

**DOĐAL YER İMİ KULLANARAK GEZGİN ROBOT
KONUM KONTROLÜNÜN OTONOM OLARAK
GERÇEKLEŐTİRİLMESİ**

**2014
YÜKSEK LİSANS TEZİ
ELEKTRONİK VE BİLGİSAYAR EĐİTİMİ**

Ahmet ÖZEREN

**DOĐAL YER İMİ KULLANARAK GEZGİN ROBOT KONUM
KONTROLÜNÜN OTONOM OLARAK GERÇEKLEŐTİRİLMESİ**

Ahmet ÖZEREN

**Karabük Üniversitesi
Fen Bilimleri Enstitüsü
Elektronik ve Bilgisayar Eğitimi Anabilim Dalında
Yüksek Lisans Tezi
Olarak Hazırlanmıştır**

**KARABÜK
Haziran 2014**

Ahmet ÖZEREN tarafından hazırlanan “DOĞAL YER İMİ KULLANARAK GEZGİN ROBOT KONUM KONTROLÜNÜN OTONOM OLARAK GERÇEKLEŞTİRİLMESİ” başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Yrd. Doç. Dr. İbrahim ÇAYIROĞLU

Tez Danışmanı, Elektronik ve Bilgisayar Eğitimi Anabilim Dalı



Bu çalışma, jürimiz tarafından oy birliği ile Elektronik ve Bilgisayar Eğitimi Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir. 11/06/2014

Ünvanı, Adı SOYADI (Kurumu)

Başkan : Doç. Dr. İsmail Hakkı TAYYAR (KBÜ)

Üye : Yrd. Doç. Dr. İbrahim ÇAYIROĞLU (KBÜ)

Üye : Yrd. Doç. Dr. Mustafa AKTAŞ (KBÜ)

İmzası



...../...../2014

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile, Yüksek Lisans derecesini onamıştır.

Prof. Dr. Mustafa BOZ

Fen Bilimleri Enstitüsü Müdürü



“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”

Ahmet ÖZEREN

ÖZET

Yüksek Lisans Tezi

DOĞAL YER İMİ KULLANARAK GEZGİN ROBOT KONUM KONTROLÜNÜN OTONOM OLARAK GERÇEKLEŞTİRİLMESİ

Ahmet ÖZEREN

Karabük Üniversitesi

Fen Bilimleri Enstitüsü

Elektronik ve Bilgisayar Eğitimi Anabilim Dalı

Tez Danışmanı:

Yrd. Doç. Dr. İbrahim ÇAYIROĞLU

Haziran 2014, 64 sayfa

Bu çalışmada bir bina içerisinde, taşıma işlerinde kullanılacak, otonom olarak hareket edebilen, verilen adrese en kısa yoldan giden, etrafındaki cisimleri algılayan, bir robot geliştirilmiştir. Robotun bulunduğu binanın haritasını çıkarması, konumunu algılayabilmesi için lazermetre, jiroskop, ivme ölçer, manyetik sensör ve enkoder kullanılmıştır.

Sensörlerden okunan bilgileri alıp haritalamayı yapan, konumu hesaplayan, en kısa yolu bulan, hareket bilgilerini oluşturmak için geliştirilen yazılım sabit bir bilgisayar da tutulmaktadır. Robot ile bilgisayarın haberleşmesi bluetooth bağlantısı ile sağlanmıştır.

Robotun alıřtırılacađı ortamda, ilk olarak evrenin ğretilmesi gerekmektedir. Bu amala 360 derece dnebilen lazer metre ve jiroskop kullanılarak, ok sayıda noktadan tarama yapılıp, ortamın haritası ıkarılır. Tarama esnasında lazer metre ile belli bir ykseklikten, duvarlar ve sabit cisimlerin mesafeleri, lm noktasına bađlı olarak veritabanına kaydedilir.

Ortamın ğretilmesinden sonra grev alanı ierisinde robot alıřtırıldıđında, etrafındaki cisimlerin uzaklıklarını tesbit ederek ortamın haritasını oluřturur ve hafızasındaki haritalarla karřılařtırarak konumunu hesaplar. Bařlangı konumunu hesaplayan robot, verilen hedef noktaya giderken srekli olarak etrafını tarar ve tekerlerden aldıđı encoder bilgilerini de kullanarak hedef noktaya gider. Hedefe giderken hafızasındaki lm noktalarını kullanarak Karınca Koloni Algoritması ile en kısa yolu hesaplayarak gitmeye alıřır.

Anahtar Szckler : Gezgin robot, otonom kontrol, lazer ile alan tarama, DCM algoritması.

Bilim Kodu : 715.1.096

ABSTRACT

M. Sc. Thesis

MOBILE ROBOT POSITION CONTROL IMPLEMENTATION AUTONOMOUSLY BY USING NATURAL BOOKMARK

Ahmet ÖZEREN

Karabük University

Graduate School of Natural and Applied Sciences

Department of Electronic and Computer Education

Thesis Advisor:

Assist. Prof. Dr. İbrahim ÇAYIROĞLU

June 2014, 64 pages

In this study, a robot which can be used for transportation in a building, which can move autonomously and goes to the destination using the shortest path, which perceives the circumjacent objects, has been developed. In order for the robot to map the building and to perceive its position; lazermeter, gyroscope and endocer have been used.

The software; created to compose the activity information that makes the mapping through the information obtained by the sensors, calculates the locational status, finds the shortest path; is kept in an immobile computer. The communication between the robot and the computer is provided by bluetooth connection.

Firstly the environmental information of the area should be introduced where the robot will be active. For this reason, a map of the environment is obtained by using

360 degrees rotatable lazermeter and gyroscope and by making multiple spot scanning. During the scanning, the distances of the walls and stable objects; depending on the measuring point; are saved in the database from a certain height. After introducing the environment, when the robot is activated in the mission area, it makes the map of the environment by determining the distances of the objects and calculates its own position by comparing this map with the others in its memory. Once calculating the starting position, the robot continuously scans the environment and together with the encoder information gathered from the wheels, it reaches the destination point. While going to the destination point, it uses the measuring points in its memory and the Ant Colony Algorithm, so that it finds the shortest way.

Key Words : Mobile robot, autonomous control, lazer scanning area, DCM algorithm.

Science Code : 715.1.096

TEŐEKKÜR

Bu tez alıőmasının planlanmasında, araőtırılmasında, yürütölmesinde ve oluşumunda ilgi ve desteęini esirgemeyen, engin bilgi ve tecrübelerinden yararlandıęım, yönlendirme ve bilgilendirmeleriyle alıőmamı bilimsel temeller ışığında őekillendiren sayın hocam Yrd.Do.Dr. İbrahim AYIROęLU'na sonsuz teőekkürlerimi sunarım.

Yardımlarından ve emeklerinden dolayı İsmail Umut DURAN, Ufuk KIZILKAYA'ya ve Mustafa TÜKER' e teőekkür ederim.

Sevgili aileme manevi hiçbir yardımcı esirgemedен yanımda oldukları için tüm kalbimle teőekkür ederim.

İÇİNDEKİLER

	<u>Sayfa</u>
KABUL.....	ii
ÖZET	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
ŞEKİLLER DİZİNİ.....	xii
ÇİZELGELER DİZİNİ	xiv
SİMGELER VE KISALTMALAR DİZİNİ.....	xv
BÖLÜM 1	1
GİRİŞ	1
BÖLÜM 2	6
OTONOM KONTROLLÜ GEZGİN ROBOTLAR HAKKINDA GENEL BİLGİLER	6
2.1. OTONOM KONTROL DURUMLARI	6
2.2. SEYRÜSEFER İŞLEMİ.....	6
2.3. LOKALİZASYON VE SEYRÜSEFER TİPLERİ.....	7
2.4. KONTROL TİPLERİ	9
2.5. HARİTALAMA TEKNİKLERİ	11
2.5.1. Izgara Tabanlı Haritalama	11
2.5.2. Öznitelik Tabanlı Haritalama	11
2.5.2.1. Yapay Yer Göstericiler	12
2.5.2.2. Doğal Yer Göstericiler	12
BÖLÜM 3	13
ROBOTUN TASARIMI	13
3.1. ROBOTUN MEKANİK TASARIMI	13
3.1.1. Gövde.....	13
3.1.2. Motorlar	14

	<u>Sayfa</u>
3.1.3. Motor Sürücüleri.....	16
3.1.4. Kuru Bakımsız Akü	17
3.1.5. Sensörler	17
3.1.5.1. İvmeölçer	18
3.1.5.2. Jiroskop	19
3.1.5.3. Manyetik Sensor	20
3.1.6. Bluetooth Bağlantı Modülü	21
3.1.7. Lazer Mesafe Sensörü	22
3.1.8. Kontrol Kartı.....	
3.2. ROBOTUN KONTROL TASARIMI	26
3.2.1. DCM (Direction Cosine Matris) Algoritması ile Açık Tespiti.....	27
3.2.1.1. Rotasyon Matrisi	28
3.2.1.2. Jiroskop Değerleri.....	31
3.2.1.3. Normalizasyon	32
3.2.1.4. Yunuslama ve Yalpalama Düzeltmesi	33
3.2.1.5. Sapma Açısının Düzeltilmesi.....	35
3.3. ROBOT ve BİLGİSAYAR YAZILIMININ TASARIMI	36
3.3.1. Bölgeyi Tanıma	36
3.3.2. Robot Lokalizasyonu	38
3.3.3. Rota Belirleme	41
3.4. YAZILIMLAR	44
3.4.1. Mikro Denetleyici Yazılımı	44
3.4.2. Bilgisayar Sunucu Programı	45
3.4.3. Benzetim Programı	46
BÖLÜM 4	48
DENEYSEL ÇALIŞMALAR	48
4.1. LAZER METRE İLE ÖLÇÜM VE HARİTA OLUŞTURMA DENEYLERİ... 48	48
4.1.1. Lazer Metre ile Yapılan Kalibrasyon ve Ölçümler.....	48
4.1.2. Euler Açık (Pusula) Ölçüm Değerleri	49
4.1.3. Robotun Oluşturduğu Harita ile Gerçek Alanın Kıyaslanması	50
4.2. SEYRÜSEFER DENEYLERİ	52

	<u>Sayfa</u>
4.2.1. Lokalizasyon Deneyleri.....	52
4.2.2. Rota Takip Deneyleri.....	53
BÖLÜM 5	56
SONUÇLAR	56
KAYNAKLAR	59
ÖZGEÇMİŞ	64

ŞEKİLLER DİZİNİ

	<u>Sayfa</u>
Şekil 2.1. Izgara tabanlı haritalama.	11
Şekil 3.1. Gövdenin genel görünümü.	13
Şekil 3.2. Arka teker diferansiyel seti.	14
Şekil 3.3. Redüktörlü DC motor.	14
Şekil 3.4. Standart metal dişli servo	15
Şekil 3.5. DC motor sürücü entegrasi.	15
Şekil 3.6. Atmega 328 denetleyicisi ve Arduino Duemilanove denetleyici kartı. ...	16
Şekil 3.7. Kuru Bakımsız Akü.	17
Şekil 3.8. İvme ölçerin çalışması.	18
Şekil 3.9. Jiroskop'un iç yapısı.	19
Şekil 3.10. Hareket anında jiroskopun çalışması.	19
Şekil 3.11. Yeryüzü manyetik alan çizgileri.	20
Şekil 3.12. 3-eksendeki manyetik alan vektörleri.	21
Şekil 3.13. HC06 Bluetooth Modül.	22
Şekil 3.14. Efactor200 01d105 Model Lazer Mesafe sensörü.	23
Şekil 3.15. Ana kontrol kartı ve devre şeması.	25
Şekil 3.16. Robotun temel çalışma şeması.	26
Şekil 3.17. DCM algoritmasının çalışma şeması.	28
Şekil 3.18. Robot üzerindeki eksenler ve dönüş açıları.	29
Şekil 3.19. Robot yazılımının tasarımı.	36
Şekil 3.20. Örnek bir tarama alanı.	37
Şekil 3.21. Robotun ölçüm yaptığı düğüm noktaları ve oluşturulan harita.	37
Şekil 3.22. Robotun lokalizasyon için odayı taraması	38
Şekil 3.23. Robotun hafızasında oluşan tarama şekli.	38
Şekil 3.24. Tarama noktalarını eşleştirilmesi.	39
Şekil 3.25. Robot lokalizasyonu akış diyagramı	40
Şekil 3.26. Doğadaki karıncaların en kısa yolu bulmadaki davranışları.	41
Şekil 3.27. Örnek bir binada robotun gezeceği düğüm noktaları.	43

Sayfa

Şekil 3.28.Örnek bir binada robotun gezinebileceği yollar.	43
Şekil 3.29.Örnek bir binada robotun KKA kullanarak bulduğu kısa yol.	44
Şekil 3.30.Örnek bir odayı tanıma anında kullanılan 6 taramanın bileşke haritası. ..	45
Şekil 3.31.Öğrenme ve gezinti aşamasında elde edilen tarama noktaları örneği.	46
Şekil 3.32.Benzetim programında örnek bir tarama uygulaması.....	47
Şekil 4.1. Pusula Uygulaması ve Robotun algıladığı yönün karşılaştırılması.	49
Şekil 4.2. Deney ortamının farklı noktalardan taranması ile elde edilen haritalar ...	50
Şekil 4.3. Deney ortamının tarama sayılarına göre karşılaştırılması.	52
Şekil 4.4. Robotun B noktasından A noktasına geldiğinde bulduğu konum.	53
Şekil 4.5. Robotun iki nokta arasındaki seyrüsefer işlemi.....	54
Şekil 4.6. Robotun belirlenen rota üzerindeki seyrüsefer işlemi..	54
Şekil 4.7. Simülasyon ortamında robotun hedefe yapmış olduğu seyrüsefer işlemi.	55

ÇİZELGELER DİZİNİ

	<u>Sayfa</u>
Çizelge 3.1. Arduino Duemilanove denetleyici kartı özellikleri.....	16
Çizelge 3.2. Efecto200 01d105 model lazer mesafe sensörü özellikleri	23
Çizelge 4.1. Lazermetre ölçüm değerleri	48

SİMGELER VE KISALTMALAR DİZİNİ

SİMGELER

cm	: Santimetre
mm	: Milimetre
ms	: Milisaniye
V	: Volt
Ω	: Ohm
K Ω	: Kiloohm
nF	: Nanofarad
A	: Amper
mA	: Miliamper
μ A	: Mikroamper
kHz	: Kilohertz
MHz	: Megahertz
KB	: Kilobyte
\$: Dolar
x,y	: Robotun koordinatları
x_1, y_1	: Robotun ilk yaptığı hesaplama sonucu bulunan koordinatları.
x_2, y_2	: Robotun ikinci yaptığı hesaplama sonucu bulunan koordinatları.
x_3, y_3	: Robotun üçüncü yaptığı hesaplama sonucu bulunan koordinatları.
v_r	: Robotun doğrusal hızı.
v_1, v_2, v_3	: Robotun doğrusal vektörleri.
α	: x eksenine ile doğrultu vektörü arasındaki açı.
β	: Yer iminin potansiyometreden okunan açısı.
γ	: x eksenine ile yer imi doğrultusu arasındaki açı.
θ	: Robotun doğrultu vektörü ile yer imi doğrultusu arasındaki açı.
ε	: Yer imi doğrultusu ile potansiyometrenin son açısı arasındaki kalan açı
C	: Batarya şarj/deşarj birimi

mA	: Miliamper
X,Y,Z	: Yeryüzü referans eksenleri
ω_p	: Oransal dönme vektörü
ω_i	: İntegral dönme vektörü
$\omega_x, \omega_y, \omega_z$: 3 eksendeki dönme vektörü
ω	: Açısal hız
A_{xyz}	: Bileşke ivme vektörü
A_e	: İvme ölçer etkisi
t	: Zaman
X_1, Y_1	: Robotun ilk koordinatları
X_2, Y_2	: Robotun hedef koordinatları
DCM	: Direction Cosine Matris

KISALTMALAR

- CCS : Custom Computer Service (Özel Bilgisayar Servisi)
- CNC : Computer Numerical Control (Bilgisayarlı Nümerik Kontrol)
- CPU : Central Processing Unit (Merkezi işlem Birimi)
- DC : Direct Current (Doğru Akım)
- PWM : Pulse Width Modulation (Darbe Genişlik Modülasyonu)
- RPM : Round Per Minutes (Dakikadaki Dönüş Sayısı)
- RFID : Radio Frequency Identification (Radyo Frekansı ile Tanımlama)
- RISC : Reduced Instruction Set Computer (Azaltılmış Komut Seti Bilgisayarı)
- SGK : Sosyal Güvenlik Kurumu
- SRI : Stanford Research Institute (Stanford Araştırma Enstitüsü)
- TTL : Transistor-Transistor Logic (Mikrodenetleyiciler Arası İletişim Türü)
- TL : Türk Lirası
- UART : Universal Asynchronous Receiver Transmitter (Evrensel Asenkron Alıcı Verici)
- UAV : Unmanned Aerial Vehicles (İnsansız Hava Araçları)
- UDP : User Datagram Protocol (Kullanıcı Veri Bloğu İletişim Kuralları)
- UPS : Uninterruptible Power Supply (Kesintisiz Güç Kaynağı)
- USB : Universal Serial Bus (Evrensel Seri Veriyolu)

BÖLÜM 1

GİRİŞ

Robotlar günlük yaşamımızda her alana girmeye başlamıştır. Robotlardan beklenen esas özellik verilen görevleri kendi kendine otomatik olarak yapmaları ve farklı durumlarda doğru kararları verip görevini tamamlamalıdır. Bu yönde bina içlerinde ve fabrika ortamlarında taşıma görevi yapan robotlar, iş hayatına ve ekonomiye katkısı ile ayrı bir öneme sahiptir. Halen otonom olarak taşıma işlemi gören robotlar daha çok çizgi izleyen ve yapay yer imleri kullanan robotlardır. Bu çalışmada bina iç ortamı fazla karmaşık olmayan, duvar ve sabit nesnelerin fazlaca bulunduğu ortamlarda kullanılacak bir otonom robot geliştirilmiştir.

Otonom mobil robotlarla ilgili çok sayıda konu ve problem açıklanmıştır. Başarılı bir otonom navigasyon için konular dört başlık altında toplanabilir [1].

- a) Ortamı algılama
- b) Ortamı temsil etme
- c) Ortam içinde konumunu bulma
- d) Ortam içinde rotayı planlama ve takip

Konum bulma üzerine oldukça yoğunlaşmış ve bu problemi çözmek için çok sayıda teknik önerilmiştir. Genel olarak gerçek dünyada mobil robotun konumunu hesaplama işlemi iki kısımda ele alınır. Bunlar relatif konumlama ve mutlak konumlamadır.

Relatif konum hesaplamada odometri (dead reckoning) ve ataletle yön bulma yaygın olarak kullanılan yöntemdir [2]. Belli bir referans noktadan hareketle konumlar artışı olarak hesaplanır [3]. Tekerlerin kaymasından ve hataların artışı olarak eklenmesinden dolayı pozisyon hataları hızla artmaktadır. Bu nedenle diğer

sensörleride (kamera ve tarama sensörleri gibi) kullanarak ile sık sık düzeltme yapmak gerekir [4].

Mutlak konumlamada ise ortamın belli özellikleri kullanılır. Bu özellikler genel olarak iki kısma ayrılır; Birincisi ortama yerleştirilen aktif işaretler (active beacons) [5], doğal [6,7] ve yapay yer imleri [8] ile mutlak konumlama, ikincisi ise farklı geometrik modeller [9,10] kullanarak harita tabanlı yöntemlerdir. Doğal ve yapay yer imlerini kullanan yöntemler daha çok görsel (vision) tabanlı yöntemlerdir. Bu yöntemlerde belli özellikler arka plan resminden çıkarılır.

Doğal yer imleri kullanan yöntemler hem iç mekanlar için hem de dış mekanlar için uygundur. Ortamdaki cisimler ve özellikler (koridor, zemin, kapı, pencere vs) zaten orada bulunmaktadır ve bunun için ekstra bir çabaya ihtiyaç yoktur [11]. Bu cisimlerin geometrik yada foto metrik özellikleri düşünülerek seçilirler. Mobil robotlar onları ortamdaki ayıran bu özelliklerini analiz eder ve konumunu ona göre hesaplar [12,13]. Ortamdaki cisimlerin özellikleri daha önceden mobil robota öğretilir. Hareket esnasında yapay sinir ağları yada bazı eşleştirme teknikleri ile bu özellikler tanınır [14-16]. Bu yöntem dinamik ortama karşı çok hassastır ve resim işleme teknikleri fazlaca zaman almaktadır. Bu yüzden gerçek zamanlı olarak konum bulma işleminde doğal yer imlerinin kullanılması zordur [17,18]. Doğal yer imlerinin kullanılması, yapay yer imlerine göre daha genel yaklaşımlardır [18].

Her ne kadar yapay yer imleri kullanan metotlar genel olmasa da ve çevre üzerinde bazı düzenlemelere ihtiyaç duysa da, karmaşık ortamlarda çok doğru ve güçlü performans verirler [19]. Kullanılan yer imleri belli şekillerde ve renklerde tasarlanırlar. Yer iminin bulunması ve ortamın haritalanması kolaydır [20]. Yer iminin resim üzerinde tespiti kolaydır fakat ortama çok sayıda yer imini koymak gerekir [21].

Robotun ortam içerisinde kendisini konumlaması için birçok teknik kullanılır. Bu konuda yaygın metotlar triangulation (üçgenleme) [22-24], trilateration (üçkenarlama) [25-28] ve bu ikisinin kombinasyonudur. Ayrıca bu metotlarla görsel sistemlerin birleştirilmesidir [28]. Üçgenleme metodunda yer imine yada hedef

noktaya olan açılar kullanılır. Üçkenarlama metodunda ise mesafeler kullanılır. Ölçümler için infrared (kırmızı ışık), ultrasound ve radyo frekansı yayın işaretçiler kullanılır. Ayrıca geometrik işaretçilerde (odanın köşesi, kenarı, kapı, pencere gibi) kullanılır.

