

Perancangan Sistem Kontrol Tendangan Berdasarkan Posisi Robot Pada *Four Wheeled Omnidirectional Mobile Robot* Berbasis *Fuzzy Logic Controller*

Muhammad Muslich Fuadi

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya
e-mail: muhammadmuslich.20027@mhs.unesa.ac.id

Muhammad Syariffuddin Zuhrie, Endryansyah, Lilik Anifah

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya
e-mail: zuhrie@unesa.ac.id, endryansyah@unesa.ac.id, lilikanifah@unesa.ac.id

Abstrak

Robot merupakan alat yang diciptakan untuk membantu menyelesaikan kebutuhan manusia secara otomatis. Untuk memacu perkembangan robot di Indonesia, setiap tahunnya diadakan kompetisi robot dengan tema yang bervariasi. Salah satunya adalah Kontes Robot Sepak Bola Indonesia kategori beroda. Selain sistem penentuan posisi, robot sepak bola juga memerlukan sistem pengendalian tendangan. Pada penelitian kali ini mekanisme penendang menggunakan solenoid untuk pendorong *kicker* yang dilengkapi dengan sistem *linier rail set* untuk sistem naik turun *kicker*, agar nantinya pola gaya tendangan lebih bervariasi. Pengendalian tendangan pada penelitian ini menggunakan metode *Fuzzy Logic Controller* dengan nilai masukan nilai jarak robot dengan gawang dan nilai arah hadap robot terhadap gawang untuk nilai keluaran merupakan nilai *delay* dari lama *trigger driver* solenoid dan nilai sudut dari titik sentuhan *kicker* dengan bola. Sistem penentuan koordinat posisi robot menggunakan sistem *gyrodometry* yang merupakan kombinasi dari *gyroscope* dan *rotary encoder*. Sistem penendang pada penelitian ini dapat bekerja dengan baik, serta sistem dari *fuzzy logic* sendiri mampu mengontrol nilai keluaran berdasarkan nilai masukan. Hasil dari pengujian merupakan pola dari tendangan yang dihasilkan berdasarkan besar nilai *delay* dan nilai sudut. Pada percobaan pertama sistem kontrol tendangan bekerja dengan baik, dengan nilai RMSE jarak 37,75 cm dan nilai RMSE *yaw* 0,81°, serta nilai RMSE jarak 37,46 cm dan nilai RMSE *yaw* 0,88° untuk percobaan kedua, untuk percobaan ketiga didapatkan nilai RMSE jarak 40,27 cm dan nilai RMSE *yaw* 1,00°.

Kata kunci: Tendangan, *gyrodometry*, Robot, *Fuzzy Logic*

Abstract

Robots are tools created to help solve human needs automatically. To spur the development of robots in Indonesia, every year robot competitions are held with varying themes. One of them is the Indonesian Football Robot Contest in the wheeled category. Apart from a positioning system, soccer robots also need a kick control system. In this research, the kicking mechanism uses a solenoid to push the kicker which is equipped with a linear rail set system for the kicker up and down system, so that the kick force pattern will be more varied. Kick control in this study uses the Fuzzy Logic Controller method with input values for the distance between the robot and the goal and the value for the direction the robot is facing towards the goal. The output value is the delay value of the solenoid driver trigger length and the angle value of the touch point of the kicker with the ball. The robot position coordinate determination system uses a *gyrodometry* system which is a combination of a *gyroscope* and a *rotary encoder*. The kicking system in this research can work well, and the fuzzy logic system itself is able to control the output value based on the input value. The results of testing the kick control system based on the robot's position are the resulting kick patterns based on the delay value and angle value. In the first experiment the kick control system worked well, with a distance RMSE value of 37.75 cm and a yaw RMSE value of 0.81°, as well as a distance RMSE value of 37.46 cm and a yaw RMSE value of 0.88° for the second experiment, for the third experiment the distance RMSE value was obtained. 40.27 cm and an RMSE yaw value of 1.00°.

Keywords: Kick, *gyrodometry*, Robot, Fuzzy Logic

PENDAHULUAN

Robot merupakan sebuah alat yang diciptakan guna membantu menyelesaikan kebutuhan manusia secara

otomatis dengan program yang telah ditanamkan pada sistem yang telah dibuat atau dengan menggunakan kontrol secara manual yang dilakukan oleh manusia. Setiap tahun, Indonesia menggelar kompetisi robot untuk

mendorong kemajuan dalam bidang robotika, dengan tema yang beragam. Salah satunya adalah Kontes Robot Sepak Bola Indonesia dengan kategori beroda. (Rachman. 2017).

Sistem kerja dari robot sepak bola beroda yaitu robot harus bisa mencetak goal ke gawang lawan atau bisa mengoper bola ke robot rekan satu timnya. Oleh karena itu robot dirancang agar bisa bergerak ke segala arah. Dengan penggunaan *Four Wheeled Omnidirectional* (Denegri, dkk. 2019), robot dapat melakukan pergerakan hingga lebih dari 8 arah (kanan, kiri, maju, mundur, serong kanan atas, serong kanan bawah, serong kiri atas, serong kiri bawah) tanpa perlu menyesuaikan arah hadapnya (Nurariyanto. 2022).

Pada robot sepak bola beroda diperlukan sistem penentuan posisi robot agar robot tidak mengalami kesulitan dalam menentukan tindakan selanjutnya (Holis. 2023). Penentuan posisi robot bisa diketahui dari pengolahan data pada sensor-sensor yang dimiliki. Metode penentuan posisi robot yang umum digunakan pada robot beroda adalah dengan menggunakan *rotary encoder* yang dilengkapi dengan sensor kompas digital untuk menentukan arah hadap robot sehingga diperoleh kecepatan putaran setiap roda. Nilai kecepatan dan arah ini diproses untuk mendapatkan posisi robot (Rachmawan. 2017).

Pengendalian tendangan pada robot sepak bola diperlukan agar tendangan yang dihasilkan dari dorongan solenoid kepada *kicker* lebih bervariasi. Dalam proses pengendalian tendangan ini digabungkan dengan *Fuzzy Logic Controller* untuk menciptakan tendangan yang bervariasi. Proses *Fuzzy Logic* nantinya akan berdasarkan posisi robot dengan gawang sehingga didapatkan pola gaya tendangan dari solenoid dan sudut perkenaan *kicker* terhadap bola dari hasil pengolahan *Fuzzy Logic Controller* (Cahyono, dkk. 2020).

Cara kerja *Fuzzy Logic Controller* tidak memperhatikan struktur internal plant, hanya mengamati kesalahan sebagai selisih antara setpoint dan keluaran sistem serta mengubah pengaturan panel kontrol untuk meminimalkan kesalahan (Ananthababu, dkk. 2017). *Fuzzy Logic Controller* terdiri dari unit *fuzzifikasi*, basis pengetahuan *fuzzy*, mesin keputusan *fuzzy*, dan unit *defuzzifikasi* (Khairullah, dkk. 2020). *Fuzzy Logic Controller* dapat mengontrol gerak robot beroda dan secara otomatis mengurangi kesalahan geraknya (Al Azka. 2021). Dengan adanya latar belakang diatas didapatkan perlunya kontrol penendang guna menunjang performa dari robot beroda, dengan *Fuzzy Logic* sebagai kontrolnya yang diharapkan dapat memberikan respon yang baik.

