

**KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**ELEKTRONİK VE BİLGİSAYAR EĞİTİMİ
ANABİLİM DALI**

DOKTORA TEZİ

**GEZGİN ROBOTLARIN KONUM BELİRLEME VE ENGEL
SAKINIM PROBLEMİNİN TEK KARTLI BİLGİSAYAR
SİSTEMİ KULLANILARAK ÇÖZÜMÜ**

SERDAR SOLAK

KOCAELİ 2016

KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRONİK VE BİLGİSAYAR EĞİTİMİ
ANABİLİM DALI

DOKTORA TEZİ

GEZGİN ROBOTLARIN KONUM BELİRLEME VE ENGEL
SAKINIM PROBLEMİNİN TEK KARTLI BİLGİSAYAR
SİSTEMİ KULLANILARAK ÇÖZÜMÜ

SERDAR SOLAK

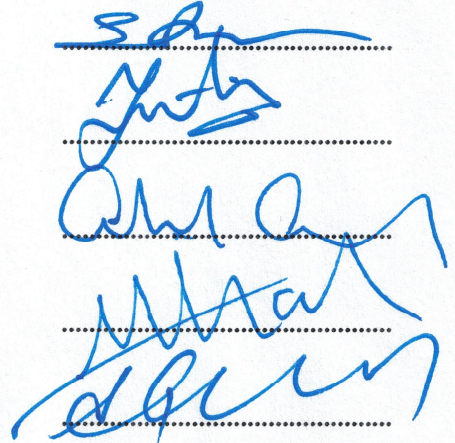
Doç.Dr. Emine Doğru BOLAT
Danışman, Kocaeli Üniv.

Prof.Dr. Mehmet YILDIRIM
Jüri Üyesi, Kocaeli Üniv.

Prof.Dr. Celal ÇEKEN
Jüri Üyesi, Sakarya Üniv.

Prof.Dr. M.Melih İNAL
Jüri Üyesi, Kocaeli Üniv.

Yrd.Doç.Dr. Ali ÇALHAN
Jüri Üyesi, Düzce Üniv.



Tezin Savunulduğu Tarih: 20.06.2016

ÖNSÖZ ve TEŞEKKÜR

Son yıllarda bilgisayarlı görme ve görüntü işleme tekniklerinin kullanılması akademik ve endüstriyel çalışmalarda yaygınlaşmıştır. Bu çalışmaların artmasıyla beraber, robotların uzaktan takip ve kontrol edilmesi, robotun bulunduğu ortamdaki sabit ve hareketli engelleri tespit ederek hareket etmesi, ortamın haritasının oluşturulması, robotun yol planlamasının yapılması gibi çeşitli uygulamalarda kullanılmaktadır. Yapılan tez çalışmasında, görüntü işleme teknikleri ve gömülü sistem kullanılarak, gezgin robotun engellere çarpmadan hedefe ulaşması sağlanmaktadır.

Tez çalışmalarım süresince her türlü desteği bana sunan, akademik çalışmalarımdaya benden desteğini esirgemeyen, değerli danışmanım Doç. Dr. Emine DOĞRU BOLAT'a, tez izleme jürisinde bulunup bana yol gösteren hocalarım Prof. Dr. Celal ÇEKEN ve Prof. Dr. Mehmet YILDIRIM'a, çalışmalarımdaya fikirleriyle bana destek olan, Prof. Dr. M. Melih İNAL ve Yrd. Doç. Dr. Umut ALTINIŞIK hocalarıma, yaptığım çalışma ile ilgili görüşlerini paylaşan değerli arkadaşlarım Okutman Uğur YILDIZ ve Önder YAKUT'a teşekkür ederim.

Tez çalışmalarım boyunca benden desteğini esirgemeyen ve sabırla destekleyen, sevgili eşim Feyza ve güzel kızım Melis'e, bugünlere gelmemde büyük emeği olan annem, rahmetli babam ve kardeşlerime teşekkür ederim.

Mayıs – 2016

Serdar SOLAK

İÇİNDEKİLER

ÖNSÖZ VE TEŞEKKÜR	i
İÇİNDEKİLER	ii
ŞEKİLLER DİZİNİ	iv
TABLolar DİZİNİ	vi
SİMGELER DİZİNİ VE KISALTMALAR	vii
ÖZET	viii
ABSTRACT	ix
GİRİŞ	1
1. GENEL BİLGİLER	5
1.1. Gezgin Robotların Kullanım Alanları	6
1.2. Gezgin Robotların Konum Belirleme Problemi	7
1.3. Gezgin Robotların Engel Sakınım Problemi	10
1.4. Gezgin Robotların Yol Planlama Problemi	14
1.5. Gezgin Robotlarda Bilgisayarlı Görme	20
1.5.1. Gömülü görme ve uygulama alanları	22
1.5.1.1. Gözetleme sistemleri	22
1.5.1.2. Otomotiv güvenliği	23
1.5.1.3. Video oyunları	23
1.5.1.4. Yüzme havuzu güvenliği	23
1.5.1.5. Görme tabanlı gezgin robotlar	24
1.5.2. Gömülü görmenin sistem seviyesinde problemleri	24
1.5.3. Gömülü görme işlemlerinde bilgisayarın rolü	25
2. KULLANILAN ARAÇLAR	26
2.1. Tek Kart Bilgisayarlar	26
2.1.1. BeagleBoard-xM	28
2.2. Angström İşletim Sistemi	29
2.2.1. Angström kurulacak olan mikroSD kartın hazırlanması	30
2.2.2. Angström dağıtımının mikroSD karta kurulması	32
2.3. OpenCV Kütüphanesi	34
2.4. Kameralar ve Yardımcı Bağlantı Araçları	35
2.5. Pioneer 3 – DX Gezgin Robotu	37
3. YÖNTEM	43
3.1. Kullanılan Görüntü İşleme Yöntemleri	43
3.1.1. İğne delik kamera perspektif modeli	43
3.1.2. Kamera kalibrasyonu	46
3.1.3. Stereo görme, epipolar geometri ve kalibrasyon işlemi	51
3.1.4. Görüntü ön işleme adımındaki kullanılan filtreler	58
3.1.4.1. Konvolüsyon işlemi	58
3.1.4.2. Gauss filtresi	59
3.1.5. HSV renk uzayı	60
3.1.6. Morfolojik işlemler	64
3.1.6.1. Genişleme işlemi	65
3.1.6.2. Aşındırma işlemi	65

3.1.6.3. Açma işlemi	66
3.1.6.4. Kapama işlemi.....	66
3.1.7. Görüntü eşikleme işlemi	67
3.1.8. Nesne ağırlık merkezlerinin bulunması	69
3.2. Kullanılan Yol Planlama Algoritması	72
3.2.1. Genetik algoritma tabanlı yol planlama algoritması.....	72
4. ÖNERİLEN DENEYSEL ÇALIŞMALAR VE YÖNTEMLER.....	80
4.1. Kapalı Ortam Gezgin Robotlar için Gömülü Hareket Planlama Sistemi	80
4.1.1. Görüntü işleme bloğu.....	82
4.1.2. Yol planlama bloğu.....	85
4.1.3. Robot hareket bloğu.....	86
4.1.4. Sistemin uygulanması ve sonuçları	87
4.2. Web Tabanlı Gezgin Robot Hareket ve Kontrol Sistemi	90
4.2.1. Web ara yüz modülü	93
4.2.1.1. Sanal engel sakınım ara yüzü	93
4.2.1.2. Gezgin robot kontrol ara yüzü	95
4.2.2. Gezgin robot kontrol modülü	97
4.2.2.1. Gezgin robot kontrol programı	98
4.2.2.2. Gezgin robot dönme açısının hesaplanması.....	100
4.2.3. Sistemin uygulanması ve sonuçları	105
4.3. Stereo Görme Tabanlı Gezgin Robot Hareket Planlama Sistemi	109
5. SONUÇ VE ÖNERİLER	118
KAYNAKLAR.....	122
KİŞİSEL YAYIN VE ESERLER.....	131
ÖZGEÇMİŞ.....	132

ŞEKİLLER DİZİNİ

Şekil 2.1. BeagleBoard-xM TKB	28
Şekil 2.2. MikroSD kart bilgileri	30
Şekil 2.3. MikroSD kart ve sabit disk bilgileri	31
Şekil 2.4. Angström işletim sistemi masaüstü görünümü	33
Şekil 2.5. Kameralar, (a) Kablosuz IP kamera (b) USB kamera	35
Şekil 2.6. P3-DX gezgin robotu.....	37
Şekil 2.7. P3-DX gezgin robotunun boyutları	38
Şekil 2.8. P3-DX gezgin robotu iç koordinat sistemi	41
Şekil 3.1. İğne delik KPM	44
Şekil 3.2. Radyal bozulma	47
Şekil 3.3. Ucuz web kamerasına ait yüzeysel bozulma	48
Şekil 3.4. Kalibrasyon işlemi sırasında Tsai ızgarası köşelerin bulunması	50
Şekil 3.5. Kalibrasyon işlemi sırasında önerilen algoritmaya ait akış şeması.....	51
Şekil 3.6. Stereo görme modeli	53
Şekil 3.7. Stereo kamera kalibrasyonu sırasında kullanılan görüntü çiftleri.....	55
Şekil 3.9. Görüntü düzeltme işlemi sonucu oluşan yeni görüntü	56
Şekil 3.10. Stereo kamera kalibrasyonu ve düzeltme süreci	57
Şekil 3.11. Stereo kamera kalibrasyonu akış şeması	58
Şekil 3.12. HSV renk uzayı konik gösterimi.....	61
Şekil 3.13. RGB-HSV renk uzayı dönüşüm görüntüleri.....	63
Şekil 3.14. GA, akış şeması.....	73
Şekil 3.15. GA için kullanılan çalışma ortamı	75
Şekil 3.16. Çalışmada kullanılan örnek bir kromozom.....	76
Şekil 3.17. Rulet çemberi ve dağılımı.....	78
Şekil 3.18. Çaprazlama İşlemi.....	79
Şekil 4.1. Gömülü hareket planlama sistemi.....	80
Şekil 4.2. Gömülü hareket planlama sistemi deney ortamı.....	81
Şekil 4.3. GİB çalışmasını gösteren blok şeması.....	83
Şekil 4.4. GİB iyileştirme (a) Kamera görüntüsü (b) İyileştirilmiş görüntü	84
Şekil 4.5. GİB, çıktısı ızgara tabanlı ortam matrisi	85
Şekil 4.6. YPB'nun çalışmasını gösteren blok şema	85
Şekil 4.7. RHB'nin çalışmasını gösteren blok şema	86
Şekil 4.8. A - B arası hareket.....	86
Şekil 4.9. Gezgin robotun gömülü hareket planlama sistemi.....	87
Şekil 4.10. Bilgisayar ortamından BeagleBoard-xM'e istek yapılması.....	88
Şekil 4.11. Bilgisayar ortamında BeagleBoard-xM'e ait ekran görüntüsü	89
Şekil 4.12. P3-DX ilerlemesi (a) Gerçek görüntü (b) Matris görüntüsü.....	89
Şekil 4.13. Terminal programından robotun ilerleyeceği yolun listelenmesi	90
Şekil 4.14. Web tabanlı gezgin robot laboratuvarı çalışmasının blok şeması.....	91
Şekil 4.15. Web tabanlı gezgin robot laboratuvarı çalışmasının modüler yapısı	92
Şekil 4.16. Web tabanlı gezgin robot laboratuvarı zaman çizelgesi.....	92
Şekil 4.17. Sanal engel sakınım web ara yüzü	94
Şekil 4.18. Gezgin robot kontrol ara yüzü	96

Şekil 4.19. Gezgin robot kontrol modülü	97
Şekil 4.20. Gezgin robot kontrol modülüne ait kontrol programı akış şeması	99
Şekil 4.21. Gezgin robot kontrol programının terminal ekranı görüntüsü	99
Şekil 4.22. P3-DX'e ait üstten görünüm	100
Şekil 4.23. Gezgin robotun HSV renk uzayındaki görüntüsü	101
Şekil 4.24. İkili görüntü (a) Sarı renk (b) Kırmızı renk	102
Şekil 4.25. Deney ortamında bulunan robotun tespit edilmesi	103
Şekil 4.26. Web tabanlı gezgin robot laboratuvar ortamı	105
Şekil 4.27. GA parametrelerinin ayarlanması	106
Şekil 4.28. Örnek deney ortamı	106
Şekil 4.29. Örnek deney sonuçları	107
Şekil 4.30. P3-DX'in kullanıcı tarafından hareket ettirilmesi	108
Şekil 4.31. P3-DX hareketinin görüntülenmesi ve sonuçların alınması	108
Şekil 4.32. Stereo görme tabanlı önerilen model	109
Şekil 4.33. Stereo görüntü çiftleri üzerinde nesne tespiti	110
Şekil 4.34. Nesne koordinatlarının bulunması	111
Şekil 4.35. Nesne yükseklik ve genişlik bilgileri	112
Şekil 4.36. Gerçek değerlere ait doğruluk yüzdesini gösteren grafik	114
Şekil 4.37. Hesaplanan değerlerdeki hata yüzdesini gösteren grafik	116
Şekil 4.38. Önerilen sistemin blok şeması	116

TABLULAR DİZİNİ

Tablo 2.1. BeagleBoard-xM'nin teknik özellikleri	29
Tablo 2.2. Kablosuz IP kameranın teknik özellikleri	36
Tablo 2.3. USB Kameranın teknik özellikleri	36
Tablo 2.4. P3-DX gezgin robotuna ait özellikler.....	38
Tablo 2.5. BeagleBoard-xM P3-Dx iletişim paketine ait bilgiler ve özellikleri	39
Tablo 2.6. İstemci tarafından gönderilen iletişim paket örneği.....	40
Tablo 2.7. Senkronizasyon paketlerine ait içerik bilgisi	41
Tablo 3.1. Kromozom uygunluk tablosu.....	78
Tablo 4.1. GA varsayılan değerleri.....	94
Tablo 4.2. Robotun dönüş açılarının hesaplanması	104
Tablo 4.3. Örnek ölçüm değerleri.....	113
Tablo 4.4. Robotun dikey yönde nesnelere olan mesafesi	115

SİMGELER DİZİNİ VE KISALTMALAR

mm	: Milimetre
nm	: Nanometre
Hz	: Hertz, (1/s)
ms	: Milisaniye, (10^{-3} s)
mV	: Milivolt, (10^{-3} V)
f	: Odak noktası
b	: Baseline (İki kamera arası mesafe)
V	: Volt
A	: Amper
θ	: Robot gerçek dönme açısı
θ_B	: Robot başlangıç açısı
θ_D	: Robotun geçici dönme açısı

Kısaltmalar

ARIA	: Advances Robotics Interface for Applications (Uygulamalar için Gelişmiş Robotik Arayüzü)
ARM	: Acorn RISC Machine (Acorn RISC Makinesi)
GA	: Genetik Algoritma
GİB	: Görüntü İşleme Bloğu
HSV	: Hue, Saturation, Value (Ton, Doygunluk, Değer)
IP	: Internet Protocol (İnternet Protokolü)
KPM	: Kamera Perspektif Modeli
OpenCV	: Open Source Computer Vision Library (Açık Kaynak Kodlu Bilgisayarlı Görme Kütüphanesi)
P3-DX	: Pioneer 3 – DX
RAM	: Random Access Memory (Rastgele Erişimli Bellek)
RGB	: Red, Green, Blue (Kırmızı, Yeşil ve Mavi Renk Uzayı)
RHB	: Robot Hareket Bloğu
ROS	: Robot Operating System (Robot İşletim Sistemi)
SBC	: Single Board Computer (Tek Kart Bilgisayar)
SIPs	: Server Information Packets (Sunucu Bilgi Paketleri)
SoC	: System on a Chip (Çip Üzerinde Sistem)
TCP/IP	: Transmit Control Protocol and Internet Protocol (İletim Denetim Protokolü / İnternet Protokolü)
TKB	: Tek Kart Bilgisayar
USB	: Universal Serial Bus (Evrensel Seri Veriyolu)
VNC	: Virtual Network Computing (Sanal Ağ Sistemi)
YPB	: Yol Planlama Bloğu

GEZGİN ROBOTLARIN KONUM BELİRLEME VE ENGEL SAKINIM PROBLEMİNİN TEK KARTLI BİLGİSAYAR SİSTEMİ KULLANILARAK ÇÖZÜMÜ

ÖZET

Yapılan tez çalışmasında, Tek Kart Bilgisayar (TKB) sistemleri kullanılarak gezgin robotların hareket ve kontrol edilmesine yönelik çalışmalar ve yöntemler sunulmaktadır. Çalışmalarda, gezgin robotun ortamda bulunan engeller ve hedefi bulması için görüntü işleme tekniklerinden faydalanılmaktadır. Ortamda bulunan nesnelere tespit edildikten sonra, ortamın ızgara tabanlı haritası çıkarılmaktadır. Çıkarılan harita üzerinde Genetik Algoritma (GA), uygulanarak robotun en kısa yoldan hedefe ulaşması sağlanmaktadır.

Tez çalışması kapsamında, gezgin robotların hareketi için üç bloktan oluşan bir sistem önerilmektedir. Görüntü işleme bloğu adı verilen ilk blok, ortamın ızgara tabanlı haritasını çıkarmakta, Yol planlama bloğu ismi verilen ikinci blok, en kısa yolu hesaplamakta ve Robot hareket bloğu adı verilen son blok ise robotun hareket etmesi için gerekli, açı, mesafe ve komut bilgilerini oluşturarak robotun hareket ettirilmesini sağlamaktadır.

Yapılan ikinci çalışmada, gezgin robotların uzaktan hareket ve kontrol ettirilmesi için, web tabanlı gezgin robot hareket ve kontrol sistemi önerilmektedir. Çalışmada, gerçek bir laboratuvar ortamında bulunan P3-DX gezgin robotunun, sanal ortamda bulunan engellere çarpmadan, sanal hedefe ulaşması sağlanmaktadır. Ayrıca P3-DX gezgin robotu, kullanıcı tarafından uygulanan hareket komutları ile yönünü ve konumunu değiştirebilmektedir.

Tez kapsamında yapılan üçüncü çalışmada, stereo görme kullanılarak mesafe kestirimi yapılmakta ve gezgin robotun ortamda bulunan engellere çarpmadan hedefe ulaşması sağlanmaktadır.

Laboratuvar ortamında hazır durumda bulunmasından dolayı P3-DX gezgin robotu kullanılmaktadır. Yapılan çalışmalarda, uygulama yazılımlarının kodlanmasında C, C++ ve PHP programlama dilleri, görüntü işleme uygulamaları için OpenCV kütüphanesi gibi açık kaynak kodlu yazılımlar kullanılmaktadır. Ayrıca robotun kontrol ve hareket ettirilmesinde gömülü sistem olarak BeagleBoard-xM kullanılmaktadır. Yapılan çalışmalar ve önerilen sistemler, küçük değişikliklerle farklı robotlara uygulanabilmektedir.

Anahtar Kelimeler: Gezgin Robotlar, Görüntü İşleme, Stereo Görme, Tek Kart Bilgisayar Sistemleri, Yol Planlama.

THE SOLUTION OF THE MOBILE ROBOTS' LOCALIZATION AND OBSTACLE AVOIDANCE PROBLEM USING SINGLE BOARD COMPUTER SYSTEM

ABSTRACT

In this thesis, the studies and methods for the motion and control of mobile robots are presented using Single Board Computer systems. In these studies, the image processing techniques are used for the detection of the objects. The grid-based map is created using detected objects. Then, the shortest path is obtained applying Genetic Algorithm to the generated map.

In the first study, a motion planning system including three blocks is proposed. The first block, image processing block, is used for generating a grid-based map of the environment. The second block, path planning block, is utilized for the shortest path. The last block, robot motion block, provides the motion of the robot.

In the second study, a web-based motion and control system is proposed for the robot. It is provided that the mobile robot navigates avoiding the virtual obstacles to reach the virtual target in the real laboratory environment. The motion of the robot can also be controlled by the users' commands.

In the last study, a stereo vision-based distance estimate is used for the robot's motion. The distance between the robot and obstacles is calculated to reach the target avoiding obstacles.

Since the P3-DX robot exists in the laboratory environment, it is used in all experimental studies. Open source software such as C, C++ and PHP languages and OpenCV library are used for the application program. For controlling and motion of the robot, BeagleBoard-xM is utilized as an embedded hardware. The realized and proposed systems can be applied to different robots with slight changes.

Keywords: Mobile Robot, Image Processing, Stereo Vision, Single Board Computer Systems, Path Planning.

GİRİŞ

Teknoloji alanındaki hızlı gelişmeler, günlük hayatta robot kullanımını yaygınlaştırmıştır. Özellikle endüstriyel ve akademik alanlarda robotlar üzerine yapılan çalışmalar hızlanmıştır. Bazı robotlar bir görevi yerine getirirken insanlardan yardım alırken, bazı robotlar da görevlerini insanlardan yardım almadan gerçekleştirmektedir. Robotların insanlardan yardım almadan görevi yerine getirme işlemi, robotun otonom özelliği olarak tanımlanmaktadır. Otonom robotlar, üzerinde bulunan lazer, sonar veya kamera gibi algılayıcılar sayesinde çevresini gözlemlemekte ve algılamaktadır. Otonom robotlar algılanan bu bilgileri kullanarak hareketlerine karar vermektedirler. Otonom robotların bazıları insan gibi yürüyen, bazıları hareket etmeden bulunduğu yerde çalışan, bazıları tekerlek veya palet kullanarak hareket eden, bazıları da uçan robotlardır [1].

Otonom gezgin robotlar askeri alanlarda, bilgi toplama, düşman sahasının gözetlenmesi, keşfedilmesi, tehlike yaratacak durumların tespit edilmesi gibi çalışmalarda kullanılmaktadır. Ayrıca insanlar için tehlike arz eden yerlere girilmesi, gözetlenmesi ve bilgilerin toplanması gibi işlemlerde de gezgin robotlar kullanılmaktadır. Bunların yanı sıra gezgin robotlar insanların ilaç, yemek ve yüklerinin taşınmasında, mayınlı arazilerin, zehirli kimyasal ve nükleer maddelerin temizlenmesinde, yüksek gerilim hatları ile ilgili çalışmaların yapılmasında kullanılabilir. Deprem, sel, yangın gibi afetlerde insanların yerlerini tespit etmek, bulmak ve kurtarmak için de otonom robotlar kullanılmaktadır [2].

Gezgin robotların otonom özelliği, üzerinde bulunan algılayıcıları kullanması ile mümkün olmaktadır. Gezgin robotların üzerinde algılayıcıların kullanılmasında, canlıların çevreyi, yaşadıkları ortamı algılama biçimlerinden esinlenilmiştir. Çoğu canlının yaşadığı çevreyi temel duyu organlarından biri olan gözleri ile algılamalarına benzer bir şekilde gezgin robotlar da ortamı, ortamdaki nesnelere, nesnelere konumlarını, gezgin robota olan uzaklıklarını, birbirlerine olan uzaklıklarını algılamak için kamera veya kameralar kullanmaktadırlar.

Bilgisayarlı görme ve görüntü işleme teknikleri son yıllarda çeşitli alanlarda yaygın olarak kullanılmaya başlanmıştır. Özellikle, tarım, tıp, biyomedikal, askeri, sanayi ve güvenlik alanlarında çok sayıda proje yapılmıştır. Ayrıca, bilgisayarlı görme, gezgin robotların kontrol ve hareket ettirilmesinde de kullanılmaya başlanmıştır. Kameralardan alınan görüntüler, görüntü işleme teknikleri kullanılarak işlenmekte ve ortamda bulunan nesnelere belirli özelliklerine göre tespit edilmektedir. Bu sayede ortam ve ortamda bulunan nesnelere ilgili konum, boyut, yer, birbirlerine ve kameralara olan uzaklıkları gibi birçok bilgi elde edilmektedir. Elde edilen bu bilgilere göre gezgin robotun deneysel ortamda hareketi sağlanmaktadır.

Gezgin robotlar ile ilgili son yıllarda akademik alanda çok sayıda tez, proje, araştırma ve makale çalışması yapılmaktadır. Bu çalışmalar incelendiğinde, gezgin robotların en büyük problemlerinin konum belirleme, engelden sakınma ve yol planlama olduğu görülmektedir.

Yapılan tez çalışmasında, tek kart bilgisayar sistemi ve stereo görme kullanılarak mesafe kestirim işlemi gerçekleştirilmekte, ayrıca gezgin robotun engel sakınım ve yol planlama problemine çözüm sunulmaktadır. [3] numaralı kaynakta verilen 2013 yılında yapılmış olan doktora tezinde kullanılan tavan kamerası devreden çıkarılmış, robot üzerine yerleştirilen iki adet USB kamera ile stereo görme kullanılarak gezgin robotun hareketi sağlanmıştır. Ayrıca, kullanılan bu kameralar gezgin robota tek kartlı bilgisayar sistemi ile bağlanarak robotun otonom olarak gezintisi sağlanmıştır. Gezgin robotun yol planlaması için Genetik Algoritma (GA) kullanılmıştır. Önerilen tez çalışmasında, açık kaynak donanım ve yazılım kullanıldığından düşük maliyetli bir sistem önerilmiştir.

Deneysel çalışmalarda, gezgin robot olarak Pioneer 3-DX kullanılmıştır. Tez çalışması kapsamında önerilen yöntemler, iletişim protokollerindeki küçük değişikliklerle her türlü gezgin robota uyarlanacak şekilde hazırlanmıştır.

Yapılan tez çalışmasının amacı, özgün katkısı ve yapılan çalışmalar aşağıda maddeler halinde özetlenmiştir.

- Çalışmanın ilk aşamasında literatür taraması yapılmış, engel sakınım, yol planlaması ve görüntü işleme konuları incelenmiştir.

- Yapılan çalışmalarda, gömülü sistem kullanılmıştır. Gömülü sistem olarak BeagleBoard-xM tek kart bilgisayar sistemi kullanılmış ve hazırlanan algoritmalar, Pioneer 3-DX gezgin robotuna seri iletişim protokolü üzerinden gönderilmiş ve robotun hareketi sağlanmıştır.
- Yapılan ilk çalışmada, robotun hareket edeceği ortamının haritası, kablosuz IP kamera ile alınmış, ortamda bulunan robot, hedef ve engeller görüntü işleme teknikleri kullanılarak oluşturulmuştur. Oluşturulan harita üzerinde GA kullanılarak gezgin robotun, hedefe en kısa yoldan ulaşması sağlanmıştır. GA, optimizasyon problemlerinde kullanılan bir teknik olmasına rağmen gezgin robotun yol planlama probleminde göstermiş olduğu başarımlar ve çözüm süresi olarak iyi sonuçlar vermiştir. Ayrıca, GA çıktı olarak, çözüm kümesi ürettiğinden alternatif iyi çözümlerde sunmaktadır. Ortamın ızgara tabanlı haritası küçük boyutlarda olmasından dolayı, dört veya beş adımda iyi çözümler üretilmektedir.
- Gezgin robotlar için 3 bloktan oluşan düşük maliyetli gezinti sistemi önerilmiştir. Önerilen sistemde, görüntü işleme bloğu, yol planlama bloğu ve robot hareket bloğu bulunmaktadır. Önerilen bu sistem, üzerindeki herhangi bir blokta yapılacak küçük değişikliklerle farklı robotlara, farklı algoritmalar uygulanabilmektedir.
- Laboratuvar ortamında bulunan gezgin robotun, internet yada web sitesi üzerinden hareket ettirilmesi ve izlenmesi için deneysel ortam oluşturulmuştur. Web sayfası üzerinde yer alan gerçek laboratuvar görüntüsü üzerine sanal engeller ve hedef yerleştirilmektedir. Gerçek ortamda yer alan gezgin robot, bu engelleri ve hedefi algılayarak en kısa yoldan hedefe ulaşmaktadır. Ayrıca gezgin robot, web sitesi kullanılarak kontrol edilebilmektedir.
- Önerilen laboratuvar çalışmasında gezgin robotun hedefe doğru gideceği yön, açı ve konum tavan kamerasından alınan görüntü kullanılarak ve görüntü işleme tekniklerinden faydalanılarak en kısa ve en az enerjiyi harcayacak şekilde bulunmaktadır.
- Gezgin robot uygulamaları için, robotun uzaktan kontrol edilebilmesi ve kablosuz olarak hareket ettirilmesi için test ortamı tasarlanmıştır.
- Gezgin robotun hareket edeceği ortamda yer alan nesnelere koordinatlarını ve boyutlarını, kamera ve görüntü işleme teknikleri kullanarak tespit eden bir yöntem önerilmiştir.

- Stereo görme kullanılarak mesafe kestirimi yapan bir yöntem önerilmiştir. Önerilen yöntem gezgin robot üzerinde uygulanmış ve sonuçlar elde edilmiştir. Ayrıca, önerilen bu sistem kullanılarak ortamın haritası çıkarılmakta, robotun engellere ve hedefe olan mesafeleri hesaplatılmakta ve en kısa yoldan robotun hedefe ulaşması sağlanmaktadır.

Bu tez çalışması kapsamında Bölüm 1’de gezgin robotlar ile ilgili çeşitli tanımlamalar, bilgisayarlı görme ve görüntü işleme teknikleri kullanılarak yapılmış çalışmalar, engel sakınım, yol planlama yöntemleri anlatılmaktadır.

Bölüm 2’de tez çalışmasında kullanılan donanımsal ve yazılımsal araçlar anlatılmaktadır.

Bölüm 3’de çalışmada kullanılan görüntü işleme teknikleri ve yol planlama yöntemleri anlatılmaktadır.

Bölüm 4’de, tez çalışması kapsamında, önerilen çalışmalara ait yöntemler, algoritmalar ve yapılan deneysel çalışmalar sunulmaktadır.

Bölüm 5’te tez çalışması kapsamında elde edilen sonuç ve öneriler yer almaktadır.

1. GENEL BİLGİLER

Robotlar, çeşitli algılayıcılar kullanarak bulunduğu ortam ve nesnelere ile haberleşen, kendisine verilen görevleri yerine getirebilmek için yapay zekâ teknikleri gibi yöntemleri kullanan ve eyleme geçirebilen, mekanik cihazlardır. Robotlar genellikle, bilgisayar veya gömülü mimariye sahip bir cihaz tarafından kontrol edilebilen kollar, bacaklardan, tekerlek, paletler veya pervanelerle hareket edebilen elemanlardan oluşmaktadır [4].

Robotlarda gezginlik kavramı, gerçek dünyadaki yerlerini değiştirme yeteneğinin olması veya olmaması olarak tanımlanabilir. Robotlar, buldukları ortamdaki yer değiştirme hareketlerini üzerinde bulunan lazer, sonar ve kamera gibi algılayıcılar sayesinde yapmaktadır.

Son yıllarda akademik ve endüstriyel çalışmalarda bilgisayarlı görme ve görüntü işleme tekniklerinin kullanılması yaygınlaşmıştır. Bu çalışmaların artmasıyla beraber, robotların uzaktan takip ve kontrol edilmesi, robotun bulunduğu ortamdaki sabit ve hareketli engelleri tespit ederek hareket etmesi, ortamın haritasının oluşturulması, robotun yol planlamasının yapılması gibi çeşitli uygulamalarda kullanılmaktadır. Kısacası robotlarda kamera ve görüntü işleme tekniklerinin yaygın kullanılması robotların gezginlik özelliğini arttırmıştır.

Gezgin robotların kendine verilen görevleri yerine getirebilmesi için bulunduğu ortamı bilmesi ve öğrenmesi gerekmektedir. Kısacası gezgin robotun kendi konumunu bilmesi gerekmektedir. Gezgin robotların gezinmesi sırasında karşılaşılan bir başka problem ise ortamda bulunan engellerden kaçınmasıdır. Gezgin robotların bulunduğu ortamda gezebilmesi için “Nerdeyim?”, “Nereye gideceğim?” ve “Nasıl gideceğim?” sorularına cevap bulması gerekmektedir [5 - 7]. Bu sorulardan “Nerdeyim?” sorusunun cevabı için konum belirleme problemine çözüm aranmaktadır. “Nereye gideceğim?”, sorusu için hedef belirleme ve “Nasıl gideceğim?” sorusu için yol planlaması problemlerine çözüm aranmaktadır.

Alt bölümlerde, gezgin robotların kullanım alanları, konum belirleme, engel sakınım, yol planlaması ve gezgin robotlar da bilgisayarlı görme işlemi ile ilgili genel bilgiler ve literatür çalışmaları sunulmaktadır.

1.1. Gezgin Robotların Kullanım Alanları

Gezgin robotlar genel kullanım alanlarına göre incelendiğinde, karasal alanda hareket eden gezgin robotlar, uçan gezgin robotlar, su altında veya su üstünde hareket eden gezgin robotlar ve uzay çalışmalarında kullanılan gezgin robotlar olmak üzere 4 farklı grupta çalışmalar yapılmaktadır [8, 9]. Karasal alanlarda hareket eden gezgin robotlar, iç ve dış mekânlardaki hareket kabiliyetlerine göre iki grupta toplanmıştır [10].

Gezgin robotlar iç mekânlarda çok amaçlı olarak kullanılmaktadır. Özellikle, kapalı alanlarda gaz kaçağlarının tespit edilmesi [11], hastanelerde hemşire veya hemşir olarak hastalara ilaç dağıtılmasında [12], dökümhane gibi kapalı alanlarda çok sıcak ve sıvı haldeki metallerin taşınmasında, insan sağlığı için zararlı bir ortamda çalışmaların yapılmasında, insanların giremeyeceği, ulaşamayacağı yerlerde veri toplanmasında, kapalı ortamda bulunan maden ocağı çalışmalarında kullanılmaktadır.

Gezgin robotlar dış mekân çalışmalarında da yaygın olarak kullanılmaktadır. Doğal afet sonucunda oluşan enkaz içerisinde arama kurtarma çalışmalarında sıklıkla yararlanılmaktadır. Gezgin robotlar tarım alanında ürün hasat, ilaçlama, zararlı böceklerin toplanması, tarım alanında yer alan belli başlı ürünlerin bulunarak taşınması [13], tarım alanında yer alan sera içinde sulama, ilaçlama, ürün toplama gibi çeşitli amaçlarla kullanılmaktadır. Askeri alanlarda, mayınlı arazilerin temizlenmesi, düşman kara sahalarının izlenmesi, takip edilmesi ve verilerinin toplanmasında kullanılmaktadır.

Uçan gezgin robotlar, genel olarak insansız hava aracı (Unmanned Aerial) olarak adlandırılmaktadır. Yaygın olarak askeri alanlarda düşman sahasını bombalama, istihbarat, lojistik ve keşif amacıyla kullanılmaktadır [14].

Su altında ve su üstünde kullanılan gezgin robotlar, su altında kullanıldıklarında deniz altında keşif yapmak ve batık gemileri aramak için kullanılmaktadır.

Uzay çalışmalarında kullanılan gezgin robotlar, uzay alanında su ve buz gibi insan için önem arz eden arařtırmalar yapmak, toprak ve ortamda bulunan malzemelerden numuneler almak, kısacası keřif yapmak ve ortamı tanımak amacıyla kullanılmaktadır.

Gezgin robotlar hangi alanda kullanılırsa kullanılsın bulunduđu noktadan, hedefe ulaşabilmesi için, ortamı tanınması, engelleri ve hedefi bulması, bulmuş olduđu engellerden sakınması, ulaşması gereken hedefe giden yolun planlamasını yapabilmesi gerekmektedir. Aynı zamanda robotun bulunduđu ortamı tanınması, robotun konumunun bulunması için önem arz etmektedir.

1.2. Gezgin Robotların Konum Belirleme Problemi

Gezgin robotun bulunduđu uygulama ortamında konum belirleme problemi, robot çalışmalarında oldukça önemlidir. Gezgin robotun konumunun bilinmesi, robotun daha sonraki hareketinin bulunması ve robota iletilmesi için gereklidir. Gezgin robot çalışma ortamında konumunu bilmiyorsa bir sonraki adımda ne yapacağını ve nereye gideceğine karar vermesi mümkün olamaz. Literatürde yapılan çođu çalışmaya göre gezgin robotun otonom olabilmesi için en önemli özellik, bulunduđu ortamda konumunu bilmesidir [15].

Gezgin robot, bulunduđu ortamda başlama durumunu bilmediğinde konumunu belirlemesi için uğraşması gerekir. Bu durumda gezgin robot bulunduđu ortamdaki konum bilgisi hakkında hatalı bilgiler kullanabilir. Konum belirleme ile ilgili, başlangıç konumu bilinen robotun konumu, başlangıç konumu bilinmeyen robotun konumu ve kaçırılmış bir robotun konumu gibi çalışmalar yapılmaktadır [16].

Gezgin robot konum belirleme problemine ait literatürde çok sayıda çalışma bulunmaktadır. Son yıllarda yapılan bazı çalışmalar aşağıda özetlenmiştir.

2016 yılında, Costa ve diğ. [17] yapılan çalışmada Lazer ve kamera kullanarak gezgin robotun platformlarında kullanılmak üzere konum belirleme işlemi için modüler yapıda yeni bir yöntem önermişlerdir. Önermiş oldukları yöntem özellik eşleştirmeye dayalı bir sistem olup lazer ve kameralardan aldıkları verileri kullanarak gezgin robotun konumunu belirlemek aynı zamanda ortamın haritasını çıkarmaktadır.

Ayrıca, önerdikleri sistemi 3 ayrı gezgin robot platform sistemi üzerinde test etme imkânı bulmuşlardır.

2016 yılında, Caremes ve diğ. [18] yapmış oldukları çalışmada, kapalı ortam gezgin robot uygulamaları için konum belirleme problemine Coğrafi Bilgi Sistemi (CBS) kullanarak yeni bir yöntem önermektedirler. Önerilen sistem, görsel veriler ve iki boyutlu lazer sistemi kullanarak kapalı ortamlarda yer alan kapıları tanımaya dayalı bir sistemdir. Ayrıca yapılan çalışmada, konum belirleme işlemi için kapı tanıma işleminin yanı sıra CBS ile beraber genişletilmiş kalman filtresi kullanılmaktadır.

2015 yılında, Shim ve Cho [19, 20] yapmış oldukları iki farklı çalışmada gezgin robotun kapalı ortamda konum belirleme problemi için çözüm önermişlerdir. Önermiş oldukları çözümde harici güvenlik kameraları kullanmış ve görüntü işleme tekniklerinden faydalanmışlardır. Yapmış oldukları her iki çalışmada da kameradan aldıkları görüntüleri HSV (Hue- Saturation- Value) renk uzayına çevirerek elde ettikleri görüntüler üzerinde çalışmışlardır. Önermiş oldukları konum belirleme probleminde, tavana yerleştirmiş oldukları güvenlik kamerasını kullanarak elde ettikleri HSV görüntülerde hızlı gölge çıkarım algoritması kullanmışlardır. Ayrıca ikinci çalışmaları tavana yerleştirilen iki güvenlik kamerası ile çalışma ortamının ızgara tabanlı ve iki boyutlu haritasını çıkarmışlardır.

2015 yılında Gu ve Chen [21] tarafından yapılan çalışmada, kapalı ortam gezgin robotlar için, kamera kullanarak iki boyutlu konum belirleme yöntemi önerilmektedir.

2015 yılında, Basit ve diğ. [22] tarafından yapılan çalışmada, tek kamera kullanarak gezgin robotun hedef izleme ve konum belirleme işlemi için bir yöntem önermişlerdir. Hedefin ortamdaki ayrıldığı durumda tekrar hedefi bulması için uyarlanabilir histogram eşik değeri tanımlanmıştır. Görüntüyü parçalara ayırarak histogram eşleştirme yöntemi uygulamışlardır. Ayrıca önerdikleri yöntemi, insansız küçük tekerlekli bir araç oluşturarak deneme imkânı bulmuşlardır.

2015 yılında, Chong ve diğ. [23] tarafından yapılan çalışmada, gerçek zamanlı konum belirleme ve haritalama yöntemlerinde kullanılan algılayıcı teknolojileri üzerine araştırma çalışması yapmışlardır. Kameranın bu çalışmalardaki avantajlarını ortaya koymaya çalışmışlardır.

2015 yılında, Fidalgo ve Ortiz [24] tarafından yapılan çalışmada, görme tabanlı haritalama ve konum belirleme yöntemleri üzerine araştırma yapılmıştır. Bu çalışmada, son yıllarda kamera kullanılarak yapılan çalışmalarda elde edilen bilgiler, karşılaştırmalar, sınıflandırılmalar, avantaj ve dezavantajlar sunulmaktadır.

2013 yılında, Mirkhoni ve diğ. [25] tarafından yapılan çalışmada gezgin robotun konum belirleme problemine yönelik yeni bir yöntem hazırlamışlardır. Yapılan bu çalışmada Harmony arama algoritmasına dayalı ve tarama eşleştirme yöntemi kullanılmaktadır.

2012 yılında, Hou ve diğ. [26] tarafından yapılan çalışmada, küre şeklindeki gezgin robotun, gezintisi ve konum belirlemesi için görsel verilere dayalı bir yöntem önermişlerdir. Gezinti ve konum belirleme problemi için, görsel veriler ve coğrafi bilgi sistemi verileri kullanılmıştır. Ayrıca görüntülerin tanımlanmasında parçacık analizi yöntemi kullanılarak verimlilik artırılmıştır.

2012 yılında, Kar [27] tarafından yapılmış olan çalışmada, gezgin robotun genel konum belirleme problemi için özellik çıkarım tabanlı imza eşleştirmeye dayalı bir yöntem önerilmektedir. Kullanılan imza eşleştirme yönteminde imzanın özellikleri, geometrik özellikler, kısıtlar, ortama bağlı kısıtlar tarafından belirlenmektedir. Önermiş olduğu yöntemin çok verimli olduğunu iddia etmiştir.

2011 yılında, Baca ve diğ. [28] tarafından yapılan çalışmada, gezgin robotlar için kapalı ortamda denge histogram özelliği kullanılarak, görme tabanlı haritalama ve konum belirleme yöntemi önerilmektedir. Yapılan çalışmada insan bellek modeline benzeyen bir yöntem kullanılmaya çalışılmıştır.

2011 yılında, Habibov [29] tarafından yapılan tez çalışmasında, gezgin robotlar için eş zamanlı konum belirleme ve haritalama üzerine bir çalışma yapılmıştır. Yapılan çalışmada, hızlı eşzamanlı konumlama ve haritalandırma (FastSLAM) algoritması ve kızılötesi algılayıcılar kullanılmıştır. Önerilen yöntem benzetim ortamında uygulanmıştır.

2010 yılında, Ataş [30] tarafından yapılan tez çalışmasında, gezgin robotlar için konum belirleme ve harita oluşturma işlemleri için iyileştirme çalışmaları yapılmıştır.

Ayrıca kör konum belirleme özelliğine sahip gezgin robotlar için parçacık filtresi uygulanmış ve sonuçlarının uygun olmadığı savunulmuştur.

2010 yılında, Loevsky ve Shimshoni [31] tarafından yapılan çalışmada, kapalı ortam gezgin robotlar ve insansız kara araçları için, güçlü ve verimli bir konum belirleme yöntemi önerilmektedir.

Sonuç olarak, gezgin robotlar için konum belirleme problemi için yapılan literatür çalışmasında, çok sayıda çalışma olduğu gözlenmiş ve bunlar arasında özellikle algılayıcı olarak kamera kullananlar seçilmiştir. Genellikle gezgin robotun konum belirleme işlemi, gerçek zamanlı konum belirleme ve harita oluşturma (Simultaneous Localization and Mapping –SLAM) uygulamalarında, görsel izleme ve üç boyutlu yapılandırma görevlerinde kullanılmaktadır. Ayrıca, konum belirleme işleminde genişletilmiş kalman filtreleri, parçalı filtreler ve ızgara tabanlı yöntemlerin yaygın olarak kullanıldığı gözlenmiştir. Bunların yanı sıra, odometri verileri ve ölçüm bilgileri kullanılması konum belirleme yöntemlerinde önemli rol oynamaktadır.

1.3. Gezgin Robotların Engel Sakınım Problemi

Gezgin robotlar için önemli problemlerden biride engel sakınım işlemidir. Robotların bulunduğu ortamda, sabit ve hareketli olarak yer alan engeller arasında, engellere çarpmadan gezinmesi işlemi “engel sakınım” problemi olarak bilinmektedir. Robotun engelleri görmesi ve tespit etmesi, tanınması, bulması ve engeli aşması üzerinde yer alan algılayıcılar sayesinde olmaktadır. Kamera, lazer, sonar vb. algılayıcılar gezgin robotun üzerine bağlanmakta veya bu algılayıcılar robot ile haberleştirilerek kullanılmaktadır.

Yapılan tez çalışmasında, algılayıcı olarak kamera kullanıldığı için literatür çalışmasında kamera ile yapılan engel sakınım çalışmaları üzerine yoğunlaşmıştır. Son yıllarda yapılan bazı engel sakınım ve gezinim çalışmaları aşağıda özetlenmektedir.

2016 yılında, Mekonnen ve diğ. [32] tarafından yapılan çalışmada gezgin robotun dolaşması için görsel kontrole dayalı hibrit bir yöntem önerilmiştir. Algoritma görsel pozisyon tabanlı ve görüntü tabanlı kontrol içermektedir. Önerilen yöntemde, gezgin robota ait gerçek pozisyon bilgisi gerekmektedir.

Ayrıca çıkış sinyallerinin doğrusallaştırılması için yapay sinir ağı ve genişletilmiş kalman filtresi kullanılmıştır.

2015 yılında, Sharma ve Chitaliya [33] tarafından yapılan çalışmada, stereo görme kullanarak engelden sakınma üzerine araştırma çalışması yapılmıştır. Çalışmada stereo görme prensiplerinden ve kullanılan görüntü eşleştirme tekniklerinden bahsedilmiştir. Literatürde yaygın olarak, yoğun farklılık algoritması, mutlak farkların toplamı, pencere tabanlı yöntem, renk bölümlenmesi, enerji tabanlı, özellik tarama yöntemi, hızlı ve güçlü özellik tarama yöntemleri kullanılmaktadır. Çalışmada bu yöntemler hakkında çeşitli bilgiler sunulmaktadır.

2015 yılında, Fava ve diğ. [34] tarafından yapılan çalışmada, gezgin robotun gezintisi için görme tabanlı bir yöntem önerilmektedir. Geliştirmiş oldukları yöntemi benzetim olarak, kapalı ortamda gerçek bir gezgin robot platformunda ve bir şehir parkında denemişlerdir. Önerilen yöntemde yazılımsal iki katmanlı bir mimari önermiş olup yüksek seviyeli bir katman ile düşük seviyeli bir katmanı birleştirmişlerdir. Yüksek seviyeli katmanda, Red- Green - Blue (RGB) görüntüleri kullanarak yol, insan, nesne tanıma, bulma algılama işlemleri gerçekleştirmişlerdir. Düşük seviyeli katmanda ise, engelden sakınma ve gezinti işlemi yapılmaktadır.

2015 yılında, Al-Mutib ve diğ.[35] tarafından yapılan çalışmada, gezgin robotlar için stereo görme tabanlı gezinti yöntemi önerilmektedir. Stereo görme kullanarak, gezgin robotlar için gerekli mesafe ölçümü ve engel tespiti gerçekleştirilmiştir. Ayrıca önerilen sistemde bulunduğu ortama uyum sağlaması amacıyla bulanık mantık tabanlı bir yapı sunulmaktadır. Ek olarak, yapmış oldukları çalışmada A* algoritmasını kullanarak en kısa yolu bulmuşlardır.

2015 yılında, Sharifi ve Chen [36] tarafından yapılan çalışmada, kapalı ortamda bulunan otonom gezgin robotun gezintisi için, görme tabanlı engel sakınım tekniği önerilmektedir. Tek kamera ile elde edilen renkli görüntüler ortalama kaydırma yöntemi kullanılarak kümelenebilir. Kümelenen bu görüntüler, grafik bölümlenme teorisine dayalı yeni bir teknik kullanılarak sınıflandırılmaktadır. Sınıflandırılmış olan bu görüntüler gezgin robot hareketi için zemin, duvar, engel gibi anlamlı bilgiler içermektedir. Çalışmada önerilen teknik benzetim ortamında hazırlanmış ve sonuçları sunulmuştur.

2015 yılında, Cao ve diğ.[37] tarafından yapılan çalışmada, stereo kamera kullanarak otonom araçların gezinmesi için etkin bir yöntem önerilmektedir. Önerilen yöntem üç bölümden oluşmaktadır. Bu bölümler, yerel uzaklık haritasının oluşturulması, eğim analizi ve engel algılama ve yol planlama olarak belirlenmiştir. Eğim modeli için yeni bir eğim analiz modeli önerilmektedir. Yol planlaması için A* algoritması kullanılmıştır.

2015 yılında, Rettkowski ve diğ. [38] tarafından yapılan çalışmada, kuşgözü bakışı, görüntü birleştirme işlemi ve genişletilmiş A* algoritması yöntemlerini birleştirerek gezgin robot gezintisi için yöntem önerilmektedir. Yapılan çalışmada kuşbakışı ile elde edilen görüntüler birleştirilmiş, ortamın haritası çıkarılmıştır. Elde edilen harita üzerinde genişletilmiş A* yol planlama yöntemi uygulanmıştır. Arama alanı ve bellek kullanımı artırılarak A* algoritmasında yenilik yapılmıştır. Ayrıca robot kontrolü FPGA (Field Programmable Gate Array) ve ARM (Acorn RISC Machine) tabanlı bir kart sayesinde sağlanmıştır.

2015 yılında, Charan ve diğ. [39] tarafından yapılan çalışmada, bilgisayarlı görme tekniği kullanılarak otomatik gezinti ve nesne bulma yöntemi önerilmektedir. Önerilen yöntemde nesnelerin işaretlenmesi yerine, yol işaretleme tekniği kullanılmıştır. Ayrıca nesnelerin bulunması için aralıklı bölme tekniği kullanılmış ve benzeyen pikseller birleştirilmiştir. Önerilen algoritma karmaşık ortamlar için denenmiş ve yeterli sonuç alındığı belirtilmiştir.

2014 yılında, Cherubini ve diğ. [40] tarafından yapılan çalışmada, tekerlekli gezgin robotlar için görme tabanlı gezintisi sırasında hareketli ortamda engel sakınım algoritması önerilmektedir. Robot üzerine takılmış bir kamera tarafından gezinti esnasında alınan görüntüler sayesinde yol bilgisi elde edilmektedir. Gidilen yol üzerinde engellerden sakınma işlemi robot üzerinde yer alan lazer sayesinde olmaktadır. Önerilen yaklaşımda, ortamda hareketli engeller bulunduğundan bu engellerin hızları kalman tabanlı bir yöntem kullanılarak hesaplanmaktadır. Elde edilen hız bilgileri hücre tabanlı bir yaklaşım kullanılarak engel pozisyonlarının tahmininde kullanılmaktadır.

Önerilen yaklaşım dış mekân ortamlar için test edilerek, gezgin robotun güvenli bir şekilde dolaştığı gözlenmiştir.