Başarılı bir navigasyon için, mobil robot engelleri algılama ve kaçma, rota planlama, haritalama, konumunu hesaplama gibi işlemleri güçlü bir şekilde gerçekleştirebilmelidir. Bunlar arasında konum hesaplama en zor konudur [18].

Konum bulma konusundaki bir çok teknik temel olarak üç kategoride toplanabilir. Geometrik yaklaşımlar, bulunduğu ortamın görüntüsünü kullanır. Bu yöntemler çalışılan ortamın verilen geometrik modeli ile, elde edilen yerel geometriyi karşılaştırır. Fakat karmaşık ortamlarda sensörler vasıtasıyla yerel geometrinin çıkarılması zor bir konudur. Topolojik yaklaşımlar, konum değişimini göstermek için birleşik grafikleri kullanır. Ortam değiştiğinde bu yöntemde birçok problemle karşılaşmaktadır. Yer imi temelli yöntemler ise daha basit ve güçlü konum hesaplamayı sağlamaktadır [18,23,24,29,30].

Lazer tarama ile robotun bulunduğu çevre taranarak haritası çıkarılmaktadır. Ortamın taranması 2 boyutlu ve 3 boyutlu olarak yapılabilmektedir. 2 boyutlu taramada çevredeki nesnelerin konumu haritalanmaktadır [31-35]. 3 boyutlu ortamın haritasını çıkarmak için 2 boyutlu tarama yapan birden fazla lazer tarama kullanılmaktadır [35-39]. Lazerlerden bir tanesi yatay taramayı yaparken, diğeri dikey taramayı gerçekleştirmektedir. Bu çalışmada 2 boyutlu tarama kullanılmıştır.

Lazer sensorlerin çalışma sistemi temel olarak iki kategoride toplanabilir. Uçuş zamanı ölçümü (time of flight) [40-44] ve Üçgenleme ölçümü (triangulation) [45,46] metodlarıdır. Uçuş zamanı ölçümüne göre çalışan lazer tarayıcılar uzun mesafelerde daha hassas çalışmaktadırlar fakat pahalı ve büyüktürler. Üçgenleme lazer tarayıcılar ise küçük ve ucuzdurlar. Uçuş zamanı lazerleri bir zamanda bir noktayı tarayabilir. Bu nedenle alanı tarama zaman alır. Üçgenleme lazerler ise aynı anda çok sayıda noktayı tarar ve tarama hızı yüksektir. Bu çalışmada uçuş zamanı lazer kullanılmıştır.

Geliştirilen robot, çevreyi öğrenme aşamasında, etraftaki nesnelere belli bir yükseklikten 6 derece aralıkla 360 derece tarama yapar. Tarama esnasında okunan mesafe bilgileri ve jiroskoptan gelen pusula bilgileri işlenerek, tarama noktasına bağlı ortamın 2 boyutlu haritası oluşturulur. Tarama noktaları bina içerisinde rastgele çok noktadan yapılarak, binanın genel haritası oluşturulur. Görev esnasında konum ve hareket hesaplamalarını yapmak için, lazer metre ile mesafe ölçümleri, jireskop ile açı ölçümleri ve arka tekerlekteki enkoderlarla ilerleme ölçümleri yapılır. Gidilecek rotanın belirlenmesi, tarama noktaları üzerinden en kısa yol hesaplanarak yapılmaktadır.

Yapılan bu tez çalışması içerik olarak aşağıdaki maddeleri kapsamaktadır;

- a) Genel amaçlı gezgin bir robotun tasarımı ve otonom kontrolü gerçekleştirilmiştir.
- b) Robot başlangıçta tüm binayı kullanıcı tarafından dolaştırılarak haritalamayı kendi hafızasında kendisi oluşturmaktadır.
- c) Jiroskop ile robotun yönü, lazer metre ile ölçüm mesafesi, ölçüm yapılan yön ise servo motora gönderilen yön bilgisi ile belirlenmektedir.
- d) Kullanılan jiroskop, ivme ölçer ve manyetik sensor entegrasyonu ile ölçülen bilgiler aynı anda işlenerek, pusula bilgisinin sağlıklı olması sağlanmıştır. Burada DCM algoritması kullanılmıştır.
- e) Servo motora bağlı lazer metre ile 6 derece aralıkla 360 derecelik alan taramaktadır.
- f) Robotun konumu ve tarama değerleri birlikte değerlendirilerek ortamın haritası oluşturulur ve başlangıçta veri tabanında tutulan değerler ile karşılaştırılarak konum hesaplanır.
- g) Robot konumunu algıladıktan sonra gideceği noktaya göre en kısa rotayı belirleyerek hedefe yönelir. Buradaki rotalama işleminde karınca kolonisi algoritması kullanılmıştır.
- h) Arka tekerlerdeki sayıcılar ve lazer metreden gelen mesafe bilgisi kullanılarak hem ön tekerdeki servo motorun yönlendirilmesi yapılır, hem de robotun hareketleri sürekli takip altında tutulur. Robot konumu konusunda hataya

düşerse hareketi durdurarak alanı yeniden tarar ve konumunu bir daha bulmaya çalışır. Konum hesaplandıktan sonra rotasında devam eder.

- i) Robotun kontrolü ATMEL 378 işlemcili Arduino kontrol kartı ve bir dizüstü bilgisayar tarafından yapılmaktadır.
- j) Robotun üzerindeki işlemci ve sabit konumda olan bilgisayarla arasındaki iletişimi sağlamak için bluetooth kullanılmıştır.
- k) Bluetooth, jiroskop, ivme ölçer, manyetik sensor, encoder ve lazer metrenin robottaki ATMEL 378 işlemcili Arduino kontrol kartı ile bağlantısı için ise kart tasarlanmıştır.
- l) Gezgin Robot çalışma süresi içerisinde enerjisini batarya ile sağlamaktadır. İstenirse batarya bitmeye yakın zamanda, park alanına gidip otomatik şarj etme özelliği de eklenebilir.

BÖLÜM 2

OTONOM KONTROLLÜ GEZGİN ROBOTLAR HAKKINDA GENEL BİLGİLER

2.1. OTONOM KONTROL DURUMLARI

Robotlar otonom durumuna göre üçe ayrılabilir [47].

Otonom olmayan robotlar: Tüm kontrolü ile insan tarafından yapılan robotlardır. Bu robotların algılama ve zekâsı sadece kullanıcı tarafından verilen komutları uygulayacak kadardır.

Yarı otonom robotlar: Bu tür robotlar belirli durumlarda kararı kendisi verirken belirli durumlarda ise karar kullanıcı tarafından verilerek işlem yapan robotlardır.

Tam otonom robotlar: Tamamen kendi başına hareket eden zeki makinelerdir. Çevreyi kendi yetenekleri ile algılar ve tüm kararları kendisi verip harekete geçerler.

Bu çalışma kapsamında geliştirilen robot “Tam Otonom Robot” olarak tanımlanabilir.

2.2. SEYRÜSEFER İŞLEMİ

Robot seyrüseferi denince, gezgin otonom robotun bir yerden başka bir yere konum değiştirmesi düşünülür. Gezgin robotların doğaları gereği, endüstriyel sabit robotların aksine görevlerini tamamlayabilmek için çoğu zaman ortamda gezinmeleri gerekmektedir. Gezgin robotun bu yer değiştirme görevini yerine getirmesi için aşağıdaki soruların cevabını bilmesi gerekir [47].

- a) Ben neredeyim? Gezgin robot tam olarak konumunu bilmesi gerekir. Gezgin robotların konumunu bilme işlemine “robot lokalizasyonu” denir.
- b) Ben nereye gidiyorum? Ezgin robotun gideceği hedef noktasını bilmesi gerekmektedir. Literatürde “hedef tanımlama” denilmektedir.
- c) Nasıl giderim? Gezgin robot bulunduğu konumu ve gideceği hedefi biliyorsa, bu iki nokta arasına nasıl gideceğini bilmesi gerekir. Literatürde buna “yol belirleme” denilmektedir.

Bu tez çalışmasında, yukarıda bahsedilen üç soruya cevap verebilen bir robot geliştirilmiştir.

2.3. LOKALİZASYON VE SEYRÜSEFER TİPLERİ

Bu tip robotlarda kullanılan lokalizasyon (konum belirleme) ve seyrüsefer (gezinti) tipleri aşağıda olduğu gibi 8 başlık altında incelenebilir.

Lokalizasyon ve Seyrüsefer Tipleri:

- a) Kodlayıcı Yardımı İle: Bu yöntemde tekerlerin dönüş miktarı kodlayıcı yardımı ile alınarak kontrol kartına aktarılır. Bu yöntemde karşılaşılabilecek sıkıntının başında, sıralı hatalar oluşması ve tek bir tekerde kullanıldığında ise takılmadan ve dönüş anından kaynaklı hatalı bilgi üretmesidir. Bu hatanın düzeltilmesi için yedek mesafe algılayıcılar ve yeniden konum algılama gerekmektedir.
- b) Aktif Yol Gösterici Yardımı İle: Bu teknik kullanarak gezgin robotun tam olarak yeri belirlenebilir. Bu metotta en az üç adet verici yol göstericiler kullanmak gerekmektedir. Bu yol göstericilerden alınan sinyaller işlenerek konum bilgisi oluşturulmaktadır. Aktif yol göstericiler lazer, ışık, radyo frekansı veya ses yayıcılar olabilir.
- c) Doğal Yer İmi Yardımı İle: Gezgin robot lokalizasyonunda doğal yer imi olarak ortamın geometrik şekli ya da ortamda bulunan çiçek, kapı, yer

döşemesi, pencereler, tavadaki ışıklar, duvarların kesişim yerleri gibi nesnelere ve özellikleri kullanılabilir. Doğal yer imini kullanmanın en büyük avantajı dışarıdan bir yer imi yerleştirmeden mekânın özellikleri kullanarak gezgin robotun lokalizasyonu yapılabilmesidir. Bu tür sistemlerin dezavantajı ise mekânı belirleyici olan yer imi doğru belirlenmez ise lokalizasyondan sağlıklı sonuçlar alınmaya bilinir. Seçilecek parametreleri işleyerek konumu hesaplayacak olan bilgisayar fonksiyonu sisteme oldukça büyük hesaplama yükü getirebilir. Bu durumda robotun çalışması yavaşlayabilir ve istenmeyen sonuçlar ortaya çıkabilir.

d) Yapay Yer İmi Yardımı İle: Yapay yer imleri robot tasarımcısı tarafından belirlenerek yer imi parametreleri robotun hafızasına önceden yüklenir. Robot mekana yerleştirilen yer imlerini algılayarak lokalizasyon ve seyrü sefer işlemi yapar.

e) Harita Tabanlı Konumlandırma: Bu yöntemde robotun üzerindeki algılayıcılardan alınan veriler bir harita ile karşılaştırılarak gezgin robot konumlandırılmaya çalışılır. Alınan veriler ile tespit edilen konum, harita üzerindeki veriler ile örtüşüyor ise gezgin robotun konumu doğru demektir. Harita tabanlı konumlandırmada iki yaklaşım vardır. Bunlardan ilki daha önceden var olan bir haritanın kullanılması, bir diğer yöntem ise robotun kendi haritasını oluşturmasıdır. Burada haritadan kastedilen, konum bilgisi sensör bilgilerinin tutulduğu bir dosyadır. Oluşturulan haritalar geometrik ve topolojik olarak ikiye ayrılabilir. Geometrik harita, robotun bulunduğu dünyadaki yerleri gerçek (x,y,z) değerleri ile gösterirken, topolojik haritalar ise robotun bulunduğu dünyayı düğüm ve yaylar şeklinde gösterir.

f) Görüntü Temelli Lokalizasyon ve Hareket Etme: Gezgin robotlarda kamera ile alınan bilgilerin karşılaştırılması ile lokalizasyon ve seyrüsefer bilgisinin bulunması amaçlanan bir yöntemdir. Bu yöntem tek ve iki kameralı olarak kullanılabilir. Kameraların görüş açısı farkından insan gözünde olduğu gibi nesnenin robota olan mesafesi hesaplanmakta veya hareket öncesi ve sonrası farkla boyut değişiminden lokalizasyon yapılmaktadır.

g) Planlı Seyrüsefer: Planlı seyrüseferde robot algılayıcıları yardımıyla sürekli olarak çevreden bilgi toplar, alınan verileri değerlendirerek bir plan oluşturur ve bu oluşan plana göre hareket eder. Ortamda ya da robotun koşullarında, bir değişiklik olduğunda örneğin pozisyon bilgisinin değişmesi durumunda sistem yeniden tüm çevreyi algılamalı ve mevcut durum için yeni bir plan oluşturmalıdır. Bu planlamayı gerçek zamanda yapması, sisteme oldukça büyük bir iş yükü getireceği gözden kaçırılmamalıdır [48].

Bu tez çalışması kapsamında geliştirilen robot, yukarıda c, e, g maddelerinde bahsedilen konuları içermektedir. Buna göre robot, çevredeki doğal yer imlerini lazer ile tesbit eder, hafızasındaki harita üzerine bulduğu çevrenin haritasını yerleştir ve planlı bir seyrüsefer işlemi gerçekleştirir.

2.4. KONTROL TİPLERİ

Gezgin robot verilen görevi yerine getirebilmek için mutlaka planlama yapmak zorundadır. Literatürde planlama tiplerine bakıldığında; davranış tabanlı, genetik algoritma tabanlı, bulanık mantık tabanlı, neuro-fuzzy tabanlı kontrol algoritmaları görülmektedir.

Literatürde kullanılan kontrol tipleri şu şekilde özetlenebilir;

- a) Davranış Tabanlı Kontrol: Davranış tabanlı kontrol aslında bir çeşit tepkisel planlamadır. Bu işlemde sistem, davranışın tetikleyicisi olan ortamdan bir dizi çıktı hareketi üretir. Sistemin çıktısı olan hareketleri; engellerden kaçma, duvarı takip etme, nesneyi tutma ya da bırakma, ortamda arama gibi basit davranışlar olabilir. Ancak bu basit davranışlar bir araya getirilerek oldukça karmaşık davranışlar oluşturmak da mümkündür. Temelinde tepkisel bir yöntem olduğu için oldukça hızlı sonuçlar vermektedir.
- b) Genetik Algoritma Tabanlı Kontrol: Genetik algoritmalar adını Darwin'in evrim teorisinden alır. Bu teoriye göre popülasyonu oluşturan bireylerden sadece iyi olanlar bir diğer popülasyona aktarılır, bu aktarılma sırasında

popülasyonun amacına uygun olmayan bireyler elenir ve amaca uygun iyi bireyler bir sonraki nesle aktarılırlar. Bu aktarım sırasında geriye hep iyi kabul edilebilecek sonuçlar kalır. Bir süre sonra çözüm daha fazla iyileşmez ve bulunan en iyi değer çözüm olarak kabul edilir. Ancak bu yöntem iteratif bir yöntem olduğundan çözüm hem en iyi çözüm olmayabilir, hem de gerçek zamanlı uygulamalarda yavaş kalabilir. Bu yüzden literatürde genetik algoritmalar ve gezgin robotla ilgili çalışmalar daha çok rota planlama problemlerine çözüm getirmek için kullanılmıştır.

c) Bulanık Mantık Tabanlı Kontrol: Boolean mantığa göre, bir olgu var ya da yok olabilir. Örneğin bir renk ya siyahtır ya da beyaz. Arada olan tonları bir çözüm olarak kabul etmez. Bulanık mantıkta ise çözüm siyah ve beyaz arasındaki tonlar şeklindedir. Bulanık mantık çözümlerini, “Eğer-O Zaman” kural cümleciklerine göre çözer. Bulanık mantıkla daha önceki yapılan çalışmalarda kızıl ötesi algılayıcılar kullanarak hedef takip sistemi, gezgin robotun yönlenme açısının hesaplanması, seyrüsefer esnasında hız ve dönme açısı parametrelerine karar verilmesi gibi durumlarda kullanılmıştır. Kısaca gezgin robotlarda bulanık kontrol merkezi girdi olarak çeşitli algılayıcı bilgilerini alır ve çıktı olarak da söz edilen hız ve dönme açı değerlerini hesaplamada kullanılmıştır.

d) Neuro-Fuzzy Tabanlı Kontrol: Problem çözümlerinde, bulanık mantık ve yapay sinir ağı metotlarının birleşmesinden ortaya çıkmıştır. Bu melez yöntem her iki metodun da avantajlarına sahiptir. Örneğin; gezgin robotun park esnasında karar vermesi ve kinematik hesaplarının yapılması bu yöntemle yaptırılabilir.

Bu çalışmada genetik algoritma gibi nümerik olarak en optimum sonucu bulan Karınca Koloni algoritması ile rota planlaması yapılmıştır.

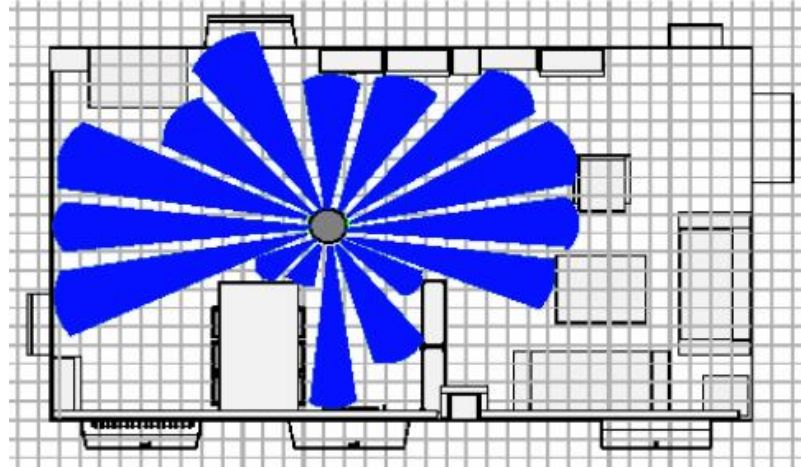
2.5. HARİTALAMA TEKNİKLERİ

Haritalar, robot konum belirleme problemlerinin çözümü için çok önemlidir. Robot uygulamalarında en genel harita gösterimi iki şekilde yapılır.

1. Izgara Tabanlı Harita
2. Öznitelik Tabanlı Harita

2.5.1. Izgara Tabanlı Haritalama

Izgara tabanlı haritalar ortamın belirli oranlarda parçalara bölünmesiyle olur. Izgara içerisindeki her bir hücre çevre hakkındaki bilgileri tutar. Bu bilgiler genellikle olasılıksal ya da hücrenin dolu, boş veya keşfedilmemiş olduğunu gösteren karakterlerden oluşur. Izgara tabanlı haritanın oluşturulması ve güncellenmesi, robotun çevreyi gezinmesi sırasında sensörlerden topladığı ölçümlere göre yapılır [50].



Şekil 2.1. Izgara tabanlı haritalama.

2.5.2. Öznitelik Tabanlı Haritalama

Öznitelik tabanlı haritalar ortam içindeki nesnelerin şekli hakkında fikir veren ve bu nesnelerin konumlarına göre ortamın betimlendiği haritalardır. Bu öznitelikler konum belirleme problemlerinde birer yer gösterici (işaretçi) olarak kullanılır. Robot

bu yer göstericilere bakarak konumunu belirler. Yer göstericiler yapay veya doğal yer göstericiler olmak üzere ikiye ayrılır [50].

2.5.2.1. Yapay Yer Göstericiler

Yer gösterici yerini belirtmek için kendisi ortama işaret gönderiyorsa aktif, göndermiyorsa pasif yer gösterici olarak adlandırılır [51]. Aktif yer göstericiler robota pozisyon bilgisi gönderen işaretçilerdir. Açık alan uygulamalarında en çok bilinen ve başarılı aktif işaretçiler GPS'lerdir. Bu sistem, uydudan gönderilen radyo işaretlerinin uçuş süresi prensibine dayanır. Ancak kapalı alan uygulamalarında kısıtlı olarak kullanılır. Pasif yer göstericiler robot tarafından algılanırlar. Robotun bulunduğu her ortama yerleştirilmesi gerekir. Robot algılayıcıları tarafından algılanabilecek her hangi bir nesne olabilir. Robot konumunu ortama sonradan ilave edilmiş bu pasif yer göstericilere bakarak belirler.

2.5.2.2. Doğal Yer Göstericiler

Robotun gezindiği ortam içinde kendiliğinden var olan nesnelere dir. Robot bu nesnelere algılayarak, sınıflayarak ve konumlarını tespit ederek haritalama yapar. Kapalı alanlar için kapı dikmesi, duvarlar, masa ayakları, kenar ve köşe noktaları doğal yer göstericilere örnek olarak verilebilir. Ortamın iki boyutlu olarak tanımlandığı durumlarda, yer gösterici tabanlı haritalar genel olarak düz çizgi ve noktalardan oluşan bir yapıdadır. Düz çizgiler, ortamdaki düzlemsel bölgeleri (duvarlar, mobilya yüzeyleri); noktalar ise ortamdaki kenar ve köşeleri (masa ve koltuk ayakları, odanın kenar veya köşe kısımlarını) temsil eder [51].

Bu çalışmada çevredeki nesnelere nelerin şekil özellikleri haritalanarak Öznitelik tabanlı bir haritalama kullanılmıştır. Dolayısıyla doğal yer göstericiler haritalamada kullanılmış olmaktadır.

