

Prototipe Sistem Auto Brake Pada Mobil Rc Menggunakan Sensor Ultrasonic Dan Metode Fuzzy

Eddy Dimiyan Enoko
 Fakultas Teknik Elektro
 Universitas Telkom
 Bandung, Indonesia
 edbydimiyan@student.telkomuniversity.ac.id

Basuki Rahmat
 Fakultas Teknik Elektro
 Telkom University
 Bandung, Indonesia
 basukir@telkomuniversity.ac.id

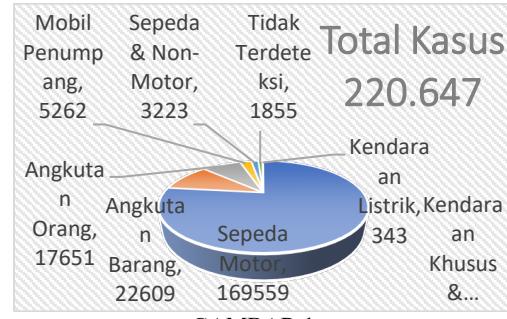
Erwin Susanto
 Fakultas Teknik Elektro
 Telkom University
 Bandung, Indonesia
 erwinelektro@telkomuniversity.ac.id

Abstrak - Keselamatan dalam berkendara merupakan aspek krusial, terutama saat menghadapi tantangan mendadak yang menyebabkan potensi kecelakaan. Penelitian ini bertujuan merancang dan mengimplementasikan sistem penggereman otomatis (*autobrake*) pada mobil *Remote Control* (RC) dengan memanfaatkan sensor ultrasonik dan logika fuzzy. Sistem ini dikembangkan untuk mendeteksi objek di depan kendaraan dan secara otomatis mengaktifkan penggereman berdasarkan kombinasi antara jarak dan kecepatan kendaraan. Komponen utama yang digunakan meliputi sensor HC-SR04 untuk pengukuran jarak serta rotary encoder sebagai pengukur kecepatan. Logika fuzzy diterapkan dengan dua variabel input jarak dan kecepatan dan satu output berupa tingkat penggereman, masing-masing menggunakan lima fungsi keanggotaan berbasis trapesium. Penelitian ini meliputi perancangan perangkat keras dan perangkat lunak, serta pengujian kinerja sistem terhadap berbagai skenario kecepatan dan jarak objek. Hasil pengujian menunjukkan bahwa sistem mampu merespons secara adaptif terhadap perubahan kondisi, dengan memberikan sinyal penggereman yang proporsional dan waktu yang tepat. Sistem ini berpotensi untuk dikembangkan lebih lanjut sebagai prototipe dasar dalam penerapan teknologi keselamatan aktif pada kendaraan pintar di masa depan.

Kata kunci: sensor ultrasonik, *autobrake*, fuzzy logic, kendaraan RC, Arduino

I. PENDAHULUAN

Angka kecelakaan lalu lintas di Indonesia menjadi isu serius yang memerlukan perhatian. Berdasarkan data Korps Lalu Lintas (Korlantas) Polri melalui Integrated Road Safety Management System (IRSMS), sepanjang Januari hingga Oktober 2024 tercatat 220.647 kasus kecelakaan lalu lintas, dengan 22.970 di antaranya mengakibatkan korban jiwa [1]. Kasus kecelakaan dapat di lihat pada Gambar 1 di bawah ini.



GAMBAR 1
 kasus Kecelakaan periode januari-Oktober

Data ini menunjukkan besarnya risiko yang dihadapi pengguna jalan. Salah satu tantangan utama bagi pengendara adalah potensi kecelakaan saat melaju dengan kecepatan tinggi, terutama di jalan yang padat. Hal ini dapat meningkatkan kemungkinan terjadinya kecelakaan jika pengemudi gagal merespons secara cepat dan mengambil keputusan yang sesuai dengan situasi.

Kemajuan teknologi kendaraan menuntut adanya peningkatan sistem keselamatan. Salah satu tantangan signifikan adalah ketidakmampuan pengemudi untuk merespons hambatan secara cepat, terutama pada kecepatan tinggi. Dalam konteks kendaraan kecil seperti mobil RC, hal ini menjadi representasi miniatur dari sistem keselamatan yang dapat diterapkan pada skala lebih besar. Penerapan sistem penggereman otomatis menjadi solusi potensial untuk merespons hambatan secara real-time.

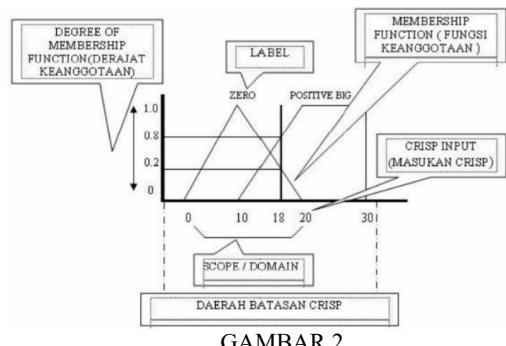
Penelitian ini memanfaatkan sensor ultrasonik sebagai detektor hambatan, serta logika fuzzy sebagai pengolah data untuk menghasilkan tingkat penggereman sesuai kondisi. Berbeda dari penelitian terdahulu yang menggunakan jumlah membership function terbatas, penelitian ini mengimplementasikan lima *membership function* untuk *input* kecepatan, jarak, dan *output* penggereman guna meningkatkan presisi dan adaptivitas sistem. Tujuan utama penelitian ini adalah membangun sistem *autobrake* yang

responsif, adaptif, dan dapat diimplementasikan pada kendaraan RC sebagai model awal.

