

**HALAMAN JUDUL**

**PREDIKSI LOKASI RELATIF BERDASARKAN GAYA BERJALAN  
(Studi Kasus Robot Sepak Bola Humanoid Krakatau FC)**

**Relative Location Prediction Based on Gait Style  
(Case Study of Krakatau FC Humanoid Soccer Robot)**

**SKRIPSI**

Untuk memenuhi sebagian persyaratan  
Mencapai derajat sarjana S-1

Diusulkan oleh:  
**SHILVIA FEBRIANTI**  
19312208



**PROGRAM STUDI S1 INFORMATIKA  
FAKULTAS TEKNIK DAN ILMU KOMPUTER  
UNIVERSITAS TEKNOKRAT INDONESIA  
BANDAR LAMPUNG  
2023**

ACC Revisi:  
25/23  
/11  
Jaka BS

ACC Revisi:  
27/2023  
/11  
Pasar Kharisma

ACC Catuh:  
27/11-2023



**HALAMAN PENGESAHAN**

**SKRIPSI**

**PREDIKSI LOKASI RELATIF BERDASARKAN GAYA BERJALAN  
(STUDI KASUS ROBOT SEPAK BOLA KRAKATAU FC)**

Dipersiapkan dan disusun oleh:

**SHILVIA FEBRIANTI**

**19312208**

Telah dipertahankan di depan Dewan Penguji

Pada tanggal 14 November 2023

Pembimbing,

Penguji,

**M. Pajar Kharisma, S.Kom., M.Cs**

**NIK. 022 10 03 02**

**Jaka Persada S, S.Kom., M.Cs**

**NIK. 021 14 02 06**

Skrripsi ini telah diterima sebagai salah satu persyaratan  
untuk memperoleh gelar sarjana  
Tanggal 27 November 2023

Fakultas Teknik dan Ilmu Komputer

Program Studi S1 Informatika

Dekan,

Ketua,



**Dr. H. Mahathir Muhammad, S.E., M.M.**

**NIK. 023 05 00 09**

**Dyah Ayu Megawaty, M.Kom**

**NIK. 022 09 03 05**

## LEMBAR PERNYATAAN

Yang bertanda tangan di bawah ini:

Nama : Shilvia Febrianti

NPM : 19312208

Program Studi : SI Informatika

Dengan ini menyatakan bahwa tugas akhir :

Judul : Prediksi Lokasi Relatif Berdasarkan Gaya Berjalan  
(Studi Kasus Robot Sepak Bola Humanoid)

Pembimbing : M. Pajar Kharisma Putra, S.Kom., M.Cs.

Belum pernah diajukan untuk diuji sebagai persyaratan untuk memperoleh gelar akademik pada berbagai tingkatan di universitas/ perguruan tinggi manapun. Tidak ada bagian dalam skripsi ini yang pernah dipublikasikan oleh pihak lain, kecuali bagian yang digunakan sebagai referensi, berdasarkan kaidah penulisan ilmiah yang benar.

Apabila dikemudian hari ternyata laporan tugas akhir yang saya tulis terbukti hasil saduran/plagiat, maka saya akan bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan sebenar-benarnya.

Bandar Lampung, 14 November 2023

Yang menyatakan,

  
  
Shilvia Febrianti  
NPM. 19312208

**HALAMAN PERNYATAAN  
PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN  
AKADEMIS**

Sebagai sivitas akademik Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Shilvia Febrianti  
NPM : 19312208  
Program Studi : SI Informatika  
Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia, **Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty-Free Right)** atas karya ilmiah saya yang berjudul :

**Prediksi Lokasi Relatif Berdasarkan Gaya Berjalan (Studi Kasus Robot Sepak Bola Humanoid)**

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Bandar Lampung  
Pada tanggal : 14 November 2023

Yang menyatakan,



Shilvia Febrianti  
NPM. 19312208

## LEMBAR PERSEMBAHAN



Persembahan skripsi ini aku persembahkan kepada :

Allah SWT sang pencipta alam semesta serta isinya, yang memberikan karunia kesehatan, rahmat dan kasih sayang-Nya sehingga dapat menyelesaikan skripsi dengan lancar.

Kepada kedua orang tua yang saya sayangi, yaitu Ibu Sumini dan Ayah Jaimun yang tak henti memberikan doa, dukungan, dan harapan diberikan kepada saya semoga sehat selalu.

Terima kasih kepada sasanak udara tercinta, rekan-rekan seperjuangan yang selalu mendukung saya sehingga saya berhasil menyelesaikan skripsi saya.

Terima kasih kepada dosen pembimbing saya Bapak M. Pajar Kharisma Putra, S.Kom., M.Cs. yang telah memberikan bimbingan dan motivasi kepada saya.

Terimakasih kepada teman-teman ukm robotika khususnya Tim Krakatau FC yang telah memberikan semangat dan dukungannya

## KATA PENGANTAR

Segala puji dan syukur kehadirat Allah SWT atas berkah, rahamat dan hidayah-Nya yang senantiasa dilimpahkan kepada penulis, sehingga bisa menyelesaikan skripsi dengan judul “Prediksi Lokasi Relatif Berdasarkan Gaya Berjalan (Studi Kasus Robot Sepak Bola Humanoid Krakatau FC)” sebagai syarat untuk menyelesaikan Program Sarjana (S1) pada Program Studi Teknik Komputer Fakultas Teknik dan Ilmu Komputer Universitas Teknokrat Indonesia.

Pada kesempatan ini, penulis menyampaikan ucapan terimakasih kepada:

1. Bapak Dr. H.M. Nasrullah Yusuf, S.E., M.B.A selaku Rektor Universitas Teknokrat Indonesia
2. Bapak Dr. H. Mahathir Muhammad, SE., MM selaku Dekan Fakultas Teknik Universitas Teknokrat Indonesia
3. Dyah Ayu Megawaty, S.kom., M.Kom selaku Ketua Program Studi Teknik Komputer Universitas Teknokrat Indonesia
4. Bapak M.Pajar Kharisma Putra, S.Kom., M.Cs selaku Dosen Pembimbing yang telah bersedia memberikan arahan dan bimbingan selama penelitian berlangsung
5. Bapak Jaka Persada Sembiring S.Kom., M.Cs. selaku Dosen Penguji yang telah memberikan saran dan nasihat demi kebaikan penelitian
6. Kedua orang tua saya Ibu Sumini dan Bapak Jaimun yang telah memberikan dukungan dan doa yang tidak ternilai
7. Kepada sanak saudara dan rekan-rekan mahasiswa serta rekan-rekan Tim Krakatau FC yang telah membantu proses penelitian

Penulis mohon maaf atas segala kesalahan yang pernah dilakukan. Semoga skripsi ini dapat memberikan manfaat untuk mendorong penelitian-penelitian selanjutnya.

Bandar Lampung, 14 November 2023

Penulis

## DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PENGESAHAN .....	ii
LEMBAR PERNYATAAN .....	iii
HALAMAN PERNYATAAN .....	iv
LEMBAR PERSEMBAHAN .....	v
KATA PENGANTAR .....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR .....	x
ABSTRAK .....	xii
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	3
1.3 Batasan Masalah Penelitian.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
1.6 Sistematika Penulisan.....	4
BAB II LANDASAN TEORI .....	5
2.1 Tinjauan Pustaka .....	5
2.1.1 Tinjauan Pustaka 1 .....	7
2.1.2 Tinjauan Pustaka 2 .....	7
2.1.3 Tinjauan Pustaka 3 .....	8
2.1.4 Tinjauan Pustaka 4 .....	8
2.1.5 Tinjauan Pustaka 5 .....	9
2.2 Dead Reckoning .....	9
2.3 Robot Humanoid .....	10
2.4 Platform Robot .....	11
2.5 Framework Krakatau .....	12
2.6 Konsep Berjalan Pada Robot Humanoid.....	14
2.7 Transformasi Geometri.....	15
2.7.1 Translasi (Pergeseran).....	16
2.7.2 Rotasi (Perputaran).....	17
2.8 Sistem Koordinat Kartesian.....	19

BAB III METODE PENELITIAN.....	20
3.1 Tahapan Penelitian .....	20
3.2 Metode Pengumpulan Data .....	21
3.3 Waktu dan Tempat Penelitian .....	21
3.4 Objek Penelitian .....	21
3.4 Alat dan Bahan Penelitian .....	22
3.5.1 Perangkat Keras (Hardware) .....	22
3.5.2 Perangkat Lunak (Software) .....	25
3.6 Penelitian .....	25
3.6.1 Rancangan Arsitektur Sistem Robot .....	25
3.6.2 Rancangan Pola Berjalan Robot .....	26
3.6.2 Rancangan Sistem Prediksi Lokasi .....	32
3.7 Rancangan Skema Pengujian .....	33
3.8 Jadwal Penelitian .....	34
BAB IV HASIL DAN PEMBAHASAN .....	35
4.1 Proses Tuning Offset Tuner .....	35
4.2 Kalibrasi Sensor Kompas .....	36
4.3 Hasil Pengujian.....	38
4.3.1 Pengujian Berjalan Maju.....	38
4.3.2 Pengujian Berjalan Mundur .....	42
4.3.3 Pengujian Berjalan Ke Kanan .....	46
4.3.4 Pengujian Berjalan Ke Kiri .....	51
4.3.5 Hasil Pengujian Keseluruhan .....	56
BAB V KESIMPULAN DAN SARAN.....	58
5.1 Kesimpulan.....	58
5.2 Saran .....	59
DAFTAR PUSTAKA .....	60



## DAFTAR TABEL

<b>Tabel 2. 1</b> Tinjauan Pustaka .....	5
<b>Tabel 3.1</b> Perangkat Keras Yang Digunakan.....	23
<b>Tabel 3.2</b> Parameter Robot Krakatau .....	29
<b>Tabel 3.3</b> Rancangan Skema Pengujian .....	33
<b>Tabel 3. 4</b> Jadwal Penelitian.....	34
<b>Tabel 4. 1</b> Posisi Aktual Robot Terhadap Target Maju.....	41
<b>Tabel 4. 2</b> Posisi Arah Hadap .....	42
<b>Tabel 4. 3</b> Posisi Aktual Robot Terhadap Target Mundur .....	45
<b>Tabel 4. 4</b> Posisi Arah Hadap .....	46
<b>Tabel 4. 5</b> Posisi Aktual Robot Terhadap Target Ke Kanan .....	49
<b>Tabel 4. 6</b> Posisi Arah Hadap .....	50
<b>Tabel 4. 7</b> Posisi Aktual Robot Terhadap Target Ke Kiri .....	54
<b>Tabel 4. 8</b> Posisi Arah Hadap .....	55
<b>Tabel 4. 9</b> Hasil Pengujian Keseluruhan Jarak Tempuh.....	56
<b>Tabel 4. 10</b> Hasil Pengujian Keseluruhan Arah Hadap.....	56

## DAFTAR GAMBAR

<b>Gambar 2.1</b> Kerangka Kerja Platform Darwin-OP .....	13
<b>Gambar 2.2</b> Gait Cycle Selama Satu Siklus. ....	14
<b>Gambar 2.3</b> Ilustrasi Gaya Berjalan Robot.....	15
<b>Gambar 2.4</b> Contoh Transformasi Translasi.....	17
<b>Gambar 2.5</b> Contoh Transformasi Rotasi .....	19
<b>Gambar 2.6</b> Koordinat Kartesian.....	19
<b>Gambar 3.1</b> Tahapan Penelitian .....	20
<b>Gambar 3.2</b> Dimensi Robot Krakatau FC .....	22
<b>Gambar 3.3</b> Diagram Blok Sistem Kontrol Pada Robot.....	26
<b>Gambar 3.4</b> Sendi Pada Robot Berkaki Dua .....	27
<b>Gambar 3.5</b> (a) Gerak Dari Sisi ke Sisi, (b) Gerak Melangkah Ke Samping, (c) Gerak Melangkah Ke Depan, (d) Gerak Berputar, Dan (e) Gerak Mengangkat Kaki. .....	27
<b>Gambar 3.6</b> Gerakan Melangkah X-Axis .....	28
<b>Gambar 3.7</b> Gerakan Melangkah Y-Axis .....	28
<b>Gambar 3.8</b> Ilustrasi Letak CoM Pada Saat Berjalan.....	29
<b>Gambar 3.9</b> Rancangan Sistem Berdasarkan Walking Engine.....	32
<b>Gambar 3.10</b> Rancangan Sistem Berdasarkan Kondisi Robot .....	33
<b>Gambar 4.1</b> Proses Pengaturan <i>Offset Tuner</i> .....	35
<b>Gambar 4.2</b> Parameter Pengaturan <i>Offset Tuner</i> .....	36
<b>Gambar 4.3</b> Proses Kalibrasi Kompas .....	37
<b>Gambar 4.4</b> Hasil Pengujian Berjalan Maju (a) Target 50 cm, (b) Target 100 cm, (c) Target 150 cm, dan Target 200 cm. ....	40

<b>Gambar 4.5</b> Grafik Hasil Pengujian Berjalan Maju .....	40
<b>Gambar 4.6</b> Hasil Pengujian Berjalan Mundur (a) Target 50 cm, (b) Target 100 cm, (c) Target 150 cm, dan Target 200 cm. ....	44
<b>Gambar 4.7</b> Grafik Hasil Pengujian Berjalan Mundur .....	45
<b>Gambar 4.8</b> Hasil Pengujian Berjalan Ke Kanan (a) Target 50 cm, (b) Target 100 cm, (c) Target 150 cm, dan Target 200 cm. ....	48
<b>Gambar 4.9</b> Grafik Hasil Pengujian Berjalan Ke Kanan.....	49
<b>Gambar 4.10</b> Hasil Pengujian Berjalan Ke Kiri (a) Target 50 cm, (b) Target 100 cm, (c) Target 150 cm, dan Target 200 cm. ....	53
<b>Gambar 4.11</b> Grafik Hasil Pengujian Berjalan Ke Kiri.....	54

## **ABSTRAK**

### **PREDIKSI LOKASI RELATIF BERDASARKAN GAYA BERJALAN (Studi Kasus Robot Sepak Bola Humanoid Krakatau FC)**

**Shilvia Febrianti  
19312208**

Robot sepak bola humanoid adalah perangkat mekanik yang dirancang untuk bermain sepak bola dengan kemampuan yang menyerupai manusia. Salah satu aspek penting dalam pengembangan robot sepak bola adalah kemampuan lokalisasi. Hal ini mengacu pada kemampuan robot untuk memproyeksikan dan memprediksi posisi relative terhadap bola, rekan satu tim dan lawan dalam permainan. Dengan kemampuan ini robot dapat mengambil keputusan yang lebih cerdas, mengejar bola, dan berkoordinasi dengan rekan satu tim secara efisien. Dead reckoning adalah salah satu teknik lokalisasi yang termasuk kedalam kategori relatif. Robot milik Krakatau FC belum mampu dalam memprediksi posisi secara akurat, prediksi posisi bertujuan untuk memberikan informasi posisi guna menyelesaikan misi dengan baik. Penelitian ini memanfaatkan prinsip perhitungan dead reckoning untuk mengetahui posisi robot dari titik awal sampai ke titik akhir berdasarkan posisi yang diketahui sebelumnya. Berdasarkan hasil pengujian metode yang diusulkan sudah dapat diimplementasikan serta metode tersebut telah diuji dan hasil pengujian tersebut memverifikasi bahwa layak digunakan walaupun masih terdapat nilai error tetapi yakni sebesar 5,3% pada jarak tempuh dan 6,3% pada arah hadap robot. penelitian ini menunjukkan meskipun terdapat beberapa ketidakakuratan, metode tersebut masih memberikan hasil yang dapat diandalkan dan cukup akurat dalam memprediksi lokasi relatif.