BÖLÜM 3

ROBOTUN TASARIMI

3.1. ROBOTUN MEKANİK TASARIMI

3.1.1. Gövde

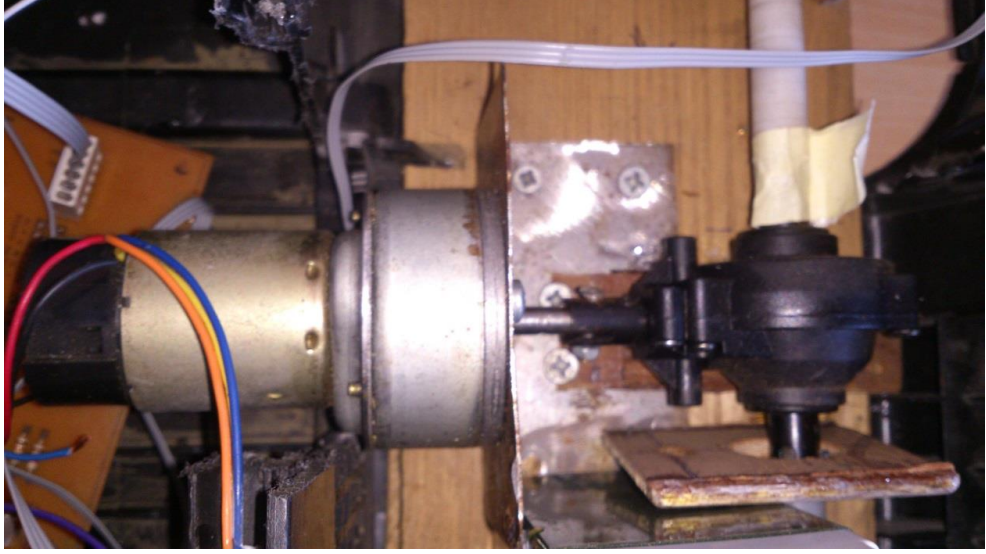
Robot gövdesi için elektrik süpürgesi olarak üretilmiş bir süpürge gövdesi kullanılmıştır. Gövdenin uzunluğu 60 cm, eni 30 cm boyutlarında, tekerlek çapı ise 15 cm uzunluğundadır. Ön teker tek teker olup, yönlendirme bu tekerden servo motor kullanılarak yapılmakta ve teker 120 derece dönebilmektedir (Şekil 3.1).



Şekil 3.1. Gövdenin genel görünümü.

Arka tekerleklerin hız tayini DC motor ile yapılmaktadır. Çalışmada her iki tekerlek için ayrı DC motor kullanmak yerine tek DC motor kullanılmıştır. İki tekerlek diferansiyel ile DC motora bağlanmıştır. Bu sayede dönüşlerde karşılaşılan arka

tekerlekler arasındaki devir probleminin kaynaklı hareket bozukluğu önlenmiş olmaktadır (Şekil 3.2).



Şekil 3.2. Arka teker diferansiyel seti.

3.1.2. Motorlar

Otonom Robotun hareketi arka tekerlerdeki diferansiyel setine bağlı redüktörlü DC motor aracılığı ile sağlamaktadır. Bu amaçla kullanılacak çeşitli motor türleri mevcuttur. Bu çalışmada düşük devirli (93 d/d) 12voltluk DC (Direct Current) motor kullanılmıştır (Şekil 3.3).



Şekil 3.3. Redüktörlü DC motor.

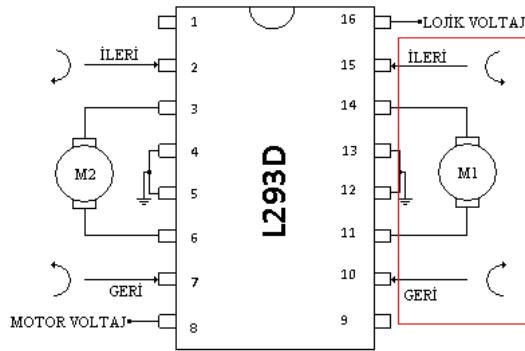
Ön teker ve kameraların kontrolünde standart metal dişli servo motor kullanılmıştır (Şekil 3.4). Servo motor kullanmanın sebebi, gönderilen PWM sinyaline göre istenilen açıya konumlanması bir önceki durumdan etkilenmeksizin olmasıdır. Çalışmanın başlangıç denemelerinde servo motor yerine step motor kullanılmıştır. Step motorların adım kontrolü takip ediliren, kontrol kartının programının tamponunda biriken hatalı bilgilerden kaynaklı sorun yaşanmıştır. Bunun yanında, lazer metre dönüş anında her hangi bir takılma yaşadıysa, örneğin 5 derece daha az döndü ise, bu hata daha sonraki tüm işlemlere yansımaktaydı. Bu nedenle çalışmada açısız dönüşlerin yapıldığı lazer metre ve ön tekerleğin kontrolünde servo motor tercih edilmiştir.



Şekil 3.4. Standart metal dişli servo

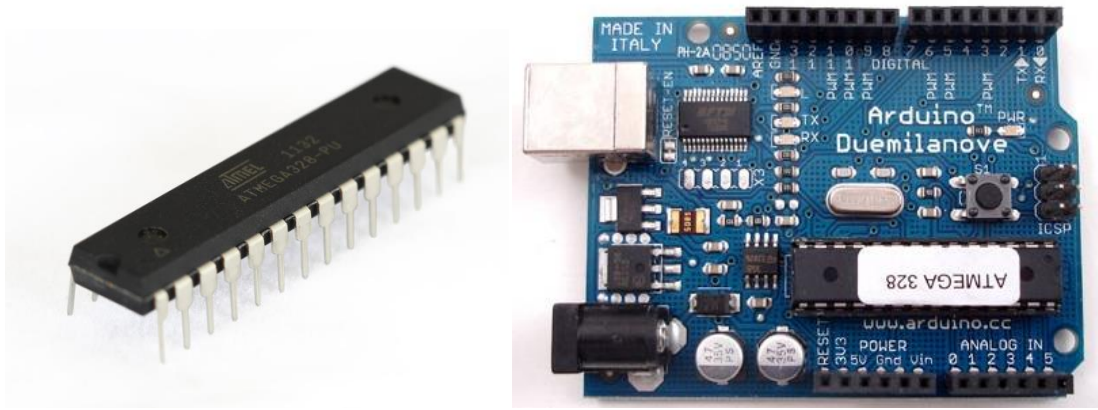
3.1.3. Motor Sürücüleri

Arka tekerleklerin hız kontrolünü yapmak için DC motor sürücü entegresi olarak, L293D entegresi ile birlikte arduino kontrol kartı kullanılmıştır (Şekil 3.5).



Şekil 3.5. DC motor sürücü entegresi.

Çalışmada kullanılan ön tekerlek ve lazer metrede kullanılan üç adet servo motorun istenilen açılarda dönmesini sağlamak, doğrudan Arduino karttan PWM sinyal gönderilerek sağlanmıştır. Kullanılan Arduino Duemilanove 328 kart, Atmega 328 tabanlı bir mikro denetleyici kartıdır (Şekil 3.6). 14 adet dijital giriş ve çıkışı mevcuttur. Bu giriş ve çıkışlardan 6 tanesi PWM çıkışı olarak kullanılabilir. 6 adet analog girişi ve 2 adet UART (Donanımsal Seri Çıkış)'a sahiptir. Mikrodenetleyici 16 MHZ hıza sahiptir. USB kablosu ile bilgisayara bağlanarak harici bir programlayıcı devre gereksinimi olmadan programlanabilmektedir. Ayrıca programlamanın yapıldığı USB girişi üzerinden de veri alışverişini gerçekleştirilebilmektedir. Mikro denetleyiciye ait özellikler aşağıdaki gibidir (Çizelge 3.1).



Şekil 3.6. Atmega 328 denetleyicisi ve Arduino Duemilanove denetleyici kartı.

Çizelge 3.1. Arduino Duemilanove denetleyici kartı özellikleri.

Çalışma Gerilimi	5V
Besleme Gerilim Aralığı	7-12V
Besleme Gerilim Limitleri	6-20V
Dijital Giriş Çıkış Pinleri	14 (6 adet PWM Çıkışı)
Analog Giriş Pinleri	6
Giriş – Çıkış Pin Akım Değeri	40 mA
3.3V Çıkış Pin Akım Değeri	50 mA
Program Hafızası	32 KB
SRAM	2 KB

Çizelge 3.1. (Devam ediyor)

EEPROM	1 KB
İşlemci Hızı	16 MHz

3.1.4. Kuru Bakımsız Akü

Bakım gerektirmeyen sızdırmaz tip kurşun-asit aküleri küçük ve hafiftirler. Yüksek performansa sahip olduklarından kullanımları çok ekonomiktir. Tam sızdırmaz ve bakım gerektirmez yapılarından dolayı içerisine su veya elektrolit ilavesine gerek yoktur. Bu akülerin yapıları, tampon şarj ve çevrimli (sürekli olarak şarj-deşarj) kullanımları için tasarlanmış olduklarından, ana ve acil enerji kaynağı olarak kullanıma çok uygundur. Çevrimli kullanımlarda % 100deşarjda 250 çevrim (bir çevrim, şarjı takiben tamdeşarjdan ibarettir), tampon şarj kullanımlarında ise 3-5 yıllık ömür elde edilebilir. Bu aküler portatif el aletlerinde, kameralarda, ölçüm cihazlarında vb yerlerde yüksek performanslı enerji kaynağı olarak çok yaygın kullanılmaktadır (Şekil 3.7). Otonom robotta iki adet akü birbirine seri olarak bağlanarak 18 V ve 12V halinde kullanılmıştır. 18V lazer tarayıcıda, 12V ise kart beslemesi ve motorlar için kullanılmıştır .



Şekil 3.7. Kuru bakımsız akü.

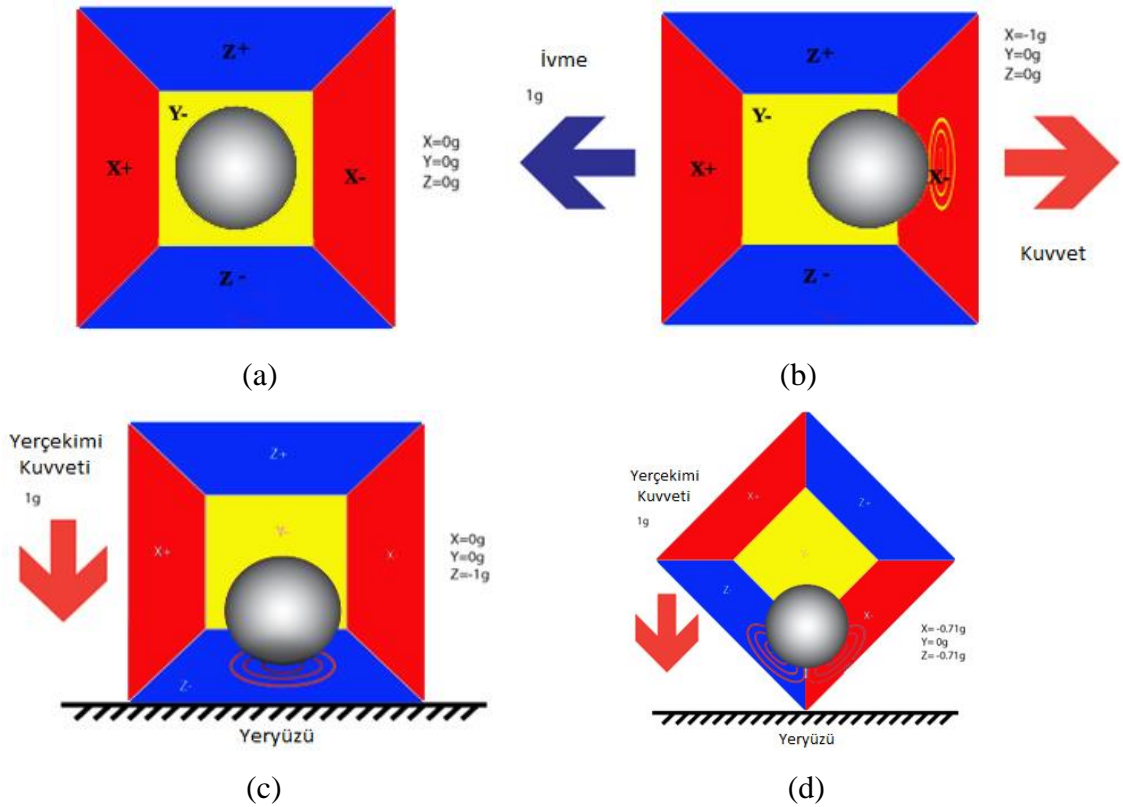
3.1.5. Sensörler

3.1.5.1. İvmeölçer

İvmeölçer, içerisinde mikro mekanik ölçüm sistemi bulunan ve x,y,z eksenlerinde oluşan ivmelerin durumuna göre gerilim üreten elektronik bir birimdir [52]. Eğer

sensör hareketsiz ise algılayıcı çıkışı yerçekimi ivmesi 1g olmaktadır. Bu özelliği ile hareket algılayıcı ve eğim sensörü olarak kullanılmaktadır.

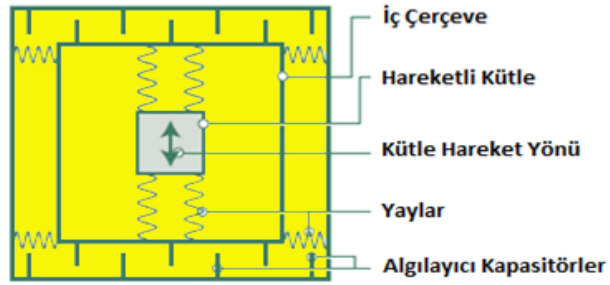
İvmeölçer bir küpün içerisindeki 6 adet iç yüzeyde, basınca duyarlı piezoelektrik madde ile bir küreden ibarettir [53]. Bu küre yerçekimi olmayan bir ortamda olduğu varsayıldığında resimdeki gibi askıda kalacaktır (Şekil 3.8.a). Kutuyu 1g kuvvet ile +X yönünde hızlandırdığımızda küre -X yönünde 1g kuvvet meydana getirecektir (Şekil 3.8.b). Bu kutunun dünyada olduğu varsayılırsa dünyanın 1g yer çekimi kuvveti nedeniyle küre -Z yönüne 1g kuvvet uygulayacaktır (Şekil 3.8.c). Kutuyu 45 derece çevirdiğimizde -X ve -Z yüzeylerine $\sqrt{2}/2$ g kuvvet uygulamaktadır. Bu da yaklaşık olarak 0,71g olmaktadır (Şekil 3.8.d). Bu çalışmada GY-521MPU-6050 ivmeölçer kullanılmıştır.



Şekil 3.8. İvme ölçerin çalışması. a) Yerçekimsiz ortam b) Hareket halinde ivmelenme, c) Yerçekimli ortam, d) Yer çekimli ortamda 45 derece açı ile duruş.

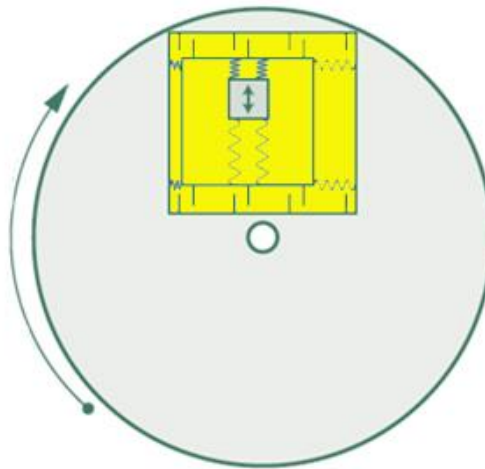
3.1.5.2. Jiroskop

Jiroskop bir nesnenin açısal hızını ölçmek için kullanılır. Elektronik jiroskoplar bir eksen etrafında döndürüldüğünde dönüş hızı ile orantılı olarak gerilim üretmektedir. Jiroskop hassasiyeti saniyede bir derece hızla döndüğünde sensörün verdiği çıkıştır. Dönme hızından kaynaklanan gerilimi doğru şekilde okumak için Jiroskop hassasiyetinin bilinmesi gerekmektedir. Jiroskop hassasiyetleri kullanılan modele göre farklılıklar göstermektedir. Jiroskop'un mekanik iç yapısı aşağıdaki gibidir (Şekil 3.9).



Şekil 3.9. Jiroskop'un iç yapısı.

Aşağıdaki şekilde dönme yönü saat yönünde belirlenmiştir. Dönme gerçekleştiğinde merkezkaç kuvveti rezonans kütlelerini dışa doğru itmektir. Ayrıca dönüş yönüne göre iç çerçeve sola kaymaktadır. Bu sayede hem açısal hız hem de dönüş yönü saptanabilmektedir (Şekil 3.10).

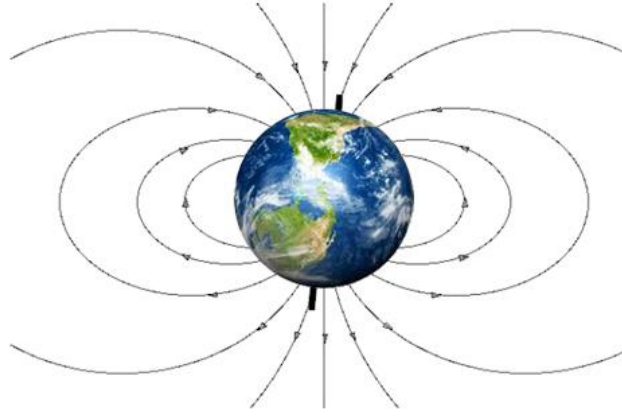


Şekil 3.10. Hareket anında jiroskopun çalışması.

Bu çalışmada kullanılan jiroskop GY-521MPU-6050 jiroskoptur. Bu jiroskop I²C seri bağlantı portunu kullanarak mikro denetleyici ile haberleşmektedir. Saniyede 2000 dereceye kadar ölçüm yapabilmekte ve 800 Hz çıkış verebilmektedir.

3.1.5.3. Manyetik Sensor

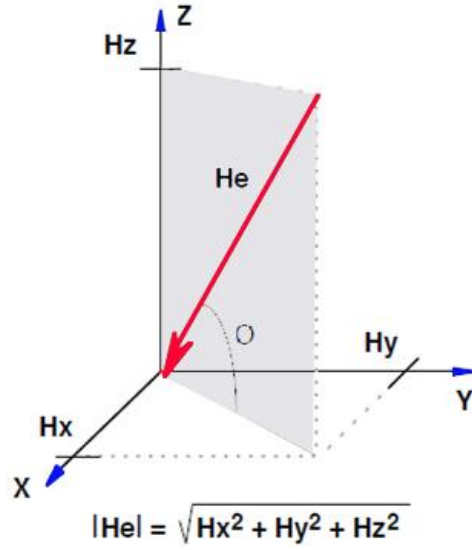
Dünyanın manyetik alanı çubuk mıknatısın manyetik alanı gibidir. Güney kutbundan başlayarak kuzey kutbuna kadar ulaşan bir manyetik alan şekli mevcuttur (Şekil 3.11). Kuzey ve güney kutupları dünyanın manyetik kutup noktaları olarak ifade edilmektedir.



Şekil 3.11. Yeryüzü manyetik alan çizgileri.

Manyetik Sensör ise dünyanın manyetik alan şiddetini ölçer. Manyetik sensör üç eksene etki eden manyetik alanın şiddetini ayrı ayrı göstermektedir. Aşağıdaki şekilde H_e yeryüzünün manyetik alan kuvvetini ifade etmektedir. H_x , H_y , H_z ise bu kuvvetin 3 eksenindeki bileşenleridir (Şekil 3.12).

Bu çalışmada kullanılan manyetik sensör GY-521MPU-6050 dir. manyetik alan şiddetini 3 eksende ölçebilmektedir. 8,1 gauss'a kadar ölçüm yapabilmektedir.



Şekil 3.12. 3-eksendeki manyetik alan vektörleri.

3.1.6. Bluetooth Bağlantı Modülü

HC06 Bluetooth-Serial Modül Kartı, Bluetooth SSP (Serial Port Standart) kullanımı ve kablosuz seri haberleşme uygulamaları için tasarlanmıştır (Şekil 3.14). Hızlı prototipleşmeye imkan sağlaması arduino ve çeşitli devrelerde rahatça kullanılabilmesi için gerekli pinler devre kartı sayesinde dışarıya alınmıştır. Standart pin yapısı sayesinde istenilen ortamlarda rahatça kontrol edilebilir. Bluetooth 2,0'ı destekleyen bu kart, 2.4 GHz frekansında haberleşme yapılmasına imkan sağlayıp açık alanda yaklaşık 10 metrelik bir haberleşme mesafesine sahiptir. HC06 modeli varsayılan halde Slave olarak ayarlanmıştır. HC05 modelini Master veya Slave olarak kullanabilir.

HC06 Bluetooth Modül Özellikleri

Bluetooth Protokolü: Bluetooth 2.0+EDR(Gelişmiş Veri Hızı)

Haberleşme Frekansı: 2.4 GHz

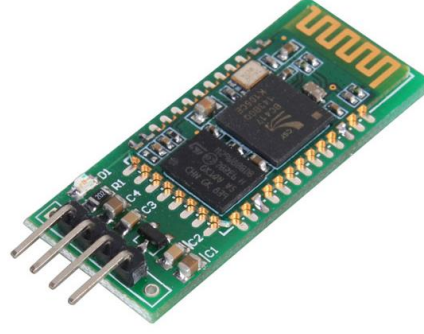
Hassasiyet: ≤ -80 dBm

Çıkış Gücü: $\leq +4$ dBm

Asenkron Hız: 2.1 MBps/160 KBps

Senkron Hız: 1 MBps/1 MBps

Güvenlik: Kimlik doğrulama ve şifreleme
Çalışma Gerilimi: 1.8-3.6V (Önerilen 3.3V)
Akım: 50 mA
Boyutları: 43x16x7mm



Şekil 3.13. HC06 Bluetooth Modül.

3.1.7. Lazer Mesafe Sensörü

Bu sensörler ışığın uçuş süresini hesaplamaya dayanan yöntemle sensör ve obje arasındaki mesafeyi ölçmektedir. Bu nedenle objenin renginin, biçiminin ve yüzey özelliklerinin anahtarlama noktasına neredeyse hiç etkisi yoktur. Koyu renkli objeler bile açık renkli bir arka planın önünde algılanabilmektedir. Objeye, ayarlanmış olan anahtarlama noktasına ulaştığında, çıkış anahtarlama yapmaktadır. Bu sensörler geniş bir algılama mesafesine sahiptir ve çok uzaktaki objeleri algılayabilmektedir.