II. KAJIAN TEORI

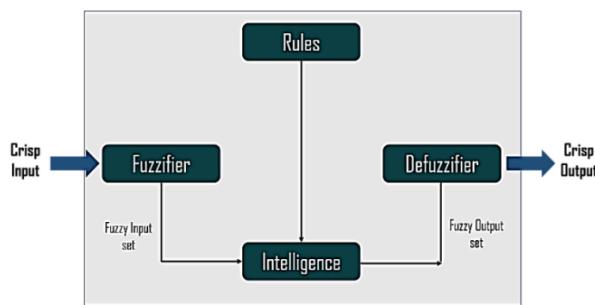
A. Teori Sistem Kontrol Fuzzy

Sistem kontrol fuzzy adalah metode pengendalian yang menggunakan logika fuzzy, yang merupakan teknik pemrosesan informasi yang mengelola data dengan memanfaatkan nilai derajat kepastian dalam rentang 0 hingga 1 [2]. Konsep logika fuzzy ini pertama kali diperkenalkan oleh Lotfi A. Zadeh pada tahun 1965, Lotfi Asker Zadeh adalah seorang ilmuwan Amerika Serikat berkebangsaan Iran dari Universitas California di Barkeley [3]. Logika fuzzy berasal dari konsep himpunan fuzzy, yang merupakan pengembangan teori himpunan klasik. Dalam sistem kontrol fuzzy, setiap variabel memiliki derajat kedalaman yang menunjukkan sejauh mana nilai tersebut termasuk dalam suatu kategori. Sistem ini banyak diterapkan pada bidang yang memerlukan respon otomatis terhadap variabel yang tidak pasti atau tidak linier. Konsep dasar logika fuzzy dapat di lihat pada Gambar 2 dibawah ini.



GAMBAR 2
Konsep Dasar Logika Fuzzy [4]

Dalam sistem kontrol fuzzy, terdapat 4 bagian utama seperti pada Gambar 3 di bawah ini:



GAMBAR 3
Arsitektur Logika fuzzy [5]

- **Aturan** : sistem ini bekerja berdasarkan aturan sederhana “*IF-THEN*” atau “*JIKA-MAKA*” yang menghubungkan kondisi tertentu dengan respon yang sesuai
- **Fuzzifikasi** : mengubah input crisp menjadi nilai fuzzy melalui fungsi keanggotaan.

- **Inference Engine** : Proses ini menyebabkan kesesuaian antara input fuzzy dan aturan, kemudian mengaktifkan aturan yang relevan untuk menghasilkan respons kontrol yang sesuai.

- **Defuzzification** : mengubah output fuzzy menjadi nilai crisp, misalnya menggunakan metode centroid

Logika fuzzy telah banyak digunakan dalam teori kontrol, teori keputusan, dan manajemen sains. Salah satu keunggulan logika fuzzy adalah kemampuannya melakukan penalaran berbasis bahasa, sehingga tidak memerlukan persamaan matematis kompleks untuk merancang objek yang akan dikendalikan. Salah satu contoh penerapan logika fuzzy dalam kehidupan sehari-hari adalah mesin cuci yang pertama kali dibuat di Jepang oleh Matsushita *Electric Industrial Company* pada tahun 1990 [6]. Dalam penelitian ini, logika fuzzy digunakan untuk mengatur intensitas penggereman mobil RC berdasarkan hambatan jarak. Hal ini memungkinkan mobil RC beroperasi dengan aman meskipun data jarak yang diterima tidak selalu konsisten.

B. Algoritma Pengontrolan Kendaraan

Algoritma kontrol berfungsi sebagai pusat pengambilan keputusan dalam sistem *autobreak* kendaraan otonom. Sistem ini memanfaatkan sensor untuk mendeteksi hambatan dan secara otomatis mengaktifkan penggereman guna mencegah atau mengurangi dampak tabrakan [7]. Pendekatan awal berdasarkan kontrol umpan balik seperti PID kemudian berkembang menggunakan logika fuzzy yang lebih adaptif terhadap kondisi dinamis.

Sistem *autobreak* kini banyak diterapkan sebagai fitur keselamatan aktif pada kendaraan modern, seperti AEB (*Automatic Emergency Braking*), yang mampu memperlambat kendaraan saat mendeteksi potensi tabrakan [8]. Dalam penelitian ini, algoritma *autobreak* diterapkan pada mobil RC dengan sensor ultrasonik untuk mendeteksi objek di depan dan mengatur penggereman secara otomatis.

III. METODE

A. Desain

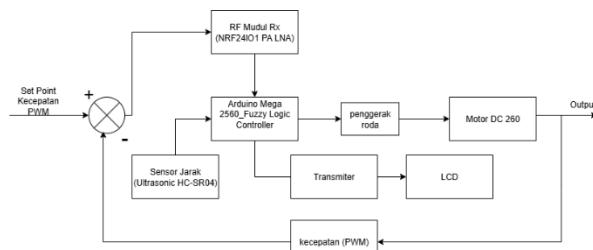
Sistem ini membahas sistem penggereman mobil otomatis. Terdapat 2 diagram blok yaitu sistem transmision dan sistem receiver.



GAMBAR 4
Diagram Blok Sistem Transmision

Pada Gambar 4 merupakan diagram blok yang menunjukkan alur kerja sistem transmision pada alat ini. *Button & potensio* merupakan masukan data untuk kontrol arah roda ke arah kiri/kanan dan maju,

kemudian Arduino Nano merupakan komponen utama pada sistem ini yang berfungsi sebagai mikrokontroler yang mengendalikan sistem, dan Modul NRF24L01 sebagai transmisi (pengirim) sinyal menuju receiver.



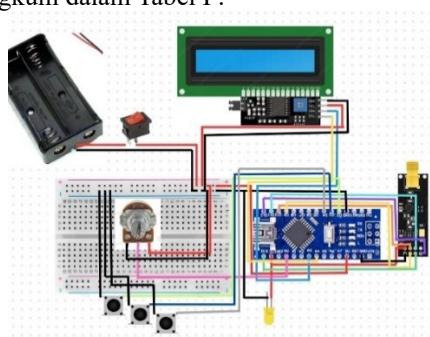
GAMBAR 5
Diagram Blok Sistem Receiver

Gambar 5 menunjukkan diagram blok prototipe *autobreak* pada mobil RC, yang menggambarkan alur kerja sistem penerima. Modul NRF24L01 menerima sinyal kendali dari pengontrol jarak jauh untuk menggerakkan roda depan melalui servo MG996R. Mikrokontroler Arduino Mega 2560 memproses perintah tersebut serta data dari sensor ultrasonik yang digunakan untuk mendeteksi jarak terhadap hambatan di depan kendaraan.

