

Implementasi Kendali Logika Fuzzy pada Robot Line Follower

Gilang Nugraha Putu Pratama*¹, Andi Dharmawan², Catur Atmaji³

¹Program Studi S1 Elektronika Instrumentasi FMIPA UGM

^{2,3}Jurusan Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta

e-mail: *¹ace.player.lang@gmail.com, ²dharmawan.andi@gmail.com,

³catur_atmaji@ugm.ac.id

Abstrak

Robot line follower merupakan robot otonom yang dapat mengikuti jalur. Jalurnya bisa berupa garis berwarna hitam diatas permukaan putih atau sebaliknya. Pada penelitian ini, robot line follower menggunakan sistem kendali logika fuzzy dengan metode Mamdani. Selama ini robot line follower kebanyakan dirancang untuk jalur dengan lebar yang tetap, namun dengan sistem kendali logika fuzzy ini dapat mengenali jalur dengan variasi lebar antara satu hingga delapan titik sensor. Robot line follower ini mengimplementasikan 18 aturan fuzzy untuk memetakan antara antecedent posisi dan lebar jalur, dengan consequent kecepatan laju robot. Aturan fuzzy terdiri dari masing-masing 9 aturan untuk kondisi jalur tunggal dan jalur percabangan dua jalur. Robot line follower ini mampu menganalisis 57 case jalur dengan kendali fuzzy, mulai dari lebar jalur 2 hingga 12 cm. Dimana 36 case analisis jalur tunggal dan 21 case jalur percabangan dua jalur. Robot line follower ini juga mampu menyesuaikan kecepatan laju sesuai lebar jalurnya.

Kata kunci— kendali logika fuzzy, kendali Mamdani, robot line follower

Abstract

Line follower Robot is an autonomous robot that can follow a track. The track can be a black line on a white surface or vice versa. In this study, the line follower robot using fuzzy logic control system by the method of Mamdani. Mostly line follower robots are designed with a fixed width, but with fuzzy logic control system it can recognize the wide variation between one to eight points of sensor. This line follower robot implements 18 fuzzy rules to map between the antecedents position and width of the line, with a consequents speed rate of the robot. There are 9 rules each for single line and two routes branching paths. This line follower robot is designed with capability to analyze 57 cases, the width of the line from 2 to 12 cm. There are 36 cases of analysis on a single line and 21 cases on two lines branching paths. This line follower robot can adjust it's speed depend on the wide of the track.

Keywords— fuzzy logic controller, Mamdani-controller, line follower robot

1. PENDAHULUAN

Selama ini sistem kendali yang sering digunakan dalam robot *line follower* adalah PID (*Proportional-Integral-Derivative*) dalam berbagai rancangan dan artikel penelitian. Sistem kendali PID merumuskan secara matematis, untuk mencapai kondisi ideal *set point* dengan cara memperbaiki nilai *error* hingga nol melalui *feedback*. Nilai *set point* dan *error* berasal dari pembacaan posisi robot *line follower* terhadap garis. Dengan menentukan *set point* robot *line follower* akan bermanuver memposisikan berada di tengah jalur, kendalanya adalah selama ini kondisi *set point* tersebut dirancang sama seperti lebar jalurnya yang ‘tetap’, tidak banyak

menggunakan variasi lebar jalur. Melalui penelitian ini dirancang robot *line follower* yang dapat mengenali jalur dengan lebar mulai 2 hingga 12 cm dengan menggunakan sistem kendali logika *fuzzy* metode Mamdani.

Telah banyak penelitian maupun artikel yang membahas tentang robot *line follower*, dan berikut sumber acuan yang mendukung penyusunan penelitian ini.

Penelitian rancangan sistem sebelumnya navigasi robot *line follower* dengan menggunakan sistem kendali logika *fuzzy* yang menganalisis pembacaan garis pada sensor garis kanan dan kiri untuk menentukan kecepatan kedua motornya. Rancangan ini menerapkan prinsip pembacaan posisi jalur dengan lebar tetap, yang akan menentukan arah lajunya [1].

Pada penelitian lainnya mengembangkan robot *line follower* dengan kendali logika *fuzzy* dimana robot *line follower* tersebut mampu berjalan diatas permukaan jalur dengan lebar yang tetap dan tanpa percabangan. Kaidah *fuzzy* yang diterapkan pada robot *line follower* tersebut adalah berdasarkan intensitas cahaya yang dipantulkan oleh permukaan jalur, dimana terdapat jalur '*black*', '*grey*', dan '*white*'. Dengan tingkat intensitas pantulan cahaya yang berbeda untuk tiga kategori tersebut, robot *line follower* dapat mengetahui posisinya terhadap jalur [2].

