	484	4	2	2,401	0,510
3,03	172	486	175	484	3	2	1,701	0,361
3,04	172	486	174	485	2	1	1,204	0,256
3,05	172	486	173	485	1	1	0,853	0,181
3,06	172	486	173	485	1	1	0,604	0,128
3,06	172	486	173	486	1	0	0,428	0,091

Waktu (s)	Koordinat Target		Koordinat Robot		Kesalahan (Pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
3,06	172	486	172	486	0	0	0,000	0,064
Rata-Rata					36,858	82,399	21,429	16,955

Tabel 4.19. Pengujian 1 Gerak Robot Menuju Target dengan Dua *Obstacle* Menggunakan Metode VFF

Waktu (s)	Koordinat Target		Koordinat Robot		Kesalahan (Pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
0,2	197	543	200	255	3	288	1,490	52,954
0,28	197	543	208	257	11	286	5,716	52,697
0,31	197	543	211	258	14	285	6,856	52,479
0,42	197	543	219	265	22	278	10,961	51,131
0,62	197	543	230	282	33	261	16,886	48,150
0,79	197	543	238	297	41	246	20,585	45,288
0,85	197	543	239	303	42	240	21,262	44,244
1,00	197	543	239	317	42	226	21,075	41,543
1,20	197	543	234	337	37	206	18,769	37,961
1,40	197	543	230	356	33	187	16,684	34,357
1,60	197	543	226	376	29	167	14,922	30,730
1,80	197	543	223	396	26	147	13,161	27,103
2,00	197	543	219	416	22	127	11,399	23,475
2,20	197	543	216	435	19	108	9,638	19,848
2,40	197	543	213	455	16	88	7,877	16,221
2,60	197	543	209	475	12	68	6,115	12,593
2,76	197	543	206	490	9	53	4,706	9,691
2,86	197	543	205	500	8	43	3,825	7,878
2,96	197	543	203	510	6	33	2,945	6,064
3,05	197	543	201	519	4	24	2,108	4,340
3,12	197	543	200	526	3	17	1,493	3,074
3,17	197	543	199	531	2	12	1,057	2,177
3,21	197	543	198	535	1	8	0,749	1,542
3,23	197	543	198	537	1	6	0,530	1,092
3,25	197	543	198	539	1	4	0,376	0,773
3,26	197	543	198	540	1	3	0,266	0,548
3,27	197	543	197	541	0	2	0,188	0,388
3,28	197	543	197	542	0	1	0,133	0,275
3,28	197	543	197	542	0	1	0,095	0,195
3,28	197	543	197	542	0	1	0,067	0,138
3,29	197	543	197	543	0	0	0,047	0,000
Rata-Rata					14,107	110,167	7,161	20,289

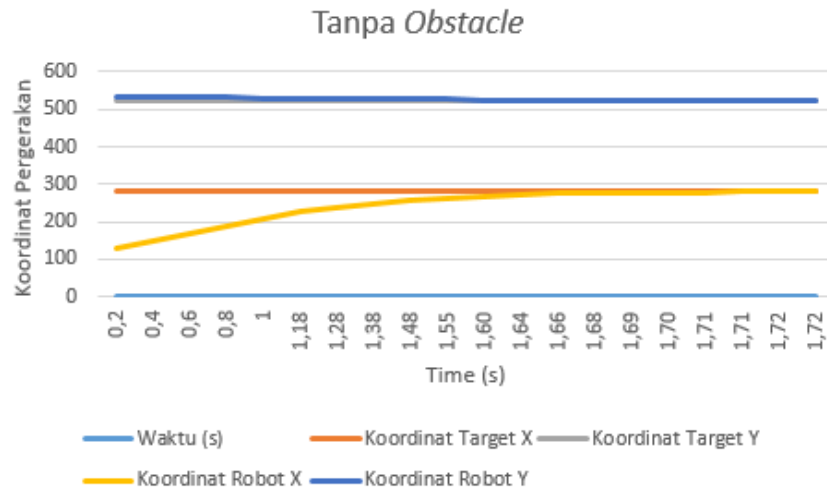
Tabel 4.20. Pengujian 2 Gerak Robot Menuju Target dengan Dua *Obstacle* Menggunakan Metode VFF

