

**T.C.  
İZMİR DEMOKRASİ ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

**MÜHENDİSLİK TEZİ  
MAKİNE MÜHENDİSLİĞİ BÖLÜMÜ**

**ROS TABANLI ÇALIŞAN ÖZERK BİR MOBİL ROBOT**

**Oğuzhan BOZOĞLU**

**Danışman  
Doç. Dr. Mehmet İTİK**



**İZMİR-2023**

**Oğuzhan  
BOZOĞLU**

**ROS TABANLI ÇALIŞAN ÖZERK BİR MOBİL ROBOT**

**2023**

## **TEZ ONAYI**

**Oğuzhan BOZOĞLU** tarafından hazırlanan “**ROS Tabanlı Çalışan Özerk Bir Mobil Robot**” adlı proje çalışması 02/02/2023 tarihinde aşağıdaki jüri üyeleri önünde İzmir Demokrasi Üniversitesi Fen Bilimleri Enstitüsü **Makine Mühendisliği Anabilim Dalı**’nda **MÜHENDİSLİK TEZİ** olarak başarı ile savunulmuştur.

**Danışman/Jüri**

**Doç. Dr. Mehmet İTİK**  
İzmir Demokrasi Üniversitesi

## **TAAHHÜTNAME**

Bu tezin İzmir Demokrasi Üniversitesi Mühendislik Fakültesi Makine Mühendisliği Bölümü'nde, akademik ve etik kurallara uygun olarak yazıldığını ve kullanılan tüm literatür bilgilerinin referans gösterilerek tezde yer aldığını beyan ederim.

**Oğuzhan BOZOĞLU**

# İÇİNDEKİLER

## Sayfa

İÇİNDEKİLER.....	I
SİMGELER VE KISALTMALAR DİZİNİ.....	II
GÖRSEL DİZİNİ.....	III
ŞEKİLLER DİZİNİ.....	IV
TEŞEKKÜR.....	V
ÖZET.....	VI
ABSTRACT.....	VII
1. GİRİŞ.....	1
2. GENEL BAKIŞ.....	3
2.1 Donanım ve Sistem Mimarisi.....	3
2.2 Sensörler.....	4
2.3 Aktüatörler.....	5
2.4 Mikroişlemci.....	6
3. OTONOM MOBİL ROBOTLAR İÇİN KONTROL TEORİSİ.....	8
3.1 Ackermann Yönteminin Açıklanması.....	9
3.2 Ackermann Geometrisinin Kodlanması ve Açıklanması.....	10
3.3 Gazebo ve RViz Ortamında Sensör ve Bileşenlerin Çıktıları.....	12
3.4 Gazebo ve RViz Ortamında Mobil Robotun Hareketi.....	13
4. SONUÇ VE ÖNERİLER.....	17
REFERANSLAR.....	18

## **SİMGELER VE KISALTMALAR DİZİNİ**

<b>ROS</b>	Robot Operating System
<b>LiDAR</b>	Laser Imaging Detection and Ranging
<b>RViz</b>	ROS Visualization
<b>IMU</b>	Ineartial Measurement Unit
<b>PID</b>	Proportional, Integral, Derivative

## GÖRSEL DİZİNİ

	<b>Sayfa</b>
Resim 1. Robot ve donanım .....	4
Resim 2. YD Lidar 2X .....	4
Resim 3. Raspberry Pi kamera modülü .....	5
Resim 4. IMU Modülü .....	5
Resim 5. 12V DC motor .....	6
Resim 6. PCA 9685 Servo Sürücü Kartı .....	6
Resim 7. L298N Voltaj Regülatörlü Motor Sürücü Kartı .....	6
Resim 8. Raspberry Pi Model 3B+ .....	7
Resim 9. Gazebo ortamında mobil robot .....	12
Resim 10. Gazebo test ortamı .....	12
Resim 11. Mobil robotun bağlantı ve koordinat gösterimi .....	13
Resim 12. Ortamın ve robotun eylemsizlik gösterimleri .....	13
Resim 13. Robotun hareketi ve sensör verileri - 1 .....	14
Resim 14. Robotun hareketi ve sensör verileri - 2 .....	14
Resim 15. Robotun hareketi ve sensör verileri - 3 .....	14
Resim 16. Robotun hareketi ve sensör verileri - 4 .....	15
Resim 17. Robotun hareketi ve sensör verileri - 5 .....	15
Resim 18. Robotun hareketi ve sensör verileri - 6 .....	15

## ŞEKİLLER DİZİNİ

	<b>Sayfa</b>
Şekil 1. Ackermann direksiyon yöntemi .....	9
Şekil 2. Kütüphanelerin aktarılması .....	10
Şekil 3. Ackermann denkleminin oluşturulması .....	10
Şekil 4. Ackermann direksiyon yönteminin Python üzerinde çizilmesi .....	11
Şekil 5. Tekerlerin hareketi .....	11



## TEŞEKKÜR

Çalışmamın ve lisans öğrenim hayatımın tüm zorlu aşamalarında bana destek olan, bilgi ve deneyimleri ile yol gösteren, manevi her yönden yardımcı olan danışman hocam Sayın Doç. Dr. Mehmet İTİK'e, öğrenim hayatım boyunca beni maddi ve manevi olarak destekleyen ve güvenini sürekli hissettiğim hep yanımda olan aileme yürekten teşekkür ederim.

Oğuzhan BOZOĞLU  
İzmir, 2023

## **ÖZET**

### **Mühendislik Tezi**

### **ROS Tabanlı Çalışan Özerk Bir Mobil Robot**

**Oğuzhan BOZOĞLU**

**İzmir Demokrasi Üniversitesi  
Fen Bilimleri Enstitüsü  
Makine Mühendisliği Anabilim Dalı**

**Danışman: Doç. Dr. Mehmet İTİK**

Bu yazıda, Robot İşletim Sistemine (ROS) dayalı otonom bir mobil robotun geliştirilmesini ve uygulanmasını sunmaktayım. Robot, Gazebo simülasyon ortamında simüle edilmiş ve RViz'de görselleştirilmiştir. Robotun kontrolü, Ackermann Direksiyon kontrol algoritmasına dayalıdır. Robotun performansı değerlendirilir ve diğer mobil robot teknolojileri ile karşılaştırılır. Bu teknolojiye yönelik gelecek planlarında, daha fazla işlevsellik ve gerçek dünya uygulamaları için daha fazla iyileştirme ve diğer ROS paketleriyle entegrasyonu içerir.

**Anahtar Kelimeler: ROS, RViz, Gazebo, Mobil Robot, Ackermann Steering**

**2023, 30 sayfa**

## **ABSTRACT**

**B.Sc. Thesis**

**An Autonomous Mobile Robot Based on ROS  
Oguzhan BOZOGLU**

**Izmir Demokrasi University  
Graduate School of Applied and Natural Sciences  
Department of Mechanical Engineering**

**Supervisor: Assoc. Prof. Dr. Mehmet ITIK**

In this article, I present the development and implementation of an autonomous mobile robot based on the Robot Operating System (ROS). The robot was simulated in the Gazebo simulation environment and visualized in RViz. The control of the robot is based on the Ackermann Steering control algorithm. The performance of the robot is evaluated and compared with other mobile robot technologies. Future plans for this technology include further improvements for more functionality and real-world applications, and integration with other ROS packages.