**Kata Kunci:** Robot Sepak Bola Humanoid, Lokalisasi, Dead Reckoning, Lokasi Relatif, Kemampuan Robot.

## **BAB I PENDAHULUAN**

### **1.1 Latar Belakang**

Robot Sepak Bola Humanoid merupakan robot dengan penampilan seperti manusia dimana terdiri dari sebuah kepala, dua buah lengan, batang tubuh, dan dua buah kaki, robot dirancang memiliki kemampuan dasar yang sama dengan milik manusia. Robot Sepak Bola tidak hanya dituntut dapat berjalan secara stabil dan cepat tetapi juga dapat melakukan berbagai tindakan kompleks serta mengidentifikasi berbagai lingkungan target statis maupun dinamis secara akurat dan real-time, melakukan koordinasi antar robot, mengetahui keberadaan posisinya dalam suatu lingkungan dan dapat mengambil keputusan dalam konteks permainan sepak bola (Hu et al., 2021). Tujuan utama dalam pengembangan robot sepak bola yaitu menguji kemampuan robot dalam menghadapi dan mengatasi berbagai tantangan kompleks.

Kemampuan mengenali posisi dan lingkungan sekitarnya disebut dengan lokalisasi robot (robot localization). Secara umum lokalisasi robot dibagi menjadi tiga kategori yaitu relative atau local localization, absolut atau global localization, dan probabilistic localization (Adriansyah, 2014). Dead reckoning merupakan satu dari bermacam-macam teknik lokalisasi yang masuk kedalam kategori relative localization. Teknik ini digunakan untuk memperkirakan posisi relatif robot berdasarkan pada posisi yang telah diketahui sebelumnya, serta pengukuran kecepatan yang digunakan untuk memperbarui posisi robot secara bertahap (Bonarini et al., 2011). Oleh karena itu, metode dead reckoning digunakan sebagai fundamental dalam penelitian robot bergerak karena metode ini sederhana, mudah

diterapkan dan dapat mengestimasi posisi secara real time tetapi kelemahan utama dari metode dead reckoning adalah akumulasi kesalahan yang tidak terbatas (Cho et al., 2011). Kemampuan dalam mengenali posisi dan lingkungan sangat dibutuhkan pada robot bergerak secara otonom karena robot perlu memahami lingkungan untuk dapat menentukan keberadaan posisinya, merencanakan rute, dan menghindari rintangan sehingga robot dapat bergerak ke posisi yang telah ditentukan (Tai et al., 2014). Secara umum keberhasilan system navigasi berbasis algoritma Dead Reckoning, ditentukan oleh tiga aktifitas yaitu, deteksi langkah, estimasi panjang langkah, dan estimasi arah langkah (Setiawan et al., 2012).

Penelitian serupa pernah dilakukan oleh (Li et al., 2015) yang berjudul A Humanoid Robot Localization Method for Biped Navigation in Human-Living Environments. Kesimpulan dari penelitian tersebut yaitu Li dkk telah berhasil mengimplementasikan metode lokalisasi berbasis visual dalam perencanaan langkah kaki real-time di lingkungan manusia. Hasil eksperimen tersebut telah mengkonfirmasi kelayakan metode yang diusulkan. Dengan kata lain, metode tersebut dapat bekerja dengan baik dalam konteks perencanaan langkah kaki real-time dalam lingkungan yang dinamis dan dihuni oleh manusia.

Oleh karena itu penulis mengusulkan untuk dilakukannya penelitian terhadap prediksi lokasi relative berdasarkan gaya berjalan dengan memanfaatkan gerakan yang dihasilkan oleh robot. Penelitian ini diimplementasikan pada robot sepak bola yang mana ruang robot bergerak di area lapangan berumput. Hal ini bertujuan agar dapat mengetahui kemana arah estimasi robot berjalan dengan keadaan yang sebenarnya. Dari gerakan berjalan akan menghasilkan informasi posisi robot berupa nilai X, Y dan Orientasi. Dari hasil data yang diperoleh tersebut

selanjutnya akan dilakukan pengukuran dengan estimasi posisi untuk mengetahui hasil error yang didapat oleh metode tersebut.

## **1.2 Rumusan Masalah**

Berdasarkan pemaparan yang telah diuraikan dalam latar belakang diatas, maka peneliti merumuskan beberapa masalah pokok dalam penelitian yaitu:

1. Bagaimana cara memprediksi lokasi relatif hanya dengan memanfaatkan gaya berjalan robot.
2. Bagaimana mengetahui keakurasian robot dalam menyelesaikan misi.

## **1.3 Batasan Masalah Penelitian**

Adapun batasan masalah dalam penelitian ini yaitu:

1. Penelitian ini berfokus pada bagian kaki dan gaya berjalan yang dihasilkan robot.
2. Pengujian dilakukan pada framework Krakatau
3. Tidak membahas kinematika robot
4. Objek penelitian yaitu robot sepak bola Krakatau FC

## **1.4 Tujuan Penelitian**

Tujuan yang ingin dicapai dari penelitian ini yaitu untuk mengetahui keakurasian robot dalam menjalankan perintah dan analisa terhadap metode yang digunakan.

## **1.5 Manfaat Penelitian**

Adapun manfaat dari penelitian ini adalah :

1. Hasil dari penelitian ini dapat digunakan sebagai bahan acuan pada penelitian selanjutnya.

2. Dapat dijadikan referensi bagi penelitian dibidang robotika khususnya sistem navigasi pada robot berkaki.

## **1.6 Sistematika Penulisan**

Sistematika penulisan bertujuan untuk memperjelas dan mempermudah penulisan. Adapun sistematika penulisan pada penelitian ini yaitu:

### **BAB 1 PENDAHULUAN**

Pendahulian berisi : Latar belakang, Rumusan Masalah, Batasan Masalah, Tujuan Masalah, Manfaat Penelitian, Tahap Penelitian dan Sistematika Penulisan.

### **BAB II LANDASAN TEORI**

Landasan teori berisi : teori pendukung lainnya yang terkait dengan objek penelitian

### **BAB III METODE PENELITIAN**

Metode Penelitian berisi : penjelasan detail terkait metode penelitian dan rancangan yang dibuat.

### **BAB IV PEMBAHASAN DAN HASIL**

Pembahasan dan Hasil berisi : pemaparan hasil yang diperoleh dari implementasi sistem berdasarkan pengujian yang telah dilakukan, kemudian akan dilakukan analisa untuk memperoleh informasi yang dibutuhkan

### **BAB V KESIMPULAN DAN SARAN**

Kesimpulan dan Saran berisi : menguraikan kesimpulan dari pembahasan yang telah dilakukan dalam penelitian dan saran yang diberikan oleh penulis untuk penelitian dimasa mendatang.

### **DAFTAR PUSTAKA**



## BAB II LANDASAN TEORI

### 2.1 Tinjauan Pustaka

Dalam penelitian ini peneliti menggunakan tinjauan pustaka pada penelitian sebelumnya yang dapat digunakan sebagai pendukung dalam penelitian, berikut merupakan tinjauan pustaka dalam penelitian ini:

**Tabel 2. 1** Tinjauan Pustaka

Nomor Tinjauan Pustaka	Penulis	Tahun	Judul
Tinjauan Pustaka 1	Ken Masuya dan Tomomichi Sugihara	2015	Dead Reckoning for Biped Robots that Suffers Less from Foot Contact Condition Based on Anchoring Pivot Estimation
Tinjauan Pustaka 2	Michal Reinstein dan Matej Hoffman	2011	Dead Reckoning in a Dynamic Quadrupped: Inertial Navigation System Aided by a Legged Odometer

**Tabel 2.1** Tinjauan Pustaka (Lanjutan)

Nomor Tinjauan Pustaka	Penulis	Tahun	Judul
Tinjauan Pustaka 3	Huang Li, Jing Xiong, Yangzhou Gan, Zeyang Xia	2015	A Humanoid Robot Localization Method for Biped Navigation in Human-Living Environments
Tinjauan Pustaka 4	Andi Adriansyah	2014	Perancangan Robot Localization Menggunakan Metode Dead Reckoning
Tinjauan Pustaka 5	Diah Liani, Ade Silvia, dan Lindawati	2017	Sistem Navigasi Pada Mobile Robot Dalam Penentuan Arah dan Pemetaan Posisi

### 2.1.1 Tinjauan Pustaka 1

Pada penelitian ini membahas tentang teknik perhitungan *dead reckoning* untuk umpan balik pada robot berkaki dua. Estimasi posisi pada robot diperoleh dengan menggabungkan informasi hanya dari sensor *internal* termasuk penyandi sudut sendi, sensor inersia, dan sensor gaya. Hal ini menggabungkan perhitungan kinematika dan integral ganda dengan cara yang saling melengkapi untuk meningkatkan akurasi. Perhitungan kinematika mempertimbangkan pergerakan kaki penyangga, khususnya, rotasi di sekitar titik tetap dan berguling di medan. Bobot pada setiap informasi disesuaikan secara otomatis berdasarkan gaya reaksi dari tanah. Melalui simulasi ditunjukkan bahwa metode ini dapat bekerja dengan baik sekalipun pada situasi kaki pendukung bergerak sehubungan dengan tanah dan bahkan tidak memiliki kontak dengan tanah (Masuya et al., 2015).

### 2.1.2 Tinjauan Pustaka 2

Pada penelitian ini membahas algoritma navigasi dalam mengestimasi postur dan jarak yang ditempuh oleh robot berkaki empat dengan menggunakan kombinasi algoritma mekanisasi Sistem Navigasi Inersia (INS) dan odometri berkaki. *Extended Kalman Filter* (EKF) digunakan untuk estimasi kesalahan dan penggabungan data dari dua sumber informasi yang independen ini. INS memproses data inersia mentah untuk memberikan informasi tentang posisi, kecepatan, dan sikap robot. Disisi lain, odometri berkaki menggunakan kombinasi sinyal sensor sendi dan sensor tekanan untuk memperkirakan kecepatan robot. Sistem navigasi ini memastikan pelacakan yang tepat dari postur robot (*roll* dan *pitch*) dan

pelacakan yang memuaskan dari posisinya dalam *interval* waktu menengah (Reinstein et al., 2011).

### 2.1.3 Tinjauan Pustaka 3

Pada penelitian ini membahas tentang perencanaan langkah kaki secara *real-time* untuk robot *humanoid*, diperlukan agar mendapatkan lokasi yang akurat dari penempatan kakinya. Dalam penelitian ini menyajikan metode lokalisasi untuk penempatan kaki pada robot *humanoid* yang bernavigasi pada lingkungan manusia. Sensor *Kinect* digunakan pertama kali untuk mendapatkan peta digital dari lingkungan *global* dan posisi serta orientasi kepala robot diekstraksi. Metode yang diusulkan digunakan dalam perencanaan langkah kaki secara *real-time* di lingkungan yang dinamis. Metode yang disajikan telah diuji pada platform robot dalam sebuah adegan eksperimental. Hasil pengujian memverifikasi bahwa metode yang disajikan layak untuk lokalisasi langkah kaki dan perencanaan langkah kaki pada robot berkaki dua (Li et al., 2015).

### 2.1.4 Tinjauan Pustaka 4

Pada penelitian ini membahas perancangan *mobile localization* menggunakan metode *dead reckoning*. Dalam merancang robot bergerak terdapat sebuah permasalahan, yaitu dimana robot dapat mengetahui posisinya dari waktu ke waktu. Permasalahan ini biasa disebut sebagai sistem lokalisasi robot. *Dead reckoning* adalah teknik lokalisasi robot bergerak yang memanfaatkan hasil dari data sensor bergerak untuk memperkirakan perubahan posisi robot dari waktu ke waktu sepanjang robot bergerak. Berdasarkan hasil pengujian, dapat dikatakan bahwa robot hasil

rancangan dapat bergerak dengan pergerakan yang sesuai dengan program yang dikehendaki, baik pergerakan lurus maupun pergerakan berbelok. Namun, terjadi penyimpangan posisi baik pada sumbu X maupun pada sumbu Y. Hal ini dikarenakan slip pada masing-masing roda yang menyebabkan akumulasi penyimpangan posisi secara keseluruhan (Adriansyah, 2014).

### 2.1.5 Tinjauan Pustaka 5

Pada penelitian ini dijelaskan bahwa *mobile* robot menggunakan sebuah *compass* sebagai penentu arah sehingga robot memiliki kemampuan untuk mengatur arah gerak untuk dapat menuju arah yang dituju. Untuk melakukan gerak kearah yang dituju yaitu menggunakan metode *waypoints*. Hasil dari penelitian yang telah dilakukan peneliti yaitu modul *compass* memiliki selisih rata-rata sebesar  $4^\circ$  setiap *interval* 5 detik antara robot 1, 2, dan 3, dan selisih sebesar  $3^\circ$  pada sensor *compass* untuk dapat mencapai gerak rotasi yang ditargetkan. Pembacaan nilai *compass* juga dipengaruhi dengan benda yang ada di sekitarnya sehingga pergerakan robot tidak sesuai ketika dijalankan di dalam area *indoor* dan akan lebih baik jika dilakukan di area *outdoor* (Liani et al., 2017)

## 2.2 Dead Reckoning

*Dead Reckoning* adalah salah satu metode lokalisasi yang termasuk dalam kategori *relative* atau *local localization*, yang digunakan untuk mengestimasi posisi saat ini dan perubahan posisi dengan menghitung jarak yang telah ditempuh dari titik sebelumnya yang telah diketahui. Metode ini diterapkan secara bertahap untuk memperbarui posisi serta orientasi pada robot sepanjang robot

bergerak (Setiawan et al., 2012) . Hal ini mengacu pada metode penentuan posisi *relative* yang sederhana, mudah diterapkan dan dapat memberikan informasi posisi secara nyata.

Algoritma *dead reckoning* adalah metode navigasi yang dapat memperkirakan posisi tanpa menggunakan sinyal GPS atau sinyal sedang tidak tersedia, biasanya sinyal tidak tersedia ketika berada di dalam ruangan sebuah gedung, di lautan, atau pada saat sedang berada di udara. Metode ini memanfaatkan gabungan data dari dua sensor yaitu *accelerometer* dan sensor *compass*, yang kemudian dilakukan perhitungan sehingga dapat menghasilkan perkiraan posisi baru berdasarkan posisi awal yang telah diketahui. Metode *dead reckoning* sudah banyak diimplementasikan diantaranya adalah untuk navigasi pesawat terbang pada saat mengudara, navigasi pada kapal laut yang sedang menjelajahi lautan, dan navigasi bagi pejalan kaki atau lebih dikenal sebagai pedestarian *dead reckoning*. Algoritma pedestarian *dead reckoning* adalah algoritma yang memanfaatkan kinematika gaya berjalan pada manusia untuk dapat memberikan perkiraan posisi pada pejalan kaki dengan menghitung jarak yang telah ditempuh terhadap posisi awal yang telah diketahui (Ramadhana et al., 2013).

### **2.3 Robot Humanoid**

Robot *humanoid* adalah jenis robot *autonomus* yang memiliki bentuk fisik secara keseluruhan menyerupai manusia dan dirancang memiliki fungsi dasar yang sama dengan milik manusia. Secara umum robot *humanoid* memiliki struktur tubuh manusia dilengkapi dengan sebuah kepala, dua buah lengan dan dua buah kaki. Meskipun demikian tidak semua jenis robot *humanoid* memiliki bentuk seperti layaknya manusia tetapi juga ada beberapa robot *humanoid* yang memiliki bentuk

hanya sebagian tubuh manusia dimulai dari pinggang hingga ke atas atau hanya berbentuk lengan saja. Selain itu robot *humanoid* juga ada yang dilengkapi dengan wajah beserta mata dan mulut layaknya manusia (Erdinal et al., 2021).

