



# **Kontes Robot Sepak Bola Beroda**

## **Tugas Akhir**

**Oleh:**

**Tajdar Hal Ata (4222001001)**

**Mochamad Rizal Fauzi (4222001010)**

**Haidar Afnan Fakhruddin (4222001026)**

**Program Studi Teknik Robotika**

**Jurusan Teknik Elektro**

**Politeknik Negeri Batam**

**2023**

# Pernyataan Keaslian Tugas Akhir

Saya yang bertandatangan dibawah ini menyatakan bahwa isi sebagian maupun keseluruhan Tugas Akhir saya yang berjudul : Augmented Monte Carlo Localization For Wheeled Soccer Robot”, “Object Detection Using Omnidirectional Camera” , “ Barelang 63 Basestation 2022”, adalah **hasil karya sendiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan, dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.** Semua referensi yang dikutip atau dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan saya ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Batam, 20 Desember 2023



---

Haidar Afnan Fakhruddin  
NIM: 4222001026



---

Mochamad Rizal Fauzi  
NIM: 4222001010



---

Tajdar Hal Ata  
NIM: 4222001001

# **Lembar Pengesahan**

**Tugas Akhir disusun untuk memenuhi salah satu syarat memperoleh gelar  
Sarjana Terapan Teknik (S.Tr.T)**

**di**

**Politeknik Negeri Batam**

**Oleh:**

**Tajdar Hal Ata**

**Dengan judul:**

**Penerapan Kontrol PID dalam Penggiring Bola pada Robot Sepak Bola Beroda**

**Tanggal Sidang: 21 12, 2023**

**Disetujui oleh :**

**Dosen Penguji I**



**1. Ir. Ahmad Riyad Firdaus, Ph.D  
NIK: 100013**

**Dosen Pembimbing**



**1. Hendawan Soebhakti, ST, MT  
NIK: 104031**

**Dosen Penguji II**



**2. Ryan Satria Wijaya, S.Tr.T., M.Tr.T.  
NIK: 121249**




# Lembar Pengesahan

Tugas Akhir disusun untuk memenuhi salah satu syarat memperoleh gelar  
Sarjana Terapan Teknik (S.Tr.T)  
di  
Politeknik Negeri Batam

Oleh:  
Haidar Afnan Fakhruddin (4222001026)

Dengan judul:  
Object Detection Using Omnidirectional Camera

Tanggal Sidang: 21 – 12 - 2023

Disetujui oleh :	
<p><b>Dosen Penguji I</b></p>  <p>1. (Ryan Satria Wijaya, S.Tr.T., M.Tr.T.) NIK: 121249</p>	<p><b>Dosen Pembimbing I</b></p>  <p>1. (Hendawan Soebhakti , S.T., M.T.) NIK: 104031</p>
<p><b>Dosen Penguji II</b></p>  <p>2. (Ahmad Riyad Firdaus, S.Si., M.T., Ph.D) NIK: 100013</p>	



# Lembar Pengesahan

Tugas Akhir disusun untuk memenuhi salah satu syarat memperoleh gelar  
Sarjana Terapan Teknik (S.Tr.T.)  
di  
Politeknik Negeri Batam

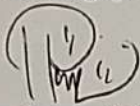
Oleh:  
Mochamad Rizal Fauzi (4222001010)

Dengan judul:  
Augmented Monte Carlo Localization For Wheeled Soccer Robot

Tanggal Sidang: 21-12-2023

Disetujui oleh :

Dosen Penguji I



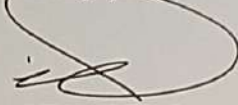
1. Ryan Satria Wijaya, S.Tr.T.,  
M.Tr.T.  
NIK: 121249

Dosen Pembimbing



1. Hendawan Soebhakti, S.T., M.T.  
NIK: 104031

Dosen Penguji II



2. Ahmad Riyad Firdaus, S.Si., M.T.,  
Ph. D  
NIK: 100013

# Penerapan Kontrol PID dalam Penggiring Bola pada Robot Sepak Bola Beroda

Tajdar Hal Ata<sup>1</sup>, Hendawan Soebhakti<sup>2</sup>

Jurusan Teknik Elektro, Politeknik Negeri Batam

Jalan Jend. Ahmad Yani, Kel. Tlk. Tering, Kec. Batam Kota, Kota Batam, Kepulauan Riau, 29461, Indonesia

Email: hendawan@polibatam.ac.id

**Abstrak**— Makalah ini membahas implementasi sistem kontrol dalam penggiring bola yang diterapkan untuk bersaing dalam kompetisi Liga Ukuran Menengah *RoboCup* dan Kontes Robot Indonesia. Desain visualisasi digunakan sebagai landasan untuk mengembangkan mekanisme penggiring bola yang sesuai dengan regulasi *RoboCup*. Mekanisme ini dirancang agar memiliki fleksibilitas optimal saat robot bergerak translasi. Pendekatan kontrol *Proportional Integral Derivative* (PID) diterapkan sebagai metode kontrol pada sistem penggiring, bertujuan untuk mengatur respons motor dribble dan menjaga bola tetap terkendali saat robot bergerak dengan kecepatan yang bervariasi atau dalam keadaan diam. Bagian hasil penelitian mengungkapkan respons sistem penggiring bola pada kecepatan mulai dari 20cm/s hingga 120cm/s pada gerak maju dan mundur, pada kecepatan 20cm/s sampai 60cm/s pada gerak *lateral*. Selain itu, dengan kecepatan 0.2rad/s hingga 1.5rad/s pada gerak berputar ke arah kanan dan kiri.

**Kata kunci**— PID kontrol, Penanganan Penggiring bola, *RoboCup*, Liga Ukuran Menengah

## I. PENDAHULUAN

*RoboCup* [1] adalah inisiatif internasional yang mengadakan kompetisi sepak bola robot untuk mendorong kemajuan dalam robotika dan kecerdasan buatan. Dalam kategori *Middle Size League* (MSL) [2], tim robot berukuran sedang bersaing untuk mengembangkan teknologi otonomi dan strategi sepak bola yang canggih. Sedangkan secara nasional terdapat kompetisi yang diselenggarakan oleh puspernas dengan kategori robot sepak bola beroda Indonesia (KRSBI-B) [3]. Dalam *RoboCup Middle Size League* (MSL) dan kontes robot Indonesia, aturan terkait menggiring bola secara alami menegaskan bahwa ketika robot sedang menggiring bola [4], ukuran bola yang dimasukkan tidak boleh melebihi sepertiga dari ukuran robot. Selain itu, selama pergerakan robot seperti maju, mundur, kanan, dan kiri, aturan menuntut agar bola berputar dengan cara yang mencerminkan pergerakan yang alami. Dalam konteks ini, perancangan penggiring dan dengan sistem kontrol *Proportional Integral Derivative* (PID) [5]–[7].

Mempertimbangkan sulitnya menentukan gerakan bola ketika berada dalam kontak terus menerus dengan robot, berbagai mekanisme dan sistem kontrol penggiring bola telah diterapkan oleh tim-tim lain dalam penyelesaian masalah ini [8], [9], [10], [11]. Beberapa di antaranya mengadopsi penggunaan motor di bagian depan, mendorong bola dengan tubuh robot [12], dan memanfaatkan jaring besi robot untuk mengontrol bola. Tim Tech United [13], [14] dan Team Hibikino [15], sebagai contoh, memanfaatkan model non-linear sebagai umpan balik dan kontrol umpan maju untuk menjaga jarak antara bola dan robot. Pendekatan ini melibatkan pengukuran sudut lengan dan penyesuaian posisi sesuai dengan jarak antara bola dan robot.

Dalam konteks karya ilmiah ini, diusulkan suatu strategi kontrol bola yang melibatkan pemanfaatan dua roda aktif yang diposisikan di bagian depan robot sepak bola. Sistem ini dirancang dengan cermat untuk fokus pada penciptaan gesekan di sisi atas bola. Pendekatan ini melibatkan penerapan kontrol

PID terkait gerakan robot dan dua roda aktif, serta penyesuaian dengan aturan yang mengatur menggiring bola dalam kompetisi *RoboCup Middle Size League* (MSL) dan Kompetisi Robot Indonesia (KRSBI-B).

## II. METODE

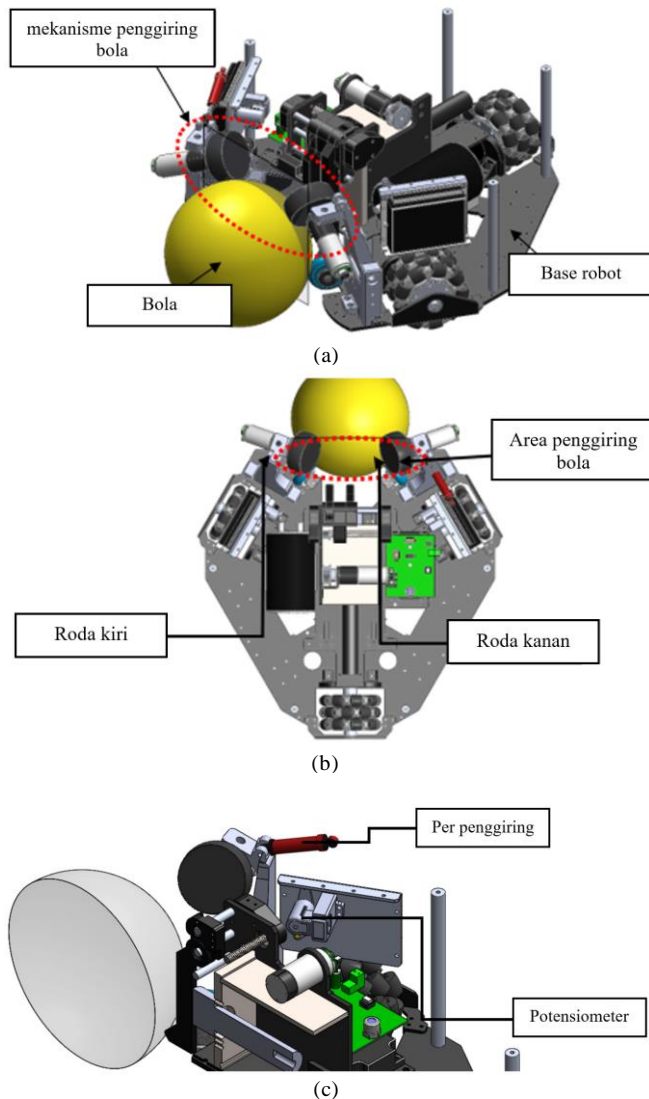
Penelitian ini dimulai dengan perancangan mekanisme penggiring bola secara mekanik yang mematuhi peraturan *RoboCup Middle Size League*. Selanjutnya, kami menerapkan pengendalian *Proportional Integral Derivative* (PID) untuk menjaga kecepatan target pada motor penggiring bola. Pendekatan ini bertujuan untuk mencegah terlepasnya bola saat robot sedang dalam proses menggiring bola.

### A. Pengembangan Mekanisme Penangan Bola

Banyak mekanisme penanganan bola untuk robot sepak bola telah didokumentasikan dalam literatur, terutama terbagi dalam dua kategori: satu melibatkan penggunaan dua roda aktif yang diposisikan di bagian depan robot, sementara yang lain mengandalkan penggunaan lengan robot. Seiring dengan berkembangnya pemilihan robot sepak bola, pendekatan yang menggunakan dua roda aktif telah diterima secara luas. Keuntungan utama dari mekanisme penanganan bola yang menggunakan roda aktif ini adalah kemampuannya untuk mempertahankan kontrol bola bahkan selama gerakan maju, mundur, kanan, dan kiri. Namun, hal ini memiliki kekurangan, termasuk kerumitan desain dan kebutuhan akan algoritma kontrol gerakan bola yang selaras dengan gerakan robot. Rotasi bola yang mulus dalam kaitannya dengan arah gerak robot sangat penting untuk mematuhi peraturan *RoboCup* dan untuk memastikan pergerakan robot yang mulus. Selain itu, mekanisme penanganan bola ini terbatas pada kemampuannya untuk mencakup hanya sepertiga permukaan bola.

Desain mekanisme penanganan bola ditunjukkan pada Gambar 1 (a) yaitu tampilan isometris, (b) tampilan atas, dan (c) tampilan setengah. Mekanisme ini menggunakan dua roda

aktif yang terpasang pada tuas untuk memutar bola. Tuas bergerak ke atas secara pasif ketika roda aktif berputar, membawa bola ke arah tubuh robot. Pegas dan peredam dihubungkan antara tubuh dan tuas untuk menyerap dampak tabrakan bola. Omni-roda kecil di bagian bawah mekanisme bertindak sebagai rol pasif untuk mencegah bola menyentuh tubuh.



Gambar 6. (a) yaitu tampilan isometris (b) tampilan atas (c) tampilan setengah

### B. Pengontrol *Proportional Integral Derivative* (PID)

Kontrol PID adalah sistem kontrol yang menggunakan umpan balik untuk menjaga variabel proses agar sesuai dengan target yang diinginkan. Kontrol PID bekerja dengan menghitung nilai error sebagai selisih antara target yang diinginkan dan variabel proses yang diukur. Untuk mendapatkan kontrol yang baik, setiap parameter dari PID harus diatur agar sistem kontrol dapat bekerja dengan stabil dan presisi. Maka dari itu diberikan persamaan kontrol :

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \quad (1)$$

Pada persamaan 1 dijelaskan sebagai berikut:

- $u(t)$  adalah penggerak yang berasal dari Pengendali, ke dalam Proses, pada waktu  $t$
- $e(t) = y_{sp}(t) - y(t)$  adalah selisih antara setpoint dan variabel proses yang diukur pada waktu  $t$

- $K_p, K_i, K_d$  masing-masing adalah konstanta  $P, I$ , dan  $D$

Metode kontrol PID digunakan dalam sistem pengendalian kecepatan motor penggiring, yang berperan sebagai sistem penggiring bola pada saat robot menggiring bola. Sesuai dengan aturan permainan yang menetapkan agar bola tidak berputar saat robot berhenti, implementasi PID sebagai sistem pengendalian penggiring digunakan untuk mengatur kecepatan motor berdasarkan sudut pembacaan sensor potensiometer. PID diterapkan pada setiap motor penggiring bola sehingga respons masing-masing motor dapat disesuaikan dengan variasi pembacaan sudut. Rincian kondisi robot yang berisikan nilai error sensor dan arah putaran roda pada bagian kanan dan kiri dapat ditemukan pada tabel 1 di bawah ini.