**Keywords: ROS, RViz, Gazebo, Mobile Robot, Ackermann Steering**

**2023, 30 pages**

## 1. GİRİŞ

Robotik alanı, son yıllarda otonom mobil robotların geliştirilmesiyle hızla gelişmektedir. Otonom mobil robotlar; tarım [1], lojistik [2], sağlık [3] ve ev otomasyonu [4] dahil olmak üzere çeşitli alanlarda verimliliği ve üretkenliği artırma potansiyelleri nedeniyle büyük ilgi görüyor. Bu robotlar, çevreyi algılamalarına ve çevreyle etkileşime girmelerine olanak tanıyan çeşitli sensörler ve aktüatörlerle donatılarak, görevlerini minimum insan müdahalesi ile gerçekleştirmelerini sağlıyor. Bu alandaki en önemli gelişmelerden biri, Robot İşletim Sisteminin (ROS) [5] ortaya çıkışıdır. ROS, robotları inşa etmek, programlamak ve simüle etmek için bir dizi araç sağlayan, robot geliştirmeye yönelik açık kaynaklı bir yazılım çerçevesidir. ROS, birden çok programlama dilini destekler ve geliştiricilerin, farklı robotlar arasında yeniden kullanılabilen ve paylaşılabilen kodlar yazmasına olanak tanır. Bu makalenin amacı, Gazebo [6] simülasyon ortamını ve RViz [7,8] görselleştirme aracını kullanarak ROS çerçevesine dayalı otonom bir mobil robotun gelişimini sunmaktır.

Gazebo simülasyon ortamı, robot ortamının gerçekçi, fizik tabanlı bir modelini sağlayan güçlü bir araçtır. Kullanıcının robotunun kontrol algoritmalarını güvenli, kontrollü bir ortamda test etmesini ve iyileştirmesini sağlar. Bu, fiziksel bir robota ihtiyaç duymadan karmaşık algoritmaların ve sistemlerin test edilmesine izin verdiği için özellikle robotik araştırmaları için kullanışlıdır. RViz ise, kullanıcının robotun durumunu ve sensör verilerini görselleştirmesini sağlayan, ROS için bir 3D görselleştirme aracıdır. Robot ortamının gerçek zamanlı bir görünümünü sağlar ve kullanıcının davranışını anlamasına yardımcı olur.

Bölüm 2'de, otonom mobil robotumuzda kullanılan donanım ve yazılım bileşenlerine kapsamlı bir genel bakış sunacağım. Bu, kullanılan çeşitli sensörler ve aktüatörlerin yanı sıra robotun geliştirilmesinde kullanılan ROS paketleri ve kitaplıklarının bir tartışmasını içerecektir.

Bölüm 3, kontrol algoritmalarının tasarımı ve uygulanabilirliği, Gazebo simülasyon ortamının kullanımı ve robotun performansının test edilmesi ve açıklanması dahil olmak üzere robotun geliştirilmesinde kullanılan metodolojiyi sunacaktır.

Bölüm 4, deneylerin sonuçlarını sunacak ve geliştirilen otonom mobil robotun geliştirildiği son aşamayı gösterecektir. Ayrıca mobil robotumuz ile yapılacak olan gelecek çalışmaları mevcut son teknoloji otonom mobil robotlarla karşılaştıracak ve robotumuzun lojistik, teslimat ve arama kurtarma gibi çeşitli alanlardaki potansiyel uygulamalarını tartışacağız.

Mevcut otonom araç ve mobil robotlara örnek olarak Nvidia Self-Driving [9], Yandex Rover [10], Universal Robot UR5 [11], Kuka Agilus [12] ve Amazon Scout [13] verilebilir. Bu robotlar, haritalama ve navigasyon, nesne algılama ve manipülasyon ve dağıtım hizmetleri gibi çeşitli görevler için kullanılır.

Sonuç olarak, ROS çerçevesine dayalı otonom bir mobil robotun geliştirilmesi, robotik alanına değerli bir katkı sağlamaktadır. Gazebo simülasyon ortamının ve RViz görselleştirme aracının kullanımı, robot geliştirme ve test etme için sağlam ve verimli bir platform sağlar. Bu yazı, otonom mobil robotumuzun gelişimine kapsamlı bir genel bakış sağlamak ve bir dizi deney ve mevcut son teknoloji sistemlerle karşılaştırma yoluyla etkinliğini göstermektedir.

## 2. GENEL BAKIŞ

Bu bölümde otonom mobil robotun mimarisi ve donanım bileşenleri detaylı olarak ele alınmaya çalışılmıştır. Robot, sistemin kolay entegrasyonunu ve bakımını sağlayan modüler bir mimari ile tasarlanması amaçlanmıştır.

### 2.1 Donanım ve Sistem Mimarisi

Robotun temel donanım bileşenlerinden biri, ana bilgi işlem birimi olarak hizmet veren Raspberry Pi'dir [14]. Raspberry Pi, robotik uygulamalar için güçlü bir platform sağlayan düşük maliyetli, kredi kartı boyutunda bir bilgisayardır. Linux tabanlı bir işletim sisteminde çalışır ve robotik uygulamalarında yaygın olarak kullanılan Python dahil olmak üzere birden çok programlama dilini destekler. Performans açısından Raspberry Pi, düşük maliyetine karşın nispeten yüksek düzeyde hesaplama gücü sağlar ve bu da onu robotik uygulamalar için cazip bir seçim haline getirir. Raspberry Pi ile kullanıldığında ROS platformu, otonom mobil robotlar oluşturmak, dağıtmak ve yönetmek için esnek ve güçlü bir platform sağlar. Ayrıca ROS; gezinme, haritalama ve algılama gibi çeşitli görevleri gerçekleştirmek için bir dizi kitaplık ve araç sağlar ve bu görevlerin tek, uyumlu bir sisteme kolayca entegre edilmesini olanak tanır [15]. Otonom bir mobil robot bağlamında, Raspberry Pi tipik olarak kendisine bağlı sensörler ve aktüatörler gibi çeşitli donanım bileşenleri ile ana işlem birimi olarak kullanılır. Raspberry Pi, bu bileşenlerle USB veya I2C gibi çeşitli arabirimler aracılığıyla iletişim kurar ve robotun farklı bileşenlerini (aktüatörler, sensörler vb.) kontrol etmek ve koordine etmek için ROS platformunu kullanır [16].

Genel olarak, Raspberry Pi, ROS ve robotik kombinasyonu, otonom mobil robotlar oluşturmak ve dağıtmak için güçlü bir platform oluşturulmasında rol oynar. Bu platform, endüstriyel otomasyondan hizmet robotlarına kadar çok çeşitli robotik uygulamaların oluşturulmasına olanak tanır ve bu uygulamaların yönetilmesi için esnek ve ölçeklenebilir bir platform temin eder.

Robottaki diğer donanım bileşenleri arasında algılama için kullanılan LiDAR, kamera ve IMU gibi sensörler ve çevre ile etkileşim için kullanılan servo ve DC motorlar gibi

aktuatörler yer alıyor. Bu bileşenlerin teknik özellikleri ve yetenekleri bu bölümde açıklanmaktadır.