Kullanılan lazer metre Efecto200 01d105 model lazer mesafe sensörüdür (Şekil 3.14)(Çizelge 3.2). Sensör doğrudan objenin üzerine çalışmaktadır. Çıkışlar, tanımlanmış olan mesafede anahtarlama yapmaktadır. Işık noktası uzak mesafelerden bile iyi bir şekilde görülebilmekte ve bu sayede ince ayarın ve işletmeye alma çalışmalarının kolayca yapılabilmesini sağlamaktadır. Çok ince ışın, küçük çaplı bir ışık noktası oluşturmaktadır. Bu sayede çok küçük parçalar bile sorunsuzca algılanabilmektedir. OLED ekranı, sensörlerin kolayca ayarlanabilmesini sağlamaktadır. Üstelik ölçüm sonuçları anında okunabilmektedir.



Şekil 3.14. Efactor200 01d105 Model Lazer Mesafe sensörü.

RS-232 arabirimi üzerinden sensörlerin fonksiyonları etkinleştirilebilmekte veya sonuçlar kaydedilebilmektedir. Ayarlar, menü kontrollü ekran üzerinden düğmelere basılarak veya harici olarak veya RS-232 arabirimi üzerinden yapılmaktadır. Buna yönelik grafiksel kullanıcı ara yüzüne sahip bir wenglor yapılandırma yazılımı, ayarların kolayca ve hızlı bir şekilde yapılmasını sağlamaktadır.

Çizelge 3.2. Efactor200 01d105 model lazer mesafe sensörü özellikleri.

Ürün özellikleri	
Optik mesafe sensörü	
Soketli	
Görünür Lazer ışığı, koruma sınıfı 2 lazer	
4-hane abece-sayısal gösterge	
Ölçme aralığı 0,2...10 m (Mesafe, % 90 geri yansıtmalı 200 x 200 mm beyaz kağıta göre tanımlanmıştır)	
Arka fon bastırmalı > 10...100 m	
Elektriksel özellikleri	
Elektriksel yapı	DC PNP
Çalışma gerilimi [V]	18...30 DC
Akım tüketimi [mA]	< 150
Tahmini ömür [h]	50000
Koruma sınıfı	III
Ters polarite koruması	Evet
Çıkışlar	
Çıkış fonksiyonu	OUT1: normalde açık / kapalı programlanabilir OUT2: normalde açık / kapalı programlanabilir veya analog (4...20 mA / 0...10 V, ölçeklendirilebilir)
Yük akımı [mA]	2 x 200
Kısa devre koruması	çıkış tarama
Aşırı yük koruması	Evet
Analog çıkış	
akım çıkışı [mA]	4...20; IEC 61131-2 ye uygun
- Maks. Yük [Ω]	250

Çizelge 3.2. (Devam ediyor)

Gerilim çıkışı [V]	0...10; IEC 61131-2 ye uygun
- Min. Yük [Ω]	5000
İzleme aralığı	
Arka fon bastırmalı	> 10...100 m
Işık hüzmesinin çapı	< 15 x 15 (Çalışma aralığı 10 m)
Ölçüm / ayar aralığı	
Ölçme aralığı [m]	0,2...10 (Mesafe, % 90 geri yansıtımlı 200 x 200 mm beyaz kağıta göre tanımlanmıştır)
Örnekleme frekansı [Hz]	1...33
Çevre koşulları	
Çalışma sıcaklığı [$^{\circ}\text{C}$]	-10...60
Koruma	IP 67
Testler / Onaylar	
EMC	EN 60947-5-2
MTTF [Yıl]	167
Mekanik özellikleri	
Gövde malzemesi	Gövde: çinko döküm; Pencere: Cam; LED pencere: Polikarbonat
Ağırlık [kg]	0,294
Gösterge / Kontroller	
Gösterge	Anahtarlama durumu 2 x LED sarı Çalışma LED yeşil Mesafe, Programlama 4-hane abece-sayısal gösterge

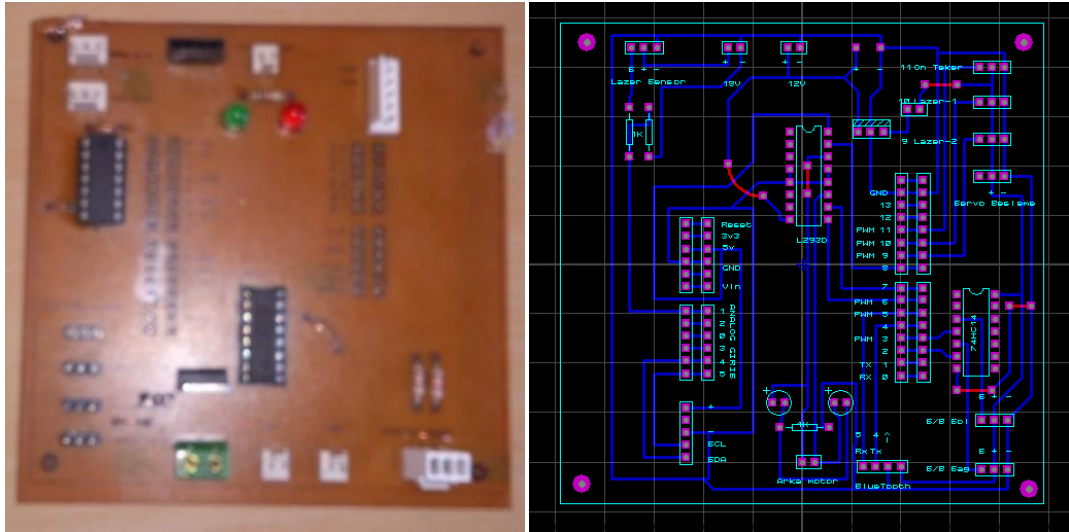
3.1.8. Kontrol Kartı

Ana kontrol kartı (Şekil 3.15) Arduino kontrol kartı üzerine takılmaktadır. Bu kart üzerinde 3 eksen jiroskop, 3 eksen ivmeölçer, 3 eksen ölçüm yapabilen manyetik sensör, bluetooth iletişim modülü, L293D motor sürücüsü ve 74HC14 entegresi bulunmaktadır. Robot bu kontrol kartı aracılığı ile bluetooth üzerinden c# programlama dili ile hazırlanmış kontrol yazılımı ile sürekli iletişim halindedir. Jiroskop, ivme ölçer ve manyetik sensörler aracılığı ile robot manyetik kuzey noktası referans alınarak 360 derecede yönünü hesaplayabilmektedir. Ayrıca motor sürücüsü, hareketi sağlayan DC motoru istenilen hızda ve istenilen yönde hareket ettirebilmektedir. Kart üzerinde 3 adet güç girişi bulunmaktadır. 12V güç girişi DC motoru ve 7805 voltaj regülatörü aracılığı ile gerilim düşürüldükten sonra servo motorları ve bluetooth modülü beslemektedir. 18V güç girişi lazer metrenin

beslemesini sağlamaktadır. 5V güç girişi ise ihtiyaç duyulması halinde servo motorlara harici güç beslemesi sağlamaktadır.

Ana Kontrol Kartına Bağlanan Devre Elemanları

- a) Arduino Due mikrodenetleyici kartı
- b) HC06 Bluetooth Modül.
- c) GY-521MPU-6050 Jiroskop
- d) L293D motor sürücüsü
- e) 74HC14 tampon
- f) 7805 Regülâtör
- g) CNY70
- h) Lazer metre
- i) Usb Portu
- j) Kontrol çıkışları

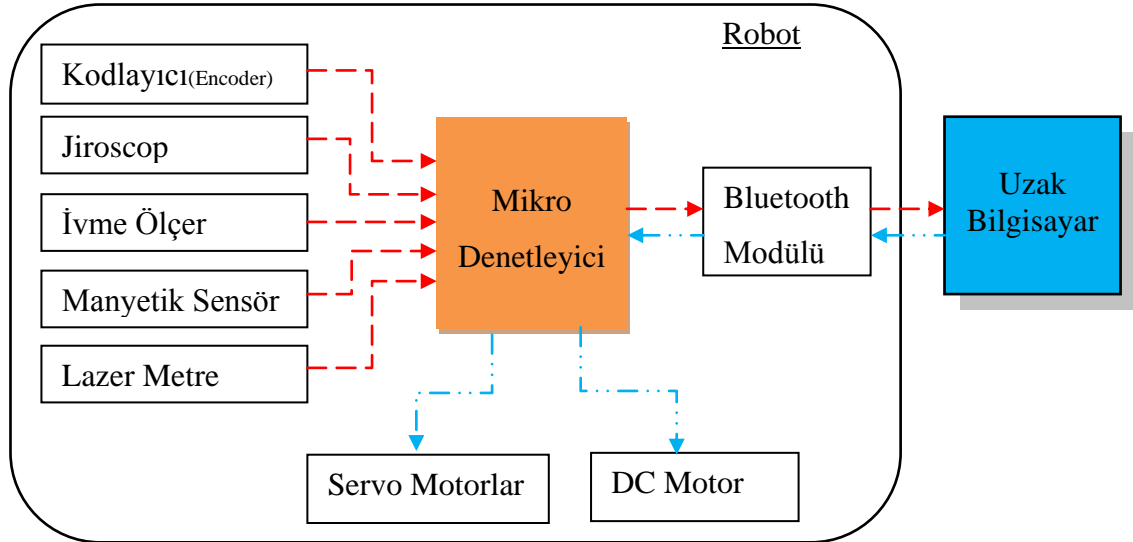


Şekil 3.15. Ana kontrol kartı ve devre şeması.

3.2. ROBOTUN KONTROL TASARIMI

Robotun karar ve denetim kısmı iki ayrı kategoride çalışacak şekilde tasarlanmıştır (Şekil 3.16). Birinci bölümde mikro denetleyici, algılayıcılardan gelen ham bilgiyi işleyip anlamlı hale getirip bluetooth modülü aracılığı ile uzaktaki bilgisayara gönderir. İkinci kısımda ise, bilgisayar gelen bilgileri işleyerek öncelikle “ben neredeyim” sorusunun cevabını bulur, ardından “nereye gideceğim” sorusunu cevaplandırır ve buna göre mikro denetleyici ile birlikte “nasıl gideceğim” sorusunun cevabını üreterek servo ve DC motorlar için gerekli sinyalleri üretir.

Robot başlangıç konum bilgisini üretmek için öncelikle jiroskop, ivmeölçer ve manyetik sensörden gelen bilgileri okur. Daha sonra lazer metre ile çevreyi 360 derece tarayarak oluşturduğu bilgileri bilgisayara gönderir. Program gelen bilgileri işler ve ortamın bir şeklini çıkarır. Elde edilen şekli, daha önceden mekânı dolaşırken oluşturduğu şekillerle kıyaslayarak lokalizasyon hesaplamalarını yapar ve başlangıç konumunu bulur.



Şekil 3.16. Robotun temel çalışma şeması.

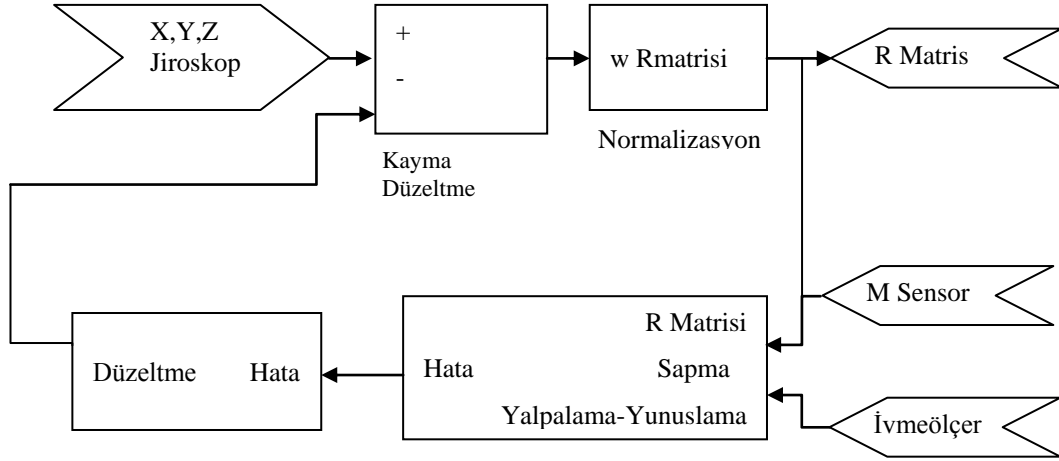
Robotun hareketi esnasında kodlayıcıdan (encoderdan) gelen bilgi, lazer metreden gelen bilgi ve ivmeölçerden gelen üç farklı mesafe bilgisi, robotun üzerindeki Atmel işlemcili Arduino kart tarafından önce işlenir daha sonra bilgisayara hesaplanan yol bilgisi kat ettiği yol ve yön bilgisi ile gönderilir.

Yön bulma işlemi için ise manyetik sensör yeterli olmasına karşın robotun eğimli ortamlarda da çalışması gerekebileceğinden kuzeye göre yönelme açısının bulunması gerekmektedir. Yönelme açısı hesaplanırken robotun x ve y eksenlerinde yaptığı eğimlerin de bulunması gerektiğinden robota jiroskop ve ivmeölçer sensörleri de eklenmiştir. Konum takibi ise arka tekerlere bağlı enceoderlardan gelen bilgi ile yapılmıştır. Aracın yönünün belirlenmesi için Direction Cosine Matrix algoritması kullanılmıştır.

3.2.1. DCM (Direction Cosine Matrix) Algoritması ile Açılı Tespiti

DCM algoritması ivmeölçer, jiroskop ve manyetik sensörden alınan değerlere göre yön durumunun sorunsuz olarak algılanmasını sağlamaktadır. Özellikle eğimli yol takip eden araçlarda sensörlerden okunan verilerin tek başına daha doğru ve kararlı olmaması nedeniyle aracın pozisyonunun ve yönünün tespiti için DCM algoritması kullanılmaktadır [54].

DCM algoritması şu şekilde çalışmaktadır. İlk giriş değerleri jiroskoptan x,y,z olarak alınmaktadır. Aracın mevcut durumunu saniyede 50 kez kontrol etmek amacı ile gerekli PWM sinyali üretilir. Burada belirli aralıklarda okunmadan kaynaklı aradaki ani değerler ve hesaplama esnasında oluşabilecek sayısal hatalardan dolayı matriste düzenli olarak sayısal değişiklikler meydana getirilmektedir. Birde jiroskop kayması birikimli olarak hatalara sebep olmaktadır. Algoritma bu hataları referans vektörleri kullanılarak saptanmakta ve PI kontrolcüsü kullanılarak dağıtılmaktadır. DCM algoritması manyetik sensörden gelen x,y,z değerleri ile sapma açısının düzeltilmesini, ivmeölçer gelen x,y,z değerleri ile de yalpalama ve yunuslama açılarının düzeltilmesini sağlamaktadır. DCM algoritması aşağıdaki şemada gösterilmiştir (Şekil 3.17).



Şekil 3.17. DCM algoritmasının çalışma şeması [54].

3.2.1.1. Rotasyon Matrisi

Bir cismin yeryüzüne göre konumu tanımlanırken cismin koordinat sisteminin, yeryüzü koordinat sistemine göre konumlandırılması gerekmektedir. Örneğin Robotun eksenini ile yeryüzü eksenini başlangıçta çakışık olarak düşünürsek ve robot z ekseninde α açısı kadar döndüğünde yeni birer x ve y eksenleri oluşur. Bu eksenler arası dönüşüm Rotasyon Matrisi ile gerçekleşmektedir.

Rotasyon matrisi 9 elemanlı bir matristir. Matrisin sütunları aracın gövde eksenine göre durum vektörlerini, satırları ise yeryüzü eksenine göre vektörlerini ifade etmektedir (Eşitlik 3.1). Sistemler arasında dönüşüm yapılmak isteniyorsa vektör rotasyon matrisi ile çarpılarak diğer sisteme dönüştürülebilir.

$$R = \begin{bmatrix} \ddots & \ddots & \ddots \\ \ddots & \ddots & \ddots \\ \ddots & \ddots & \ddots \end{bmatrix} \begin{matrix} x_e \\ y_e \\ z_e \end{matrix} \quad R^{-1} = \begin{bmatrix} \ddots & \ddots & \ddots \\ \ddots & \ddots & \ddots \\ \ddots & \ddots & \ddots \end{bmatrix} \begin{matrix} x_b \\ y_b \\ z_b \end{matrix} \quad (3.1)$$

Jiroskop belirli zaman aralıkları ile örnekleme sinyali almaktadır. Örnekleme zamanının sınırlı olması, örnekleme zaman dışındaki ölçülen değer değişiminin algılanamaması anlamına gelmektedir. Bu da ölçüm sonucunun hatalı olmasına sebep olabilmektedir. Rotasyon matrisinin özelliklerinden biri de ortogonal olmasıdır. Bu da aynı koordinat sistemine ait olan iki vektörün dik olduğunu göstermektedir. Örneğin robot sistemi ile yeryüzü sisteminin başlangıçta çakışık

olduğu düşündüğümüzde rotasyon matrisi birim matrise eşit olacaktır. Ancak sayısal hatalar nedeniyle bu değerler, giderek küçülecek ya da büyüyecektir. Bir süre sonra ya 0 olacak ya da sonsuza gidecektir.

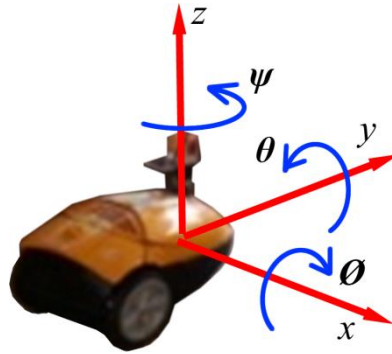
Rotasyon matrisi 9 elemanlıdır. Bu elemanlardan 3 tanesi bağımsızdır. Rotasyon matrisinin ortogonal olması matristeki bir satır ya da sütun çiftlerinin birbirine dik olduğunu göstermektedir (Eşitlik 3.2).

$$\begin{aligned} \|x_b\| = 1 & \quad x_b \perp y_b \\ \|y_b\| = 1 & \quad x_b \perp z_b \\ \|z_b\| = 1 & \quad y_b \perp z_b \end{aligned} \quad R = \begin{bmatrix} x_b & y_b & z_b \\ \dots & \dots & \dots \\ \dots & \dots & \dots \end{bmatrix} \begin{matrix} x_e \\ y_e \\ z_e \end{matrix} \quad (3.2)$$

Rotasyon matrisinin diğer özelliklerinden biri ise asimetrik matris olmasıdır. Bir asimetrik matriste bir elemanın değeri o elemanın satır ve sütuna göre simetriğinde bulunan değerın negatifine eşittir (Eşitlik 3.3).

$$\begin{vmatrix} 0 & a & b \\ -a & 0 & c \\ -b & -c & 0 \end{vmatrix} \quad (3.3)$$

Aracın yön durumu 3 adet açı ile tanımlanır. Bu açılara Euler açıları denir. Bu açılar aşağıdaki Şekil 3.18’de gösterilir.



Şekil 3.18. Robot üzerindeki eksenler ve dönüş açıları.

- Gövde, z eksenini etrafında döndürüldüğünde sapma (ψ) açısı meydana gelir
- Gövde, x eksenini etrafında döndürüldüğünde yunuslama (ϕ) açısı meydana gelir.

c) Gövde, y eksenini etrafında döndürüldüğünde yalpalama(θ) açısı meydana gelir.

Aracın koordinat sistemindeki bir vektör rotasyon matrisi ile çarpılırsa yeryüzü koordinat sistemindeki bir vektör bulunacaktır. Q yönü, hızı ya da ivmesi olan bir vektör olsun (Eşitlik 3.4).

$$Q = \begin{bmatrix} Q_x \\ Q_y \\ Q_z \end{bmatrix} \quad (3.4)$$

Q_P , Aracın referans sistemindeki bir vektör ve Q_G , yeryüzü referans sistemindeki bir vektör olsun.

Rotasyon Matrisi Eşitlik 3.5 de belirtildiği gibi olmak üzere;

$$R = \begin{bmatrix} r_{xx} & r_{xy} & r_{xz} \\ r_{yx} & r_{yy} & r_{yz} \\ r_{zx} & r_{zy} & r_{zz} \end{bmatrix} \quad (3.5)$$

$$Q_G = RQ_P \quad (3.6)$$

Direction Cosine Matris ve Euler açıları arasındaki ilişki aşağıdaki gibi gösterilmiştir (Eşitlik 3.7).

$$R = \begin{bmatrix} \cos\theta\cos\psi & \sin\theta\sin\theta\cos\psi - \cos\theta\sin\psi & \cos\theta\sin\theta\cos\psi + \sin\theta\sin\psi \\ \cos\theta\sin\psi & \sin\theta\sin\theta\sin\psi + \cos\theta\cos\psi & \cos\theta\sin\theta\cos\psi - \sin\theta\cos\psi \\ -\sin\theta & \sin\theta\cos\theta & \cos\theta\cos\theta \end{bmatrix} \quad (3.7)$$

Rotasyon matrisinin 3 sütunu, 3 eksenindeki vektörlerin, aracın referans çerçevesinden yeryüzü referans çerçevesine dönüşümüdür. 3 satırı ise 3 eksenindeki vektörlerin, yeryüzü referans çerçevesinden aracın referans çerçevesine dönüşümüdür. Rotasyon matrisi aracın yeryüzüne göre yönelimini açıklamak için gerekli tüm içeriğe sahiptir. Rotasyon matrisi Direction Cosine Matris olarak ta ifade edilebilir. Çünkü her bir giriş, araç ile yeryüzü referans sistemleri arasındaki açının kosinüsü şeklinde ifade edilir. Matrisin 3 elemanı bağımsızdır. Diğer 6'sı matrisin ortogonal olmasını sağlar.

Her bir sütundaki vektörler karşılıklı olarak diktir ve her birinin şiddeti 1'e eşittir [54].