TABEL I. KONDISI ROBOT

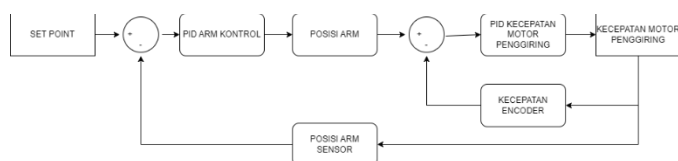
Kondisi Robot	Nilai Error Sensor Potensio Kiri	Nilai Error Sensor Potensio Kanan	Arah Putaran Roda Kiri	Arah Putaran Roda Kanan
Diam Saat Memegang Bola	$= 0$	$= 0$	Stop	Stop
Bergerak Maju	$> 0$	$> 0$	CCW	CW
Bergerak Mundur	$< 0$	$< 0$	CW	CCW
Bergerak dan Berputar Kekiri	$> 0$	$< 0$	CW	CW
Bergerak dan Berputar Kekan	$< 0$	$> 0$	CCW	CCW

### C. Sistem Kontrol

Bagian ini akan menguraikan mengenai sistem pengendalian penggiring bola pada robot sepak bola yang melibatkan pengendalian kecepatan pada dua motor dribble. Sensor sudut dalam bentuk potensiometer digunakan untuk membaca sudut, dengan rentang pembacaan aktual antara 0 hingga 23 derajat. Nilai-nilai sudut ini kemudian di-mapping menjadi rentang nilai antara -70 hingga 70 derajat. Nilai setpoint pada PID pertama ditetapkan pada 0, yang merupakan nilai tengah dari rentang potensiometer, untuk mengontrol posisi arm. Angka ini dianggap sebagai nilai optimal untuk menjaga bola pada posisi yang diinginkan.

Ketika nilai error dari PID arm mencapai atau mendekati -70, atau kurang dari 0, hal ini mengindikasikan bahwa posisi arm terlalu rendah. Dalam situasi tersebut, dribble harus menarik bola untuk mengurangi nilai error dan mengembalikan keseimbangan. Sebaliknya, jika nilai error melebihi 70, menunjukkan posisi arm terlalu tinggi, dan pada kasus tersebut, penggiring harus melepas bola untuk mengurangi nilai error dan mengembalikan posisi arm ke nilai 0. Dengan pendekatan ini, sistem PID dan kendali potensiometer digunakan untuk menjaga posisi arm pada nilai setpoint, yaitu 0, dan berusaha menjaga bola pada posisi yang diinginkan. Pada Gambar 4

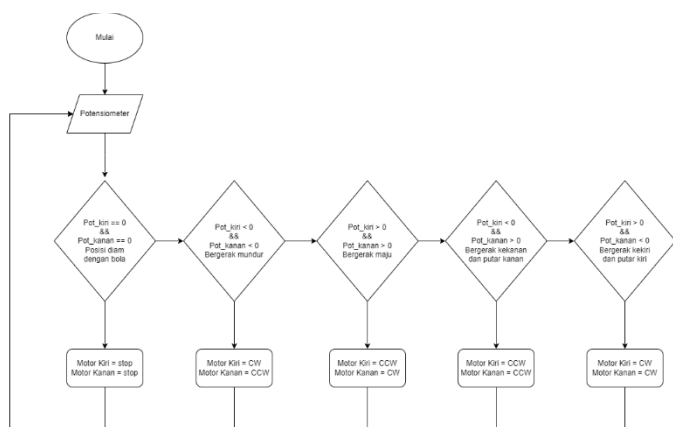
menunjukkan sistem kontrol penggiring bola pada robot sepak bola.



Gambar 4. Sistem kontrol penggiring

Penelitian ini menerapkan dua metode PID dalam sistem kontrol penggiring pada robot. Metode pertama adalah PID posisi yang berfungsi untuk mengontrol sudut sensor dengan nilai setpoint sejajardengan 0. Output dari PID posisi menjadi setpoint bagi metode kedua, yaitu PID kecepatan. PID kecepatan digunakan untuk mengendalikan kecepatan motor agar mendekati nilai setpoint tersebut. Saat robot dalam keadaan diam dan tidak memegang bola, nilai potensiometer menghasilkan angka negatif. Output dari PID posisi juga menjadi negatif, sehingga motor kanan berputar berlawanan arah jarum jam (CCW), sedangkan motor kiri berputar searah jarum jam (CW), mengakibatkan bola tertarik ke arah robot.

Pada tahap ketika robot berada dalam keadaan diam dan berhasil menangkap bola, motor penggiring aktif untuk menarik bola, mengakibatkan peningkatan sudut pada potensiometer. Setelah mencapai sudut nol, output dari PID posisi akan mencapai nilai 0. Hal ini mengakibatkan kedua motor berhenti berputar dan menjaga bola pada sudut tertentu. Selanjutnya, saat robot bergerak maju dengan bola terpegang, motor penggiring mendorong bola ke arah robot, menghasilkan nilai potensiometer yang lebih besar dari 0. Motor kanan berputar searah jarum jam (CW), sementara motor kiri berputar berlawanan arah jarum jam (CCW), sehingga bola berputar sesuai dengan arah pergerakan robot.



Gambar 5. Perhitungan untuk penggiring berdasarkan nilai eror sensor Potensiometer

Pada Gambar 5 mengindikasikan hubungan proporsional antara nilai output dari PID posisi dengan kecepatan motor. Dalam konteks ini, semakin tinggi nilai output PID posisi, maka kecepatan motor juga cenderung meningkat. Fenomena ini mencerminkan keterkaitan langsung antara pengaturan PID posisi dan respons kecepatan motor pada sistem penggiring, memberikan pemahaman lebih lanjut terhadap dinamika sistem kontrol yang diterapkan pada robot.

### III. HASIL

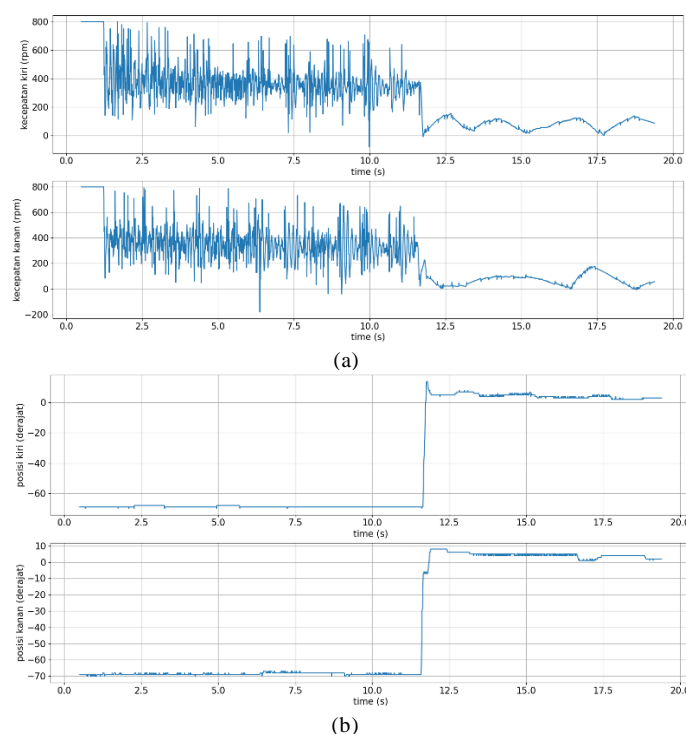
Bagian ini melaporkan hasil uji coba kontrol penggiring bola dengan variasi kecepatan mulai dari 20cm/s hingga 120cm/s saat robot bergerak maju dan mundur, pada gerakan lateral, kecepatan diuji mulai dari 20cm/s hingga 60cm/s.

Selain itu, pada gerakan berputar ditempat ke arah kiri dan kanan, kecepatan diuji mulai dari 0.2rad/s hingga 1.5rad/s. Pengujian mencakup berbagai arah gerakan seperti maju, mundur, kanan, kiri, dan berputar. Pengamatan dilakukan terhadap kecepatan motor penggiring dan nilai sensor yang merefleksikan hasil uji pada setiap pergerakan robot.

#### A. Pengujian saat menerima bola

Uji coba ini melibatkan skenario di mana robot menerima bola. Proses pengujian dilakukan dengan robot berada dalam keadaan diam, sementara bola dilemparkan ke arah robot. motor penggerak kanan menanggapi dengan berputar berlawanan arah jarum jam (CCW), sementara motor penggerak kiri berputar searah jarum jam (CW). Seperti yang dijelaskan sebelumnya sensor sudut dalam bentuk potensiometer digunakan untuk membaca sudut, dengan rentang pembacaan aktual antara 0 hingga 23 derajat. Nilai-nilai sudut ini kemudian di-mapping menjadi rentang nilai antara -70 hingga 70 derajat. Nilai setpoint pada PID pertama ditetapkan pada 0, yang merupakan nilai tengah dari rentang potensiometer, untuk mengontrol posisi arm. Angka ini dianggap sebagai nilai optimal untuk menjaga bola pada posisi yang diinginkan.

Setiap peningkatan nilai sudut mengakibatkan pengurangan nilai RPM pada motor penggiring, sesuai dengan panduan yang telah dijelaskan sebelumnya dan sesuai dengan regulasi RoboCup yang menetapkan bahwa bola tidak boleh terus berputar saat robot menangkap bola untuk menghindari pelanggaran aturan. Hasil pengujian merefleksikan nilai dari setiap sensor dan kecepatan motor penggiring. Ketika robot belum menangkap bola, sensor menunjukkan nilai -70, dan saat robot mulai menangkap bola, sensor menunjukkan peningkatan nilai yang signifikan. Sebagai respons terhadap peningkatan ini, motor penggiring kanan dan kiri menunjukkan kenaikan nilai RPM seiring dengan kenaikan nilai yang diberikan oleh sensor potensiometer. Oleh karena itu, implementasi PID dalam sistem ini bertujuan untuk mengurangi kesalahan dari kecepatan penggiring. Data hasil pengujian ditampilkan sebagai berikut.



Gambar 6. (a) respon kecepatan motor penggiring (b) respon sudut lengan motor penggiring



Dari analisis Gambar 6 (a) menampilkan data respon kecepatan motor penggiring, sedangkan (b) menampilkan data respon sudut lengan motor penggiring. Dapat dilihat bahwa setiap pergerakan robot mengacu pada nilai set point lengan yang telah ditentukan, yakni 0 derajat. Observasi terhadap posisi lengan yang berada di bawah 0 menunjukkan upaya robot dalam menarik dan menahan bola dengan kecepatan motor agar kondisi sepertiga bola sempurna masuk kedalam robot, dengan tujuan menjaga lengan robot berada pada titik set point yang telah ditetapkan sebelumnya.

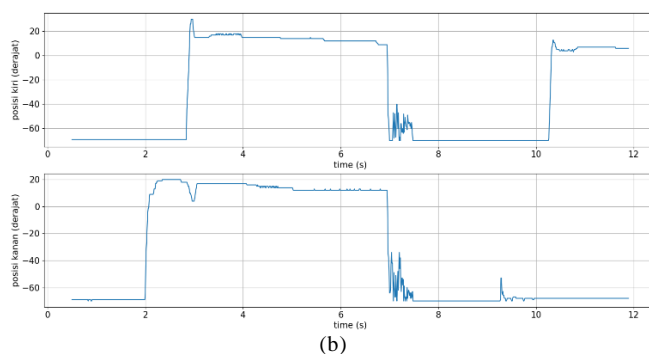
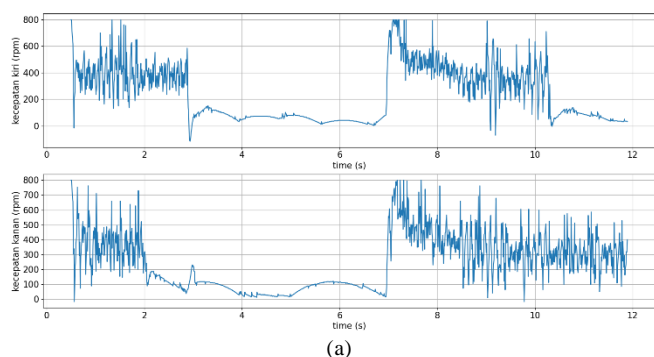
#### B. Pengujian saat menggiring bola ke belakang

Uji coba ini melibatkan skenario di mana robot melakukan gerakan mundur setelah menerima bola. Proses pengujian dilaksanakan dengan robot bergerak ke belakang dengan rentang kecepatan antara 20 cm/s hingga 120 cm/s. Pada saat robot mengarahkan bola ke belakang, motor penggerak kanan menanggapi dengan berputar berlawanan arah jarum jam (CCW), sementara motor penggerak kiri berputar searah jarum jam (CW). Evaluasi kesuksesan pengujian terdokumentasi dalam Tabel 2, yang memuat informasi mengenai berhasil atau tidaknya robot menggiring secara natural.

TABEL II. HASIL UJI COBA

Kecepatan Robot	Menggiring Secara Natural
20cm/s	Berhasil
40cm/s	Berhasil
60cm/s	Berhasil
80cm/s	Berhasil
100cm/s	Berhasil
120cm/s	Berhasil
140cm/s	Gagal

Pada tabel 2 di atas menyajikan hasil dari berbagai kecepatan yang telah diuji, sementara pada Gambar 7 tergambar data kecepatan maksimum yaitu 120cm/s yang berhasil dicapai oleh robot. Data tersebut mencakup kecepatan motor penggiring (rpm) dan posisi (derajat), serta informasi mengenai waktu yang diperoleh (detik). Sebagaimana tergambar berikut ini.



Gambar 7. (a) respon kecepatan motor penggiring (b) respon sudut lengan motor penggiring

Dari analisis Gambar 7 (a) menampilkan data respon kecepatan motor penggiring, sedangkan (b) menampilkan data respon sudut lengan motor penggiring. Dapat dilihat bahwa setiap pergerakan robot mengacu pada nilai set point lengan yang telah ditentukan, yakni 0 derajat. Observasi terhadap posisi lengan yang berada di bawah 0 menunjukkan upaya robot dalam menarik bola dengan kecepatan motor yang menyesuaikan dengan kecepatan robot, dengan tujuan menjaga lengan robot berada pada titik set point yang telah ditetapkan sebelumnya.