*Resim 1 Robot ve Donanım*

## 2.2 Sensörler

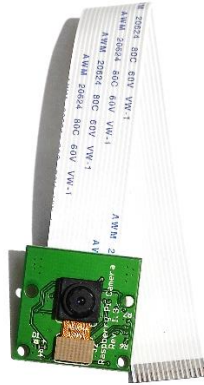
Robot, çevreyi algılamasını ve karar vermesini sağlayan bazı sensörlerle donatılmıştır. Robotta kullanılan sensörler arasında engel tespiti için bir LiDAR sensörü, görsel algılama için bir kamera ve robotun yönünü ve konumunu belirlemek için bir IMU bulunmaktadır [17].

- LiDAR sensör, 360 derecelik kapsama alanı sağlar ve 8 metreye kadar menzile sahiptir [18].



*Resim 2 YD Lidar 2X*

- Kamera, Raspberry Pi tarafından uyumlu olarak basit bir yapıya sahiptir ve görsel nesne tanıma ve izleme için kullanılır.



### Resim 3 Raspberry Pi Kamera Modülü

- IMU, robotun yönünün doğru ölçümelerini sağlar ve kesin hesaplama ve navigasyon için kullanılır.



### Resim 4 IMU Modülü

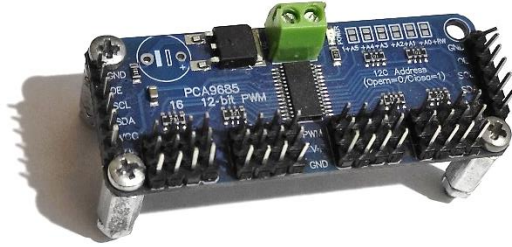
## 2.3 Aktuatörler

Robot, tekerlekleri tahrik eden ve tahrik sağlayan DC ve servo olmak üzere iki motorla donatılmıştır. Motorlar, mikro işlemciden komutlar alan ve motor hızını ve yönünü düzenleyen bir motor denetleyici tarafından kontrol edilir.

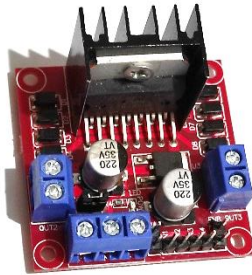




*Resim 5 12V DC Motor*



*Resim 6 PCA 9685 Servo Sürücü Kartı*



*Resim 7 L298N Voltaj Regülatörlü Motor Sürücü Kartı*

## 2.4 Mikroişlemci

Mikroişlemci, robotun merkezi bileşenidir ve çeşitli donanım bileşenlerini koordine etmekten sorumludur. Burada mikroişlemcimiz Raspberry Pi 3b+ modelidir

ve robotta kullanılan kontrol algoritmalarını uygulayan yazılımı çalıştırır. Raspberry Pi, sensörler ve aktüatörlerle iletişim kurar ve bir Wi-Fi bağlantısı üzerinden ROS platformundan komutlar ve veriler alır.



*Resim 8 Raspberry Pi Model 3B+*

Özetle, otonom mobil robotun mimarisi ve donanım bileşenleri, yapılandırılmamış ve düzgün sürüşe uygun olmayan ortamlarda çeşitli görevleri yerine getirmek için sağlam ve güvenilir bir sistem sağlamak üzere birlikte çalışacak şekilde tasarlanması amaçlanmıştır. Donanım bileşenleri, robotun görevlerini etkili bir şekilde yerine getirebilmesini sağlamak için ilk olarak maliyet faktörü göz önüne alınarak daha sonra doğruluk, güvenilirlik ve ROS platformuyla uyumluluğa göre seçilmiştir. Her çalışmada olduğu gibi yüksek kaliteli sensörlerin ve aktüatörlerin kullanımı yüksek doğruluk ve güvenilirlik sağlayacaktır.

### 3. OTONOM MOBİL ROBOTLAR İÇİN KONTROL TEORİSİ

Otonom mobil robotlar, adından da anlaşılacağı gibi, etkin bir şekilde çalışabilmeleri için kendilerini yönlendirebilmeli ve kontrol edebilmelidir. Kontrol teorisi, dinamik sistemlerin kontrolü ile ilgilenen matematik ve mühendislik disiplini ve otonom mobil robotların geliştirilmesinde çok önemli bir rol oynar [19]. Bu bölümde, Ackermann yönlendirmesine (Ackermann Steering) odaklanarak otonom mobil robotlar alanında yaygın olarak kullanılan kontrol teorilerini ve yöntemlerini keşfedeceğiz.

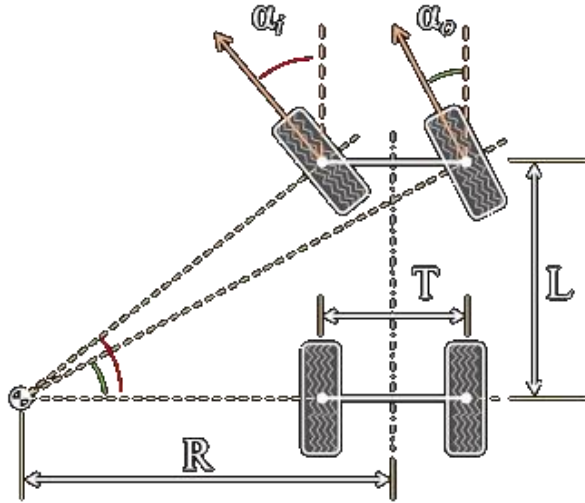
Ackermann yönlendirmesi veya yöntemi, arabalar, kamyonlar ve tekerlekli robotlar gibi hem yön kontrolü hem de manevra kabiliyeti gerektiren tekerlekli araçlarda kullanılan bir direksiyon geometrisi türüdür. Ackermann direksiyon geometrisi, dönüşler sırasında lastik aşınmasını veya tekerleklerin yana kaymasını en aza indirecek şekilde tasarlanmıştır. Bu, otonom mobil robotların stabilitesi ve manevra kabiliyeti için önemli olan daha hassas ve verimli yönlendirmeye olanak tanır [21]. Otonom mobil robotlar alanında kullanılan doğrusal ve doğrusal olmayan kontrol, uyarlamalı kontrol ve model öngörülü kontrol gibi çeşitli kontrol teorileri ve yöntemleri vardır. Bu teoriler ve yöntemler, kararlılık, yörünge izleme ve hareket planlama gibi farklı kontrol hedeflerine ulaşmak için çeşitli şekillerde birleştirilebilir ve uygulanabilir [22]. Örneğin, otonom mobil robotlarda kullanılan yaygın bir kontrol yöntemi, orantısız-integral-türev (PID) denetleyicisidir. Bu tip kontrolör, kontrol sinyallerini ayarlamak ve istenen davranışı elde etmek için robotun sensörlerinden gelen geri bildirimi kullanır. Ek olarak, uyarlanabilir kontrol yöntemleri, sistemin davranışındaki değişikliklere dayalı olarak kontrol parametrelerini gerçek zamanlı olarak ayarlamak için kullanılabilir ve robotun genel performansını iyileştirir [19].