Waktu (s)	Koordinat Target		Koordinat Robot		Kesalahan (Pixel)		Kesalahan (%)	
	X	Y	X	Y	X	Y	X	Y
0,2	539	549	173	552	366	3	67,904	0,518
0,28	539	549	181	552	358	3	66,378	0,613
0,48	539	549	200	559	339	10	62,875	1,787
0,68	539	549	217	569	322	20	59,657	3,599
0,88	539	549	236	577	303	28	56,302	5,150
0,96	539	549	243	580	296	31	54,876	5,591
1,09	539	549	256	580	283	31	52,533	5,593
1,29	539	549	275	576	264	27	48,912	4,836
1,49	539	549	295	571	244	22	45,305	3,987
1,69	539	549	315	568	224	19	41,625	3,532
1,89	539	549	335	567	204	18	37,928	3,218
2,09	539	549	354	565	185	16	34,232	2,905
2,29	539	549	374	563	165	14	30,535	2,591
2,49	539	549	394	562	145	13	26,838	2,277
2,69	539	549	414	560	125	11	23,141	1,964
2,89	539	549	434	558	105	9	19,444	1,650
3,09	539	549	454	556	85	7	15,748	1,336
3,29	539	549	474	555	65	6	12,051	1,023
3,43	539	549	488	553	51	4	9,463	0,803
3,53	539	549	498	553	41	4	7,615	0,646
3,63	539	549	508	552	31	3	5,766	0,489
3,72	539	549	517	551	22	2	4,095	0,347
3,79	539	549	523	550	16	1	2,900	0,246
3,83	539	549	528	550	11	1	2,054	0,174
3,86	539	549	531	550	8	1	1,455	0,123
3,89	539	549	533	549	6	0	1,030	0,087
3,90	539	549	535	549	4	0	0,730	0,062
3,91	539	549	536	549	3	0	0,517	0,044
3,92	539	549	537	549	2	0	0,366	0,031
3,93	539	549	538	549	1	0	0,259	0,022
3,93	539	549	538	549	1	0	0,184	0,016
3,93	539	549	538	549	1	0	0,130	0,011
3,94	539	549	539	549	0	0	0,092	0,008
Rata-Rata					129,514	9,196	24,029	1,675

B. Pembahasan

Dari Tabel 4.16 sampai 4.20 didapatkan hasil pergerakan robot menuju target menggunakan metode VFF seperti berikut.

1. Tanpa *Obstacle*



Gambar 4.12. Grafik Pengujian Gerak Robot Menuju Target Tanpa *Obstacle* Menggunakan Metode VFF

Grafik dari pencatatan pergerakan robot menuju target tanpa melewati *obstacle* dapat dilihat pada Gambar 4.12. Berdasarkan data dari Tabel 4.16, dapat diketahui rata-rata dari kesalahan koordinat robot terhadap target sebagai berikut :

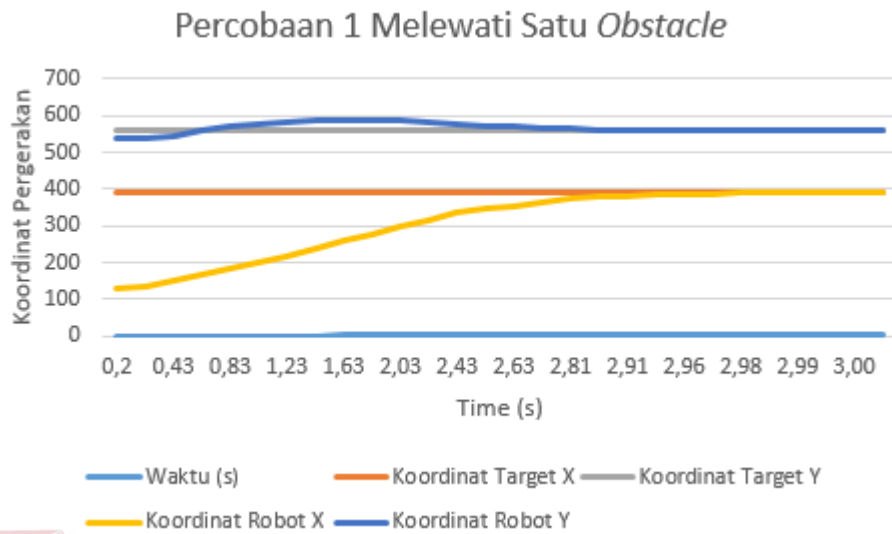
$$\text{Rata-rata kesalahan sumbu X} = \frac{776}{20} = 38,825 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{51}{20} = 2,548 \text{ pixel}$$