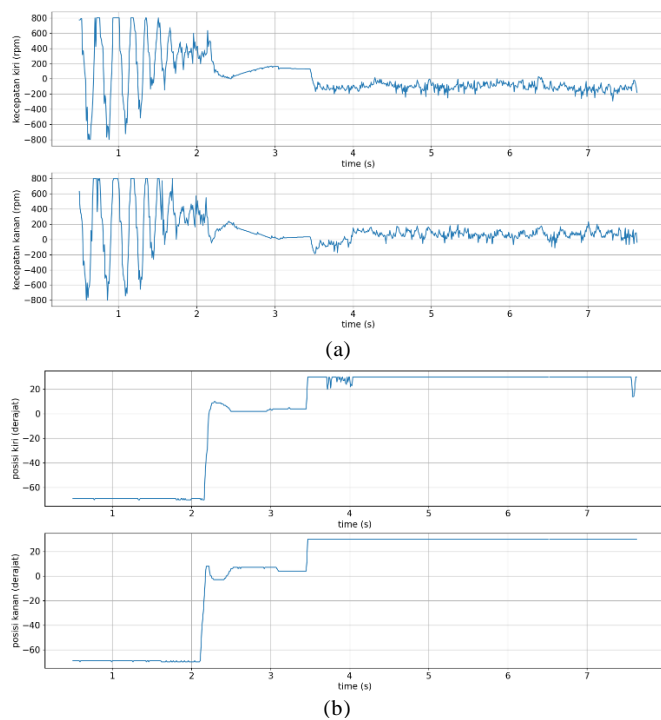
#### C. Pengujian saat menggiring bola ke depan

Uji coba ini melibatkan skenario di mana robot melakukan gerakan maju setelah menerima bola. Proses pengujian dilaksanakan dengan robot bergerak ke depan dengan rentang kecepatan antara 20 cm/s hingga 120 cm/s. Pada saat robot mengarahkan bola ke depan, motor penggerak kanan berputar searah jarum jam (CW), sementara motor penggerak kiri menanggapi dengan berputar berlawanan arah jarum jam (CCW). Evaluasi kesuksesan pengujian terdokumentasi dalam Tabel 3, yang memuat informasi mengenai berhasil atau tidaknya robot menggiring secara natural.

TABEL III. HASIL UJI COBA

Kecepatan Robot	Menggiring Secara Natural
20cm/s	Berhasil
40cm/s	Berhasil
60cm/s	Berhasil
80cm/s	Berhasil
100cm/s	Berhasil
120cm/s	Berhasil
140cm/s	Gagal

Pada tabel 3 di atas menyajikan hasil dari berbagai kecepatan yang telah diuji, sementara pada Gambar 8 tergambar data kecepatan maksimum yaitu 120cm/s yang berhasil dicapai oleh robot. Data tersebut mencakup kecepatan motor penggiring (rpm) dan posisi (derajat), serta informasi mengenai waktu yang diperoleh (detik). Sebagaimana tergambar berikut ini.



Gambar 8. (a) respon kecepatan motor penggiring (b) respon sudut lengan motor penggiring

Dari analisis Gambar 8 (a) menampilkan data respon kecepatan motor penggiring, sedangkan (b) menampilkan data respon sudut lengan motor penggiring. Dapat dilihat bahwa setiap pergerakan robot mengacu pada nilai set point lengan yang telah ditentukan, yakni 0 derajat. Observasi terhadap posisi lengan yang berada di bawah 0 menunjukkan upaya robot dalam mendorong bola dengan kecepatan motor yang menyesuaikan dengan kecepatan robot, dengan tujuan menjaga lengan robot berada pada titik set point yang telah ditetapkan sebelumnya.

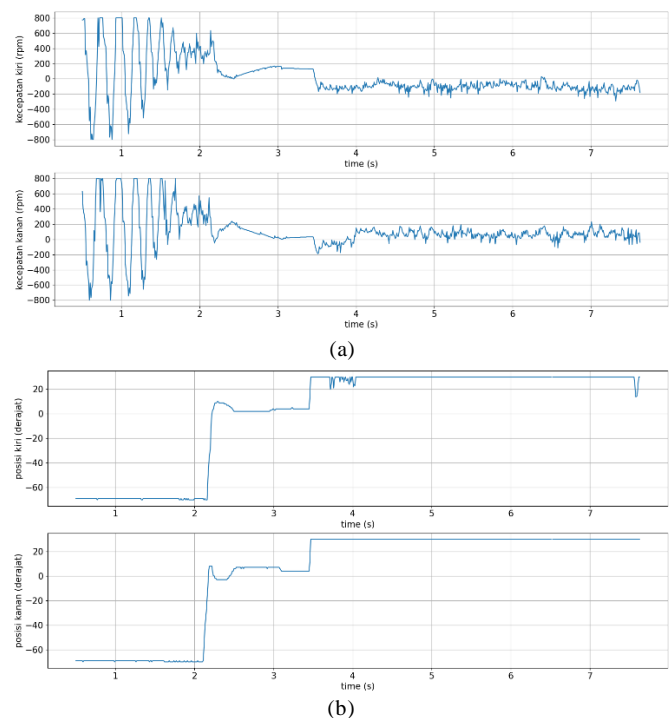
#### D. Pengujian saat menggiring bola ke kanan

Uji coba ini melibatkan skenario di mana robot melakukan gerakan *lateral* setelah menerima bola. Proses pengujian dilaksanakan dengan robot bergerak ke kanan dengan rentang kecepatan antara 20 cm/s hingga 60 cm/s. Pada saat robot mengarahkan bola ke kanan, motor penggerak kanan dan kiri berputar berlawanan dengan arah jarum jam (CCW). Evaluasi kesuksesan pengujian terdokumentasi dalam Tabel 4, yang memuat informasi mengenai berhasil atau tidaknya robot menggiring secara natural.

TABEL IV. HASIL UJI COBA

Kecepatan Robot	Menggiring Secara Natural
20cm/s	Berhasil
40cm/s	Berhasil
60cm/s	Berhasil
80cm/s	Gagal

Pada tabel 4 di atas menyajikan hasil dari berbagai kecepatan yang telah diuji, sementara pada Gambar 9 tergambar data kecepatan maksimum yaitu 60cm/s yang berhasil dicapai oleh robot. Data tersebut mencakup kecepatan motor penggiring (rpm) dan posisi (derajat), serta informasi mengenai waktu yang diperoleh (detik). Sebagaimana tergambar berikut ini.



Gambar 9. (a) respon kecepatan motor penggiring (b) respon sudut lengan motor penggiring

Dari analisis Gambar 9 (a) menampilkan data respon kecepatan motor penggiring, sedangkan (b) menampilkan data respon sudut lengan motor penggiring. Dapat dilihat bahwa setiap pergerakan robot mengacu pada nilai set point lengan yang telah ditentukan, yakni 0 derajat. Observasi terhadap posisi lengan yang berada di bawah 0 menunjukkan upaya robot dalam mengontrol bola dengan kecepatan yang menyesuaikan dengan kecepatan robot, di mana motor kanan bertanggung jawab untuk menarik bola sementara motor kiri melakukan dorongan. Dengan tujuan menjaga lengan robot berada pada titik set point yang telah ditetapkan sebelumnya.

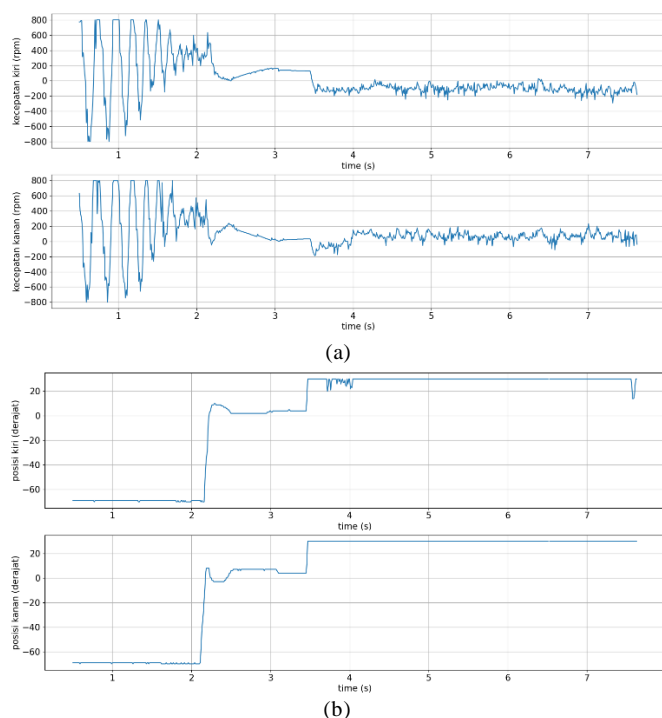
#### E. Pengujian saat menggiring bola ke kiri

Uji coba ini melibatkan skenario di mana robot melakukan gerakan *lateral* setelah menerima bola. Proses pengujian dilaksanakan dengan robot bergerak ke kiri dengan rentang kecepatan antara 20 cm/s hingga 60 cm/s. Pada saat robot mengarahkan bola ke kanan, motor penggerak kanan dan kiri berputar searah dengan arah jarum jam (CW). Evaluasi kesuksesan pengujian terdokumentasi dalam Tabel 5, yang memuat informasi mengenai berhasil atau tidaknya robot menggiring secara natural.

TABEL V. HASIL UJI COBA

Kecepatan Robot	Menggiring Secara Natural
20cm/s	Berhasil
40cm/s	Berhasil
60cm/s	Berhasil
80cm/s	Gagal

Pada tabel 5 di atas menyajikan hasil dari berbagai kecepatan yang telah diuji, sementara pada Gambar 10 tergambar data kecepatan maksimum yaitu 60cm/s yang berhasil dicapai oleh robot. Data tersebut mencakup kecepatan motor penggiring (rpm) dan posisi (derajat), serta informasi mengenai waktu yang diperoleh (detik). Sebagaimana tergambar berikut ini.



Gambar 10. (a) respon kecepatan motor penggiring (b) respon sudut lengan motor penggiring

Dari analisis Gambar 10 (a) menampilkan data respon kecepatan motor penggiring, sedangkan (b) menampilkan data respon sudut lengan motor penggiring. Dapat dilihat bahwa setiap pergerakan robot mengacu pada nilai set point lengan yang telah ditentukan, yakni 0 derajat. Observasi terhadap posisi lengan yang berada di bawah 0 menunjukkan upaya robot dalam mengontrol bola dengan kecepatan yang menyesuaikan dengan kecepatan robot, di mana motor kiri bertanggung jawab untuk menarik bola sementara motor kanan melakukan dorongan. Dengan tujuan menjaga lengan robot berada pada titik set point yang telah ditetapkan sebelumnya.

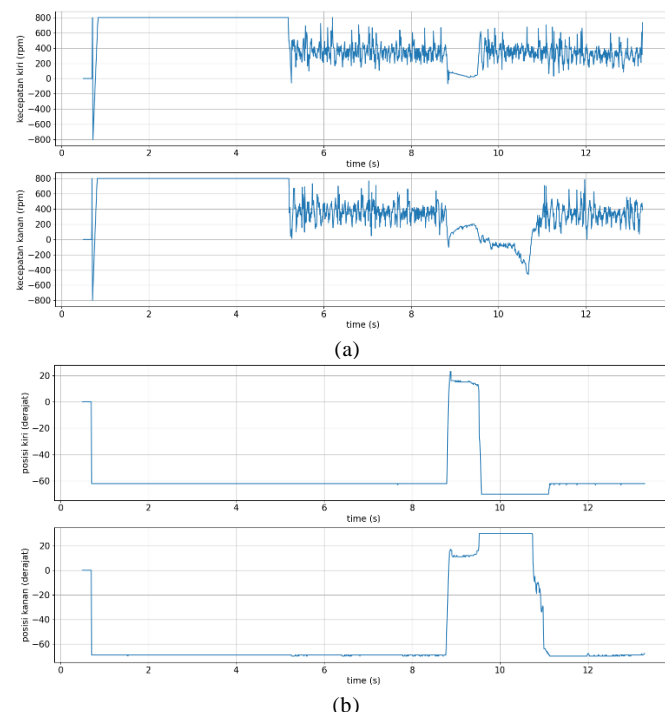
#### F. Pengujian saat memutar bola ke kanan

Uji coba ini melibatkan skenario di mana robot melakukan gerakan berputar ditempat ke arah kanan setelah menerima bola. Proses pengujian dilaksanakan dengan robot berputar ditempat ke arah kanan dengan rentang kecepatan antara 0.2rad/s hingga 1.5rad/s. Pada saat robot memutar bola ke kanan, motor penggerak kanan dan kiri berputar berlawanan dengan arah jarum jam (CCW). Evaluasi kesuksesan pengujian terdokumentasi dalam Tabel 6, yang memuat informasi mengenai berhasil atau tidaknya robot menggiring secara natural.

TABEL VI. HASIL UJI COBA

Kecepatan Robot	Menggiring Secara Natural
0.2rad/s	Berhasil
0.4rad/s	Berhasil
0.6rad/s	Berhasil
0.8rad/s	Berhasil
1.0rad/s	Berhasil
1.2rad/s	Berhasil
1.5rad/s	Berhasil
1.6rad/s	Gagal

Pada tabel 6 di atas menyajikan hasil dari berbagai kecepatan yang telah diuji, sementara pada Gambar 11 tergambar data kecepatan maksimum yaitu 1.5rad/s yang berhasil dicapai oleh robot. Data tersebut mencakup kecepatan motor penggiring (rpm) dan posisi (derajat), serta informasi mengenai waktu yang diperoleh (detik). Sebagaimana tergambar berikut ini.



Gambar 11. (a) respon kecepatan motor penggiring (b) respon sudut lengan motor penggiring

Dari analisis Gambar 11 (a) menampilkan data respon kecepatan motor penggiring, sedangkan (b) menampilkan data respon sudut lengan motor penggiring. Dapat dilihat bahwa setiap pergerakan robot mengacu pada nilai set point lengan yang telah ditentukan, yakni 0 derajat. Observasi terhadap posisi lengan yang berada di bawah 0 menunjukkan upaya robot dalam mengontrol bola dengan kecepatan yang menyesuaikan dengan kecepatan robot, di mana motor kanan bertanggung jawab untuk menarik bola sementara motor kiri melakukan dorongan. Dengan tujuan menjaga lengan robot berada pada titik set point yang telah ditetapkan sebelumnya.