Sonuç olarak, bu yazı, hareket kontrolü için kontrol teorisinin kullanımına odaklanan, ROS'a dayalı otonom bir mobil robot sunmaktadır. Robotu tasarlamak ve test etmek için Gazebo simülasyon ortamı ve RViz görselleştirme aracı kullanılacak olup, robota uygulanabilecek bir kontrol tekniği örneği olarak Ackermann Geometrisi modeli kullanılacaktır.

### 3.1 Ackermann Yönteminin Açıklanması

Ackermann Geometrisi, otonom mobil robotlar dahil olmak üzere araçlarda ve mobil robotlarda yaygın olarak kullanılan bir direksiyon mekanizması türüdür. Bir dönüş sırasında iç tekerleğin dış tekerleğe göre daha küçük bir açıyla dönmesi ilkesine dayanır, bu da her iki tekerleğin de farklı yarıçaplarda dairesel yollar izlemesiyle sonuçlanır. Bu yöntem, dengeyi sağlar ve kaymayı önleyerek aracın veya robotun kontrolünü kolaylaştırır. Ackermann direksiyonun ana formülasyonu, istenen dönüş yarıçapına ve araç geometrisine dayalı olarak ön tekerlekler için direksiyon açılarının hesaplanmasını içerir. Direksiyon açıları genellikle dingil mesafesinin (ön ve arka akslar arasındaki mesafe) ve iz genişliğinin (ön tekerleklerin merkezleri arasındaki mesafe) bir fonksiyonu olarak ifade edilir.

Ackermann yönlendirmesini otonom bir mobil robota uygulamak için, öncelikle robotun özelliklerine ve operasyonel gereksinimlerine göre istenen dönüş yarıçapı belirlenmelidir. Ardından, ön tekerleklerin yönlendirme açıları aşağıdaki formüller kullanılarak hesaplanabilir:



Şekil 1 Ackermann Direksiyon Yöntemi

$$\delta_L = \arctan\left(\frac{L}{R - \frac{T}{2}}\right), \quad \delta_R = \arctan\left(\frac{L}{R + \frac{T}{2}}\right) \quad (\text{Denklem 1})$$

Burada  $\delta_L$  ve  $\delta_R$  sırasıyla sol ve sağ tekerleklerin yönlendirme açılarıdır,  $L$  dingil mesafesi,  $T$  sağ ve sol tekerler arası mesafe ve  $R$  dönüş yarıçapıdır.

Direksiyon açıları hesaplandıktan sonra, mekanik bağlantı, hidrolik sistem veya elektrikli aktüatör gibi uygun bir çalıştırma sistemi aracılığıyla robotun ön tekerleklerine uygulanabilirler. Çalıştırma sisteminin seçimi, maliyet, güvenilirlik, güç tüketimi ve kontrol hassasiyeti gibi faktörlere bağlıdır. Yönlendirme açılarına ek olarak, otonom mobil robotlarda Ackermann yönlendirme uygulamasındaki diğer önemli hususlar arasında kontrol sisteminin kararlılığı ve doğruluğu, geri bildirim ve kontrol için kullanılan sensörler ve algoritmalar, direksiyon ile sürüş sistemi ve navigasyon sensörleri gibi robotun diğer bileşenleri arasındaki etkileşimler yer alır.

Genel olarak, Ackermann direksiyonu, otonom mobil robotlar için kontrol sisteminin temel bir bileşenidir ve ilkelerinin ve uygulamasının derinlemesine anlaşılması, etkili otonom mobil robotların tasarımı ve geliştirilmesi için çok önemlidir [20].

### 3.2 Ackermann Geometrisinin Kodlanması ve Açıklanması

Ackermann yönteminin kodlama işlemi Python programlama dili ve Visual Studio Code editörü kullanılarak gerçekleştirildi.

#### 1. Kullanılacak kütüphanelerin aktarılması

```
import numpy as np
import matplotlib.pyplot as plt
```

*Şekil 2 Kütüphanelerin aktarılması*

Matematiksel işlemleri Python üzerinde gerçekleştirebilmek için Numpy kütüphanesinden yararlanıyoruz. Yaptığımız işlemleri ise grafik üzerine uygulayabilmemiz için ise Matplotlib kütüphanesini kullanıyoruz.

#### 2. Ackermann denkleminin oluşturulması

```
def ackermann_steering(L, T, R):
    inner_l_wheel_angle = np.arctan(L / (R-(T/2)))
    inner_r_wheel_angle = np.arctan(L / (R+(T/2)))
    return inner_l_wheel_angle, inner_r_wheel_angle
```

*Şekil 3 Ackermann denkleminin oluşturulması*

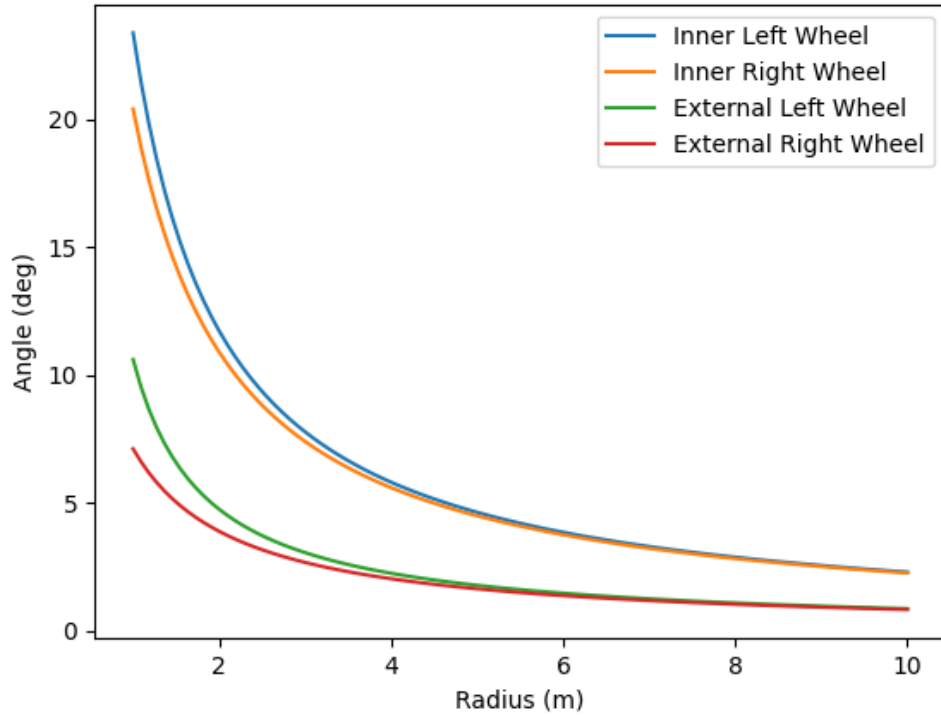
*Denklem 1* 'in Python üzerinde oluşturulması.