2012 yılında Zhang ve diğ. [41] tarafından yapılan çalışmada, gezgin robotların gezintisi sırasında yerel minimumdan kaçmak, engellerden kaçınarak optimum yolu bulmak amacıyla yeni bir metot önermişlerdir. Bu yöntemde çevrede bulunan engeller hakkında bilgi edinmek için ızgara yöntemi, engellerden kaçınma yolunu optimize etmek için evrimsel yapay potansiyel alanlar yaklaşımını kullanmışlardır. Önermiş oldukları yöntemi benzetim ortamında gerçekleştirmişlerdir.

2012 yılında, Kim ve Do [42] tarafından yapılan çalışmada, tek kamera kullanarak gezgin robotun hareketli engellerden sakınması üzerine bir yöntem önerilmektedir. Blok tabanlı hareket tahmin etme yöntemini çalışmalarında kullanmışlardır. Kamerasından alınan görüntü ilk olarak küçük görüntülere bölünmektedir. Daha sonra iki ardışık görüntüde yer alan her bloktaki hareketlilik taranarak karşılaştırma yapılmaktadır. Eşleşen bloklar içerisinde bulunan hareketlilik uyuşuyorsa, o bölümde hareketli bir nesne olmakta ve hareketli bir nesne olarak sınıflandırmaktadır. Önerilen yöntemin kapalı ortam gezgin robotlar için uygun olduğundan bahsedilmektedir.

2012 yılında, Sezer ve Gökaşan [43,44] tarafından yapılan çalışmada holonom olmayan kara taşıtlarının otonomluğu ve engellerden sakınmasını sağlamak amacıyla “Boşluğu Takip Et (BTE)” ismini verdikleri yeni bir yöntem önerilmektedir. Önerilen yöntemin hem sabit, hem de hareketli engeller üzerinde etkili olduğunu yapmış oldukları benzetim ve gerçek deney ortamında aldıkları sonuçlara göre beyan etmişlerdir.

2012 yılında, İyidir [45] tarafından yapılan tez çalışmasında, gezgin robotlar için kapalı ortamda görsel girdilere dayalı engel tanıma ve engelden sakınma yöntemi önerilmektedir. Çalışmada, robotun hareket ettiği zemin ve engeller tek bir kamera ile birbirinden ayrılmaktadır. Engel tanıma işlemi için görsel arka plan çıkarım yöntemine uyarlama yapılmıştır.

2012 yılında Öztürk [46] tarafından yapılan tez çalışmasında, görme tabanlı gezinti üzerine bir çalışma yapılmıştır. Çalışmada tek kamera kullanarak, eş zamanlı konumlandırma ve haritalandırma (SLAM) yöntemi önerilmektedir. Robotun konumunun tahmini için genişletilmiş kalman filtresi kullanılmıştır. Ayrıca önerilen yöntemin bilinmeyen bir ortamda gezgin robot için uygun olduğu ifade edilmiştir.

2010 yılında, Ng [47] tarafından yapılan doktora tez çalışmasında, gezgin robotların bilinmeyen ortamda gezintisi için kullanılan algoritmaların araştırılması ve analizi üzerine bir araştırma yapılmıştır. Literatürde yapılan böcek ve böcek ailesine ait algoritmaların detayları ve başarımları incelenmiştir. Ayrıca, gezgin robotun engellerden sakınması için yeni bir böcek algoritması önerilmiştir. “EyeSim” benzetim ortamında modellenerek başarımların analizleri gerçekleştirilmiş ve elde edilen sonuçlar sunulmuştur.

2010 yılında, Zhao ve diğ.[48] tarafından yapılan çalışmada gezgin robotlar için stereo görme tabanlı engel sakınım ve gezinti sistemi önerilmektedir. Önerilen yöntemde yerel ortamın tanımlanması için dinamik programlama tekniği ve piramit veri yapısı bölge tabanlı stereo görme ile birleştirilerek kullanılmıştır.

Literatürde gezgin robotların engel tanıma, engel sakınım ve gezintisini sağlamak için çok sayıda çalışma yapılmıştır. Yapılan araştırma çalışmasında son yıllarda yapılan ve kamera kullanılan yöntemler incelenmiştir. Görme tabanlı gezinti, engel tespit ve sakınım çalışmalarında, zemini ve engelleri ayırmak için çeşitli görüntü işleme teknikleri kullanılmaktadır. Arka plan çıkarımı, renk dönüşümleri, histogram çıkarma yöntemi, stereo görme kullanarak uzaklık haritası çıkarımı, özellik çıkarım tabanlı yaklaşımlar, görüntü bölümlenme ve birleştirme gibi çeşitli yöntemler kullanılarak ortamda yer alan engellerin tespit edilmesi ve tanınması üzerine çalışmalar yapılmıştır. Engellerin tespit edilmesinden sonra, potansiyel alan yaklaşımı, boşluğu takip et yöntemi, böcek algoritmaları, gibi çeşitli yöntemler uygulanmıştır. Özellikle son yıllarda gezgin robotlarda stereo görme, ortamın 3 boyutlu haritasının çıkarılması, engellerin yerlerinin tespit edilmesi ve engel sakınım işlemlerinde yaygın ve etkin olarak kullanılmaktadır.

1.4. Gezgin Robotların Yol Planlama Problemi

Gezgin robotlarda yol planlama problemi, üzerinde yer alan algılayıcılar tarafından tespit edilen engellere çarpmadan, yine algılayıcılar tarafından tespit edilmiş olan hedefe giden uygun yolun bulunması olarak tanımlanabilir. Uygun yolun bulunması sırasında, robotun en az enerji tüketimi gerçekleştireceği, düşük maliyetli bir yolun bulunması ve en kısa mesafeden hedefe ulaşacağı yolun bulunması gibi problemlerle uğraşmaktadır.

Yol planlama problemi farklı şekillerde sınıflandırılmaktadır. Bu farklı şekillerden biri, gezgin robotun bulunduğu çevre bilgilerinin, statik veya dinamik olmasına göre sınıflandırılmasıdır. Statik ortamlarda yapılan yol planlama işleminde sadece gezgin robot hareketli, engeller ve hedef hareketsizdir. Ancak dinamik bir ortamda yapılan yol planlama işleminde gezgin robotun hareketi dışında, ortamda yer alan engeller ve hedef herhangi bir zamanda herhangi bir yere hareket edebilmektedir. Yol planlama konusunda yapılan bir başka sınıflandırma ise yerel ve genel yol planlama olarak tanımlanabilir.

Robot hedefe ulaşmak için engellere çarpmadan bir sonraki adımına karar verirken, uygun zamanda, uygun adımda ve yeterli enerji tüketerek işlem yapması için yerel yol planlama düzeneğine ihtiyaç duymaktadır. Genel yol planlama algoritmalarında robot bulunduğu noktadan hedefe doğru hareket etmeden önce çarpışma yapmadan ilerleyeceği yolun tamamına karar vermektedir. Genel yol planlama algoritması çevrimdışı (offline path planning) yol planlama algoritması olarak da adlandırılmaktadır. Yerel yol planlama algoritmaları, genellikle çevrim içi (online path planning) yol planlama algoritması olarak adlandırılmaktadır.

Gezgin robotun yol planlama probleminde sezgisel veya hesaplama dayalı çeşitli yöntemler kullanılmaktadır. Genetik Algoritma (GA), Karınca Koloni Algoritması, Parçacık Sürü Optimizasyonu (PSO), Bakteri Potansiyel Alan Algoritması, Yapay Arı Koloni algoritması, Böcek Algoritmaları gibi çeşitli yöntemler kullanılmaktadır. Ayrıca, A*, Dijkstra, D* ve takviyeli öğrenme (Q Learning) gibi hesaplama dayalı yöntemler de bulunmaktadır.

Sezgisel ve hesaplama dayalı yöntemlerin avantaj ve dezavantajları bulunabilir. Hesaplama dayalı yöntemler, yüksek işlem maliyeti ve bellek kullanımı artmasına sebep olurken, sezgisel yöntemler ortam koşullarının zor olduğu durumlarda iyi sonuçlar vermeyebilir. Bu iki yöntemin iyi yönleri birleştirilerek hibrit yöntemler de yol planlaması için kullanılabilir.

Gezgin robotun yol planlama problemi konusunda, literatürde yapılan çalışmalarda, sezgisel ve hesaplama yönelik yapılmış çalışmaların iyileştirilmesi, hibrit yapıda düzenlenmesi ve yeni bir yöntem bulunması üzerine yoğunlaşmıştır. Son yıllarda, yol planlama problemine yönelik yapılan bazı çalışmalar aşağıda sunulmuştur.

2016 yılında, Das ve diğ. [49] tarafından yapılan çalışmada, klasik takviyeli öğrenme ve parçacık sürü optimizasyonu tekniklerinden esinlenilerek çoklu gezgin robotlar için yol planlama yöntemi önerilmektedir. Klasik takviyeli öğrenme yöntemi çok fazla matematiksel hesaplama gerektirdiğinden, maliyet ve bellek tüketimi yüksek olmaktadır. Parçacık sürü optimizasyon tekniği geliştirilmiş ve takviyeli öğrenme yöntemi ile birleştirilerek kullanılmıştır. Önerilen algoritma benzetim ortamında ve Khepera II gezgin robotu üzerinde uygulanarak test edilmiştir.

2015 yılında, Contreras-Cruz ve diğ. [50] tarafından yapılan çalışmada, gezgin robotun yol planlama probleminin çözümü için evrimsel bir yaklaşım kullanarak çözüm önerilmektedir. Önerilen yaklaşımda, yerel arama yöntemi olarak yapay arı koloni algoritması kullanılırken, yerel arama yöntemine uygun yolun bulunması için evrimsel programlama kullanılmaktadır. Önerilen yöntem, klasik olasılığa dayalı yol haritası yöntemi ile karşılaştırılmakta ve daha iyi sonuçlar verdiği ifade edilmektedir.

2015 yılında, Karami ve Hasanzadeh [51] tarafından yapılan çalışmada, gezgin robotların hareket ve yol planlama problemi karmaşık, iki boyutlu ortamlar için Genetik Algoritma tabanlı bir yaklaşım önerilmektedir. Önerilen yöntemde, genetik algorithmada yeni bir seçim operatörü önerilmiş ve elde edilen sonuçlar benzer çalışmalar ile karşılaştırılmıştır. Karşılaştırma sonucunda önerilen algoritmanın en iyi sonucu verdiği ifade edilmiştir.

2015 yılında, Charalampous ve diğ. [52] tarafından yapılan çalışmada, gezgin robotların yol planlama ve gezinti sistemi için destek vektör makinesi (Support Vector Machine - SVM) tabanlı yerel yol planlaması önerilmiştir. Orijinal destek vektör makinesi çalışmalarında, engellerin noktasal şekilde ifade edildiği ortamda, çarpışmadan kaçınmak için ortamın iki boyutlu haritası kullanılmaktadır. Önerilen yöntem, genel yol planlama yöntemlerine ait stratejileri, robot pozisyonunu ve engel kümelerini de dikkate almaktadır. Önerilen sistemin etkinliğini göstermek amacıyla, hazırlanan yerel yol planlayıcıyı gezgin robot gezinti sistemi içerisine dahil edilerek gerçek dünya problemlerinde kullanılmak üzere hazırlanmıştır.

Ortamın üç boyutlu haritası çıkarılmış ve daha sonra yol planlayıcısının kullanması için iki boyutlu haritaya çevrilmiştir. Ayrıca, önerilen sistem engel tanıma işlemini de desteklemektedir.

2015 yılında, Deepu ve diğ. [53] tarafından yapılan çalışmada, tek kamera kullanılarak gezgin robotların gezintisi için yol üretimi üzerine yöntem önerilmiştir. Kameranın yanı sıra mesafe ölçme işlemi için lazer algılayıcısı da kullanılmıştır. Önerilen yöntemde, zemin bölümlenme, kenar tanıma ve renk eşik değerlerinin belirlenmesi gibi görüntü işlemede yaygın olarak kullanılan yöntemler kullanılmıştır.

2013 yılında, Chandak ve diğ. [54] tarafından yapılan çalışmada, gezgin robotların yol planlama problemini çözmek için görüntü işleme teknikleri kullanan bir yöntem önerilmiştir. Önerilen çalışmada, dalga yüzü tabanlı bir yöntem geliştirilerek kapalı ortamda yer alan farklı geometrik şekillerde bulunan engellerin tanınması ve engelden sakınılması işlemi gerçekleştirilmiştir. Kameradan elde edilen görüntülerin işlenmesi MATLAB ortamında yapılmıştır.

2013 yılında, Miao ve Tian [55] tarafından yapılan çalışmada, gezgin robotların yol planlama problemi için gelişmiş benzetilmiş tavlama yöntemi kullanılarak dinamik bir yaklaşım önerilmektedir. Dinamik ortamda, hareketli ve sabit engeller için standart benzetilmiş tavlama yönteminden daha iyi sonuç verdiği ifade edilmektedir.

2013 yılında, Ohnishi ve Imiya [56] tarafından yapılan çalışmada, görme tabanlı konum belirleme ve gezinti sistemi önerilmektedir. Gezgin robotun üzerine yerleştirilen bir kameradan elde edilen başarılı sıralı görüntülerden optik akış ve görsel potansiyel hesabı yapılmaktadır. Sıralı görüntülerden elde edilmiş ortam resimleri kullanılarak, gezintinin sağlanması için görsel potansiyel hesabı yapılmaktadır. Robotun başlangıç pozisyonu kullanılarak hedefin yönü bulunmaktadır. Gezgin robot hedefe giden yerel yollardan dinamik bir seçim gerçekleştirmektedir. Bu yol üzerinde gezgin robot engellerden sakınma işlemini de gerçekleştirmektedir. Ayrıca, robot başlangıç pozisyonunda hedefe doğru gideceği yönü bularak ilerlemektedir. Çalışmada, elde edilen sonuçlar ve deneysel çalışmalar sunulmaktadır.

2012 yılında, Tuncer ve Yıldırım [57, 58] yapmış oldukları çalışmalarında, gezgin robotun yol ve hareket planlama problemleri için genetik algoritma tabanlı yöntem önerilmektedir. Önerilen çalışmada, standart genetik algoritmada kullanılan mutasyon işlemi üzerinde iyileştirme yapılarak, daha uygun çözümler bulunduğu ifade edilmektedir. Ayrıca çalışmalarında gezgin robotun hareket planlaması için MATLAB ortamında ara yüz tasarlamışlardır.

2012 yılında, Ardiyanto ve Miura [59] tarafından yapılan çalışmada, gezgin robotların yol planlama probleminin çözümü için, gerçek zamanlı yol planlama ve gezinti sistemi önerilmektedir. Önerilen sistemin gerçek zamanlı, dinamik ve karmaşık ortamlar içinde uygun olduğu ifade edilmektedir. Önerilen yöntemde, robotun bulunduğu noktadan hedefe olan güvenli ve kısa yollar hesaplanmaktadır. Sonrasında, robotun kinematiği ve dinamiği dikkate alınarak rastgele bir yol seçilmektedir. Rastgele seçilen yol üzerinde yol planlama kalitesini arttırmak için sezgisel kısıtlamalar uygulanmaktadır. Önerilen algoritmanın, karşılaştırma yapılarak uygun sonuçlar verdiği ifade edilmiştir. Çalışma benzetim ortamında gerçekleştirilmiş ve sonuçlarının gezgin robotların gerçek dünyadaki uygulamaları için uygun olduğu gösterilmiştir.

2011 yılında, Liu ve diğ. [60] tarafından yapılan çalışmada, gezgin robotların yol planlama problemi için uyarlanabilir genetik algoritma yapısını kullanan bir yöntem önerilmektedir. Gezgin robotun yol planlamasında kullanılmak üzere, özelleştirilmiş genetik operatörü ve ayarlanabilir parametreler kullanılmaktadır. Çözüme ait uygunluk değerleri, çaprazlama ve mutasyon işlemine bağlı olarak değişmektedir. Önerilen algoritmada, uyarlamalı olarak değişen ve optimum yolu bulan yöntem kullanılmaktadır. Elde edilen sonuçların gezgin robotların yol planlama problemi için uygun olduğu ifade edilmektedir.

2011 yılında, Talmiselvi ve diğ. [61] tarafından yapılan çalışmada, gezgin robotların optimum yol seçimi için genetik algoritma tabanlı bir yöntem önerilmektedir. Genetik algoritma kullanılarak önerilen yol seçim yönteminin test edilmesi için ızgara benzeri ortam oluşturularak uygun yol testlerinin bulunması sağlanmıştır. Oluşturulan yapay ızgara ortamında, gezgin robotun engel sakınımı ve yol planlaması için etkin ve uygun bir yöntem önerildiği ifade edilmektedir.

2011 yılında, Suvaydan [62] tarafından yapılan tez çalışmasında, gezgin robotların yol planlama problemi için karınca koloni algoritması temelli bir yöntem önerilmektedir. Önerilen yöntemin test edilmesi için bir benzetim ortamı yaratılarak dört farklı çevre tasarlanmıştır. Önerilen yöntemin test aşamasında yerel feromon ve genel feromon değerleri güncellenerek sonuçlar elde edilmiş ve başarımları testleri gerçekleştirilmiştir.

2010 yılında, Yao ve Ma [63] tarafından yapılan çalışmada, gezgin robotların yol planlama problemi için statik ortamlarda kullanılabilen genetik algoritma tabanlı bir yöntem önerilmektedir.

Yöntemde, birey veya yolun değerlendirilmesi basit bir uygunluk fonksiyonu sayesinde yapılmaktadır. Özel bir genetik operatörü kullanılarak, hesaplama maliyeti düşürülmekte ve gezgin robotlar için uygun sonuçlar üretilmektedir.

2009 yılında, Zahoo ve diğ. [64] tarafından yapılan çalışmada, maden felaketlerinde arama çalışmalarında kullanılan gezgin robotlar için genetik algoritma tabanlı genel yol planlama yöntemi önerilmektedir. Madene ait ızgara model coğrafi bilgi sistemi tarafından elde edilmektedir. Genetik algorithmada kullanılacak başlangıç çözüm kümesi, karınca koloni algoritması ve öncelikli gruplama tekniği kullanılarak oluşturulmaktadır. En kısa yolun bulunması için dört genetik operatörü ve uygunluk fonksiyonu tasarlanmıştır. Çaprazlama ve mutasyonun kendinden uyarlanabilir olması için mutasyon işleminde küçük değişiklikler yapılmış ve MATLAB ortamında test edilerek gezgin robotlar için uygunluğu ifade edilmiştir.

Sonuç olarak, gezgin robotların yol planlama problemi için yapılan literatür çalışmasında çok sayıda ulusal veya uluslararası yayın ve tez çalışmaları incelenmiş ve son yıllarda yapılan çalışmalardan bazıları bu tez çalışmasında özetlenmiştir. Tez çalışması kapsamında, kamera ve genetik algoritma kullanılarak yol planlaması yapıldığı için bu tür çalışmaların üzerinde durulmuştur. Genetik algoritma tabanlı çok sayıda çalışma bulunmaktadır. Bu çalışmalarda, genellikle uygunluk fonksiyonu için optimum değerinin ayarlanması üzerine çalışmalar yapılmıştır. Bu amaçla, başlangıç birey sayısının oluşturulmasında, çaprazlama ve mutasyon işlemlerinde yenilikler yapılmaya çalışılmaktadır.

Yapılan çalışmalarda, temel olarak yol planlama problemi genel ve yerel olarak incelenmektedir. Genel yol planlama problemlerinde, gezgin robota başlangıçtan hedefe kadar gideceği yol verilmektedir. Yerel yol planlama probleminde ise, gezgin robotun sonraki adımına karar verilirken, uygun yol, maliyet ve enerji tüketim hesabı gözetilmektedir. Yerel yol planlama problemi için yapılan çalışmalarda, gezgin robotun bulunduğu konumdan doğrudan hedefe değil de, bir veya birkaç adım sonraki gezintisine karar verdiği gözlenmektedir.

1.5. Gezgin Robotlarda Bilgisayarlı Görme

Görme yeteneđi insanođlunun bulunduđu ortamı, çevresini ve çevresinde gerçekleşen olayları algılamasında ve tepki vermesinde önemli bir rol oynamaktadır. İnsanlar, etrafında gördüđu bilgileri anlamlandırarak hareket etmektedirler.

Bilgisayarlar veya bilgisayar özelliđi taşıyan cihazların, kendisiyle haberleştilen kamera veya kameralar ile elde etmiş olduđu görüntü bilgisini işleyip analiz etmesi, bu bilgilerden anlam çıkarma işlemi “Bilgisayarlı Görme (Computer Vision)” olarak isimlendirilmektedir. Bilgisayarlı görme, elde edilen kamera görüntüleri kullanılarak gerçek dünyaya ait hesaplamaların yapılması işlemi de kapsamaktadır.

Bilgisayarlı görme son yıllarda hayatın her alanında kullanılmaya başlanmıştır. Trafik, sağlık, güvenlik, eğitim, internet, endüstriyel alanlarda çeşitli amaçlarla kullanılmaktadır. Bilgisayarlı görmenin yaygınlaşması ile beraber, gezgin robot çalışmalarında da yaygın olarak kullanılmaya başlanmıştır. Gezgin robot çalışmalarında genellikle, nesnelerin tanınması, bulunması, üç boyutlu modellerinin çıkarılması, robotun engellere çarpmadan gezintisinin sağlanması, robot gezintisi sırasında ortamın keşfinin yapılması gibi çalışmalar gerçekleştirilmektedir [65].

Bilgisayarlı görme işleminde görüntü işleme teknikleri kullanılmaktadır. Görüntü işleme, yapılırken kamera benzeri bir algılayıcı vasıtasıyla görüntü elde edilmektedir. Elde edilen görüntü, üzerinde işlemler yapılacak sayısal veriye dönüştürülmektedir. Bu işlemde sonra çeşitli algoritmalar kullanılarak veriye anlam kazandırılmaktadır. Görüntü işlemede tüm işlemler, resim üzerindeki en küçük resim elemanı olan pikseller üzerinde gerçekleştirilmektedir.

Görüntü işlemede kullanılan yöntemler, probleme göre değişmektedir. Probleme göre uygulamalar geliştirilirken, çeşitli yöntemlerin birlikte kullanılması, birtakım ölçütlerin dâhil edilmesi ve işlem hızlarının artırılması gerekebilir. Görüntü işlemek için farklı alanlarda farklı yöntemler kullanılmaktadır. Bunlardan bazıları aşağıda verilmiştir [66 - 68].

- Görüntü Dönüşümleri
- Görüntü İyileştirme
- Görüntü Onarma

- Görüntü Bölümleme
- Görüntü Sıkıştırma
- Görüntü Sunma
- Görüntü Algılama

Bilgisayarlı görme işlemi düşük, orta ve yüksek seviye olmak üzere üç aşamada gerçekleştirilmektedir [69, 70]. Bu aşamalarda gerçekleştirilen işlemler maddeler halinde aşağıda sunulmuştur.

- Düşük seviye, olarak tanımlanan ilk aşamada ortamdan alınan görüntüler, çeşitli görüntü işleme teknikleri kullanılarak farklı bir görüntüye dönüştürülmektedir. Bu aşamada, görüntü alınmakta ve ön işlemlerden geçirilmektedir. Çeşitli filtreler uygulanarak görüntü üzerinde yer alan bazı gereksiz ayrıntılar yok edilmekte ve temizlenmektedir. Bu aşamada, bazı durumlarda, görüntü bölümleme işlemi de uygulanmaktadır.
- Orta seviye, olarak tanımlanan ikinci aşamada, ilk aşamada oluşturulan yeni görüntü üzerinden özellik çıkarma işlemi gerçekleştirilmektedir. Bu bölümde, renk dönüşümleri, eşik değer uygulamaları, kenar köşe bulma işlemleri, histogram çıkarımları gibi işlemler uygulanmaktadır. Ayrıca, birbiri ile alakalı noktaların sınıflandırılması, hareket analizi, şekillerin tanımlanması gibi işlemler de gerçekleştirilir. Uygulanacak olan algoritmanın matematiksel modeli tasarlanır.
- Yüksek seviye, olarak tanımlanan son aşamada, görüntüye ait tanımlama çalışması yapılmaktadır. Model tabanlı olarak uygulamaların tahmin edilerek doğrulanması, özel parametreler kullanılarak nesne boyutu, mesafesi gibi tahminlerin yapılması vb. işlemlerin gerçekleştirildiği aşamadır. Bu aşama sonunda bilgisayarlı görme sona ereceği için, “eşleme gerçekleştirildi”, “nesne bulunamadı”, “nesneler arası mesafe 50 cm”, “ortamda 5 adet kırmızı elma bulunmaktadır”, “robotun hedefe olan uzaklığı 125cm”, gibi kesin cevaplar elde edilmektedir. Elde edilen bu veriler sayesinde görüntüleme, hareket ettirme, teşhis etme ve karar verme işlemleri gerçekleştirilmektedir.

Otonom gezgin robotlarda bilgisayarlı görme çalışmaları, robotun gezgin olması için gömülü bir sistem kullanılarak gerçekleştirilmiştir. Aksi durumda gezgin robotların,

bilgisayar ile kontrol edilmesi gerekmektedir. Bilgisayar ile kontrol edildiği durumda otonom özelliği sağlanabilirken, gezginlik özelliği tam olarak sağlanamamaktadır.

1.5.1. Gömülü görme ve uygulama alanları

Gömülü görme işlemi, gömülü sistemler üzerinde, görsel girdilerden anlam çıkarmak olarak tanımlanmaktadır. Gömülü görme, çevreyle herhangi bir fiziksel etkileşim olmaksızın, optik araçlar kullanarak cisimleri algılama ve buna göre bilgi toplama ya da çeşitli süreçleri yönetmektir.

Son yıllarda, oyunlar, otomotiv güvenliği ve gözetleme sistemleri gibi çeşitli pazarlarda yaygın olarak kullanılmaktadır. Gömülü görmenin, oyunlar, otomotiv ekipmanları ve tüketici elektroniği gibi alanlarda kullanımının artması yeni çalışmalar ve yeni taleplerin beklentisini doğurmaktadır. Ayrıca, gömülü sistemlerde kullanılan işlemcilerin, işlem gücünün artması, maliyetinin ve enerji tüketiminin düşürülmesi ile yeni tüketicilerin bu pazara girmesi beklenmektedir [71, 72].

Gömülü görme ile ilgili uygulamaların geliştirilmesinde ve uygulanmasında;

- Sistem bütünlüğü problemi,
- Gömülü görme ile ilgili pratik çözümler oluşturma konusunda sınırlı deneyimlerin bulunması,
- Gömülü görme sistemlerinde genellikle yüksek maliyet, boyut ve güç tüketimi ile ilgili kısıtlamaların bulunması,
- Kullanılmakta olan görme fonksiyonunun, özel veya paralel donanımlar gerektirmesi,
- Gömülü görme konusunda deneyimli kişilerin az olması,
- Algoritmaların dinamik ve farklı yapılarda olmasından dolayı sabit fonksiyon hesaplama motorlarının daha az etkili olması gibi zorluklar yaşanmaktadır.

Gömülü görmenin kullanım alanları her geçen gün artmaktadır. Alt bölümde yaygın kullanılan bazı alanlara örnekler sunulmuştur.

1.5.1.1. Gözetleme sistemleri

Emniyet ve güvenlik ile ilgili artan endişeler nedeniyle, gözetleme kameralarının kullanımını son 10 yıl içerisinde ciddi şekilde artış göstermektedir. İngiltere’de yakla-

şik olarak 1,85 milyon kamera kurulmuş ve bu kameraların günlük üretilen kamera kaydı $2,5 \times 10^9$ dakika olarak belirlenmiştir [71]. Elde edilen bu kamera görüntülerin el ile kontrol edilmesi imkansızdır. Bu sebepten dolayı akıllı güvenlik kameralarının önemi artmaktadır. Akıllı güvenlik kameraları olayları belirli bir türde aramak için görme tekniklerinden birini kullanmaktadır. Buradaki akıl kamera veya yerel sunucu içerisinde olmaktadır. Ancak, bu sistemlerdeki en büyük problem veya zorluk farklı ortam ve ihtiyaçlardaki doğruluğun sağlanmasıdır.

1.5.1.2. Otomotiv güvenliği

Yılda yaklaşık olarak 1,2 milyon kişi trafik kazalarında hayatını kaybetmektedir. Bu sebeple, otomotiv güvenliğinde gömülü görme uygulamaları önem kazanmıştır. Otomotiv güvenliğinde kullanılan gömülü görme uygulamalarının amacı trafik kazalarını azaltmaktır. Bu amaçla;

- Önümüzdeki araca hızlı bir şekilde yaklaşırken aracın uyarı vermesi,
- Araç yolunda yaya, bisikletli veya farklı bir madde olduğunda uyarı vermesi,
- Yanlışlıkla şeritten ayrılma olduğunda uyarı vermesi,
- Aracı süren kişinin uykusu geldiğinde sürücü uyarısı vermesi,
- Otomatik olarak sinyallerin kısılması gibi otomotiv güvenliğini artırıcı uygulamalar geliştirilmektedir.

1.5.1.3. Video oyunları

Son zamanlarda, video oyunlarına ilgi gittikçe artmaktadır. Donanımsal ve yazılımsal anlamda, 60 milyar dolarlık yıllık iş hacmine sahiptir. Görme tabanlı video oyun kontrolü, yeni tip oyunlar ve kullanıcılar oluşturmaktadır. Örneğin Microsoft Kinect, şu ana kadar en hızlı satılan tüketici elektroniği olmuştur. Piyasaya sürüldüğü ilk altı aylık dönem içerisinde yaklaşık olarak 10 milyon adet satılmıştır. Kullanımının kolay ve ucuz olması, video ses ve kontrol aracı olarak da kullanılması bu satışlarda öneme sahiptir. Ayrıca Kinect, son zamanlarda bilgisayarlı görme üzerine farklı uygulama geliştiren yazılımcılarında ilgisini çekmiştir.

1.5.1.4. Yüzme havuzu güvenliği

Dünya genelinde yılda yaklaşık olarak 400.000 ölümlü sonuçlanan boğulma vakası yaşanmaktadır. 1 ila 14 yaş arası çocuklarda kaza sonucu ölümlerde ikinci sırada yer

almaktadır. Bu ölümlerin, %19'u profesyonel cankurtaranların bulunduğu halka açık havuzlarda olmaktadır [67]. Kurulan Yüzme Havuzu İzleme sistemi sayesinde tehlike altındaki yüzücüler ve cankurtaran uyarıları izlemekte ve hızlı bir şekilde müdahale edebilmektedir.

1.5.1.5. Görme tabanlı gezgin robotlar

Görme tabanlı gezgin robot çalışmalarında, kamera veya kameralar kullanılarak insan göz yapısının çalışma mantığı modellenmeye çalışılmaktadır. Bu çalışmalarda gözün görme mantığından faydalanılarak, nesnelere arası uzaklık, nesnelere kameralara olan uzaklıkları gibi çalışmalar yapılmaktadır. Sadece nesnelere kameralara olan uzaklıklarının tespiti değil aynı zamanda nesnelere renkleri, desenleri ve şekilleri hakkındaki bilgiler de beraberinde elde edilebilmektedir.

Gezgin robotlar, kara, hava, su altı ve uzay çalışmalarında kullanılmaktadır. Ayrıca, bir bina veya oda içerisinde taşıma, temizlik ve servis amacıyla, askeri çalışmalarda, su altı ve su üstü taşıma sistemlerinde, insanların ulaşamayacağı mağara, tünel ve uzay ile ilgili görme tabanlı uygulamalarda kullanılmaktadır.

1.5.2. Gömülü görmenin sistem seviyesinde problemleri

Gömülü görme sistemleri ile ilgili yaşanan problemler aşağıda maddeler halinde sunulmuştur.

- Gömülü görme sistemlerinde başarı, görme algoritmalarının çalışmasının yanı sıra bir çok faktörün birlikte çalışmasına bağlıdır. Bunların başında, görüntü ön işleme, görüntü algılayıcıları, aydınlatma ve optik özellikler gelmektedir.
- Gömülü görme sistemleri konusunda bilinenlerin ve uzman kişilerin az olması problemi yaşanmaktadır.
- Görme ile ilgili çok sayıda algoritma olmasına rağmen en uygun ve en iyi algoritmanın seçilmesinde zorluklar yaşanmaktadır.

Görme işlemi genellikle karmaşık bir işlem olup, yüksek matematiksel işlemler gerektirmektedir. Maliyet, boyut ve enerji kısıtlamaları altında yapılacak iş için uygun işlemcinin seçilmesi gerekir. İşlemcinin etkin kullanılması için algoritmaların optimize edilmesi gerekmektedir.

1.5.3. Gml grme ilemlerinde bilgisayarın rol

Gml grme uygulaması gelitirirken ve uygulamasını gerekletirirken bilgisayarın olumlu ve olumsuz rolleri bulunmaktadır. Gml grme uygulamaları veya grme algoritmaları yaamına bir bilgisayar ile balamaktadır. Bilgisayarın avantaj saęlayan zellikleri;

- Her yerde bulunabilir ve ucuz olması,
- İyi bir gelitirme alt yapısı saęlaması,
- Çok sayıda gelitirme aracı ve ktphaneler sunmaktadır.
- Grme iin zel ktphaneler bulunmaktadır.
- eitli benzetim araları bulunmaktadır.
- rnek uygulamalar bulunmaktadır.
- Aę aralarının, kameraların, eitli grnt ve giri ıkı aygıtlarının kullanımı kolaydır.
- Grme algoritmalarının alıması iin paralel alıma yapılabilir.
- Bir gn ierisinde bilgisayar ve web kamerası kullanarak basit bir grme uygulaması hazırlanabilir.

Gml grme uygulaması gelitirirken bilgisayarın olumsuz ynlerinden bazıları aaęıda verilmitir.

- Bilgisayar gml grme sistemlerini uygulamak iin en uygun platform deęildir.
- Bazı uygulamalar bilgisayara gml olarak entegre edilebilse de, bunlar maliyet, boyut ve g tketim faktrlerine baęlıdır.
- Bilgisayarların gerek zamanlı grme uygulamalarındaki performansları yetersizdir.
- Gml grme sistemlerinde kullanılan ktphane ve araların oęu bilgisayarda bulunanlarla aynıdır. Ancak bilgisayarda gelitirmek ile gerek sistem zerinde gelitirmek arasında nemli farklılıklar bulunmaktadır.

2. KULLANILAN ARAÇLAR

Bu bölümde, yapılan tez çalışmasında kullanılan donanımsal ve yazılımsal araçlar hakkında detaylı bilgiler sunulmuştur.

2.1. Tek Kart Bilgisayarlar

Gömülü sistemlerin yaygınlaşması ile beraber kullanım alanları her geçen gün artmaktadır. Son yıllarda, kullanım kolaylığı, çabuk erişilebilirliği, maliyet düşüklüğü, AR-GE (Araştırma-Geliştirme) çalışmalarının artması, üzerine işletim sistemi kurulabilmesi, gömülü sistem olarak Tek Kartlı Bilgisayarların (TKB - Single Board Computer - SBC) akademik ve endüstriyel alanda kullanımını yaygınlaştırmaktadır.

TKB, işlevsel bir bilgisayarın gereksinim duyduğu işlemci, bellek, giriş/çıkış gibi birimlerin tek bir kart üzerinde birleştirilmesiyle oluşturulmaktadır. TKB, üzerinde genellikle kart yuvası (ekran kartı, ses kartı vb parçaları takmak için gerekli yuva) bulunmaz. Sistemde yer alan birimler (Ram, İşlemci vb.) düşük maliyetli, az güç tüketen şekilde küçük ve basit olarak tasarlanmaktadır. Depolama birimi olarak, “disk on module”, “disk on chip” ve “compact flash” gibi flash disk teknolojileri kullanılmaktadır [73-75].

TKB, yaygın olarak yuva desteği olan ve yuva desteği olmayan olarak iki farklı mimaride tanımlanmaktadır. Yuva desteği olmayan tek kartlı bilgisayarlarda, tüm bileşenler tümleşik olarak gelmektedir. Bu tür sistemler tek başına kullanılmaktadır. Yuva desteği olan tek kartlı bilgisayarlarda ise, temel işlevleri içermesinin yanı sıra büyük sistemlere takılarak bütünleşik olarak çalışabilmektedirler.

TKB, genellikle endüstriyel alanda yapılacak olan prototip çalışmalarında veya iş kontrollerinde ara yüz sağlamak amacıyla kullanılmaktadır. Ayrıca, endüstriyel alanda, gömülü durumda bulunan diğer otomasyon cihazları arasında kontrol ve iletişim ara yüzü sağlamak amacıyla da kullanılmaktadır.

TKB kullanımını, son yıllarda akademik alanda, bilgisayarlı görme, akıllı algılayıcı kullanan ev sistemlerinin kontrol ve izlenmesi [76], tıp ve biyomedikal alanlarında [77-79] nesnelerin interneti (Internet of things - IoT) alanında [80,81], insansız olarak kullanılan deniz, hava ve kara araçlarında [82], tarımda bitkiler ile ilgili çalışmalarda [83], ve robot uygulamalarında prototip çalışması yapılmak amacıyla her geçen gün artmaktadır.

Özellikle, teknolojiye yaşanan gelişmeler, TKB tarafından kullanılan ARM (Advanced RISC Machine) tabanlı işlemci ve belleklerin geliştirmesine olanak sağlamaktadır. Bu sayede, her geçen gün yeni TKB'ler piyasaya çıkmakta veya kapasiteleri artarak sürüm güncellemeleri yapılmaktadır. Günümüzde yaygın olarak kullanılan, BeagleBoard, BeagleBoard xM, Beagle Board x15, PandaBoard, PandaBoard ES, Raspberry Pi, BeagleBone, CubieBoard, BananaPi, BeagleBone Black, TKB sistemleri bulunmaktadır [84].

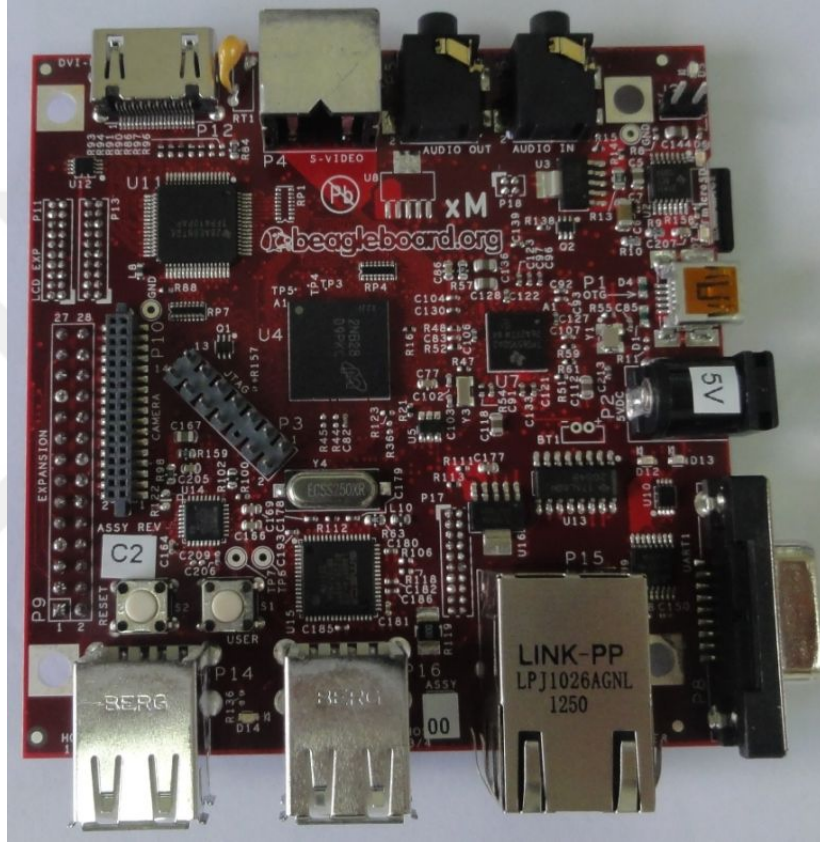
TKB'lerin, çok kartlı bilgisayarlarla karşılaştırıldığında, düşük enerji tüketimi, az yer kaplaması, boyutlarının küçük olmasından dolayı ortam uyumunun kolay sağlanması, hafif olması, maliyetlerinin düşük olması, taşınmasının kolay olması gibi avantajlı yönleri ortaya çıkmaktadır. Bu özelliklerinin yanı sıra;

- I2C, UART, USB, Seri ara birim gibi iletişimleri desteklemesi,
- TKB üzerine, Android, Linux, Windows 10 gibi çeşitli işletim sistemlerinin kurulabilir olması,
- TCP/IP, UDP gibi çeşitli iletişim protokollerini desteklemesi,
- Ses giriş ve çıkışları sayesinde, ses ile kontrol edilebilmesi,
- Görüntü çıkışı ile ekran, televizyon, paneller ile görüntülerin izlenmesi,
- Dâhili veya harici olarak, kablosuz iletişimi desteklemesi,
- Grafik işlemci ara birimi (GPU) sayesinde gerçek zamanlı görüntü işleme uygulamaları için uygun olması,
- Yönetimsel araçlar kullanılarak, uzaktan kontrol edilebilir olması,
- Özel işletim sistemleri dağıtımlarının olması gibi bir çok avantajı bulunmaktadır.

Yapılan tez çalışmasında, Texas Instrument (TI) ve DigiKey tarafından açık kaynak donanımı olarak üretilen BeagleBoard-xM TKB sistemi kullanılmaktadır.

2.1.1. BeagleBoard-xM

BeagleBoard-xM, TI ve Digi-Key tarafından üretilen açık kaynak donanım ve düşük güç tüketimli, TKB geliştirme kartı olarak bilinmektedir. Kart, 2008 yılında piyasaya çıkarılmış olan BeagleBoard kartının işlemci ve bellek gibi özelliklerinde yükseltme yapılarak ve yeni özellikler eklenerek 2010 yılında piyasaya çıkarılmıştır. Şekil 2.1’de BeagleBoard-xM’e ait resim sunulmaktadır.



Şekil 2.1. BeagleBoard-xM TKB

BeagleBoard-xM TKB üzerine takılan mikro SD kart içerisine Ubuntu, Android, Fedora, Windows CE ve Gentto gibi işletim sistemleri dağıtımları kurulabilmektedir. Ancak, BeagleBoard-xM için özel bir Linux dağıtımı olan Angström işletim sistemi diğerlerine göre daha iyi performans göstermektedir. Kart üzerinde, 4 adet USB A tipi, 1 adet USB B tipi, 1 adet seri iletişim arabirimi, ses giriş ve çıkışları, Ethernet girişi, 5 Volt güç girişi, DVI-D Çıkışı, S-Video çıkışı ve mikro SD kart yuvası bulunmaktadır. Bir çok USB desteği olan cihaz problemsiz bir şekilde çalışabilmektedir. Ayrıca, üzerinde Leopard kamera için özel giriş bulunmaktadır. BeagleBoard-xM kartına ait teknik özellikler Tablo 2.1’de sunulmaktadır [85].

Tablo 2.1. BeagleBoard-xM'nin teknik özellikleri

Birim Adı	Özelliği
İşlemci (CPU)	DM3730 / ARM Cortex A8 – 1 Ghz
Grafik İşlemci (GPU)	Power VR SGX 530 /200 Mhz
Sayısal İşaret İşleme (DSP)	TMS320C64 x+ / 800 Mhz
Bellek (RAM – DDR)	512 MB / 166 Mhz
USB Port Sayısı	4 Adet A – 1 Adet B
Depolama Birimi	Mikro SD Kart
Ekran Bağlantısı	DVI-D
S-Video	Var
Ses Giriş / Çıkış	3,5mm
Ethernet Bağlantısı	10/100
İç Birimler	I ² C, McBSP, MMC2, GPMIO, McPSI
İşletim Sistemi	Angström, Ubuntu, Android, Windows CE, Symbian, QNX
Fiyat	149\$

2.2. Angström İşletim Sistemi

BeagleBoard-xM, TKB sistemi üzerine takılan mikroSD karta Tablo 2.1’de belirtildiği üzere çok sayıda işletim sistemi kurulabilmektedir. Tez çalışması kapsamında, Android, Ubuntu ve Angström işletim sistemleri dağıtımları kurulumları yapılmış, deneysel çalışmalar sonucunda en iyi performansı Angström işletim sisteminin verdiği tespit edilmiştir. Performans değerlendirmesinde, sistemlerin kurulumları, isteklere vermiş olduğu tepkime süreleri, diğer arabirimlerle olan etkileşimleri, kablolu ve kablosuz erişim destekleri, kamera sürücülerinin desteklenmesi ve problemsiz çalıştırılması, seri arabirim üzerinden gezgin robot ile iletişimleri dikkate alınarak uygunluğu test edilmiştir.

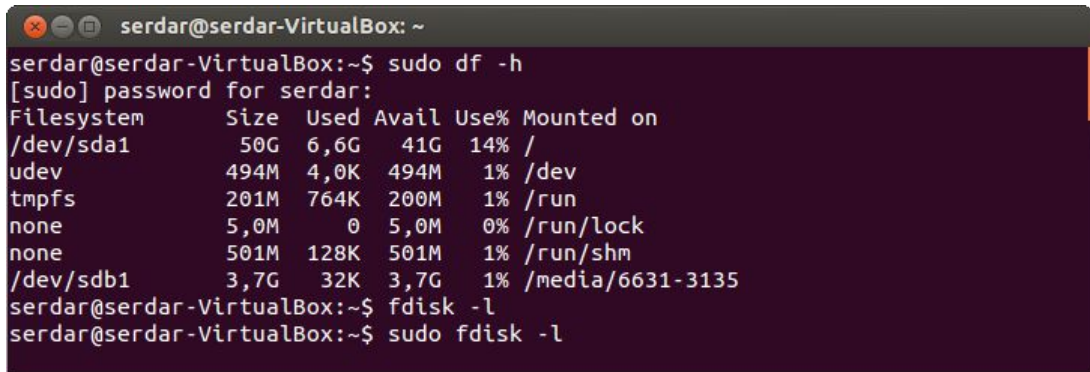
Angström işletim sistemi dağıtımını çeşitli gömülü sistemler için oluşturulmuş Linux dağıtımını olup, ücretsiz ve açık kaynak kodlu olarak sunulmaktadır. Dağıtımlara, <http://narcissus.angstrom-distribution.org/> ve <http://www.angstrom-distribution.org> adreslerinden ulaşılabilmektedir. Yapılan tez çalışmasında, <http://narcissus.angstrom-distribution.org/> web sitesinden, sadece çalışmada kullanı-

lacak özelliklere göre bir dağıtım hazırlanmış ve bu dağıtım kullanılmıştır. Angström işletim sistemi dağıtımına yönelik çalışmalar YOKTO projesi kapsamında, 2014 yılından itibaren <http://www.angstrom-distribution.org> adresinden sunulmaya başlanmıştır.

Angström işletim sistemi dağıtımına ait kurulumun yapılabilmesi için Linux işletim sistemine sahip bir bilgisayarın olması problemsiz bir kurulum yapılması açısından daha uygundur. Yapılan çalışmalarda, Ubuntu 12.04 işletim sistemi dağıtımını kullanılmıştır. Angström işletim sistemi dağıtımına ait kurulum microSD kartın hazırlanması ve sistemin kurulması olmak üzere iki aşamadan meydana gelmektedir. Bu aşamalar alt bölümde sunulmaktadır.

2.2.1. Angström kurulacak olan microSD kartın hazırlanması

Angström işletim sistemi dağıtımını kurmak için 4 GB kapasitesi olan microSD kart kullanılması uygun olmaktadır. Ubuntu işletim sistemi kurulu olan bilgisayar başlandıktan sonra, microSD kart, kart okuyucuya takılarak terminal programı açılmaktadır. Terminal programı üzerinde “sudo df -h” komutu sayesinde diskler ile ilgili kullanım istatistiklerine ulaşılmaktadır. Kullanılan bu komut sayesinde microSD kart ile ilgili çeşitli bilgilere ulaşılabilmektedir. Şekil 2.2'de “/dev/sdb1” olarak gözlemlenen satır microSD kartı ifade etmektedir. MikroSD karta ait toplam boyut, kullanılan ve kullanılabilir boyut, yüzde değeri ve bağlandığı yer bilgileri sunulmaktadır. Genel olarak microSD kart ifade edilirken “sdX” şeklinde ifade edilmektedir. Kullanılan bilgisayarda “sdb” olarak görülse de, farklı bilgisayarlarda “sdc”, “sdf” vb şekilde oluşabilmektedir.



```
serdar@serdar-VirtualBox: ~
serdar@serdar-VirtualBox:~$ sudo df -h
[sudo] password for serdar:
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1        50G   6,6G   41G   14% /
udev             494M   4,0K  494M    1% /dev
tmpfs            201M   764K  200M    1% /run
none             5,0M     0   5,0M    0% /run/lock
none             501M  128K  501M    1% /run/shm
/dev/sdb1        3,7G   32K   3,7G    1% /media/6631-3135
serdar@serdar-VirtualBox:~$ fdisk -l
serdar@serdar-VirtualBox:~$ sudo fdisk -l
```

Şekil 2.2. MikroSD kart bilgileri

Şekil 2.3'de “fdisk -l” komutunun uygulanması sayesinde elde edilen bilgiler yer almaktadır. Kullanılan bu komut sayesinde sabit disk ve harici disklere ait bilgiler görüntülenmektedir. Şekil 2.3'de görülen “sda” sabit diski, “sdb” ise microSD kartı ifade etmektedir. “Sda1”, “sda2” ve “sda5” sabit diskin üç parçaya bölünmüş olduğunu, “sdb1” ise microSD hafıza kartının tek parça olduğunu göstermektedir. Bu komutların kullanılmasındaki amaç, microSD kartın hangi sdX'te olduğunu bulmaya yönelik olarak uygulanmaktadır.

```
serdar@serdar-VirtualBox: ~
serdar@serdar-VirtualBox:~$ sudo fdisk -l

Disk /dev/sda: 55.6 GB, 55579770880 bytes
255 heads, 63 sectors/track, 6757 cylinders, total 108554240 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000b022c

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1 *         2048         106457087   53227520   83   Linux
/dev/sda2           106459134   108552191    1046529    5   Extended
/dev/sda5           106459136   108552191    1046528    82   Linux swap / Solaris

Disk /dev/sdb: 3965 MB, 3965190144 bytes
49 heads, 48 sectors/track, 3292 cylinders, total 7744512 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

   Device Boot      Start         End      Blocks   Id  System
/dev/sdb1           8192         7744511    3868160    b   W95 FAT32
serdar@serdar-VirtualBox:~$ sudo df -h
[sudo] password for serdar:
Filesystem      Size  Used Avail Use% Mounted on
```

Şekil 2.3. MikroSD kart ve sabit disk bilgileri

MicroSD karta Angström işletim sistemi dağıtımının kurulabilmesi için ilk olarak kartın uygun şekilde biçimlendirilmesi gerekmektedir.

Kart iki bölüme ayrılmalı ve ilk bölüm FAT, ikinci bölüm ise EXT olarak biçimlendirilmelidir. FAT biçiminde olan ilk bölüm yaklaşık 70 - 80 MB'lık olarak boot bölümünü oluşturmakta, EXT biçimindeki bölüm ise işletim sistemi dosyaları için kullanılmaktadır. Bu işlemin hızlı bir şekilde yapılabilmesi için “<http://www.angstrom-distribution.org/demo/beagleboard>” adresinde bulunan “mkcard.txt” dosyası içerisinde yer alan betiğin çalıştırılması gerekmektedir. Dosyanın çalıştırılması için aşağıda belirtilen işlemler sırasıyla uygulanmalıdır.