Tujuan dari penelitian ini adalah (1) Mengetahui cara merancang sistem kontrol tendangan pada *four wheeled*

omnidirectional mobile robot berdasarkan posisi robot. (2) Mengaplikasikan metode kontrol *Fuzzy Logic Controller* untuk kontrol tendangan berdasarkan posisi robot pada *four wheeled omnidirectional mobile robot*. (3) Mengetahui hasil uji respon sistem kontrol tendangan berdasarkan posisi robot pada *four wheeled omnidirectional mobile robot*.

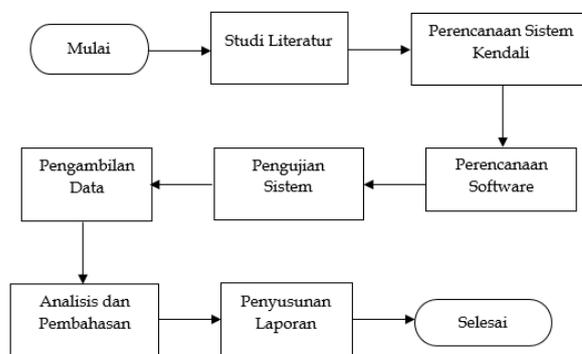
METODE

Pendekatan Penelitian

Penelitian kali ini menggunakan metode penelitian kuantitatif. Dalam buku yang ditulis oleh Kasiram (2008:149) mengenai Metodologi Penelitian Kualitatif dan Kuantitatif, mengungkapkan bahwa penelitian kuantitatif adalah suatu proses menemukan pengetahuan data berupa nomor yang dimanfaatkan dalam menganalisis pengetahuan mengenai apa yang ingin diketahui (Nurariyanto. 2022). Penulis menggunakan *software STM32CubeIDE* yang digunakan untuk memprogram *hardware* robot guna memenuhi tujuan penelitian.

Rancangan Penelitian

Diagram blok tahapan yang akan dilakukan dalam penelitian kali ini. Tahapan pertama peneliti mencari literatur dan refrensi dari penelitian yang ada hubungannya dengan penelitian kali ini yang ditunjukkan pada gambar 1.



Gambar 1. Diagram Blok Tahapan Penelitian

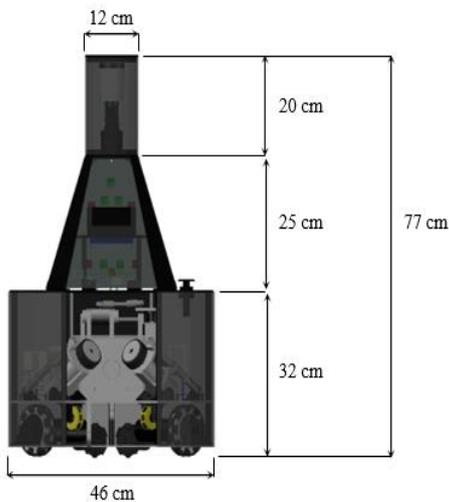
Literatur dan refrensi didapatkan dari buku, jurnal, website, serta karya orang lain. Setelah menemukan beberapa literatur diperoleh suatu pemodelan sistem dari *hardware fourwheeled omnidirectional mobile robot*. Pada tahap perancangan sistem kendali, *Fuzzy Logic Controller* digunakan sebagai metode kontrol hasil tendangan dari *fourwheeled omnidirectional mobile robot*. Kemudian dilanjutkan perancangan software, pada tahap ini dilakukan pemrograman pada mikrokontroler *STM32F4 Discovery*. Setelah itu dilakukan 3 pengujian sistem. Pengujian yang dilakukan yaitu sistem tendangan, sistem *gyrodometry*, dan sistem dari *Fuzzy Logic Controller*. Setelah pengujian dari setiap sistem

Perancangan Sistem Kontrol Tendangan Berdasarkan Posisi Robot Pada *Four Wheeled Omnidirectional Mobile Robot* Berbasis *Fuzzy Logic Controller*

akan dilakukan pengujian dari sistem keseluruhan. Dari pengujian sistem yang dilakukan akan didapatkan data untuk dilakukan analisis dan pembahasan. Kemudian dilanjutkan pada tahap yang terakhir yaitu tahap penyusunan laporan dari penelitian yang telah dilakukan.

Omnidirectional Mobile Robot

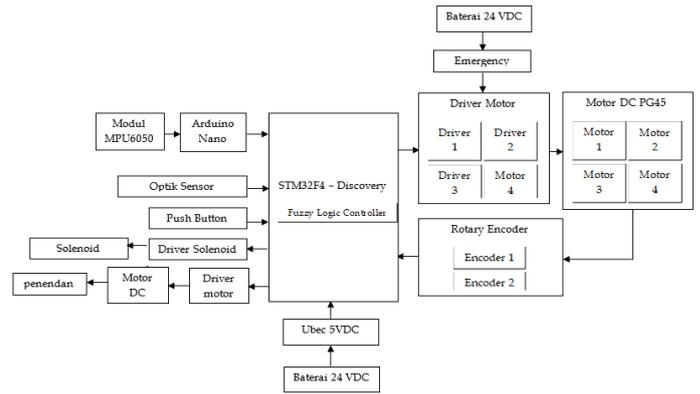
Penelitian ini memiliki struktur mekanik robot dengan dimensi 46 cm x 46 cm untuk proyeksi kelantai dan 77 cm untuk tinggi dari robot tersebut. Untuk bahan dasar pembuatan mekanik robot menggunakan aluminium serta 4 motor DC PG45 sebagai penggerak dari robot dan roda yang digunakan roda jenis *omniwheel*.



Gambar 2. Desain Robot

Robot terdiri dari tiga layer dasar yang terbuat dari plat aluminium 5 mm. Layer pertama terdapat motor penggerak, *Rotary Encoder* dan sensor optik. Pada layer kedua terdapat baterai, motor *handling*, roda bebas *handling*, penendang, solenoid, *power management robot*, driver motor, driver solenoid, *boost converter*, dan kapasitor. Layer paling atas layer ketiga terdapat mikrokontroler, mini pc, dan terdapat struktur penyangga dari kamera *omnivision* yang dilapisi oleh tabung akrilik. Pada struktur penyangga juga terdapat *push button* dan saklar serta LCD untuk menampilkan data dari robot.