### 3. Ackermann denkleminin Python ortamında çizilmesi

```
def plot_ackermann_steering(L, T):  
    R = np.linspace(1, 10, 100)  
    inner_l_angle_deg, inner_r_angle_deg = np.degrees(ackermann_steering(L, T, R))  
    external_l_angle_deg = np.degrees(np.arctan(T / (R-(L/2))))  
    external_r_angle_deg = np.degrees(np.arctan(T / (R+(L/2))))  
  
    fig, ax = plt.subplots()  
    ax.plot(R, inner_l_angle_deg, label='Inner Left Wheel')  
    ax.plot(R, inner_r_angle_deg, label='Inner Right Wheel')  
    ax.plot(R, external_l_angle_deg, label='External Left Wheel')  
    ax.plot(R, external_r_angle_deg, label='External Right Wheel')  
    ax.set_xlabel('Radius (m)')  
    ax.set_ylabel('Angle (deg)')  
    ax.legend()  
    plt.show()
```

Şekil 4 Ackermann direksiyon yönteminin Python üzerinde çizimi

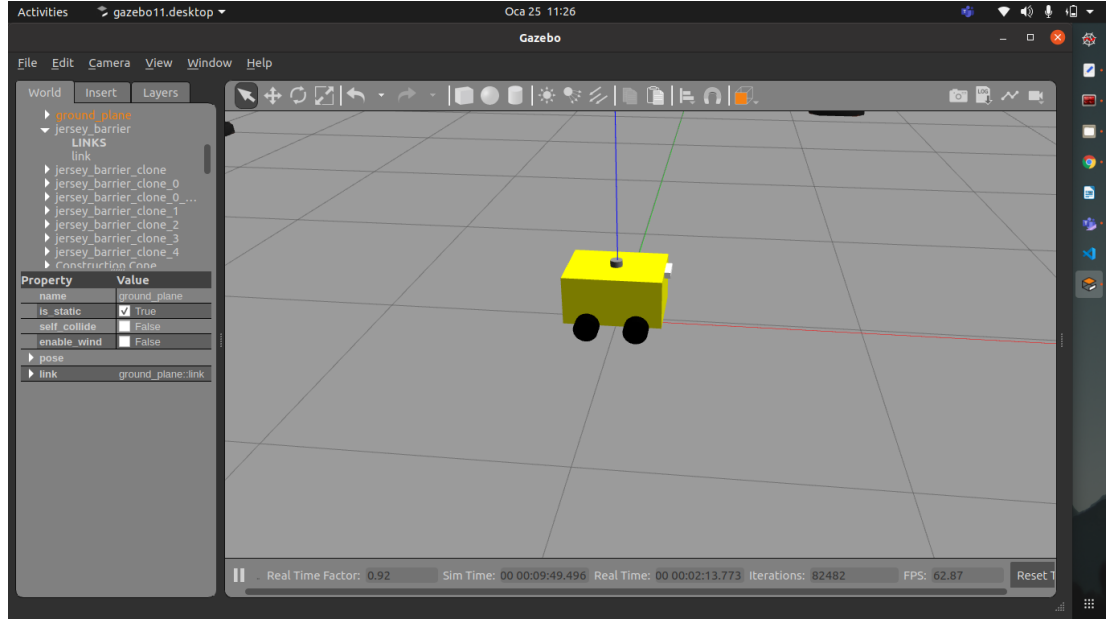
Denklemleri oluşturduktan sonra ön ve arka teker arasındaki mesafe  $L=0.4$  ve sağ ve sol tekerler arasındaki mesafe  $T=0.15$  parametrelerini vererek oluşan çizim Şekil 2' deki gibidir.



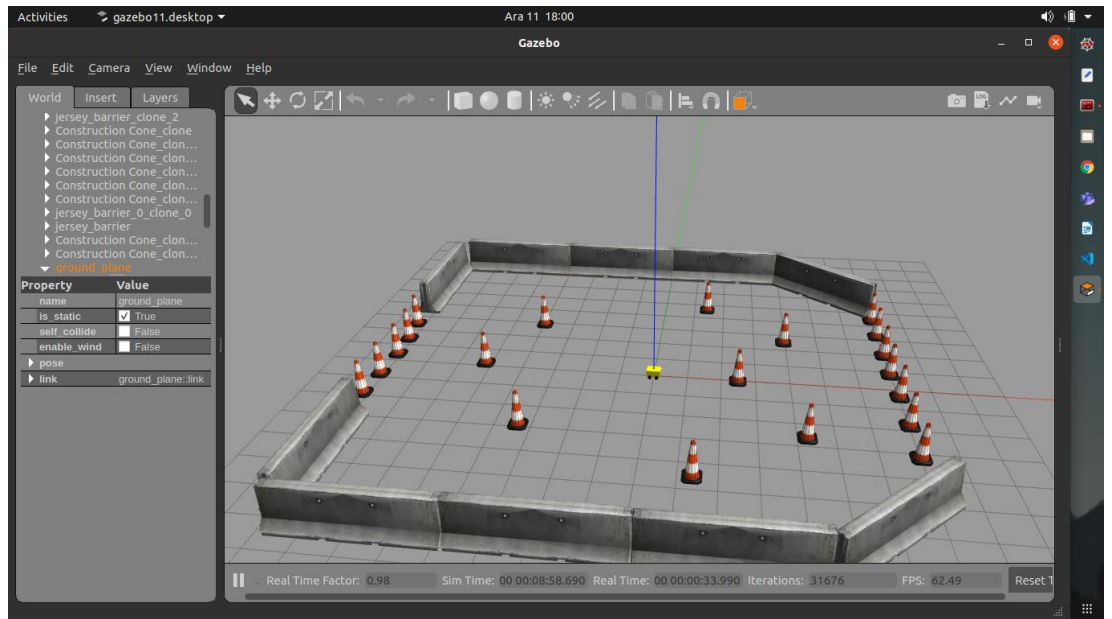
Şekil 5 Tekerlerin hareketi

Genel olarak bu örneği açıklamak gerekirse, **plot\_ackermann\_steering** işlevi, sırasıyla ön ve arka tekerlekler arasındaki mesafeyi ve sol ve sağ tekerlekler arasındaki mesafeyi temsil eden parametreler olarak L ve T'yi alır. Bu işlev daha sonra **np.linspace**'i kullanarak bir dizi olası eğri yarıçapı oluşturur ve **ackermann\_steering** işlevini kullanarak iç sol ve sağ tekerleklerin açılarını hesaplar. Dış tekerlek açıları **np.arctan** kullanılarak hesaplanır ve açılarının tümü **np.degrees** kullanılarak dereceye dönüştürülür. Son olarak, **Matplotlib** kullanılarak bir çizim oluşturulur ve aracın dönüş dairesinin yarıçapının bir fonksiyonu olarak iç ve dış tekerlek açılarını gösterir.

