

## **Sistem Obstacle Avoidance Pada Omnidirectional Mobile Robot Dengan Metode Artificial Potential Field (APF) Berbasis Fuzzy logic controller (FLC)**

**Fajar Ilham Firmansyah**

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya  
e-mail: fajar.19045@mhs.unesa.ac.id.

**Muhammad Syariffuddien Zuhrie, Miftahur Rohman, Endryansyah, Masyitah Aulia**

S1 Teknik Elektro, Fakultas Teknik, Universitas Negeri Surabaya, Teknik Elektro, Universitas Negeri Sumbawa  
e-mail: zuhrie@unesa.ac.id, miftahurrohman@unesa.ac.id, endryansyah@unesa.ac.id, masyitah.aulia@uts.ac.id.

### **Abstrak**

Tingginya angka pengaplikasian di seluruh dunia menjadikan studi dan pengembangan robot menjadi salah satu perhatian penting bagi para peneliti ilmiah pada saat ini. Salah satu jenis robot yang menarik perhatian banyak peneliti adalah *mobile robot*. *Mobile robot* yang digunakan pada penelitian ini adalah *omnidirectional mobile robot*, yang merupakan robot holonomic yang dapat berotasi dan bertranslasi secara bersamaan. Salah satu system penunjang *mobile robot* ini adalah *obstacle avoidance* yang mana merupakan sebuah sistem yang membantu robot untuk menghindari rintangan pada saat robot berjalan. Robot akan menghindari *obstacle* berdasarkan parameter-parameter yang dihasilkan oleh system dari robot into sendiri, seperti data sudut putar robot dari gyroscope, data jarak tempuh robot dari *rotary encoder*, kemudian data *image processing* dari *omnivision*. Hasil dari pengujian system *obstacle avoidance* ini merupakan nilai dari posisi robot sebelum dan sesudah menghindari *obstacle*, nilai yang dihasilkan berupa titik koordinat robot yang memiliki format (x, y) dimana x adalah titik koordinat secara horizontal dan y adalah titik koordinat secara vertical yang memiliki satuan cm. Hasil dari pengujian ini meghasilkan nilai rata-rata sebesar 41.11 cm sebelum *obstacle* dan selesai melakukan penghindaran rata-rata 42.77 cm setelah *obstacle*.

**Kata Kunci:** *Obstacle Avoidance, Fuzzy logic controller, Mobile Robot.*

### **Abstract**

The high rate of implementation worldwide has made the study and development of robots an important focus for scientific researchers at present. One type of robot that has captured the attention of many researchers is the mobile robot. The mobile robot used in this study is an omnidirectional mobile robot, which is a holonomic robot capable of rotating and translating simultaneously. One supporting system for this mobile robot is *obstacle avoidance*, which is a system that helps the robot avoid *obstacles* while it moves. The robot will avoid *obstacles* based on parameters generated by its own system, such as the robot's rotation angle data from the gyroscope, the robot's distance traveled data from the rotary encoder, and image processing data from the omnivision. The results of testing this *obstacle avoidance* system are the values of the robot's position before and after avoiding *obstacles*, expressed as coordinates (x, y), where x represents the horizontal coordinate and y represents the vertical coordinate, both in centimeters. The results of this testing yielded an average value of 41.11 cm before encountering an *obstacle* and an average of 42.77 cm after successfully avoiding the *obstacle*.

**Keywords:** *Obstacle Avoidance, Fuzzy logic controller, Mobile Robot.*

### **PENDAHULUAN**

Tingginya angka pengaplikasian di seluruh dunia menjadikan studi dan pengembangan robot menjadi salah satu perhatian penting bagi para peneliti ilmiah pada saat ini. Peranan robot dalam membantu kehidupan manusia merupakan hal yang sudah biasa ditemui pada kehidupan sehari-hari, selain membuat pekerjaan menjadi lebih mudah penggunaan robot pada kehidupan sehari-hari juga menjadikan pekerjaan yang dikerjakan lebih efisien dan lebih cepat. Seiring berjalannya waktu dan luasnya penggunaan robot pada berbagai bidang menjadikan robot

ini semakin rumit dari tahun ke tahun, baik secara fungsional maupun struktural (Ali dkk. 2016). Omnidirectional Mobile robot merupakan salah satu jenis robot generasi baru yang sering menjadi studi dan pengembangan para peneliti di dunia pada saat ini.

Omnidirectional robot merupakan robot holonomic yang bisa berotasi dan bertranslasi secara bersamaan (Williams dkk. 2010). Pada penelitian ini menggunakan Four Wheeled Omnidirectional Mobile robot, penggunaan Four Wheeled Omnidirectional Mobile robot dinilai mampu melakukan hingga lebih dari 8 arah gerakan

(kanan, kiri, maju, mundur, serong kanan depan, serong kiri depan, serong kanan belakang, serong kiri belakang) tanpa menyesuaikan arah hadapnya (Nurariyanto. 2022). Dalam penerapannya *Four Wheeled Omnidirectional Mobile robot* memerlukan suatu sistem penghindaran objek yang dianggap dapat mengganggu jalannya robot yaitu *obstacle avoidance*.