#### G. Pengujian saat memutar bola ke kiri

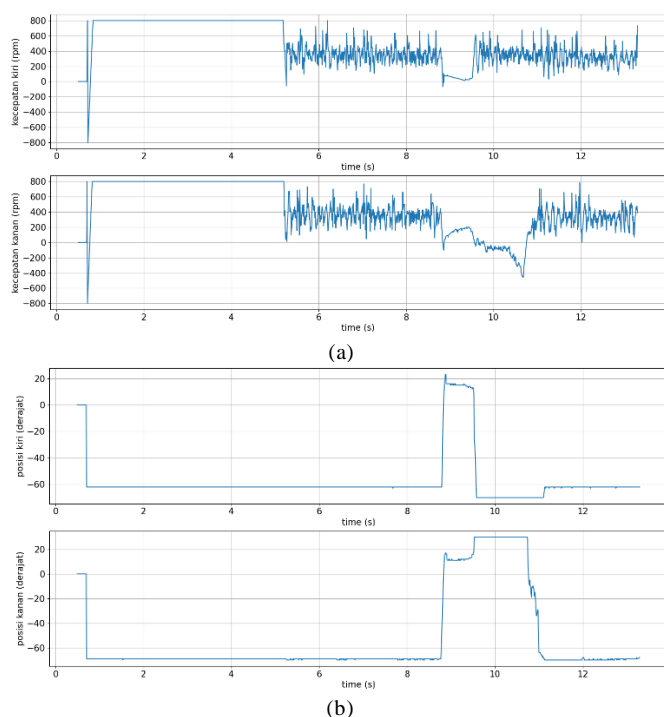
Uji coba ini melibatkan skenario di mana robot melakukan gerakan berputar ditempat ke arah kiri setelah menerima bola. Proses pengujian dilaksanakan dengan robot berputar ke arah kiri dengan rentang kecepatan antara 0.2rad/s hingga 1.5rad/s. Pada saat robot memutar bola ke kiri, motor penggerak kanan dan kiri berputar berlawanan dengan arah jarum jam (CCW). Evaluasi kesuksesan pengujian terdokumentasi dalam Tabel 7, yang memuat informasi mengenai berhasil atau tidaknya robot menggiring secara natural.

TABEL VII. HASIL UJI COBA

Kecepatan Robot	Menggiring Secara Natural
0.2rad/s	Berhasil
0.4rad/s	Berhasil
0.6rad/s	Berhasil
0.8rad/s	Berhasil

Kecepatan Robot	Menggiring Secara Natural
1.0rad/s	Berhasil
1.2rad/s	Berhasil
1.5rad/s	Berhasil
1.6rad/s	Gagal

Pada tabel 7 di atas menyajikan hasil dari berbagai kecepatan yang telah diuji, sementara pada Gambar 12 tergambar data kecepatan maksimum yaitu 1.5rad/s yang berhasil dicapai oleh robot. Data tersebut mencakup kecepatan motor penggiring (rpm) dan posisi (derajat), serta informasi mengenai waktu yang diperoleh (detik). Sebagaimana tergambarakan berikut ini.



Gambar 12. (a) respon kecepatan motor penggiring (b) respon sudut lengan motor penggiring

Dari analisis Gambar 12 (a) menampilkan data respon kecepatan motor penggiring, sedangkan (b) menampilkan data respon sudut lengan motor penggiring. Dapat dilihat bahwa setiap pergerakan robot mengacu pada nilai set point lengan yang telah ditentukan, yakni 0 derajat. Observasi terhadap posisi lengan yang berada di bawah 0 menunjukkan upaya robot dalam mengontrol bola dengan kecepatan yang menyesuaikan dengan kecepatan robot, dimana motor kiri bertanggung jawab untuk menarik bola sementara motor kanan melakukan dorongan. Dengan tujuan menjaga lengan robot berada pada titik set point yang telah ditetapkan sebelumnya.

#### IV. KESIMPULAN

Berdasarkan hasil penelitian ini, dapat ditarik kesimpulan bahwa mekanisme perancangan sistem dribble berperan besar dalam mencapai putaran bola secara alami. Keberhasilan sistem ini sangat tergantung pada penyesuaian diameter bola, karena ketidaksesuaian dapat menyebabkan bola mudah terlepas. Temuan penelitian juga menunjukkan bahwa menggiring bola dengan bergerak maju, mundur, ke kanan, ke kiri, dan berputar menggunakan kontrol PID sebagai pengatur kecepatan motor penggiring masih menghadapi tantangan stabilitas pada kecepatan robot yang beragam dalam setiap arah gerak.

Tantangan tersebut terkait dengan peningkatan suhu motor dribble ketika bola tertahan dalam keadaan tertekan selama proses penggiringan.

Peran roda penggiring bola dalam eksperimen ini juga menonjol. Pada percobaan ketika robot menggiring bola mundur dan maju dengan kecepatan 140 cm/s, serta menggiring bola ke kanan dan kiri dengan kecepatan 80 cm/s, bola langsung terlepas pada awal penggiringan, hal ini disebabkan oleh batasan penapakan roda penggiring dengan permukaan bola. Oleh karena itu, dalam penelitian berikutnya, direkomendasikan untuk mengembangkan sistem kontrol yang lebih stabil agar suhu motor dribble dapat dijaga agar tidak cepat panas. Selain itu, perlu dilakukan modifikasi pada permukaan roda agar lebih lebar, sehingga dapat meningkatkan kontak dan kekuatan gesek dengan bola, sehingga mampu menyesuaikan diri dengan kecepatan gerak robot yang lebih tinggi.

#### UCAPAN TERIMA KASIH

Terima kasih kepada tim BARELANG63 dari Politeknik Negeri Batam (POLIBATAM) atas dukungan penuh dalam penelitian ini. BARELANG63, sebagai tim yang telah berdedikasi dalam pengembangan robot Middle-Size League (MSL) beroda sejak tahun 2017, memberikan kontribusi yang sangat berharga bagi kemajuan penelitian ini. Keberhasilan dan capaian dalam penelitian ini tidak terlepas dari pemimpin proyek yaitu bapak Hendawan Soebhakti, kolaborasi dan dedikasi tim BARELANG63. Terima kasih atas upaya bersama dan dedikasi tinggi yang telah diberikan untuk mencapai hasil yang memuaskan.


#### REFERENSI

- [1] robocup.federation, "A Brief History of RoboCup." Accessed: Dec. 01, 2023. [Online]. Available: [https://www.robocup.org/a\\_brief\\_history\\_of\\_robocup](https://www.robocup.org/a_brief_history_of_robocup)
- [2] "Middle Size League (MSL)." Accessed: Dec. 01, 2023. [Online]. Available: <https://msl.robocup.org/>
- [3] B. Pengembangan Talenta, I. Pusat, P. Nasional, K. Pendidikan, and D. Teknologi, "BUKU PEDOMAN KONTES ROBOT INDONESIA (KRI) TAHUN 2023."
- [4] M. Asada *et al.*, "Middle Size Robot League Rules and Regulations for 2023," 2023.
- [5] A. Abdulameer, M. Sulaiman, M. S. M. Aras, and D. Saleem, "Tuning methods of PID controller for DC motor speed control," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 3, no. 2, pp. 343–349, Aug. 2016, doi: 10.11591/ijeecs.v3.i2.pp343-349.
- [6] Institute of Electrical and Electronics Engineers, *MilCIS 2014 : 2014 Military Communications and Information Systems Conference (MilCIS) : proceedings, 11-13 November, 2014, Canberra, Australia*.
- [7] R. P. Borase, D. K. Maghade, S. Y. Sondkar, and S. N. Pawar, "A review of PID control, tuning methods and applications," *International Journal of Dynamics and Control*, vol. 9, no. 2, Springer Science and Business Media Deutschland GmbH, pp. 818–827, Jun. 01, 2021. doi: 10.1007/s40435-020-00665-4.
- [8] F. A. Widodo and K. Mutijarsa, "Design and Implementation of Movement, Dribbler and Kicker for Wheeled Soccer Robot."
- [9] R. Barua, S. Mandal, and S. Mandal, "Motion Analysis of A Mobile Robot With Three Omni-Directional Wheels," 2015. [Online]. Available: [www.ijiset.com](http://www.ijiset.com)
- [10] J. Ren, C. Xie, J. Xiao, K. Huang, and H. Lu, "A control system for active ball handling in the RoboCup middle size league," in *Proceedings of the 28th Chinese Control and Decision Conference, CCDC 2016*, Institute of Electrical and Electronics Engineers Inc., Aug. 2016, pp. 4396–4402. doi: 10.1109/CCDC.2016.7531777.
- [11] K. Kimura, S. Chikushi, and K. Ishii, "Month (Year)," *International Journal of Computational Intelligence Systems*, pp. xx-yy, 2019, doi: 10.1080/XXXXXXXXXXXXXX.
- [12] M. Santos Serafim, J. Manuel Torres Caldinhas Simões Vaz Supervisor, P. Manuel Urbano Almeida Lima, and M. Beatriz Mendes Batalha Vieira Borges, "Ball Handling Mechanism

- s for Mobile Robot s Electrical and Computer Engineering Examination Committee October 2013 i.”
- [13] V. Meerbeeck and D. Hameeteman Peter Teurlings Eindhoven, “An alterable dynamic model of a ball-handling mechanism,” 2023.
- [14] “Development of a low-cost solution for the ball handling mechanism for a football robot”.
- [15] S. Chikushi, T. Weerakoon, T. Sonoda, and K. Ishii, “Ball Dribbling Control for RoboCup Soccer Robot,” Seagaia Convention Center.



## BIOGRAFI PENULIS

	<p>Tajdar Hal Ata, lahir di Batam, Kepulauan Riau pada tanggal 23 November 2001. Sekarang sedang menjalani perkuliahan pada semester 7 di Politeknik Negeri Batam jurusan elektronika, prodi robotika. Sejak tahun 2020 aktif dalam tim robot Politeknik Negeri Batam. Pernah mengikuti kontes robot regional, nasional dan internasional pada tahun 2021 - 2023 di divisi KRSBI-beroda.</p>
---	--

# Object Detection Using Omnidirectional Camera

Hendawan Soebhakti<sup>1</sup>, Haidar Afnan Fakhruddin<sup>2</sup>, Rizqy pratama Singarimbun<sup>3</sup> and  
Nawwafal Dzaki Musthofa<sup>4</sup>

{hendawan@polibatam.ac.id<sup>1</sup>, hdrafnan@gmail.com<sup>2</sup>, mirzanabil34@gmail.com<sup>3</sup>, and  
d.mdzaki1222@gmail.com<sup>4</sup>}

Barelang Robotics Artificial and Intelligence Lab (BRAIL), Politeknik Negeri Batam, Batam, Indonesia  
Department of Electrical Engineering, Study Program Robotics Engineering, Politeknik Negeri Batam

**Abstract.** Object detection is a technology for detecting objects in images or videos. In this research, we implemented an object detection system using an omnidirectional camera as a crucial source of information in object detection systems, especially in real-time applications. Color spaces, thresholding, and morphological transformations were used as object detection methods, and the omnidirectional camera configuration transformed the view into a panorama. The evaluation results demonstrate that this system can detect objects and their distance in real-time. The conclusion of this study is that the object detection system with an omnidirectional camera, which transforms its view into a panorama, provides high accuracy and wide viewing flexibility.

**Keywords:** object detection, omnidirectional camera, panoramic

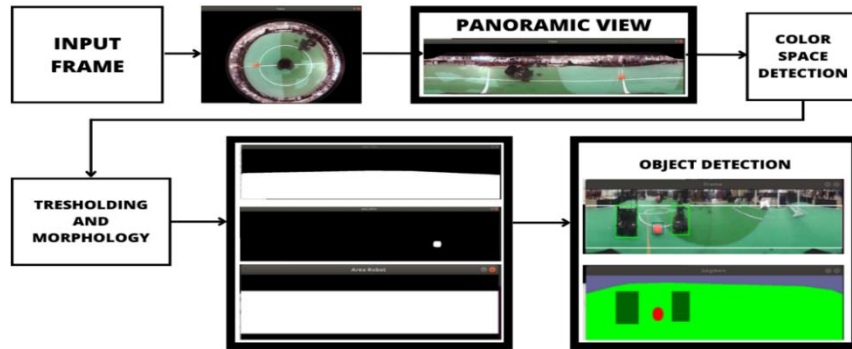
## 1 Introduction

Object detection is a crucial field in image processing, especially in the context of robot control. However, traditional cameras used in object detection face challenges such as image distortion, limited field of view, and difficulty in detecting objects in hard-to-reach areas. Omnidirectional cameras offer a solution to address these issues [1]. Omnidirectional cameras can capture images from all directions, reducing image distortion and expanding the field of view. Additionally, they can be used in challenging environments like small rooms or obstacle-filled spaces [2].

In order for a robot to engage in activities like playing soccer, it must be able to detect objects using an omnidirectional camera, as it allows the robot to have a 360° field of view without changing the camera's position [3]. In the previous method of detecting objects we used the YOLO V3 real-time object detection algorithm. This method is also used to detect objects such as balls, obstacles, goals. This method has advantages in detecting objects because of its ability to detect in realtime and is able to recognize objects well because it uses three different grid levels to detect objects at different scales in the image, so it can detect diverse objects. However, in actual play, the robot moves quite quickly, resulting in vibration of the camera and sometimes

making the view on the camera a little blurry which makes objects difficult to detect. In addition, efficiency and computational factors also affect this because, to get good results, it requires a lot and varied data labeling. The computational process in the YOLO method is also quite heavy because it uses the GPU to detect objects in realtime, making battery power consumption decrease quickly. Therefore, the right solution in this case is using the color segmentation with a wide visual perception system. This method is used because the computation process is quite light and efficient and also makes the robot can better understand the positional state of objects compared to a regular camera view. Therefore, in this study, we modify the initial radial or omnidirectional visual perspective into a panoramic view [4].

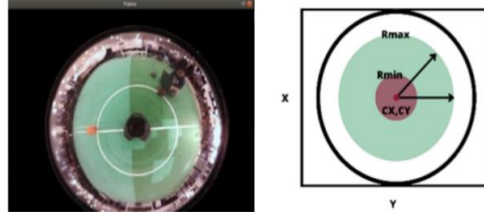
## 2 Method



**Fig. 1.** The framework of the system movement flow.

Image processing is carried out by inputting frames using an omnidirectional camera, followed by transforming the view into a panorama with the aim of determining the position and distance of detected objects. Subsequently, color space values are processed for each detected object. There are differences in the color space transformations for the field area, the ball object, and the robot object. For the field area, a good transformation is achieved in the HSV color space [5], while for the ball object, a good transformation is achieved in the HSL color space [6]. As for the robot object, a good transformation is achieved in the RGB color space [7]. The color space output for the ball object then undergoes thresholding and morphological transformation filtering stages to produce optimal binary values, and the same applies to the robot object [8], [9].