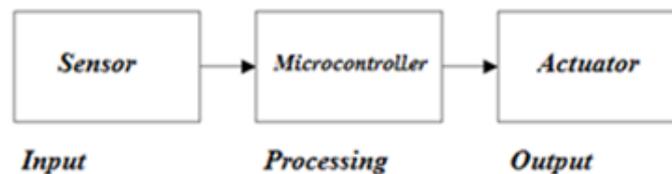
Pada penelitian sebelumnya yang juga menggunakan sistem kendali logika *fuzzy*, keduanya masih terbatas pada lebar jalur yang 'tetap'. Hal tersebut menjadi celah menarik dan potensial untuk dikembangkan, robot *line follower* yang dapat menyesuaikan kecepatannya sesuai dengan kondisi lebar jalurnya. Dengan menganalogikan bahwa robot *line follower* sebagai sebuah mobil dan jalurnya adalah sebuah jalan raya, maka menjadi logis bila pada jalan raya yang lebar laju mobil akan lebih cepat dibanding pada jalan yang sempit. Pemikiran seperti itulah yang menjadi dasar penelitian ini.

2. METODE PENELITIAN

Metode penelitian meliputi perancangan perangkat keras, kendali logika *fuzzy*, dan metode pengujian.

2.1 Perancangan Perangkat Keras

Secara garis besar sebuah robot terdiri dari tiga bagian utama, seperti pada Gambar 1.



Gambar 1 Blok diagram robot

Pada Gambar 1, blok diagram robot terdiri dari tiga bagian utama yakni sensor sebagai sistem *input*, yang berfungsi untuk mendeteksi jalur dan mengetahui posisi robot *line follower* terhadap jalur tersebut. Sensor yang digunakan adalah sensor garis terdiri dari pasangan LED (*Light-Emitting Diode*) *superbright* yang akan memancarkan berkas sinar, dan *photo-diode* sebagai penerima pantulan sinar LED *superbright* diatas permukaan yang berwarna putih atau hitam. Warna putih cenderung memantulkan sinar, sedangkan warna hitam menyerap sinar yang datang. Besarnya intensitas cahaya yang dipantulkan kembali ke *photo-diode* akibat permukaan putih maupun hitam, akan menghasilkan tegangan yang berbeda. Perbedaan tegangan saat berada diatas permukaan putih dan hitam itulah, yang digunakan untuk membedakan jalur hitam dan permukaan putih.

Bagian kedua adalah bagian *processor microcontroller* yang berfungsi sebagai 'otak' pengambil keputusan *output* apa yang harus dilakukan berdasarkan *input* dari sensor. Terakhir

adalah *actuator* bagian *output* yang melakukan hasil komputasi *processor* yang berupa dua buah motor DC di sisi kiri dan kanan robot *line follower* [3]. Kecepatan motor DC dikendalikan oleh *microcontroller* sesuai kondisi jalur yang dideteksi sensor garis.

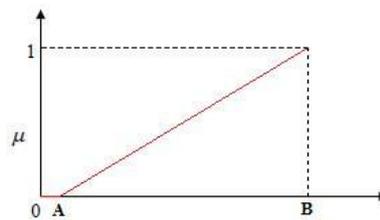
2.2 Logika Fuzzy

Logika *fuzzy* dikemukakan oleh Dr. Lofti Zadeh pada tahun 1965, yang merupakan fungsi matematika untuk menangani ketidakpastian. Dengan logika *fuzzy*, proses komputasi dilakukan dengan pendekatan bahasa alami. Teori *fuzzy* memberikan mekanisme untuk merepresentasikan bahasa alami semisal ‘banyak’, ‘sedikit’, ‘sering’, ‘jarang’. Secara umum, logika *fuzzy* memungkinkan kemampuan penalaran seperti manusia. Sebaliknya, teori tradisional himpunan biner hanya mendeskripsikan *event* yang bersifat mutlak, yakni bersifat ‘*true*’ atau ‘*false*’ saja. Berikut persamaan 1 yang merepresentasikan notasi derajat keanggotaan himpunan *fuzzy* yang dinyatakan dalam μA .

$$\mu A : x [0,1] \quad (1)$$

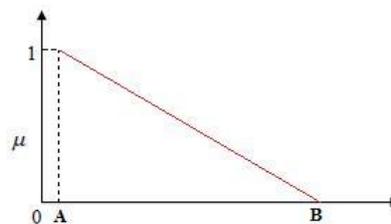
Fungsi keanggotaan $\mu A(x)$ akan bernilai ‘*true*’ (1) jika x merupakan anggota penuh dari himpunan A , dan akan bernilai ‘*false*’ (0) jika x bukan anggota himpunan A . Sementara bila derajat keanggotaan dangkan jika derajat keanggotaan berada dalam selang (0,1), misalnya $\mu A(x) = \mu$, menyatakan x sebagian anggota himpunan A dengan derajat keanggotaan sebesar μ .