Robot *humanoid* dirancang untuk dapat melakukan beberapa tugas fisik dan mental yang sama dengan manusia dalam menjalankan kehidupan sehari-harinya. Tujuan para ilmuwan dalam menciptakan dan mengembangkan robot *humanoid* adalah agar robot dapat memahami kecerdasan akal milik manusia serta dapat bertindak dan mengambil keputusan layaknya manusia sehingga robot *humanoid* memiliki potensi untuk melakukan berbagai tugas di lingkungan manusia. Robot *humanoid* memiliki kemampuan bergerak secara autonomus, dapat mengumpulkan informasi pada lingkungan manusia dengan memanfaatkan sensor-sensor seperti (*Compass* dan GPS) dan berinteraksi dengan informasi-informasi yang diperoleh dari sensor tersebut sehingga robot tidak hanya diam ditempat seperti robot manipulator yang bekerja di tempat terstruktur sehingga robot *humanoid* harus memiliki perencanaan jalur dan penghindaran rintangan untuk dapat bergerak di dalam lingkungan kompleks.

#### **2.4 Platform Robot**

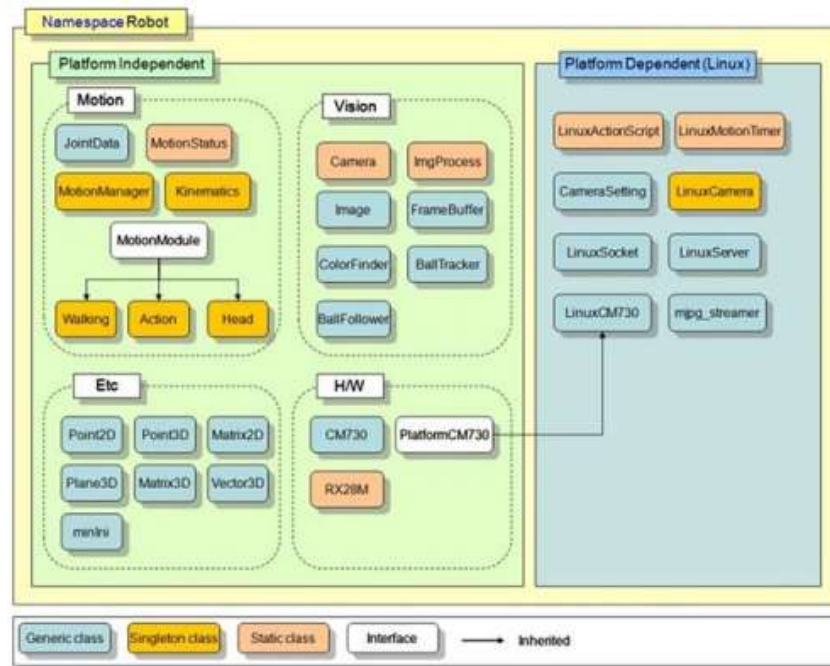
Robot sepak bola *humanoid* milik tim Krakatau FC dibangun dengan mereferensi pada platform robot Darwin-OP. Robot Krakatau di desain dengan rangka *alloy* dan memiliki 20 DOF (*Degree of Freedom*). Pada setiap DOF digerakkan menggunakan sebuah aktuator berupa servo motor. Servo sendiri diibaratkan sebagai sendi pada manusia. Robot Krakatau menggunakan 3 buah jenis servo yaitu servo MX-28T, MX-64T, dan MX-106. Pada bagian kepala terdapat 2 sendi dan setiap lengan terdapat 3 sendi, masing-masing digerakkan menggunakan servo MX-28T

sedangkan pada sendi bagian pinggang sampai dengan kaki digerakkan menggunakan gabungan dari servo MX-64T dan MX-106. Sistem kepala robot dirancang dalam 2 DOF, masing-masing untuk mendukung fungsi angguk (*Tilt*) dan fungsi geleng (*Pan*). Kedua fungsi tersebut digunakan untuk mengarahkan sistem sensor utama, yakni kamera digital PS Eye, untuk mengenali lingkungan sekitar yang dibutuhkan seperti melacak keberadaan bola dan gawang. Robot Krakatau diotaki oleh komputer papan tunggal intel NUC10i3FNH dengan *processor* intel core i3 menggunakan sistem operasi Ubuntu. *Platform* perangkat lunak pada robot Krakatau diadopsi dari *platform* Darwin yang merupakan sebuah *open platform* dan dibangun secara hirarkis dengan mempertimbangkan modularitas dan independensi.

## 2.5 Framework Krakatau

*Framework* Krakatau merupakan *framework* robot sepak bola *humanoid* yang dikembangkan oleh tim robot sepak bola Krakatau FC Universitas Teknokrat Indonesia. *Framework* Krakatau dibuat berdasarkan *framework* Darwin yang merupakan sebuah *open platform*. Robot sepak bola *humanoid* milik tim Krakatau FC secara keseluruhan mengadopsi *platform* Darwin-OP dimulai dari perancangan mekanika, gaya berjalan pada robot hingga perangkat lunak yang digunakan. *Framework* Darwin-OP terdiri dari modul komunikasi perangkat, modul gerak, modul jalan kaki, modul penginderaan, modul perilaku, modul penglihatan, dan modul diagnostik. Kerangka kerja ini telah dikembangkan dengan bahasa pemrograman C++ dimana kodenya beroperasi dengan sistem-independen (Ha et al., 2013). Berikut adalah kerangka kerja perangkat lunak pada *platform* Darwin-OP dapat dilihat pada gambar 2.1.





**Gambar 2.1** Kerangka Kerja Platform Darwin-OP

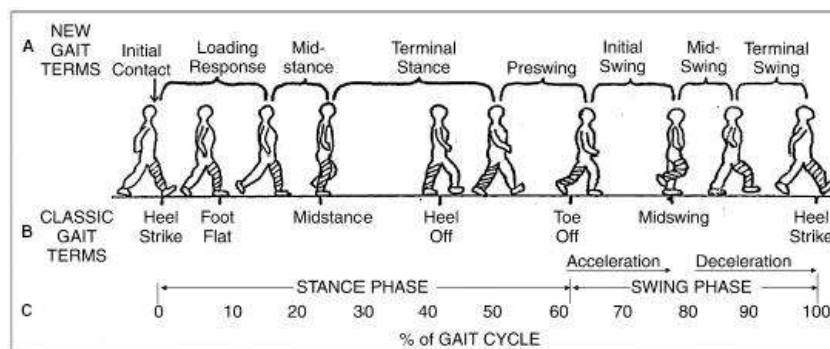
Sumber: (Ha et al., 2013)

*Framework* Krakatau sendiri dikembangkan pertama kali pada tahun 2013. Sejak dikembangkan, *framework* Krakatau telah mengalami berbagai perubahan dibandingkan baseline (*framework* Darwin). Setelah 2 tahun dalam masa pengembangan akhirnya pada tahun 2015 *framework* Krakatau versi 1.0 yang merupakan versi stabil dirilis. Pada versi 1.0 tim pengembang menambahkan fitur deteksi bola berdasarkan 2 parameter yaitu warna lapangan dan warna bola sehingga robot dapat mengeliminasi *noise* yang berada di luar lapangan. Satu tahun setelahnya pada tahun 2016 *framework* Krakatau dikembangkan kembali serta melakukan *update* pada bagian *trajectory*. Pada tahun yang sama dirilis kembali *framework* Krakatau versi 1.5 ini dimana pada versi ini tim pengembang menambahkan *balancing* pada *trajectory* robot sehingga robot dapat merespon ketidakseimbangan dengan cara menggerakkan *offset actuator* tubuh bagian atas serta menambahkan pola gerak dinamis pada robot. Pada tahun 2017 *framework*

Krakatau kembali merilis versi 2.0 dengan *update* yang cukup signifikan diantaranya deteksi bola dan gawang yang sudah menggunakan opencv serta komunikasi *multicast* dengan rekan satu tim.

## 2.6 Konsep Berjalan Pada Robot Humanoid

Robot *humanoid* memiliki mekanisme berjalan *bipedal*. Untuk menciptakan robot yang dapat berjalan dengan dua kaki terdapat berbagai kendala dalam mewujudkannya. Gerakan berjalan pada robot *bipedal* dimodelkan berdasarkan gerakan berjalan manusia yang digambarkan sebagai urutan gerak berulang. Gerakan ini disebut sebagai *gait cycle*. *Gait cycle* terdiri dari dua fase yaitu *stance phase* dan *swing phase*. Fase *stance* meliputi proses *initial contact*, *foot-flat* atau *loading response* yaitu permulaan kaki kontak dengan tanah, dan *toe-off* atau *pre-swing*. Sedangkan pada fase *swing* terdapat proses *acceleration*, *midswing*, dan *deceleration*. Durasi untuk menyelesaikan siklus *gait* dikenal sebagai *gait time*, yang terbagi dalam *stance time* dan *swing time* (Permatasari, 2016).

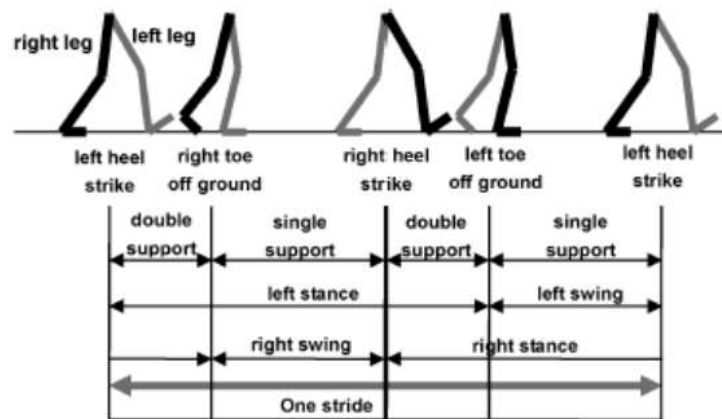


**Gambar 2.2** Gait Cycle Selama Satu Siklus.

Sumber: (Ariano et al., 2019)

Sedangkan pada robot *humanoid* gerakan berjalan robot direpresentasikan sebagai siklus berulang, atau biasa disebut juga dengan gaya berjalan atau *walking gait*. Gaya berjalan ini terdiri dari serangkaian Gerakan yang menyerupai cara berjalan

manusia. Gaya berjalan robot humanoid dapat dibagi menjadi dua fase saat berjalan yaitu fase SSP (*Single Support Phase*) dan DSP (*Double Support Phase*). SSP adalah kondisi pada saat robot berjalan hanya bertumpu pada salah satu telapak kaki sedangkan DSP adalah kondisi pada saat robot berjalan dengan bertumpu pada kedua kaki (Anragani, 2016). Ilustrasi gaya berjalan robot *humanoid* dapat dilihat pada gambar 2.4.



**Gambar 2.3** Ilustrasi Gaya Berjalan Robot

Sumber: (Yilmaz, 2010)

## 2.7 Transformasi Geometri

Transformasi geometri berasal dari kata transformasi yang berarti perubahan dan geometri yang berarti ilmu yang membahas tentang bangun. Transformasi geometri merupakan proses dalam penentuan titik koordinat baru pada sebuah bidang.

Transformasi geometri dibagi menjadi empat jenis yaitu transformasi pergeseran (translasi), pencerminan (refleksi), perputaran (rotasi), dan perbesaran (dilatasi). Pada penelitian ini, transformasi geometri digunakan untuk membuat gerakan robot bermanuver seperti berjalan lurus dan belok (Wibowo, 2017).

### 2.7.1 Translasi (Pergeseran)

Translasi atau pergeseran adalah sebuah proses yang dilakukan untuk menggeser sebuah bangun pada koordinat kartesian sebanyak nilai pergeseran sesuai dengan yang telah ditentukan dalam sumbu x, y, z. Pada transformasi translasi, ukuran dan bentuk serta arah bangun tidak berubah, yang berubah hanya posisinya terhadap posisi awal. Pada Gambar 2. 4 segitiga ABC berwarna hitam mempunyai koordinat A (3,9), B (3,3), C (6,3), apabila ditransformasikan translasi menjadi segitiga ABC warna oranye, hijau dan ungu perhitungannya adalah sebagai berikut :

1. Segitiga ABC oranye

Koordinat segitiga ABC oranye A<sub>2</sub>(-7,9), B<sub>2</sub>(-7,3), C<sub>2</sub>(- 4,3). Maka pergeseran translasi sebanyak x,y (-10,0).

2. Segitiga ABC ungu

Koordinat segitiga ABC ungu A<sub>3</sub>(3,-4), B<sub>3</sub>(3,-10), C<sub>3</sub>(6,-10). Maka pergeseran translasi sebanyak x,y (0,- 13).

3. Segitiga ABC hijau

Koordinat segitiga ABC hijau A<sub>4</sub>(3,-4), B<sub>4</sub>(3,-10), C<sub>4</sub>(6,-10). Maka pergeseran translasi sebanyak x,y (-10,- 13).

Dari perhitungan di atas maka transformasi pergeseran / translasi dapat dirumuskan sebagai berikut :

$$P_{(x,y)} \xrightarrow{T\begin{pmatrix} a \\ b \end{pmatrix}} P'_{(x+a, y+b)}$$

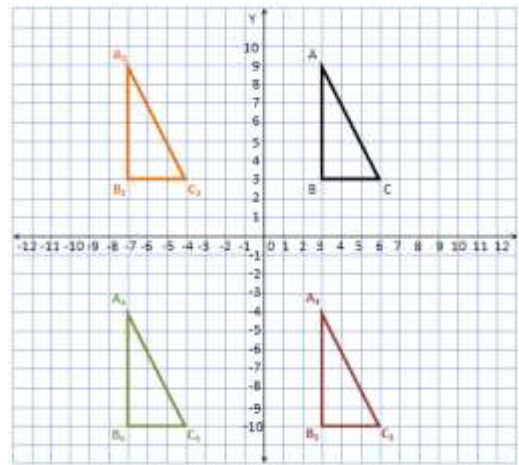
Dimana :

P = koordinat awal

P' = koordinat akhir (setelah di transformasi translasi)

a = pergeseran horisontal (+ ke kanan, - ke kiri)

$b$  = pergeseran vertikal (+ ke atas, - ke bawah)



**Gambar 2.4** Contoh Transformasi Translasi

Sumber: (Wibowo, 2017)

### 2.7.2 Rotasi (Perputaran)

Transformasi rotasi adalah jenis transformasi geometri yang digunakan untuk memutar objek atau titik dalam ruang terhadap suatu titik pusat atau sumbu rotasi. Transformasi ini mengubah orientasi atau posisi objek tanpa mengubah bentuk atau ukurannya. Dalam transformasi rotasi, objek atau titik diputar sejauh sudut tertentu dalam sekitar titik pusat tertentu atau sumbu rotasi. Pada Gambar 2. 5 segitiga ABC berwarna hitam mempunyai koordinat  $A(3,9)$ ,  $B(3,3)$ ,  $C(6,3)$ , apabila ditransformasikan rotasi mejadi segitiga ABC warna oranye, hijau dan ungu perhitungannya adalah sebagai berikut :

#### 1. Segitiga ABC oranye

Koordinat segitiga ABC oranye  $A_2(-9,3)$ ,  $B_2(-3,3)$ ,  $C_2(-3,6)$ . Maka pergeseran rotasi dengan sudut  $+90^\circ$  atau  $-270^\circ$  dengan pusat rotasi pada  $O(0,0)$ .

## 2. Segitiga ABC ungu

Koordinat segitiga ABC ungu A3(9,-3), B3(3,-3), C3(3,- 6). Maka pergeseran rotasi dengan sudut  $+270^\circ$  atau  $-90^\circ$  dengan pusat rotasi pada O(0,0). 16

## 3. Segitiga ABC hijau

Koordinat segitiga ABC hijau A4(3,-9), B4(-3,-3), C4(- 6,-3). Maka pergeseran rotasi dengan sudut  $+180^\circ$  atau  $- 180^\circ$  dengan pusat rotasi pada O(0,0).

