

**T.C.  
PAMUKKALE ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM  
DALI**

**ÖZDEŞ OLMAYAN BİLEŞENLERDEN OLUŞAN  
MULTİKOPTER YAPILARI VE ÇOKLU YAPI  
UYGULAMALARI: SÜRÜ MODELLERİ TASARIMI**

**DOKTORA TEZİ**

**AHMET ÇAĞDAŞ SEÇKİN**

**DENİZLİ, TEMMUZ - 2017**

**T.C.  
PAMUKKALE ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ  
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM  
DALI**



**ÖZDEŞ OLMAYAN BİLEŞENLERDEN OLUŞAN  
MULTİKOPTER YAPILARI VE ÇOKLU YAPI  
UYGULAMALARI: SÜRÜ MODELLERİ TASARIMI**

**DOKTORA TEZİ**

**AHMET ÇAĞDAŞ SEÇKİN**

**DENİZLİ, TEMMUZ - 2017**

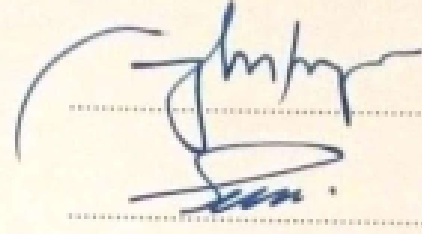
## KABUL VE ONAY SAYFASI

Ahmet Çağdaş SEÇKİN tarafından hazırlanan "ÖZDEŞ OLMAYAN BİLEŞENLERDEN OLUŞAN MULTİKOPTER YAPILARI VE ÇOKLU YAPI UYGULAMALARI: SÜRÜ MODELLERİ TASARIMI" adlı tez çalışmasının savunma sınavı 14.07.2017 tarihinde yapılmış olup aşağıda verilen jüri tarafından oy birliği / ~~oy çokluğu~~ ile Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı Doktora Tezi olarak kabul edilmiştir.

### Jüri Üyeleri

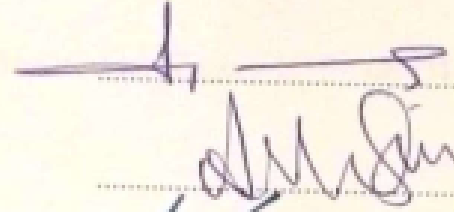
### İmza

Danışman  
Prof. Dr. Ceyhun KARPUZ



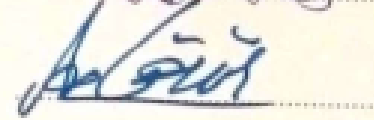
Üye  
Prof. Dr. Sezai TOKAT  
Pamukkale Üniversitesi

Üye  
Doç. Dr. Selçuk HELHEL  
Akdeniz Üniversitesi

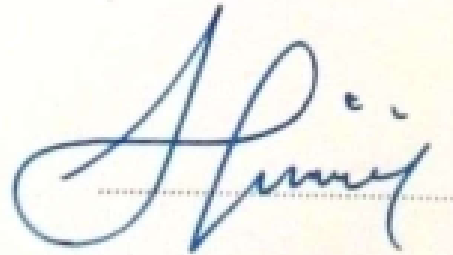


Üye  
Yrd. Doç. Dr. Ahmet ÖZEK  
Pamukkale Üniversitesi

Üye  
Yrd. Doç. Dr. Ali Kürşad GÖRÜR  
Nevşehir Hacı Bektaş Veli Üniversitesi



Pamukkale Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun  
09/07/2017 tarih ve ...34/15... sayılı kararıyla onaylanmıştır.



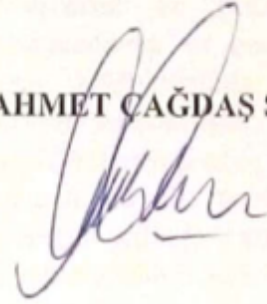
Prof. Dr. Uğur YÜCEL

Fen Bilimleri Enstitüsü Müdürü

**Bu tez çalışması Pamukkale Üniversitesi tarafından 2015 FBE031 nolu proje ile desteklenmiştir.**

**Bu tezin tasarımı, hazırlanması, yürütülmesi, arařtırmalarının yapılması ve bulgularının analizlerinde bilimsel etięe ve akademik kurallara özenle riayet edildiđini; bu alıřmanın dođrudan birincil ürünü olmayan bulguların, verilerin ve materyallerin bilimsel etięe uygun olarak kaynak gösterildiđini ve alıntı yapılan alıřmalara atfedildiđine beyan ederim.**