Terdapat tiga representasi yang digunakan dalam analisis himpunan *fuzzy*, yakni representasi secara linier, segitiga, dan trapesium. Tampak seperti pada Gambar 2, 3, 4, dan 5. Dalam representasi linier, memetakan *input crisp* ke derajat keanggotaannya dapat digambarkan sebagai suatu garis lurus. Bentuk ini paling sederhana dan merupakan pilihan yang baik untuk mendekati suatu konsep yang kurang jelas dalam tugas akhir ini. Terdapat dua keadaan himpunan *fuzzy* yang linier [4].



Gambar 2 Representasi himpunan *fuzzy* naik
Persamaan 2 fungsi keanggotaan *fuzzy* naik:

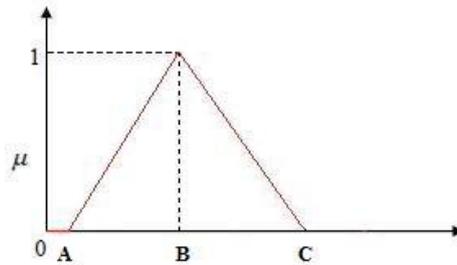
$$\mu(x) = \begin{cases} 0, & x \leq A \\ \frac{(x - A)}{(B - A)}, & A < x \leq B \end{cases} \quad (2)$$



Gambar 3 Representasi himpunan *fuzzy* turun

Persamaan 3 fungsi keanggotaan *fuzzy* turun:

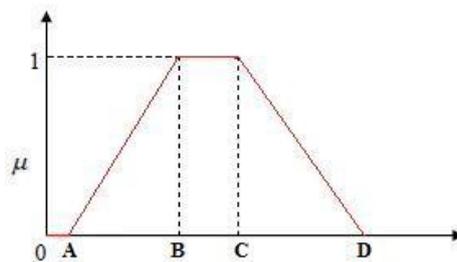
$$\mu(x) = \begin{cases} \frac{(B - x)}{(B - A)}, & A \leq x < B \\ 0, & x \geq B \end{cases} \quad (3)$$



Gambar 4 Representasi himpunan *fuzzy* segitiga

Persamaan 4 fungsi keanggotaan *fuzzy* segitiga:

$$\mu(x) = \begin{cases} 0, & A \leq x \vee x \geq B \\ \frac{(x - A)}{(B - A)}, & A < x \leq B \\ \frac{(C - x)}{(D - C)}, & C \leq x < D \\ 1, & B \leq x \leq C \end{cases} \quad (4)$$

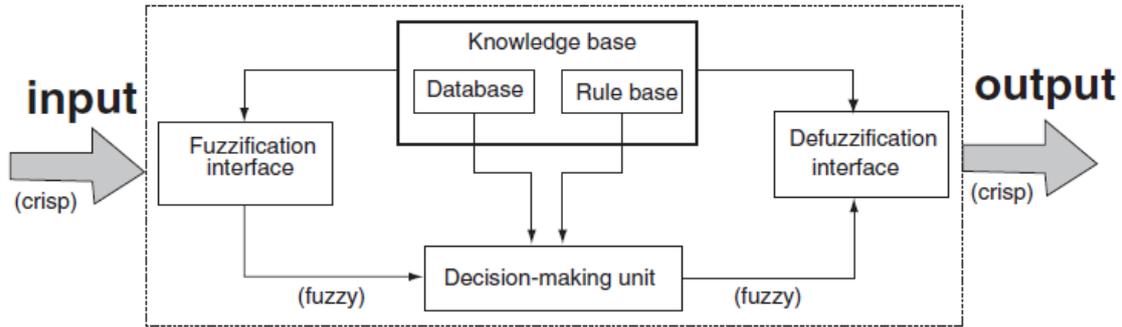


Gambar 5 Representasi himpunan *fuzzy* trapesium

Persamaan 5 fungsi keanggotaan *fuzzy* trapesium:

$$\mu(x) = \begin{cases} 0, & A \leq x \vee x \geq B \\ \frac{(x - A)}{(B - A)}, & A < x \leq B \\ \frac{(C - x)}{(D - C)}, & C \leq x < D \\ 1, & B \leq x \leq C \end{cases} \quad (5)$$

Gambar 6 menunjukkan skema dasar dari *Fuzzy Inference System*.