Dari perhitungan di atas maka transformasi rotasi / perputaran dapat dirumuskan sebagai berikut :

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x - a \\ y - b \end{bmatrix} + \begin{bmatrix} a \\ b \end{bmatrix}$$

Dimana :

x = koordinat x awal

y = koordinat y awal

x' = koordinat x akhir (setelah ditransformasi rotasi)

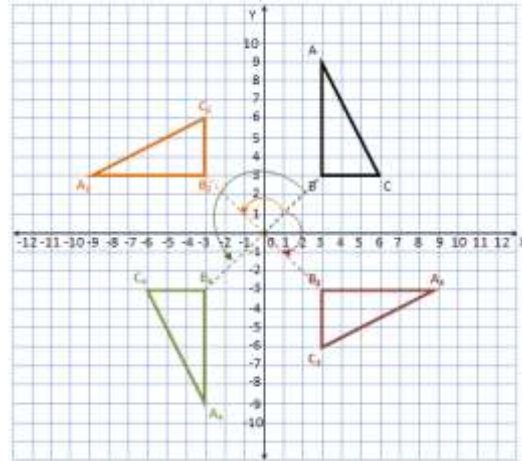
y' = koordinat y akhir (setelah ditransformasi rotasi)

$\theta$  = sudut putar

a = sumbu x pusat putar

b = sumbu y pusat putar

Pada transformasi rotasi, sudut putar ( $\theta$ ) bernilai positif (+) akan menyebabkan bangun berputar berlawanan arah jarum jam (*counter clockwise*) sedangkan apabila bernilai negatif (-) akan menyebabkan bangun berputar searah jarum jam (*clockwise*).

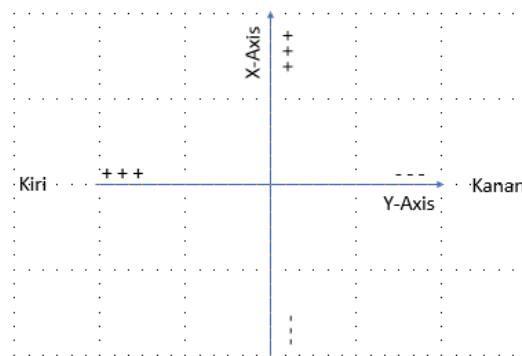


**Gambar 2.5** Contoh Transformasi Rotasi

Sumber: (Wibowo, 2017)

## 2.8 Sistem Koordinat Kartesian

Konstruksi robot terdiri dari tiga sumbu linier (prismatik). Sistem koordinat kartesius digunakan dalam matematika untuk menghitung posisi titik dalam ruang tiga dimensi menggunakan tiga nilai yang dikenal sebagai koordinat-x, koordinat-y, dan koordinat-z. Setiap sumbu memiliki kemampuan untuk bergerak dalam area sumbu x-y-z.

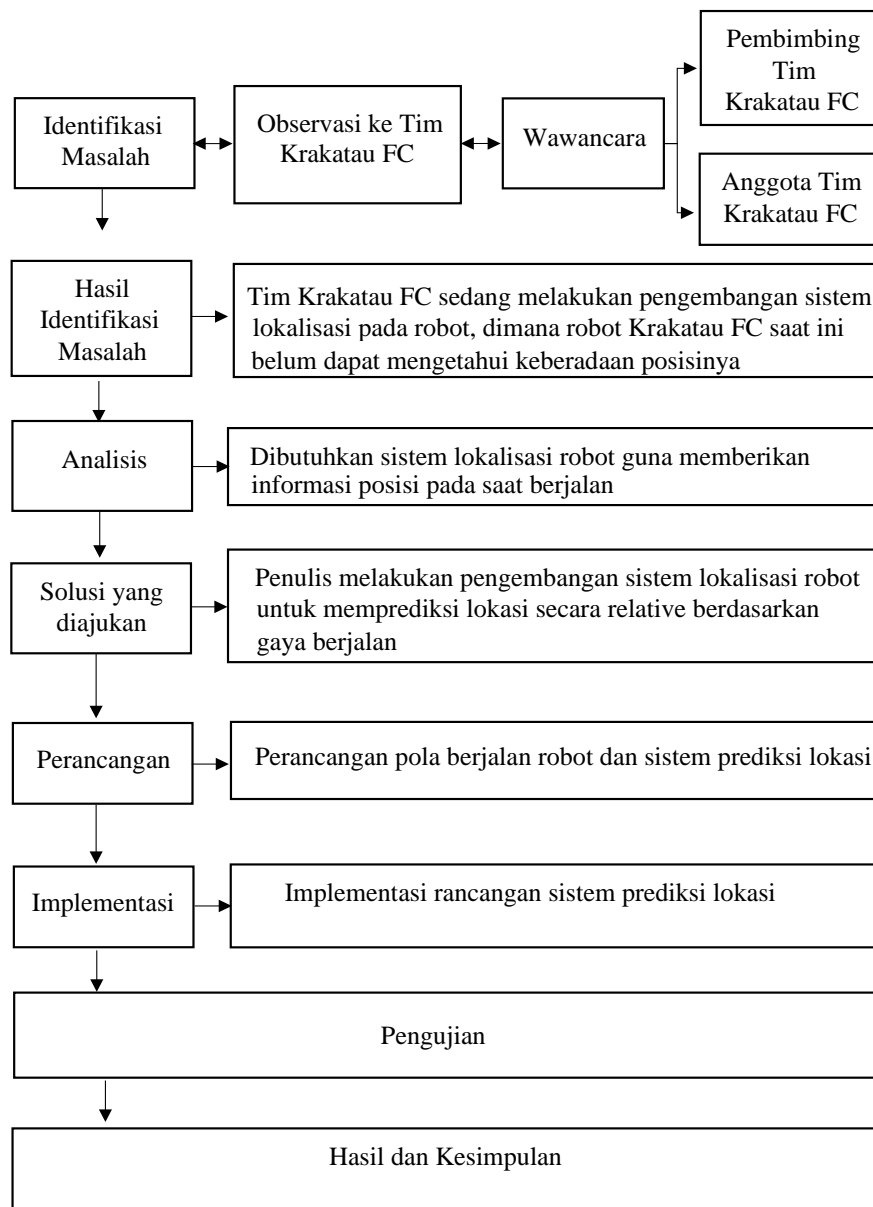


**Gambar 2.6** Koordinat Kartesian Pada Robot

## BAB III METODE PENELITIAN

### 3.1 Tahapan Penelitian

Adapun tahapan-tahapan yang akan dilakukan dalam penelitian ini tahapan tersebut dapat digambarkan dalam bentuk diagram yang dapat dilihat pada gambar 3.1.



**Gambar 3.1** Tahapan Penelitian



### **3.2 Metode Pengumpulan Data**

Metode pengumpulan data yang dilakukan pada penelitian ini, yaitu sebagai berikut:

1. Observasi

Peneliti melakukan pengamatan secara langsung pada robot sepak bola Humanoid Krakatau FC. Sehingga ditemukannya sebuah permasalahan dalam sistem navigasi milik robot Krakatau FC yang kemudian dijadikan sebagai latar belakang pada penelitian ini.

2. Studi Tinjauan Pustaka

Pada tahap studi Tinjauan Pustaka, penulis melakukan pengumpulan data berdasarkan penelitian-penelitian terdahulu yang bersumber dari Tinjauan Pustaka, jurnal penelitian, dan melakukan studi kepustakaan dalam mencari bahan bacaan dari internet yang memiliki keterkaitan dengan objek penelitian yang akan diteliti.

3. Eksperimen

Pengumpulan data pada penelitian ini dilakukan dengan melakukan tuning terhadap pergerakan robot yang kemudian dilakukan pencatatan hasil yang berkaitan dengan proses penelitian.

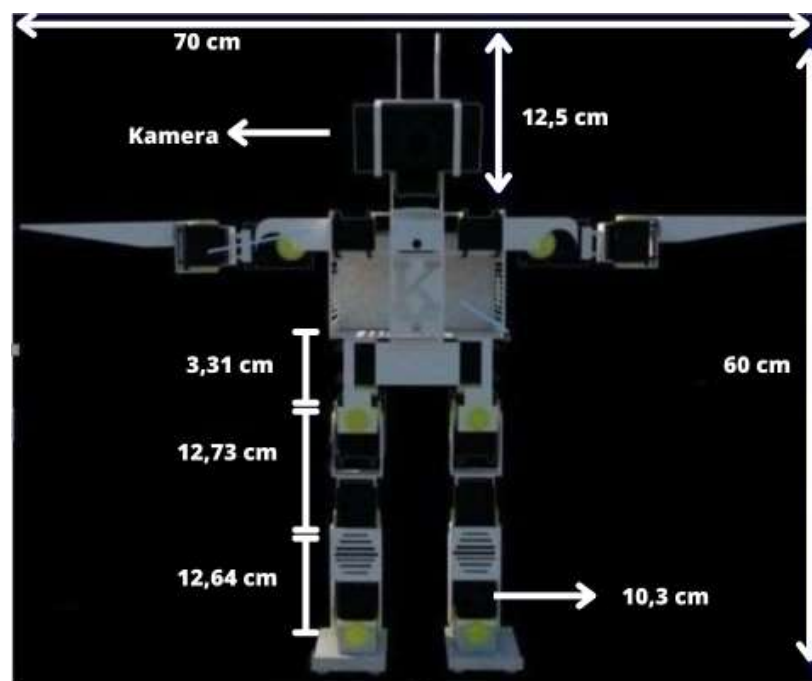
### **3.3 Waktu dan Tempat Penelitian**

Ruang lingkup penelitian ini dilakukan di Universitas Teknokrat Indonesia, khususnya di ruang Laboratorium Robotika divisi KRSBI Humanoid.

### **3.4 Objek Penelitian**

Objek penelitian yang digunakan pada penelitian ini adalah robot sepak bola *humanoid* milik tim Krakatau FC. Robot Krakatau dirancang memiliki 20 *Degree*

*Of Freedom* (DOF), dengan panjang rentang tangan = 70 cm, lebar = 12 cm, dan tinggi = 60 cm serta memiliki bobot sebesar 4.5 kg. Pada setiap dof digerakkan menggunakan sebuah actuator berupa servo motor dynamixel. Robot Krakatau secara keseluruhan menggunakan 3 jenis servo motor yang berbeda yaitu MX-28T, MX-64T dan MX-106. Rangka robot dibuat dengan aluminium *alloy* yang dipotong menggunakan mesin CNC. Dimensi robot Krakatau FC dapat dilihat pada gambar dibawah ini.



**Gambar 3.2** Dimensi Robot Krakatau FC



### 3.4 Alat dan Bahan Penelitian

Adapun alat dan bahan yang digunakan dalam penelitian ini terdiri dari perangkat keras (*hardware*) perangkat lunak (*software*), adalah sebagai berikut :

#### 3.5.1 Perangkat Keras (Hardware)

Pada penelitian ini, dibutuhkan beberapa perangkat keras untuk digunakan sebagai *tools* penunjang penelitian. Adapun rancangan perangkat keras yang digunakan pada penelitian ini dapat dilihat pada Tabel 3.1.

Tabel 3.1 Perangkat Keras Yang Digunakan

No	Perangkat Keras	Spesifikasi	Deskripsi
1	Laptop Lenovo Ideapad 130- 15ikb 	<ul style="list-style-type: none"> <li>• Processor : Intel Core i3 7020u ( 2,3 Ghz )</li> <li>• RAM : 12 GB DDR4 2133 Mhz</li> <li>• Storage : 256 GB SSD dan 80 GB HDD</li> </ul>	Sebagai media proqraming robot sepak bola Krakatau FC dan media monitoring.
2	Mini PC Intel NUC10i3FNH 	<ul style="list-style-type: none"> <li>• Intel® Core™ i3- 10110U Processor (4M Cache, up to 4.10 GHz)</li> <li>• Memory Type- DDR4-2666 1.2V SO-DIMM</li> <li>• 2.5" HDD/SSD + M.2 SATA SSD (RAID-0 RAID-1)</li> </ul>	Sebagai main controller sebagai pusat pengelola utama yang mengendalikan mulai dari kontrol internal hingga kontrol eksternal robot.

**Tabel 3.1** Perangkat Keras Yang Digunakan (Lanjutan)

3	<p>KSC Versi 2.0</p> 	<ul style="list-style-type: none"> <li>• STMicroelectronics 32F103RE ARM Cortex32-bit CPU (clocked @ 72MHz) (512KB Flash, 64KB SRAM )</li> <li>• 3-axis gyroscope, 3-axis accelerometer, s</li> <li>• USB Port, Serial Port 5x TTL Ports (for Dynamixel)</li> </ul>	<p>Sebagai sub-controller di maksudkan sebagai low level hardware interface baik untuk sensor maupun actuator.</p>
4	<p>Aktuator servo motor dynamixel</p> 	<ul style="list-style-type: none"> <li>• MCU: Cortex-M3 (72 [Mhz], 32 [bit])</li> <li>• Catu daya: 10 ~ 14.8VDC (Typ 12VDC)</li> <li>• Kecepatan Tanpa Beban: 63 rpm</li> <li>• Konsumsi arus tanpa beban : 150mA</li> <li>• Angle : 360°</li> <li>• Sensor Posisi: Contactless absolute encoder (12-bit, 360°)</li> </ul>	<p>Sebagai penggerak pada persendian robot.</p>

### 3.5.2 Perangkat Lunak (Software)

Pada penelitian ini, dibutuhkan beberapa perangkat lunak untuk digunakan sebagai *tools* penunjang penelitian. Adapun perangkat lunak yang digunakan pada penelitian ini terdiri dari:

1. Windows 10

Digunakan sebagai sistem operasi pada Laptop Lenovo ideapad.

2. Linux Ubuntu 18.04

Digunakan sebagai sistem operasi pada *Mini PC* Intel NUC10i3FNH yang terinstal pada robot.

3. Sublime Text Editor

Sublime Text Editor adalah teks editor untuk berbagai bahasa pemrograman diantaranya C, C++, C#, PHP, CSS, HTML, ASP dan banyak lagi . Sublime Text merupakan teks editor lintas *platform* dengan *Python Application Interface* (API).

4. Putty

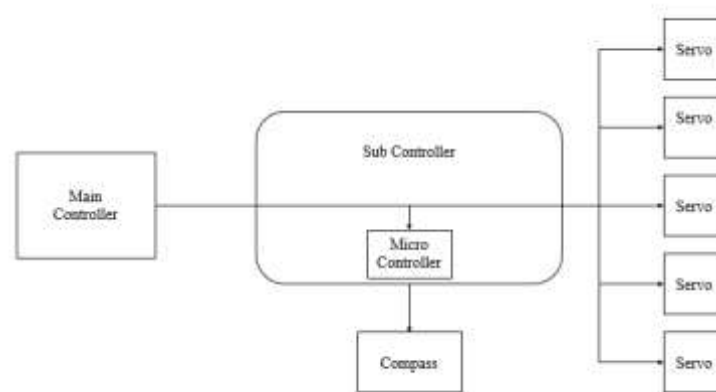
Digunakan sebagai aplikasi remote access yang memanfaatkan protocol jaringan untuk kebutuhan remote komputer jarak jauh.