*Obstacle avoidance* merupakan suatu sistem yang cukup vital pada *Four Wheeled Omnidirectional Mobile robot*. Penggunaan *obstacle avoidance* diharapkan dapat meningkatkan keamanan dan mencegah hal-hal yang tidak diinginkan baik untuk keselamatan pengguna maupun robot itu sendiri.

Penggunaan sistem *obstacle avoidance* memerlukan beberapa parameter yang mendukung sistem tersebut untuk bekerja. Pada penelitian ini beberapa parameter yang digunakan untuk mendukung kerja dari sistem *obstacle avoidance* diantaranya yaitu nilai sudut putar robot yang diperoleh oleh sensor IMU (*Inersia Measurement Unit*), nilai jarak perpindahan dan kecepatan robot yang diperoleh oleh encoder motor, serta nilai sudut dan jarak dari objek *obstacle* yang dideteksi oleh robot yang diperoleh dari omnivision camera, yang nantinya diproses dengan *Fuzzy logic controller* untuk menciptakan gerak robot yang diinginkan.

*Fuzzy logic controller* sendiri merupakan penggabungan kecerdasan manusia ke dalam proses kontrol sistem, sehingga akan memberikan respon dinamik pada sistem (Nurariyanto. 2022). Selain itu *Fuzzy logic controller* dapat mengontrol gerak mobile robot dan secara otomatis mengurangi kesalahan geraknya (Al-Azka. 2021). Dengan penggunaan *Fuzzy Controller* ini diharapkan bisa memberikan respon yang diinginkan pada sistem *obstacle avoidance*.

Adapun tujuan dari penelitian ini diantaranya, (1) Mengetahui tahapan perancangan sistem *Obstacle Avoidance* dengan menggunakan metode *Artificial potential field (APF)* pada *Omnidirectional Mobile Robot*. (2) Mengetahui hasil respon sistem *Obstacle Avoidance* pada *Four Wheeled Omnidirectional Mobile Robot*.

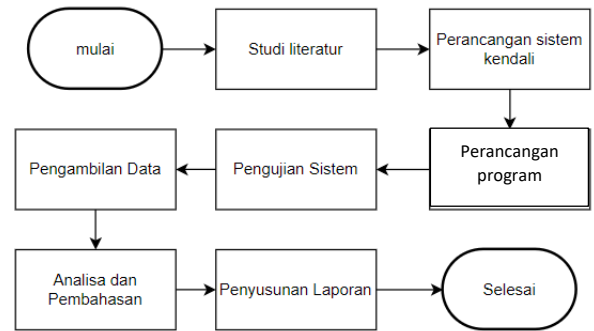
## METODE

### Pendekatan Penelitian.

Metode penelitian yang digunakan kali ini menggunakan metode pendekatan kuantitatif. dalam buku *Metodologi Penelitian Kualitatif dan Kuantitatif* yang ditulis oleh Kasiram (2008 : 149) bahwa penelitian kuantitatif merupakan suatu proses menemukan pengetahuan data berupa nomor yang dimanfaatkan dalam menganalisis pengetahuan mengenai apa yang ingin diketahui (Nurariyanto, 2022).

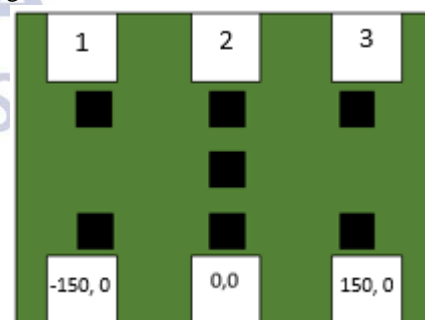
### Rancangan Penelitian.

Penelitian dilakukan melalui 7 tahap diantaranya yaitu, studi literatur, perancangan sistem kendali, perancangan software, pengujian system, pengambilan data, Analisa dan pembahasan, serta penyusunan laporan. Untuk lebih jelasnya bisa dilihat pada gambar 1.



Gambar 1. Tahapan Penelitian

Pada tahap pertama yaitu studi literatur, peneliti akan mencari serta mempelajari berbagai macam referensi dan literatur dari beberapa penelitian yang masih berkaitan. Referensi atau literatur dapat diperoleh melalui jurnal, buku atau karya penelitian orang lain. Setelah referensi berhasil didapatkan maka tahap selanjutnya adalah perancangan sistem kendali, dimana pada tahap ini akan menggunakan metode *Artificial Potential Field (APF)* yang dibarengi dengan *Fuzzy logic controller (FLC)* sebagai sistem penggerak dari *omnidirectional mobile robot*. Ketika sistem kendali selesai dirancang maka tahap selanjutnya adalah perancangan program, dimana disini akan dilakukan proses pemrograman pada mikrokontroler *STM32F4-Discovery Board*. Setelah pemrograman selesai dilakukan maka sistem akan diujikan, pada pengujian sistem robot akan diberi perintah untuk bergerak menuju tujuan secara vertikal dan diagonal dan akan menghindari satu rintangan.