## 2.1 Omni directional view to panoramic



**Fig. 2.** Input frame, Implementation function in frame.

Cameras play a crucial role in the use of robots as visual sensors to perceive their surroundings. With omnidirectional cameras, they can utilize them as one of the visual sensors that can ensure a 360-degree field of view [10], as shown in **Fig. 2**. Changes in the camera's perspective are made to encompass the detection of positions, leveraging comprehensive detection range. Thus, the transformation of frames into a panorama shape becomes more optimal by applying formulas that can adjust pixel states as desired [4]. Therefore, in **Fig. 2**, on the right side, the implementation of the function naming to be modified is shown, along with the formula for transforming the shape into a panorama.

$$X_{pano} = 2.0 \times \left( \frac{R_{max} + R_{min}}{2} \right) \times \frac{22}{7} \quad (1)$$

$$Y_{pano} = R_{max} + R_{min} \quad (2)$$

$$R = \left( \frac{Y}{Y_{pano}} \right) \times (R_{max} + R_{min}) + R_{min} \quad (3)$$

$$\theta = \left( \frac{X}{X_{pano}} \right) \times 2.0 \times \frac{22}{7} \quad (4)$$

$$X_{equal} = Cx + R \times \sin \theta \quad (5)$$

$$Y_{equal} = Cy + R \times \cos \theta \quad (6)$$



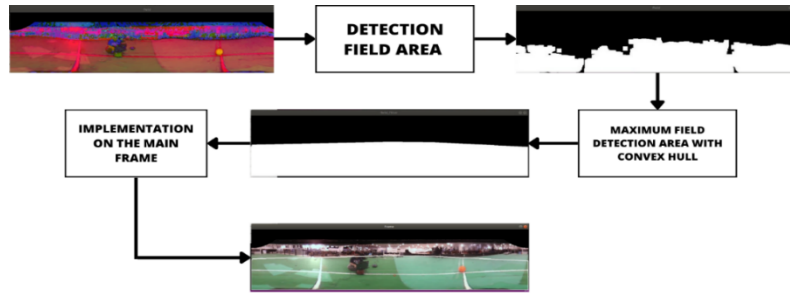
**Fig. 3.** After applying panoramic.

Determining the position of the frame size value to be transformed involves defining equations 1 and 2, resulting in the height and width of a new frame. In equation 3, the radius area y is generated between R\_(max). Meanwhile, in equation 4, the radius area x is obtained from what

is generated between  $R_{\text{min}}$ . Thus, from these equations, the values of  $X_{\text{equal}}$  and  $Y_{\text{equal}}$  can be determined, which will undergo a frame condition change [4], leading to frame transformation after going through functions from 1 to 6, as shown in **Fig. 3**.

## 2.1 Object detection

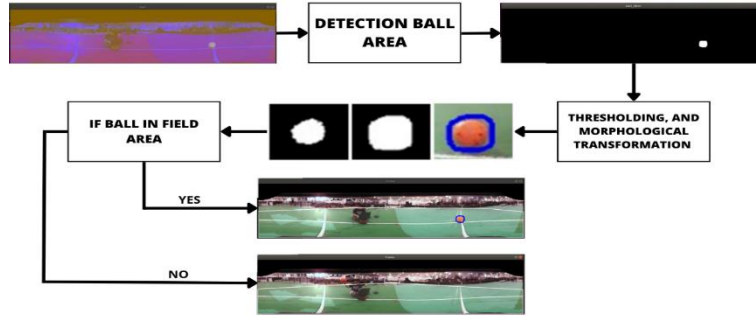
Determining the detected objects using color space serves as a method for selecting colors to define, create, and visualize individual colors. The color space process is subjective and inherently variable; its purpose is to describe colors between different hues and standardize them [7], [11]. There is also a thresholding method that achieves sensitivity in color space outcomes for the transformed frame values. Thresholding is a technique that distinguishes objects from the background by detecting differences in darkness or brightness [9], [11]. Additionally, there is a method called morphological transformation to assist in identifying target regions. Morphological transformation is a binary image process that alters the image's perception as a processing tool. Morphological transformation can change pixel values, pixel value comparisons, and alter the original structure of the frame [12]. Morphological transformation consists of several parts, namely, erosion and dilation [11]. Erosion is the process of removing object boundaries to make them part of the background based on the texture element used; this function reduces the size of objects by eroding binary values. Meanwhile, dilation is the process of merging background points into an object based on the texture element used; this function enlarges the size of objects by adding binary values [13].



**Fig. 4.** Convert to HSV frame for field detection, Biner field detection, Maximize the result obtained from Thresholding, and Morphological Transformation in the field detection, The result implemented on the frame.

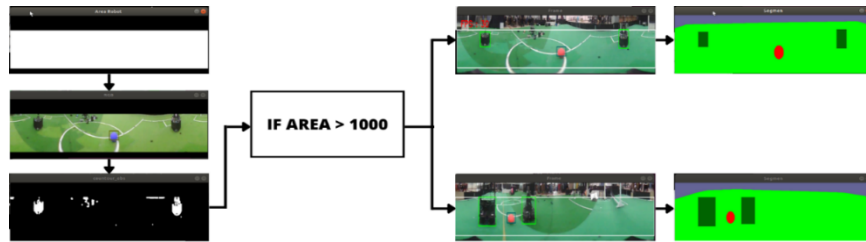
Field detection involves identifying the robot's object detection coverage area using the HSV color space method to locate green color. This generates binary values corresponding to **Fig. 4**. area. However, using HSV alone seems inadequate, as some expected values are missing. To address this, thresholding and morphological transformations adjust binary values, converting large to small and vice versa. These values are then transformed into a convex hull, enhancing object detection in the selected area **Fig. 4**.





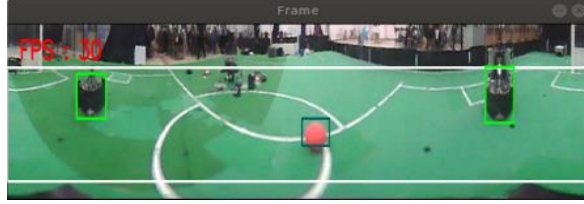
**Fig. 5.** Convert to HSL frame for ball detection, Biner ball detection, The result biner of the ball, Maximize the result obtained from Thresholding, and Morphological Transformation in the ball detection, Result on frame, The position of the ball in the field, The position of the ball outside the field.

In this phase, the ball's position within the frame is determined using the HSL color space method, chosen for superior results. Maximizing binary values in the HSL color space corresponds to the area in **Fig. 5**. Adjusting binary values through thresholding and morphological transformations enhances sensitivity, transitioning between large and small values. The maximized ball detection combines with the field area, setting the boundary for ball detection as the maximum ball area. If the ball falls outside the field area, it remains undetected, as illustrated in **Fig. 5**.



**Fig. 6.** Robot object detection flowchart.

In robot detection, the RGB color space excels, maximizing binary values as depicted in **Fig. 6**. These values undergo enhancement via thresholding and morphological transformations, akin to ball object detection. Specific boundaries are set to exclude robot look-alikes, as shown in **Fig. 6**. Objects under 1000 in area are considered undetected. Combining robot and field area boundaries forms the ultimate detection area, as in **Fig. 6**.



**Fig. 7.** Frames per second (FPS) on all object detection processing.

### 3 Results and Discussion



**Fig. 8.** (a) The size of the x and y frames as a reference for object distance, (b) Determines the value of the detected object against the pixel value.

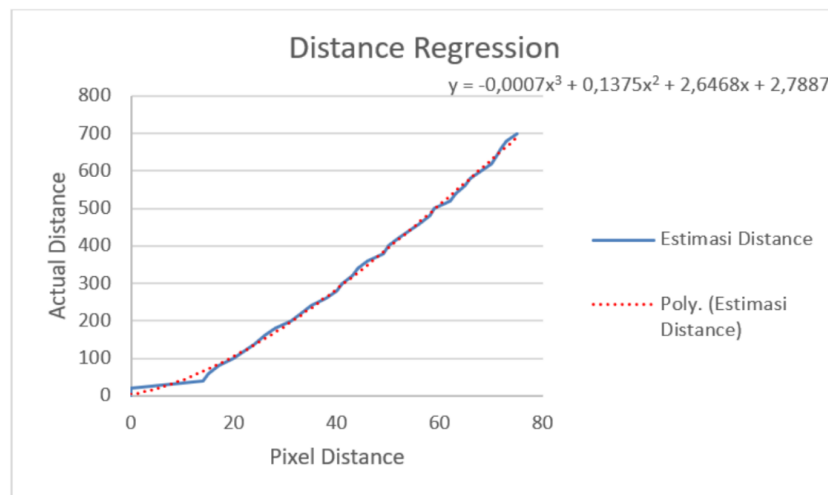
Testing an object detection system with an adjustable field-of-view omnidirectional camera in panoramic mode has significantly improved accuracy and overall effectiveness. After object detection, additional steps are crucial for the robot to determine the position and distance of detected objects. To establish a practical global point distance metric, the frame's area approximates the pixel frame value in **Fig. 8a**. Data from detected objects is condensed into a single value, correlated with the pixel frame in **Fig. 8b**, estimating object distance relative to the pixel frame. This ensures precise distance estimation by comparing pixel frame data with actual distances. Ball object detection spans 60 centimeters (camera's minimum) to 700 centimeters, while robot object detection ranges from 60 to 300 centimeters due to detection area constraints.

**Table 1.** Estimated ball detection

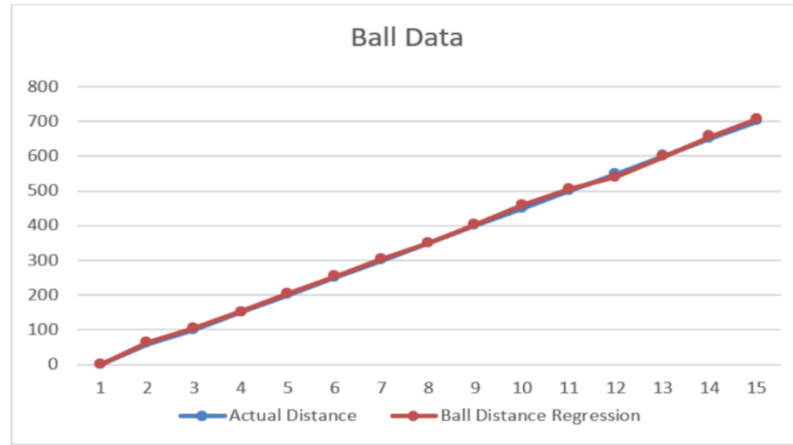
Actual Distance (cm)	Pixel Frame	Ball Distance Regression (cm)	Estimate Ball Distance Error (cm)	Detect or No Detect
0	0	0	0	No Detect
60	15	64,07	4,07	Detect
100	20	105,12	5,12	Detect
150	25	152,25	2,25	Detect
200	31	205,12	5,12	Detect
250	36	253,5	3,50	Detect
300	41	304,64	4,64	Detect
350	45	350,24	0,24	Detect
400	50	402,55	2,55	Detect
450	55	459,27	9,27	Detect
500	59	505,39	5,39	Detect
550	64	540,24	-9,76	Detect
600	68	598,46	-1,54	Detect
650	71	656,43	6,43	Detect
700	75	707,25	7,25	Detect

**Table 2.** Estimted robot detection

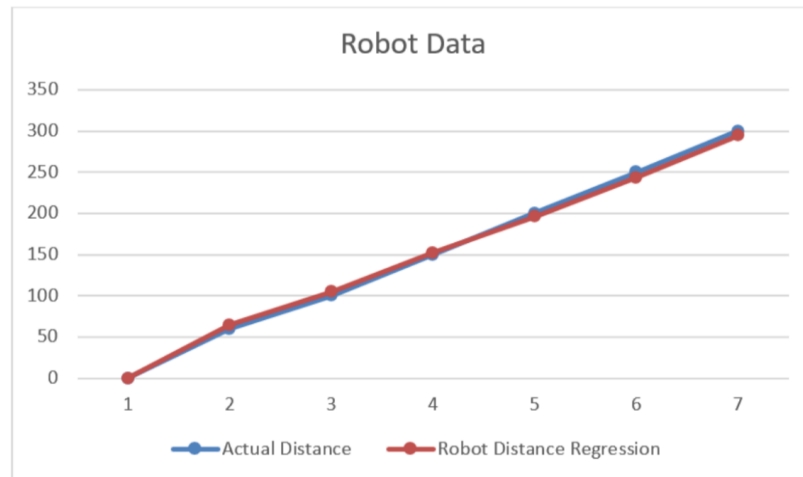
Actual Distance (cm)	Pixel Frame	Robot Distance Regression (cm)	Estimate Robot Distance Error (cm)	Detect or No Detect
0	0	0	0	No Detect
60	15	64,87	4,87	Detect
100	20	105,12	5,12	Detect
150	25	152,25	2,25	Detect
200	31	196,12	-3,88	Detect
250	36	243,61	-6,39	Detect
300	41	294,22	-5,78	Detect



**Fig. 9.** Comparison graph of the actual distance and the pixel distance of the object.



**Fig. 10 .** Comparison chart of actual distance and ball distance.



**Fig. 11.** Comparison chart of actual distance and robot distance.

## 4 Conclusion

The results obtained from testing an object detection system using an omnidirectional camera and employing color space methods, thresholding techniques, and morphological transformations indicate that this system exhibits a high level of accuracy and achieves rapid processing with a frame rate of up to 30 FPS (Frames Per Second) during image processing. Moreover, this system demonstrates robust performance in mitigating the issues of false positives and false negatives, which are often encountered in object detection systems.