3.2.1.2. Jiroskop Değerleri

Elektronik jiroskoplar araçla birlikte hareket etmekte olduğundan, dönüş hızı ile orantılı olarak çıkış üretir. Sayısal hatalar nedeni ile jiroskop hız değerinin integrali alınarak açı bilgisi doğru bir şekilde elde edilememektedir. X, Y ve Z eksenlerinde ölçülen jiroskop değerlerinden oluşan vektör, PI kontrolörden gelen düzeltme vektörü (ω_p , ω_i) ile toplanır. Oluşan yeni vektör ω olmak üzere bu vektörün t anındaki değişimi ile rotasyon matrisi çarpılır. Elde edilen matris (G_m) tekrar rotasyon matrisi ile çarpılır ve toplanır. Bu işlem "Kayma Düzeltme" aşamasında gerçekleştirilmektedir (Eşitlik 3.8,3.9).

$$G_m(t) = \begin{bmatrix} 0 & \omega_z dt & -\omega_y dt \\ -\omega_z dt & 0 & \omega_x dt \\ \omega_y dt & -\omega_x dt & 0 \end{bmatrix} \quad (3.8)$$

$$R(t) = R(t) + (R(t) \times G_m(t)) \quad (3.9)$$

3.2.1.3. Normalizasyon

Jiroskoptan okunan sürekli değerler birikimli olarak sayısal hatalara neden olmaktadır. Bu hatalar zamanla rotasyon matrisinin diklik koşulunu bozmaktadır. Rotasyon matrisin diklik koşulunu yeniden sağlanması gerekmektedir. Bu işlem normalizasyon aşamasında gerçekleştirilmektedir.

Öncelikle matrisin X ve Y satırları noktasal olarak çarpılır. Diklik koşulunun sağlanması demek, bu işlemin sonucunun 0 olmasıdır. Sonuca göre X ve Y nin birbirlerine göre dönme durumu hesaplanır. Bu dönme durumu ise hatayı oluşturur (Eşitlik 3.10,3.11).

$$X = \begin{bmatrix} r_{xx} \\ r_{xy} \\ r_{xz} \end{bmatrix} \quad Y = \begin{bmatrix} r_{yx} \\ r_{yy} \\ r_{yz} \end{bmatrix} \quad (3.10)$$

$$hata = X \cdot Y \quad (3.11)$$

Bulunan hata X ve Y vektörlerine eşit olarak paylaşılır (Eşitlik 3.12, 3.13).

$$X_{ortagonal} = X - \frac{hata}{2} Y \quad (3.12)$$

$$Y_{ortagonal} = Y - \frac{hata}{2} X \quad (3.13)$$

Z satırının diklik koşulunun sağlanması için ise X ve Y satırları çapraz çarpılır (Eşitlik 3.14).

$$Z_{ortagonal} = X_{ortagonal} \cdot Y_{ortagonal} \quad (3.14)$$

Normalizasyonun son aşamasında ise Rotasyon matrisinin her bir satırdaki vektörünün şiddeti 1'e oranlanır. Bu işlemi yapmak için hesaplanan satırdaki elemanların karelerinin toplamının karekökü ile, her satırın her elemanı bölünmektedir. Ayrıca burada Taylor Açılımı da kullanılabilir.

Satır vektörleri için büyüklük denklemleri aşağıdaki gibidir (Eşitlik 3.15-3.17).

$$X_{normalize} = \frac{1}{2} (3 - X_{ortagonal} \cdot X_{ortagonal}) X_{ortagonal} \quad (3.15)$$

$$Y_{normalize} = \frac{1}{2} (3 - Y_{ortagonal} \cdot Y_{ortagonal}) Y_{ortagonal} \quad (3.16)$$

$$Z_{normalize} = \frac{1}{2} (3 - Z_{ortagonal} \cdot Z_{ortagonal}) Z_{ortagonal} \quad (3.17)$$

3.2.1.4. Yunuslama ve Yalpalama Düzeltmesi

Jiroskoplar her ne kadar açı değişim bilgisini verse de kayma nedeniyle zamanla yanlış değerler üretmektedirler. Bunun sonucu olarak sapmalar oluşmaktadır. Burada referans vektöründe kayma olmaması gerekmektedir. Bu nedenle referans vektörlerinin ivmeölçer ve manyetik sensör ile desteklenmesi gerekmektedir.

Dikey rampalarda hareket eden bir araç öne-arkaya ve sağa-sola hareket edebilmektedir. Örneğin robotun yönü kuzeyi gösterirken robot güneye doğru da hareket edebilmektedir. Bu nedenle bu çalışmada sapma vektörünün düzeltilmesi GPS yerine manyetik sensör ile gerçekleştirilmiştir.

Jiroskoptan gelen X ve Y vektörlerinin düzeltilmesinde kullanmak için ivmeölçerin şiddetinin hesaplanması gerekmektedir. İdeal şartlarda ivme ölçere etkiyen tek kuvvet yer çekimi kuvvetidir. Farklı pozisyonlardaki şiddeti 1 e eşittir. İvme ölçerin şiddeti (A_{xyz}) 3 eksenide etkiyen kuvvetlerin kareleri toplamının karekökü alınarak bulunur (Eşitlik 3.18).

$$A_{xyz} = \sqrt{A_x^2 + A_y^2 + A_z^2} \quad (3.18)$$

İvme ölçerin şiddeti sıfırdan farklı ise araca etkiyen dış kuvvetler vardır. Bu durum ivme ölçerin ölçüm güvenilirliğini etkilemektedir. Bu nedenle düzeltme vektörü hesaplanırken ivme ölçerin etki oranı dikkate alınarak hesaplanması gerekmektedir. Etki oranı hesaplanırken 0,5 g den küçük ve 1,5 g den büyük şiddetler için ivme ölçer etkisi sıfır olarak değerlendirilmektedir. Belirtilen aralıktaki ivmeölçer etki (A_e) değerlerinin hesaplanması aşağıdaki şekilde gerçekleştirilir (Eşitlik 3.19-3.21).

$$f: [\mathbb{R}] \rightarrow [0,1] \quad (3.19)$$

$$f(x) = \begin{cases} 1 - 2 |1 - x|, & \frac{1}{2} < x < \frac{3}{2} \\ 0, & x \leq 0,5 \\ 0, & x \geq 1,5 \end{cases} \quad (3.20)$$

$$A_e = f(A_{xyz}) \quad (3.21)$$

X,Y eksenindeki düzeltme (V_{xy}) vektörü, rotasyon matrisinin z sütunu ile ivme ölçer vektörünün çapraz çarpılması ile elde edilir. Elde edilen vektör değeri ivme ölçerin etkisi ile çarpılarak PI denetleyiciye girilmektedir (Eşitlik 3.22).

$$V_{xy} = \begin{bmatrix} r_{zx} \\ r_{zy} \\ r_{zz} \end{bmatrix} \times A_{xyz} \quad (3.22)$$

PI denetleyici çıkışında ω_p ve ω_i vektörleri oluşacaktır (Eşitlik 3.23, 3.24).

$$\omega_p = K_{pxy} V_{xy} A_e \quad (3.23)$$

$$\omega_i = \omega_i + K_{ixy} V_{xy} A_e \quad (3.24)$$

Hesaplanan vektör değerleri ω_p ve ω_i vektörlerine sapma düzeltme vektörü eklendikten sonra jiroskoptan gelen vektör değeri toplanır ve döngü tekrar başa döner.

Rotasyon matrisi güncellendikçe 3 eksenindeki Euler açıları yeniden hesaplanmaktadır (Eşitlik 3.25, 3.26).

$$\theta = \text{atan2}(r_{zy}, r_{zz}) \quad (3.25)$$

$$\phi = -\text{asin}(r_{zx}) \quad (3.26)$$

3.2.1.5. Sapma Açısının Düzeltmesi

Bu işlem için manyetik sensör vektör değerleri ile yönelme vektörü değerleri hesaplanır. Manyetik sensörün 3 eksenindeki maksimum ve minimum ölçüm değerleri, yunuslama açısı ve yalpalama açısı kullanılarak yönelme vektörü hesaplanmaktadır (Eşitlik 3.27-3.29).

Manyetik sensörden okunan değerler ($P(x,y,z)$), 0 – 0,5 aralığına oranlanır. P_c düzeltilmiş pusula vektörü olmak üzere

$$P_{cx} = \frac{(P_x - P_{xmin})}{(P_{xmax} - P_{xmin})} - 0,5 \quad (3.27)$$

$$P_{cy} = \frac{(P_y - P_{ymin})}{(P_{ymax} - P_{ymin})} - 0,5 \quad (3.28)$$

$$P_{cz} = \frac{(P_z - P_{zmin})}{(P_{zmax} - P_{zmin})} - 0,5 \quad (3.29)$$

Düzeltilmiş ölçüm değerleri bulunduktan sonra θ ve ϕ açıları kullanılarak X ve Y yönündeki yönelme açıları bulunur. P_{yx} ve P_{yy} , X ve Y yönlerindeki yönelme şiddeti olmak üzere (Eşitlik 3.30-3.32);

$$P_{yx} = P_{cx} \cos(\phi) + P_{cy} \sin(\theta) \sin(\phi) + P_{cz} \cos(\theta) \sin(\phi) \quad (3.30)$$

$$P_{yy} = P_{cy} \cos(\theta) - P_{cz} \sin(\theta) \quad (3.31)$$

$$P_y = \text{atan2}(-P_{yx}, P_{yy}) \quad (3.32)$$

Bulunan değer Z yönündeki düzeltme vektörü (V_z) hesaplanmasında kullanılır. Vektörün hesaplanması için hata değerinin hesaplanması gerekmektedir (Eşitlik 3.33).

$$V_z = \begin{bmatrix} r_{zx} \\ r_{zy} \\ r_{zz} \end{bmatrix} (r_{xx} \sin(P_y) - r_{xy} \cos(P_x)) \quad (3.33)$$

Z düzeltme vektörünün bulunmasıyla W_p ve W_i vektörleri de rotasyon matrisi tekrar güncellenir ve program döngüsü yeniden başlar (Eşitlik 3.34, 3.35).

$$\omega_p = \omega_p + V_z K_{pz} \quad (3.34)$$

$$\omega_i = \omega_i + V_z K_{iz} \quad (3.35)$$

Rotasyon matrisinin güncellenmesi ile sapma açısı olan ψ hesaplanabilmektedir (Eşitlik 3.36).

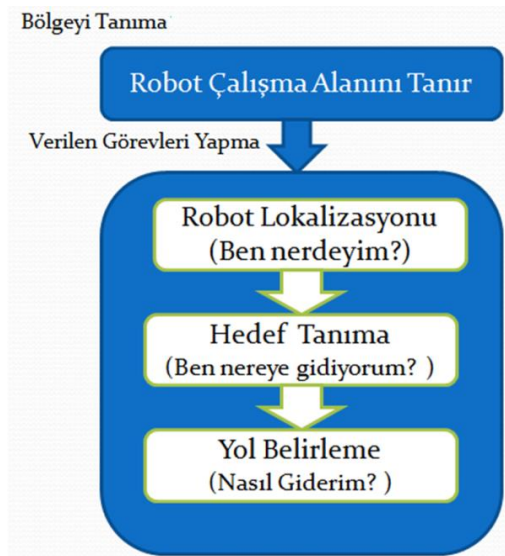
$$\psi = \text{atan2}(r_{xy}, r_{xx}) \quad (3.36)$$

3.3. ROBOT ve BİLGİSAYAR YAZILIMININ TASARIMI

Yazılım tasarımı iki aşamada ele alınmıştır. Birinci aşama, robotun çalışacağı alanın tanıtılması (öğrenme aşaması), ikinci aşama ise çalışırken yapacağı işlemlerin planlanması. Bu iki aşama Şekil 3.19’da grafiksel olarak gösterilmiştir.

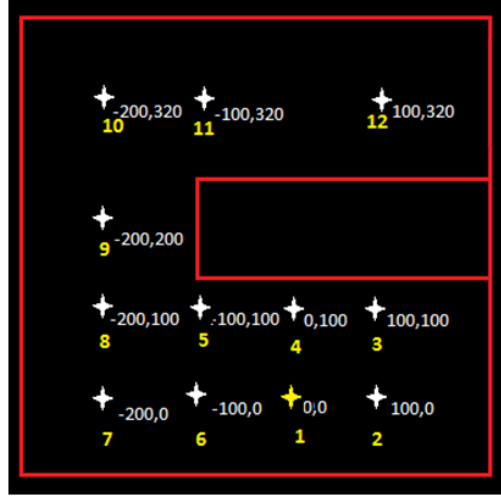
3.3.1. Bölgeyi Tanıma

Çalışma alanı, ortalama 50 cm ile 100 cm’lik adımlarla dolaştırılarak robotun hafızasında düğüm noktaları (ölçüm noktası) oluşturulur. Bu noktalara bağlı sonuçlar bir arada değerlendirilerek çevrenin haritası oluşturulur. Robot ilk konumlandığı noktayı kendi haritası için orijin (0,0) olarak kabul eder ve bir sonraki gideceği noktayı ilk nokta referans alınarak oluşturulur. Robot ölçüm sırasında lazer metreden ve jiroskoptan alınan bilgileri veri tabanındaki “harita” tablosuna “TaramaNo, KonumX, KonumY, OlcumX, OlcumY, Mesafe, Yön” şeklinde bir yapıda kaydeder.



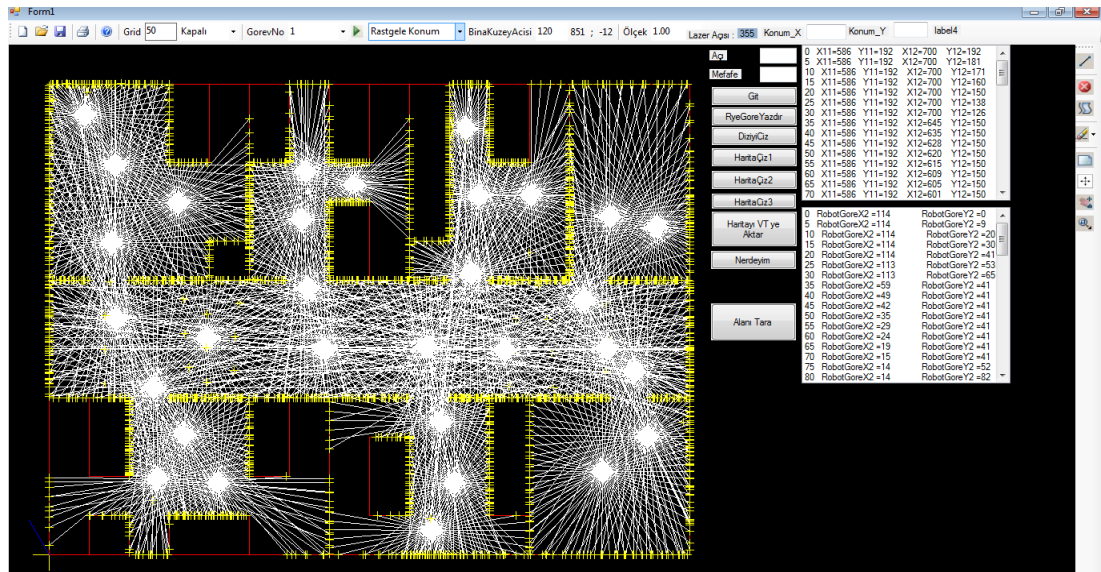
Şekil 3.19. Robot yazılımının tasarımı.

Robot taramaya başladığı noktanın koordinatı, kuzeye göre yaptığı açı değeri, ölçüm sonucu oluşan mesafe bilgileri ile hesaplamalar sonucu oluşan koordinat değerlerini, veri tabanına 6 derecede bir ölçüm yaparak yazar. İlk taramada başladığı koordinat (0,0) olarak girilir. Diğer taramalarda ise robotun aldığı yol pusula bilgileri ile hesaplanarak ilk tarama ya göre hesaplanır (Şekil 3.20).



Şekil 3.20. Örnek bir tarama alanı.

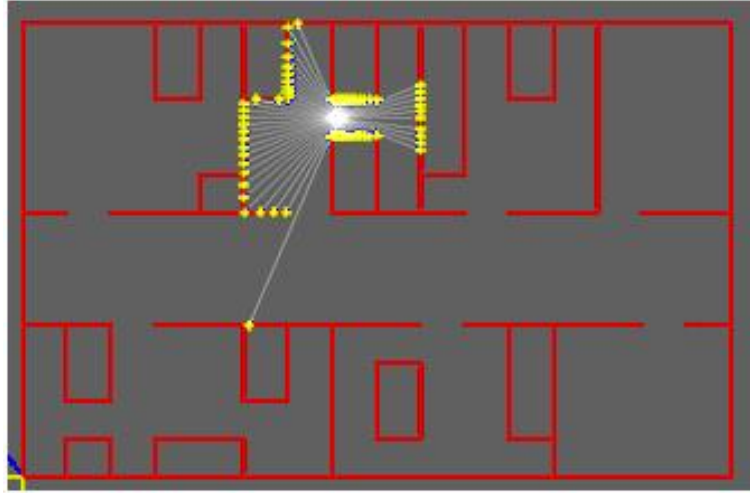
Veritabanı içinde bulunan harita tablosundaki bilgiler kullanılarak, tarama yapılan noktalar düğüm noktası kabul edilir ve buna bağlı olarak ortamın başlangıç geometrik haritası oluşturulur (Şekil 3.21).



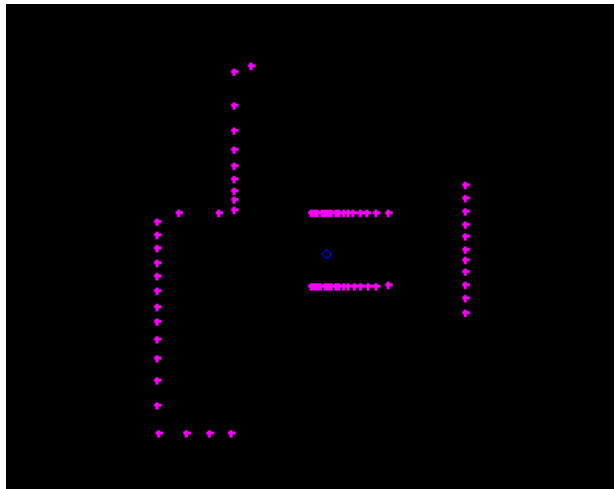
Şekil 3.21. Robotun ölçüm yaptığı düğüm noktaları ve oluşturulan başlangıç haritası.

3.3.2. Robot Lokalizasyonu

Robot kullanıcı tarafından herhangi bir noktaya bırakılıp, enerji verildiğinde ilk olarak bulunduğu alanı tarar. Lazer metreden gelen mesafe bilgileri ile jiroskoptan gelen pusula bilgilerini kullanarak taradığı alandan elde ettiği haritayı, hafızasındaki harita içine yerleştirmeye çalışır. Daha önceden taramış olduğu referans noktalarından hangisinin tarama bilgisine yakın olduğunu bulur. Böylelikle bulunduğu referans noktası kabaca robotun hangi odada ve neresinde olduğu bilgisini verir. Ardından mesafe ve pusula bilgileri ile konum hesaplama yaparak oda içindeki koordinatını daha hassas olarak elde edilir. Şekil 3.22-3.24'de bu işlemin nasıl yapıldığı, simülasyon programının ekran görüntüleriyle gösterilmiştir.

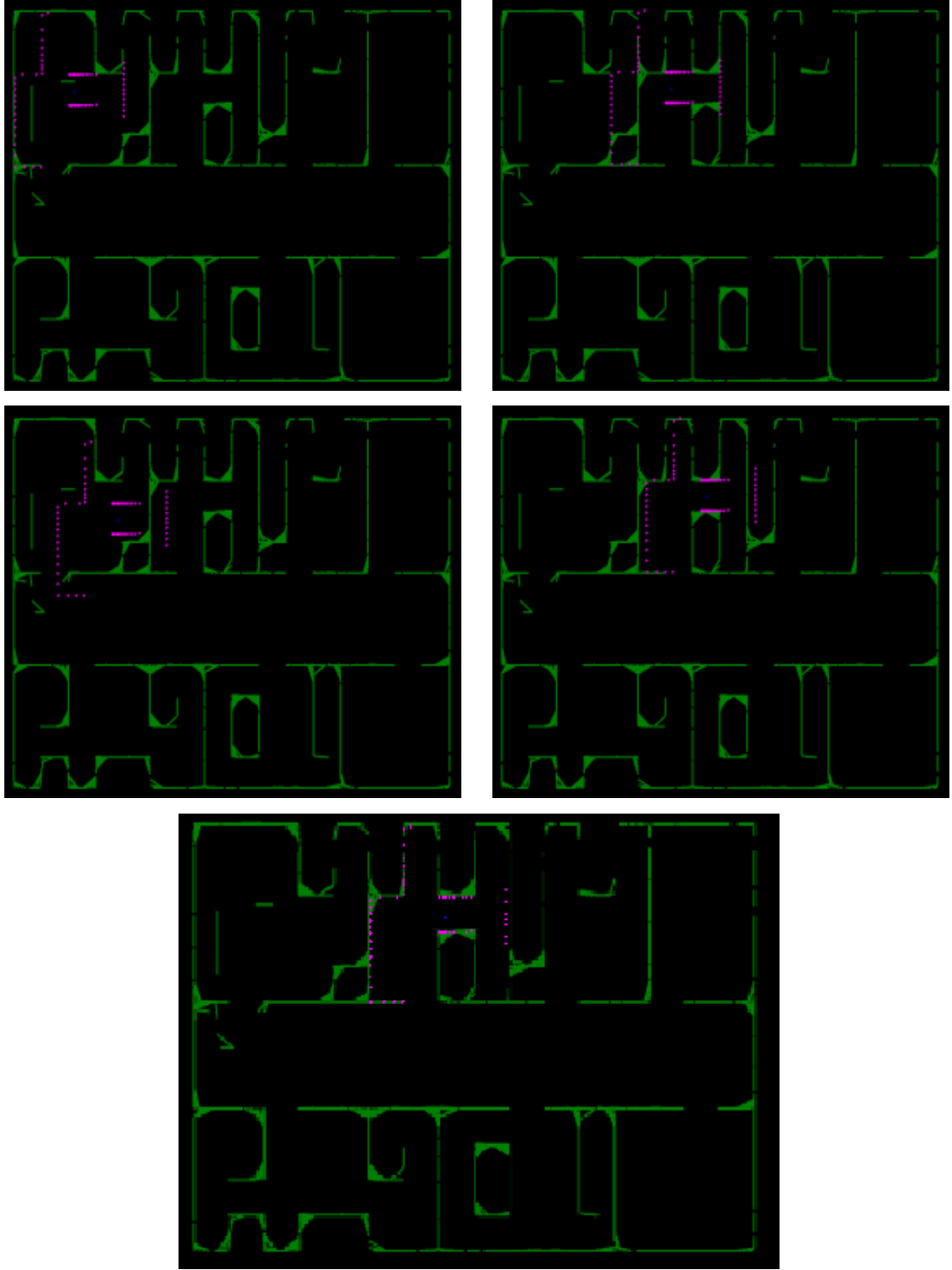


Şekil 3.22. Robotun lokalizasyon için odayı taraması (simülasyon ekranı).