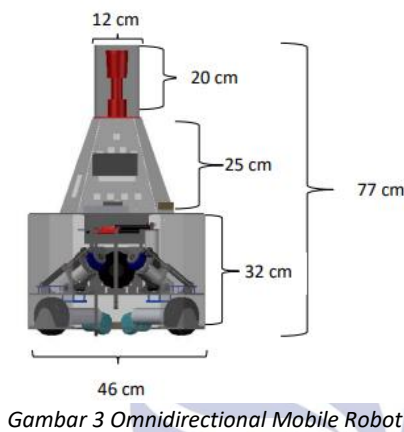


Gambar 2 Denah Lapangan

Gambar 2 merupakan denah lapangan untuk pengujian robot, pada lapangan itu terdapat 7 kotak hitam dan 6 kotak putih, 7 kotak hitam merupakan calon posisi untuk rintangan sedangkan kotak putih yang memiliki notasi angka 1, 2 dan 3 merupakan titik tujuan dari robot, sedangkan untuk kotak putih yang memiliki notasi nomor (-150, 0), (0,0) dan (150,0) adalah titik start dari robot.

Setelah pengujian berhasil dilakukan maka akan dilakukan pengambilan data, Hasil dari pengujian ini merupakan nilai dari posisi robot sebelum dan sesudah menghindari *obstacle*, nilai yang dihasilkan berupa titik koordinat robot yang memiliki format (x, y) dimana x adalah titik koordinat secara horizontal dan y adalah titik koordinat secara vertical yang memiliki satuan cm. Ketika semua data sudah didapatkan maka akan dilakukan analisa dan pembahasan untuk mengolah data yang telah didapatkan, setelah dilakukan analisa pada data maka dibuatlah laporan atas penelitian tersebut.

**Desain Plant**

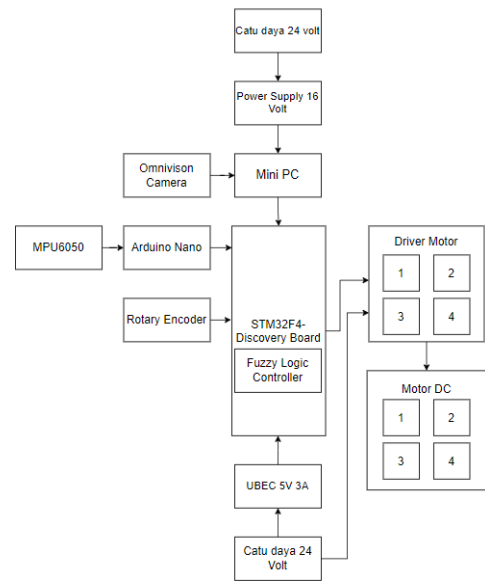


Pada gambar 3 dijelaskan bahwa robot memiliki dimensi 46 cm x 77 cm. Struktur mekanik robot menggunakan bahan dasar alumunium dengan menempatkan 4 motor DC sebagai penggerak robot dengan menggunakan roda jenis omniswheel agar robot bisa bergerak omnidirectional. Robot terdiri dari 3 lapisan dasar yang terbuat dari lembaran aluminium. Dengan lapisan pertama terdapat motor penggerak, sensor garis, kapasitor, rotary encoder dan module IGBT. Pada lapisan kedua terdapat motor sebagai handling, penendang selenoid, dan juga power management robot. Pada lapisan ketiga dibentuk struktur penyangga kamera omnivision dan juga terdapat mikrokontroler serta mini pc.

Robot memiliki 3 controller diantaranya STM32F4-Discovery Board, MiniPC intel NUC i7 dan arduino nano. MiniPC intel NUC i7 sebagai sub controller berfungsi sebagai tempat image processing berlangsung, gambar yang diperoleh dari omnivision camera akan dikirim ke miniPC untuk diproses sebelum dikirimkan lagi ke mikrokontroler. Arduino nano sebagai sub controller digunakan untuk pemrosesan data gyroscope yang diambil dari sensor IMU MPU6050, selanjutnya data gyroscope yang telah didapat akan dikirimkan menuju STM32F4-Discovery Board melalui koneksi serial. STM32F4-Discovery Board sebagai main controller, mikronkontroler STM32F4-Discovery Board akan menerima data dari kedua sub controller yang selanjutnya menjadi parameter untuk pergerakan robot. Selain menerima data dari dua sub

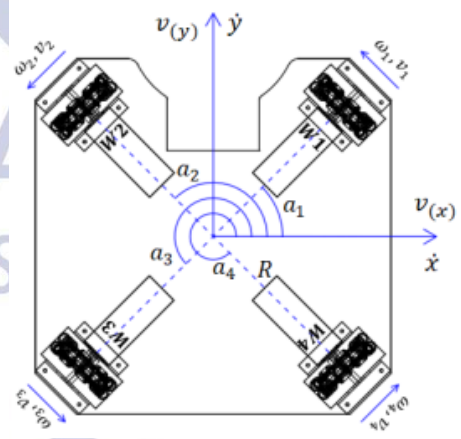
controller tersebut, main controller ini juga terkoneksi langsung dengan push button, lcd 20 x 4, rotary encoder dan juga motor penggerak.