## References

- [1] C. Premachandra and M. Tamaki : A Hybrid Camera System for High-Resolutionization of Target Objects in Omnidirectional Images, *IEEE Sens J*, vol. 21, no. 9, pp. 10752–10760, May 2021, doi: 10.1109/JSEN.2021.3059102.
- [2] E. A. Nurrohmah, B. Sena Bayu, M. M. Bachtiar, I. Kurnianto Wibowo, and R. Adryantoro : Detecting Features of Middle Size Soccer Field using Omnidirectional Camera for Robot Soccer ERSOW, in *Proceeding - ICoSTA 2020: 2020 International Conference on Smart Technology and Applications: Empowering Industrial IoT by Implementing Green Technology for Sustainable Development*, Feb. 2020. doi: 10.1109/ICoSTA48221.2020.1570615971.
- [3] IEEE Staff, 2019 International Conference on Electrical, Electronics and Information Engineering (ICEEIE). IEEE, 2019.
- [4] Guo li ji nan guo ji da xue, IEEE Consumer Electronics Society, IEEE Consumer Electronics Society. Taipei Chapter, and Institute of Electrical and Electronics Engineers, 2016 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW).
- [5] H. Zhang, S. Qu, H. Li, J. Luo, and W. Xu : A Moving Shadow Elimination Method Based on Fusion of Multi-Feature, *IEEE Access*, vol. 8, pp. 63971–63982, 2020, doi: 10.1109/ACCESS.2020.2984680.
- [6] Z. Zhong, T. Wang, K. Zeng, X. Zhou, and Z. Li : White Blood Cell Segmentation via Sparsity and Geometry Constraints, *IEEE Access*, vol. 7, pp. 167593–167604, 2019, doi: 10.1109/ACCESS.2019.2954457.
- [7] S. Zhi, Y. Cui, J. Deng, and W. Du : An FPGA-based simple RGB-HSI space conversion algorithm for hardware image processing, *IEEE Access*, vol. 8, pp. 173838–173853, 2020, doi: 10.1109/ACCESS.2020.3026189.
- [8] G. Boato, D. T. Dang-Nguyen, and F. G. B. de Natale: Morphological Filter Detector for Image Forensics Applications, *IEEE Access*, vol. 8, pp. 13549–13560, 2020, doi: 10.1109/ACCESS.2020.2965745.
- [9] A. A. Ewees, M. Abd Elaziz, M. A. A. Al-Qaness, H. A. Khalil, and S. Kim : Improved Artificial Bee Colony Using Sine-Cosine Algorithm for Multi-Level Thresholding Image Segmentation, *IEEE Access*, vol. 8, pp. 26304–26315, 2020, doi: 10.1109/ACCESS.2020.2971249.
- [10] Winarno, A. S. Agoes, E. I. Agustin, and D. Arifianto : Object detection for KRSBI robot soccer using PeleeNet on omnidirectional camera, *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 18, no. 4, pp. 1942–1953, 2020, doi: 10.12928/TELKOMNIKA.V18I4.15009.
- [11] H. Li, X. Zhao, A. Su, H. Zhang, J. Liu, and G. Gu : Color Space Transformation and Multi-Class Weighted Loss for Adhesive White Blood Cell Segmentation, *IEEE Access*, vol. 8, pp. 24808–24818, 2020, doi: 10.1109/ACCESS.2020.2970485.
- [12] Z. Gong, N. Qin, and G. Zhang : Three-Dimensional Generalized Discrete Fuzzy Number and Applications in Color Mathematical Morphology, *IEEE Access*, vol. 9, pp. 25405–25421, 2021, doi: 10.1109/ACCESS.2021.3057623.

- [13] S. Zhao, C. Chen, and Y. Luo : Probabilistic Principal Component Analysis Assisted New Optimal Scale Morphological Top-Hat Filter for the Fault Diagnosis of Rolling Bearing, *IEEE Access*, vol. 8, pp. 156774–156791, 2020, doi: 10.1109/ACCESS.2020.3019638.
- [14] X. Wei, J. Li, B. Long, X. Hu, H. Wu, and H. Li : Method for Vortex Shape Retrieval and Area Calculation Based on Convex Hull Algorithm, *IEEE Access*, vol. 9, pp. 2706–2714, 2021, doi: 10.1109/ACCESS.2020.3046651.

# AUGEMENTED MONTE CARLO LOCALIZATION FOR WHEELED SOCCER ROBOT

Mochamad Rizal Fauzi, Hendawan Soebhakti

Department of Electrical Engineering, Study Program Robotics Engineering, Politeknik Negeri Batam

Article Info	ABSTRACT
<b>Article history:</b>	Jurnal ini menyajikan penerapan teknik lokalisasi pada robot sepak bola beroda dengan Monte Carlo Localizaion untuk estimasi posisi secara global. Dengan tiga komponen utama; <i>motion model</i> menggunakan odometry, <i>measurement model</i> dengan <i>likelihood field</i> dan sampling dengan <i>low variance resampling</i> . Untuk mengatasi masalah ketika robot terdorong oleh robot lainnya, ditambahkan metode untuk memberikan posisi sembarang terhadap partikel. Percobaan dilakukan dengan menggerakkan robot ke sembarang posisi, kemudian kembali ke posisi awal. Data hasil estimasi dan odometry kemudian dibandingkan dengan posisi sebenarnya. Jumlah partikel yang digunakan adalah 100, 200, 600 dan 1000. Hasil percobaan menunjukkan dengan 1000 partikel menghasilkan rata-rata error yang sangat kecil, yaitu 1.9 cm pada sumbu x, 5,8 cm pada sumbu y dan 0,05 radian.
<b>Keywords:</b> Robot sepak bola beroda Lokaliasi Estimasi posisi Robocup MSL Omnivision	
<b>Corresponding Author:</b> Mochamad Rizal Fauzi Department of Electrical Engineering, Study Program Robotics Engineering Politeknik Neger Batam 29461 Batam Kota, Kota Batam, Indonesia Email: fauzimochamadrizal2@gmail.com	

## 1. INTRODUCTION

Lokalisasi menjadi syarat agar robot dapat melakukan navigasi secara otonom. Seperti pada Robocup Middle Size League (MSL), dimana banyak tim yang berkompetisi pada robot sepak bola beroda, merupakan contoh dari penerapan lokalisasi. Semua robot harus bermain dengan hanya menggunakan sensor yang terdapat pada robot saja. Hal ini menjadi tantangan bagi lokalisasi mengingat kecepatan robot yang tinggi dan akselerasi yang agresif dapat menyebabkan slip.

Barelang63 merupakan salah satu tim yang ikut dalam kompetisi pada MSL, dengan gambaran umum pada [1]. Untuk lokalisasi, sebelumnya menggunakan algoritma Monte Carlo Localization dengan mendeteksi landmark dari beberapa titik yang telah ditentukan, diantaranya sudut kiri-kanan, kotak penalti kiri-kanan dan tiang gawang kiri-kanan [2]. Landmark tersebut dideteksi menggunakan kamera depth dengan posisi pada bagian depan robot YOLO v3 object detection system [3]. Dari sistem tersebut menghasilkan rata-rata error 13,64 cm pada sumbu x dan 12,44 cm pada sumbu y [4]. Oleh karena itu, Untuk meningkatkan performa robot ketika bermain, maka diperlukan estimasi posisi dengan ketentuan

- Area deteksi sensor lebih menyeluruh, tidak hanya bagian depan robot, dan
- Mampu memulihkan posisi yang disebabkan kegagalan estimasi.

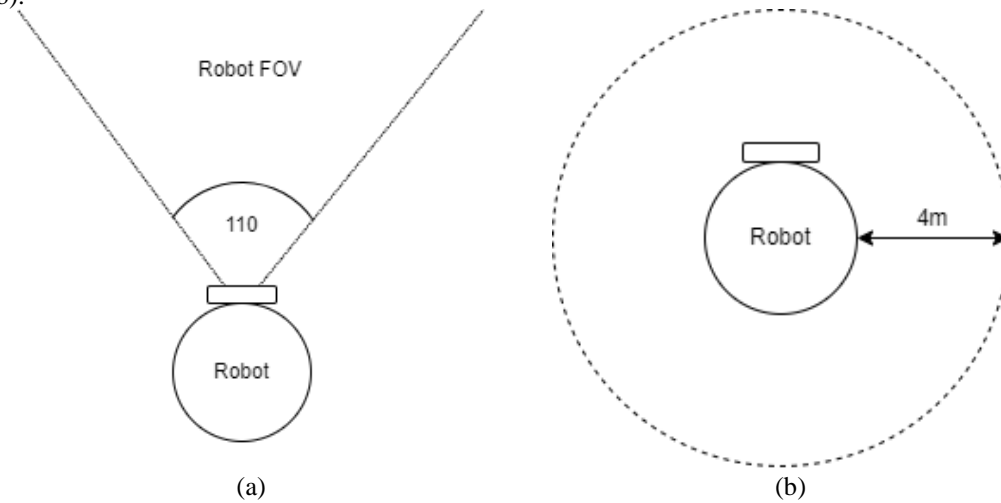
Oleh karena itu, diajukan algoritma Augmented Monte Carlo Localization untuk memulihkan posisi yang disebabkan oleh kegagalan estimasi dan menggunakan kamera omnivision untuk menjangkau area yang lebih menyeluruh.

## 2. METHOD

### 2.1. Line Detection

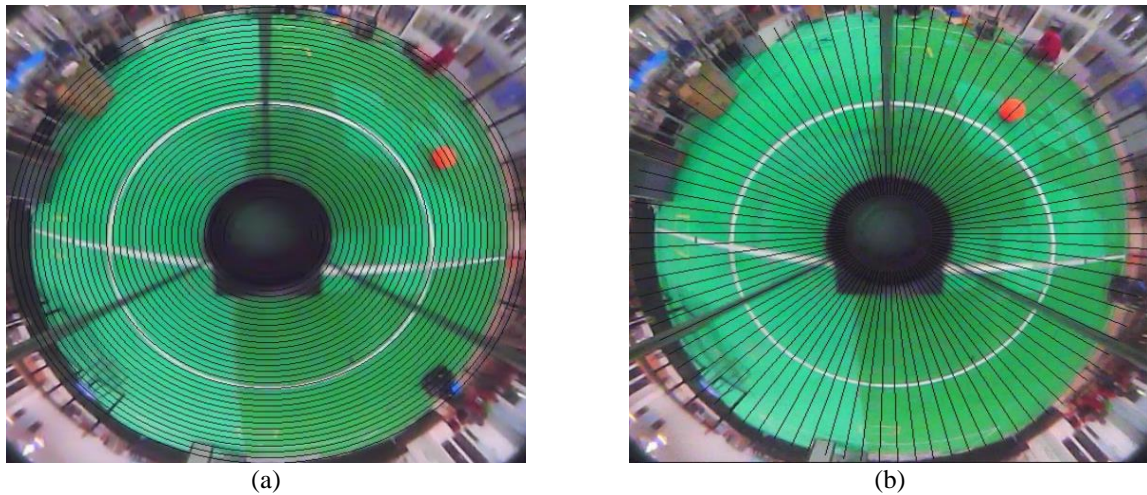
Pendeteksian garis pada robot sepak bola beroda merupakan elemen yang penting dalam sistem lokalisasi. Dengan menggunakan kamera vision, robot dapat mengidentifikasi garis-garis putih pada lapangan untuk menentukan

posisi dan orientasi relatif terhadap robot. Kamera yang digunakan merupakan kamera omnidirectional daripada menggunakan kamera depth. Dengan menggunakan kamera depth, meskipun robot dapat menjangkau jarak yang cukup jauh, tetapi area pendeteksiannya tidak cukup luas seperti pada gambar 1 (a). Sedangkan, dengan menggunakan kamera omnidirection, area pendeteksian meliputi 360 derajat dari robot dengan jarak maksimum 4meter seperti pada gambar 1 (b).



Gambar 1. Representasi Field of View (FoV) dari (a) kamera Zed2 dengan 110° dan (b) kamera omnivision dengan 360° dan radius 4m

Untuk mendeteksi garis putih pada lapangan, diterapkan dua jenis tipe pendeteksian garis yang disebut *scanlines*: radial dan circular. *Scanlines* adalah daftar dari posisi pixel pada gambar yang berbentuk sesuai dengan tipenya [5]. Gambar 2 (a) memperlihatkan *circular scanlines* dengan bentuk garis melingkar dari titik tengah dengan radius paling kecil (sesuai ukuran radius robot yang terdeteksi) sampai radius lingkaran sama dengan setengah dari ukuran frame gambar. Gambar 2 (b) memperlihatkan *radial scanlines* dengan bentuk garis memanjang dari tengah. Setelah jarak pixel dari garis putih didapatkan, dilakukan teknik interpolasi linear agar menghasilkan jarak yang sesuai dengan jarak sesungguhnya.



Gambar 2. (a) circular scanline dan (b) radial scanline

## 2.2. Odometry Motion Model Sampling

Odometry motion model menggunakan pengukuran odometry sebagai kontrol. Odometry biasanya didapat dari informasi encoder pada roda [6]–[9]. Akan tetapi, karena kemungkinan terjadinya slip, sehingga tidak ada koordinat transformasi yang tetap antara koordinat odometry dengan koordinat global.

Gambar 3 menunjukkan algoritma sample motion model odometry dengan informasi odometry  $u_t$  dan initial pose  $x_{t-1}$ . Pada baris 2 dan 3 mengkonversi relative odometry  $x, y$  menjadi koordinat global particle dengan menggunakan matriks rotasi persamaan (1). Probabilistic motion model mengasumsikan informasi odometry terganggu oleh *noise*. Sehingga, pada baris 4-6 ditambahkan probabilitas dari random zero mean gaussian model *prob()* sebagai bentuk *noise*. Baris 10 mengembalikan nilai pose  $x_t$  yang diperoleh dari pose partikel ditambah motion parameter.

1: **Algorithm** sample\_motion\_model\_odometry( $u_t, x_{t-1}$ ):

2:      $\delta x_{particle} = dx \cos(\theta_{particle}) - dy \sin(\theta_{particle})$

3:      $\delta y_{particle} = dx \sin(\theta_{particle}) + dy \cos(\theta_{particle})$

4:      $\hat{\delta} x_{particle} = \delta x_{particle} + \text{prob}(0.0, \sigma_x)$

5:      $\hat{\delta} y_{particle} = \delta y_{particle} + \text{prob}(0.0, \sigma_y)$

6:      $\hat{\delta} \theta_{particle} = \delta \theta_{particle} + \text{prob}(0.0, \sigma_\theta)$

7:      $x_{particle} = x_{particle} + \hat{\delta} x_{particle}$

8:      $y_{particle} = y_{particle} + \hat{\delta} y_{particle}$

9:      $\theta_{particle} = \theta_{particle} + \hat{\delta} \theta_{particle}$

10:    return  $x_t(x_{particle}, y_{particle}, \theta_{particle})$

Gambar 3. Algoritma untuk sampling  $p(x_t | u_t, x_{t-1})$  berdasarkan informasi odometry. Perubahan pose relatif pada waktu  $t$  adalah  $\mathbf{dx}, \mathbf{dy}$  dipetakan menjadi pose global  $\delta x = (\delta x_{particle}, \delta y_{particle})$ . Perubahan pose dengan sampling zero mean gaussian  $\hat{\delta} x = (\hat{\delta} x_{particle}, \hat{\delta} y_{particle}, \hat{\delta} \theta_{particle})$ .