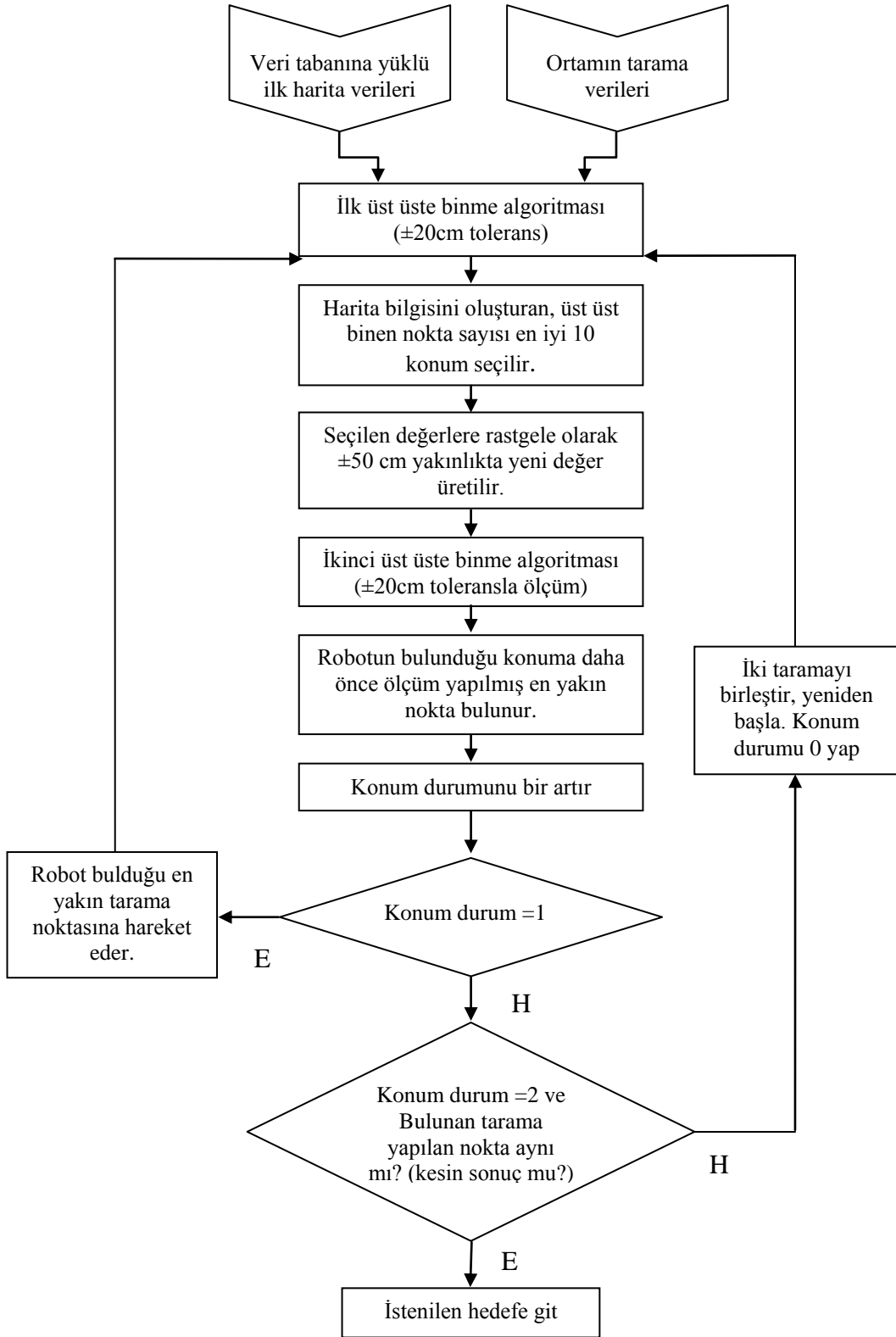


Şekil 3.23. Robotun hafızasında oluşan tarama şekli (simülasyon ekranı).

Robot lokalizasyonu için programın algoritmasının nasıl çalıştığı akış diyagramı şeklinde (Şekil 3.25)'de gösterilmiştir. Burada hafızadaki bilgiler sürekli yeni gelen tarama bilgileri ile karşılaştırılır ve gelen sonuca göre tahmini bir konum bilgisi üretir ikinci gelen tarama bilgisi ile de sağlama yapılarak kesin sonuç bulunur.



Şekil 3.24 Tarama noktalarını eşleştirilmesi (simulasyon ekranı).

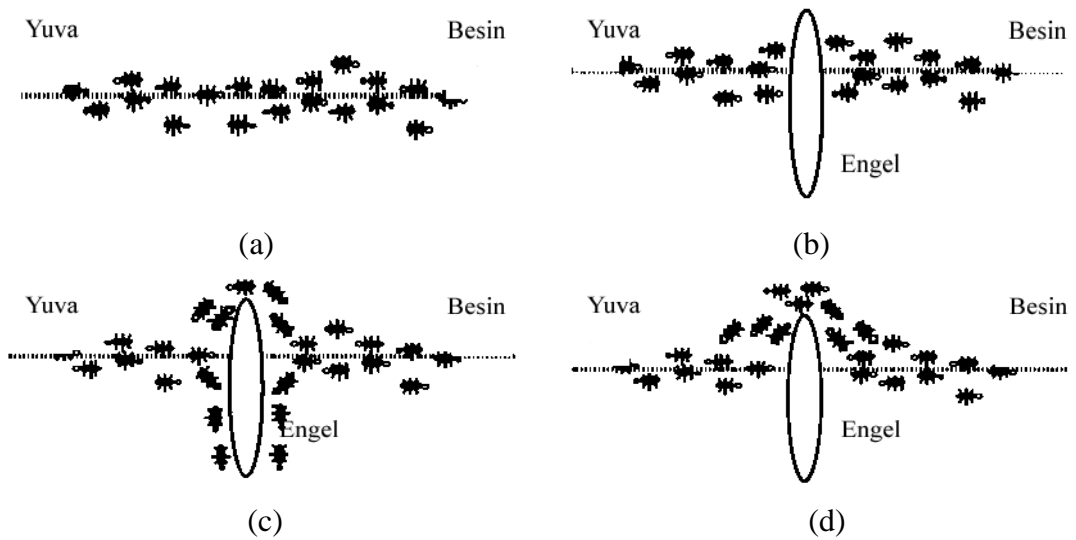


Şekil 3.25. Robot lokalizasyonu akış diyagramı.

3.3.3. Rota Belirleme

Robot kullanıcı tarafından verilen görev hedefine ulaşmak için, bölgeyi tanıma aşamasında hafızasına yüklediği düğüm noktaları kullanır. Bu noktalar üzerinde en kısa yolu hesaplamak için Karınca Kolonisi Algoritmasını (KKO) kullanır. Daha sonra belirlenen rotaya kendisi otomatik olarak hareket eder.

Karınca Koloni Algoritması, doğadaki karıncaların davranışlarından üretilmiş bir optimizasyon algoritmasıdır. Doğada karıncalar, görsel ipuçları kullanmadan yiyecek kaynağından yuvaya en kısa yolu bulma yeteneğine sahiptir [55]. Karıncalar yuvaları ile yiyecek kaynağı arasında düz bir çizgi halinde hareket ederler. Yürürken belli miktarda feromoni bırakırlar ve her karınca feromence zengin bir yönü takip etmeyi olasılıklı olarak tercih eder. Yol üzerine bir engel konulduğunda, engelin hemen önünde olan karıncalar feromon izini takip etmeye devam edemez ve bu yüzden sağa veya sola dönmek arasında bir seçim yapmak zorunda kalırlar. Bu durumda karıncaların yarısının sağa dönmeyi ve diğer yarısının sola dönmeyi seçtiği düşünülür (Şekil 3.26). Daha kısa yolu tercih eden karıncalar, aynı zaman diliminde kısa yol üzerinde daha fazla gidip geleceği için, bırakılan feromon miktarı uzun yola göre daha fazla olacaktır. Kısa yol üzerinde feromon miktarının gittikçe artması ile diğer karıncalarda o yöne doğru yönleneceklerdir.



Şekil 3.26. Doğadaki karıncaların en kısa yolu bulmadaki davranışları.

Programda Karınca Koloni Algoritması (KKO) kullanılırken, hafızada kayıtlı referans noktaların düğüm noktaları olarak kullanır. Bulunduğu nokta ile hedef nokta arasında algoritmanın adımlarını uygular. Bu adımlar şu şekildedir.

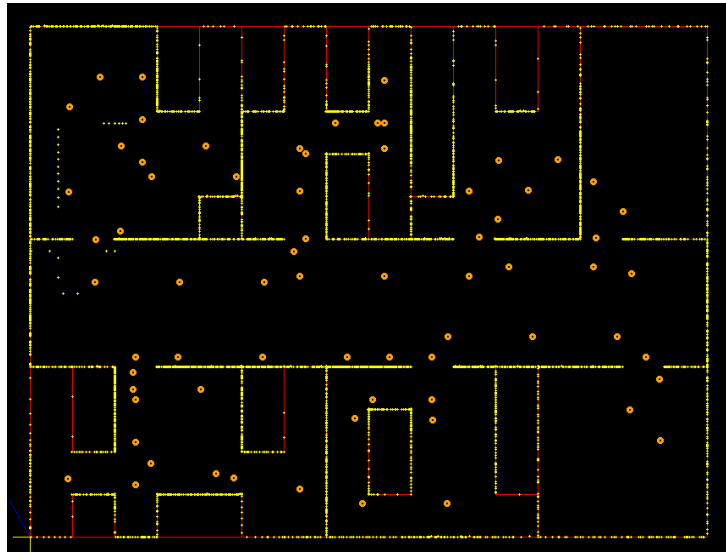
- a) Önce referans noktalar, veri tabanında ilk harita oluşturulurken robotun tarama yaptığı noktalar olarak belirlenir.
- b) Bu noktalar arasındaki güzergâh seçimi iki farklı formül kullanılarak bulunur. Birinci formül her düğüm noktasındaki feromon miktarına göre hareket ederken, ikinci formül doğrudan mesafeye göre hesap yapmaktadır (Eşitlik 3.37, 3.38). Burada, öncelikle hangi formülü kullanacağımızı seçmede belirleyici parametreler olan q ve q_0 değerlerini açıklamak gerekir. q_0 , 0 ile 1 arasında olan ve önceden belirlenen bir parametredir ($0 \leq q_0 \leq 1$). q_0 parametresi programda 0,9 olarak kullanıldı. Yani % 90 olasılıkla formülün kullanılması sağlandı. Bu değerle oynanarak programın çalışması test edilir ve uygun sonuçlar bulunmaya çalışılır. q ise her düğüm seçiminden önce program tarafından 0 ile 1 aralığında rastgele olarak üretilen bir değerdir ($q, [0,1]$). Kullanılan programda q 0,9'un üzerinde çıkmadıkça ikinci formül kullanılmamaktadır. Eğer $q \leq q_0$ şartı sağlanırsa Eşitlik 3.37 çalışır (%90 olasılıkla), şart sağlanmaz ise Eşitlik 3.38'e göre seçim yapılır (%10 olasılıkla).

$$\delta = \begin{cases} \arg \max \{ [\tau(r, u)] \cdot [\eta(r, u)]^\beta \} & \text{eğer } q \leq q_0 \\ P_k(r, s) & \text{Diğer durumda} \end{cases} \quad (3.37)$$

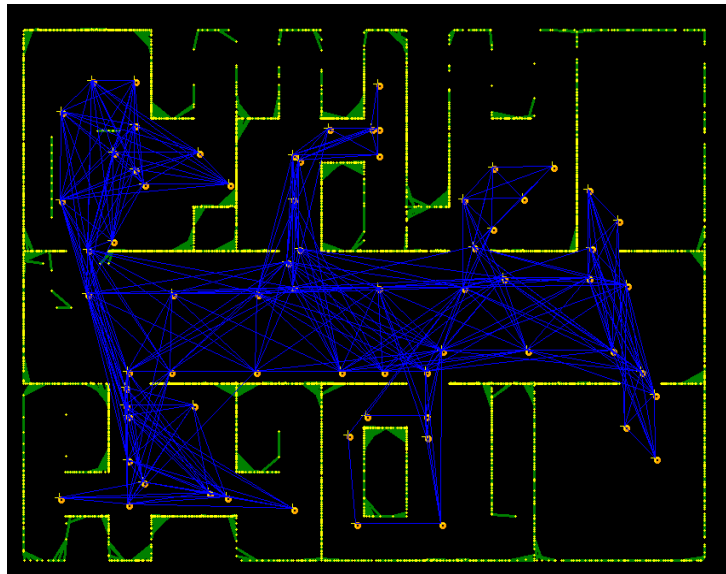
$$P_k(r, s) = \begin{cases} \frac{[\tau(r, u)] \cdot [\eta(r, u)]^\beta}{\sum_{s \in M_2} [\tau(r, u)] \cdot [\eta(r, u)]^\beta} & \text{eğer } S \notin M_k \\ 0 & \text{Diğer durumda} \end{cases} \quad (3.38)$$

Bu tür problemlerde Genetik algoritmalar da kullanılabilir. Genetik algoritmalarla bu tarz problemlerin çözümünde en çok 30 düğüm arasında optimum sonuçlar elde edilmektedir. Düğüm sayısı 30'un üzerine çıktığında genetik algoritmalar problemin çözümünde diğer yapay zeka yöntemlerine göre dezavantajlı duruma düşmektedirler. Bu çalışmada 70 civarında düğüm bulunduğu için KKO algoritması çözüm üretmede

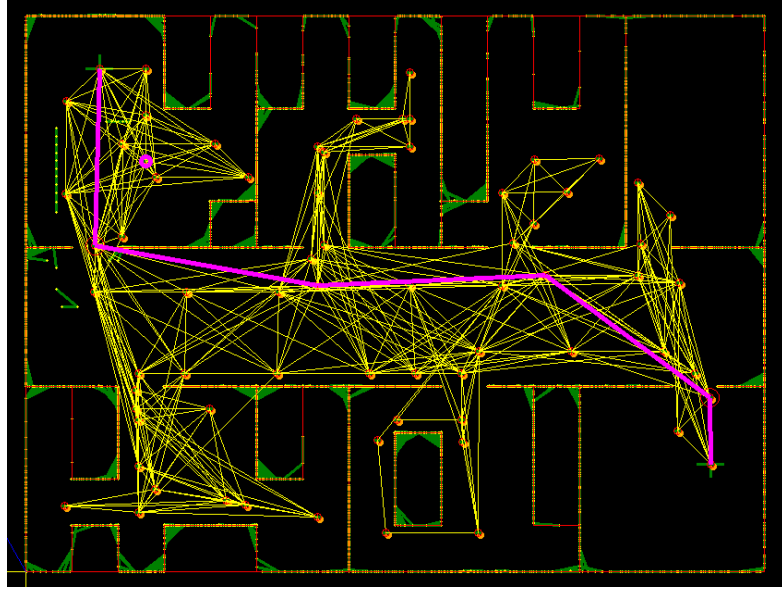
daha başarılı olmuş ve uygun bir yöntem olarak faydalanılmıştır. KKO algoritması ile 100 düğüme kadar olan problemlerde optimum çözüm elde edilebilmektedir [56]. Düğüm sayısı bu değer de üzerine çıkarsa çözüm süresi çok fazla uzamaktadır ve bu da robotun çalışmasında gecikmelere yol açabilecektir. O yüzden problemin çözüm kümesi 100 düğüme kadar olacak şekilde sınırlandırılmış ve çözüm yöntemi olarak KKO algoritması kullanılmıştır (Şekil 3.27-3.29). Böylelikle robotun aday düğümler arasında en kısa yolu minimum sürede hesaplayarak gecikmesiz hareket etmesi sağlanmıştır.



Şekil 3.27. Örnek bir binada robotun gezineceği düğüm noktaları.



Şekil 3.28. Örnek bir binada robotun gezinebileceği yollar.



Şekil 3.29. Örnek bir binada robotun KKA kullanarak bulduğu kısa yol.

3.4. YAZILIMLAR

Bu çalışmada toplam 3 adet program yazılmıştır. Birincisi robotun üzerindeki Arduino kartta çalışan Mikrodenetleyici yazılımı, diğer iki ise bilgisayar üzerinde çalışan Sunucu ve Simulasyon yazılımlarıdır.

3.4.1. Mikro Denetleyici Yazılımı

Arduino kartına C dili kullanılarak yazılmıştır. Program temel olarak iki ana işlevi yerine getirmektedir. Bunlardan birincisi robottaki algılayıcılardan gelen tüm ham veriyi işleyip anlamlı hale getirmek, diğeri ise robot donanımı ile bilgisayar yazılımı arasındaki iletişimi sağlamak amacı ile yazılmıştır. Maddeler halinde görevlerini yazacak olursak mikro denetleyici yazılımının yaptığı işler sırasıyla;

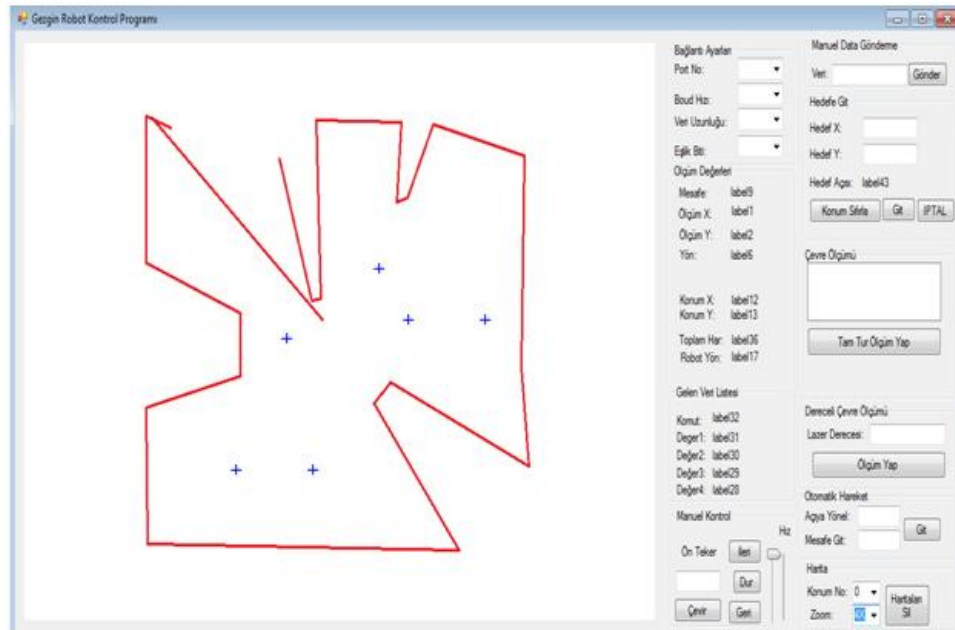
- Bluetooth ile iletişim (Sunucuya Bağlanma),
- Jiroskop, ivmeölçer ve manyetik sensörlerden verileri okuma,
- Jiroskop, ivmeölçer ve manyetik sensörlerden gelen verilerden Euler açılarının tespiti,
- Lazer sensörlerden okunan verilerin filtrelenmesi,
- Konum bilgilerinin sunucuya aktarılması,

- f) Sunucudan gelen verilerin yorumlanması,
- g) Kumanda sinyallerinin okunması,
- h) Konum ve durum değişimine göre pı hesaplamalarının yapılması,
- i) Hareket işini gerçekleştirme ve engel denetimi.

3.4.2. Bilgisayar Sunucu Programı

Robot ile bilgisayar arasındaki bağlantıyı yapar, robottan gelen bilgileri işler, yer bilgisi ve seyrüsefer işlemlerine karar verir. Aynı zamanda bilgisayarda kullanılmak için tasarlanan bu program kullanıcı ile robot arasındaki köprüdür. Programın görevleri maddeler halinde şu şekildedir;

- a) Kullanıcı başlangıç ayarlarını bu program aracılığı ile yapar. Mekânın robot tarafından tanınmasını, başlangıç girdilerinin oluşması bu program aracılığı ile gerçekleşir. Program başlangıçta kullanıcının gezdirdiği tüm noktalardan robotun duruş yönü, ölçüm yaptığı yön ve ölçümün mesafesinden iki boyutlu X,Y düzleminde bir harita oluşturur
- b) Oluşturulan haritaları birleştirir ve anlamlı tek bir alan görüntüsü elde eder (Şekil 3.30).



Şekil 3.30. Örnek bir odayı tanıma anında kullanılan 6 taramanın bileşke haritası.

- Çalışma ortamını öğrenen robot göreve başladığında çevresini yeniden tarar. Tarama bilgilerini sunucu bilgisayar programına yollar. Bilgisayar programı önceki veri tabanı bilgileri ile görev esnasındaki tarama bilgilerini karşılaştırır ve üst üste koyma algoritması ile kıyaslar. Tarama noktasının hangi referans noktaya yakın olduğunu bulur. Daha sonra son ölçümünden net koordinatını çıkarır. Bu işlemi sağlamasını yapmak amacı ile 20 cm ilerleyerek yeniden işlemi tekrar eder.
- Hedef bilgisinin kullanıcı tarafından girilmesi sağlar.
- Robotun gideceği güzergâhın hesaplama işlemi ve güzergâh adımlarını belirler.
- Elde edilen bilgilerin veri tabanında tutulması sağlar (Şekil 3.31).