Robot memiliki dua *controller* yaitu Arduino Nano menjadi slave, serta STM32F4-discovery menjadi master, *controller* di *supply* oleh baterai 24 VDC yang sebelumnya di *step down* menjadi 5 VDC menggunakan Ubec. Arduino Nano digunakan untuk pengolahan data *gyroscope* dari modul MPU6050, nantinya data *gyroscope* tersebut akan dikirimkan ke STM32F4-discovery melalui komunikasi serial. Data yang dikirimkan dari Arduino Nano akan diolah STM32F4-discovery untuk mendapatkan nilai sudut putar atau *yaw* robot.



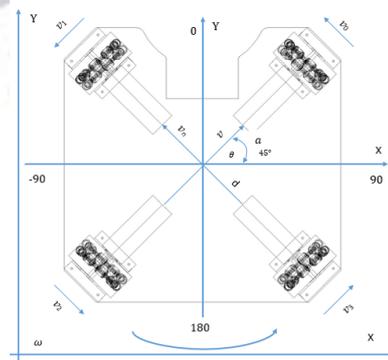
Gambar 3. Diagram Blok Harware Robot

Selain dari rangkaian slave STM32F4-discovery juga mendapatkan data dari *Rotary Encoder*, *Encoder* motor DC, sensor optik, dan *push button*. Dari data yang diterima ini nantinya digunakan STM32F4-discovery sebagai parameter untuk menggerakkan motor penggerak melalui driver motor sehingga robot dapat bergerak sesuai dengan set posisi yang diinginkan.

Untuk melakukan tendangan STM32F4-discovery memberikan perintah kepada driver solenoid agar tegangan yang ada pada kapasitor dapat diteruskan ke solenoid. Kontrol perkenaan penendang dengan bola menggunakan sistem *Linier rail set* yang digerakkan oleh motor DC PG45.

Gyrodometry

Gyrodometry merupakan penggabungan dari nilai *gyroscope* dan nilai *odometry* yang digunakan untuk mencari nilai koordinat posisi dalam sistem *positioning*. Sensor *gyroscope* merupakan sensor yang memanfaatkan momentum sudut, secara mekanis *gyroscope* berbentuk seperti roda atau cakram yang berputar dimana poros dari roda atau cakram tersebut bebas untuk mengambil orientasinya.



Gambar 4. Penempatan Roda Omni

Odometry merupakan salah satu metode yang didasari kinematika robot. Kinematika robot dibutuhkan guna mempelajari gerak robot berdasarkan struktur geometri dari kerangka robot dengan mengabaikan gaya,

torsi, dan momen yang menyebabkan robot bergerak (Nurariyanto. 2022). Kinematika dari *omnidirectional mobile robot* dapat dihitung dari susunan mekanik roda yang telah dirancang seperti pada Gambar 4.

Berdasarkan Gambar 4 dapat dibuat persamaan 1, 2, dan 3 :

$$(x, y, \theta) = (v_x(t) = \frac{dx(t)}{dt}, v_y(t) = \frac{dy(t)}{dt}, \omega(t) = \frac{d\theta(t)}{dt}) \quad (1)$$

$$\begin{bmatrix} v_x(t) \\ v_y(t) \\ \omega(t) \end{bmatrix} = \begin{bmatrix} \cos(a(t)) & \sin(a(t)) & 0 \\ -\sin(a(t)) & \cos(a(t)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} v(t) \\ v_n(t) \\ \omega(t) \end{bmatrix} \quad (2)$$

$$\begin{aligned} v_x(t) &= v(t) \cdot \cos(a(t)) + v_n(t) \cdot \sin(a(t)) \\ v_y(t) &= v(t) \cdot -\sin(a(t)) + v_n(t) \cdot \cos(a(t)) \\ \omega(t) &= \omega(t) \end{aligned} \quad (3)$$

dimana,

(x, y, θ) = posisi lokal robot
 $(v_x(t), v_y(t), \omega(t))$ = kecepatan linier sumbu statis
 $(v(t), v_n(t), \omega(t))$ = kecepatan linier sumbu robot

Pada persamaan (1) posisi lokal robot didapatkan dari integral kecepatan linier pada sumbu statis, kecepatan sumbu statis sendiri didapatkan dari kecepatan linier pada sumbu robot yang dikonversi dengan rumus trigonometri seperti pada persamaan (2). Nilai a merupakan nilai sudut antara posisi motor terhadap sumbu X, dan untuk nilai putaran atau arah hadap robot (θ) memiliki rentang antara -179,99 derajat sampai 180 derajat.

Dari Gambar 4 dapat dibuat persamaan 4 *invers kinematic*.

$$\begin{bmatrix} v_0 \\ v_1 \\ v_2 \\ v_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & d \\ -1 & 0 & d \\ 0 & -1 & d \\ 1 & 0 & d \end{bmatrix} \begin{bmatrix} v(t) \\ v_n(t) \\ \omega(t) \end{bmatrix} \quad (4)$$

$$\begin{aligned} v(t) &= \frac{1}{2} (v_3 - v_1) \\ v_n(t) &= \frac{1}{2} (v_0 - v_2) \\ \omega(t) &= \frac{(v_0 + v_1 + v_2 + v_3)}{(4d)} \end{aligned} \quad (5)$$

dimana,

$(v_0 + v_1 + v_2 + v_3)$ = kecepatan roda
 d = jarak roda dengan titik pusat

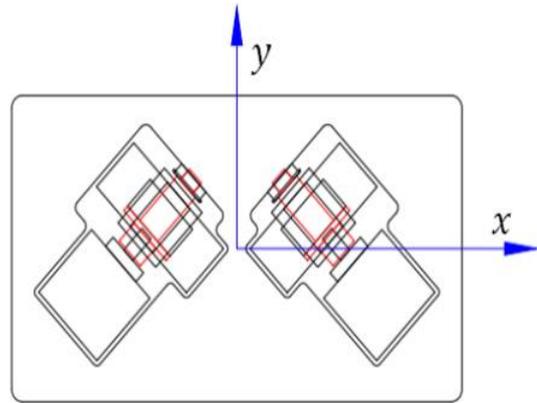
Dari persamaan 5 posisi robot masih posisi lokal sehingga perlu dikonversi dari posisi lokal menjadi posisi global.

$$\begin{bmatrix} x_g \\ y_g \\ \omega \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_l \\ y_l \\ \theta \end{bmatrix} \quad (6)$$

Dimana,

(x_g, y_g, ω) = posisi global robot
 (x_l, y_l, θ) = posisi lokal robot

Pada persamaan (6) titik referensi (0,0) yang awalnya ditengah sumbu robot saat awal mulai, dikonversikan keposisi awal robot bekerja akan tetapi untuk arah hadap robot tetap baik lokal maupun global (Fikri. 2021). *Odometry* memiliki kelemahan sistematis dan non-sistematis. Kesalahan sistematis biasanya didapatkan karena adanya kesalahan kinematika robot seperti adanya perbedaan ukuran roda dan jarak antar roda. Sedangkan kesalahan non-sistematis yang sering terjadi biasanya slip dari roda serta bidang dasar yang tidak rata. Pemasangan *Rotary Encoder* ditujukan untuk meminimalisir besarnya error non-sistematis yang mungkin terjadi sehingga dalam pemasangannya seperti Gambar 5.