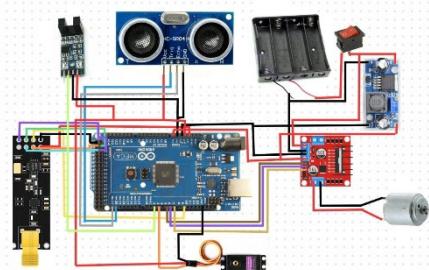
Data jarak kemudian diolah menggunakan logika fuzzy melalui tiga tahapan utama: fuzzifikasi, inferensi, dan defuzzifikasi. Hasil dari proses ini menentukan intensitas penggereman yang dikirim ke motor DC sebagai aktuator rem. Informasi jarak dan kecepatan kendaraan ditampilkan secara real-time melalui LCD, sehingga sistem mampu mengurangi kecepatan atau berhenti secara otomatis saat hambatan terdeteksi.

B. Desain Perangkat Keras

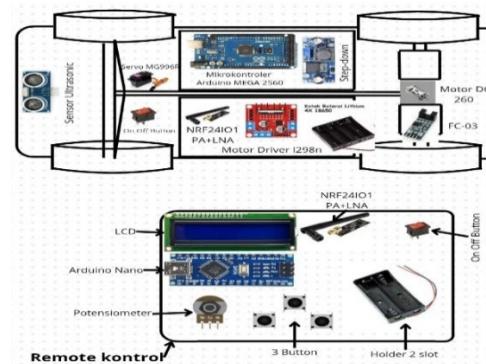
Perangkat yang digunakan dalam sistem ini adalah Arduino Nano, Modul NRF24L01 Transmisi (pengirim) dan Receiver (penerima), 3 Button, potensio, Arduino mega 2560, Motor servo MG996r, Motor DC, *step-down*, ultrasonic HC-SR04, dan LCD. Perangkat-perangkat ini akan dirangkai sehingga menjadi sebuah alat yang dapat digunakan untuk sistem penggereman otomatis. Daftar keseluruhan komponen yang terpilih dirangkum dalam Tabel I .



GAMBAR 6
Wiring Rangkaian sistem Transmitter



GAMBAR 7
Wiring Rangkaian Sistem Receiver



GAMBAR 8
Desain Alat

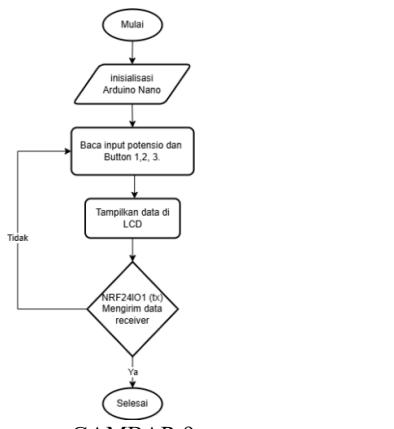
Gambar 6 dan Gambar 7 menunjukkan diagram rangkaian yang telah dirancang. Berbagai komponen dihubungkan sesuai pada skematik tersebut sehingga dapat berfungsi dengan baik dan berjalan secara keseluruhan. Desain alat dapat di lihat pada Gambar 8.

TABEL 1
Pemilihan Komponen

No	Unsur pembentuk sistem	Jenis komponen	Kompone n terpilih
1	Detektor	Modul Jarak	Ultrasonik HC-SR04
		NRF24L01 PA LNA	NRF24L01 PA LNA
2	Aktuator	Servo	Servo S0017M
		Serial Terminal	LCD 16 x 2
		Motor DC	Motor DC 260
3	Mikrokontrol er	Modul mikrokontrol er (Receiver)	Arduino Mega 2560
		Modul mikrokontrol er (Transmiser)	Arduino Nano

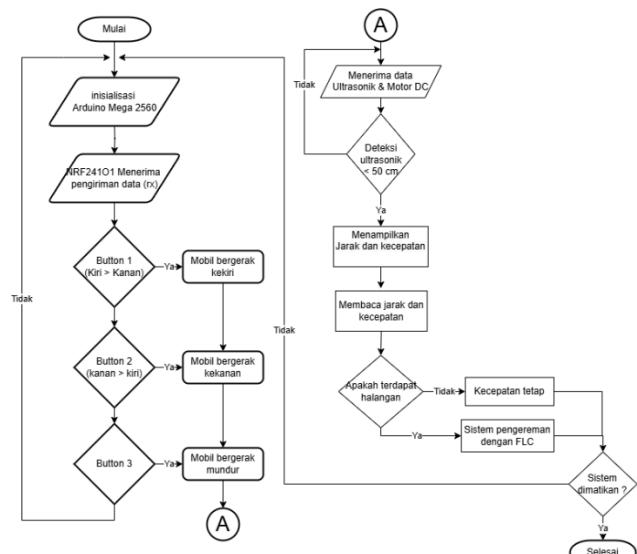
4	Komponen pendukung	Elektronik aktif	Motor Driver
	Elektronik aktif	6 buah baterai 18650 3.7V, 1200 mAh	
	Elektronik aktif	Stepdown 12 V	
	Elektronik pasif	potensio	
	Elektronik pasif	3 buah button	

C. Desain Perangkat Lunak



GAMBAR 9
Flowchart Sistem Transmiter

Pada Gambar 9 merupakan diagram alir dari sistem remot kontrol Mobil RC. dimulai dengan inisialisasi Arduino Nano untuk mempersiapkan komponen, seperti modul NRF24L01, *button*, dan potensio. Setelah sistem diinisialisasi, Arduino membaca data dari potensio, di mana potensio digunakan untuk mengontrol kecepatan dan button untuk arah kendaraan (kiri/kanan). Data yang telah dibaca kemudian dikirim ke modul NRF24L01 yang berfungsi sebagai *transmitter* untuk mengirimkan sinyal nirkabel ke *receiver* pada mobil RC. Proses ini berulang untuk memastikan komunikasi data berjalan terus menerus.