Gambar 6 Skema dasar *Fuzzy Inference System*

Blok *Fuzzy Inference System*, menggunakan aturan *fuzzy* ‘If-Then’ dalam memetakan ruang himpunan *input fuzzy* X untuk menghasilkan himpunan *output fuzzy* Y berdasarkan prinsip logika *fuzzy*. Sesuai Gambar 6 proses dimulai dengan *fuzzification* yakni memetakan nilai masukan *crisp* posisi dan lebar jalur terdeteksi robot *line follower* dalam nilai masukan *fuzzy*, kemudian dilakukan *rule evaluation* melakukan evaluasi aturan mana saja yang sesuai kondisi terkini dengan melihat *rule base*, terakhir *defuzzification* memetakan nilai keluaran *fuzzy* dalam nilai keluaran *crisp* sistem yakni kecepatan motor kanan dan kiri.

2.3 Mamdani *Fuzzy Inference System*

Mamdani-FIS merupakan salah satu metode penalaran (*inference*) yang paling sering digunakan untuk persoalan kendali logika *fuzzy*. Metode ini dikemukakan oleh Mamdani dan Assilian (1975) untuk mengendalikan mesin uap dan mendidihkan berdasarkan sintesis himpunan kendali aturan *linguistic* dari percobaan operator seorang manusia. Metode Mamdani berdasarkan pada makalah Dr. Lofti Zadeh (1973) tentang algoritma *fuzzy* untuk sistem yang kompleks dan proses pengambilan keputusan. Kelebihan metode Mamdani dibandingkan FIS yang lain, diantaranya adalah karena bersifat intuitif, mencakup bidang yang luas tidak terbatas persamaan linier, dan sesuai dengan proses *input* informasi manusia.

2.4 Perancangan Aturan *Fuzzy*

Tabel 1 menunjukkan aturan *fuzzy* yang digunakan untuk jalur tunggal dan Tabel 2 aturan untuk jalur percabangan.

Tabel 1 Aturan *fuzzy* jalur tunggal

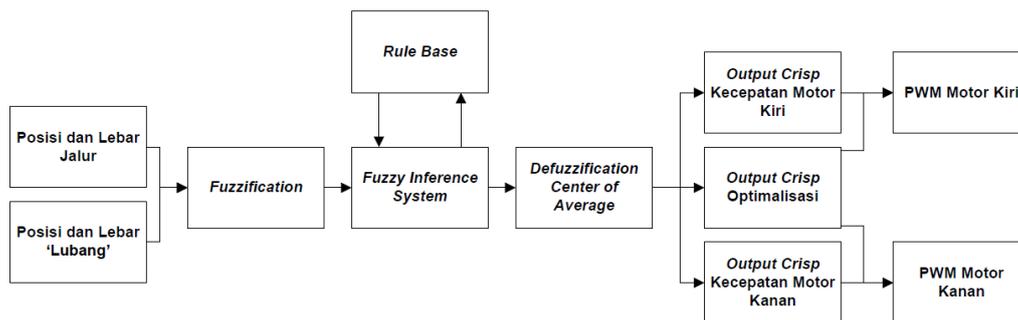
Rule Base		Lebar Jalur		
		‘sempit’	‘sedang’	‘lebar’
Posisi	‘kiri’	LM: ‘sangat pelan’ RM: ‘pelan’ OP: ‘sangat rendah’	LM: ‘pelan’ RM: ‘cepat’ OP: ‘rendah’	LM: ‘cepat’ RM: ‘sangat cepat’ OP: ‘sedang’
	‘tengah’	LM: ‘pelan’ RM: ‘pelan’ OP: ‘rendah’	LM: ‘cepat’ RM: ‘cepat’ OP: ‘sedang’	LM: ‘sangat cepat’ RM: ‘sangat cepat’ OP: ‘tinggi’
	‘kanan’	LM: ‘pelan’ RM: ‘sangat pelan’ OP: ‘sangat rendah’	LM: ‘cepat’ RM: ‘pelan’ OP: ‘rendah’	LM: ‘sangat cepat’ RM: ‘cepat’ OP: ‘sedang’

Tabel 2 Aturan *fuzzy* jalur percabangan

Rule Base		Lebar 'Lubang'		
		'sempit'	'sedang'	'lebar'
Posisi	'kiri'	LM: 'sangat cepat' RM: 'cepat' OP: 'tinggi'	LM: 'cepat' RM: 'pelan' OP: 'sedang'	LM: 'pelan' RM: 'sangat pelan' OP: 'rendah'
	'tengah'	LM: 'cepat' RM: 'pelan' OP: 'sedang'	LM: 'pelan' RM: 'sangat pelan' OP: 'rendah'	LM: 'pelan' RM: 'sangat pelan' OP: 'sangat rendah'
	'kanan'	LM: 'cepat' RM: 'sangat cepat' OP: 'tinggi'	LM: 'pelan' RM: 'sangat cepat' OP: 'sedang'	LM: 'sangat pelan' RM: 'sangat cepat' OP: 'rendah'

Dimana LM adalah kecepatan motor sisi kiri, RM adalah kecepatan motor sisi kanan, dan OP adalah optimalisasi kecepatan motor. Pada sistem robot *line follower* ini *rule base* yang digunakan berdasarkan rancangan himpunan *fuzzy* untuk lebar jalur, posisi, kecepatan motor kiri dan kanan, dan optimalisasi. Bentuk umum aturannya adalah sebagai berikut 'If' lebar jalur ('lubang') adalah A dan posisi adalah B, 'Then' kecepatan motor kiri adalah C, kecepatan motor kanan adalah D, dan optimalisasinya adalah E.