## 3.6 Penelitian

### 3.6.1 Rancangan Arsitektur Sistem Robot

Rancang bangun model sistem kontrol *hardware* terlihat pada gambar 3.3. *Hardware* robot menggunakan konsep pemisahan *low level* dan *high level* kontrol untuk mengakomodasi keperluan *sensing* dan *actuating*. *Main-controller* yang digunakan adalah sebuah Komputer papan tunggal (*Single Board Computer*). *Sub-controller* yang digunakan di tenagai oleh *Arm-cortex* m3. Penggunaan *sub-*

*controller* dimaksudkan sebagai *low level hardware interface* baik untuk sensor maupun aktuator. Pemisahan ini di butuhkan untuk meminimalkan waktu tunggu yang mungkin terjadi jika *main-controller* harus berkomunikasi secara langsung ke I/O dan data yang menuju aktuator Dynamixel tidak harus melewati mikrokontroller. Akuator yang digunakan yaitu *smart* servo dynamixel MX-28, dynamixel MX-64 dan dynamixel MX-106 yang memiliki putaran 360 derajat yang mampu menggerakkan setiap persendian robot secara *vertical* maupun *horizontal*.

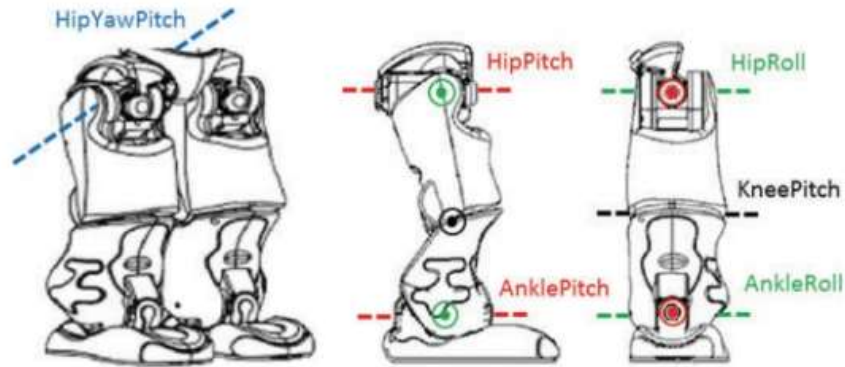


**Gambar 3.3** Diagram Blok Sistem Kontrol Pada Robot

### 3.6.2 Rancangan Pola Berjalan Robot

Robot Krakatau memiliki sistem berjalan secara bipedal dan menerapkan model *Zero Moment Point (ZMP)* untuk memprediksikan peletakan setiap sendi robot pada koordinat kartesian. Dalam penelitian titik utama yang akan digambarkan pada koordinat kartesian adalah bagian kaki robot. Setiap kaki robot memiliki enam derajat kebebasan yang dapat digerakkan (DOF), secara fungsional semua DOF diklasifikasikan menjadi tiga jenis, jenis pertama bertanggung jawab untuk abduksi kaki / adduksi (sendi pinggul dan pergelangan kaki lateral), jenis kedua untuk ekstensi / fleksi kaki (sendi panggul, lutut, dan pergelangan kaki

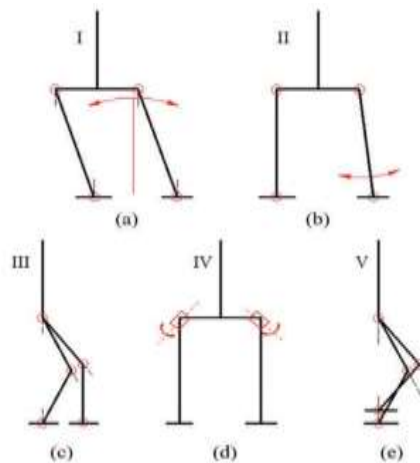
sagital), dan jenis ketiga untuk *intorsion / ekstorsion* tungkai (sendi panggul depan). Sendi-sendi yang disebutkan dapat dilihat pada gambar 3.4.



**Gambar 3.4** Sendi Pada Robot Berkaki Dua

Sumber: (Xia et al., 2017)

Dalam merancang satu siklus gerakan berjalan pada robot terdapat lima *pose* gerakan dari gerak dasar berjalan yang perlu diatur pergerakannya untuk setiap gerakan. Lima gerakan dalam gerak dasar berjalan dapat dilihat pada gambar 3.5.

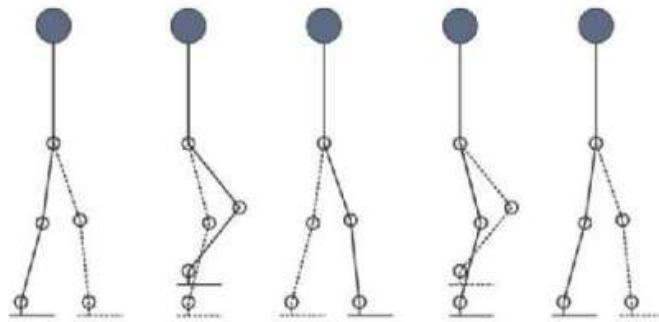


**Gambar 3.5** (a) Gerak Dari Sisi ke Sisi, (b) Gerak Melangkah Ke Samping, (c) Gerak Melangkah Ke Depan, (d) Gerak Berputar, Dan (e) Gerak Mengangkat Kaki.

Sumber: (Xia et al., 2017)

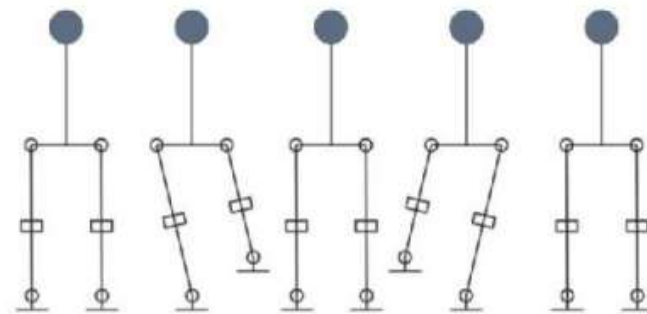
Robot Krakatau menerapkan pola berjalan dinamis, pola ini diadopsi berdasarkan model gerak pada robot Darwin-OP yang kemudian disesuaikan

berdasarkan struktur robot. *Pose* dasar pada model gerak dinamis adalah menempatkan robot untuk berjalan tegap seperti manusia. Gerakan berjalan pada robot dibuat mengikuti aturan berjalan pada manusia sehingga satu langkah untuk manusia sama dengan satu siklus pada robot. Dalam satu siklus gerakan berjalan maju membutuhkan empat *pose* gerakan menggunakan model lintasan dari ayunan kaki saat melangkah dan enam sendi yang ada pada kaki akan digunakan untuk berjalan. *Pose* berjalan pada saat melangkah maju dan berjalan ke samping dapat dilihat pada gambar 3.6 dan 3.7.



**Gambar 3.6** Gerakan Melangkah X-Axis

Sumber: (Udinus, 2013)



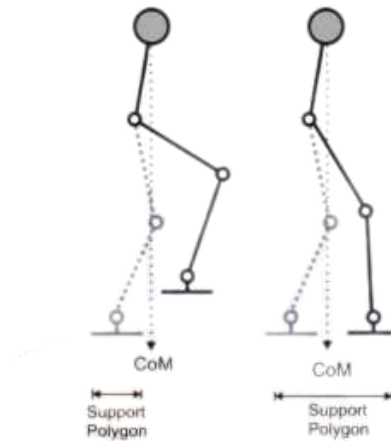
**Gambar 3.7** Gerakan Melangkah Y-Axis

Sumber: (Udinus, 2013)

Pada *pose* berjalan x-axis maupun y-axis dalam setiap gerakan melangkah terdapat 2 kondisi yaitu DSP (*Double Support Phase*) dan 2 kondisi SSP (*Single*



*Support Phase*). Pada kondisi DSP untuk titik *center of mass* berada pada posisi diantara 2 telapak kaki, sedangkan pada kondisi SSP untuk titik *center of mass* berada pada posisi tengah telapak kaki. Ilustrasi dari *center of mass* pada saat robot berjalan dapat dilihat pada gambar 3.8.



**Gambar 3.8** Ilustrasi Letak CoM Pada Saat Berjalan

Sumber: (Anragani, 2016)

Cara berjalan robot diatur berdasarkan sebuah parameter yang digunakan untuk menjaga titik pusat masa robot (*Center Of Mass*) dan dilengkapi kemampuan menyeimbangkan tubuh pada saat berjalan dengan mengubah *offset actuator* secara otomatis sesuai dengan nilai *error* yang didapat dari sensor *gyroscope*.

Adapun parameter yang digunakan untuk mengatur robot berjalan dapat dilihat pada table 3.2

**Tabel 3.2** Parameter Robot Krakatau

No	Parameter	Fungsi
1	<i>Walking mode (on/off)</i>	Menghidupkan / mematikan mode berjalan
2	<i>Mod enable (on/off)</i>	Menghidupkan / mematikan mode modifikasi
3	<i>X offset (mm)</i>	Mengatur <i>offset X</i> ke arah depan dan belakang

Tabel 3.2 Parameter Robot Krakatau (Lanjutan 1)

No	Parameter	Fungsi
4	<i>Y offset (mm)</i>	Mengatur <i>offset</i> Y ke arah kanan dan kiri
5	<i>Z offset (mm)</i>	Mengatur <i>offset</i> Z ke arah atas dan bawah
6	<i>Roll (x) offset (degree)</i>	Mengatur rotasi putaran pada X koordinat
7	<i>Pitch (y) offset (degree)</i>	Mengatur rotasi putaran pada Y koordinat
8	<i>Yaw (z) offset (degree)</i>	Mengatur rotasi putaran pada Z koordinat
9	<i>Hip pitch offset (degree)</i>	Mengatur rotasi pada pinggul pada Y koordinat
10	<i>Auto balance (on/off)</i>	Menghidupkan / mematikan keseimbangan otomatis
11	<i>Period time (msec)</i>	Mengatur waktu untuk menyelesaikan dua langkah penuh (kanan dan kiri)
12	<i>DSP ratio</i>	Mengatur rasio waktu antara kedua kaki ditanah hingga satu kaki ditanah
13	<i>Min DSP ratio</i>	Mengatur nilai <i>minimal</i> DSP ratio
14	<i>Max DSP ratio</i>	Mengatur nilai <i>maximal</i> DSP ratio
15	<i>Step forward/back ratio</i>	Mengatur rasio melangkah
16	<i>Step forward/back (mm)</i>	Menjalankan robot maju dan mundur
17	<i>Step right/left (mm)</i>	Menjalankan robot kesamping kanan dan kiri
18	<i>Step direction(degree)</i>	Mengatur arah berjalan robot
19	<i>Turning aim (on/off)</i>	Menggerakkan robot kesuatu titik ketika berjalan
20	<i>Turn offset left/right (mm)</i>	Menggerakkan <i>offset</i> robot kesamping kanan dan kiri

**Tabel 3.2** Parameter Robot Krakatau (Lanjutan 2)

No	Parameter	Fungsi
21	<i>Foot height</i> (mm)	Mengatur ketinggian kaki selama melangkah
22	<i>FB swing right/left</i> (mm)	Mengatur ayunan tangan ke samping kanan dan kiri
23	<i>RL swing right/left</i> (mm)	Mengatur ayunan tangan ke depan dan belakang
24	<i>Swing top/down</i> (mm)	Mengatur ayunan tubuh naik-turun saat berjalan
25	<i>Pelvis offset</i> (degree)	Mengatur rotasi panggul pada koordinat X
26	<i>Arm swing gain</i>	Meningkatkan ayunan lengan seiring dengan melangkah maju atau mundur. Jika kaki kiri maju maka lengan kanan mengayun
27	<i>Balance knee gain</i>	Meningkatkan keseimbangan lutut
28	<i>Balance ankle pitch gain</i>	Meningkatkan keseimbangan rotasi pergelangan kaki Y koordinat
29	<i>Balance hip roll gain</i>	Meningkatkan keseimbangan rotasi pinggul X koordinat
30	<i>Balance ankle roll gain</i>	Meningkatkan keseimbangan rotasi pergelangan kaki X koordinat
31	<i>Balance arm pitch gain</i>	Meningkatkan keseimbangan rotasi tangan Y koordinat
32	<i>Balance arm roll gain</i>	Meningkatkan keseimbangan rotasi tangan X koordinat

**Tabel 3.2** Parameter Robot Krakatau (Lanjutan 3)

No	Parameter	Fungsi
33	P gain	Mengatur PID
34	I gain	Mengatur PID
35	D gain	Mengatur PID

### 3.6.2 Rancangan Sistem Prediksi Lokasi

Algoritma merupakan proses-proses logis yang disusun secara sistematis untuk menyelesaikan suatu masalah seperti lokalisasi robot. Untuk itu peneliti perlu membuat alur algoritma yang terbagi menjadi tahapan-tahapan proses seperti berikut :



**Gambar 3.9** Rancangan Sistem Berdasarkan *Walking Engine*

Tahapan sistem berdasarkan walking engine adalah sebagai berikut:

1. Robot

Pada proses ini objek yang digunakan ialah robot sepak bola Krakatau FC.

2. Perintah Berjalan

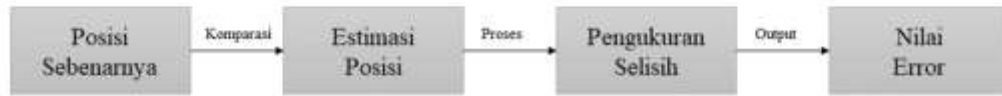
Robot akan diberikan perintah berjalan dengan menginputkan beberapa parameter yaitu parameter langkah, panjang langkah, dan arah hadap melangkah.

3. *Walking Engine*

Pada proses ini *walking engine* akan menjalankan perintah untuk berjalan sesuai dengan parameter yang ditentukan.

#### 4. Estimasi Posisi

Pada proses ini dilakukan estimasi posisi pada gerakan yang dikerjakan oleh *walking engine*.



**Gambar 3.10** Rancangan Sistem Berdasarkan Kondisi Robot

Tahapan sistem berdasarkan kondisi robot sebenarnya adalah sebagai berikut:

##### 1. Komparasi

Pada proses ini dilakukan komparasi atau membandingkan antara posisi robot yang sebenarnya dengan estimasi posisi.

##### 2. Pengukuran

Perhitungan error pada pengujian ini dilakukan dengan cara menghitung selisih antara jarak yang terukur setelah robot melakukan pergerakan ke arah target dan jarak yang diharapkan dicapai oleh robot. Rumus yang digunakan untuk menghitung nilai *error* yaitu sebagai berikut.

Selisih jarak tempuh = nilai target – nilai aktual.

### 3.7 Rancangan Skema Pengujian

Rancangan skema pengujian akan dilakukan pada robot sepak bola humanoid Krakatau FC. Beberapa pengujian yang dilakukan antara lain:

**Tabel 3.3** Rancangan Skema Pengujian

No	Pengujian	Skenario Pengujian	Data yang diuji
1	Berjalan Maju	Robot berjalan maju secara otomatis	Sumbu X dan orientasi

**Tabel 3.3** Rancangan Skema Pengujian (Lanjutan)

No	Pengujian	Skenario Pengujian	Data yang diuji
2	Berjalan Mundur	Robot berjalan mundur secara otomatis	Sumbu X dan orientasi
3	Berjalan Ke Kanan	Robot berjalan ke kanan secara otomatis	Sumbu Y dan orientasi
4	Berjalan Ke Kiri	Robot berjalan ke kiri secara otomatis	Sumbu Y dan orientasi

### 3.8 Jadwal Penelitian

Berikut adalah jadwal penelitian yang dapat menjadi acuan peneliti untuk menyelesaikan penelitian dengan tepat waktu dapat dilihat pada Tabel 3.4.