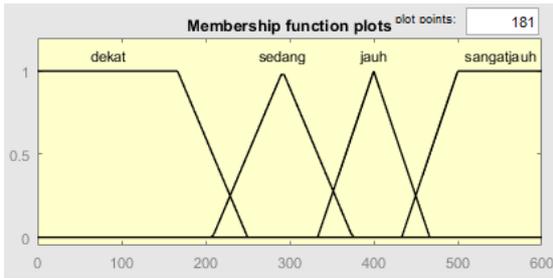
Gambar 5. Posisi Rotary Encoder (sumber; Al Azka. 2021)

Fuzzy Logic Controller

Berbeda dengan logika Boolean yang hanya memiliki nilai 0 dan 1, logika *fuzzy* memiliki derajat keanggotaan yang nilainya diantara 0 dan 1 (Firmansyah. 2023). himpunan. *Fuzzy Logic Controller* menggunakan dua parameter sebagai masukan, yaitu jarak posisi robot terhadap gawang yang didapatkan dari pembacaan sistem *odometry* dan besar nilai *yaw* atau arah hadap robot terhadap gawang yang didapatkan dari pembacaan *gyroscope*. Untuk keluaran berupa besar tegangan untuk *supply* solenoid dan besar sudut perkenaan *kicker* terhadap bola, nantinya model tendangan yang dihasilkan kearah gawang berdasarkan aturan *fuzzy* yang telah dibuat.

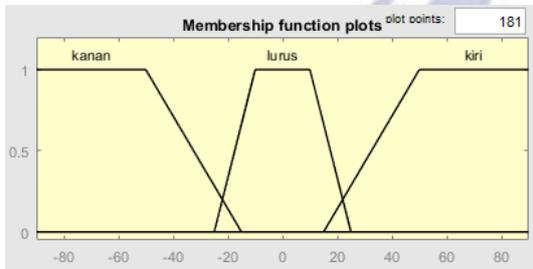
Pada penelitian kali ini memiliki dua variabel masukan yaitu jarak robot dan nilai *yaw*, serta memiliki dua variabel keluaran yaitu delay dan sudut.

Perancangan Sistem Kontrol Tendangan Berdasarkan Posisi Robot Pada *Four Wheeled Omnidirectional Mobile Robot* Berbasis *Fuzzy Logic Controller*



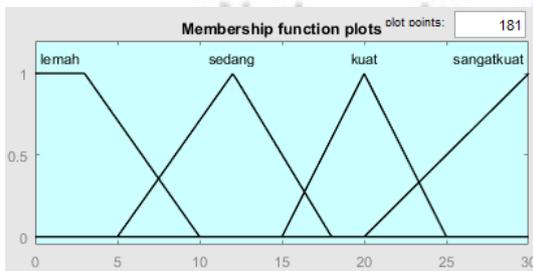
Gambar 6. *Membership Function* Variabel Jarak Robot

Nilai variabel jarak robot didasarkan pada ukuran lapangan Kontes Robot Indonesia kategori beroda yaitu berukuran 8 x 12 meter, tetapi pada penelitian kali ini hanya menggunakan setengah dari lapangan yaitu 8 x 6 meter. Pada variabel jarak menjadi empat kelas seperti yang terlihat pada Gambar 6.



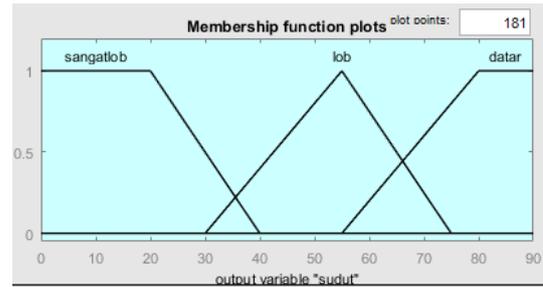
Gambar 7. *Membership Function* Variabel *Yaw*

Nilai variabel arah hadap robot atau nilai *yaw* didapatkan dari pembacaan *gyroscope* terhadap gawang, nilai *yaw* bernilai 0 saat robot dititik awal lapangan lurus menghadap gawang. Nilai *yaw* menjadi negatif apabila robot berotasi kanan dan bernilai positif apabila robot berotasi ke kiri. Pada variabel nilai *yaw* dibagi menjadi tiga kelas, untuk lurus apabila nilai *yaw* berada diantara -25 hingga 20, serta untuk kanan nilai *yaw* berada diantara -15 hingga -90 dan kiri apabila nilai *yaw* diantara 15 hingga 90.



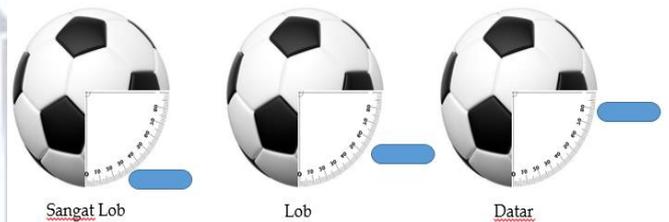
Gambar 8. *Membership Function* Delay

Nilai variabel delay merupakan nilai yang digunakan untuk mengatur besar tegangan dari kapasitor yang dialirkan ke solenoid melalui driver solenoid.



Gambar 9. *Membership Function* Sudut

Nilai variabel sudut didapatkan dari perkenaan *kicker* dengan bola, yang di ilustrasikan seperti pada Gambar 10 tinggi rendah *kicker* diatur menggunakan motor DC dengan sistem *linier rail set*.



Gambar 10. Ilustrasi Perkenaan *Kicker* dengan Bola

Setelah itu dibuat aturan *Fuzzy* untuk menentukan hasil yang dieksekusi nantinya, eksekusi yang ada berupa *delay* yang merupakan lama *trigger* tegangan untuk gaya tendangan yang dihasilkan solenoid dan sudut yang merupakan perkenaan *kicker* dengan bola pada tabel 1 dan 2.

Tabel 1. Aturan *Fuzzy* dari Delay

| Jarak <i>yaw</i> | Dekat | Sedang | Jauh | Sangat jauh |
|---------------------|--------|--------|------|-------------|
| Lurus | Lemah | Sedang | Kuat | Sangat Kuat |
| Kanan | Sedang | Kuat | Kuat | Sangat Kuat |
| Kiri | Sedang | Kuat | Kuat | Sangat Kuat |

Tabel 2. Aturan *Fuzzy* dari Sudut

| Jarak <i>yaw</i> | Dekat | Sedang | Jauh | Sangat jauh |
|---------------------|-------|--------|------|-------------|
| Lurus | Datar | Datar | Lob | Sangat Lob |
| Kanan | Datar | Lob | Lob | Sangat Lob |
| Kiri | Datar | Lob | Lob | Sangat Lob |

Berdasarkan aturan *fuzzy* yang ada dapat dibuat *inference engine* berdasarkan aturan IF-THEN yang sesuai dengan kebutuhan.