Keseluruhan rancangan sistem tampak seperti pada Gambar 7.

Gambar 7 Skema algoritma sistem *fuzzy line follower*

Kecepatan akhir laju robot *line follower* ditentukan dari nilai *crisp* kecepatan dikalikan nilai optimalisasinya. Nilai optimalisasi dipengaruhi oleh lebar jalur, makin lebar jalurnya makin tinggi nilai optimalisasinya. Hal ini ditujukan agar robot *line follower* tidak melaju terlalu cepat di jalur yang sempit.

2.5 Metode Defuzzification Center of Average

Pada sistem *fuzzy* robot *line follower* ini menggunakan metode *defuzzification center of average*. Metode *center of average*, yakni metode yang menggunakan satu titik pusat area grafik mewakili keseluruhan himpunan. Persamaan 6 menunjukkan notasi matematikanya.

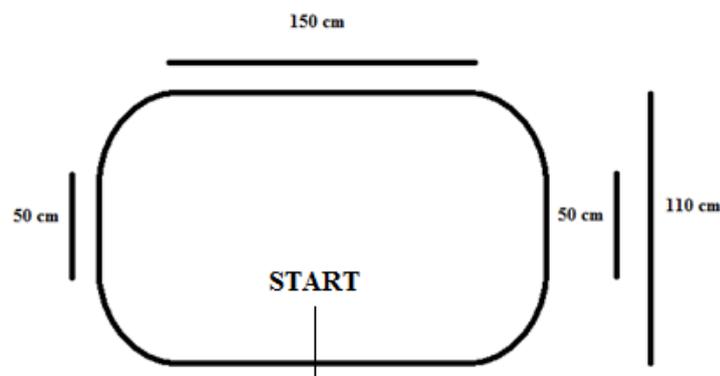
$$z^* = \frac{\sum_{l=1}^m z_l \mu_l}{\sum_{l=1}^m \mu_l} \quad (6)$$

Dengan hasil *defuzzification* (z^*) merupakan jumlah hasil kali titik pusat (z) dengan derajat keanggotaannya (μ) dari 1 sampai ke- m , yang dibagi jumlah seluruh derajat keanggotaan (μ) dari 1 sampai ke- m .

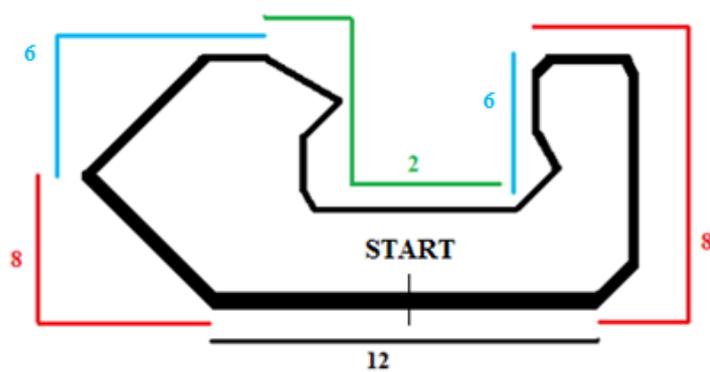
2.6 Perancangan Pengujian

Pada bagian rancangan pengujian ini meliputi pengujian fungsionalitas robot *line follower* pada jalur dengan lebar yang tetap, bervariasi, dan percabangan. Selain itu juga dengan membandingkan kinerja dengan robot *line follower* lain yang menggunakan sistem kendali PID.

Bagian penting pada bagian ini ialah jalur lintasan yang akan digunakan untuk menguji kinerja dari robot *line follower* tersebut. Terdapat empat jenis lintasan yang digunakan dalam pengujian ini yakni jalur biasa *ellipse* dengan lebar jalur empat titik sensor, jalur dengan variasi lebar jalur, jalur dengan percabangan, dan jalur lurus dengan transisi lebar jalur. Gambar 7 dan 8 menunjukkan rancangan lintasan jalur biasa dan jalur dengan variasi lebar jalur yang digunakan.