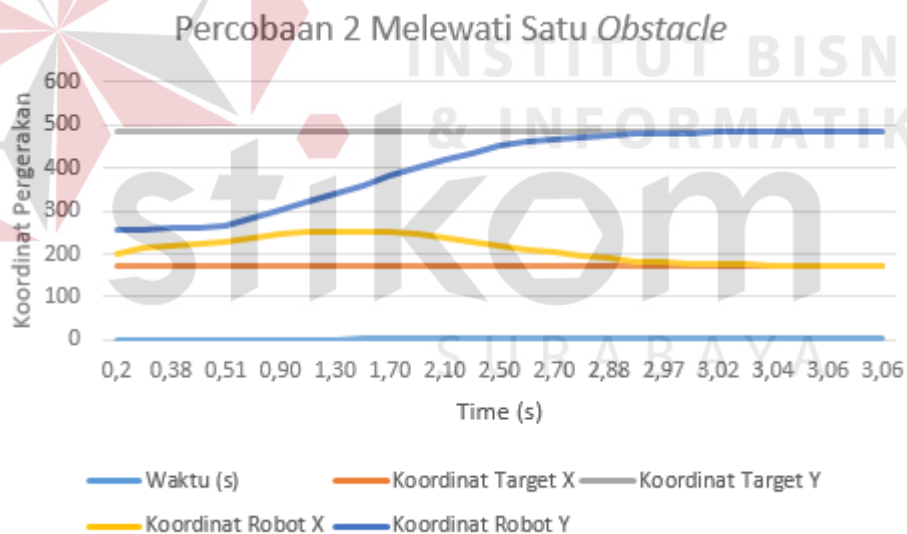
$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{776}{20} = 13,817 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{51}{20} = 0,485 \%$$

2. Melewati Satu *Obstacle*



Gambar 4.13. Grafik Percobaan 1 Pengujian Gerak Robot Menuju Target dengan Satu *Obstacle* Menggunakan Metode VFF



Gambar 4.14. Grafik Percobaan 2 Pengujian Gerak Robot Menuju Target dengan Satu *Obstacle* Menggunakan Metode VFF

Grafik dari pencatatan pergerakan robot menuju target tanpa melewati *obstacle* dapat dilihat pada Gambar 4.13 dan Gambar 4.14. Berdasarkan data dari

Tabel 4.17 dan Tabel 4.18, dapat diketahui rata-rata dari kesalahan koordinat robot terhadap target sebagai berikut :

Percobaan 1:

$$\text{Rata-rata kesalahan sumbu X} = \frac{2359}{28} = 84,263 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{329}{28} = 11,755 \text{ pixel}$$

$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{2359}{28} = 21,496 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{329}{28} = 2,107 \%$$

Percobaan 2:

$$\text{Rata-rata kesalahan sumbu X} = \frac{1069}{29} = 36,858 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{2390}{29} = 82,399 \text{ pixel}$$

$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{1069}{29} = 21,429 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{2390}{29} = 16,955 \%$$

Kesalahan rata-rata dari Percobaan 1 dan Percobaan 2 :