- MikroSD kart daha önceden bölümlendirilmediği ve biçimlendirilmediği için ilk olarak “sudo umount /dev/sdb” komutu ile kart devre dışı bırakılmalıdır.
- mkcard.txt dosyasının bulunduğu dizine geçilmelidir.
- “chmod +x mkcard.txt” komutu uygulanarak mkcard.txt dosyasına çalıştırma izni verilmelidir.
- “sudo ./mkcard.txt /dev/sdb” komutu uygulanarak artık biçimlendirme işlemi gerçekleştirilebilir.

2.2.2. Angström dağıtımının mikroSD karta kurulması

Angström işletim sistemi dağıtımını hazırlamak veya hazır olan dağıtımı bilgisayarınıza indirmek için beagleboard.org, <http://narcissus.angstrom-distribution.org/> ve <http://www.angstrom-distribution.org> siteleri hizmet vermektedir. Sitede yer alan hazır dağıtımların dışında, kullanılacak özellikler seçilerek isteğe özel dağıtımda oluşturulabilmektedir. Oluşturulan dağıtıma ait image dosyası mikroSD karta yazılmakta ve kurulum işlemi yapılmaktadır. İşletim sisteminin sağlıklı bir şekilde kurulması ve çalışması için aşağıda yer alan adımların sırasıyla dikkatli bir şekilde yapılması gerekmektedir.

- İlk olarak Ubuntu kurulu olan bilgisayarda terminal programı kullanarak, indirilen image dosyasının bulunduğu dizine girilmektedir.
- Image dosyası mikroSD kartı üzerinde oluşturulmuş Angström bölümüne açılır. Bu işlem için; “sudo tar -xvz -C /media/Angstrom -f random-0d3cc644-image-beagleboard.tar.gz” komutu uygulanır.
- İşletim sisteminin BeagleBoard-xM’de açılması için mikroSD kartın boot bölümüne sırasıyla, MLO, u-boot.bin ve uImage dosyalarının kopyalanması gerekmektedir. Bu dosyalar, mikroSD kartın Angström bölümüne 2 adımda açılan dizinlerden “boot” dizini içerisinde yer almaktadır.
- Cd /media/Angstrom/boot komutu ile ilgili dizine geçilir.
- sudo cp MLO /media/boot/ komutu kullanılarak MLO dosyası boot bölümüne kopyalanır.
- sudo cp u-boot.bin /media/boot/ ve sudo cp uImage /media/boot/ komutları ile u-boot.bin ve uImage dosyaları kopyalanır.
- BeagleBoard-xM için boot etme özelliklerinin içinde bulunduğu uEnv.txt dosyası hazırlanır.

- Bu dosyanın içerisinde ekran modu, bellek özellikleri vb. bir çok özellik yazılabilir. Hazırlanan uEnv.txt dosyasının içerisinde aşağıda yer alan bilgiler yazılmaktadır.

```
dvimode="hd720 omapfb.vram=0:8M,1:4M,2:4M"
```

```
vram=16M
```

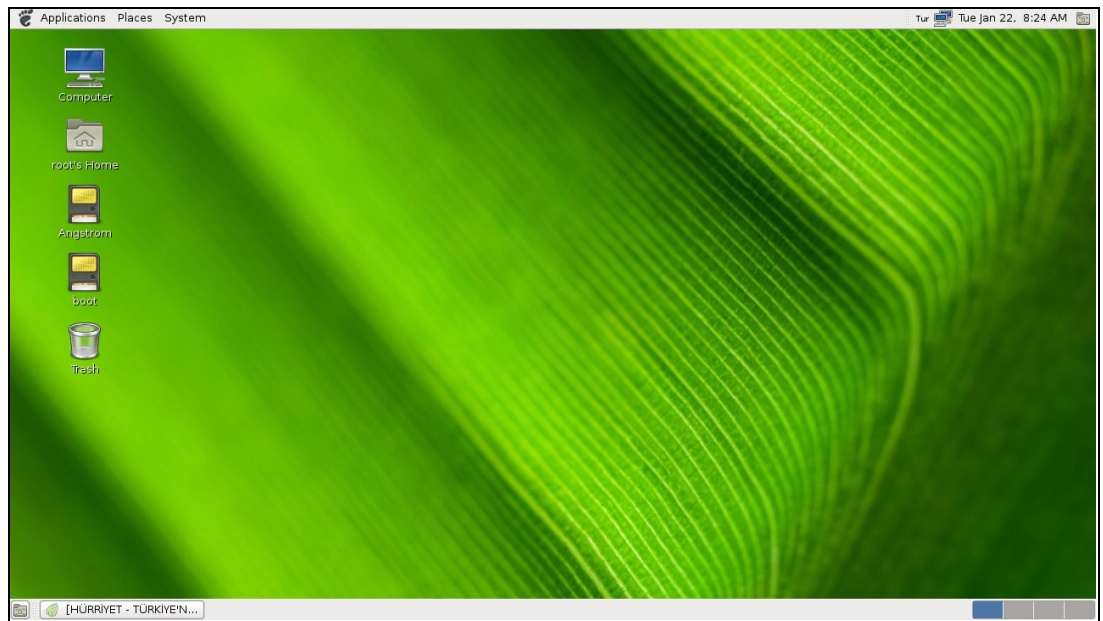
```
optargs="consoleblank=0"
```

```
console="tty0 console=ttyO2,115200n8"
```

- Oluşturulan uEnv.txt dosyası microSD kartın boot bölümüne kopyalanır. Bu işlem için “cp uEnv.txt /media/boot/” komutu uygulanmaktadır.

Yukarıda yazılan işlemler, sırasıyla yaptıktan sonra microSD kart güvenli bir şekilde devre dışı bırakılır ve BeagleBoard-xM üzerinde bulunan mikroSD kart yuvasına takılarak başlatılır. MikroSD kartı devre dışı bırakmak için “ sudo umount /dev/sdb1 ve sudo umount /dev/sdb2” komutlarının uygulanması gerekir.

BeagleBoard-xM açıldıktan sonra Angström işletim sistemi yaklaşık 20-30 dakika içerisinde kurulmaktadır. Angström işletim sistemi dağıtımına ait kurulum bittikten sonra Şekil 2.4'te bulunan görüntü elde edilmektedir. Elde edilen bu görüntü Angström işletim sistemi dağıtımına ait masaüstü ortamını göstermektedir. Program kurulum ve kontrolleri, donanımsal ayarlar ve sistem yönetimi grafik ara yüz sayesinde hızlı bir şekilde gerçekleştirilmektedir.



Şekil 2.4. Angström işletim sistemi masaüstü görünümü

İşletim sistemi, ayrıca özel işlemler için grafik ara yüz kullanılmadan terminal ekranından kontrol edilebilmektedir. Bu sayede grafik ara yüzün getirmiş olduğu ek yük ortadan kaldırılmış olmaktadır.

2.3. OpenCV Kütüphanesi

Yapılan tez çalışmasında, gezgin robotun engelli bir ortamda hareket ettirilmesi, yol planlaması ve gezintisi için görüntü işleme tekniklerinden faydalanılmıştır. Gömülü sistem platformları için uygun olan OpenCV kütüphanesi (Open Source Computer Vision Library) kullanılmıştır. OpenCV kütüphanesi, Windows, Linux, Mac OS, PSP (PlayStation Portable) platformları üzerinde çalışabilen, C diliyle yazılmış, gerçek zamanlı bilgisayarlı görme ve görüntü işleme uygulamaları için kullanılan, açık kaynak kodlu bir kütüphanedir. OpenCV kütüphanesi, Intel tarafından geliştirilmiştir. C, C++, C#, Java, Ruby, Matlab, Python gibi programlama dilleri kullanılarak gerçek zamanlı ve görüntü işleme tabanlı program geliştirmeyi desteklemektedir. İçerisinde 2000'in üzerinde görme algoritması bulunmakta ve ücretsiz olarak dağıtılmaktadır. OpenCV kütüphanesine ait temel bileşenler aşağıda sunulmaktadır [86,87];

- MLL Bileşeni: Bilgisayar öğrenmesi için gerekli istatistiksel verilere ulaşmak, mevcut verileri sınıflandırmak için kullanılan fonksiyonları/araçları içeren bir bileşendir.
- CV Bileşeni: Temel resim işleme fonksiyonları ve bilgisayarlı Görü/Görme için kullanılan yüksek seviyeli algoritmaları bünyesinde barındırmaktadır.
- HighGUI Bileşeni: Slider, form gibi pek çok sayıda nesne yaratarak kullanıcılara etkili bir ara yüz sağlamak amacıyla kullanılmaktadır. Bunların yanı sıra, video, resim dosyalarını yaratmak, kaydetmek, bellekten silmek için kullanılan metodları içerisinde barındırmaktadır.
- CXCore Bileşeni: OpenCV'ye ait özel olarak tanımlanmış veri yapılarını (cvPoint, cvSize, IplImage, cvHistogram, cvMat) ve xml dosya desteğini sağlayan özelliktir.
- CvAux Bileşeni: Deneysel algoritmaları barındırmaktadır. Yüz tanıma, hareket eşleştirme, kamera kalibrasyonu, şekil ve şablon eşleştirme gibi bir çok bilgisayarlı görme algoritmasını desteklemektedir. Bu sayede çok sayıda algoritma hızlı bir şekilde gerçekleştirilebilmektedir.

OpenCV kütüphanesi içerisinde de yer alan bilgisayarlı görme ve görüntü işleme algoritmalarının kullanılmasıyla;

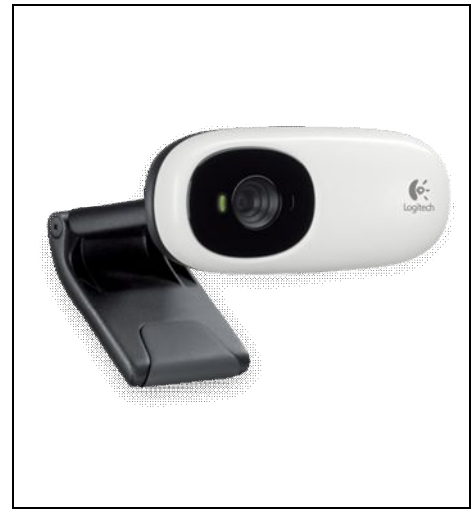
- İnsan-Bilgisayar etkileşimi,
- Nesne belirleme, bölümlenme ve tanıma,
- Yüz tanıma,
- Hareket yakalama, algılama ve takibi,
- Tekli ve çoklu kamera kullanımı,
- Kamera kullanarak derinlik hesaplama,
- Gezgin robot teknolojileri ile ilgili çalışmalar yapılmaktadır.

2.4. Kameralar ve Yardımcı Bağlantı Araçları

Tez çalışması kapsamında, gezgin robotun hareketini sağlamak, ortamda bulunan engel ve hedef gibi nesnelere tespit etmek, ortamın haritasını çıkarmak, mesafe kestirimi yapmak amacıyla 1 adet Kablosuz IP ve 2 adet USB kamera kullanılmıştır. Kablosuz IP kamera, tez çalışmasının ilk aşamalarında, deney ortamının tavanına takılarak ortam görüntülerinin alınması ve işlenmesi amacıyla kullanılmıştır. Ayrıca, Kablosuz IP kamera ve BeagleBoard-xM arasındaki veri haberleşmesi için 1 adet kablosuz modem kullanılmaktadır. USB kameralar ise, robotun üzerine takılarak ortam görüntülerinin elde edilmesi amacıyla kullanılmaktadır. Şekil 2.5'te kullanılan kameralara ait görüntüler sunulmaktadır.



(a)



(b)

Şekil 2.5. Kameralar, (a) Kablosuz IP kamera (b) USB kamera

Tablo 2.2’de çalışmada kullanılan kablosuz IP kameraya ait teknik özellikler sunulmuştur.

Tablo 2.2. Kablosuz IP kameranın teknik özellikleri

Özellik Adı	Özelliği
Algılayıcı	1,3 Megapiksel CMOS algılayıcı
Odak uzaklığı	1,7mm f / 2,4
Görüş Açısı	74,8 ⁰
Kablosuz Ağ	IEEE 802.11 b/g/n
Bellek	4MB Flash / 32 MB SDRAM
Boyutları / Ağırlığı	65mm x 80mm x 27mm / 86 gr
Arabirimler	1 Ethernet / 1 Güç / 1 Reset / 1 Anten
Güç	5 Volt DC
Video ve Görüntü Formatı	M-JPEG, AVI, JPEG
Çözünürlük	SXVGA(1280 x 960) / 30 fps
Fiyat	70\$

Tablo 2.3’te ise tez çalışmasında kullanılan USB kameralara ait teknik özellikler sunulmuştur.

Tablo 2.3. USB Kameranın teknik özellikleri

Özellik Adı	Özelliği
Algılayıcı	1,3 Megapiksel CMOS algılayıcı
Odak uzaklığı	2,3mm
Görüş açısı	68 ⁰
Bağlantı birimi	USB Kablo
Boyutları / Ağırlığı	40,4mm x 68,5mm x 31,75mm / 88 gr
Güç	5 Volt DC
Çözünürlük	640x480 / 30 fps
Fiyat	10\$

Tez çalışması kapsamında, BeagleBoard-xM ile gezgin robot arasındaki iletişim seri arabirim sayesinde gerçekleştirilmiştir.

Seri arabirimin gerçekleştirilmesi için standart olamayan bir seri arabirim kablosu kullanılmıştır. BeagleBoard-xM cihazına ait seri arabirim bilgisayar ile haberleşecek şekilde hazırlanmıştır. Gezgın robot üzerinde yer alan seri arabirim de aynı şekilde bilgisayar ile haberleşecek şekilde özel bir kablo hazırlanarak iletişim gerçekleştirilmiştir.

BeagleBoard-xM kartı, 5 Volt ile çalışmaktadır. Mobil olarak çalışmadığı durumlarda, BeagleBoard-xM üzerinde yer alan enerji girişı sayesinde çalışmalar yapılabilir. Ancak, BeagleBoard-xM gezgın robot üzerine takılarak mobil olarak dolaştırılması gerektiğinde 5 Voltluk bataryaya ihtiyaç duyulmaktadır. BeagleBoard-xM'in gezgın robot üzerinde mobil olarak dolaştırılması sırasında 5 Volt 1 Amperlik (A) "Power Bank" kullanılmıştır. Yapılan deneysel çalışmalarda, 20.000mA kullanılarak yaklaşık olarak on saat çalıştığı gözlenmiştir.

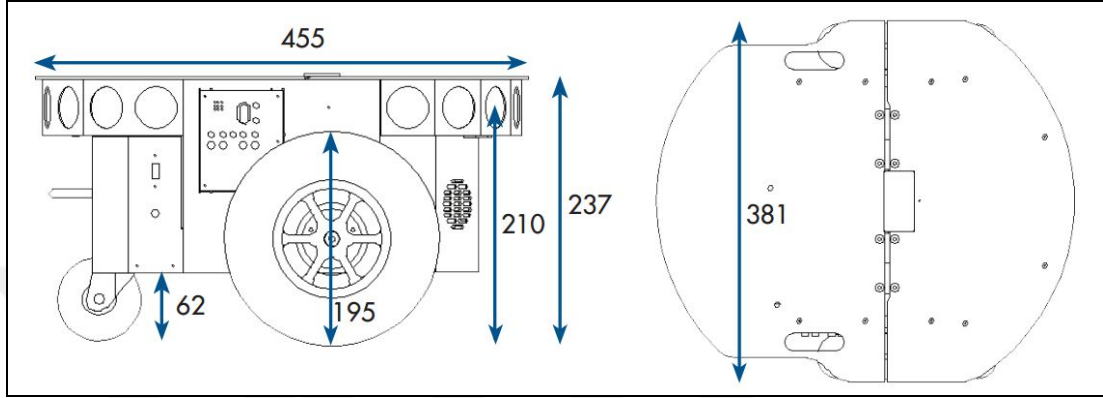
2.5. Pioneer 3 – DX Gezgın Robotu

Tez çalışması kapsamında yapılan çalışmalarda, laboratuvar ortamında hazır bulunan ve "Adept MobileRobots" firması tarafından üretilen Pioneer 3-DX (P3-DX) gezgın robotu kullanılmıştır. Son yıllarda yapılan akademik çalışmalarda, gezgın robotların yol planlama, konum belirleme, engel sakınım, çoklu robot problemleri, robotların bir biri ile koordineli hareketi gibi konularda P3-DX gezgın robotu deneysel çalışmalarda yaygın olarak kullanılmaktadır. P3-DX gezgın robotunun kapalı ortam veya laboratuvar ortamlarında araştırma ve geliştirme çalışmaları yapmak için kullanımı uygundur. Robot ön tarafından 2 büyük, arka tarafında 1 küçük olmak üzere 3 adet tekerleğe sahiptir. Şekil 2.6'da P3-DX gezgın robotuna ait resim sunulmaktadır.



Şekil 2.6. P3-DX gezgın robotu

P3-DX gezgin robotunun üzerinde standart olarak, açma kapama düğmesi, seri iletişim arabirimi, sıfırlama (resetleme) düğmesi, motor düğmesi, AUX1 ve AUX2 düğmeleri ve kontrol bilgilerini gösteren, batarya, çalışma ve iletişim durum ışıkları bulunmaktadır. Ayrıca robotun ön kısmında 8 adet sonar algılayıcı bulunmaktadır. Şekil 2.7’de P3-DX gezgin robotuna ait boyutlar sunulmaktadır.



Şekil 2.7. P3-DX gezgin robotunun boyutları [88]

P3-DX gezgin robotu 9 kg ağırlığında olup 23 kg’a kadar yük taşıma kapasitesi bulunmaktadır. P3-DX’in beslenmesi için kullanılan 3 adet batarya mevcuttur. P3-DX gezgin robotuna ait standart teknik özellikler Tablo 2.4’te sunulmuştur. Bu özelliklerin yanı sıra isteğe bağlı olarak üretici firmadan ek talepler de yapılabilmektedir.

Tablo 2.4. P3-DX gezgin robotuna ait özellikler

Özellik Adı	Özelliği
Uzunluğu	45cm (18 inch)
Genişliği	40cm (16 inch)
Yüksekliği	24,5cm (9,5 inch)
Ağırlığı ve taşıyacağı yük	9Kg / 23Kg
Ortam / Sıcaklık	Kapalı Ortam / 0 – 35 °C
Güç ve batarya kapasitesi / toplam	12 V / 9 Ah / 27 Ah
Batarya dolum süresi	2,4 Saat (Hızlı) / 12 Saat (Standart)
İletişim	RS232 – Seri Port veya USB – Seri Port
Sonar algılayıcı	8 adet önyüz (5 metre mesafe)

P3-DX gezgin robotu Robot İşletim Sistemi (Robot Operating System - ROS) kullanılarak ta yönetilebilmektedir. P3-DX gezgin robot platformu için ayrıca, açık kay-

nak kod ile yazılmış, Java, C, Python, Matlab, programlama dillerine desteği olan Linux ve Windows işletim sistemlerinde çalışabilen ARIA (Advances Robotics Interface for Applications) kütüphanesi bulunmaktadır. Ayrıca, MobileSim, MobileEyes, Mapper3 ve ARNL gibi yazılımlar ile harita oluşturma, benzetim çalışmaları, gezgin robotun kontrol edilmesi, görüntüleme ve yapılandırma işlemlerinin yapılması, geliştirilen yazılımların robot üzerinde test edilmesi işlemleri yapılabilmektedir.

Yapılan tez çalışmasında, P3-DX gezgin robotunun kontrolü, BeagleBoard-xM TKB ile gerçekleştirilmektedir. BeagleBoard-xM, P3-DX'e istekte bulunan istemci olarak davranmakta, P3-DX üzerinde yer alan sunucu ile haberleşmektedir. Aynı şekilde P3-DX'ten BeagleBoard-xM'e olan iletişimde sunucu istemci arası iletişim, Sunucu Bilgi Paketleri (Server Information Packets - SIPs) ile gerçekleştirilmektedir [4, 89]. BeagleBoard-xM ile P3-DX arasında kullanılan haberleşme paketlerinde Başlık, Bayt Sayısı, Komut Numarası, Argüman tipi, Argüman ve Sağlama bilgileri bulunur.

Tablo 2.5'te bu paket içerisinde yer alan bilgilere ait özellikler sunulmuştur.

Tablo 2.5. BeagleBoard-xM P3-Dx iletişim paketine ait bilgiler ve özellikleri

Paket Bileşeni	Boyut (Bayt)	Değer	Tanım
Başlık (Header)	2 Bayt	250 (0xFA), 251 (0xFB)	Paket Başlığı, istemci ve sunucu için aynıdır.
Bayt Sayacı	1 Bayt	N(1 - 249)	Başlık ve kendisi hariç toplam bayt sayısı
Komut Numarası	1 Bayt	0 - 255	İstemci Komut Numarası (istekte bulunulan işlem)
Argüman Tipi	1 Bayt	59(0x3B), 43(0x2B), 27(0x1B)	Robotun ileri, geri veya sola, sağa manevra yapması için kullanılan bilgidir.
Argüman	N Bayt	Veri	Genellikle 2 Bayt tamsayı
Sağlama (Checksum)	2 Bayt	Hesaplanan Değer	Paket bütünlüğünü sağlamak için kullanılır.

BeagleBoard-xM ile P3-DX arasındaki iletişimin doğru bir şekilde yapıldığını kontrol etmek amacıyla paket sonunda 2 Baytlık sağlama kullanılmaktadır. Kullanılan sağlama işlemi veya kontrolün ilk baytının elde edilmesi, komut numarası ile argümanı oluşturan ilk baytın toplanması sonucunda oluşmaktadır. İkinci bayt ise, argüman tipi ile argümana ait ikinci baytın toplanması sonucunda elde edilmektedir.

Tablo 2.6’da istemci tarafından gönderilen bir paket örneği sunulmuştur. Tabloda bayt numaraları, paket içerikleri, ilgili paket içeriğinin hangi bileşene ait olduğu ve bununla ilgili özel durum, tanımlama veya oluştururken kullanılan yöntem ifade edilmektedir. Tablo 2.6’da sunulan örnek içerik paketinde gönderilen veri, P3-DX gezgin robotun 1000 mm (mili metre) geriye doğru gitmesini sağlamaktadır. Komut numarası olarak 8 robotun hareket etmesini, 12 ise robotun dönmesini sağlamak için kullanılmaktadır. Komut numarası olarak 8, argüman tipi olarak 59 uygulandığında ileri, 27 uygulandığında geriye doğru hareket etmektedir. Ayrıca, komut numarası 12 olup 59 uygulandığında gezgin robot sağa, 27 uygulandığında ise sola doğru dönüş hareketini yapmaktadır. Hareket komutu uygulandığında, argüman olarak gönderilen 2 Baytlık veri robotun gideceği mesafeyi, dönme komutu uygulandığında ise robotun dönüş açısını ifade etmektedir. Argümanın ilk baytı düşük anlamlı kısmı, ikinci baytı ise yüksek anlamlı kısmı ifade etmektedir.

Tablo 2.6. İstemci tarafından gönderilen iletişim paket örneği

Bayt Numarası	Paket İçeriği	Bileşen Adı	Özel Durum veya Tanımlama
1	250	Başlık	Paket Başlangıcı
2	251	Başlık	Paket Başlangıcı
3	6	Bayt Sayacı	6 Baytlık veri gelecek
4	8	Komut Numarası	Gezgin robotun ileri veya geriye doğru hareket edeceğini gösteriyor.
5	27	Argüman Tipi	Gezgin robotun geriye gideceğini belirtiyor.
6	232	Argüman	Verinin düşük anlamlı kısmı
7	3	Argüman	Verinin yüksek anlamlı kısmı
8	240	Sağlama	Komut numarası ile Argüman 1.baytının toplanması
9	30	Sağlama	Argüman tipi ile Argüman 2.baytının toplanması

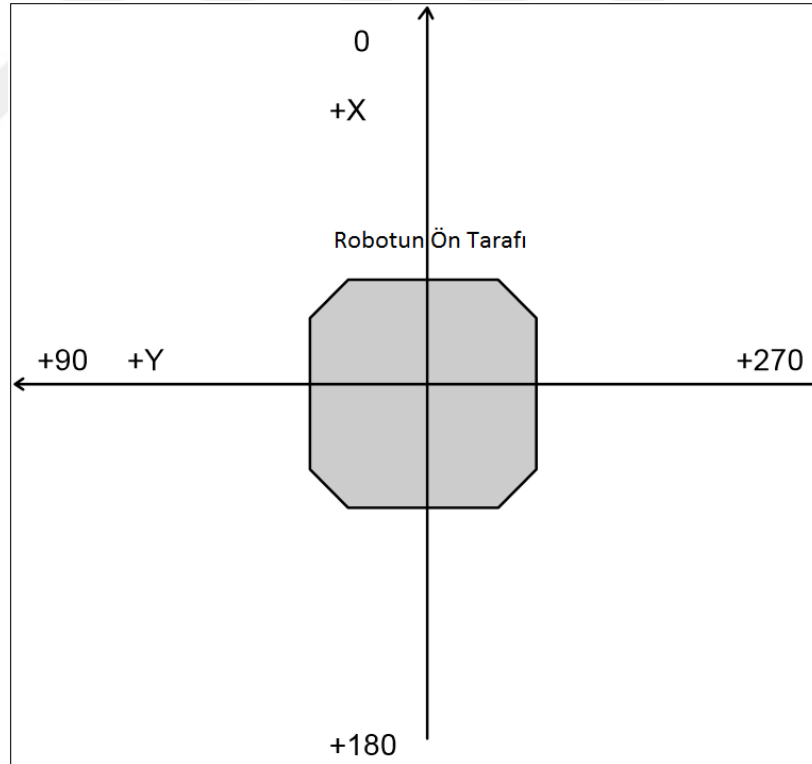
İstemci ile sunucu haberleşmesinde, bağlantı kurulmadan önce gönderilmesi gereken senkronizasyon paketleri bulunmaktadır. Bu paketler gönderildikten sonra bağlantı sağlanmakta ve komutlar gezgin robot sunucusunu harekete geçirmektedir. Senkronizasyon paketleri SYNC0, SYNC1, SYNC2 olarak isimlendirilmiştir. Tablo 2.7’de senkronizasyon paketlerine ait içerikler sunulmaktadır.

Tablo 2.7. Senkronizasyon paketlerine ait içerik bilgisi

Paket Adı	Paket İçeriği	Özelliği
SYNC0	250 – 251 – 3 – 0 – 0 – 0	Bağlantıyı başlatmak için kullanılan paketlerdir. Sırasıyla gezgin robota gönderilmektedir.
SYNC1	250 – 251 – 3 – 1 – 0 – 1	SYNC2 paketinden sonra bağlantı hazır duruma gelir.
SYNC2	250 – 251 – 3 – 2 – 0 – 2	

Şekil 2.8’de P3-DX’e ait iç koordinat sistemi sunulmuştur. P3-DX içerisinde yer alan sunucu, istemciye göndermiş olduğu SIPs içerisinde robot konumunu bildiren X, Y koordinatları ve derece cinsinden robotun açı bilgisi de bulunmaktadır.

P3-DX gezgin robotun iç koordinat sisteminde +X koordinatı 0^0 , +Y koordinatı ise 90^0 ’yi ifade etmektedir. Robotun başlangıç pozisyonu $R_{\text{pozisyon}}(0,0,0)$ olarak belirlenmektedir. Gezgin robotun hareketinden sonra, robotun yeni konumu ve açısı $R_{\text{pozisyon}}(X,Y,\theta)$ olarak güncellenmektedir.



Şekil 2.8. P3-DX gezgin robotu iç koordinat sistemi[89]

Robot iç koordinatlarının sıfırlanması gerektiği durumda 7 numaralı komut (SETO), P3-DX sunucusuna gönderilmelidir. P3-DX gezgin robotu çalışma ortamında hedefe ulaşması için gereken yolu gittikten sonra işlemini tamamlamaktadır. İşlemini ta-

mamlayan robota, üzerinde bulunan “Reset” düğmesi kullanılarak sıfırlamak gerekmektedir. Ancak robotun uzaktan kontrol edilmesi sırasında insan müdahalesi olmadan sıfırlanması için 253 numaralı komut (RESET) uygulanmalıdır. Yapılan çalışmalarda, robotun uzaktan kontrol edilmesi sırasında sıfırlama işlemi bu komut sayesinde başarılı bir şekilde gerçekleştirilmiştir.



3. YÖNTEM

Bu bölümde, yapılan tez çalışmasında kullanılan görüntü işleme teknikleri ve yol planlama algoritmaları hakkında detaylı bilgiler sunulmaktadır.

3.1. Kullanılan Görüntü İşleme Yöntemleri

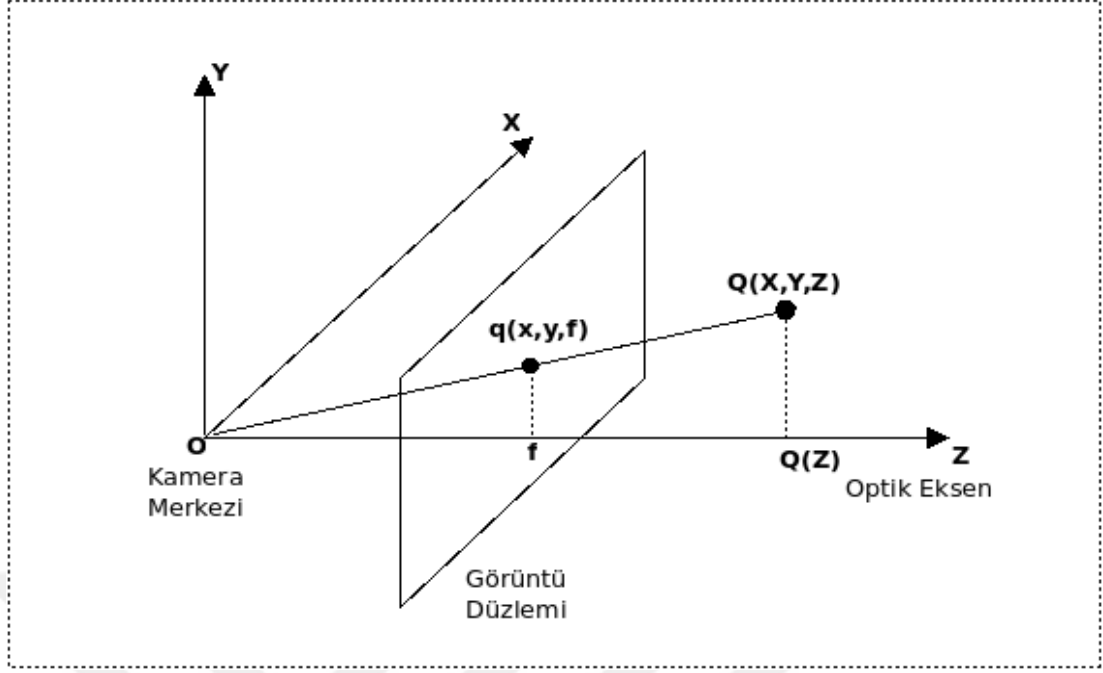
Gezgin robotun engel sakınım, yol planlama ve konum belirleme problemlerinde ortamda yer alan robot, engel ve hedeflerin bulunması aşamasında görüntü işleme teknikleri kullanılmaktadır. Gömülü görme ve görüntü işlemede etkili bir yöntem olan OpenCV Kütüphanesi kullanılarak işlem gerçekleştirilmektedir. Bu alt bölümde, çalışmada kullanılan görüntü işleme teknikleri sunulmaktadır.

3.1.1. İğne delik kamera perspektif modeli

Günlük hayatta ucuz ve kolay ulaşılabilir olması sebebiyle iğne deliği (pinhole) kameralar görüntü işleme uygulamalarında yaygın olarak kullanılmaktadır [90]. Kamera kullanılarak üç boyutlu gerçek dünyadan elde edilen görüntülerin iki boyutlu düzlem üzerinde yer alan iz düşümüne "Kamera Perspektif Modeli (KPM)" adı verilmektedir. Görüntü işleme teknikleri kullanılarak resmin işlenmesinden önce kullanılan kameraların, perspektif modelinin anlaşılması gerekmektedir. İğne delik kamera-ya ait KPM Şekil 3.1'de sunulmuştur.

Şekil 3.1'de sunulan KPM, kamera merkezine ait odak noktası, koordinatların yer aldığı optik eksen, görüntü düzlemi, üç boyutlu gerçek dünyada yer alan Q noktası ve bu noktanın görüntü düzlemi üzerine iz düşümünden oluşmaktadır.

Şekil 3.1'de Q noktasının gerçek dünyadaki koordinatları $Q(X,Y,Z)$ şeklinde ifade edilmektedir. Q noktasının kamera üzerinde yer alan görüntü düzlemindeki iz düşümü q ile ifade edilmektedir. Görüntü düzlemi ile kamera merkezi arasında odak noktası (f) kadar mesafe bulunmaktadır. Bu durumda gerçek dünyada yer alan $Q(X,Y,Z)$ noktasının görüntü düzlemi üzerindeki koordinatları $q(x,y,f)$ olarak ifade edilmektedir.



Şekil 3.1. İğne delik KPM [90]

Şekil 3.1’de yer alan KPM’e göre $OQQ(Z)$ ve Oqf üçgenleri arasındaki benzerlik ilişkileri kullanılarak Eşitlik (3.1), Eşitlik (3.2), Eşitlik (3.3) ve Eşitlik (3.4) elde edilmektedir.

$$\frac{f}{Q(Z)} = \frac{q(x)}{Q(X)} \rightarrow q(x) = f \left(\frac{Q(X)}{Q(Z)} \right) \quad (3.1)$$

$$\frac{q(x)}{f} = \frac{Q(X)}{Q(Z)} \rightarrow q(x) = f \left(\frac{Q(X)}{Q(Z)} \right) \quad (3.2)$$

$$\frac{f}{Q(Z)} = \frac{q(y)}{Q(Y)} \rightarrow q(y) = f \left(\frac{Q(Y)}{Q(Z)} \right) \quad (3.3)$$

$$\frac{q(y)}{f} = \frac{Q(Y)}{Q(Z)} \rightarrow q(y) = f \left(\frac{Q(Y)}{Q(Z)} \right) \quad (3.4)$$

Elde edilen eşitliklerde gerçek dünyada yer alan Q noktasının, görüntü düzlemi üzerindeki iz düşümleri olan $q(x)$ ve $q(y)$ değerleri elde edilmektedir. Eşitlik (3.1) ve (3.2)’de yer alan denklemlerin her ikisinde de $q(x)$ elde edilirken, Eşitlik (3.3) ve (3.4)’de $q(y)$ elde edilmektedir.

Elde edilen bu eşitlikler kullanılarak, gerçek dünyada kamera karşısında yer alan Q noktasının (X,Y,Z) , görüntü düzlemi üzerinde yer alan q noktasının (x,y) koordinatları ile eşleşmesi sağlanmaktadır[91,92]. Eşitlik (3.5) iki nokta arasındaki eşleşmeyi sunmaktadır.

$$Q(X,Y,Z) \rightarrow q(x,y) \quad Q(X,Y,Z) \rightarrow q\left(f\frac{X}{Z}, f\frac{Y}{Z}\right) \quad (3.5)$$

Gerçek dünyada yer alan nesneye ait $Q(X,Y,Z)$ noktası ile görüntü düzlemi üzerinde yer alan $q(x,y)$ noktasının homojen vektörler şeklinde ifade edilmesi durumunda, bu iki nokta birbiri ile doğrusal olarak eşleşmektedir[92]. Ayrıca, üç boyutlu gerçek dünyadan iki boyutlu düzleme geçilirken $3*4$ boyutlarında kamera izdüşümü matrisi (P) adı verilen matris kullanılmaktadır. Burada, homojen uzayda doğrusal eşleşme yapılırken, w katsayısı uygulandığında Eşitlik (3.6)'da sırasıyla verilen vektörler elde edilmektedir. Eşitlik (3.7)'de kamera iz düşüm matrisleri sunulmuştur.

$$\begin{bmatrix} x \\ y \\ w \end{bmatrix} \rightarrow \left(\frac{x}{w}, \frac{y}{w}\right) \quad \begin{bmatrix} X \\ Y \\ Z \\ w \end{bmatrix} \rightarrow \left(\frac{X}{w}, \frac{Y}{w}, \frac{Z}{w}\right) \quad (3.6)$$

$$P = \begin{bmatrix} f_x & 0 & 0 & 0 \\ 0 & f_y & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (3.7)$$

Görüntü düzlemindeki genişliğe ait toplam piksel sayısı x_t ve yüksekliğe ait toplam piksel sayısı y_t olarak ifade edilmektedir. Bu durumda, kamera iz düşüm matrisine ait yeni değerler hesaplanırken Eşitlik (3.8) kullanılmaktadır.

$$f_x = \frac{f}{x_t} \quad f_y = \frac{f}{y_t} \quad (3.8)$$

Görüntü düzlemi ile optik eksenin kesişme noktası, görüntü düzlemine ait x ve y koordinatlarının merkez noktası olarak kabul edilmektedir. Kamera görüntüsü ile, görüntü düzlemi merkezi arasında kayma olmaktadır. Eşitlik (3.7)'de sunulan P kamera iz düşüm matrisi kayma olmadığı varsayılarak yazılmaktadır.

Kaymanın olduğu varsayılarak Eşitlik (3.9)'da verilen kamera iz düşüm matrisi oluşturulmaktadır. Matris içerisinde yer alan, K_x ve K_y görüntü düzleminde bulunan x ve y koordinatlarındaki kayma miktarlarını ifade etmektedir.

$$P = \begin{bmatrix} f_x & 0 & K_x & 0 \\ 0 & f_y & K_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (3.9)$$

Eşitlik (3.6)'da kullanılan w katsayısının 1 olması durumunda Eşitlikler tekrar düzenlenmiş ve Eşitlik (3.10) elde edilmiştir.

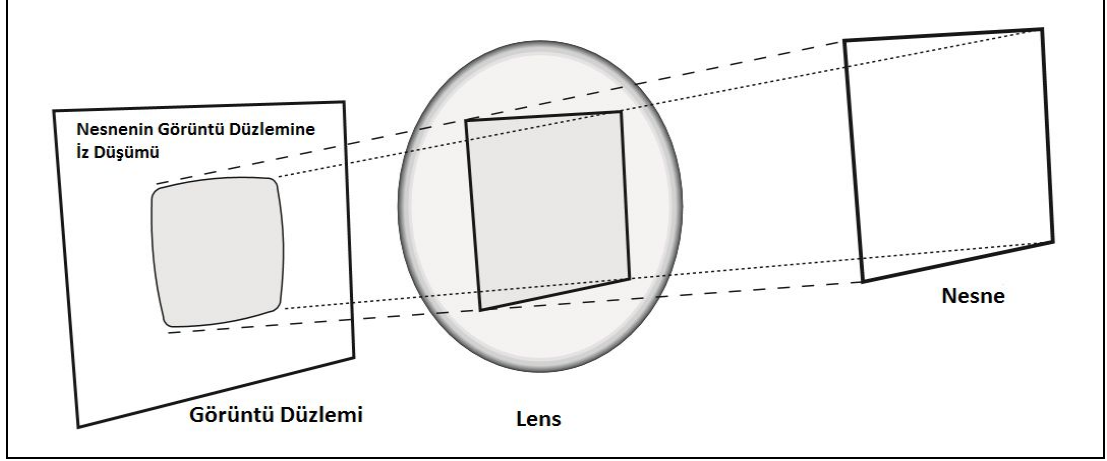
$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \rightarrow Z \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = P \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \rightarrow Z \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & K_x & 0 \\ 0 & f_y & K_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.10)$$

Görüntü düzleminde yer alan q noktasına ait x ve y koordinatları gerçek dünyada bulunan Q noktasına ait X,Y ve Z koordinatları ve kamera iz düşüm matrisi kullanılarak, Eşitlik (3.11)'de verilen denklemler elde edilmektedir.

$$x = f_x \frac{X}{Z} + K_x \quad y = f_y \frac{Y}{Z} + K_y \quad (3.11)$$

3.1.2. Kamera kalibrasyonu

Kamera Kalibrasyonu, kamera kullanılarak yapılan çalışmalarda önemli bir yer tutmaktadır. Genel olarak üç boyutlu gerçek dünyada bulunan bir cisimden alınan noktanın, iki boyutlu görüntü düzlemi üzerinde yer alan iz düşümünün doğru tespit edilmesi için kamera kalibrasyonu yapılmaktadır. Ayrıca, günlük hayatta kullanılan kameraların maliyetlerinin de düşmesiyle kullanılan kamera lenslerinin yapısından kaynaklanan bozulmalar meydana gelmektedir. Kamera kalibrasyonu yapılmadan uygulanacak olan görüntü işleme tekniklerinde hatalar oluşacağından, kalibrasyon işlemi görüntü işlemeye başlamadan önce mutlaka yapılmalıdır. Kamera lensinin oval yapısı sebebiyle, lens kenarından geçen ışınlar kırılmaya uğramakta ve radyal bozulmalara neden olmaktadır. Şekil 3.2'de kamera lensine ait radyal bozulma sunulmuştur.



Şekil 3.2. Radyal bozulma [87]

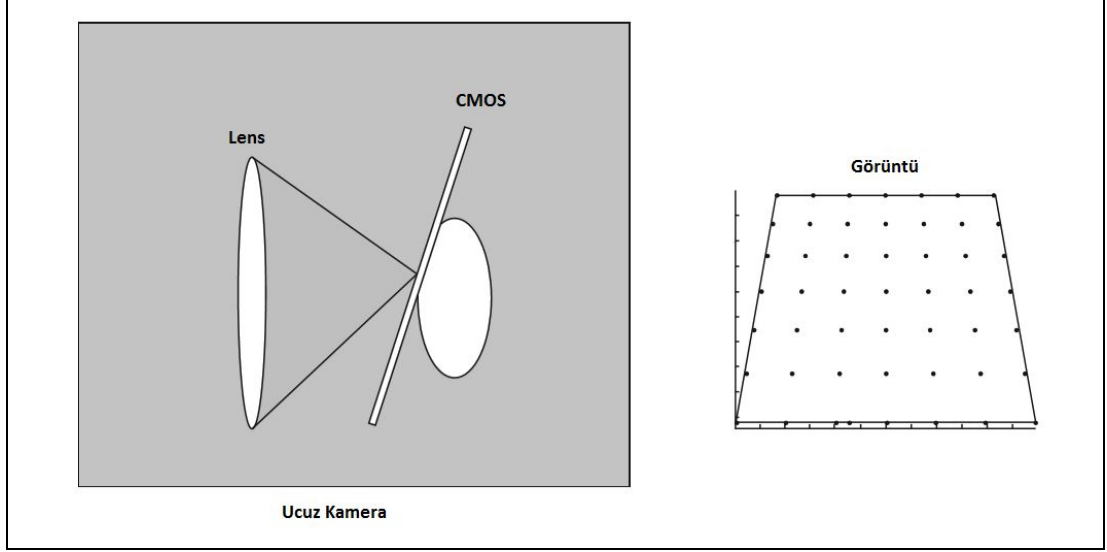
Şekil 3.2’de kare şeklinde bir nesnenin kamera lensinden geçirilerek, görüntü düzlemi üzerinde düşen iz düşümü verilmiştir. Lens’in kenarları yuvarlak olduğu için nesnenin kenarlarında bozulmalar oluşmaktadır. Burada, ideal bir kamera kullanıldığında, radyal bozulma olmadığında herhangi bir noktaya ait koordinatlar x ve y şeklinde ifade edilmektedir. Ancak bozulma olduğunda koordinatlar, x_D ve y_D olarak ifade edilmektedir. Bu durumda, x_D ve y_D Eşitlik (3.12) kullanılarak hesaplanmaktadır[87].

$$\begin{aligned}
 x_D &= x + xk_1r^2 + xk_2r^4 + xk_3r^6 & x_D &= x[1 + k_1r^2 + k_2r^4 + k_3r^6] \\
 y_D &= y + yk_1r^2 + yk_2r^4 + yk_3r^6 & y_D &= y[1 + k_1r^2 + k_2r^4 + k_3r^6]
 \end{aligned}
 \tag{3.12}$$

Eşitlik (3.12)’de verilmiş olan k_1 , k_2 ve k_3 radyal bozulma katsayıları olarak tanımlanmaktadır. Genellikle ucuz web kameralarında, k_1 ve k_2 katsayıları kullanılmaktadır. Radyal bozulmanın fazla olduğu, balıkgözü kameralarda, k_3 katsayısı da kullanılmaktadır. Bozulma çok küçük seviyede olmaktadır. Bu nedenle, Taylor Serisinin $r=0$ katsayısı etrafından genişletilerek, bozulma katsayıları bulunmaktadır.

Kamera çalışmalarında karşılaşılan bir diğer bozulma türü ise yüzeysel bozulma (tangential distortion) olarak bilinmektedir.

Özellikle, kullanılan ucuz kamera lensinin üretim aşamasındaki hatalarından dolayı, görüntü düzlemine tam olarak paralel olmamasından kaynaklanan bozulma türüdür. Şekil 3.3’te ucuz bir kameraya ait lensin görüntü düzlemine olan durumu sunulmaktadır.



Şekil 3.3. Ucuz web kamerasına ait yüzeysel bozulma [87]

Şekil 3.3'te görüldüğü üzere, görüntü düzlemi, kamera lensine paralel olmadığından görüntüde yüzeysel bozulmalar meydana gelmektedir. Eşitlik (3.13)'de x_D ve y_D sadece yüzeysel bozulma olduğu varsayılarak tekrar düzenlenmiştir [87].

$$x_D = x + [2p_1xy + 2p_2x^2 + p_2r^2] \quad (3.13)$$

$$y_D = y + [2p_2xy + 2p_1y^2 + p_1r^2]$$

Eşitlik (3.13)'te yer alan p_1 ve p_2 katsayıları yüzeysel bozulma katsayılarını ifade etmektedir. Kameradan elde edilen görüntü üzerinde, radyal ve yüzeysel bozulma olduğu durumda x_D ve y_D koordinatları Eşitlik (3.14) ve (3.15)'de verilen denklemler kullanılarak elde edilmektedir [87].

$$r = \sqrt{x^2 + y^2} \quad (3.14)$$

$$x_D = x + x[k_1r^2 + k_2r^4 + k_3r^6] + [2p_1xy + 2p_2x^2 + p_2r^2] \quad (3.15)$$

$$y_D = y + y[k_1r^2 + k_2r^4 + k_3r^6] + [2p_2xy + 2p_1y^2 + p_1r^2]$$

Radyal ve yüzeysel bozulma katsayıların 5×1 boyutlarında bir dizi şeklinde ifade edilecek olduğunda Eşitlik (3.16) elde edilmektedir[87].

$$D_K = [k_1 \quad k_2 \quad p_1 \quad p_2 \quad k_3] \quad (3.16)$$

Üç boyutlu gerçek dünyada yer alan nesneye ait herhangi bir noktanın iki boyutlu görüntü düzlemi üzerindeki yerini belirlemek için, Eşitlik (3.17) kullanılmalıdır [87].

$$q(x, y) = M_I M_D Q(X, Y, Z) \quad \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = M_I [R|T] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.17)$$

Eşitlik (3.17)'de sunulan M_I iç parametre matrisini, M_D ise dış parametre matrisini ifade etmektedir. M_I içerisinde odak noktasına ait koordinatlar ve görüntü düzlemi ile optik eksen arasındaki kayma miktarına ait x ve y değeri yer almaktadır. M_D içerisinde ise dönüşüm matrisi (R) ve öteleme vektörü (T) değerleri yer almaktadır. Eşitlik (3.18)'de R matrisi ve T vektörü sunulmuştur [87].

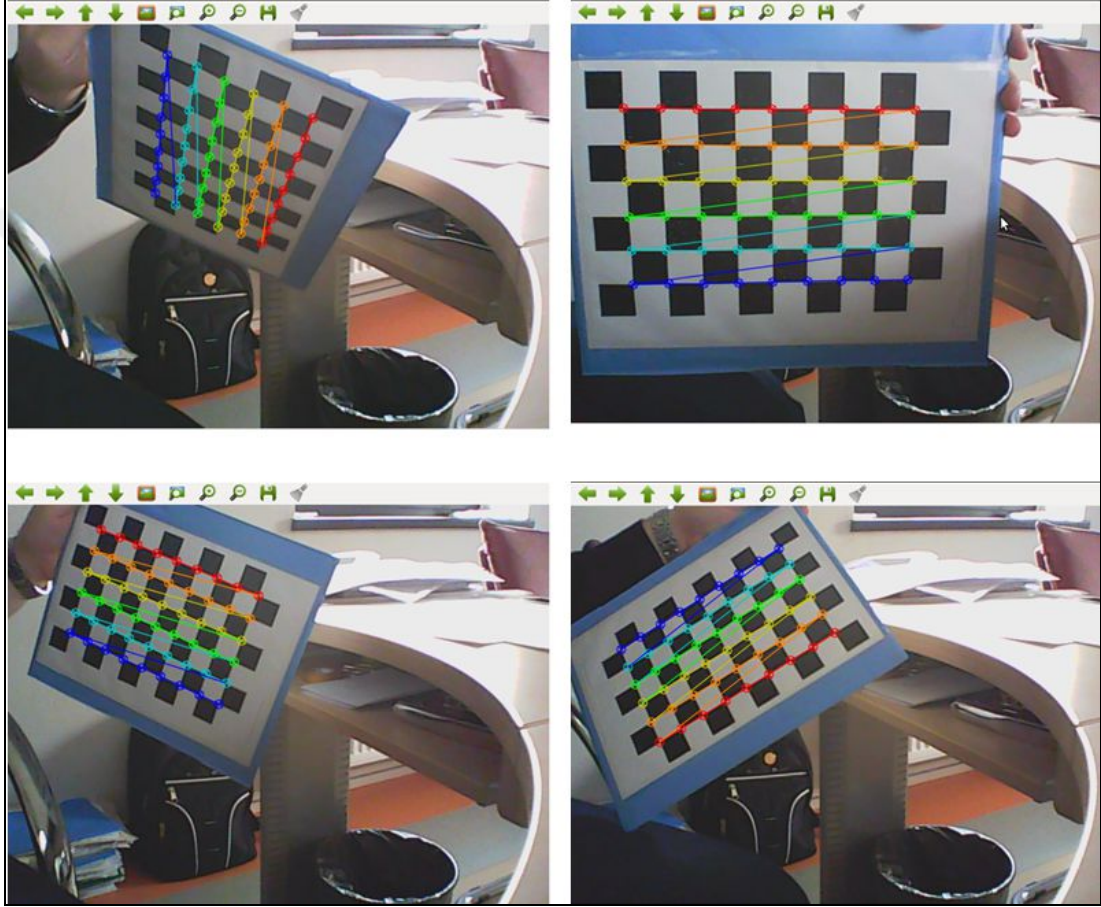
$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad T = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix} \quad (3.18)$$

Eşitlik (3.19)'da, gerçek dünyada yer alan Q noktasının görüntü düzlemi üzerindeki iz düşümünü, iç parametreler, radyal, yüzeysel bozulmalar, dönüşüm matrisi ve öteleme vektörü ile ifade edilmiş hali verilmiştir [87].