Gambar 7 Jalur lintasan biasa

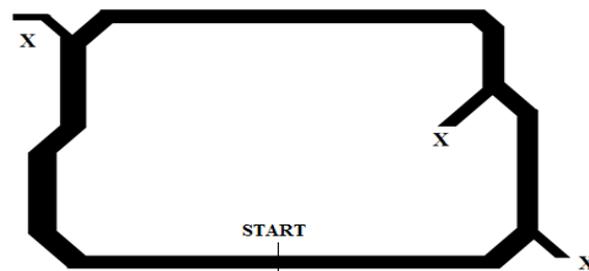


Gambar 8 Jalur lintasan lebar bervariasi

Jalur lintasan biasa pada Gambar 7 memiliki lebar jalur empat titik sensor atau 6 cm dengan panjang lintasan lurus mencapai 400 cm, sedangkan panjang jalur keseluruhan 520 cm. Jalur pada Gambar 8 dengan variasi lebar mulai dari 12 cm sepanjang 100 cm ditunjukkan dengan garis warna hitam, lebar 8 cm sepanjang 180 cm ditunjukkan dengan garis warna merah, lebar 6 cm sepanjang 120 cm ditunjukkan dengan garis warna biru, dan lebar 2 cm sepanjang 110 yang ditunjukkan dengan warna hijau.

Pada masing-masing pengujian baik jalur tunggal biasa dan jalur dengan variasi lebar, robot *line follower* diuji kemampuan dan daya tahannya selama lima menit. Selama lima menit dihitung berapa kali putaran penuh yang dapat dilakukan robot *line follower* tersebut. Selain itu juga dicatat waktu tercepat robot *line follower* untuk satu kali putaran penuh. Tujuan utama

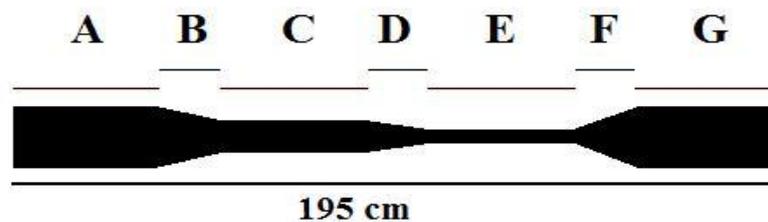
pengujian ini adalah mengetahui mampu tidaknya robot *line follower* melintasi jalur tersebut. Sedangkan untuk pengujian lintasan percabangan rancangan jalurnya tampak seperti Gambar 9.



Gambar 9 Jalur lintasan percabangan

Pada jalur percabangan memiliki lebar jalur empat titik sensor atau 6 cm pada bagian jalur yang lurus. Terdapat tiga jalur percabangan yang masing-masing salah satu percabangannya menuju jalur buntu yang ditandai huruf X. Tujuan penggunaan jalur percabangan ini untuk menguji kemampuan robot *line follower* melintasi jalur percabangan dengan menganalisis memilih jalur yang lebih lebar. Keseluruhan panjang lintasan utama yang digunakan adalah 260 cm.

Jalur pengujian terakhir adalah jalur lurus dengan transisi lebar jalur, tampak pada Gambar 10. Tujuan pengujian pada jalur ini adalah untuk mengetahui kemampuan dari robot *line follower* untuk menyesuaikan kecepatan terkait lebar jalurnya.



Gambar 10 Jalur lurus dengan transisi lebar

Jalur tersebut memiliki empat sektor lurus A, C, E, dan G, serta sektor transisi B, D, dan F. Setiap sektor lurus memiliki panjang lintasan yang sama yakni 35 cm, demikian sektor transisi semuanya sepanjang 15 cm. Panjang lintasan keseluruhan adalah 195 cm. Pengujian dilakukan dengan mencatat waktu tempuh robot *line follower* di setiap sektor, untuk kemudian dianalisis kecepatan laju di sektor tersebut.

3. HASIL DAN PEMBAHASAN

Pengujian perbandingan robot *line follower* dengan PID dan dengan kendali logika *fuzzy*. Data spesifikasi keduanya sebagaimana Tabel 3.

Tabel 3 Data perbandingan spesifikasi robot *line follower*

Data Spesifikasi Perbandingan Robot <i>Line Follower</i>		
	<i>Fuzzy Line Follower</i>	<i>PID Line Follower</i>
Panjang	22 cm	18 cm
Lebar	20 cm	12 cm
Diameter Roda	6,5 cm	4,2 cm
Waktu Laju 200 cm / 170 PWM	6 hingga 8 detik	4 hingga 5 detik
Waktu Tercepat Laju 200 cm / 170 PWM	6,3 detik	4,2 detik

Gambar 11 memperlihatkan perbandingan dimensi dua robot *line follower* tersebut, yang di sisi kiri adalah robot *line follower* dengan kendali logika *fuzzy*.