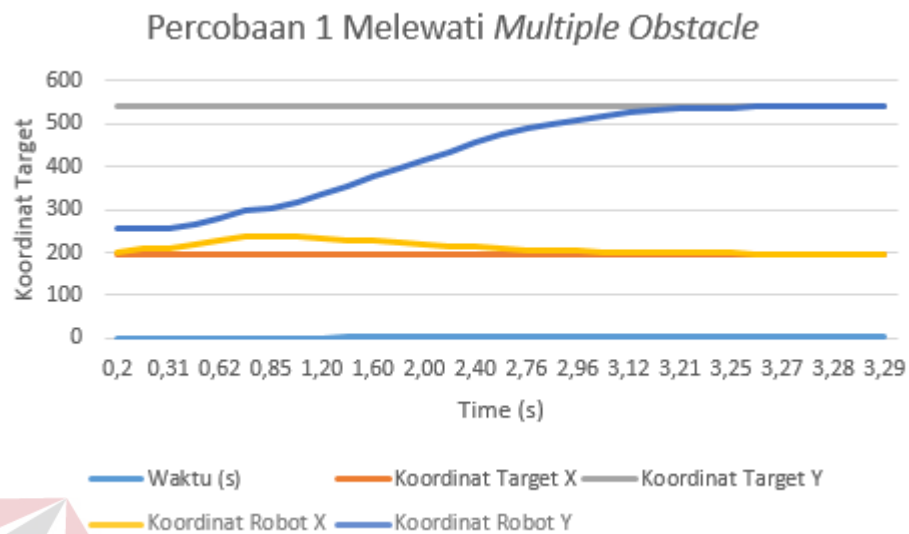
$$\text{Rata-rata kesalahan sumbu X} = \frac{84,263+36,858}{2} = 60,56 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{11,755+82,399}{2} = 47,077 \text{ pixel}$$

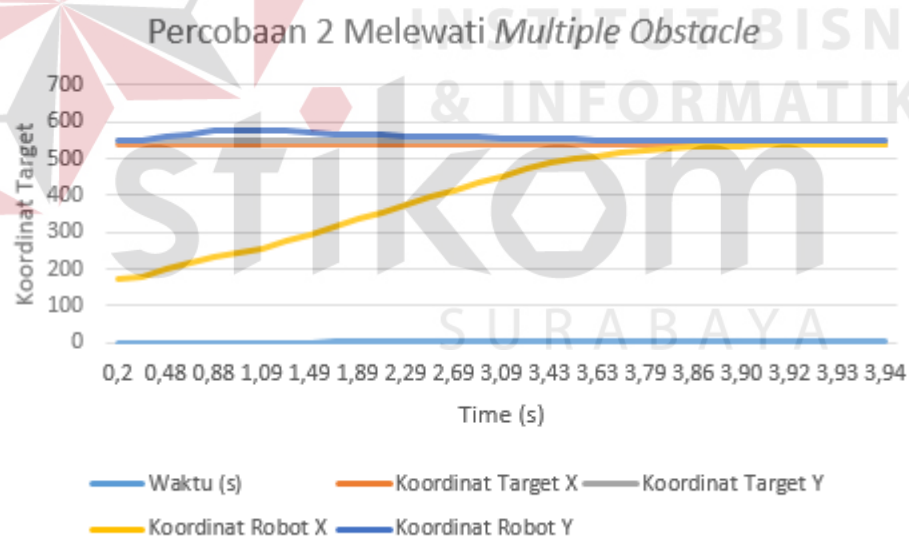
$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{21,496+21,429}{2} = 21,463 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{2,107+16,955}{2} = 9,531 \%$$

3. Melewati *Multiple Obstacle*



Gambar 4.15. Grafik Percobaan 1 Pengujian Gerak Robot Menuju Target dengan *Multiple Obstacle* Menggunakan Metode VFF



Gambar 4.16. Grafik Percobaan 2 Pengujian Gerak Robot Menuju Target dengan *Multiple Obstacle* Menggunakan Metode VFF

Grafik dari pencatatan pergerakan robot menuju target tanpa melewati *obstacle* dapat dilihat pada Gambar 4.15 dan Gambar 4.16. Berdasarkan data dari

Tabel 4.19 dan Tabel 4.20, dapat diketahui rata-rata dari kesalahan koordinat robot terhadap target sebagai berikut :