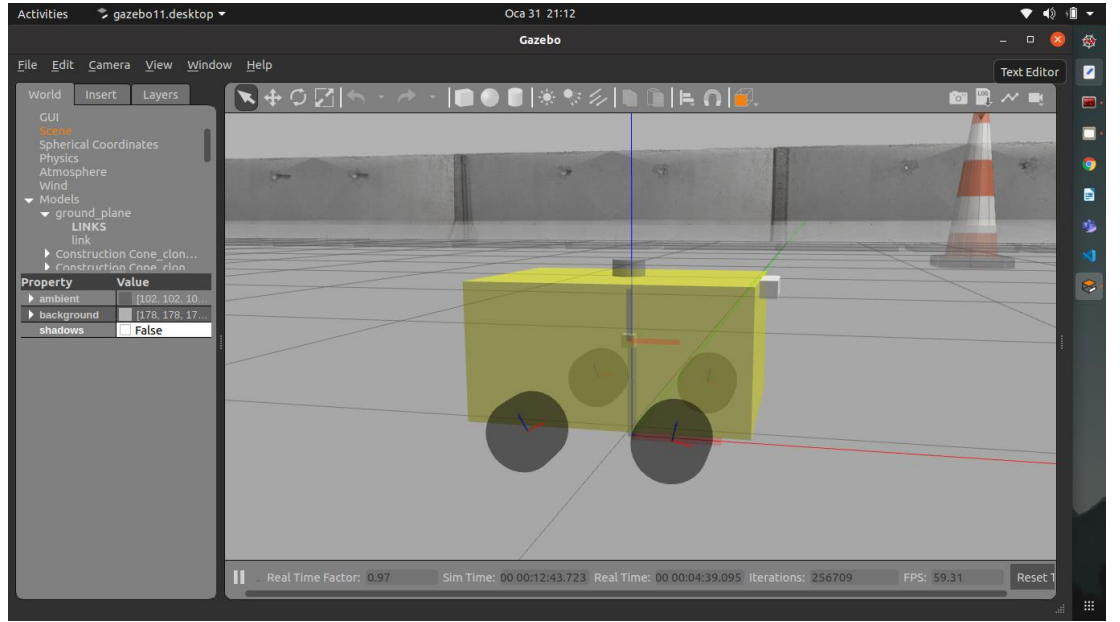
### 3.3 Gazebo ve RViz Ortamında Sensör ve Bileşenlerin Çıktıları



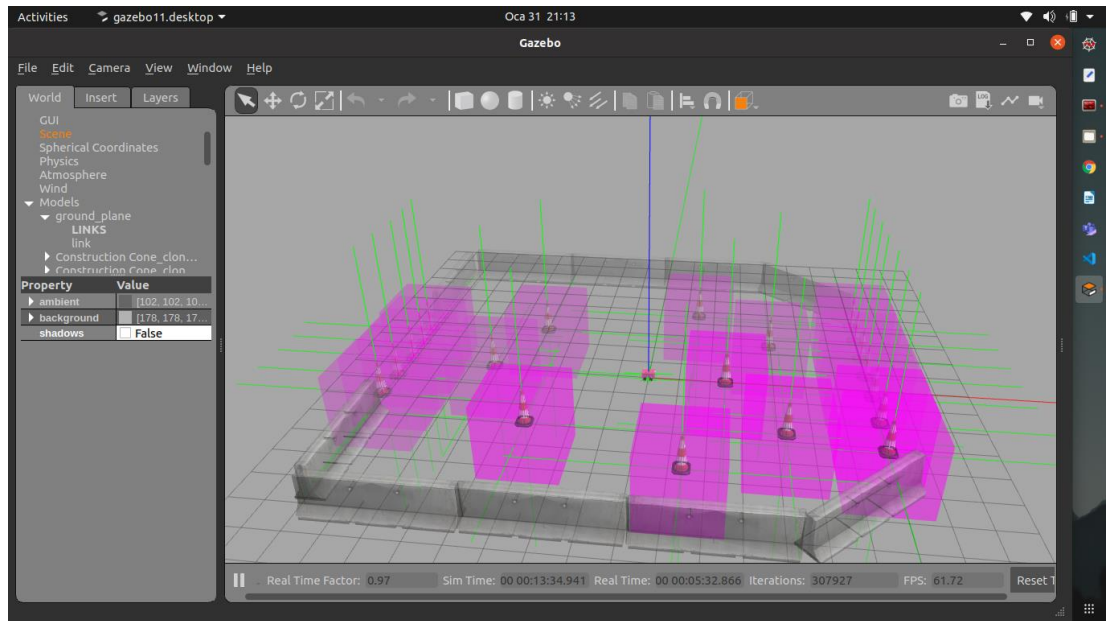
*Resim 9 Gazebo ortamında mobil robot*



*Resim 10 Gazebo test ortamı*



*Resim 11 Mobil robotun bağlantı ve koordinat gösterimi*

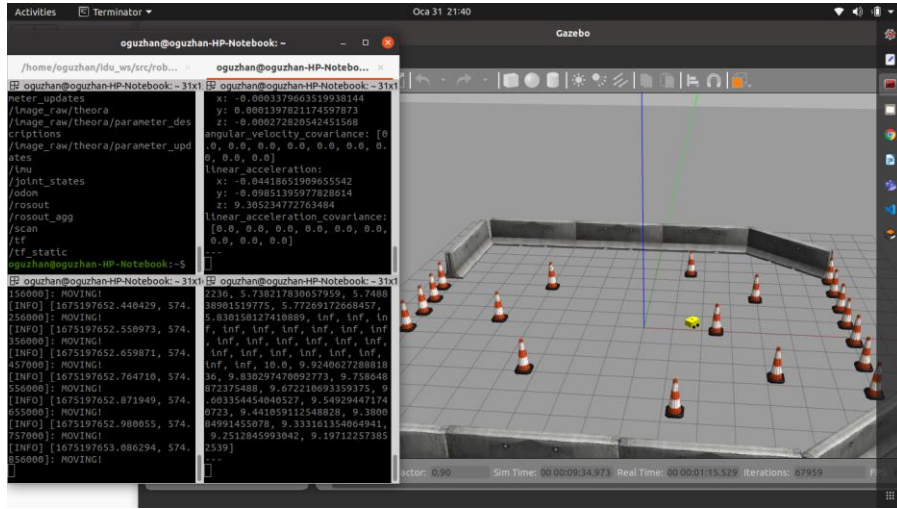


*Resim 12 Ortamın ve robotun eylemsizlik gösterimleri*

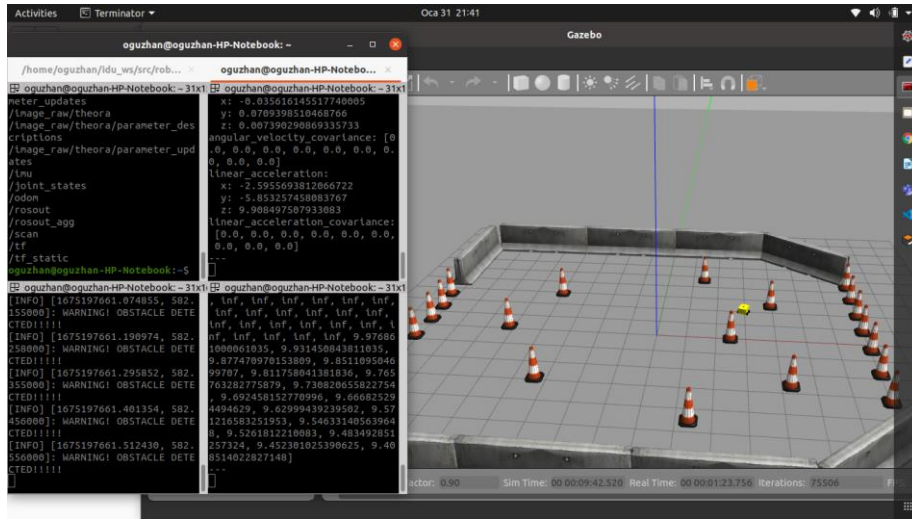
### 3.4 Gazebo ve RViz Ortamında Mobil Robotun Hareketi

Gazebo ve RViz, uygulamaya geçmeden önce robotik sistemlerin test edilmesi ve değerlendirilmesi için güvenli, verimli ve uygun maliyetli bir araç sağlayan, robot simülasyonu ve geliştirmesi için temel araçlardır. Bu sebeple testlerimizi bu ortamlarda gerçekleştiriyoruz.