Sistem penggerak pada robot ini menggunakan 4 buah motor DC yang masing-masing terhubung ke main controller melalui perantara driver motor. Motor DC yang digunakan merupakan motor DC PG45 dengan diameter 45 mm, memiliki kecepatan putaran kurang lebih 5000 rpm, dengan arus kerja 2,5 A sampai 5 A dan tegangan kerja 24 Vdc, untuk lebih jelasnya bisa dilihat pada gambar 4.



Gambar 4 Desain Hardware Omnidirectional Mobile Robot

**Invers Kinematik**



Gambar 5 Penempatan Roda (sumber: Nurariyanto, 2022)

Invers kinematik merupakan metode untuk menentukan nilai kecepatan setiap roda menggunakan referensi dari nilai jarak dan orientasi yang dituju (Al-Azka, 2021). Penempatan posisi roda sangat memengaruhi dalam pembuatan invers kinematic, pada penelitian ini digunakan robot omnidirectional dengan 4 (empat) buah roda dengan penempatan sama seperti gambar 5.

Pada gambar 5 dapat diketahui bahwa  $a_1, a_2, a_3, a_4$  merupakan sudut antar roda serta titik referensi robot,  $\omega_1, \omega_2, \omega_3, \omega_4$  merupakan kecepatan sudut setiap roda,  $v_1, v_2, v_3, v_4$  merupakan kecepatan linier roda,  $W_1, W_2, W_3, W_4$  merupakan simbol setiap roda,  $R$  merupakan jarak roda dari pusat robot,  $v_{(x)}$  merupakan kecepatan linier robot terhadap sumbu x,  $v_{(y)}$  merupakan kecepatan linier robot terhadap sumbu y, kemudian kecepatan linier robot terhadap frame dapat dilambangkan dengan  $v_{(R)} = [v_{(x)} \ v_{(y)}]^T$ , posisi koordinat dan orientasi robot terhadap koordinat global dilambangkan dengan  $X_{(g)} = [x_{(R)} \ y_{(R)} \ \Psi]^T$ , kecepatan linier dan sudut robot terhadap koordinat global XOY dilambangkan dengan  $V_{(g)} = [\dot{X} \ \dot{Y} \ \omega]$ .

Berdasarkan gambar 5 dapat dibuat persamaan 1 dimana merupakan persamaan untuk arah gerak robot pada koordinat global, kemudian persamaan 2 yang merupakan persamaan untuk resultan kecepatan linier, dan 3.

$$\varphi_{(R)} = \tan^{-1} \frac{Y_{(R)}}{X_{(R)}} \quad (1)$$

$$L = \sqrt{X_{(R)}^2 + Y_{(R)}^2} \quad (2)$$

$$\begin{bmatrix} v_{(x)} \\ v_{(y)} \\ \omega \end{bmatrix} = \begin{bmatrix} \cos(\Psi) & \sin(\Psi) & 0 \\ -\sin(\Psi) & \cos(\Psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \omega \end{bmatrix} \quad (3)$$

Kemudian didapatkan persamaan untuk untuk kecepatan total pada persamaan 4, kecepatan linier dan sudut robot terhadap koordinat global XOY pada persamaan 5 dan invers kinematic pada persamaan 6.

$$v_w = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix} = T(\theta) \cdot v_{(g)} \quad (4)$$

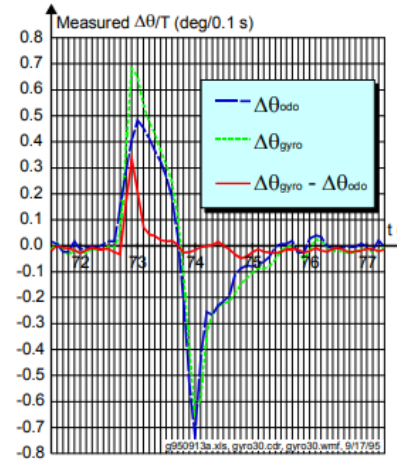
$$V_{(g)} = [\dot{X} \ \dot{Y} \ \omega]^T \quad (5)$$

$$T(\theta) = \begin{bmatrix} -\sin(\Psi + a_1) & \cos(\Psi + a_1) & R \\ -\sin(\Psi + a_2) & \cos(\Psi + a_2) & R \\ -\sin(\Psi + a_3) & \cos(\Psi + a_3) & R \\ -\sin(\Psi + a_4) & \cos(\Psi + a_4) & R \end{bmatrix} \quad (6)$$

### Gyrodometry

Gyrodometry merupakan metode penggabungan nilai gyroscope dan nilai odometry. Tujuan penggabungan kedua nilai ini dimaksudkan untuk mengurangi nilai error dari nilai odometry. Odometry sendiri merupakan pengukuran yang memanfaatkan data dari aktuator dalam memprediksi pergerakan robot (Dudek, 2008). Salah satu kekurangan dari odometry sendiri merupakan avdanya

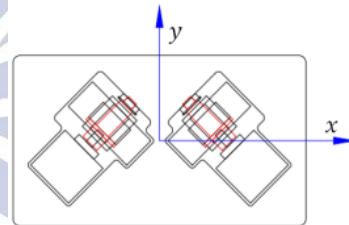
potensi error non-systematic yang diakibatkan oleh gundukan atau permukaan tidak rata (Borenstein dan Feng, 1996).