1. If (Jarak is dekat) and (yaw is lurus) then (delay is lemah) (1)
2. If (Jarak is sedang) and (yaw is lurus) then (delay is sedang) (1)
3. If (Jarak is jauh) and (yaw is lurus) then (delay is kuat) (1)
4. If (Jarak is sangatjauh) and (yaw is lurus) then (delay is sangatkuat) (1)
5. If (Jarak is dekat) and (yaw is kanan) then (delay is sedang) (1)
6. If (Jarak is sedang) and (yaw is kanan) then (delay is kuat) (1)
7. If (Jarak is jauh) and (yaw is kanan) then (delay is sangatkuat) (1)
8. If (Jarak is sangatjauh) and (yaw is kanan) then (delay is sangatkuat) (1)
9. If (Jarak is dekat) and (yaw is kiri) then (delay is sedang) (1)
10. If (Jarak is sedang) and (yaw is kiri) then (delay is kuat) (1)
11. If (Jarak is jauh) and (yaw is kiri) then (delay is sangatkuat) (1)
12. If (Jarak is sangatjauh) and (yaw is kiri) then (delay is sangatkuat) (1)

Gambar 11. Inference Engine dari Delay

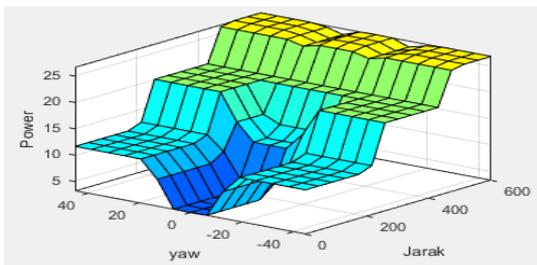
1. If (Jarak is dekat) and (yaw is lurus) then (sudut is datar) (1)
2. If (Jarak is sedang) and (yaw is lurus) then (sudut is datar) (1)
3. If (Jarak is jauh) and (yaw is lurus) then (sudut is lob) (1)
4. If (Jarak is sangatjauh) and (yaw is lurus) then (sudut is sangatlob) (1)
5. If (Jarak is dekat) and (yaw is kanan) then (sudut is datar) (1)
6. If (Jarak is sedang) and (yaw is kanan) then (sudut is lob) (1)
7. If (Jarak is jauh) and (yaw is kanan) then (sudut is sangatlob) (1)
8. If (Jarak is sangatjauh) and (yaw is kanan) then (sudut is sangatlob) (1)
9. If (Jarak is dekat) and (yaw is kiri) then (sudut is datar) (1)
10. If (Jarak is sedang) and (yaw is kiri) then (sudut is lob) (1)
11. If (Jarak is jauh) and (yaw is kiri) then (sudut is sangatlob) (1)
12. If (Jarak is sangatjauh) and (yaw is kiri) then (sudut is sangatlob) (1)

Gambar 12. Inference Engine dari Sudut

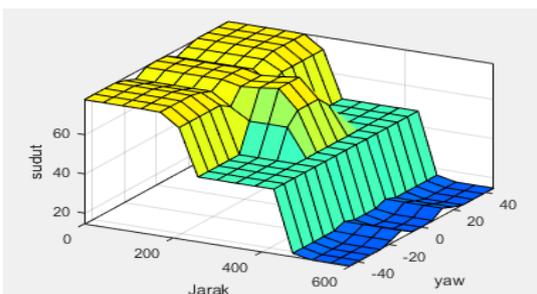
Setelah ditentukan jenis hasil yang akan dieksekusi nantinya, selanjutnya mengubah nilai derajat keanggotaan menjadi nilai yang akan dieksekusi oleh robot, nilai ini merupakan keluaran dari proses *fuzzy* yang ditunjukkan pada persamaan 7.

$$z = \frac{\int z\mu(z)dz}{\int \mu(z)dz} \quad (7)$$

Dari persamaan tersebut didapatkan grafik untuk himpunan keluaran *fuzzy* dari *delay* dapat dilihat pada Gambar 13 dan untuk himpunan keluaran *fuzzy* dari sudut dapat dilihat pada Gambar 14.



Gambar 13 Grafik Keluaran Fuzzy dari Delay



Gambar 14 Grafik Keluaran Fuzzy dari Sudut

Sistem penendang

Sistem penendang yang digunakan merupakan sistem penendang berbasis *magnetic induction* menggunakan solenoid dengan *plunger* yang mendapatkan tegangan dari kapasitor. Pengisian kapasitor menggunakan *boost converter*. Output tegangan dari kapasitor diatur menggunakan *driver* solenoid dengan tujuan agar mendapat hasil tendangan yang sesuai. Selain kontrol tegangan pada *input* solenoid dilakukan juga kontrol tinggi rendah penendang yang menggunakan *Linier rail set* yang digerakkan menggunakan motor DC PG45 sehingga tendangan dapat diatur ketinggiannya.

Desain Sistem

Pada penelitian kali ini STM32F4-discovery digunakan untuk memproses semua data yang ada pada robot, baik dari slave maupun dari sensor-sensor. Pengolahan data menggunakan bahasa pemrograman C yang diprogram menggunakan STM32CubeIDE, dari pengolahan data tersebut nanti dihasilkan parameter yang dapat digunakan untuk robot bergerak dan melakukan tendangan. Pada Gambar 15 merupakan *flowchart* dari sistem kontrol tendangan berdasarkan posisi robot berbasis *Fuzzy Logic Controller*.



Gambar 15. Flowchart Sistem Kontrol Tendangan

HASIL DAN PEMBAHASAN

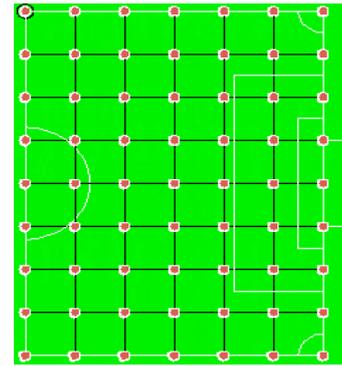
Pengujian Sistem Penendang

Pengujian sistem penendang dilakukan dengan tujuan agar dapat mengetahui kemampuan maksimal dari tendangan yang dihasilkan robot. Pada pengujian ini

Perancangan Sistem Kontrol Tendangan Berdasarkan Posisi Robot Pada *Four Wheeled Omnidirectional Mobile Robot* Berbasis *Fuzzy Logic Controller*

dilakukan pengamatan pada beberapa parameter yakni ketinggian maksimal dan jarak maksimal yang dapat dihasilkan dari tendangan robot. Pengamatan jarak dan ketinggian berdasarkan pengamatan pada alat ukur yakni meteran.