Percobaan 1:

$$\text{Rata-rata kesalahan sumbu X} = \frac{437}{31} = 14,107 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{3415}{31} = 110,167 \text{ pixel}$$

$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{437}{31} = 7,161 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{3415}{31} = 20,289 \%$$

Percobaan 2:

$$\text{Rata-rata kesalahan sumbu X} = \frac{4274}{33} = 129,514 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{303}{33} = 9,196 \text{ pixel}$$

$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{4274}{33} = 24,029 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{303}{33} = 1,675 \%$$

Kesalahan rata-rata dari Percobaan 1 dan Percobaan 2 :

$$\text{Rata-rata kesalahan sumbu X} = \frac{14,107+129,514}{2} = 71,81 \text{ pixel}$$

$$\text{Rata-rata kesalahan sumbu Y} = \frac{110,167+9,196}{2} = 89,682 \text{ pixel}$$

$$\text{Rata-rata persentase kesalahan sumbu X} = \frac{14,107+129,514}{2} = 15,59 \%$$

$$\text{Rata-rata persentase kesalahan sumbu Y} = \frac{110,167+9,196}{2} = 10,98 \%$$

4.3 Analisa Perbandingan Keseluruhan Sistem Menggunakan Metode *Fuzzy Logic Controller (Fuzzy)* dan Metode *Virtual Force Field (VFF)*

Analisa perbandingan ini dilakukan untuk mengetahui metode mana yang lebih efektif digunakan untuk mengatasi permasalahan dalam penelitian ini. Ada beberapa faktor yang dapat diuji untuk membuktikan metode yang lebih efektif.

4.3.1 Pengujian Perbandingan Waktu Tempuh Robot

A. Hasil Pengujian

Pengujian ini dilakukan dengan meletakkan koordinat awal robot, koordinat awal *obstacle*, dan koordinat target pada program yang menggunakan metode *Fuzzy* dan VFF di koordinat yang sama. Kemudian diamati manakah yang memiliki waktu tempuh lebih cepat.

Tabel 4.21. Pengujian Perbandingan Waktu Tempuh Robot Antara Kedua Metode Tanpa *Obstacle*

Pengujian	Koordinat Robot		Koordinat Target		Waktu Tempuh <i>Fuzzy</i> (s)	Waktu Tempuh VFF (s)
	X	Y	X	Y		
1.	122	227	792	556	2,5	2,51
2.	122	227	114	548	1,58	1,88
3.	166	458	792	92	2,43	2,44
4.	166	458	158	92	1,64	1,91
5.	852	537	92	92	3,25	3,78
6.	852	537	844	92	1,84	2,05
7.	858	163	92	492	3,2	3,78
8.	858	163	850	492	1,59	1,89
Rata-Rata					2,254	2,53

Tabel 4.22. Pengujian Perbandingan Waktu Tempuh Robot Antara Kedua Metode dengan Satu *Obstacle*

Pengujian	Koordinat Robot		Koordinat <i>Obstacle</i>		Koordinat Target		Waktu Tempuh <i>Fuzzy</i> (s)	Waktu Tempuh VFF (s)
	X	Y	X	Y	X	Y		
1.	122	227	320	300	792	556	2,5	2,55
2.	122	227	180	280	114	548	1,79	1,97
3.	166	458	270	330	792	92	2,48	2,45
4.	166	458	220	300	158	92	1,87	1,91
5.	852	537	900	400	92	92	3,56	3,78
6.	852	537	900	400	844	92	1,98	2,05
7.	858	163	920	200	92	492	3,56	3,84
8.	858	163	920	200	850	492	1,83	1,97
Rata-Rata							2,446	2,565

Tabel 4.23. Pengujian Perbandingan Waktu Tempuh Robot Antara Kedua Metode dengan *Multiple Obstacle*