Gambar 11 Perbandingan ukuran robot *line follower*

Pengujian pertama dilakukan pada jalur biasa seperti pada Gambar 7, selama lima menit. Dengan mencatat waktu terbaik satu putaran dan berapa putaran penuh yang berhasil dilalui selang lima menit tersebut. Tabel 4 menunjukkan hasil data pengujian tersebut.

Tabel 4 Data hasil pengujian jalur biasa

(a) Hasil kinerja robot

	<i>Fuzzy Line Follower</i>	<i>PID Line Follower</i>
Waktu Tercepat	38,6 detik	31,3 detik
Jumlah Putaran Penuh	6 kali putaran	8 kali putaran

(b) Data lintasan

Bentuk Lintasan	<i>Ellipse</i>
Panjang Lintasan	520 cm
Lebar Jalur	4 titik sensor garis setara 6 cm

Seberapa banyak putaran penuh yang dilalui dan waktu tercepat kedua robot *line follower* tidak menunjukkan kinerja sistem kendali yang digunakan, karena adanya perbedaan spesifikasi. Hasil pengujian pada jalur biasa kedua robot *line follower* mampu menyelesaikan tantangan melintasi lintasan dalam lima menit. Kesimpulan pengujian ini adalah pada lintasan jalur biasa *ellipse*, kedua robot *line follower* berhasil melaluinya.

Pengujian yang selanjutnya dilakukan pada jalur dengan lebar yang bervariasi. Jalur tersebut sesuai Gambar 8 dengan panjang jalur keseluruhan 510 cm. Tabel 5 menunjukkan data hasil pengujiannya.

Tabel 5 Data hasil pengujian lintasan lebar bervariasi

(a) Hasil kinerja robot

	<i>Fuzzy Line Follower</i>	<i>PID Line Follower</i>
Waktu Tercepat	43,8 detik	Gagal
Jumlah Putaran Penuh	5 kali putaran	0 putaran gagal

(b) Data lintasan

Panjang Lintasan	510 cm
Lebar Jalur	Bervariasi 12 cm, 8 cm, 6 cm, dan 2 cm

Lebar jalur bervariasi antara 12 cm, 8 cm, 6 cm, dan 2 cm. Dengan *start* dari posisi lebar jalur 12 cm, kemudian melintasi jalur yang menyempit menjadi 8 cm, 6 cm, hingga 2 cm.

Setelah melewati bagian lebar jalur 2 cm, lintasan akan kembali melebar menjadi 6 cm, 8 cm, dan sampai kembali ke titik *start* 12 cm. Sama seperti pada pengujian jalur biasa, pada pengujian jalur dengan lebar yang bervariasi ini juga dilakukan selama lima menit. Berdasarkan data pada Tabel 5, robot *line follower* dengan kendali logika *fuzzy* mampu melintasi jalur dengan lebar yang bervariasi. Sedangkan robot *line follower* dengan PID sama sekali tidak dapat menyelesaikan satu putaran. Hal tersebut karena rancangan robot *line follower* PID menggunakan *set point* lebar jalur empat titik sensor saja. Sebagai contoh kondisi lebar jalur delapan titik tidak terdefinisi, sehingga robot *line follower* PID sama sekali tidak bergerak. Pada bagian lebar jalur yang terdeteksi empat titik, robot *line follower* PID dapat berjalan lancar.

Pengujian yang ketiga pada jalur percabangan yang mana rancangannya tampak pada Gambar 9. Tabel 6 menunjukkan data hasil pengujian jalur percabangannya.

Tabel 6 Data hasil pengujian jalur percabangan

(a) Hasil kinerja robot *line follower*

	<i>Fuzzy Line Follower</i>
Waktu Tercepat	26,5 detik
Jumlah Putaran Penuh	11 kali putaran
Jumlah Masuk Jebakan	Tidak pernah

(b) Data lintasan

Panjang Lintasan	260 cm
Lebar Jalur	4 titik sensor garis setara 6 cm
Jumlah Percabangan	3 percabangan

Tujuan pengujian di jalur percabangan ini adalah mengetahui kemampuan robot *line follower* untuk memilih jalur yang lebih lebar dibandingkan jalur yang lebih sempit. Pada pengujian ini hanya digunakan robot *line follower* dengan kendali *fuzzy* saja, tanpa menggunakan robot *line follower* PID sebagai pembanding. Hal tersebut dikarenakan robot *line follower* PID tersebut memang tidak dirancang untuk jalur percabangan. Pengujian ini juga dilakukan selama lima menit. Pada jalur percabangan tersebut terdapat jalur jebakan yang mana lebarnya lebih sempit dibandingkan jalur percabangan utama. Jalur percabangan jebakan tersebut berujung pada jalan buntu.