Pengujian dilakukan sebanyak 15 kali percobaan, yaitu dengan 5 jenis nilai *delay* yang digunakan yakni antara 10 hingga 30, Nilai *delay* ini merupakan *delay* lama *trigger* pada *driver* solenoid. Nilai *delay* nantinya divariasikan dengan 3 macam posisi *kicker* yang diatur berdasarkan nilai pembacaan *encoder* motor dari motor penggerak sistem *linier rail set* yang terpasang pada *kicker* sebagai pengatur tinggi rendah *kicker*.



Gambar 16. Pemetaan Titik Pengujian Gyrodometry

| Posisi Kicker | Delay (ms) | Jarak (cm) | Ketinggian (cm) |
|---------------|------------|------------|-----------------|
| Datar | 10 | 80 | 1 |
| | 15 | 130 | 5 |
| | 20 | 180 | 8 |
| | 25 | 230 | 13 |
| | 30 | 320 | 30 |
| Lob | 10 | 60 | 3 |
| | 15 | 150 | 10 |
| | 20 | 190 | 18 |
| | 25 | 280 | 39 |
| | 30 | 400 | 59 |
| Sangat Lob | 10 | 55 | 11 |
| | 15 | 150 | 23 |
| | 20 | 240 | 42 |
| | 25 | 290 | 45 |
| | 30 | 380 | 109 |

Pengujian Sistem Gyrodometry

Pengujian sistem *gyrodometry* dilakukan agar mengetahui tingkat akurasi pembacaan dari *rotary encoder* dan modul MPU6050 terhadap pergerakan robot pada lapangan. Pengujian ini dibagi menjadi 2 tahapan yang pertama, pengujian terhadap perubahan dari nilai x , y , dan θ robot. Jika perubahan dari nilai x , y , dan θ robot sudah sesuai, maka dilakukan pengujian tahap kedua, yakni membandingkan pembacaan nilai x , y , dan θ pada robot dengan kondisi pada lapangan sebenarnya.

Pengujian ini dilakukan pada lapangan yang ada di laboratorium robotika dengan ukuran 8 x 6 meter, nantinya lapangan akan dipetakan menjadi ukuran 1 x 1 meter seperti pada Gambar 14 Sehingga nanti akan dilakukan pengujian sebanyak 63 kali.

Sebelum melakukan pengujian ditentukan titik (0,0) atau titik utama untuk kalibrasi sistem *gyrodometry*, kemudian pengujian dilakukan kearah sumbu x hingga titik (800,0). Setelah sumbu x selesai kembali ketitik (0,0) untuk melakukan kalibrasi kemudian, dilanjut kearah sumbu y pada titik (0,100) hingga titik (800,100) kemudian kembali ketitik (0,0) untuk kalibrasi. Pengujian ini dilakukan hingga robot pada titik (800,600) nilai koordinat robot dalam satuan centimeter. Untuk pembacaan nilai θ melihat pembacaan dari modul MPU6050 yang ditampilkan pada layar LCD bersamaan dengan pembacaan nilai *odometry* sumbu x dan y seperti tabel 4. Ilustrasi dari x , y , dan jarak bisa dilihat pada Gambar 17.



Gambar 17. Penjelasan Nilai Hasil Pengujian Gyrodometry

Tabel 4. Hasil Pengujian Sistem Gyrodometry

| RMSE X (cm) | RMSE Y (cm) | RMSE Jarak (cm) | RMSE θ ($^\circ$) |
|------------------|-------------|-----------------|----------------------------|
| 4,38 | 9,98 | 7,87 | 1,23 |
| 4,26 | 9,20 | 7,19 | 0,84 |
| 4,02 | 16,67 | 12,78 | 0,42 |
| 4,26 | 8,23 | 6,53 | 0,92 |
| 4,63 | 7,18 | 5,50 | 1,21 |
| 4,49 | 9,69 | 4,65 | 0,41 |
| 4,26 | 11,43 | 5,91 | 0,83 |
| Rata – Rata RMSE | | | |
| 4,33 | 10,34 | 7,21 | 0,84 |

Pengujian Sistem Fuzzy Logic Controller

Pengujian *fuzzy logic* digunakan untuk mengetahui apakah *fuzzy logic* dapat bekerja dengan baik dengan menggunakan nilai jarak dan nilai *yaw* pada tiap *membership function* yang memiliki nilai keanggotaan satu karena dianggap merepresentasikan nilai dari setiap keanggotaannya. Nantinya nilai dari keluaran *fuzzy logic* yakni nilai *delay* dan nilai sudut akan ditampilkan pada layar LCD.

Tabel 5. Aturan Fuzzy Logic Nilai Delay

| Yaw(°) | Jarak (cm) | | | | |
|--------|------------|--------|------|-------------|----|
| | Dekat | Sedang | Jauh | Sangat jauh | |
| Lurus | 0 | 3 | 12 | 20 | 30 |
| Kanan | -25 | 12 | 20 | 20 | 30 |
| Kiri | 25 | 12 | 20 | 20 | 30 |

Tabel 6. Aturan Fuzzy Logic Nilai Sudut

| Yaw(°) | Jarak (cm) | | | | |
|--------|------------|--------|------|-------------|----|
| | Dekat | Sedang | Jauh | Sangat jauh | |
| Lurus | 0 | 80 | 80 | 55 | 20 |
| Kanan | -25 | 80 | 55 | 55 | 20 |
| Kiri | 25 | 80 | 55 | 55 | 20 |

Pengujian *fuzzy logic* dilakukan dengan cara memposisikan robot pada titik merah dengan posisi awal robot sama seperti pengujian sistem *gyrodometry*, dengan titik kuning sebagai targetnya. Hasil dari pengujian *fuzzy logic* adalah nilai *delay* dan sudut dengan inputan nilai *yaw* dan jarak seperti pada table 5 dan 6.



Gambar 18. Pemetaan Titik Pengujian Sistem Fuzzy Logic

Gambar 18 merupakan pemetaan dari posisi dan target robot ketika pengujian sistem *fuzzy logic*, pengujian dilakukan secara urut dari titik merah terdekat dengan target yang disimbolkan dengan titik kuning, hingga titik merah terjauh dari target, sementara untuk hasil pengujiannya dapat dilihat pada table 7.