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & K_x & 0 \\ 0 & f_y & K_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3.19)$$

Kamera kalibrasyon işlemi yapılırken, dama veya satranç tahtasına benzeyen Tsai ızgarası kullanılmaktadır. Tsai ızgarası, satranç tahtası üzerinde yer alan karelerden oluşmaktadır. Kalibrasyon işlemi sırasında, Tsai ızgarasına ait kamera tarafından farklı uzaklık ve açılardan alınan görüntüler kullanılmaktadır. Gerçek zamanlı uygulamalar için farklı yer ve açılardan alınan görüntülerin kullanılması farklı durumların değerlendirilmesi açısından önemlidir. Kalibrasyon işlemi sırasında, Tsai ızgarasına ait görüntüde yer alan tüm köşelerin başarılı bir şekilde bulunması gerekmektedir. Başarılı olarak tüm köşelerin bulunduğu her görüntüye ait değerler kaydedilmektedir.

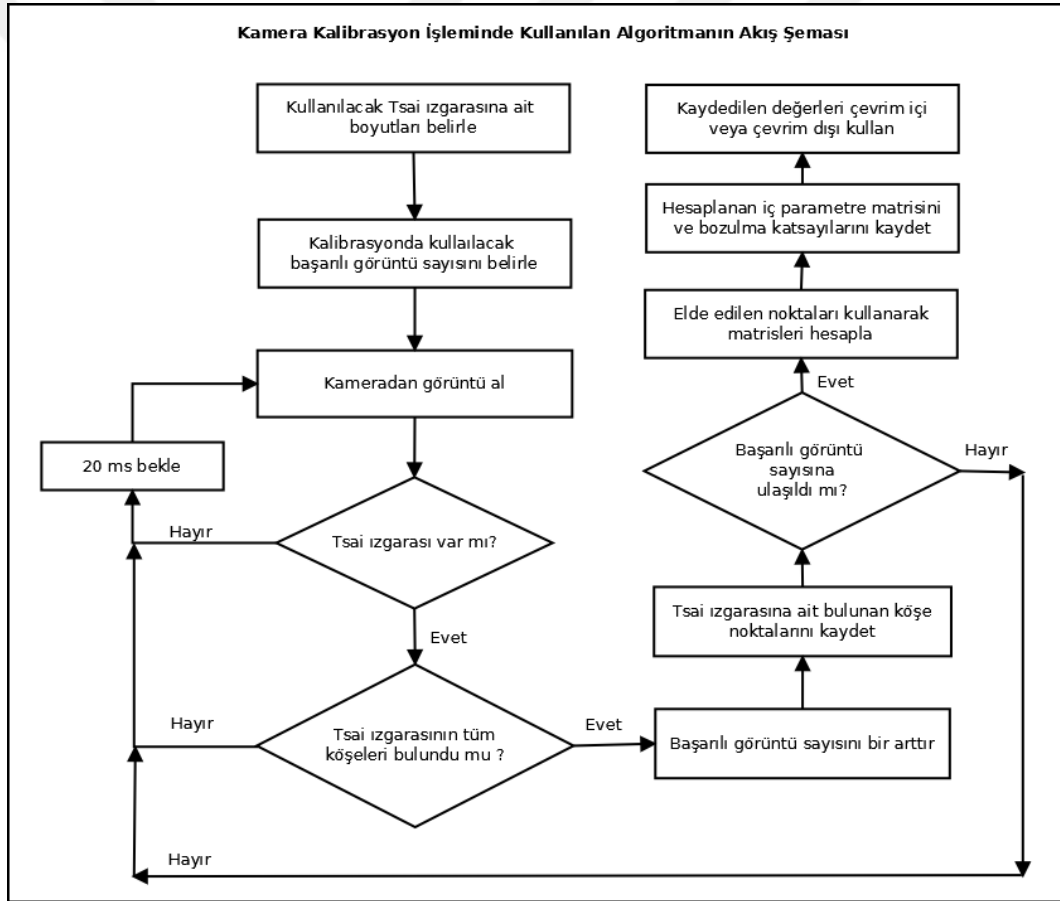
Yapılan deneysel çalışmalarda 25 adet başarılı görüntünün bulunması ve kaydedilmesi sonucunda kalibrasyon ve düzeltme işlemi gerçekleştirilmektedir. Kalibrasyon işlemi sırasından kullanılan görüntülerin az olması uygun sonuçlar vermezken, fazla olması ise kalibrasyon işlem zamanının artmasına sebep olmaktadır. Şekil 3.4'te kamera kalibrasyonu sırasında Tsai ızgarasına ait köşelerin tespit edilmesini gösteren örnek görüntüler yer almaktadır.



Şekil 3.4. Kalibrasyon işlemi sırasında Tsai ızgarası köşelerin bulunması

Şekil 3.5'te kamera kalibrasyonu yapılırken kullanılan akış şeması sunulmuştur. Algoritmanın ilk aşaması, parametrelerin belirlenmesidir. Bu aşamada, kullanılacak olan Tsai ızgarasına ait yatay ve dikey boyut, kalibrasyonun sona ermesi için yeterli örnek sayısı belirlenmektedir. Kamera kalibrasyon işleminin bir sonraki aşaması, kameradan alınan görüntünün işlenmesidir. Bu bölümde, alınan görüntü gri renk yapılmakta ve daha sonra morfolojik işlemler uygulanmaktadır. Alınan görüntüde Tsai ızgarasının olup olmadığı ve Tsai ızgarasına ait tüm köşelerin bulunup bulunmadığına bakılır. Saniyede 5 görüntü alınarak değerlendirme yapılması amacıyla alınan

görüntüler arasında 20ms bekleme yapılmaktadır. Eğer Tsai ızgarası yoksa veya alınan görüntüde tüm köşeler bulunamadıysa bir sonraki görüntü alınmaktadır. Eğer Tsai ızgarasına ait tüm köşeler başarılı bir şekilde bulunursa işaretlenmekte ve ekranda gösterilmektedir. Tüm köşelerin bulunması sonucunda, başarılı görüntü sayacı artırılmakta ve elde edilen görüntüye ait değerler kullanılmak üzere kaydedilmektedir. Kalibrasyon işleminin başında parametre olarak alınan görüntü sayısına ulaşıldığında, elde edilen tüm değerler değerlendirilerek, iz düşüm matrisi ve katsayı vektörü elde edilmektedir. Bu katsayılar ve matris o anda ve sonradan kullanılmak üzere dosyalara kaydedilmektedir. Elde edilen bu katsayılar kullanılarak görüntü düzeltme işlemi gerçekleştirilmektedir.



Şekil 3.5. Kalibrasyon işlemi sırasında önerilen algoritmaya ait akış şeması

3.1.3. Stereo görme, epipolar geometri ve kalibrasyon işlemi

İki kamera veya tek kamera kullanılarak farklı noktalardan çekilen ikiden fazla görüntü ile ortamın üç boyutlu haritasının elde edilmesi stereo görme olarak adlandırılmaktadır.

Stereo görme için verilecek en güzel model insan gözü olarak karşılaşılmaktadır [93]. Stereo görme işlemi insan gözünün yapısı örnek alınarak modellenmiştir. İnsanlar, iki gözü ile gerçek dünyadan aldıkları bilgileri kullanarak ortamda yer alan nesnelere algılamaktadırlar. Robot görmesi yapmak için insan görme işleminin bir benzerinin yapılması gerekmektedir. Bu işlem için stereo görme çalışmaları uygundur. Stereo görme kullanılarak ortamda bulunan hareketli ve sabit nesnelere hakkında bilgilere ulaşılabilmektedir.

Ayrıca, stereo görme ile derinlik bilgisi elde edilmektedir. Aynı nesnenin iki farklı kameradan elde edilen konumları arasındaki fark “stereo uzaklık (disparity)” olarak isimlendirilmektedir. İki görüntü arasında aynı noktalara eş gelen yerlerin tespit edilme işlemine stereo uzaklık eşleme (stereo matching – disparity matching) denilmektedir [94].

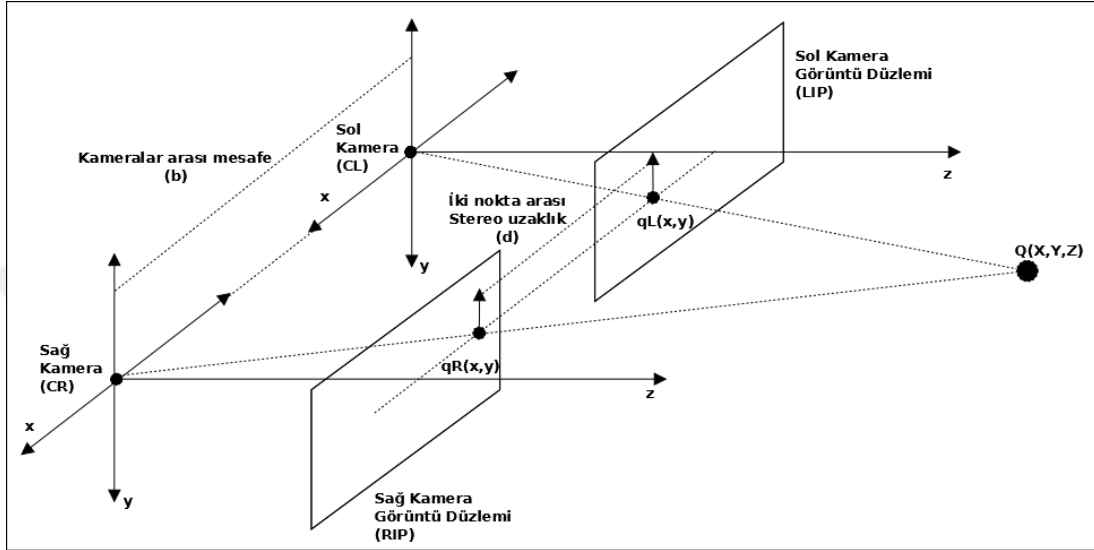
Stereo görme işlemi ile engel tespit edilmesi ve engeli aşma işlemi yapılması mümkündür. Literatürde robot görmesi ile yapılan optik akış, model tabanlı, görüntü tabanlı ve özellik çıkarımına dayalı yöntemlere göre stereo görme tabanlı çalışmaların avantajları, ortamın üç boyutlu tanımının yapılabilmesi, karmaşık engellerin tespit edilebilmesi, çevre koşullarındaki değişimden daha az etkilenmesi, ortamda yer alan engeller hakkında daha fazla bilgi edinilebilmesi yer almaktadır [94].

Stereo görme ile yapılan çalışmalarda, stereo kamera kalibrasyonu, kamera düzeltmesinin yapılması, stereo uzaklıklarının tespit edilmesi, stereo eşleştirme işleminin yapılmasında matematiksel iş yükünün fazla olması stereo görmenin dezavantajı olarak görülmektedir [94].

Çalışma ortamında yer alan nesnelere veya ortama ait görüntünün, iki farklı kamera veya tek kamera ile farklı noktalardan alınan görüntüler arasındaki geometrik ilişki “Epipolar Geometri” olarak adlandırılmaktadır [95, 96]. Şekil 3.6’da stereo görme modeli ve gerçek dünyada yer alan Q noktasının iki kamera görüntü düzlemlerine düşen izdüşümleri sunulmuştur.

Şekil 3.6’da $Q(X,Y,Z)$, gerçek dünyada yer alan Q noktasına ait koordinatları, $q_L(x,y)$, Q noktasının sol kamera görüntü düzlemi üzerine düşen izdüşümüne ait x ve y koordinatlarını, $q_R(x,y)$ ise Q noktasının sağ kamera görüntü düzlemi üzerine dü-

şen iz düşümüne ait x ve y koordinatlarını ifade etmektedir. CR, sağ kameraya ait merkez nokta, CL, sol kameraya ait merkez nokta, d , sol ve sağ görüntü düzlemleri üzerine düşen izdüşümler arasındaki Öklid uzaklığını, b ise sol ve sağ kameralar arasındaki mesafeyi ifade etmektedir. Nesneye ait herhangi bir noktanın gerçek uzaklığını bulmak için, üçgenlerin benzerlik kuralından yararlanılmaktadır.



Şekil 3.6. Stereo görme modeli

Şekil 3.6’da Q noktasının Z koordinatı, kameraların bulunduğu düzleme olan mesafeyi vermektedir. Yapılan tez çalışmasında ve stereo görme çalışmalarında, aynı özelliklere sahip iki kameranın kullanılması gerekmektedir. Dolayısıyla iki kameranın da odak noktaları (f) aynı olmalıdır. İki kameraya ait görüntü düzlemlerinin kamera merkezlerine olan uzaklıkları f kadardır. Eşitlik (3.20)’de stereo görme modeline ait denklemler ve birbiri ile olan ilişkileri sunulmuştur.

$$d = |qL_x - qR_x| \quad b = |CL - CR|$$

$$\frac{(|CL - CR| - |qL_x - qR_x|)}{(Z - f)} = \frac{|CL - CR|}{Z} \quad (3.20)$$

$$\frac{(b - d)}{(Z - f)} = \frac{b}{Z} \Rightarrow (Zb) - (Zd) = (Zb) - (fb) \Rightarrow Z = \frac{fb}{d}$$

Eşitlik (3.20)’de sunulan son denklemde f kameralara ait odak noktası olup bilinen bir değerdir. Denklemde sunulan b kameralar arası mesafe olup yine bilinen bir de-

ğerdir. Z gerçek mesafe değerin bulunması için, Q noktasının görüntü düzlemleri üzerine düşen iz düşümlerinin x koordinatları arasındaki farkın bulunması gerekmektedir.

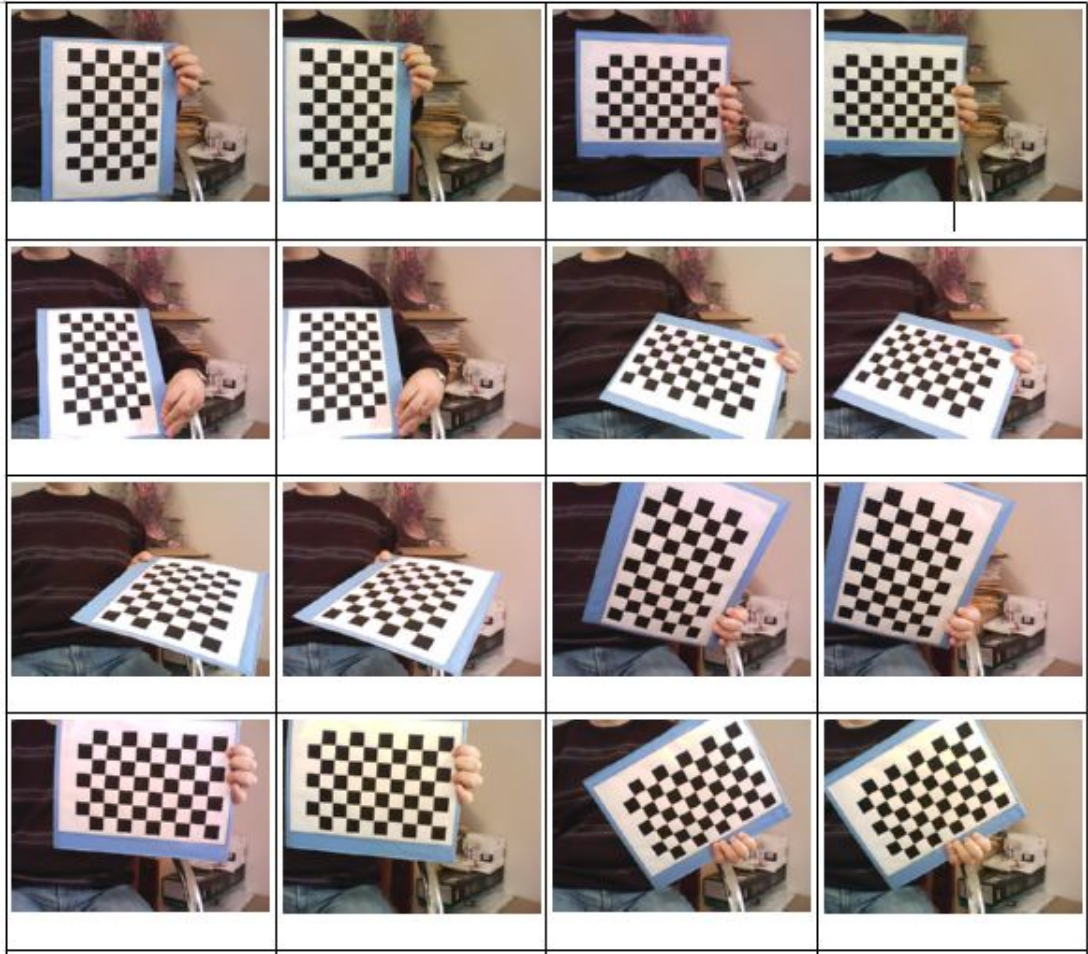
Sağ ve sol kamera ile alınan görüntülerin, görüntü düzlemi üzerindeki y koordinatları, kameralar aynı düzlem üzerinde paralel ve doğrusal olarak yerleştirildiği için aynı kabul edilmektedir. Bu piksellerin aynı y koordinatı üzerinde olması “epipolar doğru kısıtı” olarak bilinmektedir. Epipolar doğru kısıtının sağlanması, kalibrasyon işlemi sonucunda elde edilen parametreler yardımıyla olmaktadır.

Düzeltilmiş kamera görüntülerinde eşleşmesi istenen pikseller aynı y koordinatı üzerinde bulunmaktadır. Sağ ve sol kamera aynı paralel ekseninde olmasına karşın, stereo kamera kalibrasyonu yapılarak elde edilen katsayılar sayesinde düzeltilmektedir.

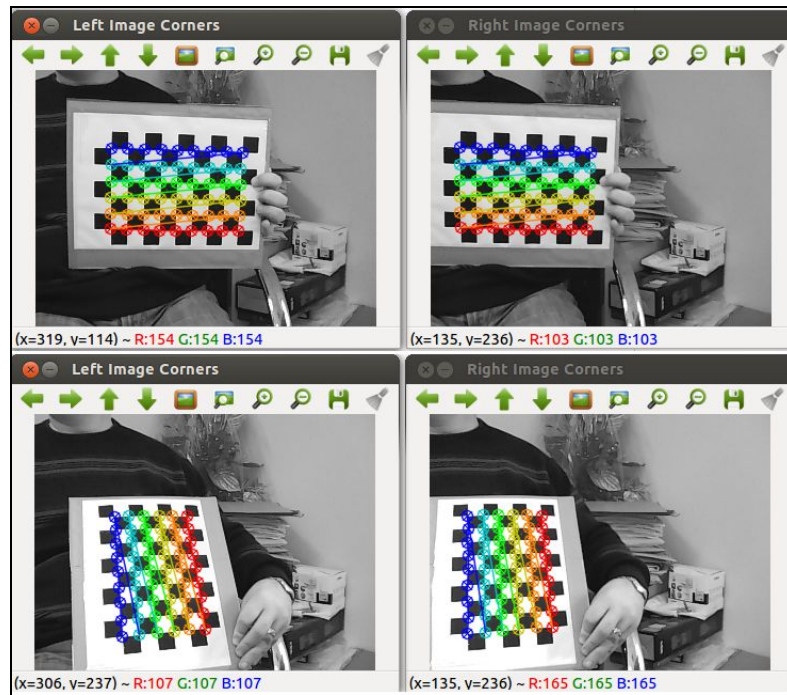
Kalibrasyon sayesinde 3 boyutlu gerçek dünyada yer alan bir noktanın, görüntü düzlemine düşen iz düşümü sayesinde koordinatları bulunmaktadır. Bir başka yöntem; kalibrasyon yapıldıktan sonra, görüntü düzlemi üzerindeki iki boyutlu bir noktadan iç ve dış parametreler kullanılarak 3 boyutlu dünyadaki konumunu çıkarma işlemi yapılabilmektedir.

Stereo kamera kalibrasyon işleminde uygulanan süreç, iğne delik kamere kalibrasyon sürecine kısmen benzemektedir. Stereo kamera kalibrasyon işleminde, iki kameradan eş zamanlı olarak alınan görüntüler işlenerek kamera modelleri, her bir kamera için iç ve dış parametre matrisleri, bozulma katsayıları ve kameraların bir biri ile olan ilişkileri elde edilmektedir. Stereo kamera kalibrasyon işleminde 9*6 boyutlarında bulunan Tsai ızgarasının farklı uzaklık ve açılardan 25 adet başarılı görüntüsü kullanılmaktadır.

Şekil 3.7’de stereo kamera kalibrasyonu sırasında kullanılan bazı görüntü çiftleri sunulmaktadır. Stereo kameradan alınan görüntü çiftleri aynı zamanda alınmaktadır. Görüntü çiftlerinin ikisinde Tsai ızgarasının tamamı yoksa başarısız bir görüntü olup, yeni bir görüntü alınmaktadır. Şekil 3.8’de ise kalibrasyon sırasında sol ve sağ kameradan alınan görüntü çiftlerinde yer alan Tsai ızgarasına ait köşelerin başarılı bir şekilde bulunmasını gösteren resim sunulmuştur.

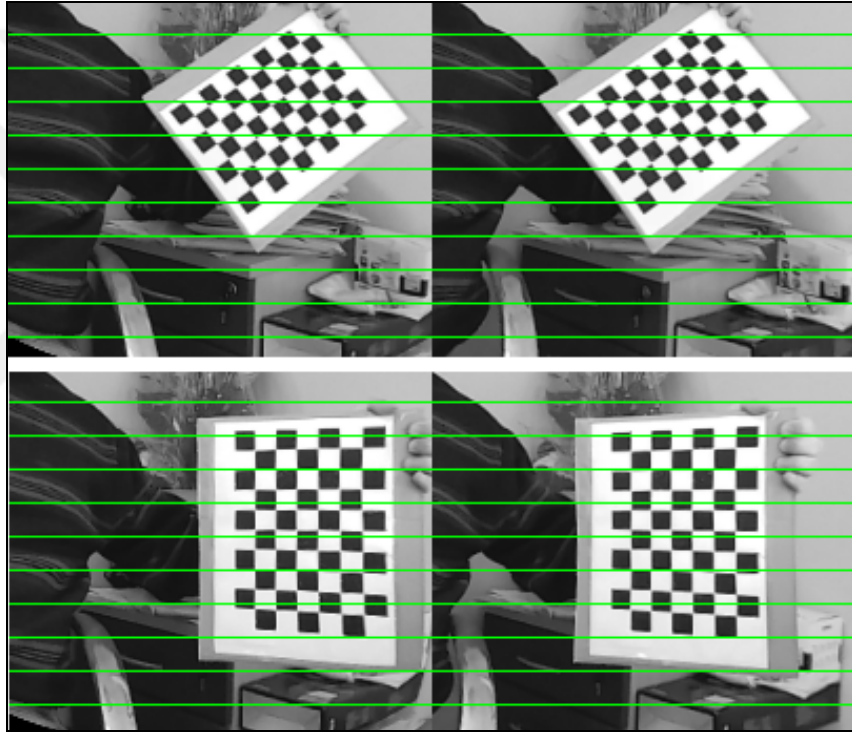


Şekil 3.7. Stereo kamera kalibrasyonu sırasında kullanılan görüntü çiftleri



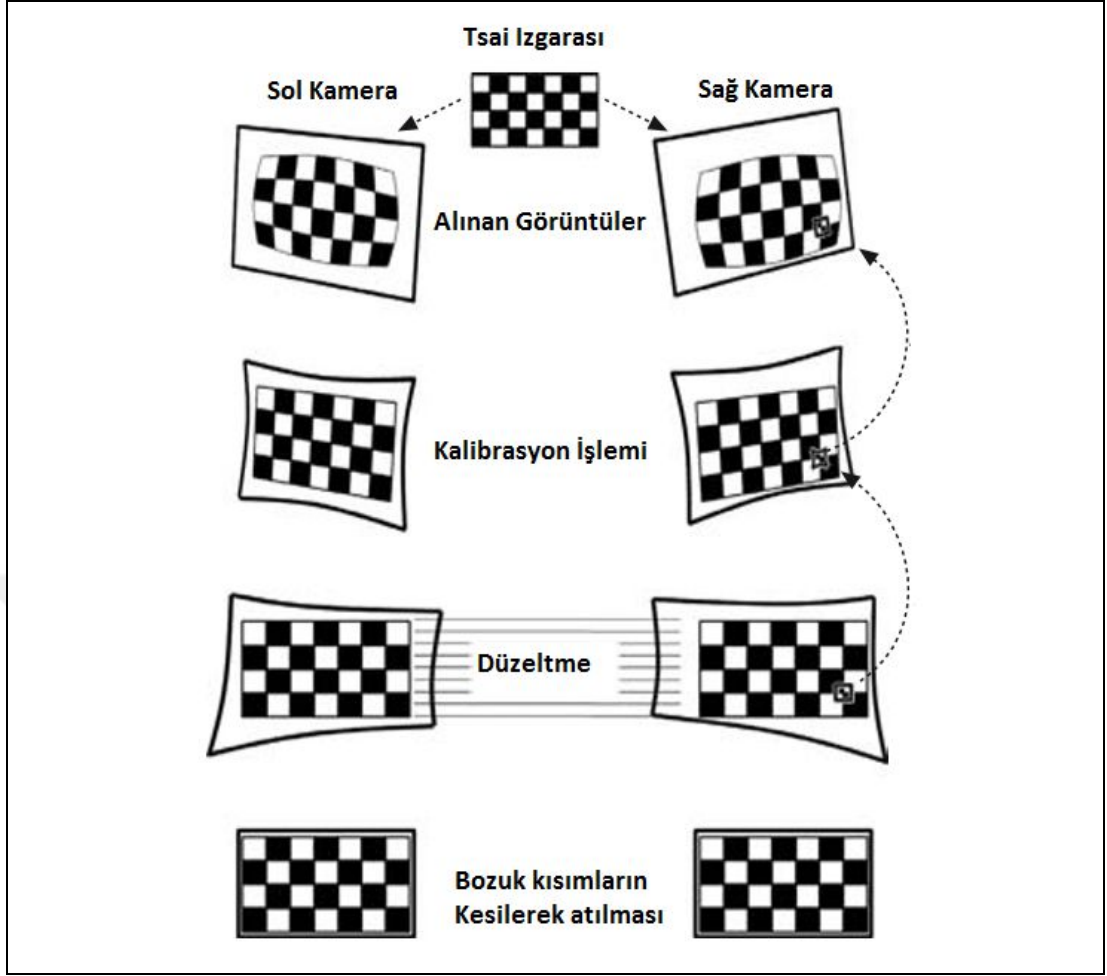
Şekil 3.8. Sol ve sağ kamerada köşelerin bulunması

Kamera kalibrasyonu sonunda elde edilen deęerlere gre matrisler tekrar dzenlenmiř ve Őekil 3.9'da sunulan grnt çiftleri elde edilmiřtir. Elde edilen grnt çiftleri zerinde dzeltme iřlemi gerekleřtirilmiř ve bozuk grlen kısımlar atılarak sadece ilgili kısımlar alınmıřtır. Őekil 3.9'da grlmekte olan yeřil çizgiler, sol ve saę kameradan alınan grntlerin aynı y koordinatını ifade etmektedir. Dzeltme iřleminden sonra iki grntnn y koordinatları aynı hizaya getirilmektedir. Bu durumda, iki grntye ait x koordinatları arasındaki farkı bulmak, gerek dnyadaki nesnenin konumu, boyutu ve kameralara olan uzaklıęı ile ilgili bilgileri bulmak iin yardımcı olmaktadır.



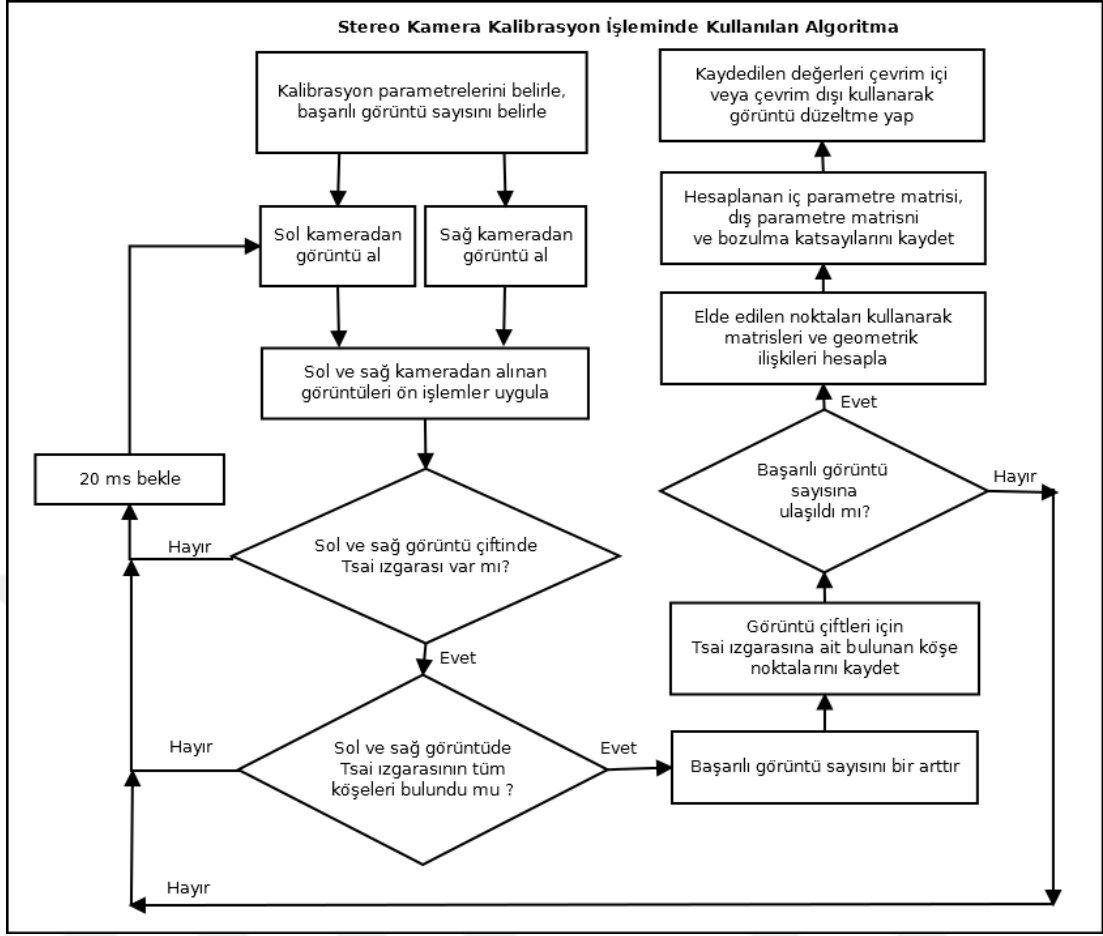
Őekil 3.9. Grnt dzeltme iřlemi sonucu oluřan yeni grnt

Stereo kamera kalibrasyonuna ve dzeltme iřlemine ait sre Őekil 3.10'da sunulmuřtur. Bu sre sırasında rnek bir Tsai ızgarası kullanılarak deęiřim gsterilmektedir. Srete ilk olarak, ortamda bulunan Tsai ızgarasına ait grntler eř zamanlı olarak sol ve saę kameradan alınmaktadır. Daha sonrasında sırasıyla, kalibrasyon iřlemi ve grnt dzeltme iřlemi gerekleřtirilmektedir. Son ařamada, dzeltilmiř olan grnt zerinde bozuk kısımlar kesilerek atılmakta ve sadece ilgilenilen kısım bırakılmaktadır.



Şekil 3.10. Stereo kamera kalibrasyonu ve düzeltme süreci [87]

Şekil 3.11’de stereo kamera kalibrasyonu işleminde kullanılan algoritma sunulmuştur. Kalibrasyon işleminde ilk olarak, parametreler belirlenmektedir. Kullanılacak olan Tsai ızgarasının boyutları, Tsai içinde yer alan karelerin boyutları, kalibrasyonun sora ermesi için başarılı görüntü sayısı belirlenmektedir. İki kameradan aynı anda alınan görüntüler üzerinde ilk olarak görüntü ön işleme yapılarak, Tsai ızgarasının tanınması ve tüm köşelerinin tespit edilmesi işlemine geçilmektedir. Alınan görüntüler, grileştirilmekte, Canny kenar tanıma algoritması uygulanmaktadır. İki görüntüde de tüm kenar ve köşeler başarılı bir şekilde bulunduysa, Tsai ızgarası üzerinde çizdirilmektedir. Ayrıca, başarılı görüntü sayacı bir artırılmakta, elde edilen değerler kaydedilmektedir. Kalibrasyon başında belirlenen değere ulaşıncaya kadar işlem tekrarlanmaktadır. Döngü bitince, elde edilen değerler kullanılarak her iki kamera için iç parametre, dış parametre, bozulma katsayıları hesaplanmaktadır. İki kamera arasındaki geometrik ilişki bulunarak elde edilen değerler sonucunda görüntü düzeltme işlemi gerçekleştirilmektedir.



Şekil 3.11. Stereo kamera kalibrasyonu akış şeması

3.1.4. Görüntü ön işleme adımında kullanılan filtreler

Çalışma ortamında bulunan nesnelerin kameralar ile bulunmasında, kamera kalibrasyon işleminden sonra elde edilen görüntülerin yumuşatılması ve gereksiz ayrıntıların atılması için filtreler kullanılmaktadır. Filtreleme işleminde, elde edilen resmin üzerinde bir filtre varmış gibi kabul edilmekte ve bu filtreye göre her piksel değeri yeniden hesaplanmaktadır. Literatürde, görüntü üzerinde yumuşatma, keskinleştirme, kenar tanıma işlemi için kullanılan çok sayıda filtre bulunmaktadır. Tez çalışması kapsamında, en iyi sonuçlar veren Gauss filtresi ve Konvolüsyon işlemi kullanılmıştır. Alt bölümlerde, bu işlemlere ait açıklamalar sunulmuştur.

3.1.4.1. Konvolüsyon işlemi

Konvolüsyon işlemi görüntü işleme uygulamalarında yaygın olarak kullanılmaktadır. Özellikle, görüntü yumuşatma, keskinleştirme, kenar belirleme gibi görüntü işleme için önem arz eden uygulamalarda kullanılmaktadır.

Konvolüsyon işleminde bir piksele ait çıkış değeri, kendisi ve komşu piksellere ait değerlerin ağırlıklı toplamı olarak hesaplanmaktadır. Kameradan alınan görüntü, giriş matrisini $f(x,y)$ ve hesaplan yeni görüntü için $f'(x,y)$ şeklinde ifade edilmektedir. Ağırlıklar matrisi veya Konvolüsyon matrisi adı verilen katsayılar $g(i,j)$ şeklinde ifade edilmektedir. Tez çalışmasında, görüntü keskinleştirme işlemi için konvolüsyon işleminde kullanılan ağırlıklar matrisi Eşitlik (3.21)'de sunulmuştur.

$$g(i, j) = \begin{vmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{vmatrix} \quad (3.21)$$

Eşitlik (3.22)'de görüntü matrisi üzerinde ağırlıklar matrisinin gezdirilmesini ifade eden denklem sunulmuştur. Denklemden sunulan \otimes operatörü konvolüsyon işlemi göstermektedir [97].

$$f'(x, y) = f(x, y) \otimes g(i, j) \quad (3.22)$$

Görüntü matrisi ile ağırlıklar matrisi işleme alınmış olup Eşitlik (3.23) ve (3.24)'te yer alan denklemler kullanılarak yeni değerler hesaplanmaktadır. M değeri, $g(i,j)$ ağırlıklar matrisine ait sütun değerini, N değeri $g(i,j)$ ağırlıklar matrisinin satır değerini ifade etmektedir.

$$f'(x, y) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} g(i, j) f(x - i, y - j) \quad (3.23)$$

$$f'(x, y) = \int \int_{i=0}^{M-1} \int_{j=0}^{N-1} g(i, j) f(x - i, y - j) d_i d_j \quad (3.24)$$

3.1.4.2. Gauss filtresi

Kamera ile ortamdan alınan görüntülerde oluşan gereksiz ayrıntı ve gürültüleri yok etmek amacıyla filtreler kullanılmaktadır. Tez çalışması kapsamında yapılan görüntü işleme uygulamalarında genellikle Gauss filtresi kullanılmaktadır.

Literatürde sunulan çalışmalarda, görüntü ön işleme adımında etkin ve yaygın bir şekilde Gauss filtresinin kullanıldığı gözlenmektedir. Kameradan alınan görüntülerde kenarların belirlenmesi ve yüksek frekanslı gürültülerin temizlenmesi sırasında, uzay

ve frekans düzleminin düşük tutulmasından dolayı Gauss filtresi yaygın olarak kullanılmaktadır.

Gauss filtresi düşük geçirgenlikli bir filtre olmasından dolayı görüntü yumuşatma işleminde de kullanılmaktadır. Gauss fonksiyonuna ait tek boyutlu denklem, Eşitlik (3.25)'de verilmiştir [97, 98]. Denklemden yer alan σ değeri, Gauss dağılımına ait standart sapmayı, x ise koordinat sistemini ifade etmektedir.

$$G(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}} \quad (3.25)$$

Eşitlik (3.26)'da iki boyutlu Gauss dağılımına ait denklem verilmiştir. Denklemden yer alan x ve y değerleri iki boyutlu koordinat sistemini ifade etmektedir [99].

$$G(x, y) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3.26)$$

Eşitlik (3.26)'da verilen iki boyutlu Gauss dağılımı, Eşitlik (3.25)'de sunulan x ve y koordinat sistemlerine göre tek boyutlu Gauss dağılımlarının çarpımı şeklinde yazılabilmektedir. Gauss dağılımının bu şekilde yazılması veya ayrıştırılması çok boyutlu sistemler içinde uygun olduğunu göstermektedir. Ayrıca, dağılımın bu şekilde ayrıştırılması çok boyutlu yapılarda hızlı sonuçlar alınmasını sağlamaktadır. Eşitlik (3.27)'de iki boyutlu Gauss dağılımının ayrıştırılmış hali sunulmuştur.

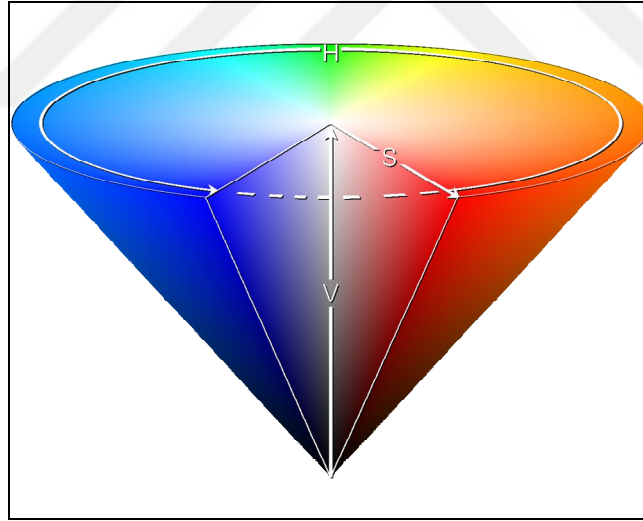
$$G(x, y) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}} \times \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{y^2}{2\sigma^2}} \Leftrightarrow G(x, y) = (G(x)G(y)) \quad (3.27)$$

3.1.5. HSV renk uzayı

Dalga boyu 400 ile 700 nm (nanometre) arasında olan ışık insan gözü tarafından görülebilmektedir. İnsan gözünün algılayabildiği kırmızı ötesi ve mor ötesi bu değer aralıkları arasında yer almaktadır. Genel olarak ışığı oluşturan renkler kırmızı, yeşil ve mavi olarak bilinmektedir. Işığı oluşturan bu üç rengin karıştırılması sonucunda diğer renkler oluşmaktadır. Kırmızı, yeşil ve mavi renklerinin, yüzde yüz oranında birleştirilmesi ile beyaz renk oluşmaktadır. Üç ana rengin ortamda bulunmaması durumunda siyah renk oluşmaktadır. Görüntü işleme için yaygın olarak kullanılan işlemlerden

birisi de renk uzayına ait dönüşüm işlemidir. Üç ana renge ait dalga boyu değer aralıkları sırasıyla, 450 – 490 nm arası mavi, 490 – 560 nm arası yeşil ve 635-700 nm arası kırmızı olarak tanımlanmaktadır.

Kameradan alınan renkli görüntüler, Kırmızı, Yeşil ve Mavi (Red, Green, Blue - RGB) renklerinin karışımından oluşmaktadır. Bu görüntüler RGB renk uzayına ait görüntülerdir. RGB renk uzayında alınan bu görüntüler, görüntü ön işleme adımından geçirildikten sonra, yapılacak çalışmaya göre grileştirme, siyah-beyaz görüntü yapma (ikili görüntü) veya renk uzayı dönüşüm işlemleri gerçekleştirilmektedir. Görüntü işlemede yaygın olarak kullanılan renk uzaylarından biri de HSV(Hue, Saturation, Value – Renk özü, Doygunluk ve Parlaklık) olarak bilinmektedir. HSV renk uzayı kullanılmasının sebebi ışık değişimlerinin kolaylıkla tanınabilmesidir. Tez çalışması kapsamında yapılan çalışmalarda RGB renk uzayında kameradan alınan bilgiler HSV renk uzayına dönüştürülerek işlenmektedir. HSV renk uzayına ait renk ölçeğini gösteren resim, Şekil 3.12’de sunulmuştur.



Şekil 3.12. HSV renk uzayı konik gösterimi [99]

Görüntüler piksellerden meydana gelmektedir. Piksellerin içerisinde ise renk bilgilerine ait değerler yer almaktadır. Gri renkte bulunan bir piksel 8 bit (1 Byte) yer kaplamakta ve bu pikselin renk aralık değeri 0-255 arasında olmaktadır. Kamera ile gerçek dünyadan alınan görüntüler genellikle renkli görüntülerden oluşmaktadır. Kamera ile RGB renk uzayında alınan renkli bir görüntüye ait piksel değeri üç byte’lık yer kaplamaktadır. Bu üç byte’lık veriye sahip olan pikselin ilk byte’ında kırmızı, ikinci byte’ında yeşil ve son byte’ında mavi renge ait değerler bulunmaktadır.

Her byte'in içerisinde 0-255 arasında o renge ait değer yer almaktadır. Eşitlik (3.28)'de $f(x,y)$, f görüntüsüne ait x ve y koordinatlarında yer alan piksel değerini ifade etmektedir. Piksel üç rengi ifade eden bir vektör şeklinde ifade edilmektedir.

$$f(x, y) = [R \quad G \quad B] \Rightarrow \begin{cases} R \rightarrow \{0 - 255\} \\ G \rightarrow \{0 - 255\} \\ B \rightarrow \{0 - 255\} \end{cases} \quad (3.28)$$

RGB renk uzayında alınan görüntülerin HSV renk uzayına çevrilerek kullanılması için ilk olarak, her bir piksel içerisinde yer alan kırmızı, yeşil ve mavi değerlerinin 0-1 arasında bir değere çevrilmesi gerekmektedir. Piksel içerisinde yer alan her bir rengin, toplam renklere oranı kullanılarak çevirme işlemi yapılmaktadır. Bu işlem için Eşitlik (3.29)'da sunulan denklemler kullanılmaktadır.

$$R_1 = \frac{R}{R + G + B} \quad G_1 = \frac{G}{R + G + B} \quad B_1 = \frac{B}{R + G + B} \quad (3.29)$$

Eşitlik (3.29)'da verilen denklemler kullanılarak elde edilen yeni renk değerlerinin toplamı 1'e eşit olmaktadır. Eşitlik (3.30)'da oluşan yeni değerlerin 1'e eşit olduğunu gösteren denklem görülmektedir.

$$R_1 + G_1 + B_1 = 1 \quad (3.30)$$

Eşitlik (3.29) kullanılarak elde edilen değerlerden en büyük (M) ve en küçük (m) olan değerler bulunmaktadır. Bulunan en büyük ve en küçük değer arasındaki fark (C) hesaplanmaktadır. Bulunan C değeri kullanılarak, HSV değerleri elde edilmektedir.

$$M = \max(R_1, G_1, B_1) \quad m = \min(R_1, G_1, B_1) \quad C = M - m \quad (3.31)$$

Elde edilen M , m ve C değerleri kullanılarak geçici H_1 değeri elde edilmektedir. Geçici H_1 değerinin hesaplanması, M değerinin kırmızı, yeşil ve mavi renklerden hangisinin olduğuna bağlı olarak değişmektedir. Ayrıca, M değeri V değerini vermektedir. S değerini elde ederken, en büyük ve en küçük renk değerleri arasındaki fark yani C değeri dikkate alınarak hesaplanmaktadır.

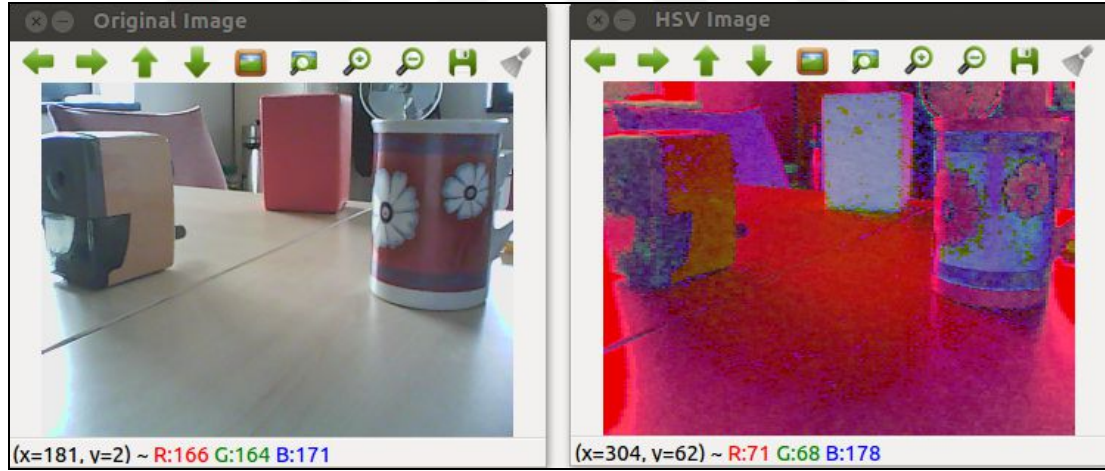
Eşitlik (3.32)'de geçici H_1 değerinin hesaplanmasını gösteren denklem verilmiştir.

$$H_1 = \begin{cases} \frac{G_1 - B_1}{C} \% 6 \rightarrow & M = R_1 \\ \frac{B_1 - R_1}{C} + 2, \rightarrow & M = G_1 \\ \frac{R_1 - G_1}{C} + 4, \rightarrow & M = B_1 \\ - \rightarrow & C = 0 \end{cases} \quad (3.32)$$

Eşitlik (3.33)'de H, S ve V değerlerinin son halini bulmak için kullanılan denklemler sunulmuştur.

$$H = 60^0 \times H_1 \quad V = M \quad S_{\text{HSV}} = \begin{cases} 0, & \rightarrow C = 0 \\ \frac{C}{V}, & \rightarrow C \neq 0 \end{cases} \quad (3.33)$$

Şekil 3.13'te Kameradan alınmış olan RGB renk uzayındaki resmin HSV renk uzayına çevrilmesi sonucunda elde edilen yeni görüntü görülmektedir.



Şekil 3.13. RGB-HSV renk uzayı dönüşüm görüntüleri

Eşitlik (3.34), (3.35), (3.36) ve (3.37) kullanılarak HSV renk uzayında bulunan görüntüler, RGB renk uzayına dönüştürülmektedir. Bu işlem için, RGB renk uzayından HSV renk uzayına dönüşüm işleminin tersi uygulanmaktadır. H, S ve V değerleri bilindiğinden C, M, m, X ve H₁ değerleri elde edilmektedir. Bu değerler kullanılarak, R, G ve B değerleri bulunmaktadır.

$$C = SV \quad M = V \quad m = M - C \quad H_1 = \frac{H}{60} \quad (3.34)$$

$$X = C(1 - |H_1 \% 2 - 1|) \quad (3.35)$$

$$(R_1, G_1, B_1) = \begin{cases} (0,0,0) & \rightarrow H \rightarrow - \\ (C, X, 0) & \rightarrow 0 \leq H_1 < 1 \\ (X, C, 0) & \rightarrow 1 \leq H_1 < 2 \\ (0, C, X) & \rightarrow 2 \leq H_1 < 3 \\ (0, X, C) & \rightarrow 3 \leq H_1 < 4 \\ (X, 0, C) & \rightarrow 4 \leq H_1 < 5 \\ (C, 0, X) & \rightarrow 5 \leq H_1 < 6 \end{cases} \quad (3.36)$$

$$(R, G, B) = [((R_1 + m)255), ((G_1 + m)255), ((B_1 + m)255)] \quad (3.37)$$

3.1.6. Morfolojik işlemler

Görüntü işlemede yaygın olarak kullanılan özelliklerden biri de morfolojik işlemlerdir. Literatürde yapılan bilgisayarlı görme ve görüntü işleme ile ilgili çok sayıda çalışmada kullanıldığı gözlenmektedir [90, 94, 100 - 103]. Morfolojik işlemler genellikle, bir görüntü içerisinde yer alan nesnelere belirginleştirmek için kullanılmaktadır. Görüntülerin temizlenmesi veya görüntü içerisinde meydana gelen kopuklukların giderilmesi amacıyla da kullanılmaktadır.

Tez çalışmasında, robotun çalışma ortamında yer alan nesnelere tespit edilmesi sırasında renk dönüşümleri ve eşikleme işlemlerinden sonra, nesne aralarındaki kopuk bağlantıların sağlanması ve ortamda bulunan gereksiz noktaların temizlenmesi amacıyla morfolojik işlemler kullanılmaktadır. Morfolojik işlemlerin kullanılması için, görüntünün gri veya ikili (siyah-beyaz) görüntüye dönüştürülmüş olması gerekmektedir. Genişletme, Aşındırma, Açma ve Kapama morfolojik işlemleri bulunmaktadır. Morfolojik işlem uygulamalarında, girdi olarak verilen görüntü üzerinde, yapısal element adı verilen ve genellikle kare şeklinde bir matris kullanılarak işlem yapılmaktadır.

Yapısal element (3x3), (5x5) olup küçük boyutlarda ikili bir görüntüden meydana gelmektedir. Morfolojik işlem uygulamalarında, nesnelere ait komşu piksel değerlerine bakılmakta ve yapılacak işleme göre ikili görüntü içerisindeki piksel değeri değiştirilmektedir. Bu işlem uygulandıktan sonra, çıkış görüntüsü içerisinde yer alan ve ilgilenilen nesne yapılan işleme göre genişlemekte veya daralmaktadır.