Gambar 6 Grafik Pengukuran gyro dan odometry (sumber: Borenstein dan Feng, 1996)

Sebuah data eksperimen pada gambar 6 mendukung asumsi bahwa gyrodometry didasarkan pada asumsi jika perbedaan antara kurva odometry dan kurva gyroscope hanya bertahan pada waktu yang sangat singkat (Nurariyanto, 2022).

Pemasangan rotary encoder tambahan seperti pada gambar 7 dimaksudkan untuk memperkecil nilai error non-systematic yang kemungkinan terjadi.



Gambar 7 Posisi Rotary encoder (sumber: Nurariyanto, 2022)

Pemasangan rotary encoder tersebut juga dapat dimanfaatkan untuk menghitung kecepatan robot pada sumbu x dan sumbu y dengan persamaan 7.

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} \cos(45^\circ) & \sin(45^\circ) \\ \sin(45^\circ) & \cos(45^\circ) \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix} \quad (7)$$

Dengan  $\omega$  sebagai kecepatan sudut rotary encoder. Kecepatan sudut rotary encoder didapatkan dari counter pada mikrokontroler yang diperbaharui setiap 1 ms (Nurariyanto, 2022). Persamaan diatas dapat menghasilkan nilai kecepatan robot pada setiap sumbu geraknya yaitu sumbu x dan sumbu y, akan tetapi pada persamaan diatas belum mendapat koreksi nilai dari gyroscope. Untuk mendapatkan koreksi nilai dari gyroscope dapat ditunjukkan dengan persamaan 8.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} -\cos(\theta) & \sin(\theta) \\ -\sin(\theta) & -\cos(\theta) \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix} \quad (8)$$

Dengan nilai  $\theta$  merupakan nilai yang telah didapatkan dari sensor gyroscope. Koreksi nilai oleh gyroscope dimaksudkan agar ketika robot melakukan manuver berputar terhadap arah hadapnya, sensor tetap menghitung kecepatan arah x dan arah y. Untuk mendapatkan nilai koordinat posisi pada sumbu X serta sumbu Y dapat menggunakan persamaan 9 dan 10.

$$X = \Sigma x \quad (9)$$

$$Y = \Sigma y \quad (10)$$

### Image Processing System

Pada penelitian ini sebuah *omnivision camera* digunakan dalam pendeteksian objek *obstacle*. Omnivision camera merupakan jenis kamera yang bisa menangkap gambar secara 360°. Seperti yang ditampilkan pada gambar 8, kamera ini terdiri dari sebuah kamera digital serta lensanya dan sebuah cermin hyperbolic. Kamera digital yang dipakai merupakan ELP Digital Camera yang memiliki resolusi sebesar 3840 x 2160p, dan mampu menangkap gambar sampai dengan 30 fps. Kamera digital ini juga dilengkapi dengan cermin hyperbolic, cermin hyperbolic sendiri merupakan sebuah cermin cembung yang diletakkan diatas lensa kamera, penggunaan cermin hyperbolic inilah yang memungkinkan kamera digital untuk menangkap gambar secara 360°, hasil gambar yang dihasilkan oleh omnivision bisa dilihat pada gambar 9.



Gambar 8 Omnivision Camera

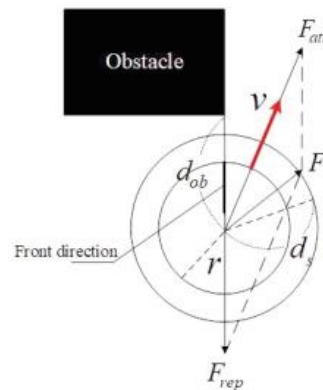


Gambar 9 Hasil Omnivision Camera

Objek *obstacle* atau rintangan akan diproses sedemikian rupa sebelum akhirnya dijadikan sebagai parameter untuk *obstacle avoidance*. Objek *obstacle* atau rintangan akan ditangkap oleh omnivision yang terhubung dengan prototype, gambar yang telah diperoleh ditangkap oleh kamera akan diproses melalui miniPC Intel NUC i7 yang selanjutnya akan diolah dengan metode OpenCV. OpenCV sendiri merupakan library open-source yang digunakan untuk mendukung sistem image processing.

Seperti namanya *Artificial potential field* (APF) bekerja dengan cara menciptakan dua jenis medan gaya buatan, dimana salah satunya menjadi gaya atraktif atau gaya tarik menarik dan yang lain menjadi gaya repulsif atau gaya tolak menolak. Pada penelitian ini objek rintangan akan dijadikan medan gaya repulsif dan titik tujuan akan dijadikan medan gaya atraktif terhadap robot.

Metode APF ini memiliki keunggulan berupa deskripsi matematis yang sederhana, kecepatan komputasi yang rendah, kecepatan respon yang tinggi, dan pergerakan yang cukup lancar (Li dkk. 2019). Sampai hari ini metode ini telah menjadi metode yang cukup efektif untuk digunakan.