Tabel 7. Hasil Pengujian Sistem Fuzzy Logic

| Nilai Pengujian | | | | Nilai Pembacaan | | | |
|-----------------|------------|------------|-----------|-----------------|------------|------------|-----------|
| Yaw (°) | Jarak (cm) | Delay (ms) | Sudut (°) | Yaw (°) | Jarak (cm) | Delay (ms) | Sudut (°) |
| 0 | 160 | 3 | 80 | 0,7 | 168 | 7 | 90 |
| 0 | 290 | 12 | 80 | 0,3 | 295 | 12 | 90 |
| 0 | 400 | 20 | 55 | -0,1 | 405,2 | 16 | 67 |
| 0 | 500 | 30 | 20 | -6,9 | 505,6 | 21 | 4 |
| -25 | 160 | 12 | 80 | -25,8 | 166,5 | 6 | 90 |
| -25 | 290 | 20 | 55 | -24,8 | 296,1 | 12 | 68 |
| -25 | 400 | 20 | 55 | -26,4 | 404,7 | 16 | 67 |
| -25 | 500 | 30 | 20 | -27,4 | 506,2 | 21 | 4 |
| 25 | 160 | 12 | 80 | 24,1 | 158,2 | 6 | 90 |
| 25 | 290 | 20 | 55 | 23 | 295,3 | 12 | 67 |
| 25 | 400 | 20 | 55 | 23,1 | 405,4 | 16 | 67 |
| 25 | 500 | 30 | 20 | 21,4 | 506,1 | 21 | 4 |

Dari hasil pengujian *fuzzy logic* terhadap nilai *delay* dan nilai sudut, dapat dilihat bahwasannya sistem *fuzzy logic* bekerja dengan baik dikarenakan hasil pembacaan dari sistem *fuzzy logic* berada pada anggota *membership function*. Sehingga dapat dilakukan pengujian seluruh system seperti table 7.

Pengujian Keseluruhan Sistem

Setelah melakukan pengujian sistem yang ada secara terpisah, tahapan selanjutnya adalah melakukan pengujian sistem secara keseluruhan dengan melibatkan semua sistem yang ada yaitu sistem penendang, sistem *gyrodometry*, dan sistem *fuzzy logic*.

Pengujian awal

Pengujian awal dilakukan bertujuan untuk memastikan semua sistem dapat bekerja secara bersamaan dengan baik, sebelum nantinya dilakukan pengambilan data seperti tabel 8.

Tabel 8. Hasil Pengujian Awal Keseluruhan Sistem

| Masukkan | | Keluaran | | |
|--------------|--------------|----------|------------|-----------|
| Sumbu X (cm) | Sumbu Y (cm) | Yaw (°) | Delay (ms) | Sudut (°) |
| 200 | 150 | -24 | 20 | 4 |
| 200 | 300 | -34 | 14 | 67 |
| 200 | 450 | -53 | 9 | 90 |
| 400 | 150 | 0 | 18 | 4 |
| 400 | 300 | 0 | 12 | 68 |
| 400 | 450 | 0 | 6 | 90 |
| 600 | 150 | 24 | 20 | 4 |
| 600 | 300 | 34 | 14 | 68 |
| 600 | 450 | 53 | 9 | 90 |

Pengujian akhir

Pengujian akhir dilakukan pada lapangan yang berukuran 8 x 6-meter yang telah di petakan menjadi ukuran 2 x 1,5-meter sehingga terdapat 12 petak yang disetiap petak diberikan titik percobaan untuk lebih jelasnya pada Gambar 19. Pada 12 titik percobaan nantinya tiap titik akan dilakukan 3 kali percobaan dengan tujuan agar hasil tendangan yang didapatkan memiliki nilai respon yang baik.



Gambar 19. Pemetaan Titik Pengujian Keseluruhan Sistem

Titik berwarna putih merupakan posisi awal robot atau titik (0,0,0) robot. Kemudian titik berwarna merah merupakan titik posisi robot sedangkan titik kuning merupakan target atau arah hadap robot saat di titik merah. Ketika robot sudah pada titik merah nantinya robot akan diberikan bola kemudian robot akan melakukan tendangan ke arah target, dengan nilai delay dan nilai sudut sesuai dengan aturan *fuzzy logic* yang telah dibuat.

Tabel 9. Percobaan Pertama

| Set Posisi | | Tendangan | | Posisi Robot | |
|------------|---------|------------|-----------|--------------|---------|
| Jarak (cm) | Yaw (°) | Delay (ms) | Sudut (°) | Jarak (cm) | Yaw (°) |
| 592 | -28 | 24 | 4 | 599 | -29 |
| 465 | -36 | 19 | 66 | 460 | -35,5 |
| 355 | -51 | 14 | 67 | 347,2 | -51,2 |
| 285 | -75 | 11 | 90 | 273 | -74,3 |
| 525 | 0 | 21 | 4 | 524 | 1,1 |
| 375 | 0 | 15 | 67 | 371 | 1,1 |
| 255 | 0 | 9 | 90 | 221 | 1,1 |
| 75 | 0 | 2 | 90 | 70,6 | -0,1 |
| 592 | 28 | 23 | 4 | 562,3 | 28,6 |
| 465 | 36 | 17 | 67 | 414,6 | 36,8 |
| 355 | 51 | 11 | 90 | 286,5 | 51,8 |
| 285 | 75 | 8 | 90 | 198,3 | 76 |

Tabel 10. Percobaan Kedua

| Set Posisi | | Tendangan | | Posisi Robot | |
|------------|---------|------------|-----------|--------------|---------|
| Jarak (cm) | Yaw (°) | Delay (ms) | Sudut (°) | Jarak (cm) | Yaw (°) |
| 592 | -28 | 24 | 4 | 598,2 | -27,8 |
| 465 | -36 | 19 | 67 | 460 | -35,4 |
| 355 | -51 | 14 | 67 | 346,2 | -51,5 |
| 285 | -75 | 11 | 90 | 275 | -74,6 |
| 525 | 0 | 21 | 5 | 525 | 1,8 |
| 375 | 0 | 15 | 66 | 370 | 1,5 |
| 255 | 0 | 9 | 90 | 220 | 0,9 |
| 75 | 0 | 2 | 90 | 69,5 | -0,2 |
| 592 | 28 | 23 | 4 | 559,5 | 27,4 |
| 465 | 36 | 17 | 67 | 416,9 | 37,2 |
| 355 | 51 | 12 | 90 | 288,4 | 51,5 |
| 285 | 75 | 8 | 90 | 198,3 | 74,5 |

Tabel 11. Percobaan Ketiga

| Posisi | | Tendangan | | Posisi Robot | |
|------------|---------|------------|-----------|--------------|---------|
| Jarak (cm) | Yaw (°) | Delay (ms) | Sudut (°) | Jarak (cm) | Yaw (°) |
| 592 | -28 | 24 | 5 | 599 | -29,4 |
| 465 | -36 | 19 | 66 | 458,2 | -37,4 |
| 355 | -51 | 14 | 67 | 342,5 | -51,1 |
| 285 | -75 | 10 | 90 | 261,7 | -75,1 |
| 525 | 0 | 21 | 5 | 525 | 1,9 |
| 375 | 0 | 15 | 67 | 371,1 | 0,9 |
| 255 | 0 | 9 | 90 | 221,7 | 1,1 |
| 75 | 0 | 2 | 90 | 71,4 | 0,1 |
| 592 | 28 | 23 | 4 | 561 | 28,1 |
| 465 | 36 | 17 | 67 | 414,6 | 36,4 |
| 355 | 51 | 11 | 90 | 284,5 | 52,1 |
| 285 | 75 | 7 | 90 | 189,8 | 76,1 |

Setelah dilakukan pengujian akhir keseluruhan sistem didapatkan hasil yang sesuai dengan aturan *Fuzzy Logic* yang telah dibuat. Pengujian dilakukan 3 kali percobaan pada 12 titik pengujian sehingga total ada 36 kali tendangan, dari 36 kali tendangan didapatkan hasil tendangan yang sesuai dengan aturan *fuzzy* yang telah dibuat sehingga sistem ini dianggap bekerja dengan baik, dikarenakan nilai dari keluarannya bergantung pada nilai masukkan yang diterima.