GAMBAR 10
Flowchart Sistem Receiver

Pada Gambar 10 proses dimulai dengan inisialisasi Arduino Mega 2560 sebagai pusat pemrosesan data pada sistem penerima. Modul NRF24L01 (rx) kemudian menerima data yang dikirimkan oleh *transmitter* berupa nilai *button* atau potensio. Arduino memproses data tersebut dengan membandingkan nilai *button*. Jika nilai kiri lebih besar, kendaraan akan belok kiri; jika nilai kanan lebih besar, kendaraan akan belok kanan; dan potensio untuk mengontrol kecepatan. Setelah itu, sistem dilanjutkan ke inisialisasi sensor ultrasonic dan motor DC. Jika jarak yang terdeteksi <50 cm, maka sistem akan menampilkan informasi jarak dan kecepatan pada LCD. Data tersebut kemudian diproses dan dianalisis lebih lanjut. Sistem akan memeriksa apakah hambatan berada pada jarak yang aman atau tidak. Jika tidak ada hambatan, kendaraan akan melaju dengan kecepatan normal. Namun, jika ada hambatan, sistem akan otomatis mengaktifkan fitur penggeraman dengan bantuan *fuzzy logic controller* (FCL). FCL ini bertugas mengatur tingkat penggeraman berdasarkan data jarak dan kecepatan kendaraan yang terdeteksi.

Setelah penggeraman selesai dilakukan, sistem akan kembali memenuhi kondisi. Jika sistem dimatikan secara manual, proses akan berhenti. Namun, jika sistem tetap menyala, siklus deteksi hambatan, analisis, dan penggeraman akan terus berulang. Diagram alir ini menggambarkan bagaimana sistem *autobreak* bekerja secara otomatis untuk meningkatkan keselamatan kendaraan saat menghadapi hambatan secara *real-time*.

IV. HASIL DAN PEMBAHASAN

Pada bagian ini akan dijabarkan hasil dan diskusi dari sistem perangkat yang telah dibuat. Pengujian tersebut dilakukan agar dapat mengetahui hasil dari setiap sistem dan perangkat yang sudah dirancang. Dengan tujuan sistem dan perangkat yang telah dibuat dapat berjalan sesuai rancangan

A. Pengujian Sensor Jarak

Kalibrasi sensor ultrasonik HC-SR04 bertujuan untuk memancarkan tingkat akurasi pembacaan jarak oleh sensor terhadap jarak sebenarnya. Pengujian dilakukan dengan mengukur hasil pembacaan sensor dengan pengukuran aktual menggunakan alat ukur roll meter. Data pada Tabel II, menunjukkan nilai error sebesar 0,15. Hal ini menjadikan pengujian tidak perlu dilakukan lagi, dikarenakan nilai error yang sudah cukup dapat ditoleransi ketika melakukan pengukuran HC-SR04.

TABEL 2
Pengambilan Data HC-SR04

No	Jarak Sesungguhnya (cm)	Waktu (μ s)	Jarak yang dibaca HC-SR04 (cm)	Error Rate (%)
1	5	331	5	0
2	10	637	10	0
3	15	943	16	6,66
4	20	1167	20	0
5	25	1435	24	4
6	30	1762	30	0
7	35	2021	34	2,85
8	40	2286	39	2,5
9	45	2678	45	0
10	50	2929	51	2
Rata-Rata Error $((JS-JYD)/JYD) \times 100\%$				0,15

B. Pengujian Motor Servo

Pengujian ini bertujuan untuk memastikan bahwa sudut yang diperintahkan pada motor servo benar-benar tercapai secara akurat. Berdasarkan data pada Tabel III, dari 10 kali pengujian, sebanyak 8 di antaranya menunjukkan kesesuaian penuh antara nilai yang diprogram dengan derajat aktual yang dihasilkan oleh motor servo.

TABEL 3
Pengambilan Data Servo

No	Derasat Pada Program	Derasat sesungguhnya	Hasil
1	0	0	Sesuai
2	6	6	Sesuai
3	12	12	Sesuai
4	19	20	Kurang Sesuai
5	24	24	Sesuai
6	30	30	Sesuai
7	37	38	Kurang Sesuai
8	42	42	Sesuai
9	45	45	Sesuai
10	54	54	Sesuai

C. Pengujian NRF24L01 PA+LNA

Pengujian kalibrasi NRF24L01 PA+LNA menunjukkan bahwa rata-rata delay mengalami peningkatan seiring bertambahnya jarak. Pada jarak 1 meter, rata-rata delay hanya sekitar 0,90 ms. Hal ini menunjukkan bahwa proses pengiriman dan penerima data berlangsung sangat cepat dan efisien begitu juga dengan jarak 2 meter, rata-rata dela hanyalah sekitar 1,12 ms. Ketika jarak mulai bertambah, delay memang mengalami sedikit peningkatan, namun tetap tergolong stabil dan masih dalam batas wajar. pada jarak 3-5 meter, transmisi mulai menurun ditandai dengan adanya packet loss. Secara keseluruhan dari 15 pengambilan data hanya 11 yang berhasil dan 4 gagal. Dengan demikian, modul NRF24L01 PA+LNA optimal digunakan pada jarak 1-2 meter, sementara pada jarak menengah hingga jauh komunikasi berkurang.

TABEL 4
Pengambilan Data NRF24L01 PA+LNA

No.	Jarak Meter	TX	RX	Delay (ms)	Status
1	1 M	1007.80	1008.60	0.85	Berhasil
2		1196.40	1197.20	0.90	Berhasil
3		1410.30	1411.10	0.95	Berhasil
4	2 M	1289.10	1290.00	1.05	Berhasil
5		1506.80	1507.70	1.10	Berhasil
6		1168.50	1169.40	1.20	Berhasil
7	3 M	1371.20	1372.60	1.40	Berhasil
8		1586.70	1588.10	1.55	Berhasil
9		1670.20	-	-	Gagal
10	4 M	1234.90	1236.40	1.85	Berhasil
11		1459.50	1461.00	1.95	Berhasil
12		1091.10	-	-	Gagal
13	5 M	1607.80	-	-	Gagal
14		1616.20	-	-	Gagal
15		1623.50	1625.80	2.45	Berhasil

D. Pengujian FC-03 RPM

Pada pengujian ini bertujuan untuk memastikan bahwa kecepatan sensor (seperti rotary encoder, dan tachometer) memberikan data yang akurat dan konsisten. Data Tabel V menunjukkan tingkat error pada sensor FC-03 menunjukkan variasi nilai, dengan error tertinggi sebesar 2,5% pada pengukuran ke-2 dan error terendah sebesar 1,7% pada beberapa pengukuran. Hal ini menunjukkan bahwa sensor FC-03 dapat memberikan hasil yang cukup akurat dan stabil.