3.1.6.1. Genişleme işlemi

İkili sayısal görüntü içerisinde yer alan nesnelerin, yapısal element ile işleme tabi tutulup komşu piksel değerlerinde değişiklik yapılarak kalınlaştırma ve büyüme işlemi “genişleme” olarak tanımlanmaktadır. Genişleme işlemi literatürde yapılan çalışmaların bazılarında “Genleşme (Dilation)” olarak adlandırılmaktadır. Genişleme işlemi uygulandığında, görüntü içerisinde yer alan deliklerin doldurulması sağlanmaktadır. Giriş olarak verilen görüntüdeki her bir piksel değeri, yapısal element görüntüsünün orta noktası ile eşleştirmekte ve komşu piksellerdeki değişim çıkış görüntüsüne yansıtılmaktadır. Genişleme işlemi matematiksel olarak ifade edilmek istendiğinde, $f(x, y)$ giriş görüntüsü, $Y(i, j)$ yapısal element ve $f'(x, y)$, çıkış görüntüsünü ifade etmektedir. Yapısal elementin boyutları $(N \times N)$ şeklinde bir kare matris ve genellikle N tek sayı şeklindedir. Eşitlik (3.38) ve (3.39)'da ikili görüntüler için morfolojik genişleme işlemine ait matematiksel ifade sunulmuştur. \oplus operatörü morfolojik olarak genişlemeyi ifade etmek için kullanılmaktadır [103].

$$f'(x, y) = f(x, y) \oplus Y(i, j) \quad (3.38)$$

$$f'(x, y) = \bigcup_{i=0}^{N-1} \bigcup_{j=0}^{N-1} Y(i, j) \cap f(x - i, y - j) \quad (3.39)$$

Eşitlik (3.40)'da gri seviyeli görüntüler için morfolojik genişleme işlemine ait matematiksel ifade sunulmuştur[103].

$$f'(x, y) = f(x, y) \oplus Y(i, j) \Rightarrow \text{Max}_{i,j} [f(x - i, y - i) + Y(i, j)] \quad (3.40)$$

3.1.6.2. Aşındırma işlemi

İkili sayısal görüntü içerisinde yer alan nesnelerin, yapısal element ile işleme tabi tutularak komşu piksel değerlerinde değişiklik yapılarak daraltılması ve küçültülmesi işlemi “aşındırma (erosion)” olarak tanımlanmaktadır. Aşındırma işlemi, genişleme işleminin tersine görüntü içinde yer alan nesnenin sınırlarını küçültmek için kullanılmaktadır. Nesne içerisinde bulunan delikler büyümektedir. Bazı durumlarda, yapısal elementin özelliğine göre nesnelere veya nesnelerin parçaları birbirinden ayrıştırılabilir. Aşındırma işlemi için matematiksel denklemlerde, \ominus operatörü kullanılmaktadır.

İkili görüntüler için uygulanacak olan aşındırma morfolojik işlemine ait matematiksel ifadeler Eşitlik (3.41) ve (3.42)'de sunulmuştur [103].

$$f'(x, y) = f(x, y) \ominus Y(i, j) \quad (3.41)$$

$$f'(x, y) = \bigcap_{i=0}^{N-1} \bigcap_{j=0}^{N-1} Y(i, j) \cup f(x + i, y + j) \quad (3.42)$$

Eşitlik (3.43)'de gri seviyeli görüntüler için morfolojik aşındırma işlemine ait matematiksel ifade görülmektedir [103].

$$f'(x, y) = f(x, y) \ominus Y(i, j) \Rightarrow \text{Min}_{i,j} [f(x + i, y + i) + Y(i, j)] \quad (3.43)$$

3.1.6.3. Açma işlemi

Görüntü işleme uygulamalarında kullanılan morfolojik işlemlerden önce aşındırma işlemi, oluşan çıkış görüntüsü üzerinde genişletme işlemi uygulanmasına açma (opening) işlemi denilmektedir. Görüntü içerisinde yer alan nesnelere, yapısal elementin özelliğine göre temizlenmektedir [100]. Açma işlemi uygulandığında, birbirine yakın mesafede bulunan iki nesne birbirinden ayrılmış olmaktadır. Morfolojik açma işlemi Eşitlik (3.44)'de sunulduğu şekilde ifade edilmektedir. Morfolojik açma işlemi için \circ operatörü kullanılmaktadır.

$$f'(x, y) = f(x, y) \circ Y(i, j) \Rightarrow [f(x, y) \ominus Y(i, j)] \oplus Y(i, j) \quad (3.44)$$

3.1.6.4. Kapama işlemi

Girdi görüntüsü üzerinde morfolojik genişleme işleminden sonra oluşan çıkış görüntüsü kullanılarak aşındırma işleminin uygulanması kapama (closing) olarak adlandırılmaktadır. Birbirine yakın mesafelerde bulunan nesnelere birbiri ile birleştirilmiş olmaktadır. Birbirine yakın noktalar arasındaki boşluklar kapanırken, nesnenin ana hatları iyice belirginleşmektedir [100]. Morfolojik kapama işlemi Eşitlik (3.45)'de sunulduğu şekilde ifade edilmektedir. Morfolojik kapama işlemi için \bullet operatörü kullanılmaktadır.

$$f'(x, y) = f(x, y) \bullet Y(i, j) \Rightarrow [f(x, y) \oplus Y(i, j)] \ominus Y(i, j) \quad (3.45)$$

3.1.7. Görüntü eşikleme işlemi

Görüntü eşikleme (image thresholding) işlemi, görüntü bölümlenme işlemlerinde yaygın olarak kullanılmaktadır. Arka plan görüntüsü ile nesnenin görüntüsünü ayırmak için kullanılmaktadır. Genellikle eşikleme işlemi, grileştirilmiş bir görüntünün ikili görüntüye dönüştürülmesi sonucunda oluşan işlem olarak tanımlanmaktadır. Görüntü eşikleme işleminde belirli bir eşik değeri belirlenmektedir. Bu değer üzerinde bulunan piksel değerleri beyaz, altında bulunan değerler siyah olacak şekilde düzenlenmektedir. $f(x, y)$ giriş görüntüsünü, $f'(x, y)$ çıkış görüntüsünü ve E eşikleme değerini ifade etmektedir. Bu durumda yeni görüntüyü oluşturmak için kullanılan denklem Eşitlik (3.46)'da sunulmuştur.

$$f'(x, y) = \begin{cases} \min & \rightarrow f(x, y) \leq E \\ \max & \rightarrow f(x, y) > E \end{cases} \quad (3.46)$$

Eşitlik (3.46)'da sunulan min ve max değerleri, yeni görüntüde kullanılacak yeni değerlerdir. Yapılacak olan uygulamanın durumuna göre min ve max değerleri, 0, E , $f(x, y)$ ve 255, olarak belirlenebilmektedir. Aynı uygulamanın içerisinde min ve max değerleri birbirinden farklı değerler olmak zorundadır. Burada kullanılan E değerinin belirlenmesi önemli bir problemdir. Yapılan çalışmalarda, kümeleme tabanlı bir yöntem olan “otsu” metodu kullanılmaktadır. Grileştirilmiş bir görüntünün, ikili resme çevrilmesinde eşik değerini belirlemek için kullanılan etkin bir yöntemdir. Otsu yönteminde grileştirilmiş resim arka plan ve ön plan pikselleri olmak üzere iki sınıfa ayrılmaktadır. Ön plan ve arka plan sınıfları için varyans değerleri hesaplanmaktadır. İki sınıf için hesaplanan varyans değerlerinin en küçük olduğu değer anında sınıfların arasındaki varyans değeri en büyük değer olmaktadır. Eşitlik (3.47) kullanılarak, sınıflar arası ağırlıklı varyans hesaplanmaktadır [104,105]. Denklemden k_1 ve k_2 iki sınıfa ait olasılık dağılımı, t , eşik değerini, σ_1^2 ve σ_2^2 iki sınıfa ait varyans değerlerini ifade etmektedir.

$$\sigma_w^2(t) = k_1(t)\sigma_1^2(t) + k_2(t)\sigma_2^2(t) \quad (3.47)$$

Arka plan ve ön plan olarak sınıflandırılmış, sınıflara ait olasılık dağılımları Eşitlik (3.48) ve sınıflara ait ortalamalar Eşitlik (3.49) kullanılarak hesaplanmaktadır.

$$k_1(t) = \sum_{i=1}^t O(i) \quad k_2(t) = \sum_{i=t+1}^I O(i) \quad (3.48)$$

$$\mu_1(t) = \sum_{i=1}^t \frac{iO(i)}{k_1(t)} \quad \mu_2(t) = \sum_{i=t+1}^I \frac{iO(i)}{k_2(t)} \quad (3.49)$$

Eşitlik (3.50)'de $\sigma_1^2(t)$ ve $\sigma_2^2(t)$ varyans değerlerinin hesaplanması görülmektedir.

$$\sigma_1^2(t) = \sum_{i=1}^t [i - \mu_1(t)]^2 \frac{O(i)}{k_1(t)} \quad \sigma_2^2(t) = \sum_{i=t+1}^I [i - \mu_2(t)]^2 \frac{O(i)}{k_2(t)} \quad (3.50)$$

Verilen eşitlikler kullanılarak değerler hesaplanmakta ve $\sigma_w^2(t)$ varyansı için en uygun değeri sağlayan t değeri eşik değeri olarak bulunmaktadır.

Görüntü eşikleme işlemi bazı durumlarda renkli görüntüler üzerinde de uygulanmaktadır. RGB renk uzayında bulunan bir görüntü için uygulanabileceği gibi, RGB renk uzayından HSV renk uzayına çevrilmiş bir görüntü üzerinde de eşikleme işlemi uygulanmaktadır. Bu durumda HSV renk uzayında alınan görüntüde bulunan her pikselin H, S ve V değerlerinin belirli bir değer aralığında olup olmadığına bakılmaktadır. Aynı şekilde RGB renk uzayında da R, G ve B değerlerinin belirli aralıklarda olup olmadığına bakılmaktadır. RGB ve HSV renk uzaylarında ilgili değer aralıkları Eşitlik (3.51)'de sunulmuştur.

$$\text{HSV} = \begin{cases} H \in & \{0..2\pi\} \\ S \in & \{0..1\} \\ V \in & \{0..1\} \end{cases} \quad \text{RGB} = \begin{cases} R \in & \{0..255\} \\ G \in & \{0..255\} \\ B \in & \{0..255\} \end{cases} \quad (3.51)$$

HSV renk uzayında elde edilmiş bir görüntü üzerinde, belirli bir rengi yalnız bırakmak için, H, S ve V değerlerinin alabileceği en düşük ve en yüksek değerler verilerek piksel değerlerinin bu aralıkta olup olmadığı kontrol edilmektedir. Bu üç değer de aynı anda sağlanması durumunda ortamda bulunan renk değeri, siyah veya beyaz yapılar ikili görüntüye çevrilmektedir. Aynı uygulama RGB renk uzayından alınmış bir görüntüde yer alan pikselin R, G ve B değerleri için de uygulanabilir. Eşitlik (3.52)'de HSV renk uzayında bir pikselin değişimini gösteren ifade sunulmuştur.

$$f'(x, y) = \begin{cases} 255 & \rightarrow H(\min, \max) \cap S(\min, \max) \cap V(\min, \max) \\ 0 & \rightarrow \text{Uymuyorsa} \end{cases} \quad (3.52)$$

Eşitlik (3.53)'de RGB renk uzayında bir pikselin değişimini gösteren ifade verilmiştir.

$$f'(x, y) = \begin{cases} 255 & \rightarrow R(\min, \max) \cap G(\min, \max) \cap B(\min, \max) \\ 0 & \rightarrow \text{Uymuyorsa} \end{cases} \quad (3.53)$$

3.1.8. Nesne ağırlık merkezlerinin bulunması

Kameradan alınan görüntülerde ortamda bulunan nesnelere ait konum, boyut, yer ve ağırlık merkezi veya koordinatlarının bulunması için görüntü işlemede moment alma yöntemi kullanılmaktadır. Görüntü içerisindeki piksel yoğunluğunun ağırlıklı ortalaması moment olarak tanımlanabilir. Moment alma işlemi görüntü tanıma ve görüntü sınıflandırma işlemleri için de kullanılmaktadır. Plaka tanıma, karakter tanıma ve analizi, parmak izi analizlerinin yapılması, cisimlere ait alanların bulunması, desen analizlerinin çıkarılması gibi işlemlerde moment uygulamaları kullanılmaktadır. Görüntü işlemede moment kullanımına yönelik çok sayıda araştırma ve geliştirme çalışması yapılmaktadır [106 - 108].

Ortamda bulunan her bir nesne ile ilgili, konum, yer, boyut ve alan gibi özelliklerin çıkarılması, gezgin robot çalışmalarında robotun hareketi için önemlidir. Tez çalışmasında, elde edilen ikili görüntüler üzerinde her bir nesne için moment alma işlemi uygulanarak, o nesne ile ilgili ağırlık merkezleri bulunmaktadır. Bu sayede nesnelere arasındaki mesafeler ve nesnelere kameralara olan uzaklıkları da bulunabilmektedir.

Çalışmada, ortamda bulunan nesnelere eşikleme işlemi uygulandıktan sonra her bir nesneye ait alanlar hesaplanmaktadır. Ortamda bulunan bir nesnenin alanı, görüntü ikili resme çevrildiği için o nesneyi temsil eden değerlerin toplamı şeklinde ifade edilmektedir. Nesneye ait alanın hesaplanması için Eşitlik (3.54)'de verilen denklem kullanılmaktadır.

$$O_{\text{alan}} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} O(x, y) \quad (3.54)$$

Eşitlik (3.54)'te O_{alan} alanı bulunacak olan nesneyi ifade etmektedir. $O(x, y)$ nesnenin ilgili koordinatlardaki değerini ifade etmektedir.

Görüntü işleme uygulamalarında kullanılan moment denklemi Eşitlik (3.55) ve (3.56)'da sunulmuştur [106 - 108]. $m_k pq$ ifadesi k cisminin p ve q derecesinden momentini ifade etmektedir. Eşitlikte yer alan p ve q değerleri 0, 1, 2 gibi katsayıları, $O(x, y)$ momenti alınacak nesneye ait ikili görüntüyü ifade etmektedir. M ve N ise görüntünün satır ve sütun sayısını vermektedir.

$$m_k pq = \int \int_{x \ y} x^p y^q O(x, y) dx dy \quad (3.55)$$

$$m_k pq = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q O(x, y) \quad (3.56)$$

Eşitlik (3.55) ve (3.56)'da p ve q değerleri yerlerine 0 değeri verilerek, 0. dereceden moment hesaplanmakta ve bu denklem görüntüye ait alanı vermektedir. Eşitlik (3.57)'de 0.dereceden momenti hesaplayan denklem verilmiştir.

$$p = 0 \quad q = 0 \rightarrow m_k 00 = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} O(x, y) \quad (3.57)$$

Eşitlik (3.58) ve (3.59)'da p ve q değerlerine sırasıyla 1 değeri verilerek 1.dereceden moment hesaplaması yapılmaktadır.

$$p = 1 \quad q = 0 \rightarrow m_k 10 = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x O(x, y) \quad (3.58)$$

$$p = 0 \quad q = 1 \rightarrow m_k 01 = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} y O(x, y) \quad (3.59)$$

Ortamda bulunan k indis numaralı nesneye ait ağırlık merkezlerinin koordinatlarını bulmak için, 1.dereceden momentlerin, 0.dereceden momentlere oranını bulmak gerekmektedir. Eşitlik (3.60)'ta k indisli nesnenin x koordinatı, Eşitlik (3.61)'de ise y koordinatı hesaplanmaktadır.

$$\bar{x}_k = \frac{m_{10}}{m_{00}} \Rightarrow \bar{x}_k = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} xO(x, y)}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} O(x, y)} \quad (3.60)$$

$$\bar{y}_k = \frac{m_{01}}{m_{00}} \Rightarrow \bar{y}_k = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} yO(x, y)}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} O(x, y)} \quad (3.61)$$

Eşitlik (3.60) ve (3.61)'den yararlanılarak nesnelerin merkezi noktalarına ait koordinatlar elde edilmektedir. Bu değerler kullanılarak, merkezi momentler hesaplanmaktadır. Eşitlik (3.62)'de merkezi momente ait denklem verilmiştir.

$$\mu_{kpq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^p (y - \bar{y})^q O(x, y) \quad (3.62)$$

Momentlerin normalize edilmiş halini gösteren denklem Eşitlik (3.63) ve (3.64)'de görülmektedir[87].

$$\eta_{kpq} = \frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^p (y - \bar{y})^q O(x, y)}{\left(\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} O(x, y) \right)^{\left(\frac{p+q}{2} \right) + 1}} \quad (3.63)$$

$$m_{k00} = \mu_{k00} \text{ ve } \gamma = \left(\frac{p+q}{2} \right) + 1 \rightarrow \eta_{kpq} = \frac{\mu_{kpq}}{\mu_{k00}^\gamma} \quad (3.64)$$

Merkezi momentin doğrusal kombinasyonu Hu momentleri olarak adlandırılmaktadır. Hu momentindeki amaç, farklı normalizasyonu hesaplanmış olan merkezi momentleri birleştirmektir. Hu momentleri, görüntüye ait ölçek, yansıma ve döndürme gibi farklı yönlerini yansıtan bir işlev hesaplamaya çalışmaktadır. Hu momentine ait 7 adet değer hesaplanmakta olup bu değerlerin hesaplanmasını gösteren denklemler Eşitlik (3.65)'de verilmiştir. Denklemden yer alan H_{1k} ifadesi k nesnesine ait Hu momentinin birinci hesaplanan değerini ve η_k k indisli nesnesinin normalize edilmiş momentini ifade etmektedir [87].

$$H_{1k} = (\eta_k 20 + \eta_k 02)$$

$$H_{2k} = (\eta_k 20 - \eta_k 02)^2 + 4\eta_k 11$$

$$H_{3k} = (\eta_k 30 - 3\eta_k 12)^2 + (3\eta_k 21 - \eta_k 03)^2$$

$$H_{4k} = (\eta_k 30 + \eta_k 12)^2 + (\eta_k 21 + \eta_k 03)^2$$

$$H_{5k} = (\eta_k 30 - 3\eta_k 12)(\eta_k 30 + 3\eta_k 12) \left[(\eta_k 30 + \eta_k 12)^2 - 3(\eta_k 21 + \eta_k 03)^2 \right] \\ + (3\eta_k 21 - \eta_k 03)^2 (\eta_k 21 + \eta_k 03) \left[3(\eta_k 30 + \eta_k 12)^2 - (\eta_k 21 + \eta_k 03)^2 \right] \quad (3.65)$$

$$H_{6k} = (\eta_k 20 - \eta_k 02) \left[(\eta_k 30 + \eta_k 12)^2 - (\eta_k 21 + \eta_k 03)^2 \right] \\ + 4\eta_k 11 (\eta_k 30 + \eta_k 12) (\eta_k 21 + \eta_k 03)$$

$$H_{7k} = (3\eta_k 21 - \eta_k 03)(\eta_k 30 + \eta_k 12) \left[(\eta_k 30 + \eta_k 12)^2 - 3(\eta_k 21 - \eta_k 03)^2 \right] \\ - (\eta_k 30 + 3\eta_k 12)(\eta_k 21 + \eta_k 03) \left[3(\eta_k 30 + \eta_k 12)^2 - (\eta_k 21 + \eta_k 03)^2 \right]$$

3.2. Kullanılan Yol Planlama Algoritması

Tez çalışması kapsamında, çalışma ortamında bulunan robot, engel ve hedef bilgileri tespit edildikten sonra, gezgin robotun hedefe ilerlemesi için yol planlaması yapılmaktadır. Literatürde yapılan çalışmalarda, gezgin robotun yol planlama problemi için, çok sayıda yöntem sunulmaktadır. Yapılan tez çalışmasında gezgin robotun yol planlama probleminde genetik algoritma kullanılmıştır.

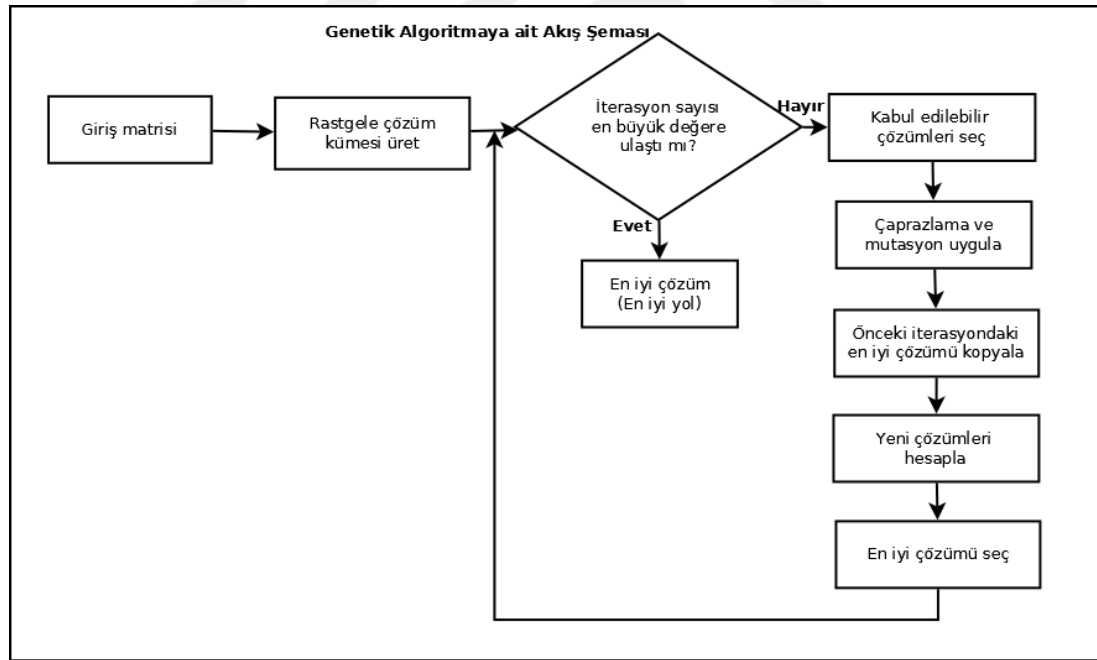
3.2.1. Genetik algoritma tabanlı yol planlama algoritması

Eniyileme problemleri ile uğraşan bilgisayarlılar genellikle doğada bulunan canlıları ve bu canlıların hareketlerini izleyerek sonuç elde etmeye çalışırlar. Literatürde yapılan çalışmalarda böcek, arı ve karınca gibi canlıların hareketleri incelenmiş ve bu hareketler bilgisayarda modellenerek çeşitli sistemlere uygulanmıştır. GA, evrimsel süreci yansıtmaya çalışan eniyileme yöntemlerinden biri olarak karşılaşılmaktadır. GA'ya ait temel çalışma ilkeleri 1975 yılında John Holland tarafından önerilmiş ve çalışmalar yapılmaya başlanmıştır [109]. GA, evrimsel süreçte yer alan çaprazlama, mutasyon ve seçim işlemlerini kullanır ve işlemleri matematiksel olarak gerçekleştirerek arama uzayında en iyi çözüme ulaşır.

GA, özellikle karmaşık problemlerde eniyileme çözümünü bulmak için kullanılmaktadır. Yapılan bazı çalışmalarda yapısının da uygun olmasından dolayı diğer algoritmalarla beraber kullanılarak hibrit çalışmalar da yapılmaktadır. GA'nın temel mantığı, tek bir çözüm yerine farklı çözümlerden oluşan bir çözüm kümesi üretmeye dayanmaktadır [110, 111].

GA çözüm kümesi içerisinde yer alan çözümler birbirinden bağımsız olduklarından, aynı anda birçok çözüm değerlendirmeye alınmaktadır. Bu sayede istenilen çözüme hızlı bir şekilde ulaşılma imkânı olmaktadır. GA en iyi çözüm kümesini yaratırken, en iyi çözümün hayatta kalması ilkesine bağlı olarak işlem yapmaktadır.

GA, çözüm kümesi içerisinde yer alan her bir çözüm, kromozom veya birey olarak adlandırılmaktadır. Bireyleri veya kromozomları oluştururken kullanılan parametreler ise gen olarak adlandırılmaktadır. Şekil 3.14'te GA'ya ait akış şeması sunulmuştur [111].



Şekil 3.14. GA, akış şeması

Tez çalışması kapsamında, A*, Takviyeli öğrenme algoritmaları da gerçekleştirilmiştir. GA tabanlı yol planlama algoritması hız ve alternatif çözüm kümelerinin üretilmesinden dolayı tercih edilmiştir. Yapılan deneysel çalışmalarda, çözüm BeagleBoard-xM üzerinde ortalama 1s altında sürelerde gerçekleştirilmektedir.

Şekil 3.14'te GA'ya ait sunulan akış şemasına ait açıklamalar maddeler halinde sunulmuştur [3, 112].

- Arama yapılacak, arama uzayı, GA'ya ait parametre değerleri, kullanılacak olan kodlama sistemi belirlenmekte ve arama uzayına ait küme GA'ya giriş olarak verilmektedir.
- Kodlanmış olan bireylerden rastgele seçim yapılmakta ve başlangıç çözüm kümesi, yani başlangıç nüfusu oluşturulmaktadır. Başlangıç nüfusu içerisinde yer alan her çözüm bireyi temsil etmektedir. Nüfus içerisinde bulunan birey sayısı çözüm kümesinin büyüklüğünü göstermektedir.
- Çözümü bulunmak istenen probleme ait amaç fonksiyonu ve uygunluk değerleri belirlenmektedir.
- Çözüm kümesi, yani başlangıç nüfusunda yer alan tüm bireylere ait amaç fonksiyonu kullanılarak uygunluk değerleri hesaplanmaktadır. Bu değere bakılarak bireyin çözüm kümesi için uygun olup olmadığına bakılmaktadır. Aynı zamanda en uygun olan yani en iyi çözümler de bu değere göre belirlenebilmektedir.
- Başlangıçta belirlenen algoritmayı sonlandırma özelliklerinin gerçekleşip gerçekleşmediği kontrol edilmektedir. Eğer istenilen sonuca ulaşıldıysa algoritma sonlandırılmakta ve çözüm kümesi içerisinde bulunan en iyi birey en iyi çözümü vermektedir. Probleme ait çözümler bir küme şeklinde verildiğinden diğer iyi çözümlere de ulaşılabilir. Eger istenilen sonuca ulaşılmadığında yeni bireylerin oluşturulması gerçekleşmektedir. Bireyler için seçim olasılıkları verilmektedir. Bu olasılık değerlerine göre bireyler rastgele gerçekleşecek bir seçime tabi tutulmaktadır. Çözüm kümesi içerisinde bulunan en iyi bireyler, bir sonraki çözüm kümesine devredilmektedir. Bu sayede iyi bireylerin neslini devam ettirmesi sağlanmaktadır. Kısacası, o zamana kadar bulunmuş olan en iyi çözümlerin kaybolmaması sağlanmaktadır.
- Problem çözümü için belirlenmiş olan parametreler kullanılarak, bireyler üzerinde çaprazlama ve mutasyon işlemi gerçekleştirilmektedir. Bu arada amaç çözüm kümesi için daha iyi sonuçlar elde edebilmektir. Çaprazlama nüfusun çeşitliliğini arttırmaktadır. Mutasyon kullanımı ise çözüm kümesinde bir tıkanıklık olmasında genel çözüme ulaşılması için kullanılan bir özelliktir.

- Yeni oluşturulan bireyler için, amaç fonksiyonu kullanılarak uygunluk değerleri tespit edilmektedir. Yeni nesil seçim işleminin ne kadar devam edeceği, problemin çözümüne başlarken belirlenmektedir. Belirlenen nesil (iterasyon) veya belirli bir uygunluk değerine gelindiğinde algoritma sona erdirilmektedir.

Yapılan tez çalışmasında, gezgin robotun dolaşacağı ortamın haritası GA için çalışma ortamını oluşturmaktadır. Genel olarak yapılan çalışmalarda, 8x10 boyutlarında bir matris kullanılmaktadır. Şekil 3.15'te ortamı gösteren matris sunulmuştur.

[0 0]	[0 1]	[0 2]	[0 3]	[0 4]	[0 5]	[0 6]	[0 7]	[0 8]	[0 9]
[1 0]	[1 1]	[1 2]	[1 3]	[1 4]	[1 5]	[1 6]	[1 7]	[1 8]	[1 9]
[2 0]	[2 1]	[2 2]	[2 3]	[2 4]	[2 5]	[2 6]	[2 7]	[2 8]	[2 9]
[3 0]	[3 1]	[3 2]	[3 3]	[3 4]	[3 5]	[3 6]	[3 7]	[3 8]	[3 9]
[4 0]	[4 1]	[4 2]	[4 3]	[4 4]	[4 5]	[4 6]	[4 7]	[4 8]	[4 9]
[5 0]	[5 1]	[5 2]	[5 3]	[5 4]	[5 5]	[5 6]	[5 7]	[5 8]	[5 9]
[6 0]	[6 1]	[6 2]	[6 3]	[6 4]	[6 5]	[6 6]	[6 7]	[6 8]	[6 9]
[7 0]	[7 1]	[7 2]	[7 3]	[7 4]	[7 5]	[7 6]	[7 7]	[7 8]	[7 9]

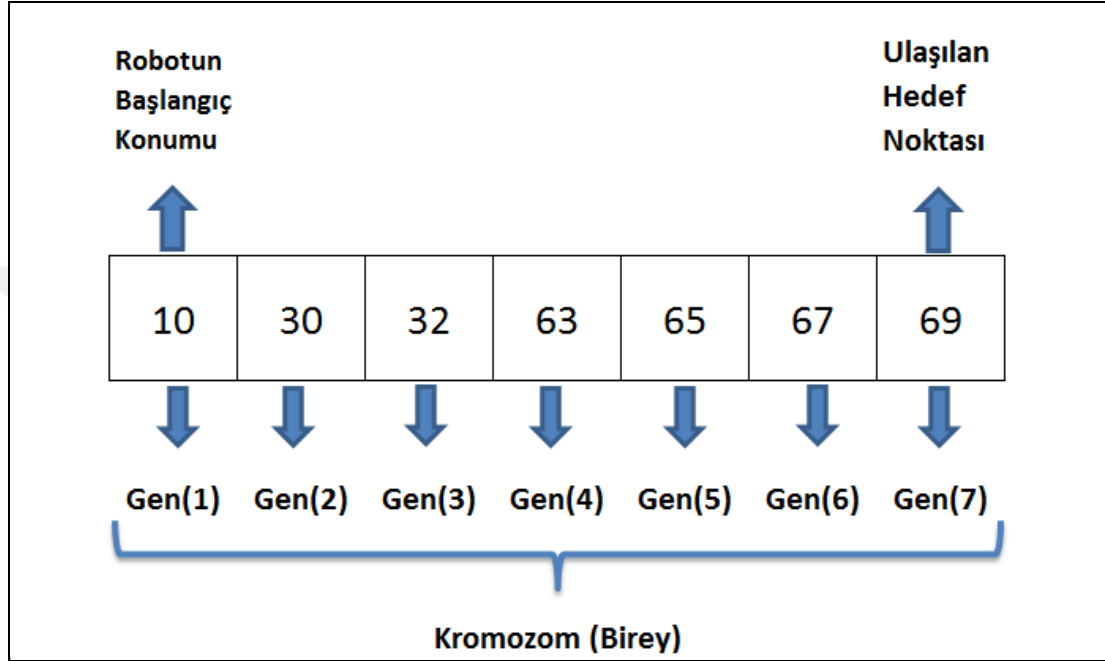
Şekil 3.15. GA için kullanılan çalışma ortamı

Şekil 3.15'te sunulan ortam matrisinde, kareler içerisinde yer alan ilk değer satır (x), ikinci değer ise sütun (y) değerini ifade etmektedir. Ortam matrisinde, beyaz alanlar boş yolları, yeşil alanlar ise engel olan kareleri temsil etmektedir.

Ortamda bulunan sarı renk başlangıç noktasını (gezgin robotu P3-DX) ve mavi renk ise hedef noktasını ifade etmektedir.

Yapılan çalışmada, seri tamsayı kodlama kullanılmış, başlangıç durumu ve ilerlenecek adımlar (x,y) koordinat ikilisiyle belirtilerek birey yani kromozomlar oluşturulmuştur. Kromozomlar başlangıç noktasından hedefe kadar gidilen tüm adımları ifade

etmektedir. Her adımda yer alan (x,y) ikilisi ise gen olarak tanımlanmaktadır. Şekil 3.16'da seri ve tamsayı olarak kodlanmış kromozom örneği bulunmaktadır. Kromozomun ilk geni robota ait başlangıç noktasını belirtirken, son gen ise ulaşılan hedef noktasını göstermektedir. Tez çalışmasında problemin durumuna göre kromozomun gen sayısı uyarlanabilir olarak ayarlanmıştır.



Şekil 3.16. Çalışmada kullanılan örnek bir kromozom

Başlangıç nüfusunda yer alan kromozomlar rastgele oluşturulmaktadır. Kromozomların rastgele oluşturulmasından kaynaklanan bazı problemler ortaya çıkmaktadır. Örneğin kromozomların içerisinde, engellerin olduğu genler de bulunmaktadır. Bundan dolayı, oluşturulan başlangıç nüfusunun içerisinde hatalı ve çözüme uygun olmayan kromozomlar yer almaktadır. Bunları elimine etmek amacıyla, kromozomlar oluşturulurken genlerin engellere denk gelip gelmedikleri kontrol edilmektedir. Engelle denk gelen genler için rastgele yeni bir gen oluşturulmaktadır.

Yapılan deneysel çalışmalar sonucunda başlangıç nüfusu 30 olarak kabul edilmiştir. Ancak, uygulanacak probleme ve problemin zorluğuna göre başlangıç nüfusu uyarlanabilir şekilde ayarlanmaktadır.

Çalışmada, robotun en kısa yoldan hedefe ulaşması için yeni bireylerin seçilmesi ve bu bireylerin uygunluklarının hesaplanması amacıyla Eşitlik (3.66) kullanılmıştır.

$$f_i = \sum_{k=0}^{M-1} \sqrt{(G_{k+1}(x) - G_k(x))^2 + (G_{k+1}(y) - G_k(y))^2} \quad (3.66)$$

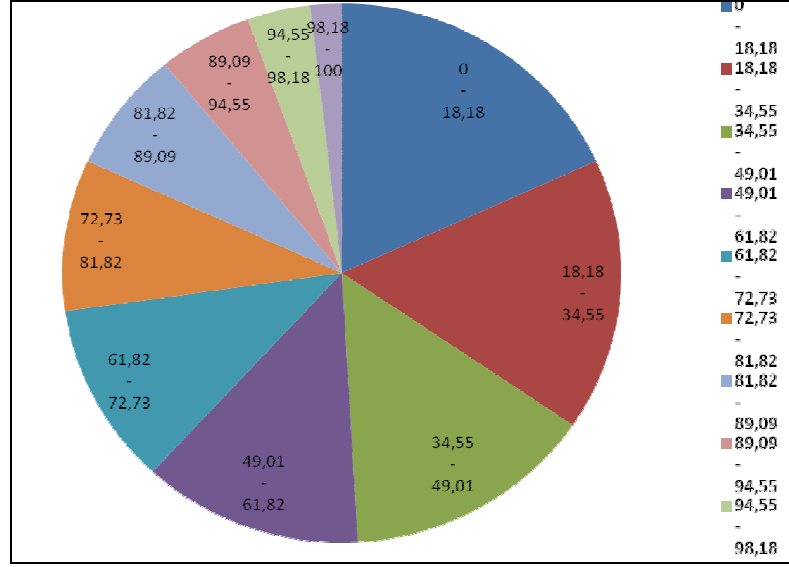
Formülde yer alan f_i amaç fonksiyonu kullanılarak elde edilen i.bireye ait maliyet değerini, M, toplam gen sayısını, $G_k(x)$ k. gene ait satır koordinatını, $G_k(y)$ k. gene ait sütun koordinatını ifade etmektedir. Her birey için f_i maliyet değeri hesaplanmakta ve f_i maliyet değeri minimum olan bireyler ilgili nesildeki en uygun birey olarak belirlenmektedir. Amaç fonksiyonu başlangıç noktasından, hedef noktasına kadar olan her iki nokta arasındaki Öklid uzaklıklarının toplamı şeklinde tanımlanmaktadır. İki gen arasında engel olma durumu, yolun kapalı olma durumu gibi istenmeyen durumlar göz önüne alındığında, amaç fonksiyonu Eşitlik (3.67)'de sunulduğu şekilde uygun olmayan bireyleri elemek amacıyla tekrar düzenlenmektedir.

$$f_i = \sum_{k=0}^{M-1} \sqrt{(G_{k+1}(x) - G_k(x))^2 + (G_{k+1}(y) - G_k(y))^2} + (CP \times ES) \quad (3.67)$$

Formülde sunulan CP ceza puan katsayısını ve ES ise başlangıç noktası ile hedef arasında karşılaşılan engel sayısını ifade etmektedir. Bu durumda, engelli yollardaki amaç fonksiyonu sonunda hesaplanan maliyet değeri yüksek çıkacağından bu değerler elenecektir.

Çalışmada, birey seçme işlemi rulet çemberi adı verilen yöntemle belirlenmektedir [3, 111]. Amaç fonksiyonu kullanılarak, elde edilen maliyet değerleri küçükten büyüğe doğru sıralanarak uygunluk değerleri bulunmaktadır. Uygunluk değeri, bireyin bir sonraki nesle aktarılma olasılığını belirlemek için kullanılır. Tablo 3.1 ve Şekil 3.17'de kullanılan rulet çemberine ait dağılım görülmektedir [3].

Uygunluk değerleri ve rulet çemberi oluşturulduktan sonra, 0-100 arasında rastgele değerler seçilmekte ve rulet çemberi üzerinde denk düşen parçalar alınarak bir sonraki nesle aktarılmaktadır. Rastgele 0-100 arasında 10 adet değer bilgisayara seçtirilmiştir. $R_s = \{12, 13, 18, 23, 36, 52, 59, 73, 88, 96\}$ sayıları seçilmiş olsun. Görüldüğü üzere yüzdesel değeri yüksek olan yerlerden daha fazla sayılar gelmiştir. Kısacası, maliyeti düşük olarak hesaplanan bireyler daha fazla seçilirken, yüksek maliyetli bireyler daha az seçilmektedirler.



Şekil 3.17. Rulet çemberi ve dağılımı

Tablo 3.1. Kromozom uygunluk tablosu

Birey no	1	2	3	4	5	6	7	8	9	10
Amaç fonksiyonu kullanılarak elde edilen değer	15	18	23	32	45	78	89	250	320	345
Uygunluk değeri	10	9	8	7	6	5	4	3	2	1
Seçilme olasılığı	0,1818	0,1636	0,145	0,127	0,109	0,09	0,073	0,055	0,036	0,018
Yüzde (%)	18,18	16,36	14,55	12,73	10,91	9,091	7,273	5,455	3,636	1,818
Değer aralıkları	0 - 18,18	18,18 - 34,55	34,55 - 49,01	49,01 - 61,82	61,82 - 72,73	72,73 - 81,82	81,82 - 89,09	89,09 - 94,55	94,55 - 98,18	98,18 - 100

Tez çalışmasında kullanılan GA üzerinde gerçekleştirilen bir başka işlem ise çaprazlama evrimsel olayıdır. İyi özellikli bireylerin genleri belirli noktalardan kesilerek çocuk bireyler oluşturulmaktadır. Çalışmada tek nokta çaprazlama işlemi rastgele olarak gerçekleştirilmiştir. Şekil 3.18'da örnek bir çaprazlama işlemi sunulmuş ve yeni çocuk birey gösterilmiştir. Ayrıca, o zamana kadar bulunmuş olan en iyi çözümün bir sonraki nesle aktarılması sağlanmaktadır. Bu işlem, en iyinin bir sonraki nesle aktarılması olarak adlandırılmaktadır. Çözüm kümesinin yerel minimuma takılmaması için, herhangi bir kromozom seçilerek rastgele bir gen değerinin değiştirilmesi mutasyon olarak bilinmektedir.

Çalışmada rastgele bir gen seçilerek değer değişimi yapılmakta ve çeşitlilik sağlanmaktadır.

Birey 1						Çocuk1					
32	35	15	39	69		32	35	45	65	69	
Birey 2						Çocuk2					
32	44	45	65	69		32	44	15	39	69	
		Çaprazlama Noktası									

Şekil 3.18. Çaprazlama İşlemi

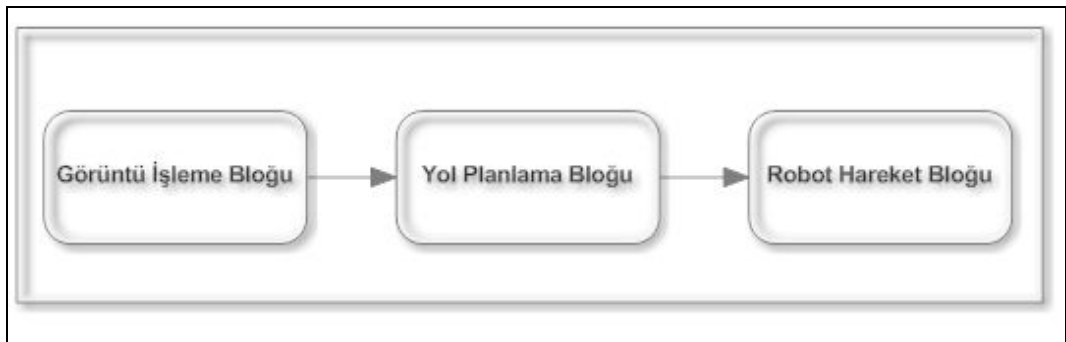


4. ÖNERİLEN DENEYSEL ÇALIŞMALAR VE YÖNTEMLER

Tez çalışması kapsamında, gezgin robotların dolaşımı için üç farklı deneysel çalışma yapılmıştır. Yapılan tüm çalışmalar, laboratuvar ortamında gerçekleştirilmiş ve P3-DX gezgin robotu üzerinde uygulanmıştır. Bu çalışmalardan birincisinde, kapalı ortamda bulunan gezgin robotlar için düşük maliyetli gezinti sistemi önerilmektedir. Önerilen gezinti sistemi, modüler yapıda olup farklı gezgin robotlar için uyarlanabilir şekilde hazırlanmıştır. Tez çalışması kapsamında yapılan ikinci çalışmada, gezgin robotun uzaktan kontrolü için, web tabanlı gezgin robot kontrol sistemi önerilmiştir. Tez çalışması kapsamında yapılan son çalışmada ise, gezgin robotun üzerine yerleştirilen TKB ve kameralar ile robotun hareket etmesi bulunduğu noktadan hedefe ilerlemesi sağlanmıştır. Alt bölümlerde yapılan ve önerilen çalışmaların detayları sunulmaktadır.

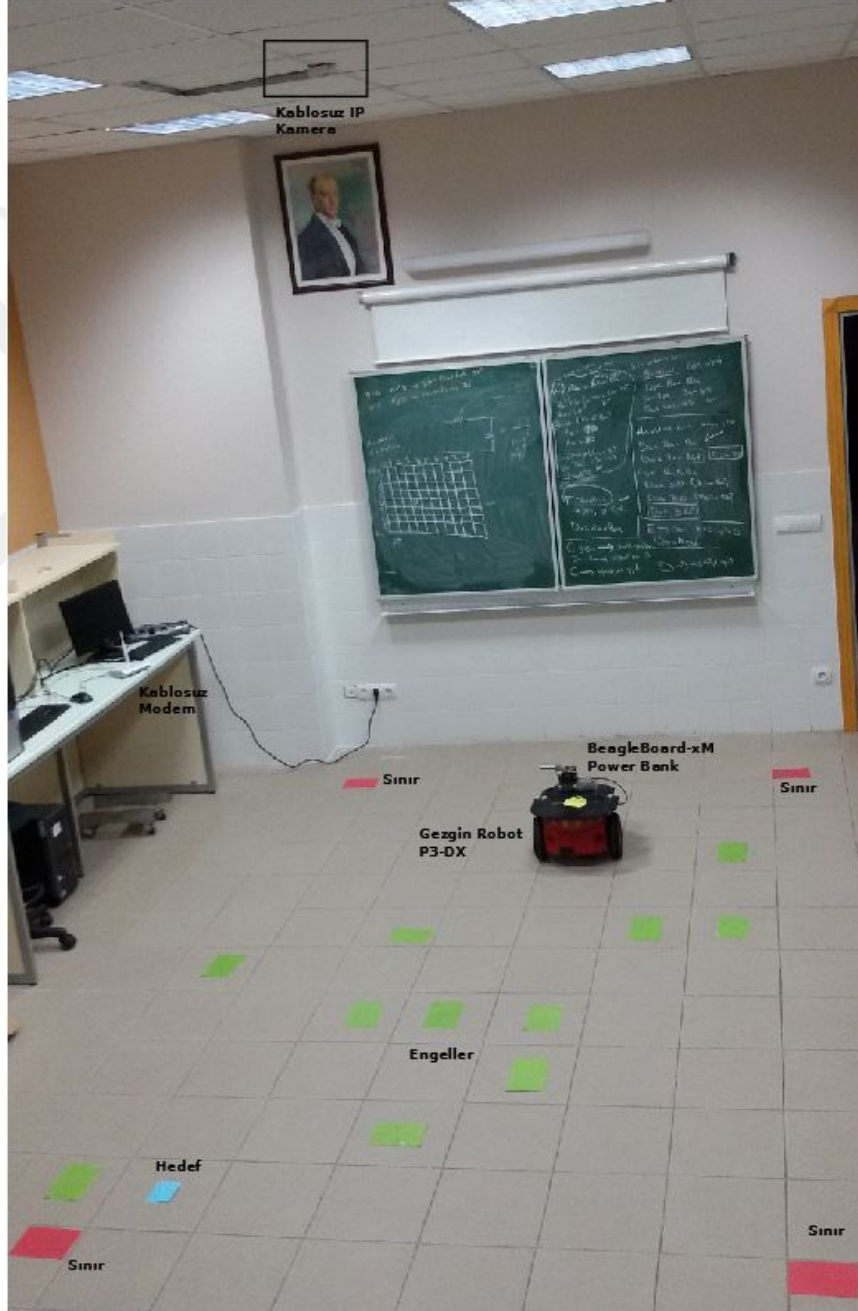
4.1. Kapalı Ortam Gezgin Robotlar için Gömülü Hareket Planlama Sistemi

Yapılan çalışmada, kapalı ortamda bulunan gezgin robotların başlangıç noktasından, hedefe giderken engellere çarpmadan ulaşması için düşük maliyetli, uyarlanabilir ve modüler yapıda bir sistem önerilmektedir. Önerilen sistem; Görüntü İşleme Bloğu (GİB), Yol Planlama Bloğu (YPB) ve Robot Hareket Bloğu (RHB) isimlerinde üç bloktan meydana gelmektedir. Sistemde bulunan bloklarda yapılacak küçük değişikliklerle farklı algoritmalar, farklı robot ve görüntü işleme teknikleri kullanılabilir. Şekil 4.1'de önerilen hareket planlama sistemine ait blokları gösteren şema görülmektedir.



Şekil 4.1. Gömülü hareket planlama sistemi

GİB, görüntü işleme tekniklerine göre kameradan almış olduğu görüntüyü işleyerek ortamın haritasını çıkarmaktadır. YPB, ortamın haritasın üzerinde yol planlama tekniği uygulayarak en kısa yolu üretmektedir. RHB, en kısa yolu kullanarak robotun her adımda dönüş açısını ve ilerleyeceği mesafeyi hesaplamakta ve seri arabirim üzerinden robota hareket komutu olarak iletmektedir. Şekil 4.2'de kapalı ortam gezgin robotlar için önerilen sistemin deney ortamı görülmektedir.



Şekil 4.2. Gömülü hareket planlama sistemi deney ortamı

Şekil 4.2'de görüldüğü gibi deney ortamının hazırlanması için, 1 adet kablosuz IP kamera, 1 adet kablosuz modem, 1 adet P3-DX gezgin robotu, 1 adet BeagleBoard-xM, 5 volt 1 amper Power Bank, ortamda bulunan engeller için yeşil renkli nesnelere, hedef için mavi renkli nesne, sınırlar için kırmızı renkli nesnelere kullanılmaktadır.

Önerilen çalışmada, bir bilgisayar yardımı ile TKB olarak kullanılan BeagleBoard-xM'e bağlanmakta ve sistem çalıştırılmaktadır. Sistem ilk olarak kablosuz modem üzerinden, kablosuz IP kameraya bağlanmakta ve ortamın görüntüsünü almaktadır. Alınan görüntü BeagleBoard-xM üzerinde görüntü işleme teknikleri ile işlenmekte ve ortamın haritası çıkarılmaktadır. Ortamın haritası üzerinde, GA tabanlı yol planlama algoritması kullanılarak gezgin robotun ilerleyeceği yol bulunmaktadır. Elde edilen yol bilgileri kullanılarak, P3-DX gezgin robotunun dönüş açısı ve ilerleyeceği mesafeler bulunmaktadır. Bulunan açı ve mesafe bilgileri her adımda P3-DX'e seri arabirim üzerinden iletilmekte ve robotun hedefe doğru hareketi sağlanmaktadır.

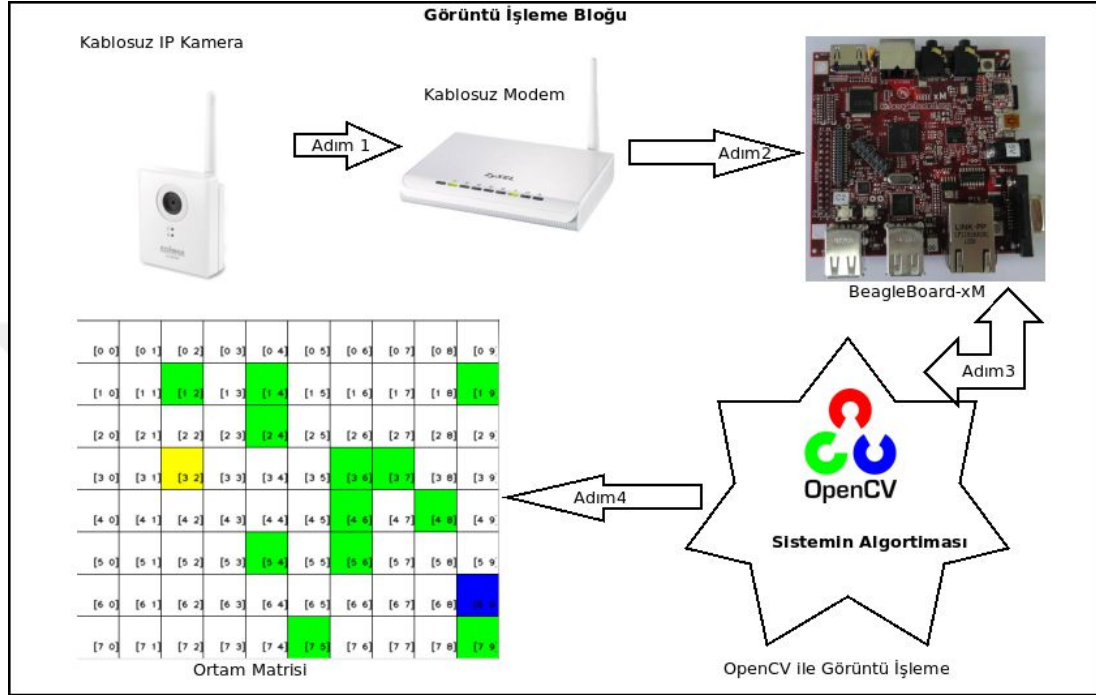
4.1.1. Görüntü işleme bloğu

Kapalı ortam gezgin robotlar için tasarlanan, gömülü hareket planlama sistemine ait ilk blok GİB'dur. GİB, BeagleBoard-xM üzerinde çalışmakta ve ortamın haritasını görüntü işleme tekniklerine göre çıkarmakta kullanılan bloktur. Yapılan deneysel çalışmalarda, ortamın haritasını çıkarmak için renk tanıma tabanlı bir sistem önerilmektedir. Ayrıca ortamın haritasını çıkarmak için, ortamda yer alan nesnelere kenar ve köşelerinin tanınması gibi farklı özelliklerine göre çıkarılabilir.