Gambar 10 Ilustrasi metode APF (sumber: Shih-An Li dkk, 2019)

Artificial Potential Field (APF) memiliki model matematis seperti yang ditunjukkan pada persamaan 11 dimana merupakan persamaan untuk gaya atraktif, persamaan 12 yang merupakan gaya repulsif dan persamaan 13 yang merupakan komposisi dari kedua gaya.

$$F_{att} = \alpha \times v, \quad (11)$$

$$F_{Rep} = \begin{cases} \beta \times (d_s - d_{ob})^2 \times v, & d_{ob} \leq d_s \\ 0, & d_{ob} > d_s \end{cases} \quad (12)$$

$$F = F_{att} \times F_{Rep} \quad (13)$$

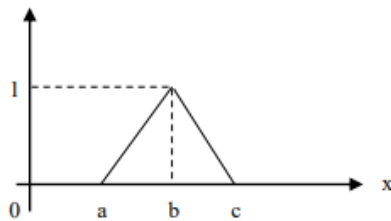
Dimana  $F_{att}$  merupakan gaya atraktif,  $F_{Rep}$  merupakan gaya repulsif,  $F$  merupakan komposisi gaya  $\alpha$  merupakan

nilai parameter dari gaya atraktif,  $\beta$  merupakan nilai parameter dari gaya repulsif,  $v$  merupakan vektor dari kecepatan robot,  $d_{ob}$  merupakan jarak antara robot dan rintangan, dan  $d_s$  merupakan jarak aman antara robot dan rintangan atau titik tujuan.

**Fuzzy logic controller.**

*Fuzzy logic controller (FLC)* adalah algoritma pengendalian sistem atau proses berdasarkan pengetahuan manusia tentang bagaimana mengontrol sistem tanpa menggunakan model matematika (Nurariyanto, 2022). FLC didasarkan pada fuzzy logic yang menyediakan cara untuk mengkonversi strategi kontrol linguistik yang berbasis pada pengetahuan ahli menjadi strategi kontrol otomatis (Lee, 1990). Berbeda dengan logika boolean yang hanya memiliki nilai 0 dan 1 saja, logika fuzzy memiliki nilai diantara 0 dan 1 yang dinamakan derajat keanggotaan.

Pada penerapannya, variabel pada fuzzy logic dideskripsikan dalam bentuk himpunan fuzzy yang memiliki beberapa bentuk, diantaranya adalah segitiga, trapezoidal, gaussian, gaussian-bell, dan sigmoid. Nilai dari derajat keanggotaan juga ditentukan oleh himpunan fuzzy yang dipakai, untuk menentukan derajat keanggotaan dari himpunan fuzzy yang digunakan diperlukan fungsi dari himpunan tersebut, fungsi ini dibuat berdasarkan persamaan dari garis yang membentuk himpunan fuzzy tersebut. Sebagai contoh, pada gambar 11 merupakan himpunan fuzzy segitiga dan pada persamaan 15 merupakan himpunan fuzzy segitiga.



Gambar 11 Himpunan Fuzzy Segitiga (Wahab, 2017)

$$\mu_A[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ \frac{x-a}{b-a}; & a < x < b \\ \frac{b-x}{b-a}; & b < x < a \\ 1; & x = b \end{cases} \quad (15)$$

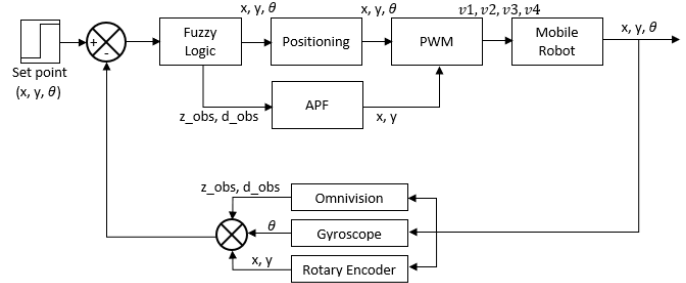
Dimana  $\mu_A[x]$  merupakan derajat keanggotaan, x adalah nilai dari variabel a, b, c yang mana secara berturut-turut merupakan nilai awal, tengah dan akhir.

**HASIL DAN PEMBAHASAN**

**Perancangan Sistem**

Sistem *obstacle avoidance* pada penelitian kali ini memanfaatkan metode *artificial potential field*. Seperti

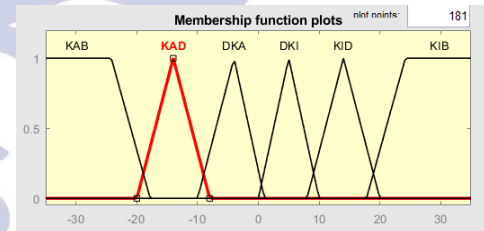
yang telah dibahas sebelumnya, metode ini memanfaatkan medan potensial buatan yang dibuat berdasarkan parameter-parameter yang dibutuhkan yang mana pada penelitian ini merupakan nilai jarak yang akan ditempuh oleh robot dan juga nilai jarak serta nilai sudut atau posisi dari *obstacle*.