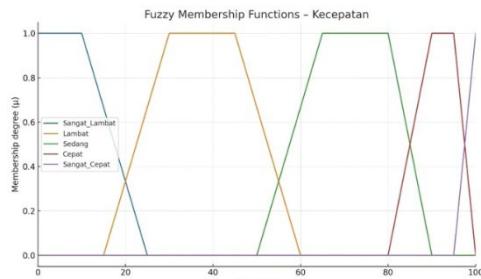
TABEL 5
Pengambilan Data RPM

No	PPR (pulsa Per Rotasi)	RPM dari Sensor	RPM dari tachometer	Selisih (%)
1	20	135	132	2.3
2	20	123	120	2.5
3	20	153	150	2.0
4	20	147	144	2.1
5	20	162	166	2.4

6	20	147	144	2.1
7	20	159	162	1.9
8	20	177	174	1.7
9	20	183	180	1.7
10	20	177	180	1.7

E. Simulasi Logika Fuzzy Menggunakan Matlab

Sistem kontrol logika fuzzy ini memiliki dua input, yaitu kecepatan dari *prototype* dan jarak dengan hambatan di depannya. Sementara itu, keluaran dari sistem ini adalah tingkat pengereman pada mobil RC. Setiap input dalam logika fuzzy memiliki fungsi keanggotaan (*membership function*) yang direpresentasikan dalam bentuk kurva trapesium. Dengan 5 nilai keanggotaan (*membership function*) pada masukan kecepatan seperti sangat lambat, lambat, sedang, cepat, sangat cepat pada masukan jarak berupa sangat dekat, dekat, sedang, jauh, sangat jauh dan 5 nilai keanggotaan (*membership function*) keluaran berupa Rem sangat kecil, kecil, sedang, besar, dan sangat besar. Pada kecepatan dapat dilihat pada Gambar 11, pada nilai keanggotaan jarak dapat dilihat pada Gambar 12, dan nilai keanggotaan pengereman dapat dilihat pada Gambar 13. Rule yang digunakan terdapat 25 rule yang dapat di lihat pada Gambar 14.

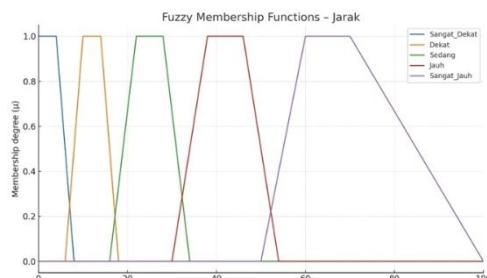


GAMBAR 11

Membership Function Kecepatan

Input 1 Kecepatan

- Sangat_Lambat = [0 0 10 25]
- Lambat = [15 30 45 60]
- Sedang = [50 65 80 90]
- Cepat = [80 90 95 100]
- Sangat_Cepat = [95 100 100 100]

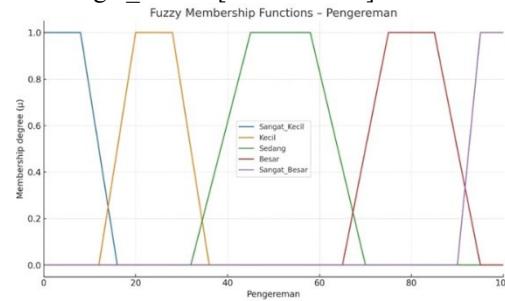


GAMBAR 12

Membership Function Jarak

Input 2 Jarak

- Sangat_Dekat = [0 0 4 8]
- Dekat = [6 10 14 18]
- Sedang = [16 22 28 34]
- Jauh = [30 38 46 54]
- Sangat_Jauh = [50 60 70 100]



GAMBAR 13

Membership Function Pengereman

Output Pengereman

- Sangat_Kecil = [0 0 8 16]
- Kecil = [12 20 28 36]
- Sedang = [32 45 58 70]
- Besar = [65 75 85 95]
- Sangat_Besar = [90 95 100 100]

```

1. If (Kecepatan is Sangat_Cepat) and (Jarak is Sangat_Dekat) then (Pengereman is Sangat_Besar) (1)
2. If (Kecepatan is Sangat_Cepat) and (Jarak is Dekat) then (Pengereman is Sangat_Besar) (1)
3. If (Kecepatan is Sangat_Cepat) and (Jarak is Sedang) then (Pengereman is Besar) (1)
4. If (Kecepatan is Sangat_Cepat) and (Jarak is Jauh) then (Pengereman is Sedang) (1)
5. If (Kecepatan is Sangat_Cepat) and (Jarak is Sangat_Jauh) then (Pengereman is Sangat_Besar) (1)
6. If (Kecepatan is Cepat) and (Jarak is Sangat_Dekat) then (Pengereman is Sangat_Besar) (1)
7. If (Kecepatan is Cepat) and (Jarak is Dekat) then (Pengereman is Besar) (1)
8. If (Kecepatan is Cepat) and (Jarak is Sedang) then (Pengereman is Sedang) (1)
9. If (Kecepatan is Cepat) and (Jarak is Jauh) then (Pengereman is Sangat_Besar) (1)
10. If (Kecepatan is Cepat) and (Jarak is Sangat_Jauh) then (Pengereman is Sangat_Besar) (1)
11. If (Kecepatan is Sedang) and (Jarak is Sangat_Dekat) then (Pengereman is Besar) (1)
12. If (Kecepatan is Sedang) and (Jarak is Dekat) then (Pengereman is Sedang) (1)
13. If (Kecepatan is Sedang) and (Jarak is Sedang) then (Pengereman is Sedang) (1)
14. If (Kecepatan is Sedang) and (Jarak is Jauh) then (Pengereman is Kecil) (1)
15. If (Kecepatan is Sedang) and (Jarak is Sangat_Jauh) then (Pengereman is Sangat_Kecil) (1)
16. If (Kecepatan is Lambat) and (Jarak is Sangat_Dekat) then (Pengereman is Sedang) (1)
17. If (Kecepatan is Lambat) and (Jarak is Dekat) then (Pengereman is Sedang) (1)
18. If (Kecepatan is Lambat) and (Jarak is Sedang) then (Pengereman is Kecil) (1)
19. If (Kecepatan is Lambat) and (Jarak is Jauh) then (Pengereman is Sangat_Kecil) (1)
20. If (Kecepatan is Lambat) and (Jarak is Sangat_Jauh) then (Pengereman is Sangat_Kecil) (1)
21. If (Kecepatan is Sangat_Lambat) and (Jarak is Sangat_Dekat) then (Pengereman is Kecil) (1)
22. If (Kecepatan is Sangat_Lambat) and (Jarak is Dekat) then (Pengereman is Kecil) (1)
23. If (Kecepatan is Sangat_Lambat) and (Jarak is Sedang) then (Pengereman is Sangat_Kecil) (1)
24. If (Kecepatan is Sangat_Lambat) and (Jarak is Jauh) then (Pengereman is Sangat_Kecil) (1)
25. If (Kecepatan is Sangat_Lambat) and (Jarak is Sangat_Jauh) then (Pengereman is Sangat_Kecil) (1)