### 2.3. Measurement Model

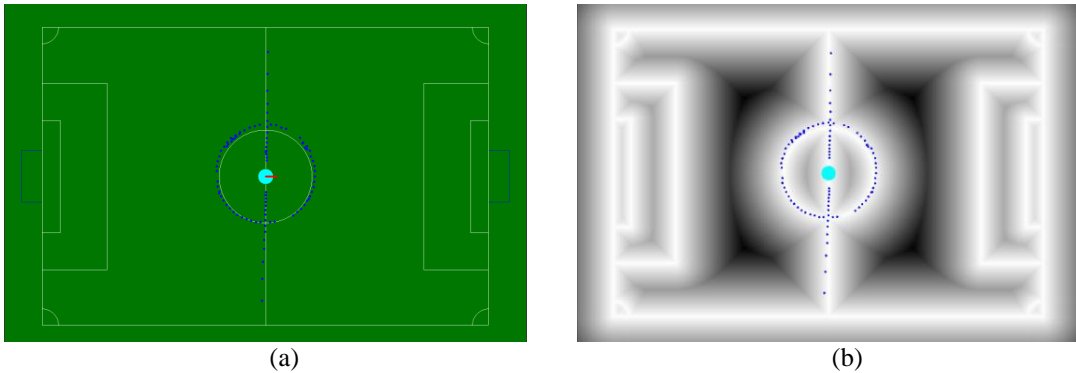
Likelihood fields merupakan salah satu algoritma yang digunakan untuk menentukan pose relatif pada lapangan dengan menghitung importance weight  $w$  dari particle [10], [11]. End points dari sensor  $z_t$  diproyeksikan menjadi koordinat global dari map. Oleh karena itu, diperlukan data koordinat global robot dan point sensor.  $x_t(x \ y \ \theta)^T$  menyatakan pose robot pada waktu  $t$ . Koordinat lokal dari sensor terhadap robot dinotasikan dengan  $(x_{k,sens} \ y_{k,sens})^T$ . Data tersebut kemudian dipetakan menjadi koordinat global dengan matriks rotasi seperti pada persamaan (1) menghasilkan persamaan (2). Hasil perkalian matriks tersebut ditunjukkan pada persamaan (3).

$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \quad (1)$$

$$\begin{pmatrix} x_{z_t^k} \\ y_{z_t^k} \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_{k,sens} \\ y_{k,sens} \end{pmatrix} \quad (2)$$

$$\begin{pmatrix} x_{z_t^k} \\ y_{z_t^k} \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} x_{k,sens}\cos\theta - y_{k,sens}\sin\theta \\ x_{k,sens}\sin\theta + y_{k,sens}\cos\theta \end{pmatrix} \quad (3)$$

Gambar 4 (a) menunjukkan visualisasi dari robot berwarna cyan yang berada di tengah-tengah lapangan dengan *measurement*  $z_t^k$  berupa point-point berwarna biru yang telah dipetakan menjadi koordinat global. Sedangkan pada gambar 4 (b) merupakan visual dari array yang berisi *error-distribution* yang digunakan untuk likelihood field dengan konfigurasi: lokasi semakin gelap, maka semakin kecil kemungkinan terdapat garis pada lokasi tersebut.



Gambar 4. (a) environment lapangan. (b) Likelihood field.

Gambar 5 merupakan algoritma untuk menghitung importance weight  $w$  dengan menggunakan likelihood field. Point-point sensor diproyeksikan menjadi posisi global pada baris 5-6 dengan persamaan (3). Kemudian, pada baris 7 mencari nilai minimum jarak dari point sensor  $\langle x, y \rangle$  terhadap posisi  $\langle x', y' \rangle$  line point pada map. Pada baris 9 memperbarui nilai weight berdasarkan jumlah point dan nilai error yang diperoleh pada baris 8.



```

1: Algorithm likelihood_field_scan_line_point_model( $z_t, x_t, m$ ):
2:    $error = 0$ 
3:    $n = size(all\ k)$ 
4:   for all  $k$  do
5:      $x_{z_t^k} = x + x_{k,sens} \cos \theta - y_{k,sens} \sin \theta$ 
6:      $y_{z_t^k} = y + x_{k,sens} \sin \theta + y_{k,sens} \cos \theta$ 
7:      $dist = \min \left\{ \sqrt{(x_{z_t^k} - x')^2 + (y_{z_t^k} - y')^2} \mid (x', y') \text{ occupied in } m \right\}$ 
8:      $error = error + dist$ 
9:    $w = 1.0 \cdot error^{-1} \cdot n^{-1}$ 
10:  return  $w$ 

```

Gambar 5. Algoritma pendeteksian garis dengan model likelihood field

## 2.4. Augmented Monte Carlo Localization

Augmented Monte Carlo Localization atau AMCL merupakan varian dari algoritma Monte Carlo Localization (MCL) untuk mengatasi masalah estimasi posisi dan orientasi (pose) pada robot otonom [12]–[16]. AMCL diperoleh dengan menggabungkan probabilistic motion dan perceptual models ke dalam algoritma particle filter.

```

1: Algorithm Augmented_MCL( $\mathcal{X}_{t-1}, u_t, z_t, m$ ):
2:   static  $w_{slow}, w_{fast}$ 
3:    $\mathcal{X}_t = \mathcal{X}_{t-1} = \emptyset$ 
4:   for  $m = 1$  to  $M$  do
5:      $x_t^{[m]} = \text{sample\_motion\_model}(u_t, x_{t-1}^{[m]})$ 
6:      $w_t^{[m]} = \text{measurement\_model}(z_t, x_t^{[m]}, m)$ 
7:      $\tilde{\mathcal{X}}_t = \tilde{\mathcal{X}}_t + \langle x_t^{[m]}, w_t^{[m]} \rangle$ 
8:      $w_{avg} = w_{avg} + \frac{1}{M} w_t$ 
9:   endfor
10:   $w_{slow} = w_{slow} + \alpha_{slow}(w_{avg} - w_{slow})$ 
11:   $w_{fast} = w_{fast} + \alpha_{fast}(w_{avg} - w_{fast})$ 
12:  for  $m = 1$  to  $M$  do
13:    with probability  $\max\{0.0, 1.0 - w_{fast}/w_{slow}\}$  do
14:      add random pose to  $\mathcal{X}_t$ 
15:    else
16:      draw  $j \in \{1, \dots, N\}$  with probability  $\propto w_t^{[j]}$ 
17:      add  $x_t^{[j]}$  to  $\mathcal{X}_t$ 
18:    endwith
19:  endfor
20:  return  $\mathcal{X}_t$ 

```

Gambar 6. Variasi MCL yang menambahkan sample acak. Banyaknya sample acak ditentukan dengan membandingkan  $w_{slow}$  dan  $w_{fast}$

Gambar 6 menunjukkan algoritma AMCL yang merepresentasikan *the belief*  $bel(x_t)$  dari banyak  $M$  partikel seperti pada persamaan (4). Baris 5 menunjukkan sample dari motion model dengan menggunakan partikel dari *present belief* sebagai point start. Kemudian, perhitungan measurement model pada baris 6 untuk menentukan importance weight dari partikel tersebut. Fungsi **motion\_model** yang digunakan adalah odometry motion model yang terdapat pada gambar 3 dan **measurement\_model** yang digunakan adalah likelihood field model yang terdapat pada gambar 5. **Augmented\_MCL** menghitung empirical measurement likelihood pada baris 8, dan mempertahankan short-term dan long-term averages dari likelihood tersebut pada baris 10-11. Algoritma ini membutuhkan  $\alpha_{slow}$  dan  $\alpha_{fast}$  dengan ketentuan seperti pada persamaan (5). Parameter  $\alpha_{slow}$  dan  $\alpha_{fast}$  merupakan peluruhan untuk exponential filters yang memperkirakan rata-rata long-term dan short-term. Pada baris 13, ketika proses resampling, random sample ditambahkan dengan probability nilai tertinggi antara 0 dengan 1 dikurangi  $\alpha_{fast}$  dibagi  $\alpha_{slow}$  seperti pada persamaan (6). Sebaliknya, dilakukan resampling dengan *low variance resampling* seperti pada baris 16-17.

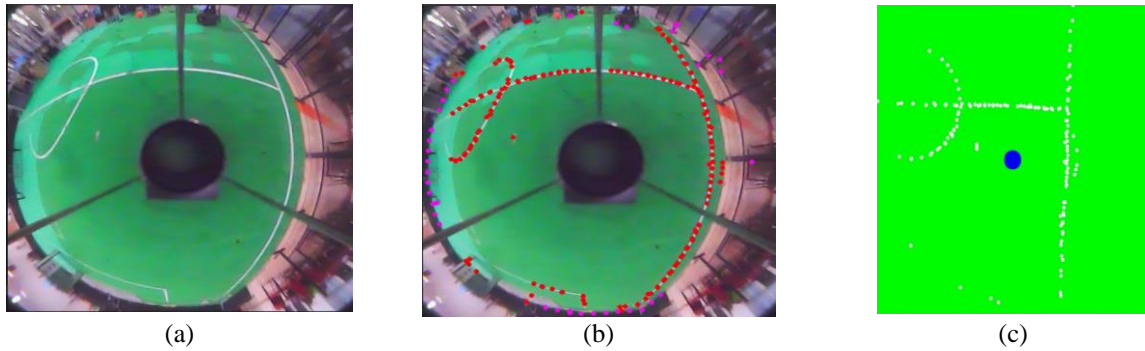
$$\mathcal{X}_t = \{x_t^{[1]}, x_t^{[2]}, \dots, x_t^{[M]}\} \quad (4)$$

$$0 \leq \alpha_{slow} \ll \alpha_{fast} \quad (5)$$

$$\max\left\{0.0, 1.0 - \frac{w_{fast}}{w_{slow}}\right\} \quad (6)$$

### 3. RESULT AND DISCUSSION

#### 3.1 Hasil Pendeteksian Garis



Gambar 7. hasil pendeteksian garis (a) raw image, (b) point-point dari garis dan (c) hasil regresi

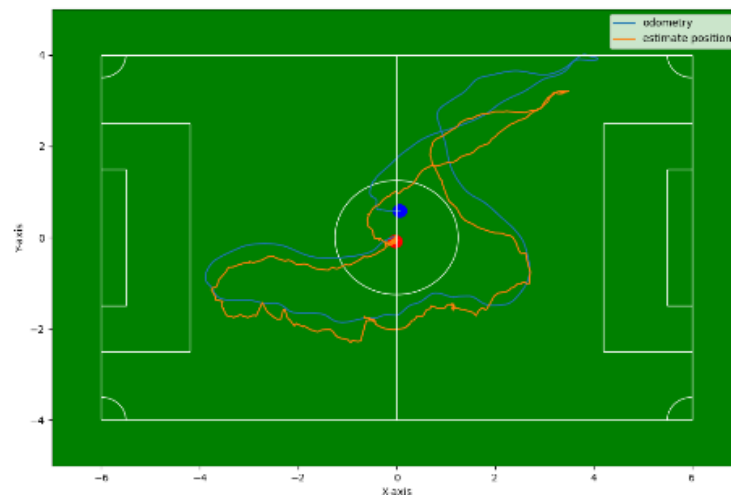
Pada gambar 7 (a) menunjukkan gambar yang dihasilkan oleh kamera omnivision. Dengan menggunakan *scanlines*, diperoleh informasi titik-titik yang didapatkan dari garis lapangan seperti pada gambar 7 (b). Titik-titik tersebut kemudian diinterpolasi agar sesuai dengan jarak sesungguhnya seperti pada gambar 7 (c).

#### 3.2 Hasil Estimasi Posisi

Pada percobaan ini robot digerakkan secara manual dengan remote dari android. Robot digerakkan secara acak dengan kecepatan maksimum 1,2 m/s, kemudian kembali ke posisi awal. Setelah itu, dibandingkan posisi akhir dari odometry encoder dengan estimasi dari AMCL. Hasil percobaan dapat dilihat pada Tabel 1 dan gambar (8).

Table 1. Rata-rata error posisi robot terhadap posisi berhenti

Jumlah Partikel	Rata-rata Error (m)					
	x_odom	y_odom	theta_odom (rad)	x_est	y_est	theta_est (rad)
100	0.5910	0.6359	0.0465	0.05967	0.13225	0.2621
200	0.4701	0.7705	0.0603	0.06030	0.10035	0.0945
600	0.4159	0.9959	0.0590	0.05417	0.18417	0.0601
1000	0.54061	0.3304	0.0252	0.01922	0.05812	0.0597



Gambar 8. percobaan dengan 1000 partikel. Lingkaran biru: data odometry, lingkaran merah: hasil estimasi

Berdasarkan Tabel 1, dengan jumlah 100 partikel, hasil estimasi menghasilkan rata-rata error 0.059 m pada sumbu x, 0.132 m pada sumbu y dan 0.262 rad pada theta, sementara dengan odometry menghasilkan 0.591 m error pada sumbu x, 0.635 m error pada sumbu y dan 0.046 rad pada theta. Tingkat akurasi dari estimasi AMCL terus meningkat berbanding lurus dengan jumlah partikel yang digunakan. Hal ini terlihat pada percobaan dengan menggunakan 1000 partikel yang menghasilkan rata-rata error 0.0192 m pada sumbu x, 0.0581 m pada sumbu y dan

0.059 rad pada theta meskipun dengan error odometry 0.540 m pada sumbu x, 0.330 pada sumbu y dan 0.025 rad pada theta. Visualisasi dari percobaan dengan 1000 partikel ditunjukkan pada gambar 8.