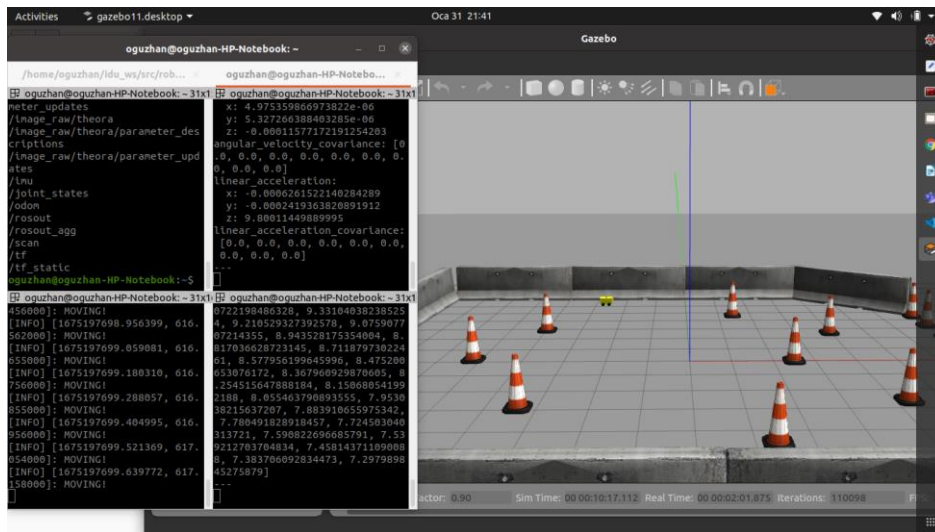




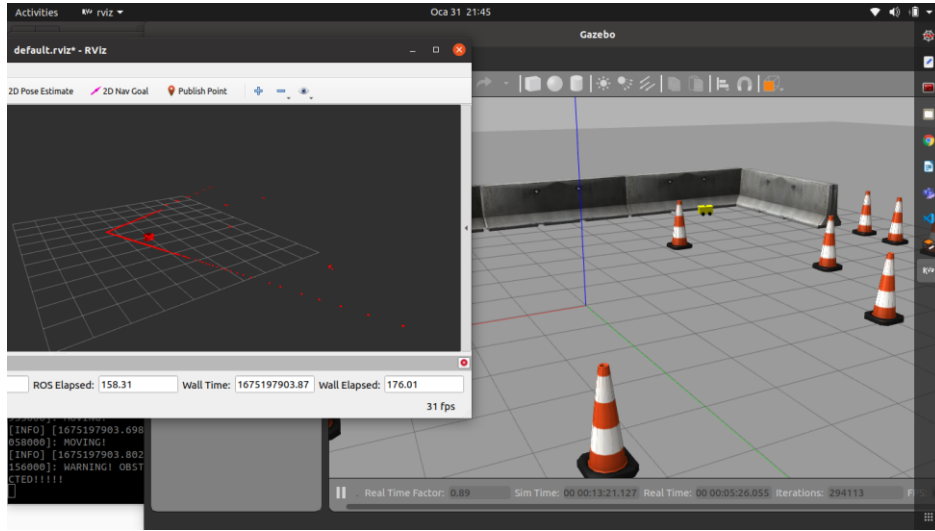
Resim 13 Robotun hareketi ve sensör verileri – 1



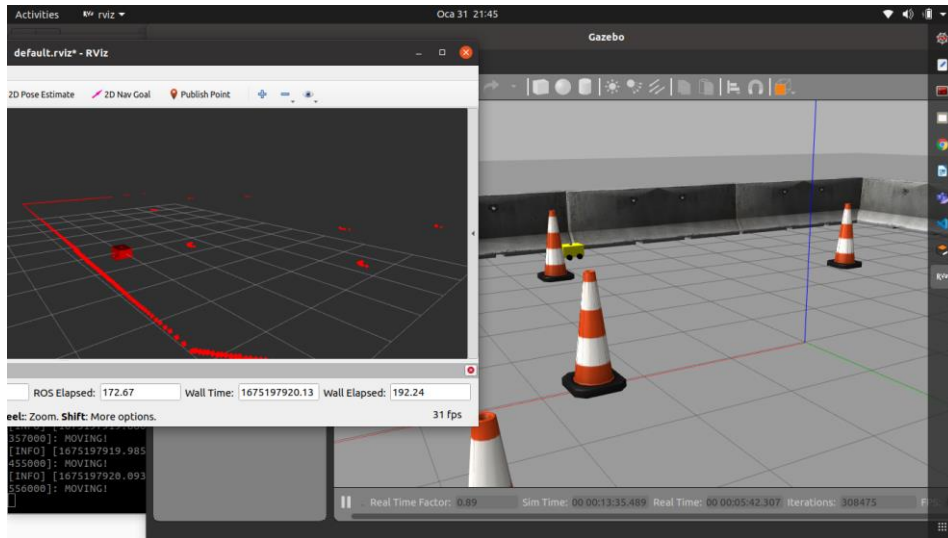
Resim 14 Robotun hareketi ve sensör verileri - 2



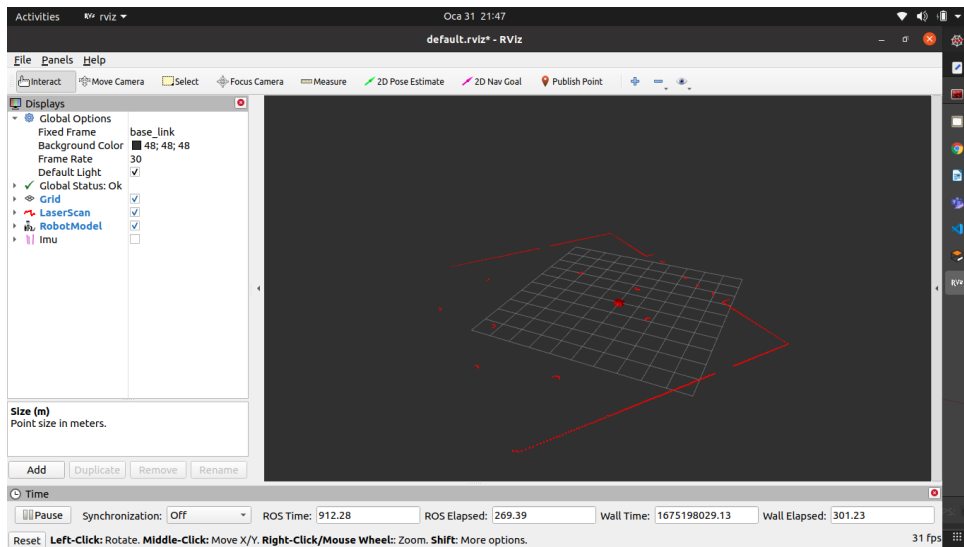
Resim 15 Robotun hareketi ve sensör verileri - 3



*Resim 16 Robotun hareketi ve sensör verileri – 4*



*Resim 17 Robotun hareketi ve sensör verileri - 5*



*Resim 18 Robotun hareketi ve sensör verileri - 6*

RViz ve Gazebo ortamında yapılan testler görsellerdeki gibidir ve robot hareketini ve uygulamalarını daha iyi anlamamızı sağlamaktadır. Bu görsellerde IMU ve LiDAR verilerinden gelen çıktıları görebilmekteyiz. Bu sensörler yardımıyla ROS platformunda robotun anlık konum verilerini alabiliyor ve engel saptama da mesafeleri ölçebiliyoruz. Ayrıca RViz ortamı bu verilerin görselleştirmesinde de rol oynamaktadır. Bu şekilde çalışmalarımızı daha verimli hale getirebilmekteyiz.