```

GAMBAR 14

Rule Fuzzy Logic

TABEL 6
Rule Fuzzy Logic

n o	kecepatan	Jarak	Pengerema n
1	Sangat_Cepat	Sangat_Dek at	Sangat_Bes ar
2	Sangat_Cepat	Dekat	Sangat_Bes ar
3	Sangat_Cepat	Sedang	Besar
4	Sangat_Cepat	Jauh	Sedang
5	Sangat_Cepat	Sangat_Jau h	Sedang
6	Cepat	Sangat_Dek at	Sangat_Bes ar
7	Cepat	Dekat	Besar
8	Cepat	Sedang	Sedang
9	Cepat	Jauh	Kecil
10	Cepat	Sangat_Jau h	Sangat_Kec il
11	Sedang	Sangat_Dek at	Besar
12	Sedang	Dekat	Sedang

1 3	Sedang	Sedang	Sedang
1 4	Sedang	Jauh	Kecil
1 5	Sedang	Sangat_Jau h	Sangat_Kec il
1 6	Lambat	Sangat_Dek at	Sedang
1 7	Lambat	Dekat	Sedang
1 8	Lambat	Sedang	Kecil
1 9	Lambat	Jauh	Sangat_Kec il
2 0	Lambat	Sangat_Jau h	Sangat_Kec il
2 1	Sangat_Lamb at	Sangat_Dek at	Kecil
2 2	Sangat_Lamb at	Dekat	Kecil
2 3	Sangat_Lamb at	Sedang	Sangat_Kec il
2 4	Sangat_Lamb at	Jauh	Sangat_Kec il
2 5	Sangat_Lamb at	Sangat_Jau h	Sangat_Kec il

F. Integrasi sistem RX dan TX

Prototype pada Gambar 15 merupakan hasil dari integrasi sistem antara perangkat keras dan perangkat lunak, di mana sensor ultrasonik, *driver* motor L298N, modul komunikasi NRF24L01, dan mikrokontroler Arduino Mega 2560 bekerja bersama melalui algoritma logika fuzzy untuk mengatur kecepatan kendaraan secara otomatis.



GAMBAR 15
Integrasi Sistem RX

Pada program, proses penggereman kendaraan berbasis logika fuzzy dimulai dari penerimaan data kecepatan *set-point* (*dataReceived.pwm*) yang dikirim melalui modul NRF24L01 dan pembacaan jarak kendaraan (*distance*) menggunakan sensor ultrasonik yang diakses melalui library NewPing. Kedua nilai tersebut menjadi *input* fuzzy, yaitu *input* 1 (kecepatan) dan *input* 2 (jarak), yang didefinisikan

melalui fungsi *setupFuzzy()*. Di dalam fungsi ini, dibuat 5 *membership function* (FuzzySet) untuk masing-masing variabel *input* yaitu, *input* 1 (kecepatan) terdiri dari himpunan Sangat Lambat, Lambat, Sedang, Cepat, dan Sangat Cepat, sedangkan *input* 2 (jarak) terdiri dari Sangat Dekat, Dekat, Sedang, Jauh, dan Sangat Jauh, dan untuk *output* (pengereman) juga memiliki 5 *membership function* yaitu Sangat Kecil, Kecil, Sedang, Besar, dan Sangat Besar.

Hubungan antar-himpunan ini diatur dalam 25 aturan fuzzy (fuzzy rules) yang menggunakan metode *joinWithAND* sebagai operator logika. Contohnya, jika kecepatan cepat dan jarak dekat, maka *outputnya* adalah rem besar. Aturan ini mengatur tingkat penggereman sesuai kondisi kecepatan dan jarak aktual. Saat program berjalan di fungsi *loop()*, nilai kecepatan dari *remote* dikonversi ke skala 0–100 menggunakan (*map(dataReceived.pwm, 0, 254, 0, 100)*), lalu dimasukkan ke input fuzzy bersama dengan nilai jarak dari sensor (*fuzzy->setInput(...)*). Proses *fuzzify()* kemudian menghitung derajat keanggotaan tiap *input* terhadap fungsi keanggotaan yang sudah didefinisikan, dan *defuzzify(1)* menghasilkan nilai *crisp* penggereman (rem) dengan rentang 0–100 sesuai hasil inferensi fuzzy.

Nilai penggereman ini kemudian dikonversi menjadi rem ke skala PWM (*map(rem, 100, 0, 254, 0)*), sehingga semakin besar nilai penggereman, semakin besar pula pengurangan PWM motor. Nilai hasil pemetaan (*pwmRemap*) dikurangkan dari PWM awal yang diterima dari *remote* (*dataReceived.pwm*) untuk mendapatkan *pwmGabungan*. Nilai ini selanjutnya dibatasi (*constrain*) agar tidak di bawah 0 atau melebihi 254. Jika hasilnya ≤ 0 , fungsi *stopMotor()* dijalankan untuk menghentikan kendaraan sepenuhnya. Jika lebih besar dari 0, fungsi *forwardMotor(pwmGabungan)* dijalankan untuk menggerakkan motor maju dengan PWM yang sudah dikurangi sesuai tingkat penggereman fuzzy.

Dengan sistem ini, logika fuzzy tidak hanya menghitung berapa besar penggereman yang diperlukan, tetapi juga langsung mempengaruhi sinyal PWM yang masuk ke *driver* motor L298N. Hal ini memastikan bahwa semakin dekat jarak ke objek dan semakin tinggi kecepatan, maka sistem akan secara otomatis memberikan penggereman yang lebih besar, hingga menghentikan kendaraan jika jarak sudah sangat dekat, sehingga mencegah tabrakan. Proses ini menunjukkan bahwa integrasi antara perangkat keras, algoritma fuzzy logic, dan pengendalian motor berhasil membentuk sistem yang utuh, di mana setiap komponen saling mendukung untuk mencapai fungsi penggereman otomatis yang aman.