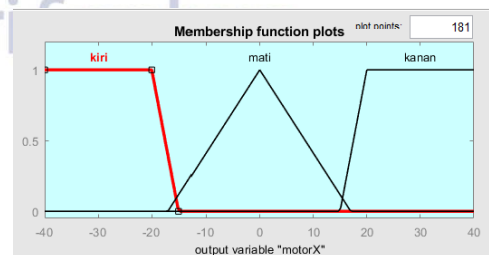
Gambar 12 Diagram Blok Sistem *Obstacle Avoidance*

Pada gambar 12 dijelaskan bahwa system *obstacle avoidance* ini memiliki 2 (dua) tahap yaitu *positioning* dan *obstacle avoidance* dengan metode APF, sesuai dengan parameter yang didapat fuzzy logic akan menentukan kapan ketika mobile robot melakukan *positioning* dan kapan mobile robot melakukan *obstacle avoidance*.

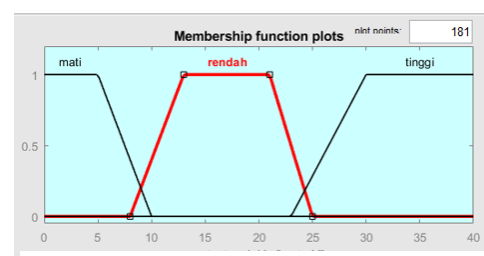
*Fuzzy logic controller (FLC)* juga akan berperan dalam menentukan posisi *obstacle* yang akan dihindari. Pada penelitian kali ini posisi *obstacle* akan dibedakan menjadi 6 (enam) jenis yaitu, depan kanan (DKA), depan kiri (DKI), kanan depan (KAD), kanan belakang (KAB), kiri depan (KAD) dan kiri belakang (KAB), keenam posisi *obstacle* tersebut sekaligus menjadi nama untuk fungsi keanggotaan dari FLC yang akan dirancang seperti pada gambar 13.



Gambar 13 Membership Fuction *Obstacle*



Gambar 14 Membership Motor X



Gambar 15 Membership Fuction Motor Y

Pada gambar 14 dan 15 merupakan output yang dihasilkan dari *fuzzy logic controller*. Yang dimana output dari motor x bernilai -40 sampai dengan 40 yang memiliki 3 (tiga) fungsi keanggotaan yaitu kanan, kiri dan mati dan motor y yang bernilai 0 sampai dengan 40 dengan fungsi keanggotaan yaitu mati, rendah dan tinggi. Output dari *fuzzy logic controller* ini akan membantu dalam menentukan batas kecepatan yang sesuai untuk digunakan berdasarkan posisi dari *obstacle* yang telah diketahui.

Ketika fungsi keanggotaan input dan output dari *fuzzy logic controller* telah dibuat, Langkah selanjutnya pada perancangan ini adalah membuat rule atau aturan. Rule ini akan memetakan fungsi-fungsi keanggotaan yang telah dibuat sesuai dengan aturan IF-THEN yang sesuai dengan kebutuhan, pada penelitian kali ini akan dibuat 6 (enam) macam rule, diantaranya.

1. IF (zObs is DKA) THEN (motorX is kiri) (motorY is mati)
2. IF (zObs is DKI) THEN (motorX is kanan) (motorY is mati)
3. IF (zObs is KAD) THEN (motorX is mati) (motorY is tinggi)
4. IF (zObs is KAB) THEN (motorX is mati) (motorY is rendah)
5. IF (zObs is KID) THEN (motorX is mati) (motorY is tinggi)
6. IF (zObs is KIB) THEN (motorX is mati) (motorY is rendah)

### Pengujian Sistem

Pada pengujian sistem kali ini robot akan diberikan diuji sebanyak 9 kali dengan 3 posisi start, 3 posisi finish dan 9 posisi *obstacle* yang berbeda beda. Hasil dari pengujian ini merupakan nilai dari posisi robot sebelum dan sesudah menghindari *obstacle*, nilai yang dihasilkan berupa titik koordinat robot yang memiliki format (x, y) dimana x adalah titik koordinat secara horizontal dan y adalah titik koordinat secara vertical yang memiliki satuan cm.

Tabel 1. Pengujian sistem *Obstacle Avoidance*

Percobaan Ke-	Titik start (x, y)	Posisi <i>obstacle</i> (x, y)	Sebelum <i>obstacle</i> (x, y)	Setelah <i>obstacle</i> (x, y)
1	0,0	0, 100	0, 51	0, 167
2	0,0	0, 250	0, 237	0, 345
3	0,0	0, 450	0, 423	0, 476
4	-150, 0	-100, 100	0, 50	0, 110
5	-150, 0	0, 250	0, 230	0, 350
6	-150, 0	100, 450	0, 400	0, 470
7	150, 0	100, 100	0, 70	0, 120
8	150, 0	0, 250	0, 223	0, 340
9	150, 0	-100, 450	0, 396	0, 457

Data pada tabel 1 akan diolah kembali untuk mencari nilai dari selisih dari posisi *obstacle* dengan nilai posisi robot saat akan menghindari, dengan cara mengurangi nilai koordinat *obstacle* dengan nilai koordinat sebelum *obstacle* serta mengurangi nilai koordinat setelah *obstacle* dengan nilai koordinat *obstacle*.

