

ÖZGEÇMİŞ (AHMET ÇETİN)

16.01.1996 tarihinde KONYA'da doğdu. İlk ve orta öğrenimini KONYA'da tamamlayarak Mehmet Hanife Yapıcı Anadolu Lisesi'nden 2014 yılında mezun oldu. Aynı yıl üniversite sınavını kazanarak Selçuk Üniversitesi Mühendislik Fakültesi Elektrik– Elektronik Mühendisliği Bölümü'ne girdi. Halen bu bölümde öğrenimini sürdürmektedir.

ÖZGEÇMİŞ (NEVZAT BOL)

11.07.1996 tarihinde ANKARA'da doğdu. İlk ve orta öğrenimini ANKARA'da tamamlayarak Beypazarı Anadolu Lisesi'nden 2014 yılında mezun oldu. Aynı yıl üniversite sınavını kazanarak Selçuk Üniversitesi Mühendislik Elektrik–Elektronik Mühendisliği Bölümü'ne girdi. Halen bu bölümde öğrenimini sürdürmektedir.

TEŞEKKÜR

Bütün çalışmalarımızda değerli bilgi ve tecrübeleriyle bize yol gösteren, gerekli araştırma ve geliştirme çabalarımızda yardımlarını esirgemeyen danışmanımız Doktor Öğretim Üyesi Akif DURDU'ya ve bölümümüzün değerli tüm öğretim elemanlarına, her türlü maddi manevi katkılarını esirgemeyen ailemize ve öğrenci arkadaşlarımıza teşekkür ederiz.

ÖZET

Bu tez kapsamında geliştirilen Otonom Görev Robotu ile fabrika depo gibi ortamlarda yük taşıma işlemlerinin otonom yapılması sağlanmıştır. Robotun, hedef noktalara anlık oluşturulan güzergâhlar üzerinden otonom olarak yönlendirilmesi sağlanmıştır. Robotun bulunduğu ortamda otonom çalışabilmesi için eş zamanlı haritalandırma ve konumlandırma (simultaneous localization and mapping - SLAM) kullanılmıştır.

İlk olarak robotun 3 boyutlu modellemesi Solidworks programı ile yapılmıştır. Yapılan tasarıma göre prototip üretilerek mekanik ve elektronik aksamın montajı sağlanmıştır. Robotun kontrolü için bir mikroişlemci (asus tinkers board) ve bir mikrodenetleyici(arduino) kullanılmıştır. Mikroişlemci ve mikrodenetleyici arasında serial haberleşme protokolleri kullanılmıştır.

Robotun otonom olarak hedeflere yönlendirilmesi bilgisayar tabanlı arayüz üzerinden gerçekleştirilmektedir. Robotun verilen görevleri otonom olarak yapabilmesi için robot operating system (ROS) ve python kullanılmıştır. ROS ve python kullanılarak geliştirilen algoritma ile robotun çalıştığı ortamın haritası çıkarılmış çıkarılan harita üzerinde robotun konumlandırılması sağlanmış ve path planning algoritmaları ile robotun hedef noktalara ulaşması için güzergâh planlaması yapılmıştır. ROS tabanlı navigation algoritmaları kullanılarak robotun hedef noktaya yönlendirilmesi sağlanmıştır.