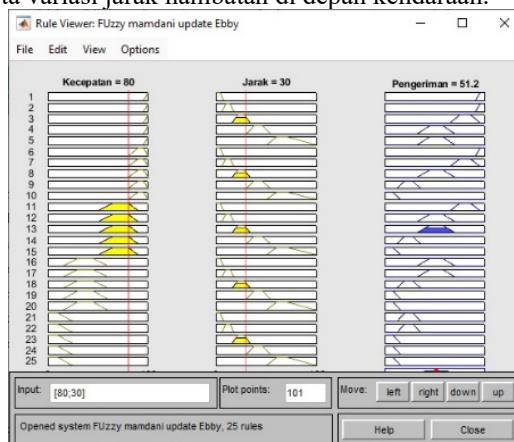


GAMBAR 16
Integrasi Sistem TX

Remote kontrol pada Gambar 16 ini berbasis Arduino Nano yang berfungsi sebagai pengirim sinyal kendali ke RX. TX dikendalikan oleh pengguna untuk mengatur arah dan kecepatan kendaraan. Arduino nano sebagai mikrokontroler, potensiometer berfungsi sebagai input manual dari pengguna untuk menentukan kecepatan yang diinginkan, NRF24l01 untuk mengirim data secara nirkabel ke RX, 3 button untuk mengendalikan arah ke kiri, kanan, dan mundur, LCD 16x2 untuk menampilkan status sistem atau nilai input.

G. Hasil Perbandingan Alat dan Matlab

Pengujian dilakukan untuk membandingkan hasil pengereman sistem fuzzy Mamdani yang diterapkan pada prototipe mobil RC dengan hasil keluaran dari sistem fuzzy pada MATLAB. Pengujian dilakukan sebanyak 20 kali dengan variasi kecepatan awal (dinyatakan dalam PWM), serta variasi jarak hambatan di depan kendaraan.



GAMBAR 17
Rule Viewer MATLAB

Berdasarkan hasil pengujian sistem logika fuzzy Mamdani yang ditampilkan melalui Rule Viewer MATLAB seperti pada Gambar 17, penulis memasukan input kecepatan sebesar 80 % dengan jarak 30 cm, sistem menghasilkan output pengereman sebesar 51,2. Hasil ini sesuai dengan tampilan pada LCD Arduino yang menunjukkan nilai PWM sebesar 203 (sekitar 80%), jarak 30 cm, dan output pengereman sebesar 51 seperti pada Gambar 18 di bawah ini,



GAMBAR 18
Tampilan Hasil Pengujian TX

Hasil menunjukkan kesesuaian antara simulasi MATLAB dan implementasi aktual di mikrokontroler. Hal ini membuktikan bahwa sistem fuzzy yang diimplementasikan pada Arduino berhasil mereplikasi logika fuzzy yang telah dirancang di MATLAB secara akurat. Selain itu, berdasarkan Tabel VI di bawah ini.

TABEL 7
Perbandingan Alat dan Matlab

No	Kecepatan Awal (PWM)	Kecepatan Hambatan (%) (Matlab)	Jarak Hambatan (cm)	Pengereman mobil	Pengereman Matlab	Selisih
1	254	100	10	96	96.4	0.4
2	254	100	15	96	96.1	0.1
3	229	90	20	51	51.2	0.2
4	229	90	25	51	51.2	0.2
5	203	80	30	51	51.2	0.2
6	203	80	35	24	24	0
7	178	70	40	24	24	0
8	178	70	45	24	24	0
9	152	60	50	24	24	0
10	152	60	55	6	6.79	0.79
11	127	50	60	6	6.49	0.49
12	127	50	65	6	6.49	0.49
13	102	40	70	5	5.96	0.96
14	102	40	75	6	6.22	0.22
15	76	30	80	6	6.49	0.49
16	76	30	85	6	6.79	0.79
17	51	20	90	7	7.08	0.08
18	51	20	95	7	7.38	0.38
19	25	10	100	50	50	0

2	25	10	100	50	50	0
0						
Rata-Rata Selisih (total selisih : 20)				0.2		
				9		

Pada Tabel VI menunjukkan hasil pengujian sistem, Pengujian dilakukan untuk membandingkan hasil sistem fuzzy pada mobil RC dengan simulasi MATLAB. Pengujian berhasil memberikan respon pengereman yang bervariasi sesuai dengan hambatan kedekatan. Semakin dekat jaraknya, maka output PWM yang dihasilkan semakin besar sehingga motor DC semakin lambat atau bahkan berhenti. Artinya, sistem fuzzy logic mampu mengatur tingkat pengereman secara adaptif dan responsif.

H. Hasil Pengujian

Pengujian sistem pengereman otomatis pada mobil RC dilakukan dengan dua variasi kondisi, yaitu Variasi 1 seperti pada Gambar 19 dengan jarak awal 80 cm dan hambatan tiba-tiba pada jarak 30 cm, serta Variasi 2 seperti pada Gambar 20 dengan jarak awal 120 cm dan hambatan tiba-tiba pada jarak 15 cm.



GAMBAR 19
Pengujian Alat variasi 1

TABEL 8
Perbandingan Alat dan Matlab

n o	P W M	Ja ra k A w al	obst ical dad aka n	Jarak Terba ca penge rema n	waktu penge rema n	Penge reman
1	60	80	30	27	38 ms	rem
2	65	80	30	28	37 ms	rem
3	70	80	30	25	38 ms	rem
4	75	80	30	19	44 ms	rem
5	80	80	30	19	47 ms	rem
6	85	80	30	18	45 ms	rem
7	90	80	30	18	44 ms	rem
8	95	80	30	16	42 ms	rem
9	10 0	80	30	10	42 ms	rem

1	10						
0	5	80	30	6	53 ms	tabrak	

Pada Variasi 1, sistem mampu mendeteksi hambatan dan melakukan pengereman dengan baik pada kecepatan rendah hingga sedang (PWM 60–105). Dengan posisi hambatan berada pada 30 cm dimana pengereman pada jarak 27–6 cm dengan waktu pengereman berkisar antara 37–47 ms. Namun, pada kecepatan tinggi (PWM 105), jarak terbaca hanya 6 cm sehingga mobil tidak sempat berhenti dan terjadi tabrakan. Hal ini menunjukkan bahwa semakin tinggi kecepatan, semakin kecil jarak deteksi sensor, dan semakin lama waktu yang dibutuhkan sistem untuk menghentikan kendaraan.