Harita	Gezinti
Kimi	Konu
17952	2422
17953	2423
17954	2424
17955	2425
17956	2426
17957	2427
17958	2428
17959	2429
17960	2430
17961	2431
17962	2432
17963	2433
17964	2434
17965	2435
17966	2436
17967	2437
17968	2438
17969	2439
17970	2440
17971	2441

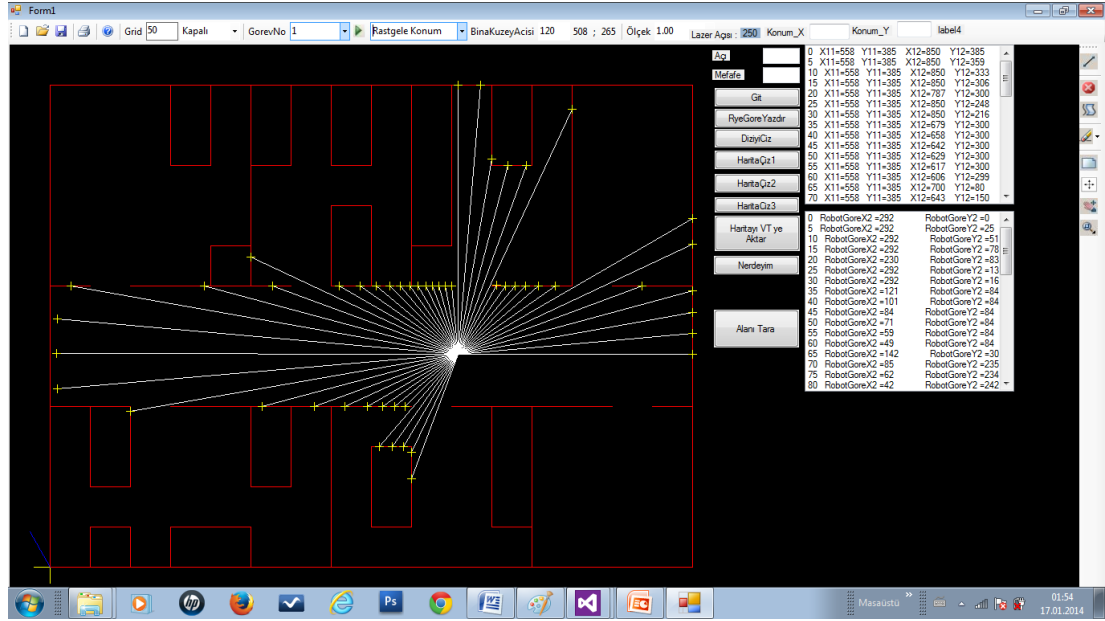
Şekil 3.31. Öğrenme ve gezinti aşamasında elde edilen tarama noktaları örneği.

3.4.3. Benzetim Programı

Robotun gerçekte planlanan tüm uygulamalarının bilgisayar ortamında testi için kullanılan programdır. Bu program altında şu işlemler gerçekleştirilir;

- Mekânı ölçekli çizme,
- Robotun hafızasının oluşturulması,
- Lazerle mekânı tarama benzetimi,
- Yer belirleme benzetimi,
- Hedefe yönelme ve en kısa yolu bulma benzetimi.

Benzetim programının kullanılması; donanımsal yapının kurulmadan önce, tarama ve seyrüsefer uygulamalarında nelerin olacağı, aralarındaki mantıksal bağlantının nasıl kurulacağı, kullanılacak yöntemlerin kapsamı gibi konuları önceden görme ve ona göre karar almayı sağlamıştır. Alınan tek taramalık bir örnekleme Şekil 3.32’de gösterilmiştir.



Şekil 3.32. Benzetim programında örnek bir tarama uygulaması.

BÖLÜM 4

DENEYSEL ÇALIŞMALAR

4.1. LAZER METRE İLE ÖLÇÜM VE HARİTA OLUŞTURMA DENEYLERİ

4.1.1. Lazer Metre ile Yapılan Kalibrasyon ve Ölçümler

Lazer metre genellikle stabil ölçümler yapmaktadır. Ölçüm yapılan bölgedeki yüzey yansımından kaynaklı ve ölçüm değerine göre lazer metre tarafından üretilen 0-10 volt aralığındaki voltajın mikro denetleyici tarafından 10 bit olarak dönüşümü yapılır. Bu nedenle her birim arası değer farkı 0,9765625 cm'lik aralıklar olarak hesaplandığından milimetrik değerler oluşmamaktadır. Buda 1 cm'den daha yakın ölçümler de algılanan değerlerin aynı olarak algılanmasını sağlamış, filtreleme görevi görmüştür. Lazer metre ile yapılan ölçümlerde ise 1 metrelik mesafelerde ölçülen noktanın renginin, değerleri etkilemediği, cam yüzeylerde ise $\pm 0,9765625$ cm'lik sapmaların olduğu gözlemlenmiştir. 5 metrelik mesafe ölçümlerinde ise 1 metredeki ölçümler gibi hassasiyetin değişmediği, fakat 10 metre ölçümlerinde zemin rengine ve yansımaya bağlı olarak $\pm 1,953125$ cm sapmaların olduğu gözlemlenmiştir. 10 metre den daha fazla olan bölümlerde ise lazer metre değer üretmemektedir.

Çizelge 4.1. Lazermetre ölçüm değerleri.

Mesafe	Ölçüm Yapılan Alan	Sonuç
100cm	Siyah	99,699 cm
100cm	Beyaz Yüzey	99,699 cm
100cm	Cam yüzey	97,635cm ile 100,253cm aralığında
500cm	Siyah	499,023 cm
500cm	Beyaz Yüzey	499,023 cm
500cm	Cam yüzey	493,164 cm ile 502,929 aralığında
1000cm	Beyaz	1000 cm

Çizelge 4.1. (Devam ediyor)

1000cm	Sarı	999,023 cm
1000cm	Gri	999,023 cm
1000cm	Siyah	996,093cm ile 1000 aralığında
1000cm üzeri	Boşluk	Değer üretmemektedir.

4.1.2. Euler Açı (Pusula) Ölçüm Değerleri

Kullanılan GY-521MPU-6050 tümleşik Jiroskop, İvme ölçer ve Manyetik sensör entegresinden okunan değerlerin DCM algoritması ile hataların tolere edilmesi sonucu bilgisayarda okunan robot yön değerleri ile cep telefonu pusulasından okunan sonuçların aynı olduğu gözlemlendi. Örnek bir ölçüm değeri Şekil 4.1 de gösterilmiştir. Burada bir cep telefonunun pusula programı ile robotun bilgisayara gönderdiği değerler karşılaştırılmış ve ölçülen değerlerin tutarlılığı gözlemlenmiştir. Robotun üzerindeki pusula ölçüm değeri 200 derece bilgisi ile ölçülen cep telefonundaki 200 derece pusula değeri aynı olduğu gözlemlenmiştir.

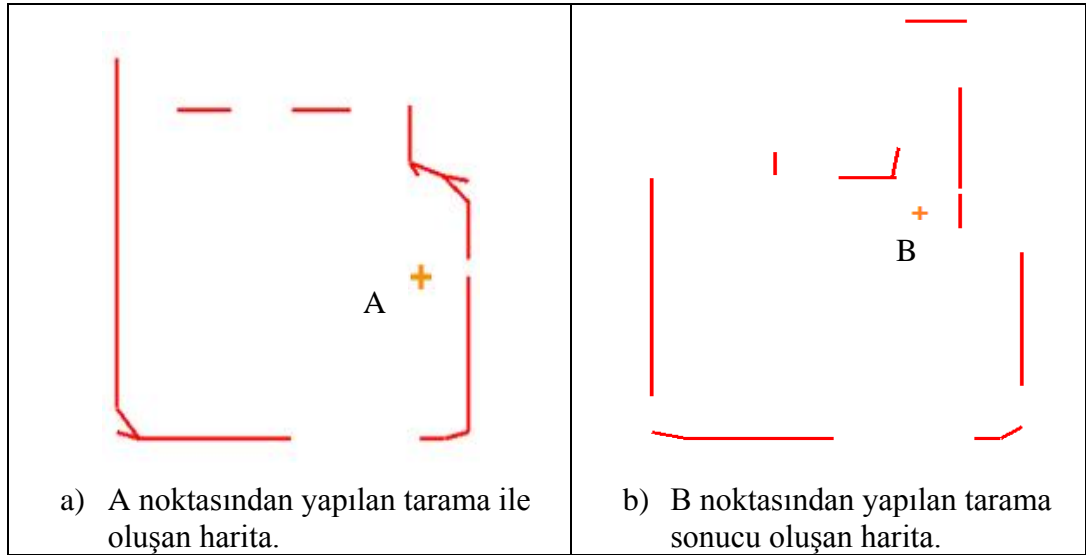


Şekil 4.1. Pusula Uygulaması ve Robotun algıladığı yönün karşılaştırılması.

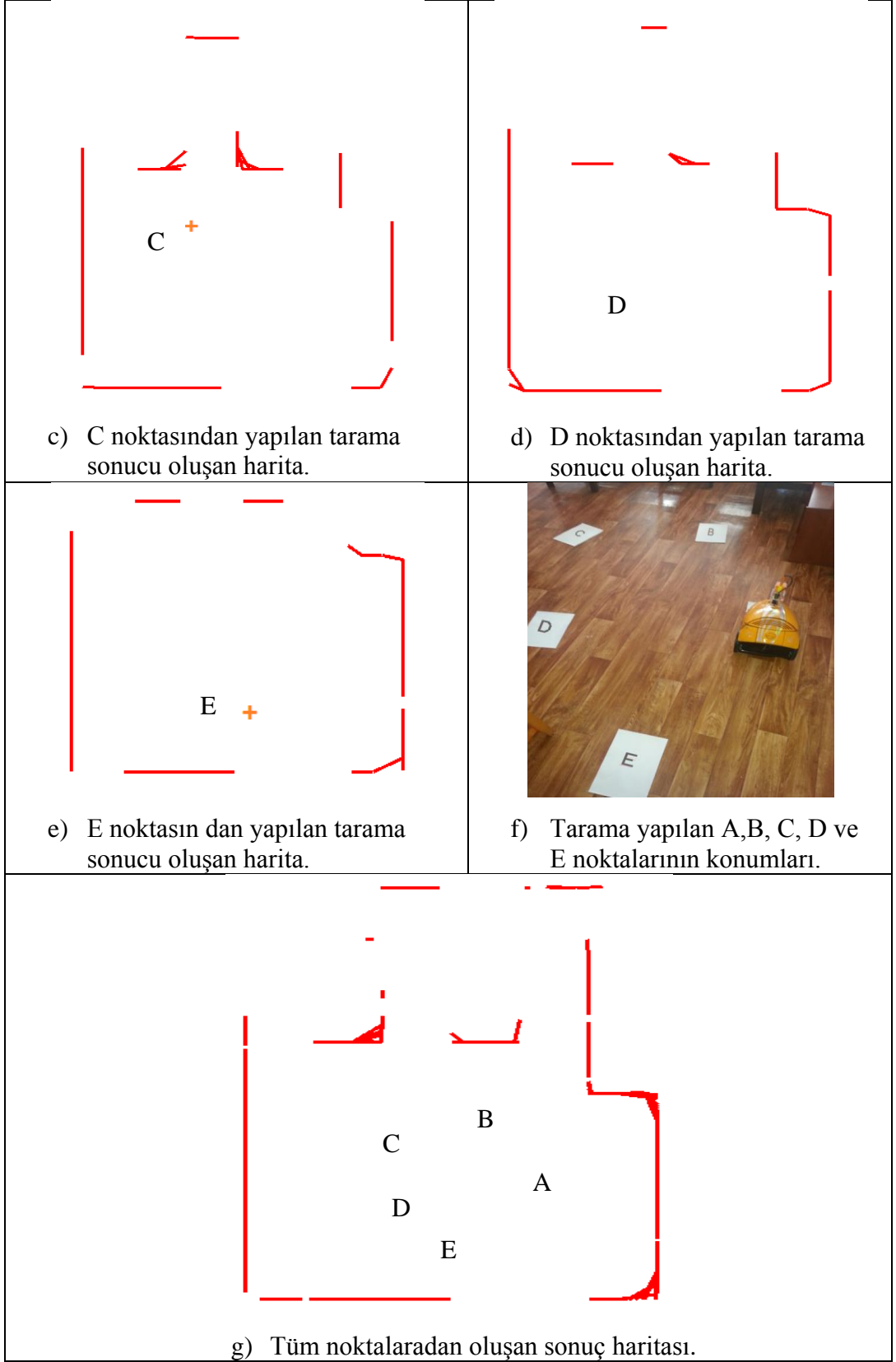
4.1.3. Robotun Oluşturduğu Harita ile Gerçek Alanın Kıyaslanması

Robotun çalıştığı ortamlar, bina içi ve birçok eşyanın bulunduğu ortamlar olduğu için, tarama esnasında robotun etrafı algılaması insan gözünün algıladığından farklı olmaktadır. Tarama esnasında direk duvarlardan alınan ölçümler, duvarların şeklini ortaya koymasına rağmen, sandelye ve masa ayakları ve bunun gibi süresiz olan cisimler, farklı bir geometri ortaya koymaktadır. Bazen bu geometri tarama noktasının yerine göre de farklılık göstermektedir. Bu yüzden robot taradığı her farklı noktada odanın farklı bir haritasını oluşturmaktadır. Bu yüzden algoritmanın temel mantığında konumlar, gezinme esnasında herhangi bir noktadan elde edilen haritanın, daha önceki öğrenme aşamasında elde edilen haritalardan hangisine yakın olduğunu bulunarak elde edilmiştir.

Şekil 4.2’de verilen resimlerde bir büro içerisinde farklı noktalardan elde edilen taramaların, odanın şeklini farklı farklı çıkardığı ortaya konulmuştur. Örnekte, robotun gezinme yaparken her bir tarama için oluşturduğu haritalar ile taramalar sonrası oluşan sonuç haritası verilmiştir.

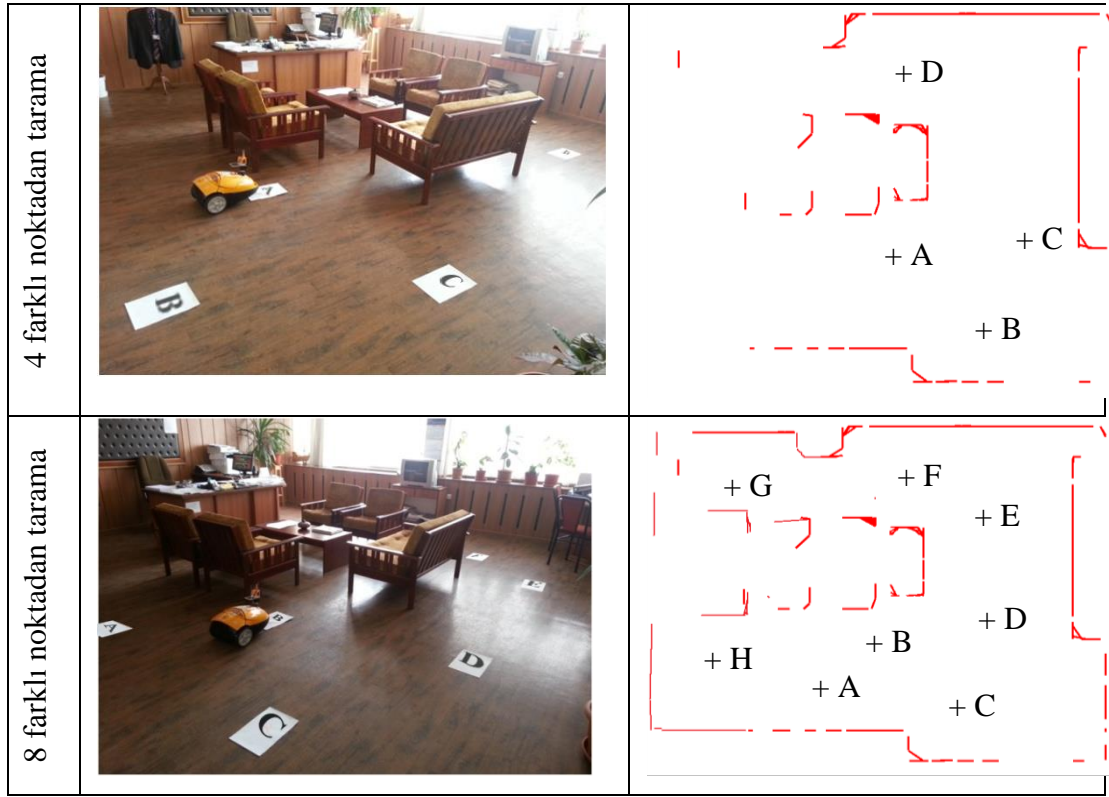


Şekil 4.2. Deney ortamının farklı noktalardan taranması ile elde edilen haritalar (a,b,c,d,e), tarama noktalarının yerleri (f) ve oluşan bütünleşik harita (g).



Şekil 4.2. (Devam ediyor)

Robot öğrenme aşamasında bina içerisinde birçok referans noktadan tarama yaparken her bir nokta için ayrı bir harita oluşturur. Daha sonra bu haritalar birleştirilerek binanın gerçek haritası ortaya çıkarılır. Aşağıda Şekil 4.3’de deney ortamında yapılan referans tarama noktalarının sıklığına bağlı olarak odanın gerçek haritası oluşturulmuştur. Referans tarama noktası ne kadar fazla olursa ve dağılımı ne kadar geniş olursa elde edilen haritada o kadar odanın gerçek görünümünü yansıtmaktadır. 8x6 metre ölçülerindeki deney ortamınının 4 ve 8 adet yapılan ölçümle en son elde edilen haritalar gösterilmiştir.



Şekil 4.3. Deney ortamının tarama sayılarına göre karşılaştırılması.

4.2. SEYRÜSEFER DENEYLERİ

4.2.1. Lokalizasyon Deneyleri

Robot pusulasında yardımıyla, sadece belirli gezme noktalarını referans alarak yer belirleme işlemlerinde yüksek başarı sağlamıştır. Robot ölçüm yaptığı her yeni noktanın koordinatını belirlerken, kendi hafızasındaki ilk tarama yaptığı (0,0) orijin

noktasını referans almaktadır. Şekil 4.4'te robotun tarama yaptığı noktanın ve ekrandaki koordinatı gözükmektedir. Robot burada ilk tarama işlemine B noktasından başlamaktadır. Robot B noktasını (0,0) noktası kabul eder, A noktasına vardığında ise B noktasını referans alarak konumunun (-125,122) olduğunu hesaplar.



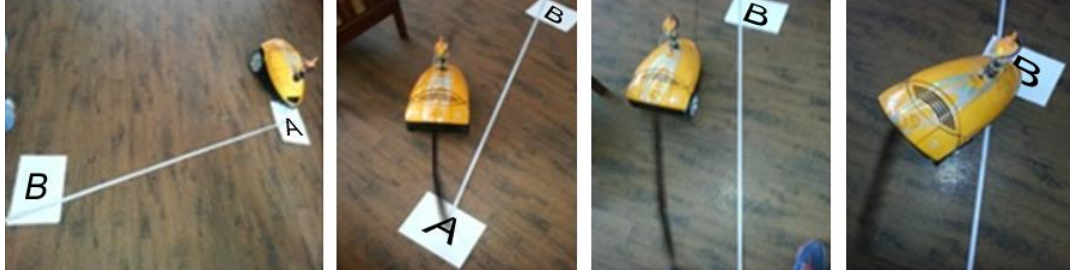
Şekil 4.4. Robotun B noktasından A noktasına geldiğinde bulunduğu konum.

4.2.2. Rota Takip Deneyleri

Deneylerde robota aynı oda içerisinde belli noktalara gitme görevleri verilmiştir. Çok yakın mesafelerde, örneğin 30 cm gibi bir uzaklıkta, robotun yönü hedefe direkt bakmadığı için, hedef noktaya robot varabilmek için manevra yapmak zorunda kalmış ve yolu 178 cm olarak kat etmiştir. Şekil 4.5'de robot bulunduğu A noktası konumundan 141 cm uzaklıkta açılı olarak duran B hedef noktasına gitme görevi verilmiş ve robotun hedefe yönelerek başarı ile vardığı görülmüştür. 50 cm den daha fazla olan hedeflerde ise maksimum %3 lik sapmalarla hedefe başarı ile ulaşmıştır.

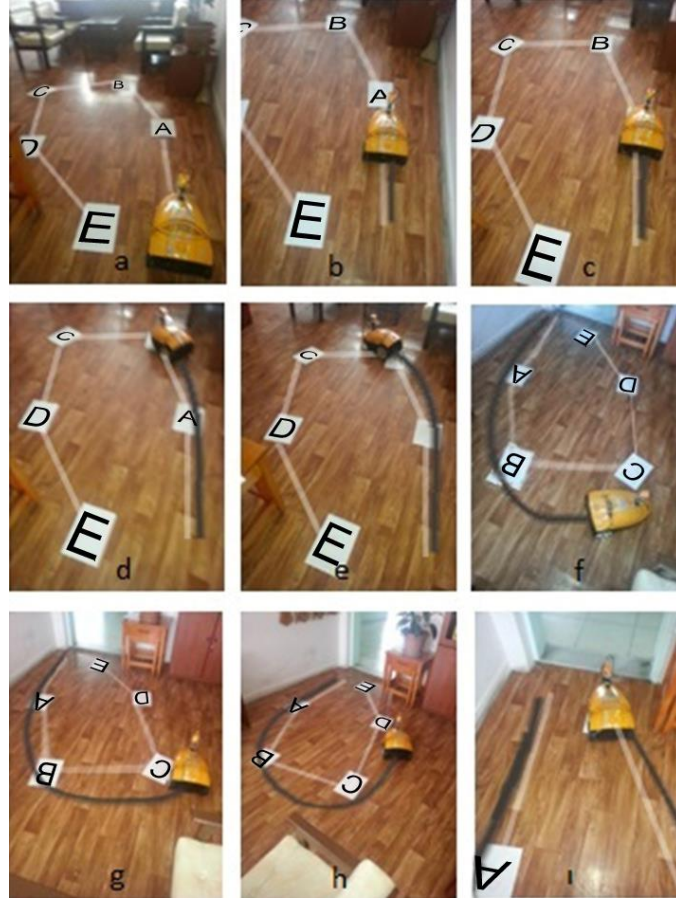
Bir sonraki deneyde Şekil 4.6'da gösterildiği gibi robota mevcut konumdan yeni bir hedef girilmiş, rota üzerindeki belirli noktaları takip etmesi istenmiştir. Robotun takip edebileceği en kısa yol beyaz renk ile çizilmiş, robotun gerçekte hareket ettiği güzergâh ise siyah renkle gösterilmiştir. Gözlem sonucunda herhangi bir sapmaya

uğramaksızın rota üzerindeki noktaları takip ederek hedef noktaya ulaştığı gözlemlenmiştir.