**Tabel 3. 4** Jadwal Penelitian

Kegiatan	Juni				Juli				Agustus				September				Oktober				November			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Observasi Masalah	■	■	■	■																				
Studi Literasi					■	■	■	■																
Perancangan Sistem									■	■	■	■	■	■										
Implementasi														■	■	■	■	■	■	■				
Pengujian														■	■	■	■	■	■	■				
Penulisan Laporan														■	■	■	■	■	■	■	■	■		
Seminar Akhir																						■		

## BAB IV HASIL DAN PEMBAHASAN

### 4.1 Proses Tuning Offset Tuner

Pada tahap ini dilakukan proses *tuning* terhadap *pose* robot, hal ini bertujuan untuk menyeimbangkan *pose* pada robot. *Offset* tuner merupakan suatu alat atau fitur yang krusial dalam kalibrasi dan penyesuaian parameter sistem pada berbagai perangkat, termasuk robotika, kendaraan otonom, dan sensor lainnya. *Offset tuner* bekerja dengan melakukan penyesuaian terhadap nilai *offset* pada suatu sistem. *Offset* adalah nilai tambahan yang digunakan untuk mengkompensasi atau menghilangkan kesalahan yang mungkin muncul dalam pembacaan sensor atau operasi sistem. Pada umumnya, kesalahan ini bisa disebabkan oleh faktor-faktor seperti toleransi manufaktur, perubahan suhu, atau perubahan lingkungan lainnya. Proses penyesuaian *offset tuner* melibatkan langkah-langkah untuk menentukan dan mengatur ulang nilai *offset* agar sistem memberikan hasil yang lebih akurat dan konsisten. Proses *tuning* dapat dilihat pada gambar 4.1 dan 4.2 berikut.



**Gambar 4.1** Proses Pengaturan *Offset Tuner*

```

root@krakatau: /home/krakatau/Desktop/KRSBIH/Linux/project/offset_tuner
ID: 1 (R_SHO_PITCH) [2048] -45 [16384| 0000 -16384 0 0 0
ID: 2 (L_SHO_PITCH) [2048] 74 [2122| 2092 -30 8 201 234
ID: 3 (R_SHO_ROLL) [2048] 310 [16384| 0000 -16384 0 0 0
ID: 4 (L_SHO_ROLL) [2048] -200 [1848| 1898 50 8 201 234
ID: 5 (R_ELBOW) [2048] -90 [16384| 0000 -16384 0 0 0
ID: 6 (L_ELBOW) [2048] 120 [2168| 2164 -4 8 201 234
ID: 7 (R_HIP_YAW) [2048] -30 [2018| 2021 3 23 201 234
ID: 8 (L_HIP_YAW) [2048] 120 [2168| 2165 -3 23 201 234
ID: 9 (R_HIP_ROLL) [2048] -60 [1988| 1984 -4 23 201 234
ID:10 (L_HIP_ROLL) [2048] 42 [2090| 2088 -2 23 201 234
ID:11 (R_HIP_PITCH) [2048] 180 [2228| 2232 4 23 201 234
ID:12 (L_HIP_PITCH) [2048] -20 [2028| 2022 -6 23 201 234
ID:13 (R_KNEE) [2048] -101 [1947| 1942 -5 23 201 234
ID:14 (L_KNEE) [2048] 40 [2088| 2084 -4 23 201 234
ID:15 (R_ANK_PITCH) [2048] -20 [2028| 2032 4 23 201 234
ID:16 (L_ANK_PITCH) [2048] -20 [2028| 2032 4 23 201 234
ID:17 (R_ANK_ROLL) [2048] 241 [2289| 2295 6 23 201 234
ID:18 (L_ANK_ROLL) [2048] 26 [2074| 2067 -7 23 201 234
ID:19 (HEAD_PAN) [2048] 60 [2108| 2101 -7 8 201 234
ID:20 (HEAD_TILT) [2048] -260 [1788| 1778 -10 8 201 234
GOAL OFFSET MODVAL PRSPOS ERRORS P_GAIN I_GAIN D_GAIN
]

```

**Gambar 4.2** Parameter Pengaturan *Offset Tuner*

Berikut merupakan penjelasan dari parameter yang terdapat pada *offset tuner*:

1. *GOAL*: adalah nilai yang ditetapkan untuk ID DYNAMIXEL.
2. *OFFSET*: adalah nilai yang ditetapkan untuk *offset* DYNAMIXEL.
3. *MODVAL*: Ini adalah posisi tujuan "baru
4. *PRSPOS*: adalah posisi saat ini dari DYNAMIXEL ID..
5. *ERRORS*: adalah perbedaan antara *PRSPOS* dan *MODVAL*.
6. *P\_GAIN*: adalah nilai yang ditetapkan untuk penguatan P (proporsional).
7. *I\_GAIN*: adalah nilai yang ditetapkan untuk penguatan I (Integral)..
8. *D\_GAIN*: adalah nilai yang ditetapkan untuk penguatan P (diferensial).

## 4.2 Kalibrasi Sensor Kompas

Kalibrasi sensor kompas adalah langkah esensial dalam memastikan akurasi dan kehandalan navigasi pada berbagai perangkat elektronik, seperti ponsel pintar, *drone*, atau perangkat berbasis sensor inersia. Salah satu metode kalibrasi yang



umum digunakan melibatkan pembagian sensor kompas ke dalam delapan arah mata angin. Kalibrasi sensor kompas dengan membagi delapan arah mata angin melibatkan langkah-langkah yang sistematis untuk mengkompensasi anomali magnetik dan faktor-faktor lingkungan lainnya yang dapat memengaruhi akurasi pembacaan. Proses ini dimulai dengan penempatan perangkat pada posisi datar dan bebas dari pengaruh magnetik eksternal. Setelah itu, sensor kompas diarahkan ke delapan arah mata angin (utara, timur, selatan, barat, dan empat arah interkardinal) untuk mengukur dan merekam nilai pembacaan di setiap arah. Hal ini penting untuk menjaga konsistensi dan keakuratan penentuan arah, terutama saat perangkat bergerak atau berada di lingkungan dengan potensi anomali magnetik. Proses kalibrasi kompas pada robot sepak bola humanoid dapat dilihat pada gambar 4.3 dibawah ini.



**Gambar 4.3** Proses Kalibrasi Kompas

### 4.3 Hasil Pengujian

Dalam bab pengujian ini, sistem yang telah dirancang dan dibuat pada bab perancangan akan mengalami serangkaian pengujian. Fokus utama dari bab ini adalah untuk mengevaluasi kinerja pergerakan robot sepak bola *humanoid* dan menganalisis bahwa pergerakannya tersebut sudah sesuai dengan tujuan yang telah ditetapkan sebelumnya. Pengujian melibatkan beberapa aspek, seperti pengujian berjalan maju, pengujian berjalan mundur, pengujian berjalan ke kanan dan pengujian berjalan ke kiri. Keseluruhan tujuan dari bab ini adalah untuk menilai dan memvalidasi sejauh mana sistem dapat memenuhi kriteria kinerja yang telah ditetapkan sebelumnya. Pada pengujian ini memiliki tujuan utama untuk mengevaluasi sejauh mana tingkat presisi robot dalam berjalan sesuai perintah.

#### 4.3.1 Pengujian Berjalan Maju

Proses pengujian ini melibatkan gerakan berjalan robot yang dilakukan di area lapangan berumput dengan menambahkan nilai tertentu pada sumbu kartesian. Pengujian ini dilakukan dengan cara meletakkan robot pada posisi awal atau titik 0 dimana posisi robot diam, kemudian pada posisi akhir robot setelah bergerak akan diukur dengan menggunakan alat ukur berupa meteran, dan perbedaan yang didapatkan antara nilai yang diinginkan (*set point*) dengan nilai aktual dianggap sebagai nilai *error*. Metode pengujian pada tahap ini melibatkan perubahan nilai pada sumbu x dari posisi yang diharapkan untuk mengevaluasi seberapa baik sistem merespon perintah pergerakan. Pada pengujian ini dilakukan sebanyak sepuluh kali dimana robot diperintahkan berjalan maju dengan memberikan kecepatan pada sumbu x sebesar 2,5 cm pada setiap langkahnya dan jumlah langkah yang berbeda-beda mulai dari 10 langkah sampai dengan 40

langkah pada setiap pengujian. Berikut merupakan pengujian dari gerak berjalan maju dapat dilihat pada gambar 4.4.



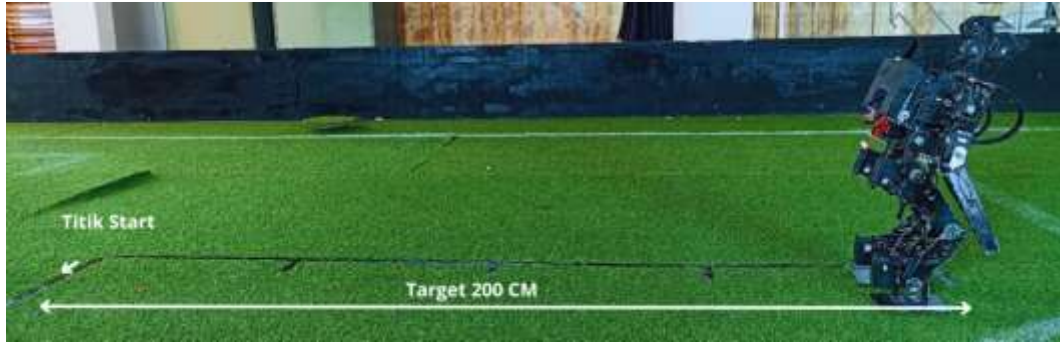
(a)



(b)



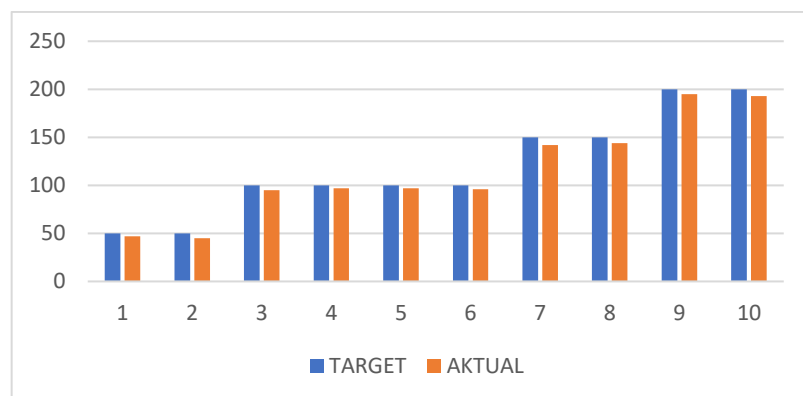
(c)



(d)

**Gambar4.4** Hasil Pengujian Berjalan Maju (a) Target 50 cm, (b) Target 100 cm, (c) Target 150 cm, dan Target 200 cm.

Pengujian dilakukan di lapangan berumput dimana robot akan diberikan perintah berjalan maju dengan target jarak tempuh mulai dari 50 cm hingga 200 cm. Pada pengujian ini robot diberikan nilai parameter berupa kecepatan yaitu 2,5 cm pada setiap langkah dan jumlah langkah yaitu mulai dari 10 langkah hingga 40 langkah. Dapat dilihat pada gambar 4.4 dimana robot sudah berusaha mendekati target yang diinginkan. Nilai target yang ditetapkan menggambarkan performa yang diharapkan atau tujuan yang ingin dicapai, sementara nilai aktual mencerminkan kinerja sebenarnya dari robot saat menjalankan pergerakan maju. Pada pengujian robot berjalan maju, terdapat perbandingan antara nilai target dan nilai aktual yang tercermin dengan jelas. Perbandingan tersebut dapat dilihat pada grafik gambar 4.5.



**Gambar 4.5** Grafik Hasil Pengujian Berjalan Maju

Pengujian telah dilakukan sebanyak sepuluh kali untuk mengevaluasi akurasi robot dalam mencapai jarak pendek maupun jarak jauh. Rangkuman hasil pengujian posisi robot terhadap target berjalan maju ini dapat dilihat dalam Tabel 4.1 dan 4.2 yang terlampir di bawah ini.

**Tabel 4. 1** Posisi Aktual Robot Terhadap Target Maju

No	Posisi Target (cm)	Posisi Aktual (cm)	Selisih (cm)	<i>Error</i> (%)
1	50	47	3	6%
2	50	45	5	10%
3	100	95	5	5%
4	100	97	3	3%
5	100	97	3	3%
6	100	96	4	4%
7	150	142	8	5,3%
8	150	144	6	4%
9	200	195	5	2,5%
10	200	193	7	3,5%
Rata – Rata Posisi Aktual		115.1		
Rata – Rata Presentase <i>Error</i>			4,6%	

Hasil dari pengujian robot berjalan maju dapat dilihat pada tabel 4.1, diketahui bahwa rata-rata posisi aktual terhadap target maju sebesar 115,1cm. Nilai *error* pengukuran didapatkan dari pembagian nilai selisih dengan nilai target kemudian dikalikan 100%.

$$Error = \frac{\text{Selisih}}{\text{Target}} \times 100\%$$

Pada keseluruhan pengujian didapatkan nilai rata-rata *error* terhadap target maju yaitu sebesar 4,6%. Sedangkan hasil pengujian arah hadap dapat dilihat pada tabel 4.2 yang terlampir dibawah ini.

**Tabel 4. 2** Posisi Arah Hadap

No	Posisi Target (°)	Posisi Akhir (°)	Selisih (°)
1	0	5	5
2	0	5	5
3	0	3	3
4	0	7	7
5	0	3	3
6	0	6	6
7	0	4	4
8	0	7	7
9	0	8	8
10	0	9	9
Rata – Rata <i>Error</i>			5,7

Hasil pengujian pada arah hadap robot terdapat penyimpangan sebesar  $5,7^\circ$ , nilai tersebut merupakan selisih antara target dengan posisi akhir dari robot pada saat melakukan pergerakan berjalan maju. Hal ini mengindikasikan bahwa adanya deviasi antara posisi yang diinginkan dan posisi yang dicapai oleh robot dalam pergerakan maju. Kemungkinan besar, nilai error ini terjadi karena setiap nilai servo pada posisi default kaki robot tidak sepenuhnya akurat. Selain itu nilai error ini disebabkan oleh gesekan kaki robot dengan lapangan yang terjadi secara terus menerus sehingga dapat merubah arah hadap posisi robot dan ketidak mampuan robot dalam menjaga keseimbangan tubuh pada saat berjalan sehingga mempengaruhi performa robot dalam berjalan.