#### 4. CONCLUSION

Berdasarkan hasil yang didapat, AMCL dapat memberikan posisi yang optimal untuk lokalisasi pada robot sepakbola beroda dengan rentang error 0.01m pada sumbu x, 0,05m pada sumbu y dan 0,05 radian pada orientasi robot. Jumlah partikel optimal yang digunakan dari percobaan ini adalah 1000 partikel. Dengan jumlah partikel tersebut, menghasilkan error yang kecil. Penerapan algoritma ini dapat digabungkan dengan *traction control* daripada dengan kecepatan robot secara konstan se hingga tidak terjadi slip yang terlalu besar.


#### ACKNOWLEDGEMENTS

Penelitian ini didukung oleh tim BARELANG63 dari Politeknik Negeri Batam (POLIBATAM). BARELANG63 merupakan salah satu tim dari POLIBATAM yang berfokus mengembangkan robot *Middle-Size League* (MSL) beroda sejak 2017.

#### REFERENCES

- [1] H. Soebhakti *et al.*, "BARELANG 63 Team Description 2022".
- [2] B. Sugandi, S. Prayoga, I. A. Riandi, and D. G. Tinambunan, "Goal Detection and Opponent Avoidance Algorithm for Wheeled Robot Soccer using Color Filtering and Contour Extraction," in *2018 International Conference on Applied Engineering (ICAE)*, Batam: IEEE, Oct. 2018, pp. 1–5. doi: 10.1109/INCAE.2018.8579403.
- [3] H. Soebhakti, S. Prayoga, R. A. Fatekha, and M. B. Fashla, "The Real-Time Object Detection System on Mobile Soccer Robot using YOLO v3," in *2019 2nd International Conference on Applied Engineering (ICAE)*, Batam, Indonesia: IEEE, Oct. 2019, pp. 1–6. doi: 10.1109/ICAE47758.2019.9221734.
- [4] Y. Pratama, H. Soebhakti, S. Prayoga, and B. Budiana, "Localization System on Wheel Robot Soccer," in *Proceedings of the 5th International Conference on Applied Engineering, ICAE 2022, 5 October 2022, Batam, Indonesia*, Batam, Indonesia: EAI, 2023. doi: 10.4108/eai.5-10-2022.2327751.
- [5] A. J. R. Neves, A. Trifan, and B. Cunha, "UAVision: A Modular Time-Constrained Vision Library for Color-Coded Object Detection," in *Computational Modeling of Objects Presented in Images. Fundamentals, Methods, and Applications*, vol. 8641, Y. J. Zhang and J. M. R. S. Tavares, Eds., in *Lecture Notes in Computer Science*, vol. 8641, Cham: Springer International Publishing, 2014, pp. 351–362. doi: 10.1007/978-3-319-09994-1\_35.
- [6] K. V. Utama, R. A. Fatekha, S. Prayoga, D. S. Pamungkas, and R. P. Hudhajanto, "Positioning and Maneuver of an Omnidirectional Robot Soccer," in *2018 International Conference on Applied Engineering (ICAE)*, Batam: IEEE, Oct. 2018, pp. 1–5. doi: 10.1109/INCAE.2018.8579148.
- [7] J. J. Kon, "Pose and Velocity Estimation for Soccer Robots".
- [8] Kok Seng Chong and L. Kleeman, "Accurate odometry and error modelling for a mobile robot," in *Proceedings of International Conference on Robotics and Automation*, Albuquerque, NM, USA: IEEE, 1997, pp. 2783–2788. doi: 10.1109/ROBOT.1997.606708.
- [9] M. A. Ismail, D. Purwanto, and A. Arifin, "Soccer Robot Localization Based on Sensor Fusion From Odometry and Omnivision," in *2022 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, Surabaya, Indonesia: IEEE, Jul. 2022, pp. 273–278. doi: 10.1109/ISITIA56226.2022.9855313.
- [10] S. Huang, H.-Z. Huang, Q. Zeng, and P. Huang, "A Robust 2D Lidar SLAM Method in Complex Environment," *Photonic Sens.*, vol. 12, no. 4, p. 220416, Dec. 2022, doi: 10.1007/s13320-022-0657-6.
- [11] A. Howard, M. J. Matark, and G. S. Sukhatme, "Localization for mobile robot teams using maximum likelihood estimation," in *IEEE/RSJ International Conference on Intelligent Robots and System*, Lausanne, Switzerland: IEEE, 2002, pp. 434–439. doi: 10.1109/IRDS.2002.1041428.
- [12] S. Thrun, W. Burgard, and D. Fox, "PROBABILISTIC ROBOTICS".
- [13] F. R. Q. Aini, A. N. Jati, and U. Sunarya, "A study of Monte Carlo localization on robot operating system," in *2016 International Conference on Information Technology Systems and Innovation (ICITSI)*, Bandung - Bali, Indonesia: IEEE, Oct. 2016, pp. 1–6. doi: 10.1109/ICITSI.2016.7858235.
- [14] H. Abualkebash and H. Ocak, "Improved Global Localization and Resampling Techniques for Monte Carlo Localization Algorithm," *Int. J. Appl. Math. Electron. Comput.*, vol. 8, no. 3, pp. 102–108, Oct. 2020, doi: 10.18100/ijamec.800166.
- [15] A. Muzio, L. Aguiar, M. R. O. A. Maximo, and S. C. Pinto, "Monte Carlo Localization with Field Lines Observations for Simulated Humanoid Robotic Soccer," in *2016 XIII Latin American Robotics Symposium and IV Brazilian Robotics Symposium (LARS/SBR)*, Recife, Brazil: IEEE, Oct. 2016, pp. 334–339. doi: 10.1109/LARS-SBR.2016.63.
- [16] B. Sugandi and A. D. Zain, "Localization of Wheeled Soccer Robots Using Particle Filter Algorithm," in *2019 2nd International Conference on Applied Engineering (ICAE)*, Batam, Indonesia: IEEE, Oct. 2019, pp. 1–4. doi: 10.1109/ICAE47758.2019.9221670.

## BIOGRAPHIES OF AUTHORS

	<p>Mochamad Rizal Fauzi, lahir di Garut, Jawa Barat pada tanggal 01 September 2002. Sekarang sedang menjalani perkuliahan pada semester 7 di Politeknik Negeri Batam jurusan elektronika, prodi robotika. Sejak tahun 2020 aktif dalam tim robot Politeknik Negeri Batam. Pernah mengikuti kontes robot regional dan nasional pada tahun 2022 - 2023 di divisi KRSBI-beroda.</p>

## FORMULIR PENGAJUAN MAJU SIDANG DAN PEMBIMBING TUGAS AKHIR

Saya yang bertanda tangan di bawah ini:

Nama : Tajdar Hal Ata

NIM : 4222001001

Telah melaksanakan proses bimbingan dengan dosen pembimbing dalam perumusan Tugas Akhir/Proyek Akhir (TA/PA). Hasil proses pra bimbingan tersebut memutuskan :

Judul TA/PA : Kontrol dribel menggunakan PID untuk robot sepak bola beroda



Dosen Pembimbing 1 : Hendawan Soebhakti , S.T., M.T. [  ]

Judul tersebut telah diverifikasi dan disetujui oleh dosen pembimbing.

.

Batam, 20 Desember 2023

Mahasiswa



NIM: 4222001001

## FORMULIR PENGAJUAN MAJU SIDANG DAN PEMBIMBING TUGAS AKHIR

Saya yang bertanda tangan di bawah ini:

Nama : Haidar Afnan Fakhruddin

NIM : 4222001026

Telah melaksanakan proses bimbingan dengan dosen pembimbing dalam perumusan Tugas Akhir/Proyek Akhir (TA/PA). Hasil proses pra bimbingan tersebut memutuskan :

Judul TA/PA : Object Detection Using Omnidirectional Camera



Dosen Pembimbing 1 : Hendawan Soebhakti , S.T., M.T. [ ] Judul

tersebut telah diverifikasi dan disetujui oleh dosen pembimbing.

Batam, 20 Desember 2023

Mahasiswa,



NIM: 4222001026

## FORMULIR PENGAJUAN MAJU SIDANG DAN PEMBIMBING TUGAS AKHIR

Saya yang bertanda tangan di bawah ini:

Nama : Mochamad Rizal Fauzi

NIM : 4222001010

Telah melaksanakan proses bimbingan dengan dosen pembimbing dalam perumusan Tugas Akhir/Proyek Akhir (TA/PA). Hasil proses pra bimbingan tersebut memutuskan :

Judul TA/PA : Augmented Monte Carlo Localization

Dosen Pembimbing 1 : Hendawan Soebhakti , S.T., M.T. [  ]

Judul tersebut telah diverifikasi dan disetujui oleh dosen pembimbing.

.

Batam, 20 Desember 2023

Mahasiswa,



Mochamad Rizal Fauzi













NIM: 4222001010


.



**FORMULIR LOOGBOOK BIMBINGAN DAN PENGAJUAN  
SEMINAR PROPOSAL/SIDANG TUGAS AKHIR**

Nama : Tajdar Hal Ata  
 NIM : 4222001001  
 Pembimbing I : Hendawan Soebhakti , S.T., M.T.  
 Judul : Kontrol dribel menggunakan PID untuk robot sepak bola beroda

No	Hari/Tgl	Rincian Kegiatan	TTD Pembimbing I
1	07/09/2023	Penjelasan terkait tugas akhir	
2	14/09/2023	Diskusi untuk menentukan fokus penelitian	
3	21/09/2023	Evaluasi dan perbaikan rencana penelitian & identifikasi kerangka konsep	
4	28/09/2023	Diskusi tentang metodologi penelitian	
5	05/10/2023	Evaluasi dan perbaikan metodologi penelitian	
6	12/10/2023	Monitoring pengumpulan data	
7	19/10/2023	Analisis data secara berkala.	
8	26/10/2023	Diskusi hasil penelitian.	
9	02/11/2023	Diskusi hasil penelitian lanjutan.	
10	09/11/2023	Evaluasi hasil penelitian	
11	16/11/2023	Perbaikan hasil penelitian	
12	23/11/2023	Persiapan pra sidang.	

13	30/11/2023	Simulasi siding dengan pertanyaan pembimbing	
14	07/12/2023	Evaluasi kesiapan dan penyelesaian administasi & penetapan jadwal.	

Berdasarkan hasil bimbingan yang telah dilaksanakan selama 4 bulan dan telah disetujui oleh dosen pembimbing, maka dengan ini saya mengajukan diri sebagai peserta Seminar Proposal /Sidang Tugas Akhir\*.











Batam, 20 Desember 2023  
Mahasiswa



NIM: 4222001001

**FORMULIR LOGBOOK BIMBINGAN DAN PENGAJUAN  
SEMINAR PROPOSAL/SIDANG TUGAS AKHIR**

Nama : Haidar Afnan Fakhruddin  
 NIM : 4222001026  
 Pembimbing I : Hendawan Soebhakti , S.T., M.T.  
 Judul : Object Detection Using Omnidirectional Camera

No	Hari/Tgl	Rincian Kegiatan	TTD Pembimbing I
1	07/09/2023	Menentukan judul topik Tugas Akhir	
2	14/09/2023	Mencari referensi topic	
3	21/09/2023	Melakukan bimbingan dan membahas kesulitan yang terjadi pada topic yang diambil	
4	28/09/2023	Melakukan penelitian tentang topic yang dibahas	
5	05/10/2023	Memulai uji coba merubah pandangan omnidirectional camera ke panorama	
6	12/10/2023	Melakukan percobaan realtime camera pada robot	
7	19/10/2023	Melakukan pertemuan terkait perkembangan penelitian	
8	26/10/2023	Melakukan percobaan pendeteksian objek seperti bola dan obstacle	
9	02/11/2023	Melakukan integrasi sistem pandangan panorama dengan deteksi objek	
10	09/11/2023	Melakukan uji coba sistem keseluruhan	
11	16/11/2023	Melakukan bimbingan dan memperbaiki kekurangan sistem	
12	23/11/2023	Melakukan pengambilan data	
13	30/11/2023	Mengevaluasi data yang diambil	
14	07/12/2023	Melakukan finalisasi, menyusun buku Tugas Akhir dan memvalidasi dengan dosen pembimbing	

Berdasarkan hasil bimbingan yang telah dilaksanakan selama \_\_\_\_4\_\_\_\_ bulan dan telah disetujui oleh dosen pembimbing, maka dengan ini saya mengajukan diri sebagai peserta ~~Seminar Proposal~~ /Sidang Tugas Akhir\*.

Batam, 20 Desember 2023















Mahasiswa

A handwritten signature in black ink, appearing to be 'Hafid'.

NIM: 4222001026

**FORMULIR LOGBOOK BIMBINGAN DAN PENGAJUAN  
SEMINAR PROPOSAL/SIDANG TUGAS AKHIR**

Nama : Mochamad Rizal Fauzi  
 NIM : 4222001010  
 Pembimbing I : Hendawan Soebhakti , S.T., M.T.  
 Judul : Augmented Monte Carlo Localization for Wheeled Soccer Robot

No	Hari/Tgl	Rincian Kegiatan	TTD Pembimbing I
1	07/09/2023	Menentukan judul topik Tugas Akhir	
2	14/09/2023	Mencari referensi topic	
3	21/09/2023	Melakukan bimbingan dan membahas kesulitan yang terjadi pada topic yang diambil	
4	28/09/2023	Melakukan penelitian tentang topic yang dibahas	
5	05/10/2023	Pembahasan analisis data & Persiapan untuk pengumpulan data.	
6	12/10/2023	Monitoring pengumpulan data.	
7	19/10/2023	Analisis data yang telah terkumpul.	
8	26/10/2023	Diskusi hasil penelitian.	
9	02/11/2023	Diskusi hasil penelitian lanjutan.	
10	09/11/2023	Persiapan presentasi sidang.	
11	16/11/2023	Evaluasi dan perbaikan hasil penelitian.	
12	23/11/2023	Latihan presentasi siding & Diskusi untuk memastikan keakuratan dan kelengkapan presentasi.	
13	30/11/2023	Persiapan sidang	
14	07/12/2023	Persiapan sidang	

Berdasarkan hasil bimbingan yang telah dilaksanakan selama \_\_\_\_4\_\_\_\_ bulan dan telah disetujui oleh dosen pembimbing, maka dengan ini saya mengajukan diri sebagai peserta ~~Seminar Proposal~~ /Sidang Tugas Akhir\*.

Batam, 20 Desember 2023  
Mahasiswa

A handwritten signature in black ink, consisting of a stylized 'R' with a horizontal line through it.

NIM: 4222001010