İÇİNDEKİLER

ÖZGEÇMİŞ, TEŞEKKÜR.....	i
-------------------------	---

ÖZET	ii
------------	----

İÇİNDEKİLER	iii
-------------------	-----

1.GİRİŞ	1
---------------	---

2.AGV SİSTEMLERİ.....	2
-----------------------	---

2.1 AGV Sistemi Nedir?	2
------------------------------	---

2.2 AGV Sistemleri Nasıl Çalışır?	2
---	---

2.2.1 AGV Sistemi Parçaları	3
-----------------------------------	---

2.2.1.1 Sistem Denetleyicisi	3
------------------------------------	---

2.2.1.2 Operatör Arayüzü	3
--------------------------------	---

2.2.1.3 I/O	3
-------------------	---

2.3 AGV Sistem Navigasyon Teknolojisi	3
---	---

2.3.1 Lazer Navigasyon	4
------------------------------	---

2.3.2 Doğal Navigasyon	4
------------------------------	---

2.3.3 Range Navigasyon	5
------------------------------	---

2.3.4 Spot Navigasyon	6
-----------------------------	---

2.3.5 Manyetik bantlı Navigasyon	7
--	---

2.3.6 İndüktif Kablolu Navigasyon	7
---	---

2.3.7 Multi Navigasyon	8
------------------------------	---

2.4 Agv Şarj Sistemleri	9
-------------------------------	---

2.4.1 Fırsata Bağlı Şarj	9
--------------------------------	---

2.4.2 Operatöre Bağlı Şarj	9
----------------------------------	---

2.4 Agv Neden İyi Bir Yatırımdır?.....	9
--	---

3. PROGRAMLARIN TANITIMI	10
--------------------------------	----

3.1 ROS	10
---------------	----

3.1.2 ROS Ne İşe Yarar?	10
-------------------------------	----

3.1.3 ROS Tarihçesi	10
---------------------------	----

3.1.4 ROS Nasıl Çalışır?.....	11
-------------------------------	----

3.1.5 ROS ve Komponentler	12
3.1.5.1 ROS Core	12
3.1.5.2 ROS Stacks & Packages	12
3.1.5.3 ROS Komut Satırı	13
3.1.5.4 ROS İletişim Yolları	13
3.1.6 ROS Alternatifleri	14
3.1.7 Neden ROS ?	15
3.2 OPENCV	16
3.2.1 OpenCV Bileşenleri	17
3.2.2 Alternatif Görüntü İşleme Kütüphaneleri	17
3.2.3 Raspberry Pi için OpenCV Kurulumu	18
3.2.4 Windows İşletim Sistemi İçin OpenCV Kurulumu	22
3.3 SolidWorks	23
3.3.1 Neden SolidWorks?	24
4 MALZEMELERİN TANITIMI	25
4.1 Raspberry Pi	25
4.1.1 Raspberry Pi Modelleri	27
4.1.2 Raspbian Kurulumu	28
4.1.3 Arduino İle Raspberry Pi Farkı	30
4.2 Arduino	31
4.2.1 Arduino Bileşenleri	31
4.2.2 Arduino Mega 2560	32
4.2.2.1 Arduino Mega 2560 Teknik Özellikler	34
4.2.2.2 Güç	34
4.2.2.3 Giriş Çıkışlar	35
4.2.2.4 Haberleşme	36
4.2.2.5 Programlama	36
4.2.2.6 USB Aşırı Akım Koruma Rölesi	36

4.3 Motorlar	37
4.3.1 DC Motorlar	37
4.3.1.1 DC Motorların Çalışma Prensibi	38
4.3.2 Redüktörlü Motorlar	40
4.3.3 Enkoderler	42
4.3.4 DC Motor Sürücüler	46
4.4 Lidar	49
4.4.1 Lidar Çalışma Prensibi	49
4.5 Lipo Pil	50
4.5.1 Lipo Pil Teknik Özellikleri	50
4.5.2 Lipo Pil Nasıl Şarj Edilir?	51
4.5.3 Lipo Pil Kullanımı	52
4.5.4 Lipo Pil Saklama Koşulları	53
5. GERÇEKLEŞTİRİLEN SİSTEMİN YAPISI VE YÖNTEMLER	54
5.1 Sistemin Mekanik Tasarımı	54
5.2 Yöntemler	56
5.2.1 SLAM Nedir?	56
5.2.2 SLAM Paradigmanları	57
5.2.2.1 Extended Kalman Filters (Genişletilmiş Kalman Filtresi)	57
5.2.2.2 Particle Filter (Parçacık Filtresi)	59
5.2.2.3 Graf Tabanlı Optimizasyon Teknikleri	60
5.3 Haritalandırma	61
5.3.1 Hector SLAM Algoritması ile Haritalama	62
5.3.2 Gmapping Algoritması ile Haritalama	62
6. SONUÇLAR	64
6.1 Sonuçların Tartışılması	64
6.2 Özsonuçlar ve Öneriler	64

KAYNAKLAR	65
EKLER	66
EK-1 ARDUİNO KODLARI	66
EK-2 HECTOR MAPİNG	82
EK-3 GMAPPİNG	83
EK-4 ROBOT MODELİ	85
EK-5 ROBOTUN SİMULASYON RESİMLERİ	91
EK-6 ROBOTUN GERÇEK RESİMLERİ	92