#### 4. SONUÇ VE ÖNERİLER

Sonuç olarak bu yazıda, ROS'a dayalı otonom bir mobil robotun geliştirilmesine ilişkin bir çalışma gerçekleştirdim. Robotun davranışını tasarlamak, simüle etmek ve görselleştirmek için Gazebo simülasyon ortamı ve RViz görselleştirme aracı kullanıldı. Robot, Ackermann Direksiyon kontrolü kullanılarak kontrol edilmesi amaçlandı. Ayrıca bu kontrol yöntemine PID eklenerek geliştirilmesi sağlanabilir.

Robotun kontrolü ve işletimi için birincil yazılım platformu olarak ROS'un kullanılması, robotun verimli ve etkili bir şekilde geliştirilmesini sağladı. Robotun performansı, robot davranışının gerçek zamanlı görselleştirilmesi için RViz'in entegrasyonu da kullanıldı.

Gelecekte, ek kontrol ve algılama algoritmalarının uygulanması, robotun performansını daha da iyileştirebilir. Bu teknolojinin tarım, inşaat ve lojistik gibi çeşitli endüstrilerde uygulanması muazzam bir potansiyele sahiptir ve çok sayıda avantaj sunmaktadır. Diğer mobil robot teknolojileriyle karşılaştırıldığında, bu çalışmada ROS kullanımı, açık kaynaklı bir platform, geniş bir geliştirici topluluğu ve kapsamlı bir kullanılabilir paket kitaplığı gibi çeşitli avantajlar sunar. Bu çalışmanın sonuçları, otonom mobil robotların geliştirilmesinde ROS kullanmanın potansiyeli ve fizibilitesine dair genel kanıtlar sunmaktadır.

## REFERANSLAR

- [1] SAEED, Raza A., et al. Conceptualization and Implementation of a Reconfigurable Unmanned Ground Vehicle for Emulated Agricultural Tasks. *Machines*, 2022, 10.9: 817.
- [2] C. Wang and D. Du, "Research on logistics autonomous mobile robot system," 2016 IEEE International Conference on Mechatronics and Automation, Harbin, China, 2016, pp. 275-280, doi: 10.1109/ICMA.2016.7558574.
- [3] FIORINI, Paolo; ALI, Khaled; SERAJI, Homayoun. Health care robotics: A progress report. In: *Proceedings of International Conference on Robotics and Automation*. IEEE, 1997. p. 1271-1276.
- [4] Pileun Kim, Jingdao Chen, Jitae Kim and Yong K. Cho, "SLAM-driven intelligent autonomous mobile robot navigation for construction applications", *Workshop of the European Group for Intelligent Computing in Engineering*, pp. 254-269, 2018.
- [5] ROS website: <https://www.ros.org> (Son ziyaret 28/01/2023).
- [6] Gazebo, <https://gazebo.org/home> (Son ziyaret 28/01/2023).
- [7] RViz, <http://wiki.ros.org/rviz> (Son ziyaret 28/01/2023).
- [8] RViz, <http://wiki.ros.org/rviz/UserGuide> (Son ziyaret 28/01/2023).
- [9] Nvidia Self-Driving, <https://www.nvidia.com/en-us/self-driving-cars/> (Son ziyaret 28/01/2023).
- [10] Yandex Rover, <https://sdg.yandex.com/deliveryrobot/> (Son ziyaret 28/01/2023).
- [11] Universal Robot UR5 robotik kol, <https://www.universal-robots.com/products/ur5-robot/> (Son ziyaret 28/01/2023).
- [12] Kuka Agilus, <https://www.kuka.com/en-gb/products/robotics-systems/industrial-robots/kr-agilus> (Son ziyaret 28/01/2023).
- [13] Amazon Scout teslimat robotu, <http://amazon.com/scout> (Son ziyaret 29/01/2023).
- [14] Raspberry Pi, <https://www.raspberrypi.com> (Son ziyaret 29/01/2023).
- [15] NAGLAK, John E.; PAGE, Brian R.; MAHMOUDIAN, Nina. Backseat control of sandshark auv using ros on raspberrypi. In: *OCEANS 2018 MTS/IEEE Charleston*. IEEE, 2018. p. 1-5.
- [16] S. Gatesichapakorn, J. Takamatsu and M. Ruchanurucks, "ROS based Autonomous Mobile Robot Navigation using 2D LiDAR and RGB-D Camera," 2019 First International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP), Bangkok, Thailand, 2019, pp. 151-154, doi: 10.1109/ICA-SYMP.2019.8645984.
- [17] Borenstein, J., Everett, H.R., Feng, L. and Wehe, D. (1997), Mobile robot positioning: Sensors and techniques. *J. Robotic Syst.*, 14: 231-249. [https://doi.org/10.1002/\(SICI\)1097-4563\(199704\)14:4<231::AID-ROB2>3.0.CO;2-R](https://doi.org/10.1002/(SICI)1097-4563(199704)14:4<231::AID-ROB2>3.0.CO;2-R)
- [18] YD Lidar 2X, <https://www.ydlidar.com/products/view/6.html> (Son ziyaret 29/01/2023).
- [19] Douadi, Lounis, Davide Spinello, and Wail Gueaieb. "Dynamics and Control of a Planar Multibody Mobile Robot for Confined Environment Inspection." *Journal of Computational and Nonlinear Dynamics* 10, no. 1 (2014): 011005. doi:10.1115/1.4027303.
- [20] QIU, Quan, et al. Extended Ackerman Steering Principle for the coordinated movement control of a four wheel drive agricultural mobile robot. *Computers and Electronics in Agriculture*, 2018, 152: 40-50.

- [21] MITCHELL, Wm C.; STANIFORTH, Allan; SCOTT, Ian. *Analysis of ackermann steering geometry*. SAE Technical Paper, 2006.
- [22] NASCIMENTO, Tiago P.; DÓREA, Carlos ET; GONÇALVES, Luiz Marcos G. Nonholonomic mobile robots' trajectory tracking model predictive control: a survey. *Robotica*, 2018, 36.5: 676-696.
- [23] TZAFESTAS, Spyros G. *Introduction to mobile robot control*. Elsevier, 2013.
- [24] CUCCI, Davide A.; MATTEUCCI, Matteo. Position tracking and sensors self-calibration in autonomous mobile robots by gauss-newton optimization. In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014. p. 1269-1275.