Maksud dari nilai sebelum *obstacle* yaitu nilai posisi koordinat robot sebelum menghindari *obstacle* ketika robot sedang berjalan seperti yang tercantum pada tabel 2.

Tabel 2. Nilai Selisih Sebelum *Obstacle*

Percobaan Ke-	Posisi <i>obstacle</i> (x, y)	Sebelum <i>obstacle</i> (x, y)	Selisih sebelum (cm)
1	0,100	0,51	49
2	0,250	0,237	63
3	0,450	0,423	27
4	-100,100	0,50	50
5	0,250	0,230	20
6	100,450	0,400	50
7	100,100	0,70	30
8	0,250	0,223	27
9	-100,450	0,396	54
Rata-rata			41. 11

Maksud dari nilai sebelum *obstacle* yaitu nilai posisi koordinat robot sebelum menghindari *obstacle* ketika robot sedang berjalan seperti yang tercantum pada tabel 3.

Tabel 3. Nilai Selisih Setelah *Obstacle*

Percobaan Ke-	Posisi <i>obstacle</i> (x, y)	Setelah <i>obstacle</i> (cm)	Selisih setelah (cm)
1	0, 100	0, 167	67
2	0, 250	0, 345	45
3	0, 450	0, 476	26
4	-100, 100	0, 110	10
5	0, 250	0, 350	100
6	100, 450	0, 470	20
7	100, 100	0, 120	20
8	0, 250	0, 340	90
9	-100, 450	0, 457	7
Rata-rata			42. 77

Dari tabel diatas dapat diketahui bahwa robot akan melakukan penghindaran terhadap *obstacle* rata-rata 41.11 cm sebelum *obstacle* dan selesai melakukan penghindaran rata-rata 42.77 cm setelah *obstacle*.

### PENUTUP

#### Kesimpulan

Hasil dari penelitian yang telah dilakukan yang telah dilakukan oleh peneliti dapat disimpulkan bahwa sistem

*obstacle avoidance* menggunakan metode *Artificial potential field (APF)* berbasis *Fuzzy logic controller (FLC)* dapat berkerja dengan cukup baik. Robot dapat menghindari rintangan yang ada dan dapat mencapai posisi tujuan yang diinginkan.

Pada pengujian *obstacle avoidance* robot dapat menghindari *obstacle* dengan jarak rata-rata 41.11 cm sebelum *obstacle* dan selesai melakukan penghindaran dengan jarak rata-rata 42.77 cm setelah *obstacle*.

### Saran

Berdasarkan penelitian yang telah dilakukan terdapat beberapa saran yang dapat dilakukan agar mendapatkan hasil yang lebih maksimal, diantaranya. Perlunya penambahan sistem kontrol lainnya seperti PI, PD, PID dan sebagainya untuk mendapatkan respon sistem yang lebih baik, kemudian Perlunya sistem pendeteksian *obstacle* yang lebih baik agar menimalisir noise atau gangguan dari objek lain.

### 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, September 2021, 547-555.
- Borenstein, J, dan L Feng. 1996. *Gyrodometry: A New Method for Combining Data from Gyros and Odometry in Mobile Robots*. Proceedings of the 1996 IEEE International Conference on Robotics and Automation. April 1996. Mineapolis, Minnesota.
- Dudek, Michael J.G. 2008. *Inertial Sensor, GPS, and Odometry*. Springer Berlin Heidelberg. pp. 477-490. Heidelberg, Jerman.
- Kasiram, Moh. 2008. *Metodologi Penelitian*, Malang: UIN-Malang Pers.
- Lee, Chuen Chien. 1990. *Fuzzy Logic Control System: Fuzzy logic controller, Part II*. IEEE Transcation On System, Man and Cybernetics. Volume 20 No. 02. April 1990.
- Li, Shih An, Li-Hsiang Chou, Tsung-Han Chang, Chao-Hsu Yang dan Yu-Cheng Chang. 2019. *Obstacle avoidance of mobile robot based on HyperOmni vision*. Sensors and Materials 31(3): 1021–36.
- Nurariyanto, kris. 2021. *Sistem Positioning Pada Four-Wheeled Omnidirectional Mobile Robot Menggunakan Metode Gyrodometry Berbasis PI-Fuzzy Controller*. Jurnal Teknik Elektro Volume 11 No. 01, Januari 2022, 77-87.

Wahab, Faisal; Sumardiono, Arif; Tahtawi, Adnan Rafi Al; Mulayari, Agus Faisal Aziz. 2017. *Desain dan Purwarupa Fuzzy Logic Control untuk Pengendalian Suhu Ruangan*. JTERA - Jurnal Teknologi Rekayasa; Vol.2 No.1, Juni 2017. p. 1-8.

Williams, Robert L., dan Jianhua Wu. 2010. *Dynamic Obstacle Avoidance for an Omnidirectional Mobile Robot*. Journal of Robotics 2010: 1–14.