No	Koordinat Robot		Koordinat <i>Obstacle 1</i>		Koordinat <i>Obstacle 2</i>		Koordinat <i>Obstacle 3</i>		Koordinat Target		Waktu Tempuh <i>Fuzzy</i> (s)	Waktu Tempuh VFF (s)
	X	Y	X	Y	X	Y	X	Y	X	Y		
1.	122	227	320	300	180	220	150	300	792	556	2,65	2,66
2.	122	227	180	220	100	310	130	400	114	548	1,61	2,6
3.	166	458	270	330	370	350	400	230	792	92	2,5	2,51
4.	166	458	220	300	300	230	220	150	158	92	1,86	1,98
5.	852	537	900	400	970	330	680	250	92	92	3,57	3,63
6.	852	537	900	400	970	330	890	250	844	92	2,07	2,36
7.	858	163	920	200	900	330	750	300	92	492	3,61	3,94
8.	858	163	920	200	900	330	820	270	850	492	1,87	7,17
Rata-Rata											2,468	3,356

4. Pembahasan

Dari pengujian yang dicatat pada Tabel 4.21 sampai Tabel 4.23, dapat diketahui rata-rata waktu tempuh robot jika menggunakan metode *Fuzzy* dan VFF adalah sebagai berikut :

A. Tanpa *Obstacle*

$$\text{Rata-rata waktu tempuh menggunakan } Fuzzy = \frac{18,03}{8} = 2,254 \text{ detik}$$

Rata-rata waktu tempuh menggunakan VFF = $\frac{20,24}{8} = 2,53$ detik

Sehingga perbandingan waktu tempuh robot menuju target tanpa melewati *obstacle* antara kedua metode adalah :

$$\frac{\text{Waktu Tempuh Fuzzy}}{\text{Waktu Tempuh VFF}} = \frac{2,254}{2,53} = \frac{1}{1,123}$$

B. Melewati Satu *Obstacle*

Rata-rata waktu tempuh menggunakan *Fuzzy* = $\frac{19,57}{8} = 2,446$ detik

Rata-rata waktu tempuh menggunakan VFF = $\frac{20,52}{8} = 2,565$ detik

Sehingga perbandingan waktu tempuh robot menuju target dengan melewati satu *obstacle* antara kedua metode adalah :

$$\frac{\text{Waktu Tempuh Fuzzy}}{\text{Waktu Tempuh VFF}} = \frac{2,446}{2,565} = \frac{1}{1,049}$$

C. Melewati *Multiple Obstacle*

Rata-rata waktu tempuh menggunakan *Fuzzy* = $\frac{19,74}{8} = 2,468$ detik

Rata-rata waktu tempuh menggunakan VFF = $\frac{26,85}{8} = 3,356$ detik

Sehingga perbandingan waktu tempuh robot menuju target dengan melewati *multiple obstacle* antara kedua metode adalah :

$$\frac{\text{Waktu Tempuh Fuzzy}}{\text{Waktu Tempuh VFF}} = \frac{2,468}{3,356} = \frac{1}{1,36}$$

4.3.2 Pengujian Perbandingan Jarak Tempuh Robot

A. Hasil Pengujian

Pengujian ini dilakukan dengan meletakkan koordinat awal robot, koordinat awal *obstacle*, dan koordinat target pada program yang menggunakan metode *Fuzzy* dan VFF di koordinat yang sama. Kemudian diamati manakah yang memiliki jarak tempuh lebih pendek.

Tabel 4.24. Pengujian Perbandingan Jarak Tempuh Robot Antara Kedua Metode Tanpa *Obstacle*

Pengujian	Koordinat Robot		Koordinat Target		Jarak Tempuh <i>Fuzzy (pixel)</i>	Jarak Tempuh VFF (<i>pixel</i>)
	X	Y	X	Y		
1.	122	227	792	556	757,931	759,364
2.	122	227	114	548	389,256	510,729
3.	166	458	792	92	729,401	731,416
4.	166	458	158	92	416,697	522,299
5.	852	537	92	92	1059,89	1271,56
6.	852	537	844	92	492,856	577,934
7.	858	163	92	492	1038,38	1270,36
8.	858	163	850	492	396,769	512,853
Rata-Rata					660,148	769,563

Tabel 4.25. Pengujian Perbandingan Jarak Tempuh Robot Antara Kedua Metode dengan Satu *Obstacle*