Berdasarkan hasil pengujian yang tampak pada Tabel 6, dapat disimpulkan robot *line follower* dengan kendali logika *fuzzy* tersebut berhasil melalui jalur percabangan. Dengan mampu melintasi lintasan percabangan sebanyak 11 kali putaran penuh dalam lima menit, serta catatan waktu tercepat 26,5 detik untuk satu kali putaran. Robot *line follower* tersebut juga berhasil memilih percabangan jalur yang lebih lebar, tidak masuk jebakan jalur percabangan.

Pengujian terakhir dilakukan pada jalur lurus (35 cm) dengan transisi lebar (15 cm), tampak seperti pada Gambar 10. Data pengujian tampak pada Tabel 7.

Tabel 7 Data hasil pengujian jalur lurus dengan transisi

	Sektor Lurus				Sektor Transisi					
	A	C	E	G	A-B-C	C-D-E	E-F-G	G-F-E	E-D-C	C-B-A
Waktu	1,2	2,1	3,2	1,3	0,8	1,1	0,7	0,6	1,1	0,7
Kecepatan	29,2	16,7	10,9	26,9	18,75	13,64	21,43	25	13,64	21,43
Keterangan:										
Waktu dalam detik										
Kecepatan dalam cm/detik										

Tujuan pengujian ini adalah untuk mengetahui kemampuan penyesuaian kecepatan robot *line follower* sesuai dengan lebar jalurnya. Setiap sektor lurus panjangnya 35 cm dan setiap sektor transisi panjangnya 15 cm. Sesuai Gambar 10 sektor lurus A dan G memiliki lebar 12 cm, sektor C lebarnya 6 cm, dan sektor E lebarnya 3 cm. Sektor transisi A-B-C dari 12 cm menyempit menjadi 6 cm, transisi C-D-E menyempit dari 6 cm menjadi 3 cm, transisi E-F-G melebar dari 3 cm menjadi 12 cm, transisi G-F-E menyempit dari 12 cm menjadi 3 cm, transisi E-D-C melebar dari 3 cm menjadi 6 cm, dan transisi C-B-A melebar dari 6 cm menjadi 12 cm.

Berdasarkan Tabel 7 data hasil pengujian menunjukkan bahwa kecepatan robot *line follower* berubah sesuai dengan lebar jalurnya. Semakin lebar jalurnya maka semakin cepat laju robot *line follower* tersebut. Kesimpulan pengujian ini dengan menggunakan kendali logika *fuzzy*, robot *line follower* berhasil menyesuaikan kecepatannya sesuai lebar jalurnya. Dimana kecepatan robot *line follower* berbanding lurus dengan lebar jalurnya.

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan dapat diambil kesimpulan bahwa telah berhasil diimplementasikan sistem kendali logika *fuzzy* pada robot *line follower* yang mampu melalui lintasan jalur tunggal dan percabangan berdasarkan definisi dan aturan *fuzzy* yang diterapkan. Adapun rincian hasil penelitian sebagai berikut:

1. Robot *line follower* berhasil melintasi jalur dengan lebar jalur yang tetap secara baik, tidak keluar dari jalurnya.
2. Robot *line follower* berhasil menganalisis dan melalui jalur percabangan, dengan memilih jalur yang lebih lebar pada percabangan sesuai dengan rancangan yang diinginkan.
3. Robot *line follower* berhasil menganalisis perubahan lebar jalur untuk menyesuaikan kecepatannya. Semakin lebar jalurnya, maka kecepatan robot *line follower* semakin cepat.

5. SARAN

Pada penelitian ini masih terdapat beberapa hal yang perlu disempurnakan. Pengembangan implementasi sistem kendali *hybrid*, dengan mengadopsi sistem ini dan menggabungkannya dengan sistem lain yang sesuai.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada keluarga, dosen, teman, dan pihak-pihak lain yang tidak dapat disebutkan satu per satu, yang telah memberi dukungan terhadap penelitian ini.

DAFTAR PUSTAKA

- [1] Azlan, N.Z., Zainudin, F., Yusuf, H.M., Toha, S.F., Yussoff, S.Z.S., Osman, N.H., 2007, *Fuzzy Logic Controlled Miniature LEGO Robot for Undergraduate Training System*, International Islamic University Malaysia, Kuala Lumpur, Malaysia
- [2] Ibrahim, D., Alshanableh, T., 2009, *An Undergraduate Fuzzy Logic Control Lab Using a Line Following Robot*, Wiley Periodicals, Inc., Lefkosa.
- [3] Sivanandam, S.N., Sumathi S., dan Deepa S.N., 2007, *Introduction to Fuzzy Logic using MATLAB*, Springer-Verlag, Berlin.
- [4] Solikin, F., 2011, Aplikasi Logika Fuzzy dalam Optimalisasi Produksi Barang menggunakan Metode Mamdani dan Metode Sugeno, *Skripsi*, Jurusan Pendidikan Matematika, UNY, Yogyakarta.