GİB, örnek çalışmada girdi olarak tavana yerleştirilen kablosuz IP kameradan görüntüyü almaktadır. Alınan görüntü sonucunda, ortam, ortamda bulunan engeller, hedef ve robotun konumlarının bulunduğu ızgara tabanlı haritayı çıktı olarak vermektedir. Izgara tabanlı harita üzerinde, ortamda yer alan nesnelere konumları satır ve sütunlar şeklinde sunulmaktadır.

Ortamda bulunan robot sarı, engeller yeşil, hedef mavi ve sınırlar kırmızı renktedir. Kırmızı renkli nesnelere ile robotun çalışma alanı tespit edilmektedir. Yeşil renkli nesnelere ile ilgili karelerde engel olduğu ve bu karelerden robotun geçemeyeceği belirlenir.

Robotun üzerine takılan sarı renk ile robotun çalışma ortamındaki koordinatları belirlenmektedir. Mavi renk ise robotun ulaşması gereken hedeftir. Yapılan bu deneysel çalışmada robotun hedef yönüne doğru baktığı varsayılmaktadır. Şekil 4.3’de GİB çalışma adımlarını gösteren blok şema sunulmuştur.



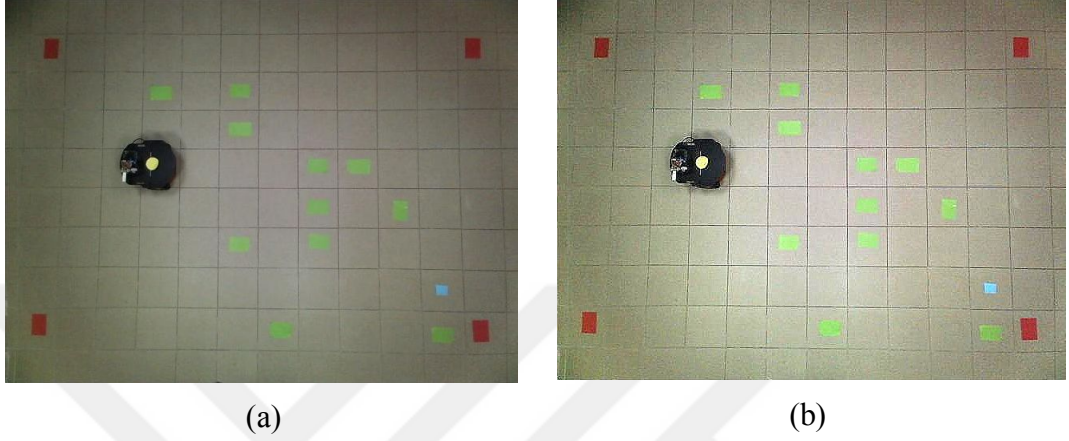
Şekil 4.3. GİB çalışmasını gösteren blok şeması

Sistem çalıştırıldıktan sonra, kablosuz modem üzerinden, kablosuz IP kameraya istekte bulunmaktadır. Kameradan alınan görüntü BeagleBoard-xM üzerinde OpenCV Kütüphanesi yardımıyla işlenmektedir. Ortamın ızgara tabanlı haritasının çıkarılmasında, en önemli bölüm sistemin algoritmasının bulunduğu görüntü işleme kısmıdır. Bu bölümde, farklı problemler için farklı algoritmalar geliştirilebilmektedir. Deneysel çalışmalarda renk tanıma tabanlı ortam haritasının çıkarılması için gerekli algoritma maddeler halinde sunulmuştur.

- Kameradan görüntünün alınması,
- Alınan görüntü üzerinde, görüntü iyileştirme işleminin uygulanması, (deneysel çalışmada konvolüsyon işlemi uygulanmaktadır.),
- RGB renk uzayından, HSV renk uzayına dönüşüm işlemi,
- Ortamda bulunan renkli nesnelere eşikleme işleminin uygulanması,
- Çalışma sınırlarının belirlenmesi,

- Ortam resminin analiz edilerek, nesnelerin ızgara tabanlı haritada konumlarının belirlenmesi işlemlerinden oluşmaktadır.

Şekil 4.4'te alınan kamera görüntüsü ve iyileştirme işleminden sonra elde edilen görüntü verilmiştir.



Şekil 4.4. GİB iyileştirme (a) Kamera görüntüsü (b) İyileştirilmiş görüntü

Şekil 4.4 (b)'de sunulan görüntü üzerinde ilk olarak HSV renk dönüşüm işlemi uygulanmıştır. Kırmızı renk ile belirtilen kısımlar için eşikleme işlemi uygulanmakta ve robot çalışma alanı dışında kalan kısımlar atılmaktadır. Bu işlemden sonra, çalışma ortamının görüntüsü (8x10) boyutlarında matrise bölünmektedir.

Görüntü üzerinde sarı renk için eşikleme işlemi uygulanmakta ve robotun bulunduğu konum matriste işaretlenmektedir. Sonrasında sırasıyla hedef ve engeller içinde aynı işlem yapılmaktadır. Bu bölümde kullanılan matrisin çözünürlüğü probleme bağlı olarak değiştirilebilmektedir.

Uygulanan işlemler sonucunda, GİB, çıktı olarak ortam haritasını matris olarak üretmektedir. Matris veya ortam haritası üzerinde, ortamda yer alan nesnelerin, yerleri işaretlenmiştir.

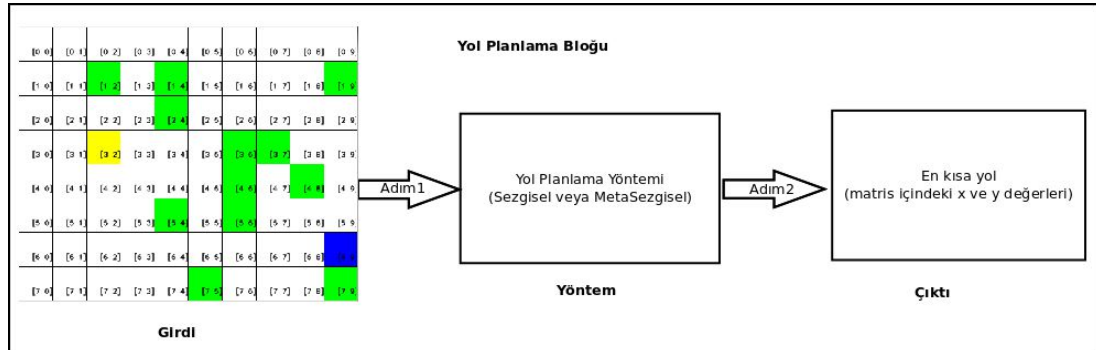
Şekil 4.5'te sunulduğu üzere, ortam matrisi üzerinde yer alan sarı renk robotun konumunu, yeşil renkler engellerin konumlarını, mavi renk ise hedefin konumunu göstermektedir. Robotun hareket edeceği alanlar beyaz olarak belirtilmektedir. Ayrıca, her bir satır ve sütun koordinatları harita üzerinde sunulmaktadır. İlk değer satırı, ikinci değer ise sütunu ifade etmektedir.

[0 0]	[0 1]	[0 2]	[0 3]	[0 4]	[0 5]	[0 6]	[0 7]	[0 8]	[0 9]
[1 0]	[1 1]	[1 2]	[1 3]	[1 4]	[1 5]	[1 6]	[1 7]	[1 8]	[1 9]
[2 0]	[2 1]	[2 2]	[2 3]	[2 4]	[2 5]	[2 6]	[2 7]	[2 8]	[2 9]
[3 0]	[3 1]	[3 2]	[3 3]	[3 4]	[3 5]	[3 6]	[3 7]	[3 8]	[3 9]
[4 0]	[4 1]	[4 2]	[4 3]	[4 4]	[4 5]	[4 6]	[4 7]	[4 8]	[4 9]
[5 0]	[5 1]	[5 2]	[5 3]	[5 4]	[5 5]	[5 6]	[5 7]	[5 8]	[5 9]
[6 0]	[6 1]	[6 2]	[6 3]	[6 4]	[6 5]	[6 6]	[6 7]	[6 8]	[6 9]
[7 0]	[7 1]	[7 2]	[7 3]	[7 4]	[7 5]	[7 6]	[7 7]	[7 8]	[7 9]

Şekil 4.5. GİB, çıktısı ızgara tabanlı ortam matrisi

4.1.2. Yol planlama bloğu

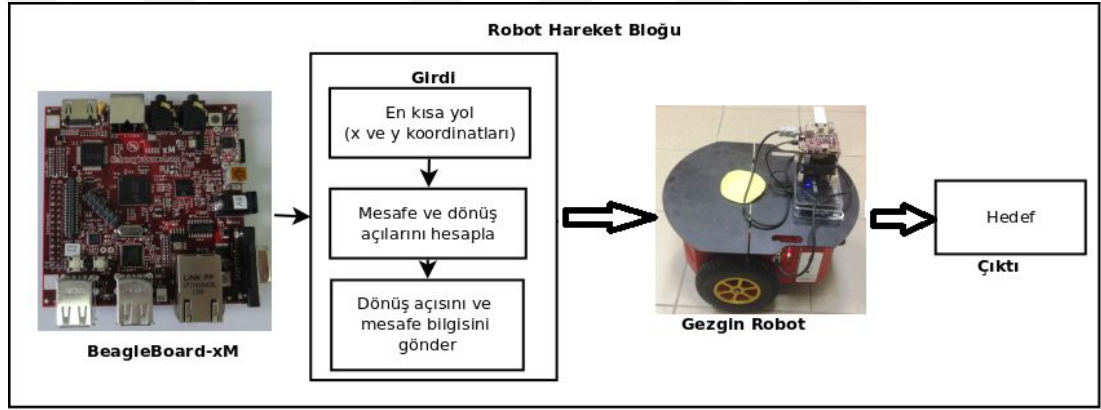
YPB, gezgin robotun takip edeceği yolu üreten bölümdür. YPB, GİB tarafından üretilen ızgara tabanlı ortam matrisini girdi olarak almaktadır. Çıktı olarak, gezgin robotun takip etmesi gereken en kısa yola ait koordinatlar üretilmektedir. Ortam matrisi üzerinde en kısa yol üretilirken sezgisel veya meta sezgisel yöntemlerden biri kullanılabilir. Literatürde yapılan çalışmalar incelendiğinde, GA, karınca koloni algoritması, A*, Dijkstra vb. algoritmaların kullanıldığı gözlenmektedir. Yapılan çalışmada, yol planlaması yapmak amacıyla GA kullanılmaktadır. GA, çalışmasına yönelik kullanılan yöntem Bölüm 3'te anlatıldığı şekilde uygulanmaktadır. Şekil 4.6'da YPB'nin çalışmasını gösteren blok şema sunulmuştur.



Şekil 4.6. YPB'nun çalışmasını gösteren blok şema

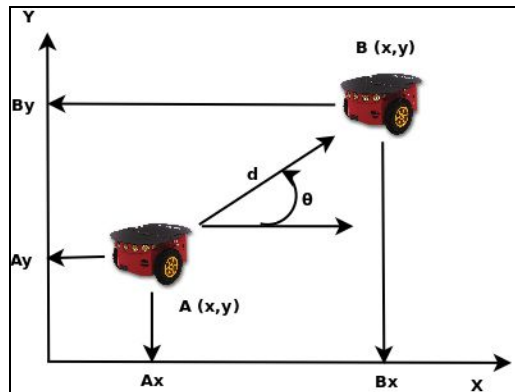
4.1.3. Robot hareket bloğu

RHB, gezgin robotun harekete geçip hedefe ilerlemesini sağlayan bloktur. YPB'nin üretmiş olduğu en kısa yola ait koordinat bilgilerini girdi olarak almaktadır. Almış olduğu bu girdi bilgisini kullanarak gezgin robotun hedefe doğru ilerlemesi için gerekli, dönüş açıları ve mesafeleri hesaplamaktadır. Hesaplanan bilgiler, robotun anlayacağı komutlar haline getirilmekte ve seri arabirim üzerinden gezgin robota iletilmektedir. Deney ortamında P3-DX gezgin robotunun olmasından dolayı, komutlar bu robotun anlayacağı paketlere dönüştürülerek yollanmaktadır. RHB'de hesaplanan değerler farklı robotlar için farklı iletişim arabirimleri üzerinden uygulanabilmektedir. Şekil 4.7'de RHB'nin çalışmasını gösteren blok şema görülmektedir.



Şekil 4.7. RHB'nin çalışmasını gösteren blok şema

RHB'de dönüş açısı ve mesafe bilgileri hesaplandıktan sonra, her adımda P3-DX'e gönderilmekte ve robotun bir sonraki noktaya hareket etmesi sağlanmaktadır. Şekil 4.8'de gezgin robotun hareketi görülmektedir.



Şekil 4.8. A - B arası hareket

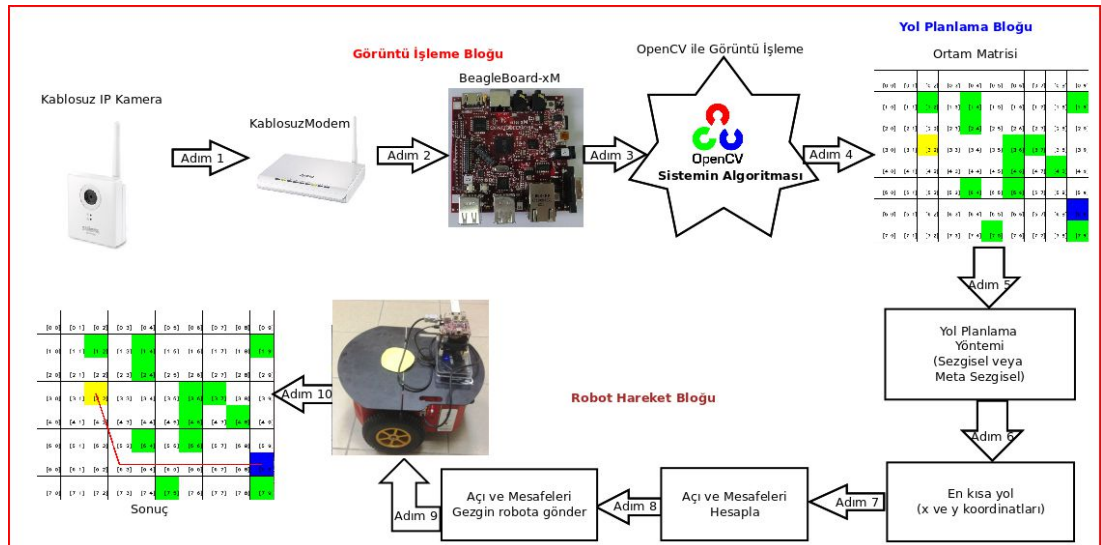
Şekil 4.8’de sunulduğu üzere, gezgin robotun bulunduğu ilk konum A noktası, ilerleyeceği nokta B noktasıdır. Gezgin robotun bulunduğu noktanın koordinatları $A(x,y)$ ilerleyeceği noktanın koordinatları ise $B(x,y)$ olarak ifade edilmektedir. Gezgin robotun A noktasından B noktasına ilerlerken yaptığı dönüş açısı θ , ilerleyeceği Öklid uzaklığı d ile ifade edilmektedir. Eşitlik (4.1)’de dönüş açısının, Eşitlik (4.2)’de ise Öklid mesafesinin hesaplanması verilmiştir.

$$\theta = \arctan\left(\frac{B_y - A_y}{B_x - A_x}\right) \quad (4.1)$$

$$d = \sqrt{(B_x - A_x)^2 + (B_y - A_y)^2} \quad (4.2)$$

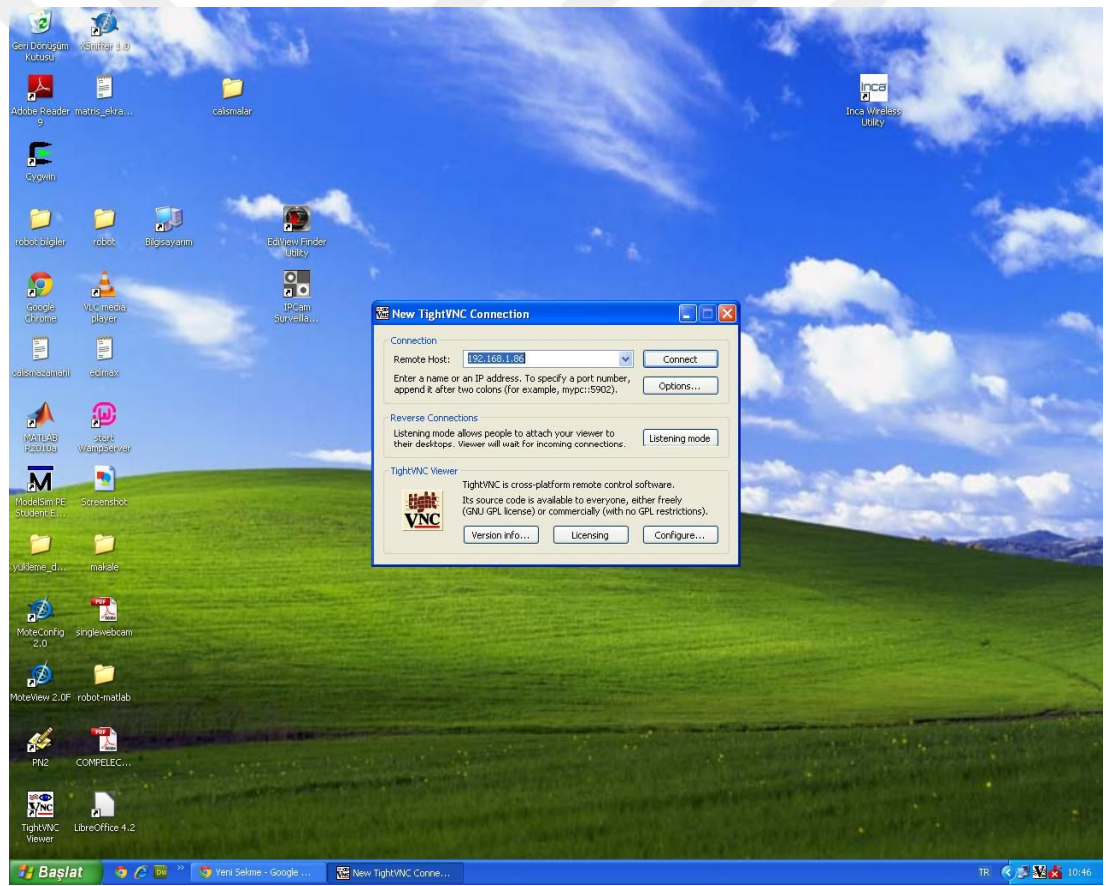
4.1.4. Sistemin uygulanması ve sonuçları

Önerilen sistem, gezgin robotun deney ortamında kablosuz bir şekilde hareket etmesi için planlanmış, geliştirilmiş ve gerçekleştirilmiştir. Kapalı ortam gezgin robotlar için önerilen gömülü hareket planlama sistemine ait genel blok şema Şekil 4.9’da sunulmuştur. Şekil 4.9’da GİB, YPB ve RHB bloklarının birbiri ile iletişimi genel olarak anlatılmakta, ilk adım olan görüntünün kablosuz IP kameradan alınmasından, son adım olan robotun hedefe ilerlemesine kadar olan işlem adımları gösterilmektedir. Sistemin çalıştırılması ve BeagleBoard-xM üzerinden görüntülerin alınması için deney ortamında bulunan bir bilgisayar kullanılarak uzak masaüstü bağlantısı sağlanmıştır.



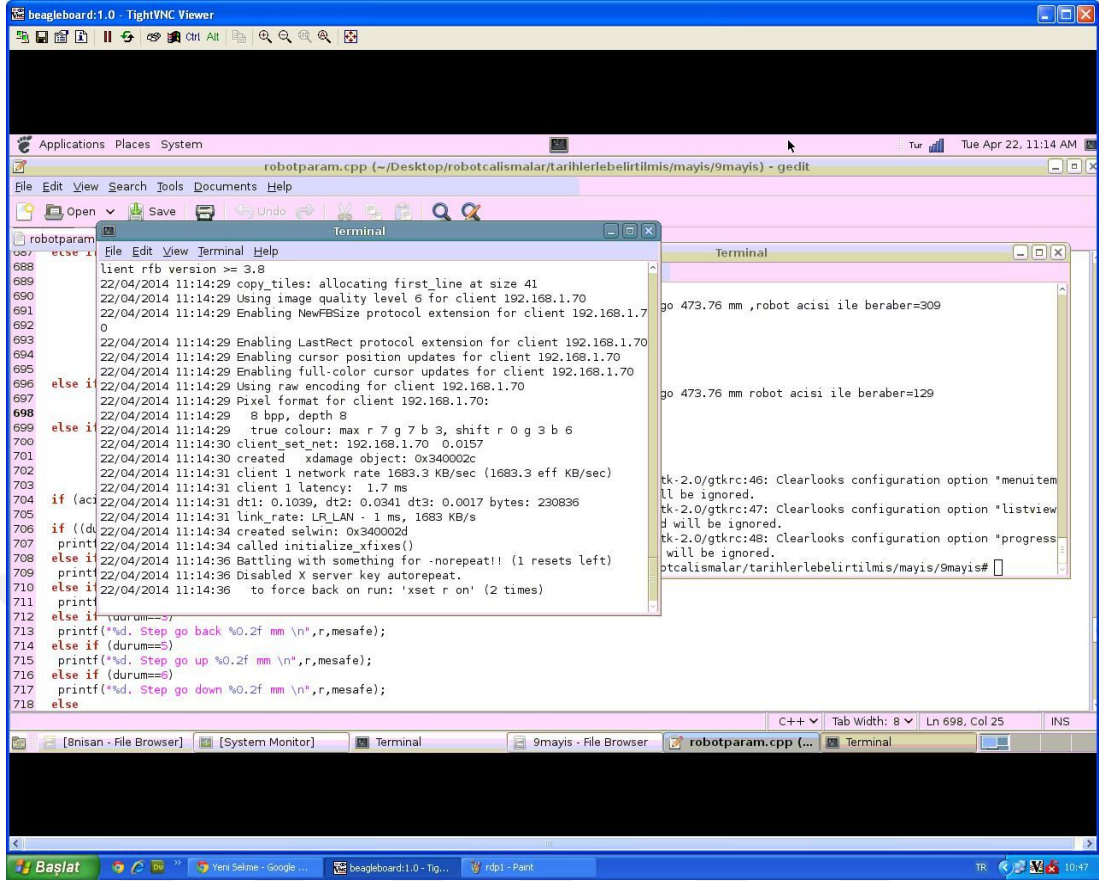
Şekil 4.9. Gezgin robotun gömülü hareket planlama sistemi

Önerilen çalışmanın deneysel ortamda gerçekleştirilmesi için ilk olarak, kablosuz modeme, kablosuz IP kamera, BeagleBoard-xM ve ekran görüntülerinin kaydedileceği bilgisayara IP adresleri tanımlanmaktadır. Tüm cihazlar aynı modem üzerinden haberleşecek şekilde yapılandırılmaktadır. IP adreslerinin ayarlanmasından sonra, bilgisayar ve BeagleBoard-xM üzerine grafik masaüstü paylaşımı sağlayan VNC (Virtual Network Computing) programları kurulmaktadır. BeagleBoard-xM üzerinde bulunan VNC yazılımı çalıştırılmakta ve bilgisayar üzerinde bulunan VNC yazılımından BeagleBoard-xM IP adresi yazılarak istekte bulunulmaktadır. Şekil 4.10'da laboratuvar ortamında VNC kurulu bir bilgisayardan BeagleBoard-xM'e istek yapılan görüntü sunulmuştur.



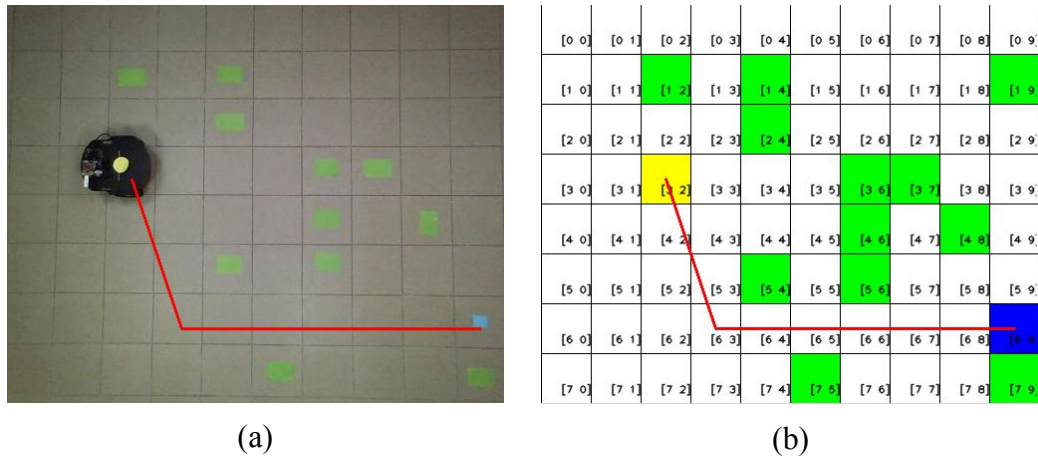
Şekil 4.10. Bilgisayar ortamından BeagleBoard-xM'e istek yapılması

BeagleBoard-xM ekran görüntüsünün alınabilmesi için, yapılan istekten önce, BeagleBoard-xM içerisinde yer alan VNC programı çalıştırılması gerekmektedir. Bilgisayardan yapılan istek sonucunda, BeagleBoard-xM'e başarılı bir şekilde bağlanmış ve alınan ekran görüntüsü Şekil 4.11'de görülmektedir.



Şekil 4.11. Bilgisayar ortamında BeagleBoard-xM'e ait ekran görüntüsü

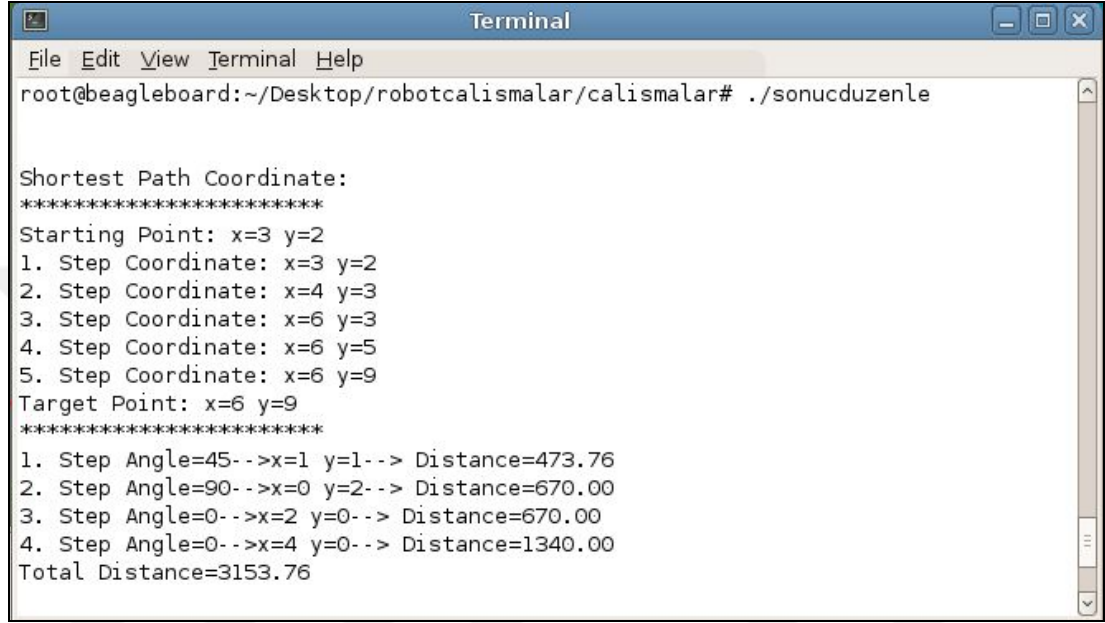
Şekil 4.12'de programın çalıştırılması sonucunda P3-DX gezgin robotunun hedefe ilerleyeceği yol, gerçek kamera görüntüsü ve matris üzerinde gösterilmiştir.



Şekil 4.12. P3-DX ilerlemesi (a) Gerçek görüntü (b) Matris görüntüsü

Şekil 4.13'de programın çalıştırılması ve sonucunda elde edilen ekran görüntüsü sunulmuştur. Gezgin robotun başlangıç noktasından hedefe giderken her adımda izlenmesi gereken yol gösterilmektedir. Ayrıca, hangi adımda kaç derecelik bir açı ile

döneceği ve döndükten sonra ilerleyeceği mesafeler listelenmiştir. Robotun bulunduğu adımdan bir sonraki adıma ilerlemesi sırasında x ve y koordinatları arasındaki fark hesaplanmaktadır. Elde edilen sonuçların en alt kısmında robotun hedefe ilerlerken gitmiş olduğu toplam yol bilgisi hesaplanmaktadır. Program, BeagleBoard-xM üzerine kurulu olan Angström işletim sisteminin terminal ekranından çalıştırılmıştır.



```
Terminal
File Edit View Terminal Help
root@beagleboard:~/Desktop/robotcalismalar/calismalar# ./sonucduzenle

Shortest Path Coordinate:
*****
Starting Point: x=3 y=2
1. Step Coordinate: x=3 y=2
2. Step Coordinate: x=4 y=3
3. Step Coordinate: x=6 y=3
4. Step Coordinate: x=6 y=5
5. Step Coordinate: x=6 y=9
Target Point: x=6 y=9
*****
1. Step Angle=45-->x=1 y=1--> Distance=473.76
2. Step Angle=90-->x=0 y=2--> Distance=670.00
3. Step Angle=0-->x=2 y=0--> Distance=670.00
4. Step Angle=0-->x=4 y=0--> Distance=1340.00
Total Distance=3153.76
```

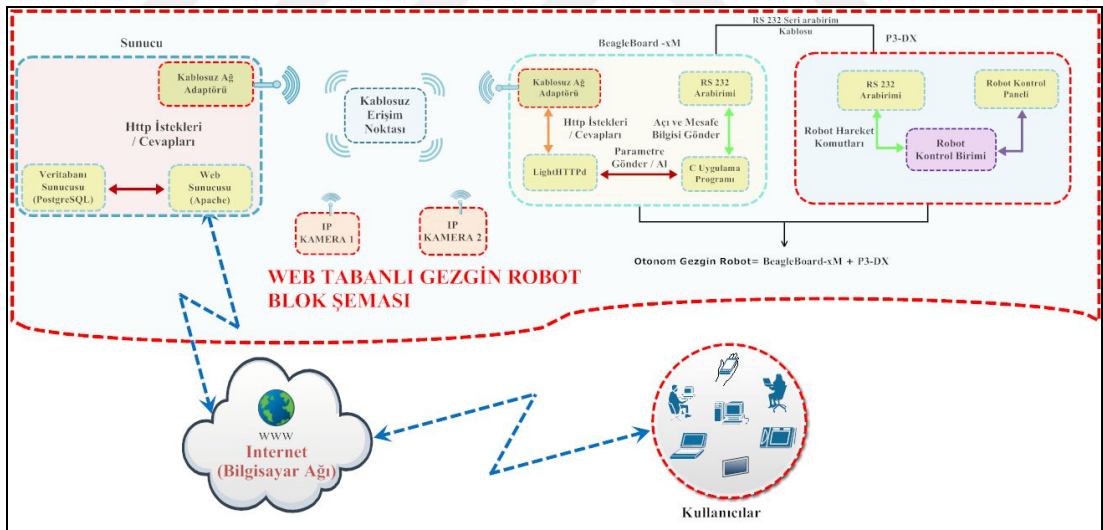
Şekil 4.13. Terminal programından robotun ilerleyeceği yolun listelenmesi

4.2. Web Tabanlı Gezgin Robot Hareket ve Kontrol Sistemi

Tez çalışması kapsamında önerilen bir başka çalışma ise, gezgin robotun uzaktan kontrol ve hareketlerinin takip edilmesine yönelik web ara yüzünün ve kontrol programının tasarlanması, geliştirilmesi ve gerçekleştirilmesidir.

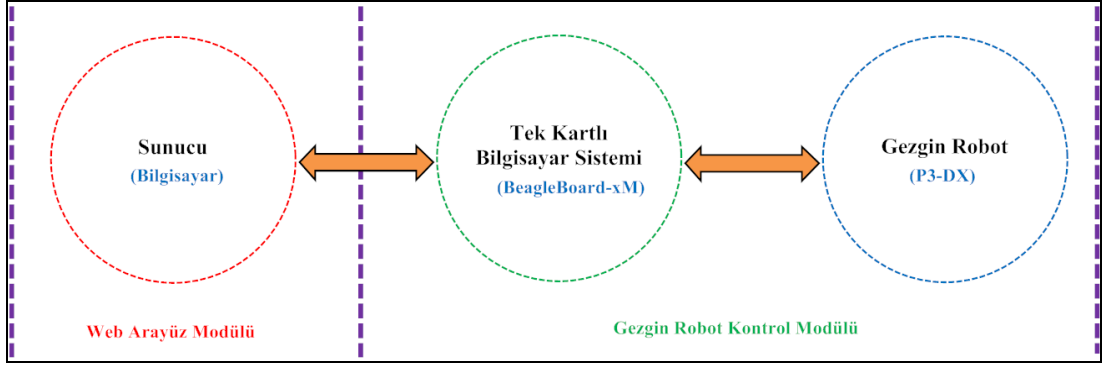
Önerilen çalışmada, kapalı bir laboratuvar ortamında fiziksel olarak bulunan gezgin robotun web üzerinden gerçek zamanlı kontrol ve takibi için sanal laboratuvar ortamı oluşturulmuştur. Web sitesi üzerinde yer alan sanal laboratuvarda, kapalı ortamda yer alan laboratuvarın gerçek zamanlı görüntüsü kablosuz IP kamera kullanılarak elde edilmektedir. Elde edilen bu görüntünün üzerine kullanıcı tarafından sanal engeller ve hedef yerleştirilmektedir. Gezgin robotun, sanal engellere çarpmadan sanal hedefe ulaşması sağlanmakta ve web üzerinden hareketi izlenmektedir. Ayrıca, robotun gönderilen komutlarla uzaktan elle hareket ettirilmesi sağlanmaktadır.

Şekil 4.14’te web tabanlı gezgin robot laboratuvarının çalışmasını gösteren blok şeması sunulmuştur. Ortamda bulunan kameralar, BeagleBoard-xM, uygulama sunucusu olarak kullanılan cihazlar, kablosuz modem üzerinden haberleşmektedir. Kullanıcılar, sanal laboratuvar ortamına, bilgisayar ağı üzerinden veya internet üzerinden erişebilmektedirler. Kullanıcılar tarafından yapılan istekler bilgisayarın üzerinde bulunan uygulama sunucusuna iletilmektedir. İletilen istekler, kablosuz modem üzerinden BeagleBoard-xM üzerinde bulunan lightHTTpd sunucusuna iletilerek gelen parametreler doğrultusunda uygulama programını çalıştırmaktadır. Uygulama programı, aldığı değerlere göre robotun ilerlemesi için hareket komutlarını oluşturarak seri arabirim üzerinden gezgin robota iletmektedir. Bu sayede gezgin robot gerçek laboratuvar ortamındaki hareketini yaparak sanal engellere çarpmadan, sanal hedefe ulaşmaktadır. Robotun hareket ettiği yol bilgisi, mesafe, dönüş açıları, toplam ilerlediği mesafe ve toplam harcadığı zaman kullanıcıya web ara yüzü sayesinde sunulmaktadır. Robotun sanal engeller arasında yol planlaması GA kullanılarak gerçekleştirilmektedir.



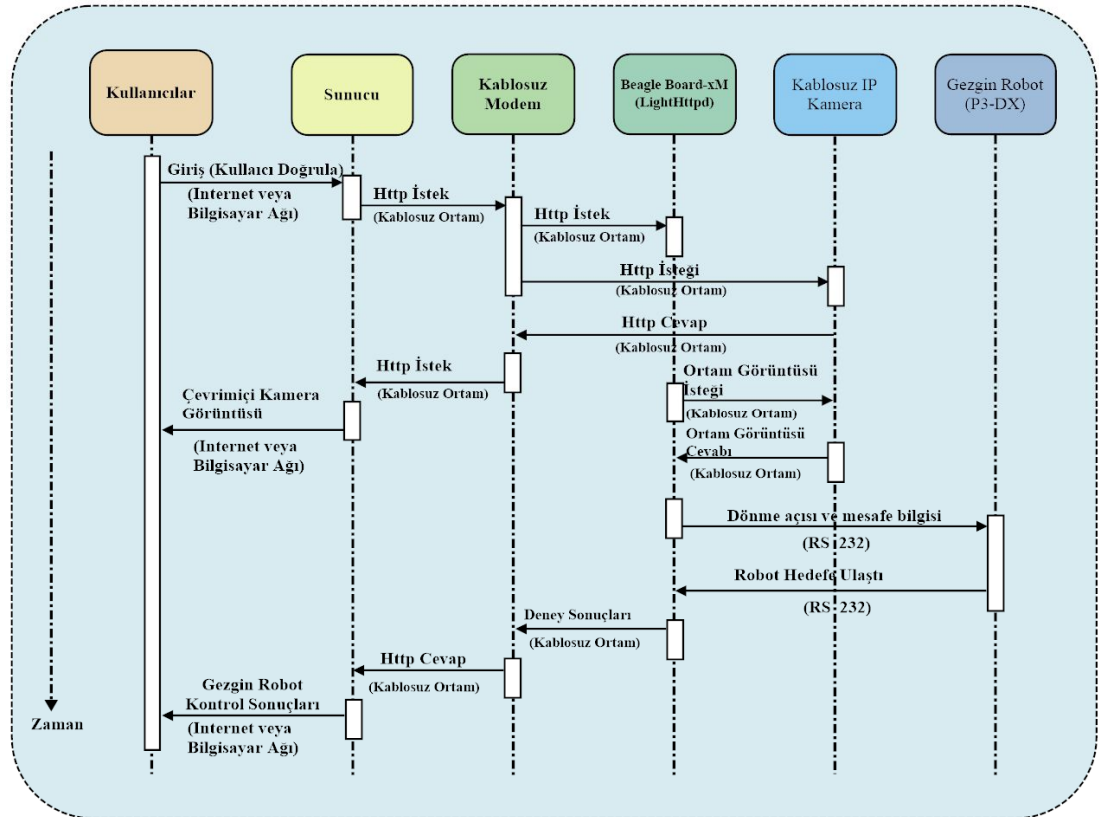
Şekil 4.14. Web tabanlı gezgin robot laboratuvarı çalışmasının blok şeması

Web tabanlı gezgin robot laboratuvar ortamı uyarlanabilir iki modül şeklinde önerilmiştir. Bilgisayar üzerinde bulunan web ara yüz modülü ve TKB ile gezgin robotu kapsayan gezgin robot kontrol modülü olarak tasarlanmıştır. Şekil 4.15’te bu iki modül ve birbirleri ile olan iletişimleri sunulmuştur. Bilgisayar üzerinde kurulu web sunucusu, TKB üzerindeki web sunucusuyla haberleşmekte, TKB de gezgin robotla iletişime geçmektedir.



Şekil 4.15. Web tabanlı gezgin robot laboratuvarı çalışmasının modüler yapısı

Şekil 4.16'da önerilen web tabanlı gezgin robot laboratuvarına ait zamanlama çizelgesi bulunmaktadır. Zamanlama çizelgesinin en üstünde ortamı oluşturan bileşenler sunulmaktadır. Zamanlama çizelgesinde, kullanıcı tarafından yapılan istek sonucunda ortamı oluşturan her bir cihazın birbiri ile ilgili zamansal iletişimi gösterilmektedir. Kullanıcılar, ortama internet veya bilgisayar ağı üzerinden erişirken, laboratuvar ortamındaki cihazlar kablosuz haberleşmektedir. P3-DX ile BeagleBoard-xM arasındaki iletişim seri arabirim ile gerçekleşmektedir.



Şekil 4.16. Web tabanlı gezgin robot laboratuvarı zaman çizelgesi

4.2.1. Web ara yüz modülü

Web ara yüz modülü, kullanıcıların web tabanlı gezgin robot laboratuvarını kullanarak deney yapmasını, uzaktan kontrol ve takip etmesini sağlamaktadır. Oluşturulan modül, bilgisayar üzerinde kurulu olan uygulamaya sunucusu Apache web sunucu ve PostgreSQL veri tabanı sunucusu kurularak hazırlanmıştır. Modül, kablosuz ağ adaptörü kullanılarak, erişim noktası üzerinden platformu oluşturan diğer bileşenler ile kablosuz olarak haberleşmektedir. Geliştirilen platform, Apache web sunucusu üzerinden http iletişim protokolünü kullanarak dış dünya ortamına yayın yapmaktadır. Kullanıcılar, internet üzerinden geliştirilen web-tabanlı platforma erişebilmektedirler. Geliştirilen sistem ve bileşenleri hakkındaki bilgilere siteye giriş yapmadan erişilirken, deneysel ortam için sisteme giriş yapılması gerekmektedir. Sisteme giriş yapan bir kullanıcı;

- Deney ortamını canlı olarak görüntüleyebilmekte,
- Sanal engellerden sakınma ara yüzünü kullanarak, gerçek ortamda, gerçek zamanlı deney (uygulama) yapabilmekte,
- İnsansız mobil robot ara yüzünü kullanarak, uzaktan, gerçek zamanlı mobil robotu kontrol edebilmekte,
- Geliştirilen platformun bileşenleri hakkında detaylı teknik bilgilere ulaşılabilir.

Web ara yüz modülü, sanal engel sakınım ara yüzü ve gezgin robot kontrol ara yüzü olmak üzere iki bileşenden oluşmaktadır.

4.2.1.1. Sanal engel sakınım ara yüzü

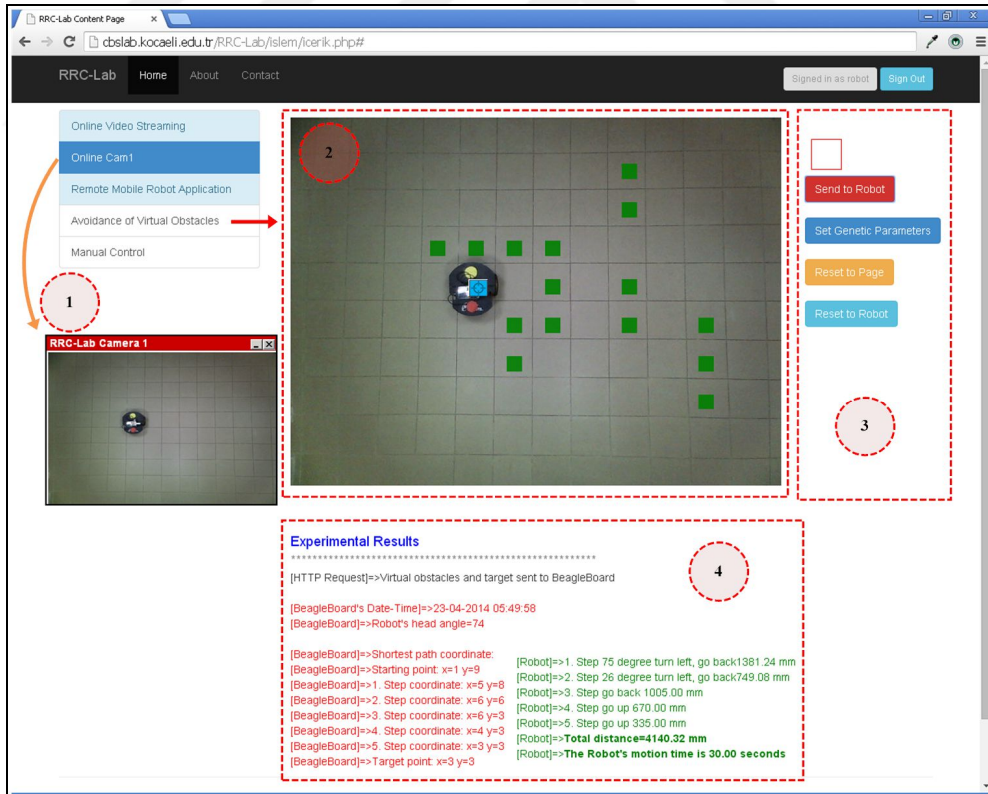
Sanal engel sakınım ara yüzü kısmında, gerçek laboratuvar ortamına ait görüntü üzerinde işlem yapılmaktadır. Kullanıcı bu bölümde, gerçek laboratuvar ortamı görüntüsü üzerine sanal engeller yerleştirmektedir. Ayrıca, robotun gitmesi gereken hedef sanal laboratuvar ortamında bulunmalıdır. Aksi durumda, “hedef bulunamadı” mesajı alınmaktadır. Robotun ortamdaki konumu kameradan alınan görüntü kullanılarak bulunmaktadır. Robot ortamda bulunamazsa, “robot bulunamadı”, ortam hazır değil mesajı verilmektedir. Ortamın haritası ve hedefe ilerlemek için kullanılan GA yöntemine ait değerler parametre olarak robot kontrol modülüne gönderilmektedir. GA yöntemi için varsayılan değerler Tablo 4.1’de sunulmuştur. Tablo 4.1’de sunulan

değerler, deneysel çalışmalar sonucunda elde edilmiş en uygun değerlerdir. Bu değerler kullanıcının isteğine bağlı olarak değiştirilebilmektedir.

Tablo 4.1. GA varsayılan değerleri

Özellik Adı	Değeri
Nüfus Boyutu	20
Adım Sayısı	4
Çaprazlama Oranı	1
Mutasyon Oranı	0,1
Nesil Sayısı	100

Şekil 4.17’de sanal engel sakınım ara yüzüne ait ekran görüntüsü sunulmuştur.



Şekil 4.17. Sanal engel sakınım web ara yüzü

Şekil 4.17’ de sunulan sanal engel sakınım ara yüzü, kamera ham görüntüsü (1), ortam görüntüsü, sanal engel ve hedeflerin bulunduğu deney ortamı (2), GA parametre-

lerinin ayarlandığı ve robota komutların gönderildiği bölüm (3) ve deneysel sonuçların görüntülediği bölüm (4) olmak üzere dört bölümden oluşmaktadır.

Kamera ham görüntüsünün bulunduğu (1) bölümde, robotun gerçek ortamdaki gezintisi sanal engeller olmadan sunulmaktadır.

Sanal engel ve hedeflerin bulunduğu deney ortamı (2) bölümünde, kullanıcı gerçek ortam görüntüsü üzerine engeller yerleştirebilmektedir. Ayrıca ortamın istediği noktasına hedefi koyabilmektedir. Engel ve hedef üst üste konulamayacak şekilde ayarlanmaktadır. Bu bölümde robotun hedefe doğru ilerlerken kat ettiği yol izlenebilmektedir.

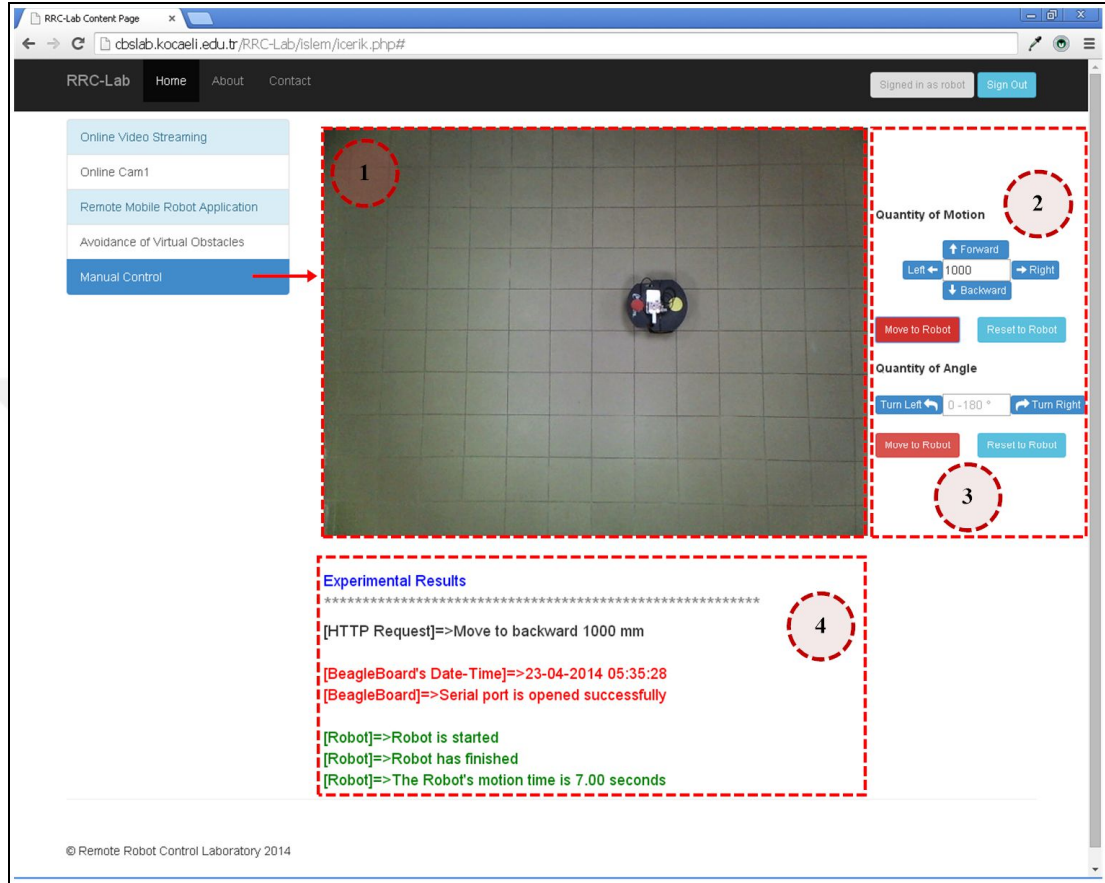
Robota komutların gönderildiği bölüm (3), GA parametreleri, robot ve ortamın sıfırlandığı ve hedef görüntüsünün bulunduğu kısımdır. GA için ayarlama yapılmadığı durumda, varsayılan değerlere göre işlem yapılmaktadır. Bu bölümde yer alan hedef, fare yardımıyla deney ortamına (2) taşınabilmektedir. Ayrıca deney ortamının sıfırlanması veya robotun işlem yapıldıktan sonra açısız anlamda sıfırlanıp hazır duruma getirilmesi sağlanmaktadır.