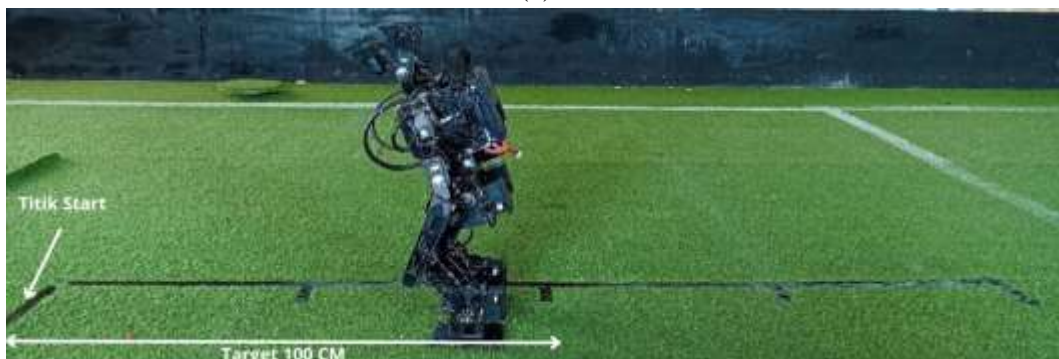
#### 4.3.2 Pengujian Berjalan Mundur

Proses pengujian ini melibatkan gerakan berjalan robot yang dilakukan diarea lapangan berumput dengan menambahkan nilai tertentu pada sumbu

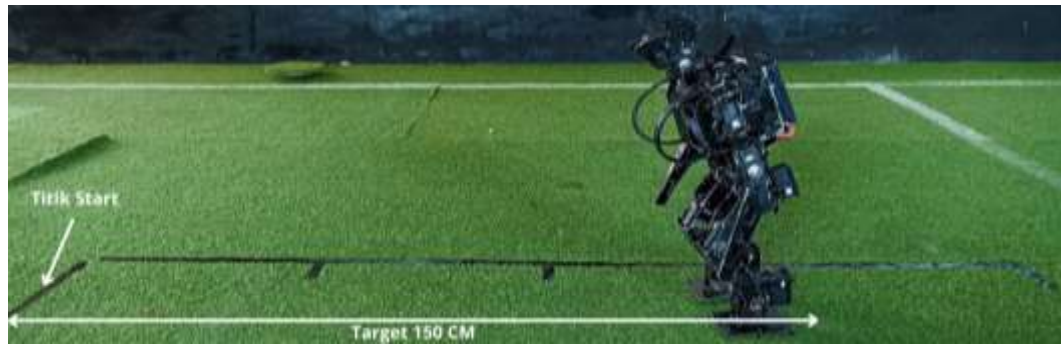
kartesian. Pengujian dilakukan dengan cara meletakkan robot pada posisi awal atau titik 0 dimana posisi robot diam. Kemudian ketika robot telah berhenti bergerak maka akan diukur dengan menggunakan alat ukur berupa meteran, dan perbedaan yang didapatkan antara nilai yang diinginkan (*set point*) dengan nilai aktual dianggap sebagai nilai *error*. Metode pengujian pada tahap ini melibatkan perubahan nilai pada sumbu x dari posisi yang diharapkan untuk mengevaluasi seberapa baik sistem merespon perintah pergerakan. Pada pengujian ini dilakukan sebanyak sepuluh kali dimana robot diperintahkan berjalan mundur dengan memberikan kecepatan pada sumbu x sebesar  $-2,5$  cm pada setiap langkahnya dan jumlah langkah yang berbeda-beda mulai dari 10 langkah sampai dengan 40 langkah pada setiap pengujian. Berikut merupakan pengujian dari gerak berjalan maju dapat dilihat pada gambar 4.6.



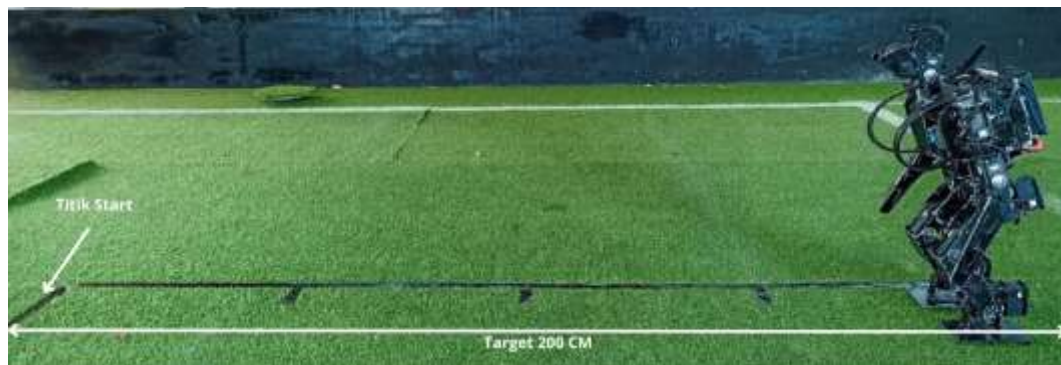
(a)



(b)



(c)

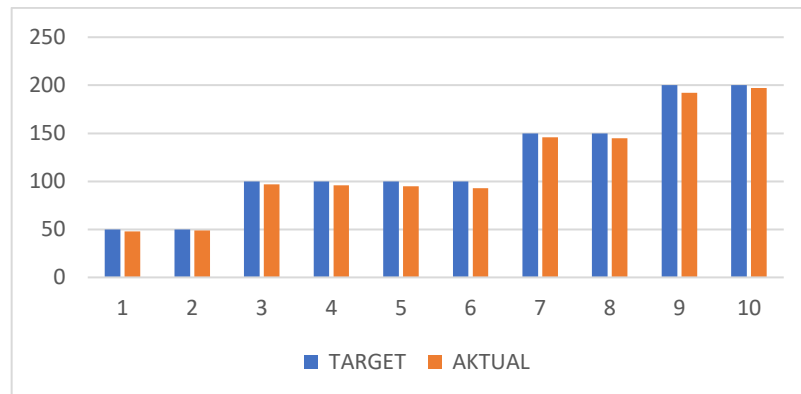


(d)

**Gambar 4.6** Hasil Pengujian Berjalan Mundur (a) Target 50 cm, (b) Target 100 cm, (c) Target 150 cm, dan Target 200 cm.

Pengujian dilakukan pada lapangan berumput dimana robot diberikan perintah berjalan mundur dengan target jarak tempuh mulai dari 50 cm sampai dengan 200 cm. Pada pengujian ini robot diberikan nilai parameter berupa kecepatan yaitu  $-2,5$  cm pada setiap langkah dan jumlah langkah keseluruhan yaitu 10 langkah sampai dengan 40 langkah. Dapat dilihat pada gambar 4.6 dimana robot sudah berusaha mendekati target yang diinginkan. Nilai target yang ditetapkan menggambarkan performa yang diharapkan atau tujuan yang ingin dicapai, sementara nilai aktual mencerminkan kinerja sebenarnya dari robot saat menjalankan pergerakan mundur. Pada pengujian robot berjalan maju, terdapat perbandingan antara nilai target dan nilai aktual yang tercermin dengan jelas. Perbandingan tersebut dapat dilihat pada grafik gambar 4.7.





**Gambar 4.7** Grafik Hasil Pengujian Berjalan Mundur

Pengujian telah dilakukan sebanyak sepuluh kali untuk mengevaluasi akurasi robot dalam mencapai jarak pendek maupun jarak jauh. Rangkuman hasil pengujian posisi robot terhadap target berjalan mundur ini dapat dilihat dalam Tabel 4.3 dan 4.4 yang terlampir di bawah ini.

**Tabel 4. 3** Posisi Aktual Robot Terhadap Target Mundur

No	Posisi Target (cm)	Posisi Aktual (cm)	Selisih (cm)	<i>Error</i> (%)
1	50	48	2	4%
2	50	49	1	2%
3	100	97	3	3%
4	100	96	4	4%
5	100	95	5	5%
6	100	93	7	7%
7	150	146	4	2,6%
8	150	145	5	3,3%
9	200	192	8	4%
10	200	197	3	1,5%
Rata – Rata Posisi Aktual		115,8		
Rata – Rata Presentase <i>Error</i>			3,4%	

Hasil dari pengujian robot berjalan mundur dapat dilihat pada tabel 4.3, diketahui bahwa rata-rata posisi aktual terhadap target maju sebesar 115,8cm. Nilai *error*

pengukuran didapatkan dari pembagian nilai selisih dengan nilai target kemudian dikalikan 100%.

$$Error = \frac{\text{Selisih}}{\text{Target}} \times 100\%$$

Pada keseluruhan pengujian didapatkan nilai rata-rata error terhadap target mundur yaitu sebesar 3,4%. Sedangkan hasil pengujian arah hadap dapat dilihat pada tabel 4.4 yang terlampir dibawah ini.

**Tabel 4. 4** Posisi Arah Hadap

No	Posisi Target (°)	Posisi Akhir (°)	Selisih (°)
1	0	8	8
2	0	10	10
3	0	3	3
4	0	8	8
5	0	10	10
6	0	9	9
7	0	5	5
8	0	5	5
9	0	7	7
10	0	4	4
Rata – Rata <i>Error</i>			6,9

Hasil pengujian pada arah hadap robot terdapat penyimpangan sebesar 6,9°, nilai tersebut merupakan selisih antara target dengan posisi akhir dari robot pada saat melakukan pergerakan berjalan mundur. Hal ini mengindikasikan bahwa adanya deviasi antara posisi yang diinginkan dan posisi yang dicapai oleh robot dalam pergerakan mundur. Kemungkinan besar, nilai error ini terjadi karena setiap nilai servo pada posisi default kaki robot tidak sepenuhnya akurat. Selain itu nilai error ini disebabkan oleh gesekan kaki robot dengan lapangan yang terjadi secara terus

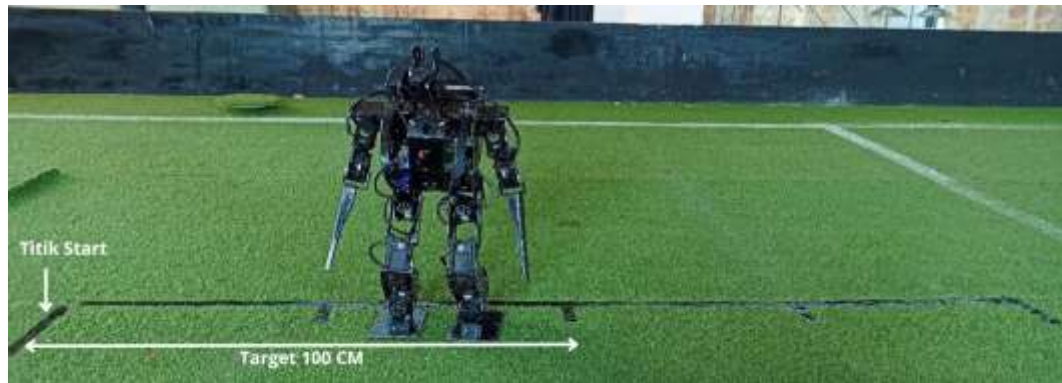
menerus sehingga dapat merubah arah hadap posisi robot dan ketidak mampuan robot dalam menjaga keseimbangan tubuh pada saat berjalan sehingga mempengaruhi performa robot dalam berjalan.

### 4.3.3 Pengujian Berjalan Ke Kanan

Pengujian ini melibatkan gerakan berjalan robot yang dilakukan di area lapangan berumput dengan menambahkan nilai tertentu pada sumbu kartesian. Cara pengujian dilakukan dengan meletakkan robot pada titik 0 atau posisi awal dan kemudian pada posisi akhir robot bergerak akan dilakukan pengukuran dengan menggunakan alat ukur berupa meteran, dan perbedaan yang didapatkan antara nilai yang diinginkan (*set point*) dengan nilai aktual dianggap sebagai nilai *error*. Metode pengujian pada tahap ini melibatkan perubahan nilai pada sumbu y dari posisi yang diharapkan untuk mengevaluasi seberapa baik sistem merespon perintah pergerakan. Pada pengujian ini dilakukan sebanyak sepuluh kali dimana robot diperintahkan berjalan ke kanan dengan memberikan kecepatan pada sumbu y sebesar -2,5 cm pada setiap langkahnya dan jumlah langkah yang berbeda-beda mulai dari 10 langkah sampai dengan 40 langkah pada setiap pengujian. Berikut merupakan pengujian dari gerak berjalan ke kanan dapat dilihat pada gambar 4.8.



(a)



(b)



(c)

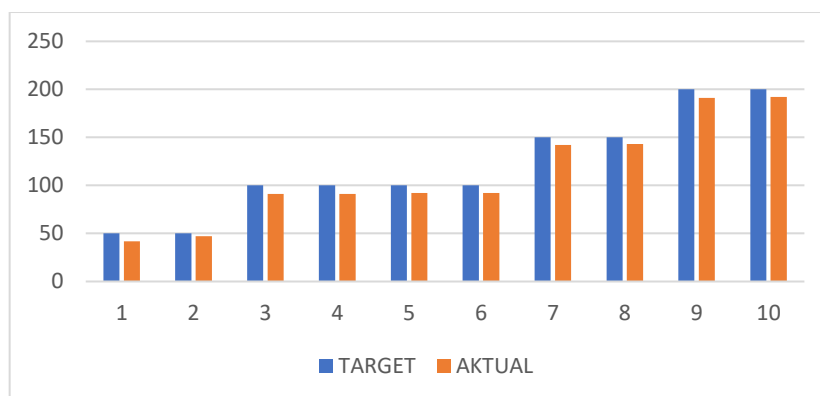


(d)

**Gambar 4.8** Hasil Pengujian Berjalan Ke Kanan (a) Target 50 cm, (b) Target 100 cm, (c) Target 150 cm, dan Target 200 cm.

Pengujian dilakukan pada lapangan berumput dimana robot diberikan perintah berjalan ke kanan dengan target jarak tempuh sepanjang 50 cm sampai dengan 200 cm. Pada pengujian ini robot diberikan nilai parameter berupa kecepatan yaitu -2,5 cm pada setiap langkah dan jumlah langkah keseluruhan yaitu mulai dari 10 langkah sampai dengan 40 langkah. Dapat dilihat pada gambar 4.8 dimana robot sudah berusaha mendekati target yang diinginkan. Nilai target yang ditetapkan

menggambarkan performa yang diharapkan atau tujuan yang ingin dicapai, sementara nilai aktual mencerminkan kinerja sebenarnya dari robot saat menjalankan pergerakan ke kanan. Pada pengujian robot berjalan ke kanan, terdapat perbandingan antara nilai target dan nilai aktual yang tercermin dengan jelas. Perbandingan tersebut dapat dilihat pada grafik gambar 4.9.



**Gambar 4.9** Grafik Hasil Pengujian Berjalan Ke Kanan

Pengujian telah dilakukan sebanyak sepuluh kali untuk mengevaluasi akurasi robot dalam mencapai jarak pendek maupun jarak jauh. Rangkuman hasil pengujian posisi robot terhadap target berjalan ke kanan ini dapat dilihat dalam Tabel 4.5 dan 4.6 yang terlampir di bawah ini.

**Tabel 4.5** Posisi Aktual Robot Terhadap Target Ke Kanan

No	Posisi Target (cm)	Posisi Aktual (cm)	Selisih (cm)	<i>Error</i> (%)
1	50	42	8	16%
2	50	47	3	6%
3	100	91	9	9%
4	100	91	9	9%
5	100	92	8	8%
6	100	92	8	8%

**Tabel 4.5** Posisi Aktual Terhadap Target Ke Kanan (Lanjutan)

No	Posisi Target (cm)	Posisi Aktual (cm)	Selisih (cm)	<i>Error</i> (%)
7	150	142	8	5,3%
8	150	143	7	4,6%
9	200	191	9	4,5%
10	200	192	8	4%
Rata – Rata Posisi Aktual		112,3		
Rata – Rata Presentase <i>Error</i>			7,4%	

Hasil dari pengujian robot berjalan ke kanan dapat dilihat pada tabel 4.5, diketahui bahwa rata-rata posisi aktual terhadap target ke kanan sebesar 112,3cm. Nilai *error* pengukuran didapatkan dari pembagian nilai selisih dengan nilai target kemudian dikalikan 100%.

$$Error = \frac{\text{Selisih}}{\text{Target}} \times 100\%$$

Pada keseluruhan pengujian didapatkan nilai rata-rata *error* terhadap target ke kanan yaitu sebesar 7,4%. Sedangkan hasil pengujian arah hadap dapat dilihat pada tabel 4.6 yang terlampir dibawah ini.