GAMBAR 20
Pengujian Alat variasi 2

TABEL 9
Perbandingan Alat dan Matlab

n o	P W M	Ja ra k A w al	obst ical dad aka n	Jarak Terba ca penge rema n	waktu penge rema n	Penge reman
1	60	12 0	15	24	47 ms	rem
2	65	12 0	15	27	44 ms	rem
3	70	12 0	15	24	45 ms	rem
4	75	12 0	15	21	42 ms	rem
5	80	12 0	15	15	47 ms	rem
6	85	12 0	15	14	44 ms	rem
7	90	12 0	15	10	42 ms	rem
8	95	12 0	15	8	45 ms	rem

9 0	10 0	12 0	15	5	44 ms	tabrak
1 0	10 5	12 0	15	4	43 ms	tabrak

Pada Variasi 2, meskipun jarak awal lebih jauh yaitu 120 cm, posisi rintangan yang lebih dekat (15 cm) membuat sistem penggereman bekerja lebih berat. Pada kecepatan rendah hingga sedang (PWM 60–105), mobil masih mampu berhenti dengan jarak terbaca 24–4 cm dan waktu penggereman relatif stabil pada kisaran 42–47 ms. Namun, pada PWM tinggi (100 dan 105), jarak terbaca sudah mencapai 4 cm sehingga mobil tidak sempat berhenti dan menabrak rintangan. Dari hasil ini dapat dilihat bahwa posisi rintangan memiliki pengaruh yang lebih besar dibandingkan jarak awal kendaraan terhadap penggereman.

Secara keseluruhan, waktu penggereman pada kisaran yang cukup konsisten yaitu 37–47 ms, yang menunjukkan bahwa respon logika fuzzy cukup stabil. Namun, keterbatasan terlihat pada kondisi kecepatan tinggi, di mana sistem tidak mampu menghentikan kendaraan tepat waktu. Hal ini membuktikan bahwa sistem fuzzy logic dapat bekerja untuk kecepatan rendah hingga sedang, sementara untuk kecepatan tinggi diperlukan optimasi lebih lanjut, baik pada sensitivitas sensor ultrasonik, penyesuaian aturan fuzzy, maupun penentuan batas aman kecepatan kendaraan agar sistem penggereman otomatis dapat berfungsi maksimal tanpa menimbulkan tabrakan.

V. KESIMPULAN

Berdasarkan hasil perancangan, implementasi, dan pengujian prototipe sistem rem otomatis pada mobil RC yang menggunakan sensor ultrasonik dan metode logika fuzzy, dapat disimpulkan bahwa sistem ini mampu mendeteksi hambatan di depan kendaraan secara real-time dengan menggunakan sensor HC-SR04, yang memiliki rata-rata error rendah sebesar 0,15%. Metode logika fuzzy yang digunakan terdiri dari dua input, yaitu jarak dan kecepatan, serta satu output berupa tingkat penggereman. Masing-masing input dan output memiliki lima fungsi memori yang terbukti mampu memberikan respon penggereman terhadap kondisi jalan. Hasil pengujian menunjukkan bahwa sistem bekerja dalam mengatur kecepatan kendaraan untuk menghindari tabrakan, berdasarkan nilai jarak dan kecepatan aktual. Dengan demikian, sistem yang dirancang telah berhasil memenuhi tujuan penelitian

dalam meningkatkan keselamatan operasional kendaraan RC, khususnya dalam menghadapi skenario rintangan yang muncul secara tiba-tiba.

REFERENSI

- [1] S. Aulia, “Sepanjang 2024 Angka Kecelakaan Lalu Lintas di Indonesia Tembus 220.647 Kasus,” otomotif.kompas.com. Accessed: Jan. 14, 2025. [Online]. Available: <https://otomotif.kompas.com/read/2024/11/22/171200115/sepanjang-2024-angka-kecelakaan-lalu-lintas-di-indonesia-tembus-220.647>
- [2] N. L. Mauliddiyah, “Prototype Sistem Penggereman Otomatis Pada Mobil RC Dengan Sistem Kendali Fuzzy Logic Berbasis Arduino MEGA 2560,” vol. 2560, p. 6, 2021.
- [3] R. Adolph, *LOGIKA FUZZY TAHANI by INDAH WAHYUNI* 48. 2016.
- [4] Junaidi, “Implementasi Fuzzy Logic Dengan Metode Mamdani Untuk Sistem Pendukung Keputusan Kinerja Dosen,” *J. Inf. Syst.*, vol. 3, no. 1, pp. 17–27, 2023, doi: 10.61488/jis.v3i1.256.
- [5] S. R. Yahya *et al.*, *Metode Fuzzy Logic*, no. May. 2023.
- [6] A. Saelan, “Logika Fuzzy,” Makalah If2091 Struktur Diskrit Tahun 2009. Accessed: Oct. 27, 2024. [Online]. Available: <https://raharja.ac.id/2020/04/06/logika-fuzzy/#:~:text=Logika fuzzy dapat digunakan dalam matematik dari objek yang dikendalikan>.
- [7] DI, “Kontroller Kendaraan Otonom: Teknologi dan Algoritma Kontrol untuk mencakup Mobilitas Mandiri,” Kmtech.Id. Accessed: Oct. 27, 2024. [Online]. Available: <https://www.kmtech.id/post/kontroller-kendaraan-otonom-teknologi-dan-algoritma-kontrol-untuk-mencapai-mobilitas-mandiri>
- [8] Cycy, “Revolusi Industri Otomotif Peran Kecerdasan Buatan dalam Mobil Masa Depan,” Wartamu.Id. Accessed: Oct. 27, 2024. [Online]. Available: <https://www.wartamu.id/revolusi-industri-otomotif-peran-kecerdasan-buatan-dalam-mobil-masa-depan/>