Deneysel sonuçların görüntülediği bölüm (4), kullanıcılara aşamalı olarak bilgileri sunmaktadır. Deneye başlama zamanı, robotun başlangıç açısı, robotun adım adım takip etmesi gereken yol, robot ve hedef konumu, her adımda gerçekleştirdiği dönüş açısı ve ilerleme mesafesi, toplam ilerlediği yol bilgisi ve kullanıcının istekte bulunduğu andan, robotun hedefine ulaştığı ana kadar olan süre bilgileri hesaplamakta ve kullanıcılara sunulmaktadır.

4.2.1.2. Gezgin robot kontrol ara yüzü

Gezgin robot kontrol ara yüzü kullanılarak, robotun engelsiz ortamda, sağa, sola, ileri ve geri gitmesi sağlanmaktadır. Robot bu sayede, deney ortamında istenilen noktaya ilerletilebilmektedir. Ayrıca robot ilerleme yapmadan, kullanıcının verdiği değerlere göre sadece dönme işlemi de yapabilmektedir. Robot'un deneysel ortamda çalışmadığı gözlemlendiğinde, uzaktan sıfırlama işlemi de yapılabilmektedir. Robotun deney ortamını terk etmemesi için komutlar gönderilirken deney sınırlarından dışarı çıkıp çıkmadığı kontrol edilmektedir.

Şekil 4.18’de gezgin robot kontrol ara yüzü sunulmuştur. Deney ortamı (1), ilerleme değerlerinin ayarlanması (2), dönüş açılarının ayarlanması (3) ve deneysel sonuçların gösterilmesi (4) olmak üzere dört kısımdan oluşmaktadır.



Şekil 4.18. Gezgin robot kontrol ara yüzü

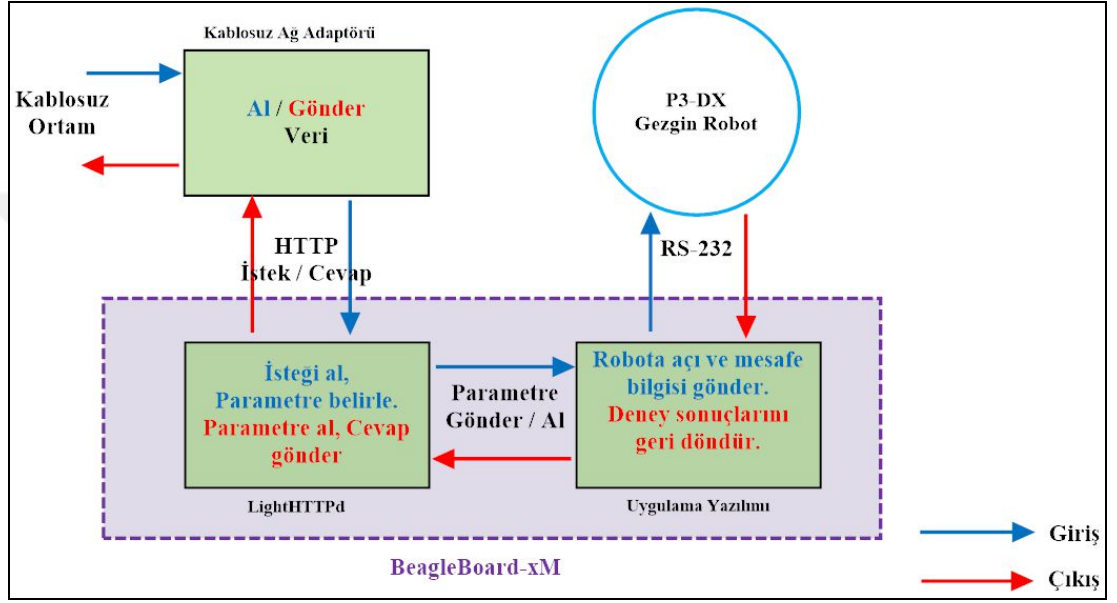
Deney ortamı (1) bölümünde robotun hareketleri izlenmektedir. İlerleme değerlerinin ayarlanması (2) bölümünde, 0 ila 1000 mm arasında değerler verilerek robotun ilgili yöne doğru ilerlemesi sağlanmaktadır.

Dönüş açılarının ayarlanması (3) bölümünde ise, sağa veya sola 0^0 ila 180^0 arasında değerler verilerek ilgili yöne doğru dönmesi sağlanmaktadır.

Deneysel sonuçların gösterilmesi (4) bölümünde, gezgin robot kontrol programına yapılan istek, deney zamanı, robot durumları ve robotun çalışma süresi sunulmaktadır. Bu bölümde yapılan tüm istekler, bilgisayar üzerindeki web sunucusu sayesinde, gezgin robot kontrol modülüne iletilmekte, değerlendirilmekte ve robotun hareketi sağlanmaktadır.

4.2.2. Gezgin robot kontrol modülü

Gezgin robot kontrol modülü iki bölümden meydana gelmektedir. TKB olarak kullanılan BeagleBoard-xM üzerinde çalıştırılan kontrol yazılımları ve gezgin robot olarak kullanılan P3-DX' i kapsamaktadır. BeagleBoard-xM, P3-DX ile seri arabirim üzerinden haberleşirken, web ara yüz modülü ile kablosuz olarak haberleşmektedir. Şekil 4.19' da gezgin robot kontrol modülünün yapısına ait şema sunulmaktadır.



Şekil 4.19. Gezgin robot kontrol modülü

BeagleBoard-xM'in üzerine Angström işletim sistemi, OpenCV görüntü işleme kütüphanesi ve LightHTTPd web sunucusu kurulmuştur. Ayrıca web ara yüz modülü ile kablosuz iletişim kurulması için USB kablosuz ağ adaptörü kullanılmıştır. Web ara yüz modülü üzerinden gelen istekler, BeagleBoard-xM üzerinde kurulu olan LightHTTPd web sunucusuna iletilmektedir. LightHTTPd web sunucusu bu istekleri değerlendirdikten sonra C programlama dili kullanılarak yazılmış kontrol programlarından uygun olanı çalıştırılmaktadır. Web ara yüz modülü üzerinden, engellerin ve hedefin olduğu laboratuvar ortamının resmi geldiğinde görüntü işleme teknikleri kullanılarak yazılmış program çalıştırılmaktadır. Robotun konumu kablosuz IP kameradan alınan görüntü ile program tarafından tespit edilmektedir. Robotun konumu ve yönü, engeller ve hedefin matris içindeki yeri belirlendikten sonra GA kullanılarak en kısa yol bulunmaktadır. Bulunan en kısa yola göre robotun dönüş açıları ve ilerleyeceği mesafeler hesaplanarak robotun anlayacağı komutlar haline getirilmektedir.

Elde edilen bu komutlar sırasıyla robota gönderilmekte ve robot hedefe doğru ilerlemektedir. Bu işlem yapılırken kullanıcı robotun hareketini web sayfası üzerinden izleyebilmektedir. Robot hareketini tamamladığında, kullanıcı ekranda robotun gittiği yolun koordinatlarını, giderken yaptığı hareket manevralarını, ilerlediği toplam mesafeyi ve işlemi yaparken harcamış olduğu süreyi görmektedir.

Web ara yüz modülü üzerinden, ileri, geri, sağa ve sola bilgileri ile beraber 0 ila 1000 mm arasında mesafe bilgisi geldiğinde robotun manüel olarak hareket etmesini sağlayan program çalıştırılmaktadır. Bu programda ilerlemenin dışında 0^0 ila 180^0 arasında sağa veya sola doğru dönmeyi sağlayan komutlar da yer almaktadır. Göndermiş olduğu komut sonunda kullanıcı robotun hareketini takip edebilir. Ayrıca ekranda robotun durumunu, ilerleyeceği veya döneceği mesafeyi ve işlemleri yaparken harcanan süreyi görmektedir.

4.2.2.1. Gezgin robot kontrol programı

Önerilen sisteme ait en önemli bileşendir. BeagleBoard-xM içerisinde bulunan LightHTTPd web sunucusu tarafından değerlendirilen istekler sonucunda, robotun sanal engeller arasında dolaşarak hedefe ulaşması istendiğinde çalıştırılmaktadır. Şekil 4.20’de gezgin robot kontrol modülünde kullanılan, kontrol programına ait akış şeması sunulmuştur.

LightHTTPd sunucusu tarafından tetiklenen program almış olduğu parametre değerlerine göre çalışmaktadır. Programın aldığı parametre değerleri GA ve ortam matrisine yöneliktir. Program, kablosuz IP kameradan ortama ait görüntü alınmakta ve gezgin robotun konum, matris içindeki yeri ve robotun başlangıç açısı hesaplanmaktadır.

Robot konumu ve ortam matrisi içerisindeki yeri hesaplandıktan sonra, yol planlama algoritması kullanılarak en kısa yol hesaplanmaktadır. Yol planlama algoritması olarak GA tabanlı yöntem kullanılmaktadır. GA tabanlı yol planlama algoritması Bölüm 3’te detaylı olarak anlatılmıştır.

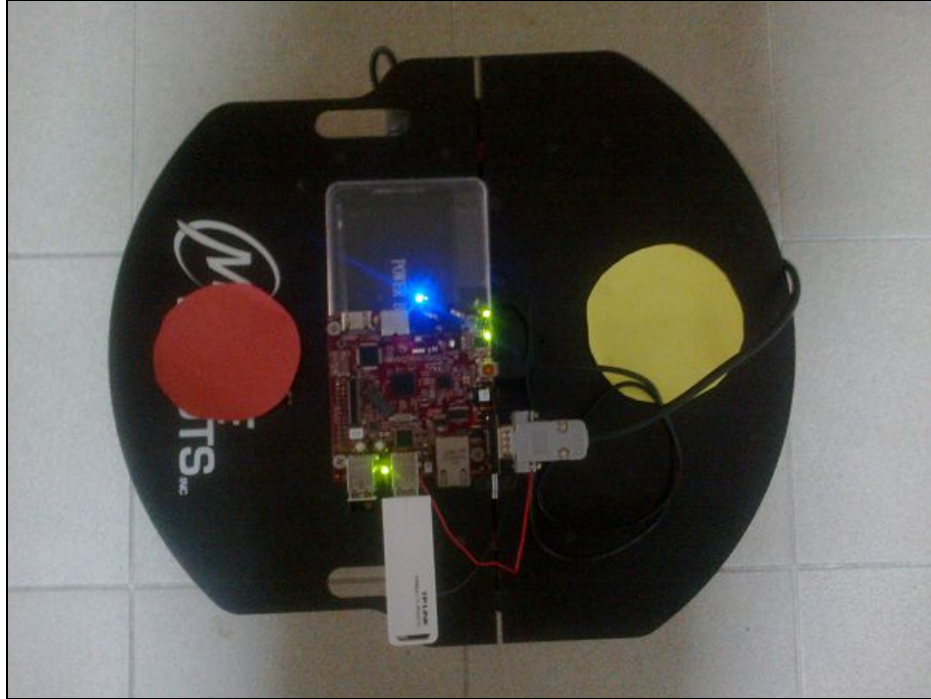
En kısa yol bilgilerine göre robotun her adımda yapacağı dönüş açısı ve ilerleyeceği mesafe bilgileri hesaplanmakta ve P3-DX’in anlayacağı komutlara dönüştürülerek seri arabirim üzerinden gönderilmektedir.

Şekil 4.21’de gezgin robot kontrol programının terminal ekranından çalıştırılması sonucunda elde edilen bilgiler görülmektedir. Robotun üzerinde bulunan sarı ve kırmızı renklere ait koordinat değerleri, robotun açısı, robotun koordinatları, matris içerisindeki yeri, en kısa yol bilgileri, robotun her adımda gösterdiği ilerleme miktarı ve robotun hedefe ilerlerken geçirdiği süre bilgileri sunulmuştur.

4.2.2.2. Gezgin robot dönme açısının hesaplanması

Gezgin robotun fiziksel olarak yer aldığı laboratuvar ortamında yerinin, yönünün ve açısının doğru bir şekilde bulunması robotun hedefe, engellere çarpmadan ulaşmasında çok önemlidir. Çalışmada, robotun yer ve açısının hesaplanması için kablosuz IP kameradan alınan görüntü kullanılmaktadır.

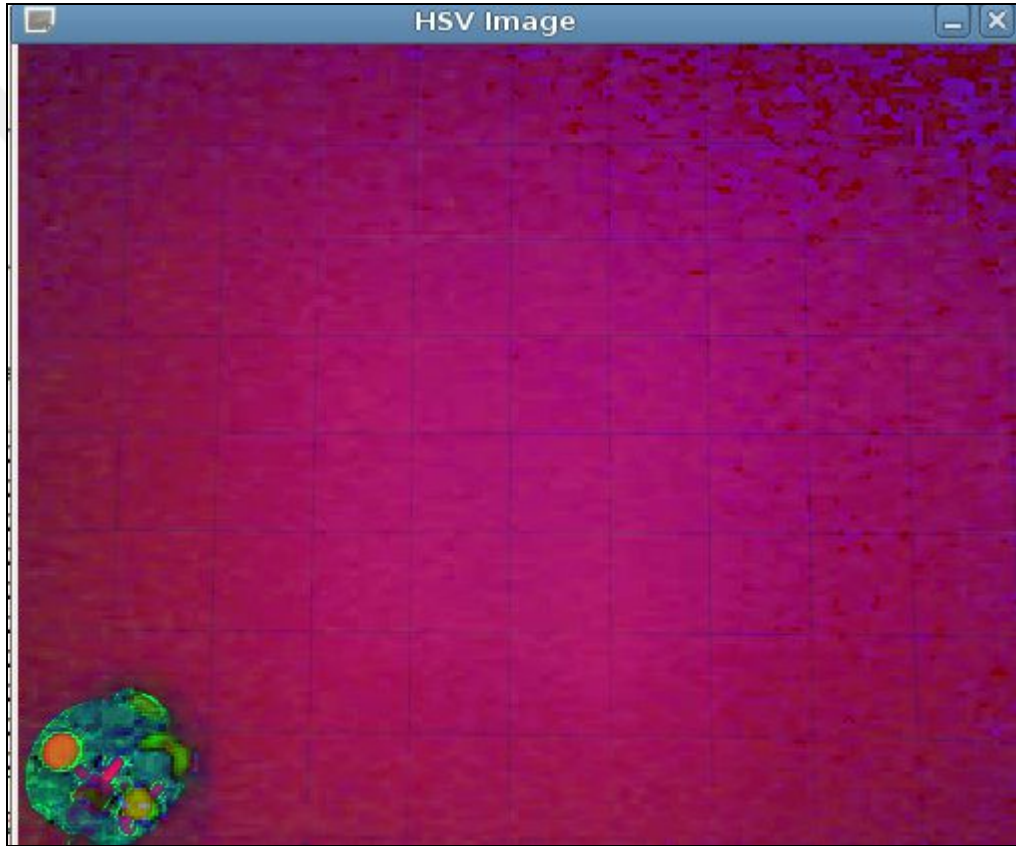
Gezgin robotun yönünün belirlenmesi renk tabanlı görüntü işleme tekniği kullanılarak gerçekleştirilmektedir. Gezgin robotun üzerine sarı ve kırmızı olmak üzere iki adet renkli nesne yerleştirilmiştir. Sarı renkli nesne robotun ön tarafını kırmızı renkli nesne ise robotun arka tarafını temsil etmektedir. Şekil 4.22’de laboratuvar ortamında kullanılan P3-DX’e ait görüntü bulunmaktadır. P3-DX’in ön tarafında sarı, arka tarafında ise kırmızı nesne ve ortada BeagleBoard-xM bulunmaktadır.



Şekil 4.22. P3-DX’e ait üstten görünüm

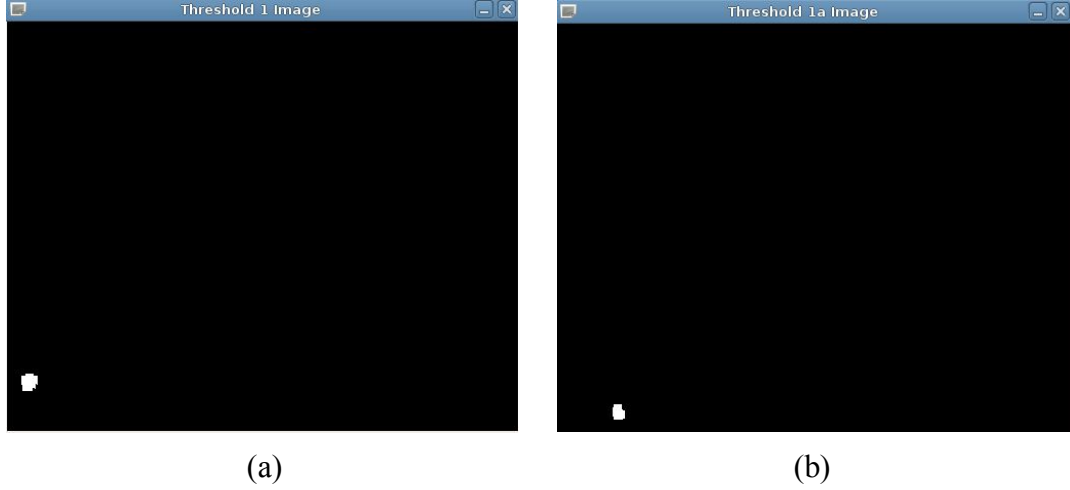
BeagleBoard-xM'in çalışması için 5 V, 1A'lik bir batarya kullanılmaktadır. Ayrıca Şekil 4.22'de görüldüğü üzere kablosuz iletişim sağlamak için, kablosuz ağ adaptörü kullanılmıştır.

Açı ve konum bulma çalışmasının ilk aşamasında, kablosuz IP kameradan görüntü alınmaktadır. Kablosuz IP kameradan görüntü alındıktan sonra, Gauss filtresi ile görüntü üzerindeki gürültüler yok edilmektedir. RGB renk uzayındaki görüntü HSV renk uzayına çevrilmektedir. Şekil 4.23'de HSV renk uzayında oluşan yeni görüntü sunulmuştur.



Şekil 4.23. Gezgin robotun HSV renk uzayındaki görüntüsü

HSV renk uzayına çevrilen görüntü kullanılarak iki adet görüntü yaratılmaktadır. İlk görüntü üzerinde sadece sarı renkler bırakılarak eşikleme işlemi, ikinci görüntüde ise sadece kırmızı renkler bırakılarak eşikleme işlemi uygulanmaktadır. Bu işlemler sonunda, iki adet farklı ikili görüntü elde edilmektedir. Ortamda bulunan hatalı piksellerin yok edilmesi amacıyla, iki görüntü üzerinde morfolojik aşındırma ve genişletme işlemleri uygulanmaktadır. Şekil 4.24'de morfolojik işlemlerin uygulanması sonucunda oluşan iki görüntü sunulmuştur.



Şekil 4.24. İkili görüntü (a) Sarı renk (b) Kırmızı renk

Elde edilen ikili görüntünün her biri için 0 ve 1. dereceden momentleri hesaplanmaktadır. Sarı ve kırmızı renkli nesnelere ait koordinatlar, bölüm 3'te anlatıldığı şekilde 0. ve 1. dereceden momentler kullanılarak hesaplanmaktadır. Sarı renkli nesnenin koordinatları S_x ve S_y , kırmızı renkli nesnenin koordinatları K_x ve K_y olarak ifade edilmektedir. θ_B robotun başlangıç konumundaki açısını ifade etmektedir. Eşitlik (4.3) kullanılarak robotun başlangıç açısı hesaplanmaktadır.

$$\theta_B = \arctan\left(\frac{S_y - K_y}{S_x - K_x}\right) \quad (4.3)$$

Sarı ve kırmızı renklerin birbirine göre konumlarına bakılarak, robot başlangıç açısı güncellenmektedir. Açılarının güncellenmesi sırasında Eşitlik (4.4)'de sunulan denklemler kullanılmaktadır. Bu eşitlikler kullanılarak robotun hangi yönde ilerleyeceği bulunmaktadır.

$$\theta_B = \begin{cases} \theta_B + 180 & \rightarrow (K_x > S_x) \vee (K_y < S_y) \\ 180 - \theta_B & \rightarrow (K_x > S_x) \vee (K_y > S_y) \\ \theta_B & \rightarrow (K_x < S_x) \vee (K_y > S_y) \\ 360 - \theta_B & \rightarrow (K_x < S_x) \vee (K_y < S_y) \\ 180 & \rightarrow (K_x > S_x) \vee (\theta_B = 0) \\ 0 & \rightarrow (K_x < S_x) \vee (\theta_B = 0) \\ 90 & \rightarrow (K_y > S_y) \vee (\theta_B = 90) \\ 270 & \rightarrow (K_y < S_y) \vee (\theta_B = 90) \end{cases} \quad (4.4)$$

Şekil 4.25’de deney ortamında bulunan gezgin robot tespit edilmiş olup, üzerinde yer alan renkli nesnelerin konumları bulunmuştur.



Şekil 4.25. Deney ortamında bulunan robotun tespit edilmesi

Örnek çalışmada K_x ve K_y (58,05; 377,89), S_x ve S_y (21,56; 352,93) olarak hesaplanmıştır. Hesaplanan değerler, Eşitlik (4.3)’de yerine konulduğunda $\theta_B = 34^\circ$ olarak bulunmuştur. Koordinat değerleri incelendiğinde, $(K_x > S_x)$ ve $(K_y > S_y)$ olduğu gözlenmektedir. Bu durumda açılar yeniden hesaplanarak, robotun dönüş açısı, $\theta_B = 180 - \theta_B \Rightarrow \theta_B = 146^\circ$ olarak güncellenmektedir. Yapılan işlem sonucunda robot (+x) yönüne doğru sağ taraftan 146° dönecektir. Koordinatlar bulunduktan sonra hesaplanan θ_B açısı (-x) yönünde bulunan açıyı ifade etmektedir. Bu durumda, robotun başlangıç konumu ve bir sonraki adımda bulunacağı konum Eşitlik (4.5)’te verildiği şekilde tanımlanmaktadır. Eşitlik (4.6) kullanılarak robotun ilerleyeceği noktanın koordinatları ile bulunduğu noktanın koordinatları arasındaki farklar hesaplanmaktadır. Robotun ilerlemesi gereken yol ise Eşitlik (4.7)’den yararlanılarak Öklid uzaklığı şeklinde hesaplanmaktadır.

$$\mathbf{R}_B = \begin{bmatrix} \mathbf{R}_x^B \\ \mathbf{R}_y^B \\ \theta_B \end{bmatrix} \quad \mathbf{R}_{(B+1)} = \begin{bmatrix} \mathbf{R}_x^{(B+1)} \\ \mathbf{R}_y^{(B+1)} \\ \theta_{(B+1)} \end{bmatrix} \quad (4.5)$$

$$\mathbf{R}_{\Delta_x} = |\mathbf{R}_x^{(B+1)} - \mathbf{R}_x^B| \quad \mathbf{R}_{\Delta_y} = |\mathbf{R}_y^{(B+1)} - \mathbf{R}_y^B| \quad (4.6)$$

$$\mathbf{R}_{\text{Mesafe}} = \sqrt{|\mathbf{R}_x^{(B+1)} - \mathbf{R}_x^B|^2 + |\mathbf{R}_y^{(B+1)} - \mathbf{R}_y^B|^2} \quad (4.7)$$

Robotun bulunduğu konumdan bir sonraki konuma gitmesi için ilk olarak dönüş açısının hesaplanması gerekmektedir. Dönüş açısı Eşitlik (4.8) kullanılarak hesaplanmaktadır.

$$\theta_D = \arctan\left(\frac{\mathbf{R}_{\Delta_y}}{\mathbf{R}_{\Delta_x}}\right) \quad (4.8)$$

Gezgin robotun gerçek dönme açılarının hesaplanması Tablo 4.2’de sunulmuştur.

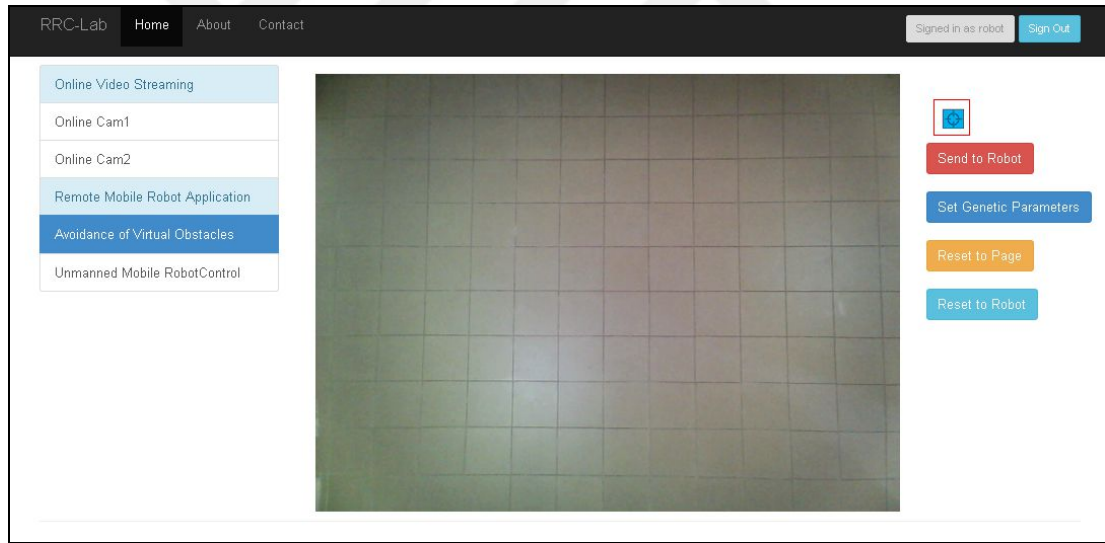
Tablo 4.2. Robotun dönüş açılarının hesaplanması

Koşul	Robot Yönü	Robot Açısı (θ)
$(\mathbf{R}_{\Delta_x} > 0) \vee (\mathbf{R}_{\Delta_y} > 0)$	İleri	$\theta = \theta_B + \theta_D$ $\theta_B < \theta_D \rightarrow \theta = 360 - (\theta_D - \theta_B)$
$(\mathbf{R}_{\Delta_x} < 0) \vee (\mathbf{R}_{\Delta_y} > 0)$	Geri	$\theta_B > \theta_D \rightarrow \theta = \theta_B - \theta_D$ $\theta = \theta + 180$
$(\mathbf{R}_{\Delta_x} < 0) \vee (\mathbf{R}_{\Delta_y} = 0)$	Geri	$\theta = \theta_B$
$(\mathbf{R}_{\Delta_x} > 0) \vee (\mathbf{R}_{\Delta_y} = 0)$	İleri	$\theta = \theta_B$
$(\mathbf{R}_{\Delta_x} = 0) \vee (\mathbf{R}_{\Delta_y} < 0)$	İleri	$\theta_B < 90 \rightarrow \theta = 360 - (90 - \theta_B)$ $\theta_B > 90 \rightarrow \theta = \theta_B - 90$
$(\mathbf{R}_{\Delta_x} = 0) \vee (\mathbf{R}_{\Delta_y} > 0)$	İleri	$\theta = \theta_B + 90$ $\theta_D = 180 - \theta_D$
$(\mathbf{R}_{\Delta_x} < 0) \vee (\mathbf{R}_{\Delta_y} < 0)$	Geri	$\theta_B < \theta_D \rightarrow \theta = 360 - (\theta_D - \theta_B)$ $\theta_B > \theta_D \rightarrow \theta = \theta_B - \theta_D$ $\theta = \theta + 180$
$(\mathbf{R}_{\Delta_x} > 0) \vee (\mathbf{R}_{\Delta_y} < 0)$	İleri	$\theta_B < \theta_D \rightarrow \theta = 360 - (\theta_D - \theta_B)$ $\theta_B > \theta_D \rightarrow \theta = \theta_B - \theta_D$

Tablo 4.2’de robota ait gerçek dönme açısı θ , hesaplanan dönüş açısı θ_D ve robotun başlangıç açısı θ_B dikkate alınarak hesaplanmaktadır. Ayrıca Tablo 4.2’de robotun ilerleyeceği yön bilgisi de bulunmaktadır.

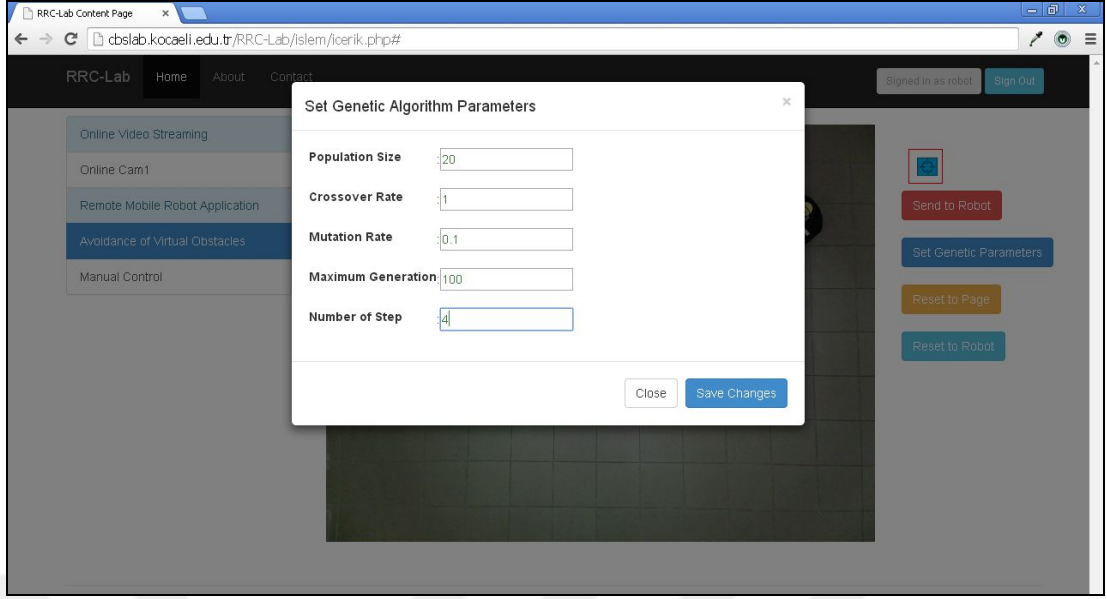
4.2.3. Sistemin uygulanması ve sonuçları

Yapılan çalışmada, internet veya bilgisayar ağı üzerinden gezgin robotu kontrol ettirmek için, web tabanlı gezgin robot laboratuvar sistemi önerilmektedir. Sistem, <http://cbslab.kocaeli.edu.tr> adresinden yayın yapmaktadır. Sistemde, kayıtlı bir kullanıcı giriş yaptıktan sonra, ortamda hazır durumda bulunan P3-DX robotunu hareket ettirebilmektedir. Ayrıca, ortamın gerçek görüntüsü üzerine sanal engeller ve sanal hedef yerleştirilmektedir. Gerçek ortamda bulunan gezgin robot, sanal engellere çarpmadan, sanal hedefe ulaşmaktadır. Şekil 4.26’da oluşturulan laboratuvar ortamı sunulmuştur.



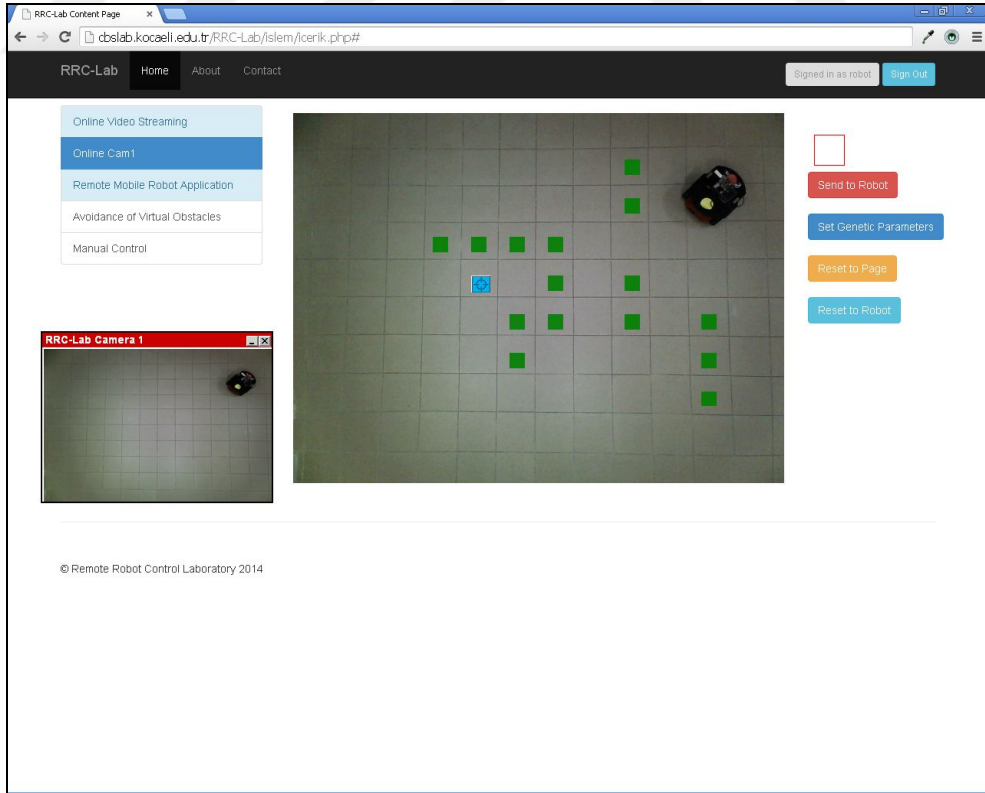
Şekil 4.26. Web tabanlı gezgin robot laboratuvar ortamı

Şekil 4.26’da robotun sanal engellerden sakınması işlemi seçilmiş durumdadır. Kullanıcı, varsayılan GA değerlerini kullanabileceği gibi, yeni değerler de belirleyebilmektedir. Kullanıcı, engeller ve hedeften oluşan ortamı, gezgin robot kontrol modülüne gönderebilmektedir. Ayrıca, robot ve ortam görüntüsünü sıfırlama işlemini de gerçekleştirebilmektedir. Şekil 4.26’da bulunan deneysel ortam için yapılan çalışmalarda, gezgin robotun hedefe ulaşabilmesi için adım sayısının dört veya beş olması yeterli olmaktadır. Şekil 4.27’de GA parametrelerinin ayarlanmasına yönelik görüntü sunulmuştur.



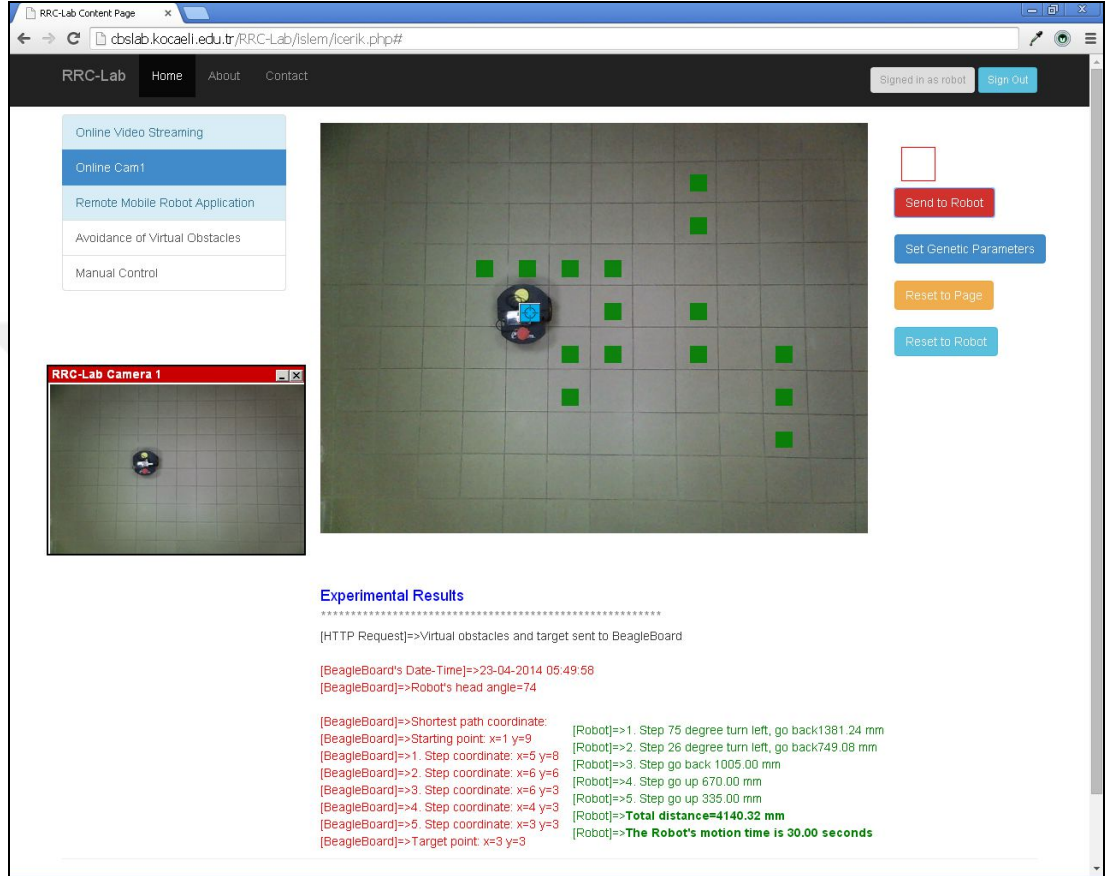
Şekil 4.27. GA parametrelerinin ayarlanması

Şekil 4.28’de örnek bir durum oluşturulmuştur. P3-DX bulunduğu noktadan mavi renkte belirlenmiş olan hedefe doğru, yeşil renkteki engellere çarpmadan hareket ettirilecektir.



Şekil 4.28. Örnek deney ortamı

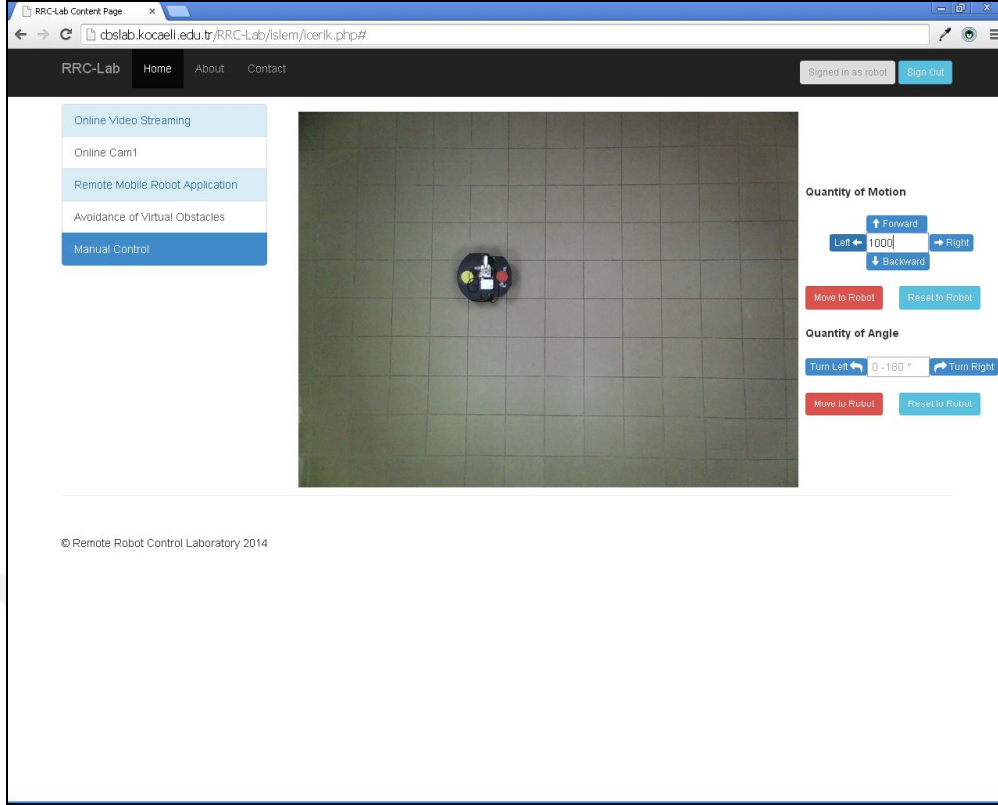
Şekil 4.29’da gezgin robot hareketini gerçekleştirmiş ve hedefe ulaşmıştır. Hedefe ulaşırken elde ettiği sonuçlar, deneyi yapan kullanıcıya bilgi olarak sunulmaktadır. Kullanıcı aynı zamanda, deney esnasında, P3-DX’in hareketlerini kamera görüntüsü sayesinde izleyebilmektedir.



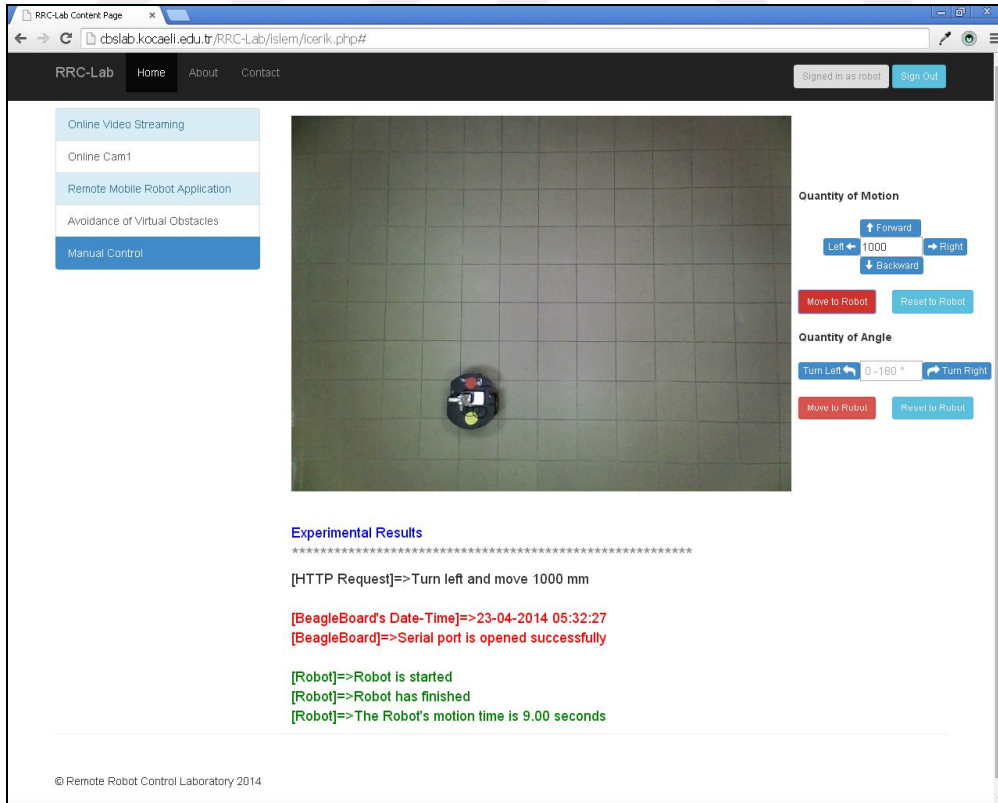
Şekil 4.29. Örnek deney sonuçları

Web tabanlı gezgin robot laboratuvar ortamında bulunan P3-DX robotu uzaktan kullanıcı isteğine göre hareket ettirilebilmektedir. Kullanıcı, ileri, geri, sağa veya sola doğru 0 ila 1000 mm arasında mesafe bilgisi göndererek P3-DX'in hareketini sağlamaktadır. Ayrıca, P3-DX'e sadece dönüş açısı gönderilebilmektedir. Robota gönderilecek olan dönüş açısı $0 - 180^0$ arasında sola veya sağa doğru hareket edecek şekilde dir.

Şekil 4.30'da P3-DX'e sola doğru 1000 mm gitmesi için komut gönderilmekte, Şekil 4.31'de ise robotun sola dönerek 1000 mm ilerlediği gözlenmektedir. Ayrıca, deney sonuçları kısmında, istekte bulunulan komut, deney zamanı, robotun durumu ve işlemi tamamlama süreleri bulunmaktadır.



Şekil 4.30. P3-DX'in kullanıcı tarafından hareket ettirilmesi



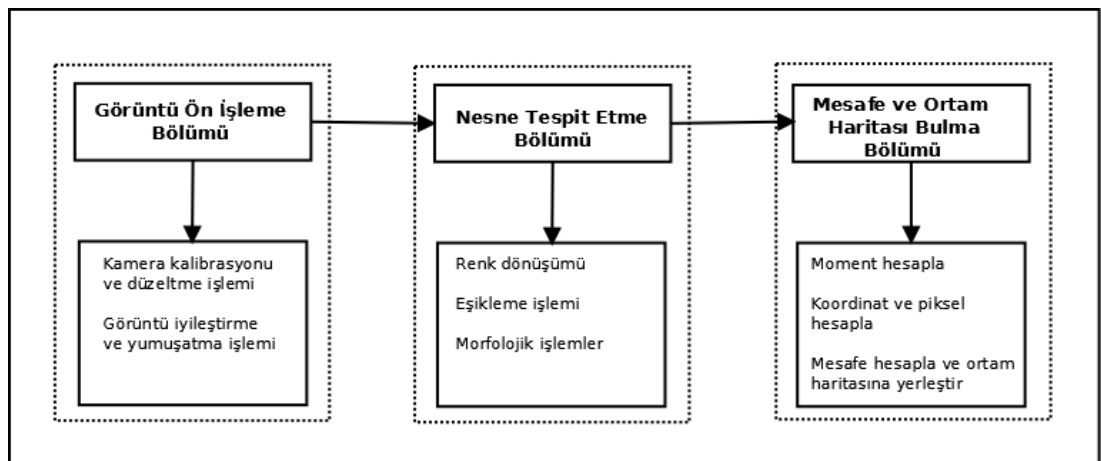
Şekil 4.31. P3-DX hareketinin görüntülenmesi ve sonuçların alınması

4.3. Stereo Görme Tabanlı Gezgin Robot Hareket Planlama Sistemi

Tez çalışmasında önerilen bir diğer yöntem, gezgin robotların bulunduğu ortamda engellere çarpmadan ilerlemesi için stereo görme özelliğine dayanan gezinti sistemidir. Çalışmada, gezgin robotun çalışma ortamında bulunan engellere çarpmadan hedefe ulaşması sağlanmaktadır. Gezgin robotun hedefe ulaşması sırasında en kısa yolu bulması için GA kullanılmaktadır.

Stereo görme tabanlı gezinti sisteminde, P3-DX gezgin robotu, BeagleBoard-xM TKB, iki adet USB kamera, 5 V, 1 A'lık batarya, ortamda bulunan engeller için kırmızı renkli nesnelere, hedef için mavi renkli nesnelere kullanılmaktadır. Bu çalışmada, önceki çalışmalarda kullanılan, kablosuz IP kamera devreden çıkarılmıştır. Sistemde kullanılan kameralar, gezgin robotun üzerine yerleştirilerek robotun hareketi sağlanmaktadır. BeagleBoard-xM'in çalıştırılması laboratuvar ortamında bulunan bilgisayardan bağlantı kurularak gerçekleştirilmektedir.

Bölüm 4.1'de gezgin robotlar için önerilen gömülü hareket planlama sistemi üç blok olarak gerçekleştirilmiştir. Bu sistemde, ilk blok olan GİB, kamara girdilerinden ortam haritasının çıkarılmasını hedeflemektedir. Bu kapsamda yapılan yeni çalışmada, stereo görme tabanlı yaklaşım kullanılarak ortamın ızgara tabanlı haritası oluşturulmaktadır. Şekil 4.32'de stereo görme kullanarak yapılan çalışmada ortam haritasının çıkarılması için önerilen model sunulmuştur.

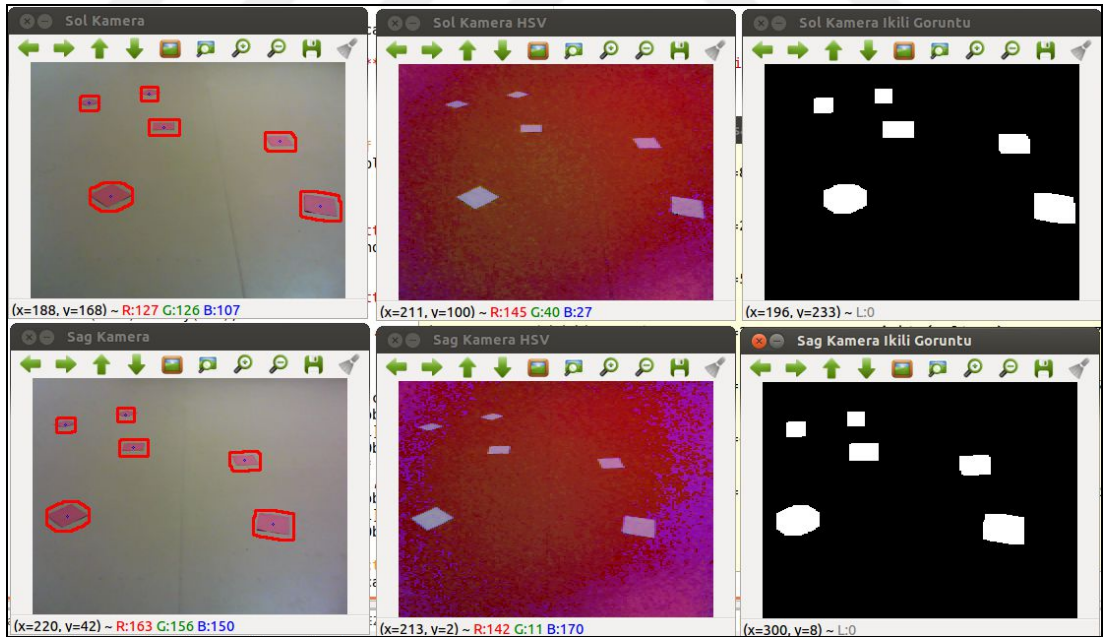


Şekil 4.32. Stereo görme tabanlı önerilen model

Önerilen modelde, işlemler Görüntü Ön İşleme, Nesne Tespit Etme ve Mesafe ve Ortam Haritası Bulma bölümleri olmak üzere üç aşamada gerçekleştirilmektedir.

Görüntü ön işleme bölümünde, kamera kalibrasyon işlemi gerçekleştirilmektedir. Kalibrasyon sonucunda elde edilen değerler kullanılarak sağ ve sol görüntü düzeltme işlemi yapılmaktadır. Kalibrasyon işlemi, matematiksel olarak maliyetli bir iş olduğundan program her çalıştırıldığında yapılmamaktadır. Çevrimdışı olarak önceden çalıştırılmakta ve değerler elde edilmektedir. Elde edilen değerler kullanılarak, görüntü düzeltmesi gerçekleştirilmektedir. Daha sonra sağ ve sol kamera görüntüleri üzerinde, Gauss filtresi uygulanarak görüntü iyileştirme ve yumuşatma işlemi yapılmaktadır.

Nesne tespit etme bölümünde, ortamda bulunan nesnelere belirlenmektedir. Kırmızı renkli nesnelere engelleri, mavi renkli nesne ise hedefi temsil etmektedir. Sol ve sağ kameradan alınan görüntüler, HSV renk uzayına dönüştürülmektedir. Ortamda bulunması istenen nesnelere renk değerleri belirlenerek eşikleme işlemi uygulanmaktadır. Eşikleme işleminden sonra, morfolojik olarak genişleme ve aşındırma işlemleri uygulanmaktadır. Ayrıca, her bir nesne grubuna ait kenarlar tespit edilmekte ve tek bir nesne olarak işaretlenmektedir.