**Tabel 4. 6** Posisi Arah Hadap

No	Posisi Target (°)	Posisi Aktual (°)	Selisih (°)
1	0	10	10
2	0	10	10
3	0	3	3
4	0	3	3

**Tabel 4.6** Posisi Arah Hadap (Lanjutan)

No	Posisi Target (°)	Posisi Aktual (°)	Selisih (°)
5	0	10	10
6	0	6	6
7	0	9	9
8	0	4	4
9	0	8	8
10	0	7	7
Rata – Rata <i>Error</i>			7

Hasil pengujian pada arah hadap robot terdapat penyimpangan sebesar 7°, nilai tersebut merupakan selisih antara target dengan posisi akhir dari robot pada saat melakukan pergerakan berjalan ke kanan. Hal ini mengindikasikan bahwa adanya deviasi antara posisi yang diinginkan dan posisi yang dicapai oleh robot dalam pergerakan ke kanan. Kemungkinan besar, nilai error ini terjadi karena setiap nilai servo pada posisi default kaki robot tidak sepenuhnya akurat. Selain itu nilai error ini disebabkan oleh gesekan kaki robot dengan lapangan yang terjadi secara terus menerus sehingga dapat merubah arah hadap posisi robot dan ketidak mampuan robot dalam menjaga keseimbangan tubuh pada saat berjalan sehingga mempengaruhi performa robot dalam berjalan.

#### 4.3.4 Pengujian Berjalan Ke Kiri

Proses pengujian ini melibatkan gerakan berjalan robot yang dilakukan di area lapangan berumput dengan menambahkan nilai tertentu pada sumbu kartesian. Pengujian dilakukan dengan cara meletakkan robot pada titik 0 atau

posisi awal yang kemudian pada posisi akhir robot setelah bergerak yang sebenarnya akan diukur dengan menggunakan alat ukur berupa meteran, dan perbedaan yang didapatkan antara nilai yang diinginkan (*set point*) dengan nilai aktual dianggap sebagai nilai *error*. Metode pengujian pada tahap ini melibatkan perubahan nilai pada sumbu y dari posisi yang diharapkan untuk mengevaluasi seberapa baik sistem merespon perintah pergerakan. Pada pengujian ini dilakukan sebanyak sepuluh kali dimana robot diperintahkan berjalan ke kiri dengan memberikan kecepatan pada sumbu y sebesar 2,5 cm pada setiap langkahnya dan jumlah langkah yang berbeda-beda mulai dari 10 langkah sampai dengan 40 langkah pada setiap pengujian. Berikut merupakan pengujian dari gerak berjalan ke kiri dapat dilihat pada gambar 4.7.



(a)



(b)





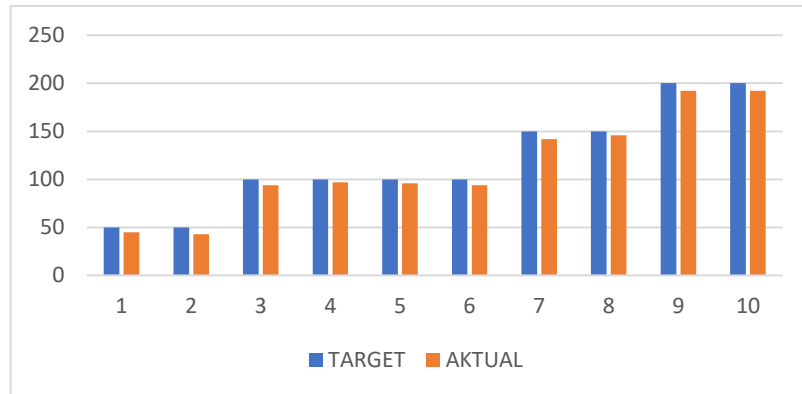
(c)



(d)

**Gambar 4.10** Hasil Pengujian Berjalan Ke Kiri (a) Target 50 cm, (b) Target 100 cm, (c) Target 150 cm, dan Target 200 cm.

Pengujian dilakukan pada lapangan berumput dimana robot diberikan perintah berjalan ke kiri dengan target jarak tempuh sepanjang 50 cm sampai dengan 200 cm. Pada pengujian ini robot diberikan nilai parameter berupa kecepatan yaitu 2,5 cm pada setiap langkah dan jumlah langkah keseluruhan yaitu 10 langkah sampai dengan 40 langkah. Dapat dilihat pada gambar 4.10 dimana robot sudah berusaha mendekati target yang diinginkan. Nilai target yang ditetapkan menggambarkan performa yang diharapkan atau tujuan yang ingin dicapai, sementara nilai aktual mencerminkan kinerja sebenarnya dari robot saat menjalankan pergerakan ke kiri. Pada pengujian robot berjalan ke kiri, terdapat perbandingan antara nilai target dan nilai aktual yang tercermin dengan jelas. Perbandingan tersebut dapat dilihat pada grafik gambar 4.11.



**Gambar 4.11** Grafik Hasil Pengujian Berjalan Ke Kiri

Pengujian telah dilakukan sebanyak sepuluh kali untuk mengevaluasi akurasi robot dalam mencapai jarak pendek maupun jarak jauh. Rangkuman hasil pengujian posisi robot terhadap target berjalan ke kiri ini dapat dilihat dalam Tabel 4.7 dan 4.8 yang terlampir di bawah ini.

**Tabel 4. 7** Posisi Aktual Robot Terhadap Target Ke Kiri

No	Posisi Target (cm)	Posisi Aktual (cm)	Selisih (cm)	<i>Error</i> (%)
1	50	45	5	10%
2	50	43	7	14%
3	100	94	6	6%
4	100	97	3	3%
5	100	96	4	4%
6	100	94	6	6%
7	150	142	8	5,3%
8	150	146	4	2,6%
9	200	192	8	4%
10	200	192	8	4%
Rata - Rata		114,1		
Rata – Rata Presentase <i>Error</i>			5,9%	

Hasil dari pengujian robot berjalan ke kiri dapat dilihat pada tabel 4.7, diketahui bahwa rata-rata posisi aktual terhadap target ke kiri sebesar 114,1cm. Nilai *error*

pengukuran didapatkan dari pembagian nilai selisih dengan nilai target kemudian dikalikan 100%.

$$Error = \frac{\text{Selisih}}{\text{Target}} \times 100\%$$

Pada keseluruhan pengujian didapatkan nilai rata-rata error terhadap target ke kiri yaitu sebesar 5,9%. Sedangkan hasil pengujian arah hadap dapat dilihat pada tabel 4.8 yang terlampir dibawah ini.

**Tabel 4. 8** Posisi Arah Hadap

No	Posisi Target (°)	Posisi Aktual (°)	Selisih (°)
1	0	8	5
2	0	10	5
3	0	3	3
4	0	8	7
5	0	10	3
6	0	9	6
7	0	5	4
8	0	5	7
9	0	7	8
10	0	8	9
Rata – Rata <i>Error</i>			5,7

Hasil pengujian pada arah hadap robot terdapat penyimpangan sebesar 5,7°, nilai tersebut merupakan selisih antara target dengan posisi akhir dari robot pada saat melakukan pergerakan berjalan ke kiri. Hal ini mengindikasikan bahwa adanya deviasi antara posisi yang diinginkan dan posisi yang dicapai oleh robot dalam pergerakan ke kiri. Kemungkinan besar, nilai error ini terjadi karena setiap nilai servo pada posisi default kaki robot tidak sepenuhnya akurat. Selain itu nilai error ini disebabkan oleh gesekan kaki robot dengan lapangan yang terjadi secara terus

menerus sehingga dapat merubah arah hadap posisi robot dan ketidak mampuan robot dalam menjaga keseimbangan tubuh pada saat berjalan sehingga mempengaruhi performa robot dalam berjalan.

#### 4.3.5 Hasil Pengujian Keseluruhan

Dalam mengevaluasi keseluruhan pengujian dari serangkaian uji coba yang mencakup berbagai pergerakan seperti berjalan maju, berjalan mundur, berjalan ke kanan, dan berjalan ke kiri yang dilakukan sebanyak sepuluh kali pada setiap pergerakan, maka hasil tersebut dipresentasikan secara rinci dan terstruktur dalam sebuah tabel. Tabel ini akan mencakup data hasil uji coba berbagai pergerakan robot yang telah dilakukan. Hasil keseluruhan pengujian dapat dilihat pada tabel 4.9 dan 4.10 yang terlampir dibawah ini.

**Tabel 4. 9** Hasil Pengujian Keseluruhan Jarak Tempuh

No	Pengujian	Rata- Rata Posisi Aktual (cm)	Rata – Rata <i>Error</i> (%)
1	Berjalan Maju	115,1	4,6
2	Berjalan Mundur	115,8	3,4
3	Berjalan Ke Kanan	112,3	7,4
4	Berjalan Ke Kiri	114,1	5,9
Rata - Rata		114,3	5,3

**Tabel 4. 10** Hasil Pengujian Keseluruhan Arah Hadap

No	Pengujian	Rata- Rata Posisi Akhir (°)	Rata – Rata <i>Error</i> (%)
1	Berjalan Maju	5,7	5,7
2	Berjalan Mundur	6,9	6,9

**Tabel 4. 11** Hasil Pengujian Keseluruhan Arah Hadap (Lanjutan)

No	Pengujian	Rata- Rata Posisi Akhir (°)	Rata – Rata <i>Error</i> (%)
3	Berjalan Ke Kanan	7	7
4	Berjalan Ke Kiri	5,7	5,7
Rata - Rata		6,3	6,3

kesimpulan dari pengujian keseluruhan yang telah diakumulasikan dalam tabel di atas, dapat disimpulkan bahwa rata-rata jarak tempuh yang diperoleh adalah sebesar 114,3cm dengan tingkat kesalahan sebesar 5,3%. Sementara itu, pada penyimpangan arah hadap, nilai rata-rata yang didapatkan adalah sebesar 6,3° dengan tingkat kesalahan sebesar 6,3%. hasil ini memberikan gambaran komprehensif tentang performa robot dalam berbagai pergerakan, dengan memperhatikan akurasi dan ketepatan dalam mencapai target jarak dan arah hadap yang diinginkan.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan uraian yang telah dijelaskan pada BAB I sampai dengan BAB IV dan hasil analisis serta pengujian yang telah dilakukan pada setiap kategori, maka ditarik kesimpulan sebagai berikut:

1. Penelitian ini berhasil mengimplementasikan metode dead reckoning untuk memprediksi lokasi relative dimana melibatkan pengukuran dan perhitungan perubahan posisi berdasarkan gerakan robot dari titik awalnya. Hasil pengujian metode tersebut memverifikasi bahwa metode tersebut layak digunakan, meskipun terdapat nilai error, yakni sebesar 5,3% pada jarak tempuh dan 6,3% pada arah hadap robot. penelitian ini menunjukkan meskipun terdapat beberapa ketidakakuratan, metode tersebut masih memberikan hasil yang dapat diandalkan dan cukup akurat dalam memprediksi lokasi relatif.
2. Hasil pengujian pada berjalan maju terdapat nilai error jarak tempuh sebesar 4,6% dan penyimpangan arah sebesar  $5,7^{\circ}$ , pengujian berjalan mundur terdapat nilai error jarak tempuh sebesar 3,4% dan penyimpangan arah sebesar  $6,9^{\circ}$ , pengujian berjalan ke kanan terdapat nilai error jarak tempuh sebesar 7,4% dan penyimpangan arah sebesar  $7^{\circ}$  dan pengujian berjalan ke kiri terdapat nilai error jarak tempuh sebesar 5,9% dan penyimpangan arah sebesar  $7,3^{\circ}$ .
3. Ketidak akuratan dari nilai yang dihasilkan terjadi karena beberapa factor yaitu gesekan antara kaki robot dengan arena uji coba, gerakan tambahan yang dihasilkan oleh robot pada saat berjalan dikarenakan ketidak mampuan

robot untuk menyeimbangkan tubuh pada saat berjalan, dan mekanika robot kurang presisi.

## **5.2 Saran**

Untuk pengembangan penelitian ini, beberapa saran yang dapat diusulkan meliputi:

1. Menambahkan sensor dan melakukan kalibrasi guna meminimalisir kesalahan dari pergerakan robot sehingga pada akhirnya robot dapat bergerak secara presisi.
2. Mengeksplorasi dan melakukan penelitian lebih lanjut untuk mengatasi masalah akurasi dalam pengukuran jarak menggunakan prinsip dead reckoning sehingga robot Krakatau FC memiliki perangkat dead reckoning yang baik atau melakukan penelitian dengan menggunakan metode lain.

## DAFTAR PUSTAKA

- Adriansyah, A. (2014). Perancangan Robot Localization Menggunakan Metode Dead Reckoning. *Sinergi*, 18(1), 25–30.
- Anragani, E. N. (2016). Sistem Perencanaan Gerakan Berjalan Robot Humanoid R2CR9 Bioloid GP Menggunakan Metode Proyeksi Bidang Kartesian. 10–29.
- Harjoko, A. (2019). Sistem Pendeteksian Marker pada Analisis Gait Menggunakan Pengolahan Citra Digital. *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, 3(2), 137–146.
- Leonardo, P. (2011). Dead Reckoning for Mobile Robots Using Two Optical Mice. 87–94. <https://doi.org/10.5220/0001143000870094>
- Baek, K. R. (2011). A dead reckoning localization system for mobile robots using inertial sensors and wheel revolution encoding. *Journal of Mechanical Science and Technology*, 25(11), 2907–2917. <https://doi.org/10.1007/s12206-011-0805-1>
- Candra, F. (2021). Analisa Implementasi Sensor Gyroscope Pada Pergerakan Robot Humanoid Berbasis Open Cm 9.04. 8, 1–9.
- Asama, H. (2013). Development of open platform humanoid robot DARwIn-OP. *Advanced Robotics*, 27(3), 223–232. <https://doi.org/10.1080/01691864.2012.754079>
- Cai, J. (2021). Real-Time Evaluation Algorithm of Human Body Movement in Football Training Robot. *Mathematical Problems in Engineering*, 2021. <https://doi.org/10.1155/2021/9932737>
- Zhang, J. (2015). A humanoid robot localization method for biped navigation in human-living environments. 2015 IEEE International Conference on Cyber



- Technology in Automation, Control and Intelligent Systems, IEEE-CYBER 2015, March 2019, 540–544. <https://doi.org/10.1109/CYBER.2015.7287997>
- Silvia, A. (2017). Sistem navigasi pada. Seminar Nasional Teknologi Informasi, Bisnis, Dan Desain 2017, 6(1), 129–134.
- Sugihara, T. (2015). Dead reckoning for biped robots that suffers less from foot contact condition based on anchoring pivot estimation. *Advanced Robotics*, 29(12), 785–799. <https://doi.org/10.1080/01691864.2015.1011694>
- Permatasari, G. A. (2016). ANALISIS PERBEDAAN PENGARUH SEPATU BERHAK WEDGE DAN NON-WEDGE TERHADAP GAIT DAN KESEIMBANGAN.
- Pratomo, A. (2013). Implementasi Algoritma Dead Reckoning untuk Perbaikan Penentuan Posisi di Ruang Indoor untuk Pejalan Kaki (Vol. 2, Issue 1, pp. 1–5).
- Hoffmann, M. (2011). Dead reckoning in a dynamic quadruped robot: Inertial navigation system aided by a legged odometer. *Proceedings - IEEE International Conference on Robotics and Automation*, 617–624. <https://doi.org/10.1109/ICRA.2011.5979609>
- Kawuri, L. D. (2012). Sistem Navigasi untuk Individu Pejalan Kaki Berbasis Prinsip Dead Reckoning. *Transmisi*, 14(2), 42-47–47.
- Chan, C.-C. (2014). Decentralized EIF-based global localization using dead-Reckoning, KINECT and laser scanning for autonomous omnidirectional mobile robot. 1. <https://ieeexplore.ieee.org/document/6871497>
- Udinus. (2013). *Prosiding Isrsc 2013*.
- Wibowo, W. T. (2017). Implementasi Inverse Kinematics Pada Pergerakan Robot

Quadruped. Its, 1–110.

Xiong, J. (2017). A central pattern generator approach to footstep transition for biped navigation. *International Journal of Advanced Robotic Systems*, 14(1), 1–9. <https://doi.org/10.1177/1729881416682708>

Yilmaz, M. (2010). *Humanoid Robot Omnidirectional Walking Trajectory*. August.