Dari 3 kali percobaan dengan 12 titik pengujian juga didapatkan hasil bahwa sistem *gyrodometry* nilai errornya akan terus bertambah tergantung banyaknya gerakan pada robot, semakin robot banyak bergerak maka semakin besar nilai errornya seperti pada tabel 9, 10, dan 11.

PENUTUP

Simpulan

Berdasarkan hasil perancangan dan pengujian yang telah didapatkan pada penelitian ini, maka dapat disimpulkan bahwa perancangan sistem kontrol tendangan berdasarkan posisi robot dimulai dengan melakukan perencanaan mekanik dan hardware robot yang kemudian dilanjutkan melakukan perencanaan software dari robot. Setelah tahapan perencanaan dilakukan pengujian sistem yang ada pada robot, di dapatkan hasil sistem penendang bekerja secara baik dengan jarak mencapai 4-meter dan tinggi mampu mencapai 1 meter. Pada pengujian sistem *gyrodometry* sebagai sistem positioning robot cukup efektif, hal ini dibuktikan dengan nilai RMSE 4,33 cm untuk sumbu x , RMSE 10,73 cm untuk sumbu y , dan RMSE 0,89° untuk nilai yaw.

Metode Fuzzy Logic Controller diterapkan sebagai pengambilan keputusan ketika melakukan tendangan berdasarkan parameter yang ada. Pengaplikasian sistem kontrol tendangan berdasarkan posisi robot berhasil dibuat dengan menerapkan fuzzy logic sebagai kontrolnya. Hal ini ditunjukkan dengan hubungan antara besarnya nilai jarak dan yaw terhadap nilai delay dan sudut.

Hasil respon dari pengujian sistem didapatkan bahwa robot dapat melakukan tendangan kearah gawang dengan pola tendangan berdasarkan posisi robot terhadap gawang dan bola hasil tendangan masuk kedalam gawang sehingga dapat disimpulkan respon dari sistem kontrol tendangan bekerja dengan baik, dengan nilai RMSE jarak 37,75 cm dan nilai RMSE yaw 0,81° pada percobaan pertama, serta nilai RMSE jarak 37,46 cm dan nilai RMSE yaw 0,88 ° untuk percobaan kedua, untuk percobaan ketiga didapatkan nilai RMSE jarak 40,27 cm dan nilai RMSE yaw 1,00 °.

Saran

Berdasarkan pengujian yang dilakukan terdapat beberapa saran untuk pengembangan pengembangan sistem *positioning* robot terhadap lapangan yang lebih presisi agar tidak perlu kalibrasi secara rutin. Seperti sistem posisi robot menggunakan kamera sehingga tidak terpengaruh permukaan lapangan yang tidak rata.

Perancangan sistem mekanik dan hardware yang lebih andal agar jarak dan tinggi tendangan yang dihasilkan bisa lebih maksimal lagi.

DAFTAR PUSTAKA

- Al-Azka, Muhammad Hisyam. 2021. *Rancang Bangun Sistem Positioning Mobile Robot Omnidirectional Wheel Menggunakan STM32 Berbasis Fuzzy Logic Controller*. Jurnal Teknik Elektro Volume 10 No. 03.
- Ananthababu. B, Ganesh. C, dan Pavithra. C. V. 2017. *Fuzzy Based Speed Control of BLDC Motor with Bidirectional DC-DC Converter*. Proceedings of 2016 Online International Conference on Green Engineering and Technologies, IC-GET 2016.
- Cahyono. K, Wibowo. I. K, dan Bachtiar. M. M. 2020. *A New Kicker System of Wheeled Soccer Robot Ersow Using Fuzzy Logic Method*. In IES 2020 - International Electronics Symposium: The Role of Autonomous and Intelligent Systems for Human Life and Comfort (pp. 219–225). Institute of Electrical and Electronics Engineers Inc.
- Denegri. E, Munoz-Panduro. E, dan Ramos. O. E. 2019. *Step-by-step Development of an Omnidirectional Mobile Robot*. In SHIRCON 2019 - 2019 IEEE Sciences and Humanities International Research Conference. Institute of Electrical and Electronics Engineers Inc.
- Fikri, Achmad Akmal. 2021. *Mapping And Localization System Pada Mobile Robot Menggunakan Metode SLAM Berbasis LiDAR*. Journal of Information Engineering and Educational Technology, 5(1), 27–33.
- Firmansyah, Fajar Ilham. 2023. *Sistem Obstacle Avoidance Pada Omnidirectional Mobile Robot Dengan Metode Artificial Potential Field (APF) Berbasis Fuzzy logic controller (FLC)*. Jurnal Teknik Elektro Volume 12(2), 6–13.
- Holis, Nur. 2023. *Rancang Bangun Sistem Positioning Robot Beroda Dengan Four Omnidirectional Wheels Menggunakan Mpu6050 Dan Rotary Encoder Berbasis Pd-Fuzzy Controller*. Jambura Journal of Electrical and Electronics Engineering.
- Kasiram, Moh. 2008. *Metodologi Penelitian*. Malang: UIN-Malang Pers.
- Khairullah. M. R, Purwanto. D, dan Kusuma. H. 2020. *Sistem Kontrol Jarak Tendang Bola pada Robot Sepak Bola Beroda Menggunakan Fuzzy Logic*. Surabaya: Institut Teknologi Sepuluh Nopember.
- Nurariyanto, Kris. 2022. *Sistem Positioning Pada Four-Wheeled Omnidirectional Mobile Robot Menggunakan Metode Gyrodometry Berbasis PI-Fuzzy Controller*. Jurnal Teknik Elektro Volume 11(1), 77–87.

Perancangan Sistem Kontrol Tendangan Berdasarkan Posisi Robot Pada *Four Wheeled Omnidirectional Mobile Robot* Berbasis *Fuzzy Logic Controller*

Rachman, Aulia Aditya. 2017. *Sistem Perencanaan Rute Gerak Pada Robot Sepakbola Beroda*. Surabaya: Institut Teknologi Sepuluh Nopember.

Rachmawan, Adnan. 2017. *Penentuan Posisi Robot Sepak Bola Beroda Menggunakan Rotary Encoder dan Kamera*. Tugas Akhir ITS, 4(November), 172–176.