Şekil 4.33. Stereo görüntü çiftleri üzerinde nesne tespiti

Şekil 4.33'de sol ve sağ kameradan alınan görüntüler üzerinde, görüntü ön işleme adımları uygulanmış, sonrasında nesne tespit işlemine geçilmiştir. Ortamda sadece engelleri ifade eden kırmızı renkler bulunmaktadır.



Şekil 4.35. Nesne yükseklik ve genişlik bilgileri

Ortamda bulunan nesnelerin koordinatları hesaplanırken, moment alma yöntemi kullanılmaktadır. Ortamda bulunan her bir nesneye ait moment hesabı yapılırken, ikili görüntü üzerinde sadece o anda değerlendirilmeye alınan nesne bırakılmakta ve moment hesabı yapılmaktadır. Koordinatlar bulunurken sıfır ve birinci dereceden momentler hesaplanmış ve kullanılmıştır.

Nesnelerin kameralara olan uzaklığının bulunması için, görüntü düzlemi üzerinde bulunan görüntülerin x koordinatları kullanılmaktadır. Sol ve sağ görüntü üzerinde yer alan aynı noktalar arasındaki x koordinat farkı, mesafe kestirimi yapmak için kullanılmaktadır. Yakında bulunan nesneler için x koordinat farkı yüksek değerler olurken, uzaktaki nesneler için küçük değerler olmaktadır. Mesafe hesaplanırken, Eşitlik (4.9)'da verilen denklem kullanılmaktadır. D_i nesneye ait i . noktanın kameralara olan mesafesi, b iki kamera arasındaki uzaklık, f odak noktası, d ise iki görüntüde aynı noktalara düşen x koordinat farkıdır. Ölçülen mesafenin doğruluğunu artırmak amacıyla ilgili nesne üzerinden çok sayıda örtüşen nokta alınmaktadır. Bu noktaların ortalaması alınarak mesafe kestirimi yapılmaktadır. Eşitlik (4.10)'da ortamda bulunan her bir nesneye ait ortalama değer hesaplanmaktadır.

$$D_i = \frac{fb}{d} \quad (4.9)$$

$$D_{\text{ort}} = \frac{1}{N} \sum_{i=1}^N D_i \quad (4.10)$$

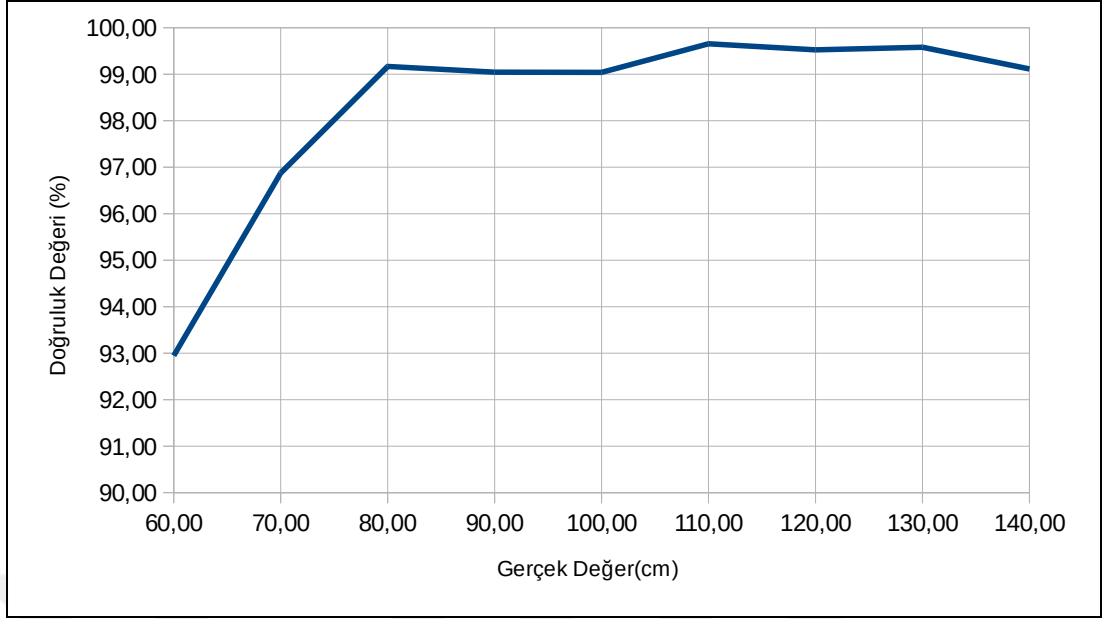
Örnek deneysel çalışmada, ortamda bulunan aynı nesne için eşleşen farklı noktalar alınarak ortalama ölçüm değeri oluşturulmuştur. Ayrıca aynı nesnenin, farklı uzaklıklardan çok sayıda ölçüm değerleri alınarak ortalama değer hesaplanmıştır. Bu çalışma yapılırken, stereo kameraların karşısında bir adet nesne konularak ölçüm değerleri alınmıştır. Tablo 4.3’de alınan ölçüm değerleri sunulmuştur.

Tablo 4.3. Örnek ölçüm değerleri

Gerçek Değer(cm)	Ölçülen Ortalama Değer (cm)	Ortalama Piksel Değeri	Ölçümler arası fark	Doğruluk (%)	Hata (%)
60	64,55	45,68	4,55	92,95	7,05
70	72,28	38,73	2,28	96,88	3,12
80	80,48	33,28	0,48	99,17	0,83
90	89,99	27,95	0,02	99,04	0,96
100	99,04	25,25	0,96	99,04	0,96
110	109,62	22,20	0,38	99,65	0,35
120	120,58	20,51	0,58	99,52	0,48
130	130,27	18,46	0,27	99,58	0,42
140	140,72	16,38	0,72	99,11	0,89

Alınan ölçüm değerlerinde, 60–140 cm uzaklıklarda bulunan tek bir nesne için gerçekleştirilmiştir. Ölçüm değeri 60 cm altında olduğunda doğruluk değerleri ciddi oranda azalmaktadır. Bu durum, nesnenin iki kamera görüntü içerisinde tam olarak yer almamasından kaynaklanmaktadır.

Ayrıca, ortamda bulunan nesne kameralardan uzaklaştırıldığında, doğruluk değerleri düşmektedir. Referans numarası 98 olan çalışmada 64-120 cm arasındaki nesnelere ait doğruluk tablosu sunulmaktadır. Şekil 4.36’da 60 – 140 cm arasında yapılan ölçümlerin doğruluk değerleri sunulmuştur. 60 cm uzaklıkta yaklaşık % 93 doğrulukta ölçülen nesne, 140 cm uzaklıkta bir nesne yaklaşık %99 doğrulukta ölçülmektedir.



Şekil 4.36. Gerçek değerlere ait doğruluk yüzdesini gösteren grafik

Yapılan çalışmada, kullanılan ölçme tekniği, gezgin robot üzerinde uygulandığında, stereo kameraların bakış açılarına göre bazı problemler oluşmaktadır. Çalışma, kameraların karşıya doğru baktığı ve nesnelerin kameralara karşı dik olduğu varsayılarak hazırlanmıştır. Ancak, kameraların yere doğru baktığı ve nesnelerin kameraların karşısında dik olmadığı durumlarda, formülün hatalı sonuçlar verdiği gözlemlenmiştir. Bu durumda elde edilen verilerden ve ölçülen gerçek değerlerden faydalanılarak, eğri uydurma yöntemi kullanılarak sonuçlar alınmıştır. Ayrıca doğruluk tablosu yönteminden sonuçlar elde edilirken faydalanılmaktadır. Sonuçlar elde edilirken, sol ve sağ kamera görüntü çiftleri arasındaki x koordinat farkı, robotun ileriye (D_{dikey}) doğru gitmesi gereken mesafeyi verirken, sol kamera y koordinat konumu yanlara (D_{yatay}) doğru ilerlemesini ve robotun dönüş yönünü ifade etmektedir. Bu durumda, robotun, herhangi bir engele olan Manhattan uzaklıkları da bulunmaktadır. Deneysel çalışmalar sonunda elde edilmiş veriler kullanılarak oluşturulan Eşitlik (4.11) kullanılarak robotun dikeyde gideceği yol hesaplanmaktadır. Robota, Öklid uzaklığında bulunan nesneye ait uzaklık ise Eşitlik (4.12)'de sunulan denklem ile elde edilmektedir. Ayrıca robotun dönüş açısı, Eşitlik (4.13)'de görülen denklem kullanılarak hesaplanmaktadır.

$$D_{dikey} = 11264e^{-0,0508*\Delta x} \quad (4.11)$$

$$D = \sqrt{D_{\text{dikey}}^2 + D_{\text{yatay}}^2} \quad (4.12)$$

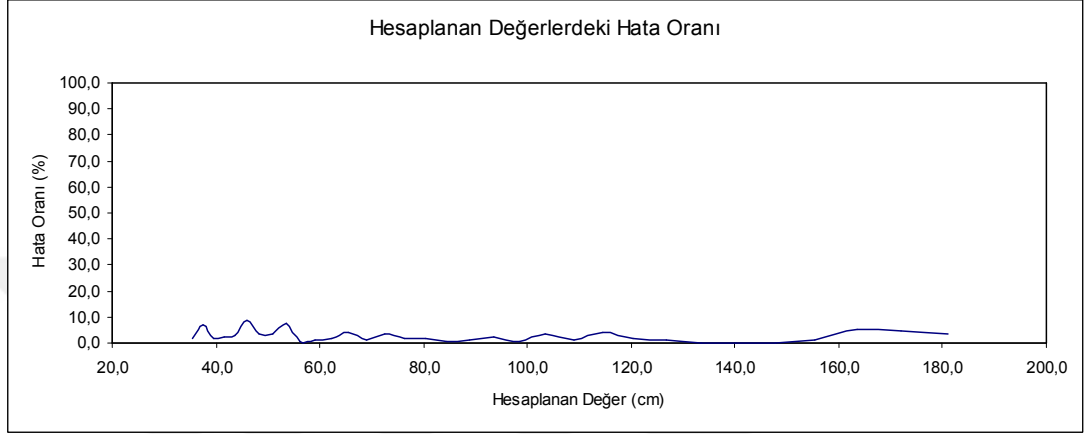
$$\theta_B = \arctan\left(\frac{D_{\text{yatay}}}{D_{\text{dikey}}}\right) \quad (4.13)$$

Tablo 4.4’de dikey yönde hareket edecek robot için hesaplanan mesafe verileri bulunmaktadır. Tabloda, gerçek değer, piksel farkları, Eşitlik (4.11) ile hesaplanan değer, fark ve hata oranları sunulmaktadır.

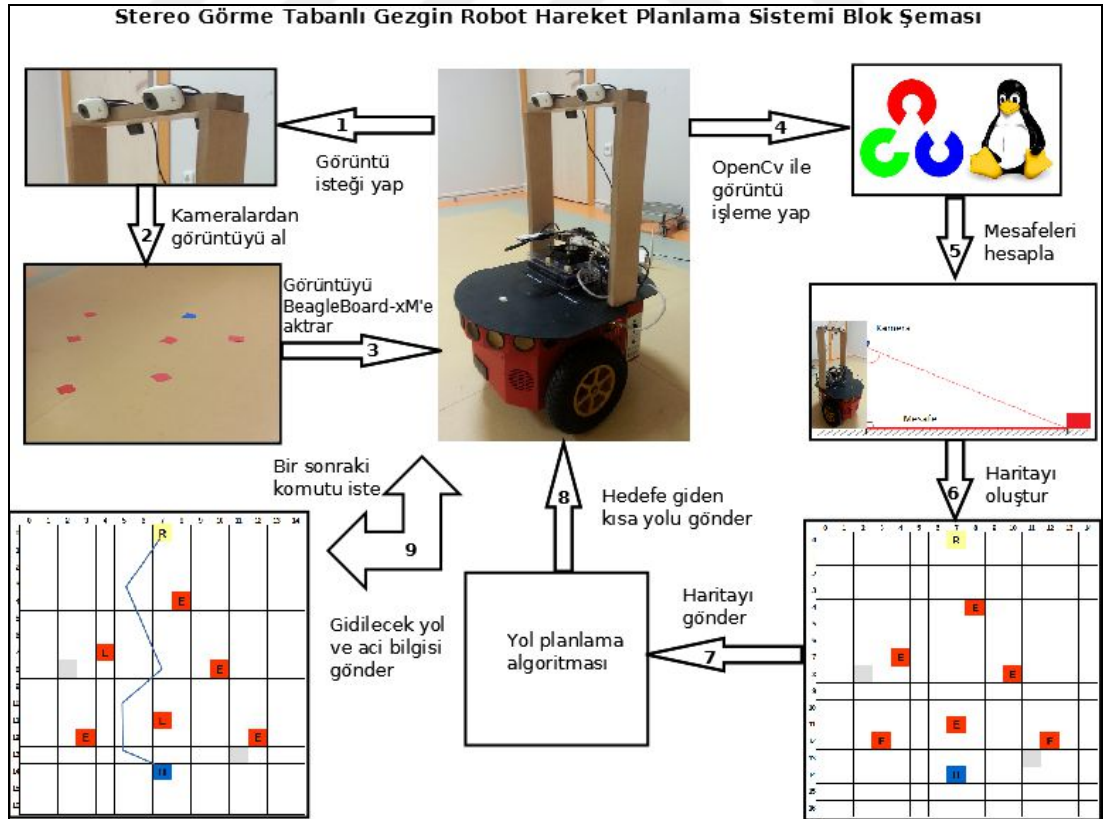
Tablo 4.4. Robotun dikey yönde nesnelere olan mesafesi

Gerçek Değer(cm)	Ölçülen Ortalama Değer (cm)	Ortalama Piksel Fark Değeri	Ölçümler arası Fark (cm)	Hata (%)
180	171,95	37	8,05	4,7
157,5	155,33	39	2,17	1,4
140	140,33	41	0,33	0,2
125	126,77	43	1,77	1,4
110	114,52	45	4,52	4,0
100	103,46	47	3,46	3,3
91,5	93,46	49	1,96	2,1
85	84,43	51	0,57	0,7
75	76,28	53	1,28	1,7
68	68,91	55	0,91	1,3
56	56,24	59	0,24	0,4
50	48,29	62	1,71	3,5
45	43,62	64	1,38	3,2
40	39,41	66	0,59	1,5

Şekil 4.37’de hesaplanan değerlerdeki hata oranlarını gösteren grafik verilmiştir. 30 ila 50 cm arasında uzaklıkta bulunan nesnelerin, kameraların görüş açısı içerisinde tam olarak yer almamasından kaynaklanan dalgalanma ve hata oranında yükselme gözlemlenmektedir. Nesnelerin kameraya olan uzaklıkları, 160 cm’yi geçtiğinde hata oranının yükseldiği gözlenmiştir.



Şekil 4.37. Hesaplanan değerlerdeki hata yüzdesini gösteren grafik



Şekil 4.38. Önerilen sistemin blok şeması

Önerilen çalışmada kullanılan stereo görme tabanlı gezgin robot hareket planlama sistemine ait blok şema Şekil 4.38’de sunulmuştur. Gezgin robot üzerinde bulunan BeagleBoard-xM, kameralara ortamın görüntüsünü almak için istekte bulunmaktadır. Kameralardan ortam görüntüsü alınarak, OpenCV kütüphanesi yardımıyla işlenmektedir. Ortamda bulunan nesnelere tespit edildikten sonra, kameralara olan mesafeler hesaplanmaktadır. Hesaplanan mesafe bilgilerine göre, nesnelere harita üzerindeki yerleri belirlenmektedir. Çalışma ortamına ait harita bilgisi GA’ya gönderilerek en kısa yol bilgisi elde edilmektedir. Elde edilen en kısa yol bilgisi kullanılarak, gezgin robotun bulunduğu noktadan, bir sonraki adımda gideceği noktaya olan mesafe ve açı bilgileri hesaplanmaktadır. Hesaplanan bilgiler robotun anlayacağı hareket komutlarına dönüştürülerek, seri arabirim üzerinden P3-DX’e gönderilmekte ve robotun hareket etmesi sağlanmaktadır. P3-DX gezgin robot her bir hareketini tamamladığında seri arabirim üzerinden BeagleBoard-xM’e SIPs paketi göndererek işlemi tamamladığını bildirmektedir. Bu işlem robot hedefe ulaşana kadar devam etmektedir.

5. SONUÇ VE ÖNERİLER

Tez çalışmasında, gezgin robotun otonom olarak çalışabilmesi amacıyla, algılayıcı olarak kameralardan yararlanılmıştır. Ayrıca, gezgin robot olarak kullanılan P3-DX'i kontrol etmek amacıyla gömülü sistem kullanılmıştır. Gömülü sistem olarak, BeagleBoard-xM TKB kullanılmıştır. Ortamda bulunan engel, hedef ve robot gibi nesnelerin tespit edilmesi amacıyla kablosuz IP kamera ve USB kameralar kullanılmaktadır. BeagleBoard-xM'in P3-DX ile beraber mobil dolaşması için 5 V, 1 A'lık PowerBank ismi verilen batarya kullanılmıştır. BeagleBoard-xM TKB üzerine diğer dağıtımlardan daha iyi performans gösteren Angström dağıtımı kurulmuş ve kullanılmıştır. Kameralardan elde edilen görüntülerin işlenmesi OpenCV adı verilen açık kaynak kodlu bilgisayarlı görme kütüphanesi kullanılarak gerçekleştirilmiştir. OpenCV gömülü sistem ve gerçek zamanlı gömülü görme uygulamalarında iyi performans vermektedir.

Tez çalışması kapsamında, genellikle açık kaynak kodlu yazılımlara ve donanımlara öncelik verilmiştir. Bu sebeple yapılan çalışmalarda, düşük maliyetli sistemler ve çözümler önerilmektedir.

Tez çalışması kapsamında yapılan ilk çalışmada, gezgin robotların hareket ve kontrolü için üç bloktan oluşan sistem önerilmiştir. GİB, ilk blok olup görüntü işleme teknikleri kullanarak ortamın haritasını çıkarmaktadır. Bu bölümde girdi olarak kameralardan alınan görüntü kullanılmaktadır. YPB, ikinci blok olup girdi olarak ortamın haritasını alarak, en kısa yolu çıktı vermektedir. Çalışmada, en kısa yolun bulunması amacıyla GA kullanılmaktadır. Yapılan deneysel çalışmalar sonucunda, GA tabanlı yol planlama algoritmasında seri ve tamsayı kodlama yöntemi kullanılmıştır. Laboratuvar çalışma ortamının küçük olmasından dolayı, GA en kısa yolu ortalama dört adımda ve hızlı bir şekilde hesaplamaktadır. Girdi olarak ortamın ızgara tabanlı haritasını alan yol planlama algoritması için farklı bir algoritma kullanılarak seri ve tamsayılardan oluşan en kısa yol çıktısı da üretilebilmektedir. En kısa yol bilgisini girdi olarak alan RHB, çıktı olarak robot dönüş açıları ve mesafe bilgilerini üretmektedir. Ayrıca, kullanılan robota göre hareket komutları üretilmektedir.

Üretilen hareket komutları, gezgin robota çeşitli ara birimler üzerinden gönderilmektedir. Deneysel çalışmada kullanılan P3-DX gezgin robotu için oluşturulan hareket komutları, seri arabirim kullanılarak gönderilmiş ve robotun hedefe doğru ilerlemesi sağlanmıştır.

Çalışmada, önerilen bloklar ve kullanılan araçlar küçük değişikliklerle farklı sistemler için uygun şekilde çalışabilmektedir. Örneğin, GIB, bloğunda renk tanıma yerine şekil tanıma veya kenar algılama kullanılarak, ortamdaki nesnelere benzer şekilde ızgara tabanlı harita içerisine yerleştirilebilmektedir. Aynı şekilde, YPB için farklı algoritmalar kullanılarak en kısa yol çıktısı üretilmektedir. Araç olarak kullanılan BeagleBoard-xM yerine teknolojinin gelişmesiyle beraber çıkan yeni TKB'lerden biri kullanılabilir. P3-DX gezgin robotunun yerine, farklı robot ve iletişim protokolleri kullanılabilir.

Gezgin robotun uzaktan kontrol ve hareket ettirilmesi amacıyla ilk çalışma geliştirilerek, web tabanlı gezgin robot hareket ve kontrol sistemi önerilmiştir. Çalışmada, laboratuvar ortamının gerçek görüntüsü üzerine, sanal engeller ve hedef yerleştirilmekte, robotun bu engellere çarpmadan, hedefe ulaşması sağlanmaktadır. Robot hareketini gerçek laboratuvar ortamında yapmaktadır. Kullanıcı, en kısa yolu bulmak için kullanılan GA'ya ait parametreleri değiştirerek aldığı sonuçları inceleyebilmektedir. Ayrıca kullanıcılar, web ara yüzü sayesinde, gezgin robotu uzaktan, hareket ettirebilmektedirler. Gezgin robotun, yönü ve başlangıç açısı robot üzerinde bulunan sarı ve kırmızı renklerin görüntü işleme teknikleri ile işlenmesi sonucunda gerçekleştirilmektedir. Başlangıç açısı, robot dönme açısı, robotun ilerleyeceği yönün bulunması hedefe ulaşılmasında çok önemlidir. Sistem başarılı bir şekilde açıları ve yönü bulmakta, aynı zamanda en kısa mesafeden robota dönüşünü yaptırmaktadır.

P3-DX gezgin robotu dönüş açısını zaten en kısa mesafeden ve en az enerji tüketen şekilde gerçekleştirmektedir. Ancak, yapılan konum belirleme ve robot açısının bulunması çalışması, kullanılacak herhangi bir robot için az enerji tüketen ve en kısa mesafeden dönecek şekilde tasarlanmıştır.

Tez çalışmasının son bölümünde, stereo görme tabanlı gezgin robot hareket sistemi önerilmiştir. Kamera kalibrasyon işleminden sonra, stereo olarak alınan görüntüler işlenmekte ve ortamda bulunan nesnelere ve koordinatları tespit edilmektedir.

Bu koordinatlara göre, robot, engeller ve hedef ızgara tabanlı haritaya yerleştirilmektedir. Aynı zamanda, her bir nesnenin kameralara yani robota olan mesafeleri hesaplanmaktadır. Mesafe hesaplama işleminde üçgen benzerlikleri sayesinde elde edilen formülden ve doğruluk tablosu (look up table) yöntemlerinden faydalanılmaktadır. Ancak, bazı durumlarda hatalı sonuçlar verdiği gözlenmektedir. İki yöntem birleştirilmiş ve eğri uydurma yöntemi ile harmanlanarak kullanılmıştır. Bu sayede nesnelere robota olan mesafeleri ve robotun dönüş açıları bulunmuş ve robotun hareketi sağlanmıştır.

GA, optimizasyon problemlerinde kullanılan bir teknik olmasına rağmen gezgin robotun yol planlama probleminde göstermiş olduğu başarı ve çözüm süresi olarak iyi sonuçlar vermiştir. Ayrıca, GA çıktı olarak, çözüm kümesi ürettiğinden alternatif iyi çözümlerde sunmaktadır. Ortam ızgara tabanlı haritasının küçük boyutlarda olmasından dolayı, dört veya beş adımda iyi çözümler ürettiği görülmüştür. Adım sayısının benzer özellikteki yol planlama problemlerinde artırmanın avantaj sağlamadığı gözlenmiştir. TKB olarak kullanılan BeagleBoard-xM üzerinde 1s altında çözümler ürettiği gözlenmiştir. Çift çekirdekli 2,66 GHz işlemciye sahip bilgisayar ortamında yapılan çalışmalarda GA ortalama 40 ms sürede çözüm üretmiştir.

Stereo görme kullanılarak nesnelere tespit edilmesi ve ortamın ızgara tabanlı haritasının çıkarılması ortalama 1 s altında gerçekleştiği gözlenmiştir. Ortamda nesne sayısının artması ve farklı renkte nesnelere eklenmesi önerilen algoritmayı çok fazla etkilememektedir.

Kamera ile aynı doğrultuda yer alan nesnelere mesafe kestiriminde 60 ila 140 cm arası değerler dikkate alınmış ve %93 ila %99 arasında doğrulukta mesafe kestirimi yapıldığı gözlenmiştir. Ayrıca, kameranın açılı olarak gezgin robotun üzerine yerleştirilmesi sonucunda 40 ila 180 cm arasında değerler için mesafe kestirimi gerçekleştirilmiş ve %95 ila %99 doğruluk oranı elde edilmiştir.

Ayrıca, otonom sistemlerin başarımlarını değerlendirmesi çalışmalarında kullanılmak için çok sayıda bileşeni içerisinde barındıran bir test düzeneği geliştirilmiştir.

Tez çalışması boyunca yapılan çalışmalar, kapalı ortamlarda gerçekleştirmiş ve başarılı sonuçlar alınmıştır. Ancak, gezgin robotların dış mekânlarda ve karmaşık ortam-

larda çalışacak şekilde yeni çalışmalar yapılması gerekmektedir. Farklı gömülü sistem mimarileri, sonar, lazer ve kamera gibi farklı algılayıcıların beraber kullanılması ile gerçek dünyadaki karmaşık problemler üzerine çalışmalar yapılabilir. Ayrıca gezgin robotların konum belirleme, engel sakınım ve yol planlama gibi problemleri için farklı algoritmalar geliştirilerek önerilen test düzeneği üzerinde gerçekleştirilebilir.



KAYNAKLAR

- [1] Yılmaz N., Sağırođlu Ő., Bayrak M., Genel Amaçlı Web Tabanlı Mobil Robot:SUNAR, *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 2006, **21**(4), 745-752.
- [2] Gürel U., Çoktörel Gezgin Robot Gruplarında Müzayede Yöntemi İle Görev Dağıtımı, Doktora Tezi, Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü, Eskişehir, 2014, 360831.
- [3] Tuncer A., Otonom Araçlar İçin Yol Bulma Probleminin Genetik Algoritmalar Ve FPGA İle Çözümü Ve Gerçekleştirilmesi, Doktora Tezi, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Kocaeli, 2013, 335451.
- [4] Arıcı V., Engellerin Bulunduđu Ortamda Gezgin Robotun En iyi Yolu Bulması ve İzlemesi, Yüksek Lisans Tezi, Başkent Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2008, 216711.
- [5] Leonard J., Durrant-Whyte H., *Directed Sonar Sensing for Mobile Robot Navigation*, Kluwert Academic Publisher, Boston, 1992.
- [6] Negenborn R., Robot Localization and Kalman Filters on Finding your position in a Noisy World, M.S. thesis, Utrecht University, 2003.
- [7] Gürel U., Gezgin Robotlar için Seyrüsefer ve Kamera Takip Sistemi, Yüksek Lisans Tezi, Eskişehir Osmangazi Üniversitesi, Fen Bilimleri Enstitüsü, Eskişehir, 2006, 184118.
- [8] Ali S. A., Ghaffari M., Liao X., Hall E., Mobile Robotics, Moving Intelligence, Editor: Burchli J., *Mobile Robotics, Moving Intelligence*, Pro Literatur Verlag, Germany/ARS, Austria, 95-116, 2006.
- [9] Uzer M., S., Yılmaz N., Bayrak M., Görme Tabanlı Mobil Robot ile Farklı Renklerde Nesnelerin Gerçek Zamanlı Takibi, *Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi*, 2010, **25**(4), 759 -766.
- [10] DeSouza, G. N., Kak, A. C., Vision for Mobile Robot Navigation: A survey, *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 2002, **24**(2), 237-267.
- [11] Martinez D., Moreno J., Tresanchez M., Teixido M., Font D., Pardo A., Marco S., Palacin J., Experimental application of an autonomous mobile robot for gas leak detection in indoor environments, *17th International Conf. on Information Fusion, FUSION 2014*, Salamanca, Spain, 7-10 July 2014.

- [12] Yalman S., Haşiloğlu A., Hastanelerde Eş-Zamanlı İlaç Dağıtımını Yapan Hemşire /Hemşir Robotunun Geliştirilmesi, *Tıp Teknolojileri Ulusal Kongresi, TIPTEKNO15*, Muğla, Türkiye, 15–18 Ekim 2015.
- [13] Özdemir D., Köse C., Gezgin Robotların Çiftliklerde Ürün Yeri Belirleme Ve Taşıma İşlemlerinde Kullanımı, *Erzincan Üniversitesi Fen Bilimleri Dergisi*, 2008, **1**(1), 49–61.
- [14] Horsman G., Unmanned aerial vehicles: A preliminary analysis of forensic challenges, *Digital Investigation*, 2016, **16**, 1-11.
- [15] Cox I. J., Blanche: Position estimation for an autonomous robot vehicle, *IEEE/RSJ International Workshop on Intelligent Robots and Systems'89*, Tsukuba, Japan, 4-9 September 1989.
- [16] Hatipoğlu K. H., Genişletilmiş Kalman Süzgeci İle Gezgin Robot Konumunun Belirlenmesi, Yüksek Lisans Tezi, Başkent Üniversitesi Fen Bilimleri Enstitüsü, Ankara, 2007, 199498.
- [17] Costa C. M., Sobreira H. M., Sousa A. J., Veiga G. M., Robust 3/6 DoF self-localization system with selective map update for mobile robot platforms, *Robotics and Autonomous Systems*, 2016, **76**, 113–140.
- [18] Caremas C. F., Serrano F. J., Moreno V., Curto B., Aragon J. F. R., Alves R., A real-time indoor localization approach integrated with a Geographic Information System (GIS), *Robotics and Autonomous Systems*, 2016, **75**, 475–489.
- [19] Shim J. H., Choa Y. I., A Shadow Removal Method for a Mobile Robot Localization using External Surveillance Cameras, *Procedia Computer Science*, 2015, **56**, 150-155.
- [20] Shim J. H., Choa Y. I., A Mobile Robot Localization using External Surveillance Cameras at Indoor, *Procedia Computer Science*, 2015, **56**, 502-507.
- [21] Gu D., Chen K. S., Design and performance evaluation of wiimote-based two-dimensional indoor localization systems for indoor mobile robot control, *Measurement*, 2015, **66**, 95–108.
- [22] Basit A., Dailey M. N., Moonrinta J., Laksanacharoen P., Joint localization and target tracking with a monocular camera, *Robotics and Autonomous Systems*, 2015, **74**, 1-14.
- [23] Chong T. J., Tang X. J., Leng C. H., Yogeswaran M., Ng O. E., Chong Y. Z., Sensor Technologies and Simultaneous Localization and Mapping (SLAM), *Procedia Computer Science*, 2015, **76**, 174 – 179.
- [24] Fidalgo E. G., Ortiz A., Vision-based topological mapping and localization methods: A survey, *Robotics and Autonomous Systems*, 2015, **64**, 1–20.

- [25] Mirkhani M., Forsati R., Shahri M., Moayedikia A., A Novel Efficient Algorithm for Mobile Robot Localization, *Robotics and Autonomous Systems*, 2013, **61**, 920–931.
- [26] Hou K., Sun Y., Jia Q., Zhang Y, An Autonomous Positioning and Navigation System for Spherical Mobile Robot, *Procedia Engineering*, 2012, **29**, 2556-2561.
- [27] Kar A., Linear-time robot localization and pose tracking using matching signatures, *Robotics and Autonomous Systems*, 2012, **60**, 296–308.
- [28] Baca B., Salvi J., Cufi X., Appearance-based mapping and localization for mobile robots using a feature stability histogram, *Robotics and Autonomous Systems*, 2011, **59**, 840–857.
- [29] Habibov A., Gezin Robotlarla Eş Zamanlı Konum Belirleme Ve Haritalama, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, 2011, 296891.
- [30] Ataş Y., Gezin Robot Sistemlerinde Konumlandırma ve Haritalama, Yüksek Lisans Tezi, TOBB Ekonomi ve Teknoloji Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2010, 289981.
- [31] Loevsky I., Shimshoni I, Reliable and efficient landmark-based localization for mobile robots, *Robotics and Autonomous Systems* , 2010, **58**, 520-528.
- [32] Mekonnen G., Kumar S., Pathak P. M., Wireless hybrid visula servoing of omnidirectional wheeled mobile robots, *Robotics and Autonomous Systems*, 2016, **75**, 450-462.
- [33] Sharma P. S., Chitaliya N. G., Obstacle Avoidance Using Stereo Vision: A Survey, *International Journal of Innovative Research in Computer and Communication Engineering*, 2015, **3**(1), 24-29.
- [34] Fava A. D., Satler M., Tripicchio P., Visual navigation of mobile robots for autonomous of indoor and outdoor areas, *23rd Mediterranean Conference on Control and Automation (MED)*, Torremolinos, Spain, 16-19 June 2015.
- [35] Al-Mutib K., Mattar E., Alsulaiman M., Implementation of Fuzzy Decision Based Mobile Robot Navigation using Stereo Vision, *Procedia Computer Science*, 2015, **62**, 143-150.
- [36] Sharifi M., Chen X., Introducing A Novel Vision Based Obstacle Avoidance Technique for Navigation of Autonomous Mobile Robots, *2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA)*, Auckland, New Zealand, 15–17 June 2015.
- [37] Cao T., Xiang Z. Y., Liu J. L., Perception in Disparity: An Efficient Navigation Framework for Autonomous Vehicles With Stereo Cameras, *IEEE Transactions On Intelligent Transportation Systems*, 2015, **16**(5), 2935-2948.

- [38] Rettkowski J., Gburek D., Göhringer D., Robot Navigation based on an Efficient Combination of an Extended A* algorithm, Bird's Eye View and Image Stitching, *IEEE 2015 Conference on Design and Architectures for Signal and Image Processing*, Krakow, Poland, 23–25 September 2015.
- [39] Charan S. G., Manjunath M., Niranjana S., Kranthi K. G. J., Nutan P. V., Monovision based Automated Navigation and Object Detection, *IEEE International Conference on Robotics, Automation, Control and Embedded Systems–RACE 2015*, Chennai, India, 18–20 Şubat 2015.
- [40] Cherubini A., Spindler F., Chaumette F., Autonomous Visual Navigation and Laser-Based Moving Obstacle Avoidance, *IEEE Transactions On Intelligent Transportation Systems*, 2014, **15**(5), 2101-2110.
- [41] Zhang Q., Chen D., Chen T., An Obstacle Avoidance Method of Soccer Robot Based on Evolutionary Artificial Potential Field, *Energy Procedia*, 2012, **16**, 1792-1798.
- [42] Kim J., Do Y., Moving obstacle avoidance of a mobile robot using a single camera, *Procedia Engineering*, 2012, **41**, 911-916.
- [43] Sezer V., Gokasan M., A novel obstacle avoidance algorithm: Follow the Gap Method, *Robotics and Autonomous Systems*, 2012, **60**, 1123–1134.
- [44] Sezer V., A Novel Obstacle Avoidance Approach For Nonholonomic Ground Vehicle Autonomy, Doktora Tezi, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2012, 323683.
- [45] İyidir İ. K., Visual Obstacle Detection And Avoidance For Indoor Mobile Robots, Yüksek Lisans Tezi, Işık Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2012, 320751.
- [46] Öztürk C. N., Vision-Based Autonomous Navigator, Yüksek Lisans Tezi, Fatih Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2012, 318519.
- [47] Ng J., An Analysis of Mobile Robot Navigation Algorithms in Unknown Environments, PhD. Thesis, The University of Western Australia, Electrical, Electronic and Computer, Australia, 2010.
- [48] Zhao Y., Cheng W., Jia L., Ma S., The Obstacle Avoidance and Navigation based on Stereo Vision for Mobile Robot, *IEEE 2010 International Conference on Optoelectronics and Image Processing*, Haiko, Hainan, Chine, 11-12 November 2010.
- [49] Das P. K., Behera H. S., Panigrahi B. K., Intelligent-based multi-robot path planning inspired by improved classical Q-learning and improved particle swarm optimization with perturbed velocity, *Engineering Science and Technology, an International Journal*, 2016, **19**, 651-669.

- [50] Contreras-Cruz M. A., Ayala Ramirez V., Hernandez-Belmonte U. H., Mobile robot path planning using artificial bee colony and evolutionary programming, *Applied Soft Computing*, 2015, **30**, 319-328.
- [51] Karami A. H., Hasanzadeh M., An adaptive genetic algorithm for robot motion planning in 2D complex environments, *Computers and Electrical Engineering*, 2015, **43**, 317–329.
- [52] Charalampous K., Kostavelis I., Gasteratos A., Thorough robot navigation based on SVM local planning, *Robotics and Autonomous Systems*, 2015, **70**, 166–180.
- [53] Deepu R., Honnaraju B., Murali S., Path Generation for Robot Navigation using a Single Camera, *Procedia Computer Science*, 2015, **46**, 1425–1432.
- [54] Chandak A., Gosavi K., Giri S., Agrawal S., Kilkarni P., Path Planning for Mobile Robot Navigation using Image Processing, *International Journal of Scientific & Engineering Research*, 2013, **4(6)**, 1490–1496.
- [55] Miao H., Tian Y. C., Dynamic robot path planning using an enhanced simulated annealing approach, *Applied Mathematics and Computation*, 2013, **222**, 420–437.
- [56] Ohnishi N., Imiya A., Appearance-based navigation and homing for autonomous mobile robot, *Image and Vision Computing*, 2013, **31(6-7)**, 511–532.
- [57] Tuncer A., Yıldırım M., Dynamic path planning of mobile robots with improved genetic algorithm, *Computers and Electrical Engineering*, 2012, **38**, 1564–1572.
- [58] Tuncer A., Yıldırım M., Erkan K., A Motion Planning System for Mobile Robots, *Advances in Electrical and Computer Engineering*, 2012, **12(1)**, 57-62.
- [59] Ardiyanto I., Miura J., Real-time navigation using randomized kinodynamic planning with arrival time field, *Robotics and Autonomous Systems*, 2012, **60(12)**, 1579–1591.
- [60] Liu C., Liu H., Yang J., A Path Planning Method Based on Adaptive Genetic Algorithm for Mobile Robot, *Journal of Information & Computational Science*, 2011, **8(5)**, 808 -814.
- [61] Tamiselvi, Shalinie M., Hariharasudan, Optimal Path Selection for Mobile Robot Navigation Using Genetic Algorithm, *IJCSI International Journal of Computer Science Issues*, 2011, **8(4)**, 433-440.
- [62] Suvaydan F., Mobil Robotlar için Yol Planlama Problemi ve Karınca Kolonisi Algoritması ile Yol Planlama Problemlerinin Optimal Çözümü, Yüksek Lisans Tezi, Düzce Üniversitesi, Fen Bilimleri Enstitüsü, 2011, 309239.

- [63] Yao Z., Ma L., A Static Environment-Based Path Planning Method by Using Genetic Algorithm, *International Conference on Computing, Control and Industrial Engineering*, Wuhan, Chine, 5–6 June 2010.
- [64] Zahoo J., Zhu L., Liu G., Liu G., Han Z., A Modified Genetic Algorithm for Global Path Planning of Searching Robot in Mine Disasters, *Proceedings of the 2009 IEEE International Conference on Mechatronics and Automation*, Changchun, China, 9–12 August 2009.
- [65] Khusheef A. S., Investigation on the Mobile Robot Navigation in an Unknown Environment, M.S. thesis, Australia Edith Cowan University, Faculty of Computing, Health and Science, School of Engineering, 2013.
- [66] Asmaz K., Görüntü İşleme İle İki Boyutlu Cisimlerden Grafik Modeller İçin Veri Eldesi, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, 2006, 182806.
- [67] Karakoç M., Görüntü İşleme Teknikleri Ve Yapay Zekâ Yöntemleri Kullanarak Görüntü İçinde Görüntü Arama, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, 2011, 287601.
- [68] Young I. T., Gerbrands J. J., Vliet L. J. V., *Fundamentals of Image Processing*, Delft University of Technology, Netherlands, 1998.
- [69] https://en.wikipedia.org/wiki/Computer_vision, (Ziyaret Tarihi: 20 Mart 2016).
- [70] <http://e-bergi.com/y/Bilgisayarli-Gormeye-Giris>, (Ziyaret Tarihi: 20 Mart 2016).
- [71] Bier J., Implementing Vision Capabilities in Embedded Systems, BDTI, http://www.bdti.com/private/pubs/BDTI_ESC_Embedded_Vision.pdf, (Ziyaret Tarihi: 20 Mart 2016).
- [72] Gregori E., Introduction to Embedded Vision and the OpenCV Library, BDTI, <http://www.embedded-vision.com>, (Ziyaret Tarihi: 20 Mart 2016).
- [73] https://en.wikipedia.org/wiki/Single-board_computer, (Ziyaret Tarihi: 25 Mart 2016).
- [74] Alee N, Rahman M., Ahmad R. B., Performance Comparison of Single Board Computer: A Case Study of Kernel on ARM Architecture, *IEEE The 6th International Conference on Computer Science & Education (ICCSE 2011)*, SuperStar Virgo, Singapore, 3–5 August 2011.
- [75] Solak S., Bolat E. D., A Real Time Industrial Application of Single Board Computer Based Color Detection System, *IEEE 8th International Conference on Electrical and Electronics Engineering (ELECO'13)*, Bursa, Turkey, 28–30 November 2013.

- [76] Gomez A., Cuinas D., Catala P., Xin L., Li W., Conway S., Lack D., Use of Single Board Computers as Smart Sensors in the Manufacturing Industry, *Procedia Engineering*, 2015, **132**, 153-159.
- [77] Yakut O., Solak S., Bolat E. D., Implementation of a Web-Based Wireless ECG Measuring and Recording System, *ICMPMS 2015 : 17th International Conference on Medical Physics and Medical Sciences*, İstanbul, Turkey, 26–27 October 2015.
- [78] Kasundra C. T., Shirsat A. S., Raspberry-Pi Based Health Monitoring System, *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 2015, **4**(8), 7147–7154.
- [79] Yakut O., Solak S., Bolat E. D., Measuring ECG Signal Using e-Health Sensor Platform, *International Conference on Chemistry, Biomedical and Environment Engineering (ICCBEE'14)*, Antalya, Turkey, 7-8 October 2014.
- [80] Gupta M. S. D., Patchava V., Menezes V., Healthcare based on IoT using Raspberry Pi, *IEEE, 2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, Noida, India, 8–10 October 2015.
- [81] Toshniwal K., Conrad J. M., A Web-based Sensor Monitoring System on a Linux-based Single Board Computer Platform, *IEEE Proceedings of the SoutheastCon 2010 (SoutheastCon)*, Concord, NC, USA, 18–21 March 2010.
- [82] More V., Kumar H., Kaingade S., Gaidhani P., Gupta N., Visual Odometry using Optic Flow for Unmanned Aerial Vehicles, *IEEE 2015 International Conference on Cognitive Computing and Information Processing (CCIP)*, Noida, India, 3–5 March 2015.
- [83] Damodaran S., Stanley P. K., Herbal leaves feature extraction using Beagle board, *IEEE 2014 International Conference on Electronics and Communication Systems (ICECS)*, Coimbatore, India, 13–14 February 2014.
- [84] Nayyar A., Puri V., A Review of Beaglebone Smart Board's- A Linux/Android Powered Low Cost Development Platform based on ARM Technology, *IEEE 2015 9th International Conference on Future Generation Communication and Networking*, Jeju Island, South Korea, 25–28 November 2015.
- [85] Beagleboard.org, BeagleBoard-xM Rev C System Reference Manuel, <http://www.beagleboard.org>, (Ziyaret Tarihi: 26 Mart 2016).
- [86] Intel, *Open Source Computer Vision Library Reference Manuel*, Intel Corporation, USA, 2001.
- [87] Bradski G., Kaehler A., *Learning OpenCV: Computer Vision with the OpenCVLibrary*, O'Reilly Media, USA, 2008.
- [88] <http://www.mobilerobots.com/Libraries/Downloads/Pioneer3DX-P3DXRevA.sflb.ashx>, (Ziyaret Tarihi: 04 Nisan 2016).

- [89] <http://www.mobilerobots.com/ResearchRobots/PioneerP3DX.aspx>, (Ziyaret Tarihi: 04 Nisan 2016).
- [90] Solak S., Bolat E. D., İnal M., Kapalı Ortamlarda Bulunan Renkli Nesnelerin Gerçek Zamanlı Tespiti, *Otomatik Kontrol Ulusal Toplantısı TOK'2015*, Denizli, Türkiye, 10–12 Eylül 2015.
- [91] Zhai S., Fidan B., Single View Depth Estimation Based Formation Control of Robotic Swarms: Fundamental Design and Analysis, *IEEE 16th Mediterranean Conference on Control and Automation*, Ajaccio, Fransa, 25–27 June 2008.
- [92] Karataş E., Tek Bakış Açısı Derinlik Kestirimi Yöntemi ile Robot Denetimi, Yüksek Lisans Tezi, TOBB Ekonomi ve Teknoloji Üniversitesi, Fen Bilimleri Enstitüsü, Ankara, 2010, 289959.
- [93] Özgündüz E, Stereo Görme ile Hareketli Görüntülerde Engellerin Uzaklık ve Boyutlarının Gerçek Zamanlı Bulunması, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2008, 237166.
- [94] Solak S., Bolat E. D., Distance estimation using stereo vision for indoor mobile robot applications, *IEEE 2015 9th International Conference on Electrical and Electronics Engineering (ELECO)*, Bursa, Turkey, 26–28 November 2015.
- [95] Li X., Visual Navigation in Unmanned Air Vehicles with Simultaneous Location and Mapping (SLAM), PhD. Thesis, Cranfield University, Cranfield Defence and Security, United Kingdom, 2014.
- [96] Akkaya A. E., Talu M. F., Nesne Modelleme: Video imgeleri Kullanılarak F-Matrisinin Hesaplanması, *DEÜ Mühendislik Fakültesi Dergisi*, 2014, **16**(48), 9–20.
- [97] Alam M. S., Islam M. N., Bal A., Karim M. A., Hyperspectral target detection using Gaussian filter and post-processing, *Optics and Lasers in Engineering*, 2008, **46**, 817– 822.
- [98] Lai X., Wang H., Xu Y., A Real-time Range Finding System with Binocular Stereo Vision, *International Journal of Advanced Robotic Systems*, 2012, **9**(27), 1-9.
- [99] https://tr.wikipedia.org/wiki/HSV_renk_uzayı, (Ziyaret Tarihi: 25 Nisan 2016).
- [100] Karhan M., İmge İşleme Yöntemleri İle Kayıslarda Yaprak Delen Hastalığı Sonucu Oluşan Lekelerin Tespiti, Yüksek Lisans Tezi, Namık Kemal Üniversitesi, Fen Bilimleri Enstitüsü, Tekirdağ, 2011, 305259.
- [101] Khan A. A., Üniversite Kampüsündeki Araç Plaka Tanıma ve Takibi için Matematik ve Algoritmik İlkeler, Yüksek Lisans Tezi, Kahramanmaraş Sütçü İmam Üniversitesi, Fen Bilimleri Enstitüsü, Kahramanmaraş, 2013, 338760.

- [102] Gümüřçü A., Video Anomaly Detection Using Morphological Processes, Yüksek Lisans Tezi, Gaziantep Üniversitesi , Fen Bilimleri Enstitüsü, 2012, 318458.
- [103] Le T. H. N., Bui T. D., Suen C. Y., Ternary Entropy-based Binarization of Degraded Document Images Using Morphological Operators, *International Conference on Document Analysis and Recognition*, Beijing, China, 2011.
- [104] Otsu N., A Threshold Selection Method from Gray-Level Histograms, *IEEE Transactions on Systems, Man, and Cybernetics*, 1979, **9**(1), 62-66.
- [105] https://en.wikipedia.org/wiki/Otsu%27s_method, (Ziyaret Tarihi: 1 Mayıs 2016).
- [106] Papakostas G. A., Koulouriotis D. E., Karakasis E. G., Tourassis V. D., Moment-based local binary patterns: A novel descriptor for invariant pattern recognition applications, *Neurocomputing*, 2013, **99**, 358–371.
- [107] Mercimek M., Gülez K., Mumcu T. V., Real Object Recognition Using Moment Invariants, *Sadhana*, 2005, **30**(6), 765–775.
- [108] Köker R., Öz C., Ekiz H., Zengin A., Bilgisayar Görmesine Dayalı Yapay Sinir Ağı Tabanlı Bir Gerçek Zamanlı Cisim Sınıflandırma Sistemi Tasarımı ve Uygulaması, Elektrik – *IEEE-Elektronik - Bilgisayar Mühendisliği Sempozyumu ve Fuarı - ELECO'2002*, Bursa, Türkiye, 18-22 Aralık 2002.
- [109] Holland J. H., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Michigan, 1975.
- [110] https://tr.wikipedia.org/wiki/Genetik_algoritma, (Ziyaret Tarihi: 3 Mayıs 2016).
- [111] Yıldırım M., Genetik algoritmalar ve benzetilmiş tavlama ile uzun dönem üretim genişletme planlaması, Doktora Tezi, Kocaeli Üniversitesi Fen Bilimleri Enstitüsü, 2003, 135971.
- [112] Emel G. G., Taşkın Ç., Genetik Algoritmalar ve Uygulama Alanları, *Uludağ Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi*, 2002, **21**(1), 129-152.

KİŞİSEL YAYIN VE ESERLER

- [1] **Solak S.**, Bolat E. D., Distance estimation using stereo vision for indoor mobile robot applications, *IEEE 2015 9th International Conference on Electrical and Electronics Engineering (ELECO'15)*, Bursa, Turkey, 26–28 November 2015.
- [2] Yakut O., **Solak S.**, Bolat E. D., Implementation of a Web-Based Wireless ECG Measuring and Recording System, *ICMPMS 2015: 17th International Conference on Medical Physics and Medical Sciences*, İstanbul, Turkey, 26 – 27 October 2015.
- [3] **Solak S.**, Bolat E. D., İnal M., Kapalı Ortamlarda Bulunan Renkli Nesnelerin Gerçek Zamanlı Tespiti, *Otomatik Kontrol Ulusal Toplantısı TOK'2015*, Denizli, Türkiye, 10–12 Eylül 2015.
- [4] Yakut O., **Solak S.**, Bolat E. D., Measuring ECG Signal Using e-Health Sensor Platform, *International Conference on Chemistry, Biomedical and Environment Engineering (ICCBEE'14)*, Antalya, Turkey, 7-8 October 2014.
- [5] **Solak S.**, Bolat E. D., A Real Time Industrial Application of Single Board Computer Based Color Detection System, *IEEE 2013 8th International Conference on Electrical and Electronics Engineering (ELECO'13)*, Bursa, Turkey, 28–30 November 2013.

ÖZGEÇMİŞ

1978 yılında İzmit'te doğdu. İlk, orta ve lise öğrenimlerini Kocaeli'de tamamladı. 2002 yılında Kocaeli Üniversitesi Mühendislik Fakültesi Bilgisayar Bilimleri Mühendisliğinden mezun oldu. 2008 yılında, Kocaeli Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği anabilim dalında yüksek lisans eğitimini tamamlayarak mezun oldu. 2002 yılından beri Kocaeli Üniversitesi Enformatik Bölümünde öğretim görevlisi olarak çalışmaktadır. 2011 yılında Kocaeli Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Bilgisayar Eğitimi anabilim dalında doktora eğitimine başladı. Evli ve bir çocuk babasıdır.