Pengujian	Koordinat Robot		Koordinat <i>Obstacle</i>		Koordinat Target		Jarak Tempuh <i>Fuzzy (pixel)</i>	Jarak Tempuh VFF (<i>pixel</i>)
	X	Y	X	Y	X	Y		
1.	122	227	320	300	792	556	757,931	767,188
2.	122	227	180	280	114	548	473,891	546,649
3.	166	458	270	330	792	92	741,137	739,948
4.	166	458	220	300	158	92	495,948	522,299
5.	852	537	900	400	92	92	1183,8	1271,56
6.	852	537	900	400	844	92	552,192	577,934
7.	858	163	920	200	92	492	1182,69	1291,45
8.	858	163	920	200	850	492	489,021	546,479
Rata-Rata							734,576	782,939

Tabel 4.26. Pengujian Perbandingan Jarak Tempuh Robot Antara Kedua Metode dengan *Multiple Obstacle*

No	Koordinat Robot		Koordinat Obstacle 1		Koordinat Obstacle 2		Koordinat Obstacle 3		Koordinat Target		Jarak Tempuh Fuzzy (pixel)	Jarak Tempuh VFF (pixel)
	X	Y	X	Y	X	Y	X	Y	X	Y		
1.	122	227	320	300	180	220	150	300	792	556	775,42	779,037
2.	122	227	180	220	100	310	130	400	114	548	373,481	712,778
3.	166	458	270	330	370	350	400	230	792	92	748,462	746,113
4.	166	458	220	300	300	230	220	150	158	92	486,369	507,038
5.	852	537	900	400	970	330	680	250	92	92	1154	1125,21
6.	852	537	900	400	970	330	890	250	844	92	554,851	580,977
7.	858	163	920	200	900	330	750	300	92	492	1185,5	1334,64
8.	858	163	920	200	900	330	820	270	850	492	492,994	1805,72
Rata-Rata											721,384	948,937

B. Pembahasan

Dari pengujian yang dicatat pada Tabel 4.24 sampai Tabel 4.26, dapat diketahui rata-rata jarak tempuh robot jika menggunakan metode *Fuzzy* dan VFF adalah sebagai berikut :

A. Tanpa Obstacle

Rata-rata jarak tempuh menggunakan *Fuzzy* = $\frac{5281,18}{8} = 660,148 \text{ pixel}$

Rata-rata jarak tempuh menggunakan VFF = $\frac{6156,52}{8} = 796,565 \text{ pixel}$

Sehingga perbandingan jarak tempuh robot menuju target tanpa melewati *obstacle* antara kedua metode adalah :

$$\frac{\text{Jarak Tempuh Fuzzy}}{\text{Jarak Tempuh VFF}} = \frac{660,148}{796,565} = \frac{1}{1,166}$$

B. Melewati Satu *Obstacle*

Rata-rata jarak tempuh menggunakan *Fuzzy* = $\frac{5876,606}{8} = 734,576 \text{ pixel}$

Rata-rata jarak tempuh menggunakan *VFF* = $\frac{6263,51}{8} = 782,939 \text{ pixel}$

Sehingga perbandingan jarak tempuh robot menuju target dengan melewati satu *obstacle* antara kedua metode adalah :

$$\frac{\text{Jarak Tempuh Fuzzy}}{\text{Jarak Tempuh VFF}} = \frac{734,576}{782,939} = \frac{1}{1,066}$$

C. Melewati *Multiple Obstacle*

Rata-rata jarak tempuh menggunakan *Fuzzy* = $\frac{5771,07}{8} = 721,384 \text{ pixel}$

Rata-rata jarak tempuh menggunakan *VFF* = $\frac{7591,498}{8} = 948,937 \text{ pixel}$

Sehingga perbandingan jarak tempuh robot menuju target dengan melewati *multiple obstacle* antara kedua metode adalah :

$$\frac{\text{Jarak Tempuh Fuzzy}}{\text{Jarak Tempuh VFF}} = \frac{721,384}{948,937} = \frac{1}{1,315}$$