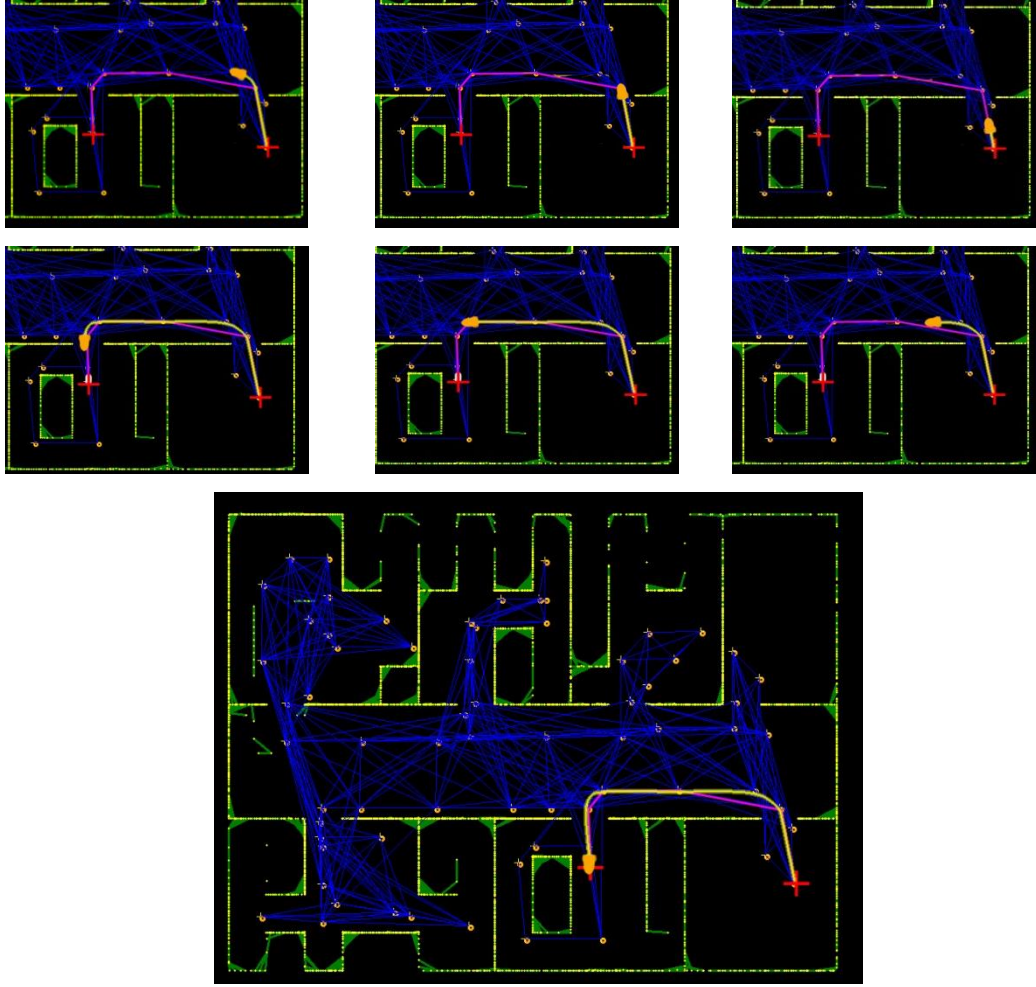


Şekil 4.5. Robotun iki nokta arasındaki seyrüsefer işlemi.

Simülasyon ortamında yapılan seyrüsefer denemelerinde de gerçek ortamdakine benzer sonuçlar alınmıştır. Şekil 4.7’de gösterilen örnek simülasyon ortamında, bir odadaki kaynak noktadan, hedefe ulaşırken robotun yaptığı manevralar gösterilmiştir.



Şekil 4.6. Robotun belirlenen rota üzerindeki seyrüsefer işlemi. Beyaz çizgiler en kısa rotayı gösterirken, siyah çizgi robotun izlediği rotayı göstermektedir.



Şekil 4.7. Simülasyon ortamında robotun hedefe varırken yapmış olduğu seyrüsefer işlemi.

BÖLÜM 5

SONUÇLAR

Bu tez çalışmasında gerçekleştirilen otonom kontrollü gezgin robot, bina içlerinde taşıma işlerinde kullanılmak üzere tasarlanmıştır. Robot bulunduğu binayı tanınması, istenen adreslere otonom olarak gitmesi, rotayı en kısa yol olacak şekilde planlaması sebebiyle, fabrikalarda forkliftlerde, bürolarda çay ve evrak taşımada, hastanelerde yemek dağıtım işlerinde ve buna benzer birçok kapalı mekana sahip ortamlarda kullanılabilir.

Robotun tasarımından elde edilen sonuçlar maddeler halinde şu şekilde tanımlanabilir.

- a) Robotun lazer mesafesi siyah zeminlerde 8 metreden daha fazla olduğunda, konum hesaplama hassasiyeti % 3 den daha fazla arttığı için kullanılacak mekanın en geniş açıklığı 8 metreyi geçmemelidir.
- b) Yön bulma bilgileri, DCM algoritması kullanılarak zeminin eğiminden veya dönme kaymalarından kaynaklı hatalar olmaksızın doğru bir şekilde alınması sağlanmıştır.
- c) Robotun bulunduğu ortamı öğrenmesi otonom ve manuel olarak gezdirilip yapılabilmektedir. Manuel olarak ortamın tanıtılması, oda içerisinde ulaşılamayan ölü noktaların kalmamasını sağladığından tercih edilmelidir.
- d) Robotun mekanik tasarımında ön tekerleklerde açisal dönmeyi sağlayan servo motor ile arka tekerdeki tek DC motora bağlı diferansiyel sistemi sayesinde, istenilen hedeflere sorunsuz ulaşabilen karmaşık hesaplara gerek duymayan bir mekanik sistem elde edilmiştir.

- e) Robotun arka iki tekerleğinde kullanılan enkoder ile de alınan yol hesaplanmıştır. Bu sayede iki veri ile alınan yol uzunluğu hesaplanmış olmaktadır. Özellikle bu işlem lazer metrenin ölçüm yapamadığı 10 metre üzeri düz yollarda mesafeyi kontrol etme olanağı oluşturmuştur.
- f) Robotun donanımsal teknolojisinde Lazer metre, ivmeölçer, jiroskop, manyetik sensör, enkoder gibi bir çok cihaz kullanılmıştır. Bu cihazlar bütünleşik hale getirilip ATMEL378 işlemci üzerindeki yazılım ile işlenmiş bluetooth teknolojisi ile bilgisayar yazılımına gönderilerek kontrol edilmiştir. Böylece komple bütünleşik bir sistem elde edilmiştir.
- g) Robot rota takibi esnasında belirlenen hedef noktalardan ortalama 5 metre açıklığa sahip bir oda içerisinde 5-10 cm arasında bir hata ile geçmektedir.
- h) Tasarlanan sistemde yön tayininde jiroskop, ivme ölçer, manyetik sensörden gelen pusula bilgisi, lazer metreden gelen mesafe bilgileri aynı anda kullanılarak konum hesabı daha kolay hale getirilmiştir.
- i) Tasarlanan sistemde belirlenen güzergahlar üzerinde robotun hareket edeceği yol bilgisi ise yine jiroskop, ivme ölçer, manyetik sensörden gelen pusula bilgisi, lazer metreden ve enkoder den kat edilen yol bilgisi sayesinde robot sadece yön ve aldığı yol bilgisini kullanarak seyrüseferini sorunsuz tamamlamaktadır.

Bunun yanı sıra geliştirilen robot, bazı alanlarda sınırlı kalmaktadır. Bu konular geliştirmeye açık olup, yapılabilecek geleceğe yönelik çalışmalarla birlikte şu şekilde özetlenebilir;

- a) Robotun mekanik tasarımı, ev ve benzeri alanlarda kullanılabilir yapıda tasarlanmıştır. Bu tasarım geliştirilerek arazi, fabrika vb. ortamlarda kullanılacak şekilde daha büyük ve güçlü yapıda GPS modülü de eklenerek tasarlanabilir.

b) Robotun kontrolü uzaktan sabit bir bilgisayar platformu kullanılarak yapılmıştır. Bu da robotun platforma bağımlı hale gelmesini sağlamakta. Bu sorunun çözümü için robotun üzerinde taşıdığı tabletler ile kontrol gerçekleştirilebilir.

KAYNAKLAR

1. Talluri, R., and Aggarwal, J. K., "Position Estimation Techniques For An Autonomous Mobile Robot-A Review", Handbook of Pattern Recognition and Computer Vision, *World Scientific Publishing Company*, Singapore, 765-796 (1998).
2. Barshan, B. and Durrant-Whyte, H. F., "Inertial sensing for mobile robotics", *IEEE Transactions on Robotics and Automation*, 11 (3): 328–342 (1995).
3. Hu, H., and Gu, D., "Landmark-based navigation of industrial mobile robots", *International Journal of Industry Robot*, 27 (6): 458–467 (2000).
4. Borenstain, J., "The CLAPPER: a dual-drive mobile robot with internal correction of dead-reckoning errors", *IEEE International Conference on Robotics and Automation*, San Diego, CA, 4: 3085-3090 (1994).
5. Kleeman, L., "Optimal estimation of position and heading for mobile robots using ultrasonic beacons and dead-reckoning", *IEEE International Conference on Robotics and Automation*, Nice, 3: 2582-2587 (1992).
6. Leonard J., and Durrant-Whyte, H. F., "Mobile robot localization by tracking geometric beacons", *IEEE Transactions on Robotics and Automation*, 7: 376-382 (1991).
7. Madsen, C. B., Andersen, C. S., "Optimal landmark selection for triangulation of robot position", *Journal of Robotics and Autonomous Systems*, 23 (4): 277-292 (1998).
8. Durrant-Whyte, H. F., "An autonomous guided vehicle for cargo handling applications", *International Journal of Robotics Research*, 15 (5): 407-440 (1996).
9. Elfes, A., "Sonar-based real-world mapping and navigation", *IEEE Journal of Robotics and Automation*, 3 (3): 249 - 265 (1987).
10. Thrun S. and Bucken, A., "Integrating grid-based and topological maps for mobile robot navigation", *Proc. of the 13th National Conference on AI*, Portland Oregon (1996).
11. Borenstein, J., Everett, H. R., and Feng, L., "Where am I? Sensors and Methods for Mobile Robot Positioning", *Research Report, The University of Michigan*, (1996).

12. Ayala, V. Hayet, J. B., Lerasle, F., and Devy, M., “Visual Localization of a mobile robot in indoor environments using planar landmarks”, *Proceedings of 2000 IEEE International Conference on Intelligent Robots and Systems*, Takamatsu, 1: 275-280 (2000).
13. Mata, M., Armingol, J. M., de la Escalera, A., and Salichs, M. A., “A visual landmark recognition system for topological navigation of mobile robots”, *Proceedings of 2001 IEEE International Conference on Robotics & Automation*, 2: 1124-1129 (2001).
14. Dellaert, F., Burgard, W., Fox, D., and Thrun, S., “Using the CONDENSATION algorithm for robust, vision-based mobile robot localization”, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Fort Collins, CO, 2: 588-594 (1999).
15. Murrieta-Cid, R., Briot, M., and Vandapel, N., “Landmark identification and tracking in natural environment”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Victoria, BC, 1: 179–184 (1998).
16. Sim, R., and Dudek, G., “Mobile robot localization from learned landmarks”, *IEEE/RSJ Conference on Intelligent Robots and Systems*, Victoria, BC, 2: 1060-1065 (1998).
17. Se, S., Lowe, D.G. and Little, J., “Vision-based global localization and mapping for mobile robots”, *IEEE Transactions on Robotics*, 21: 217-226 (2005).
18. Yoon, K. J., and Kweon, I. S., “Artificial landmark tracking based on the color histogram”, *International Conference on Intelligent Robots and Systems*, Maui, HI, 4: 1918-1923 (2001).
19. Jang, G., Kim, S., Lee, W., and Kweon, I., “Color landmark based self-localization for indoor mobile robots”, *IEEE International Conference on Robotics & Automation*, Washington, DC, USA, 1:1037-1042 (2002).
20. Ahn, S. J., Rauh, W., and Recknagel, M., “Circular coded landmark for optical 3d-measurement and robot vision”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Kyongju, 2: 1128 – 1133 (1999).
21. Kim, Y. G., An, J. and Lee, K. D., “Localization of Mobile Robot Based on Fusion of Artificial Landmark and RF TDOA distance under indoor sensor network”, *International Journal of Advanced Robotic Systems*, 8 (4): 203-211 (2011).
22. Deans, M. and Herbert, M., “Experimental comparison of techniques for localization and mapping using a bearing-only sensor”, *ISER'00 Seventh International Symposium on Experimental Robotics*, Honolulu, Hawaii. 395–404 (2000).

23. Sim, R., Dudek, G., “Mobile robot localization from learned landmarks”, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Victoria, BC, 2: 1060–1065 (1998).
24. Betke, M., and Gurdits, L., “Mobile robot localization using landmarks”, *IEEE Transactions on Robotics and Automation*, 13 (2): 251-263 (1997).
25. Goel, P., “Robust localization using relative and absolute position estimates”, *IEEE International Conference on Intelligent Robots and Systems*, Kyongju, 2: 1134-1140 (1999).
26. Kantor, G. A. and Singh, S., “Preliminary results in range-only localization and mapping”, *IEEE Conference on Robotics and Automation*, 2: 1818–1823 (2002).
27. Moerno, L., Armingol, J. M., Garrido, S., de la Escalera, A. and Salichs, M. A., “A genetic algorithm for mobile robot localization using ultrasonic sensors”, *Journal of Intelligent and Robotic Systems*, 34: 135-154 (2002).
28. Se, S., Lowe, D., and Little, J., “Vision-based mobile robot localization and mapping using scale-invariant features”, *IEEE International Conference on Robotics and Automation*, 2: 2051-2058 (2001).
29. Carbonaro, A., and Zingaretti, P., “Landmark matching in a varying environment”, *Advanced Mobile Robots Second EUROMICRO workshop on*, Brescia, 147-153 (1997).
30. Thrun, S., “Finding landmarks for mobile robot navigation”, *IEEE International Conference on Robotics and Automation*, Leuven, 2: 958–963 (1998).
31. Amigoni, F., Caglioti, V., and Galtarossa, U., “A mobile robot mapping system with an information-based exploration strategy”, *Proceedings of the First International Conference on Informatics in Control, Automation and Robotics*, Setubal, Portugal, 71-78 (2004).
32. Amigoni, F., and Caglioti, V., “An information-based exploration strategy for environment mapping with mobile robots, robotics and autonomous systems”, *Journal of Robotics and Autonomous Systems*, 58 (5): 684-699 (2000).
33. Sanchez, C., Soriano, A., Valles, M., Vendrell, E., and Valera, A., “GEMA: Geometrical matching analytical algorithm for fast mobile robots global self-localization”, *Journal of Robotics and Autonomous Systems*, 62: 855-863 (2014).
34. Hiremath, S. A., Heijden, G., Evert, F. K., Stein, A., and Braak, C. J. F., “Laser range finder model for autonomous navigation of a robot in a maize field using a particle filter”, *Computers and Electronics in Agriculture* 100: 41-50 (2014).

35. Nunez, P., Vazquez-Martin, R., Toro, J. C., Bandera, A., and Sandoval, F., “Natural landmark extraction for mobile robot navigation based on an adaptive curvature estimation”, *Journal of Robotics and Autonomous Systems*, 56: 247-264 (2008).
36. Thrun, S., Fox, D., and Burgard, W., “A real-time algorithm for mobile robot mapping with application to multi-robot and 3D mapping”, *Proceedings of the IEEE International Conference on Robotics and Automation*, San Francisco, CA, 1: 321–328 (2000).
37. Hahnel, D., Burgard, W., and Thrun, S., “Learning compact 3D models of indoor and outdoor environments with a mobile robot”, *Proceedings of the Fourth European Workshop on Advanced Mobile Robots*, Lund, Sweden, (2001).
38. Surmann, H., Nüchter, A., and Hertzberg, J., “An autonomous mobile robot with a 3D laser range finder for 3D exploration and digitalization of indoor environments”, *Journal of Robotics and Autonomous Systems*, 45: 181–198 (2003).
39. Basaca-Preciado, L. C., Sergiyenko, O. Y., Rodriguez-Quinonez, J. C., Garcia, X., Tyrsa, V. V., Rivas-Lopez, M., Hernandez-Balbuena, D., Mercorelli, P., Podrygalo, M., Gurko, A., Tabakova, I., and Starostenko, O., “Optical 3D laser measurement system for navigation of autonomous mobile robot”, *Optics and Lasers in Engineering*, 54: 159-169 (2014).
40. Ye, C., and Borenstein, J., “A novel filter for terrain mapping with laser rangefinders”, *IEEE Transactions on Robotics and Automation*, 20 (5): 913-921 (2004).
41. Cole, D. M., and Newman, P. M., “Using laser range data for 3D SLAM in outdoor environments”, *IEEE International Conference on Robotics and Automation*, Orlando USA, 1556–1563 (2006).
42. Asadi, E., and Bozorg, M., “A decentralized architecture for simultaneous localization and mapping”, *IEEE/ASME Transactions on Mechatronics*, 14 (1): 64-71 (2009).
43. Baglietto, M., Sgorbissa, A., Verda, D., and Zaccaria, R., “Human navigation and mapping with a 6 DOF IMU and a laser scanner”, *Journal of Robotics and Autonomous Systems*, 59 (12): 1060-1069 (2011).
44. Tungadi, F., and Kleeman, L., “Discovering and restoring changes in object positions using an autonomous robot with laser rangefinders”, *Journal of Robotics and Autonomous Systems*, 59 (6): 428-443 (2011).
45. Yoshida, K., and Hirose, S., “Laser triangulation range finder available under direct sunlight”, *IEEE International Conference on Robotics and Automation*, Philadelphia, PA, 3: 1702–1708 (1998).

46. Konolige, K., Augenbraun, J., Donaldson, N., Fiebig, C., and Shah, P., “A low-cost laser distance sensor”, *IEEE International Conference on Robotics and Automation*, Pasadena, CA, 3002–3008 (2008).
47. Negenborn, R., “Robot localization and Kalman filters on finding your position in a noisy world”, M. Sc. Thesis, *Utrecht University*, 156 (2003).
48. Xiaochuan, W., “Developing reactive controllers for mobile robots navigation in unknown environments using infrared range sensors”, M. Sc. Thesis, *Guelph University*, 156 (2004).
49. Jenkin, M., Milios, E., Jasiobedzki, P., Bains, N., and Tran, K., “Global Navigation for ARK.”, *IEEE/RSJ International Conference on Intelligent Robotics and Systems*, Yokohama, Japan, 3: 2165-2171 (1993).
50. Elfes, A., “Using occupancy grids for mobile perception and navigation”. *Journal of Computer*, 22 (6): 4657 (1989).
51. Alp, O., “Gezgin robotlarda eş anlı haritalama ve konum belirleme”, Yüksek Lisans Tezi, *Başkent Üniversitesi Fen Bilimleri Enstitüsü*, Ankara, 38-43 (2008).
52. Sezer, B., “İvme sensör tabanlı laboratuvar acil destek uyarı sistemi”, Yüksek Lisans Tezi, *Kocaeli Üniversitesi Fen Bilimleri Enstitüsü*, Kocaeli, 17-19 (2011).
53. Velagic, J., Osmic, N., Hodsic, F. and Siljak, H., “Outdoor navigation of a mobile robot using GPS and GPRS communication system”, *53. International Symposium*, Zadar, 173-177 (2011).
54. İnternet: Premerlani, W., and Bizard, P., “Direction Cosine Matrix IMU: Theory”, <http://gentlenav.googlecode.com/files/DCMDraft2.pdf> (2009).
55. Dorigo, M., and Gambardella, L. M., “Ant colonies for the traveling salesman problem”, *Biosystems*, 43 (2): 73-81 (1996).
56. Tüker, M., Ballı, S., ve Pembeci, İ., “Çok etmenli sistemlerde netlogo ile karınca kolonisi optimizasyonu”, *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi*, 19 (2): 88-96 (2013).

ÖZGEÇMİŞ

Ahmet ÖZEREN 1982 yılında Gaziantep’de doğdu. İlköğretimini ve Lise eğitimini yine aynı ilde tamamladı. 2004 Yılında Süleyman Demirel Üniversitesi Teknik Eğitim Fakültesi, Elektronik ve Bilgisayar Eğitimi Bölümünden iyi derece ile mezun oldu. 2004 yılında Isparta Uluborlu Endüstri Meslek Lisesinde Öğretmen olarak göreve başladı. 2005 yılında Yozgat Yerköy Cumhuriyet İlköğretim Okulunda, 2006 yılında Bartın Gürgen pınarı İlkokulunda görev yaptı. 2007 Yılından itibaren Bartın Teknik ve Endüstri Meslek Lisesinde görev yapmaktadır. 2010 yılında Karabük Üniversitesi, Teknik Eğitim Fakültesi, Elektronik ve Bilgisayar Eğitimi Anabilim Dalı’nda başladığı yüksek lisans eğitimini yine aynı üniversitede tamamladı.

ADRES BİLGİLERİ

Adres : Karabük Üniversitesi
Fen Bilimler Enstitüsü
Balıklar kayası Mevkii / KARABÜK

Tel : (505) 218 57 38

E-posta : ahmetozeren@hotmail.com