**AHMET AĐDAŐ SEKİN**



## ÖZET

**ÖZDEŞ OLMAYAN BİLEŞENLERDEN OLUŞAN MULTİKOPTER  
YAPILARI VE ÇOKLU YAPI UYGULAMALARI: SÜRÜ MODELLERİ  
TASARIMI  
DOKTORA TEZİ  
AHMET ÇAĞDAŞ SEÇKİN  
PAMUKKALE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ  
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI  
(TEZ DANIŞMANI: PROF. DR. CEYHUN KARPUZ)  
DENİZLİ, TEMMUZ - 2017**

Bu tezde, özdeş olmayan multikopter tiplerini ve çoklu robot uygulamalarını çalıştırabilecek uçuş denetim, konumlandırma ve çoklu robot denetim sistemleri tasarlanması amaçlanmıştır. Amaç doğrultusunda öncelikle otonom olarak çalışabilecek uçan robot sistemi tasarımı sunulmuştur. Robotun uçuş denetiminde irtifa ve durum denetimi gerçekleştirilmiş olup uçuş için PID denetim kullanılmıştır. Tasarlanan sistem dört rotorlu ve altı rotorlu hava araçlarının uçuş denetimi için denenmiştir. Bu uygulamanın ardından, çoklu robot uygulamalarında önemli olan konumlandırma problemine görüntü işleme tabanlı çözümler üretilmiştir. Konumlandırma için görsel eşleme ile robotlar arası konumlandırma ve tek gözlü etkin poz takibi yöntemleri oluşturulmuştur. Bu yöntemlerden görsel eşleme tabanlı konumlandırmada robotlar aşağı yönde bakan kameralardan aldıkları eşsiz görüntüleri birbirleriyle paylaşmakta ve dağıtık olarak konum tahmini yapmaktadırlar. Tek gözlü etkin poz takibinde ise robotların çalışma esnasında ön tarafında bulunan kameradan aldıkları görüntüler ile her robot kendi konum değişimini hesaplamaktadır. Son olarak, robotların konum bilgileri ve bulanık mantık kullanılarak akın etme ve düzen alma için bir model önerilmiştir. Bu modelde robotlar düzen deseninin tanımlı olduğu bölgeye kadar Boids algoritmasından türetilen bulanık akın algoritmasını kullanarak ulaşmakta ve sonrasında geometrik deseni oluşturan belirli hedef noktalara gitme eylemini gerçekleştirmektedir. Tez sonucunda tasarlanan denetim sistemi ile farklı tipte hava araçlarının uçuş denetimi gerçekleştirilmiştir. Önerilen konumlandırma yöntemleri benzetimler ve gerçek görüntüler üzerinde çalıştırılarak uygulanabilirlikleri ortaya konmuştur. Birden çok robotun verilen görevi bireysel veya işbirlikli olarak yerine getirmesi incelenmiş ve geliştirilen bulanık akın algoritmasının sağladığı avantaj ve dezavantajlar ortaya konmuştur.

**ANAHTAR KELİMELEER: Multikopter, Çoklu robotik sistem, Sürü davranışı, Görüntü işleme**

## **ABSTRACT**

### **MULTICOPTER STRUCTURES WITH NON-IDENTICAL COMPONENTS AND MULTIPLE ARCHITECTURE APPLICATIONS: SWARM MODELS**

**PH.D THESIS**

**AHMET AĐDAŐ SEKİN**

**PAMUKKALE UNIVERSITY INSTITUTE OF SCIENCE  
ELECTRICAL AND ELECTRONICS ENGINEERING**

**(SUPERVISOR:PROF. DR. CEYHUN KARPUZ)**

**DENİZLİ, JULY 2017**

In this thesis, it is aimed to design flight control, positioning and multi-robot control systems which can operate non-identical multicopter types and multi-robot applications. For the purpose, firstly, a flying robot system design which can work autonomously was presented. PID control was used for the flight. The designed system has been tested for flight control of quadcopter and hexacopter. Subsequently, image processing based solutions have been produced for positioning problems which are important in multi-robot applications. Inter-robot positioning with feature matching based model matching and monocular odometry methods were created. From these methods, in the feature matching based positioning, the robots share and match the unique images they receive from down facing cameras and distribute for the position estimation. In monocular odometry mode, each robot calculates its own position change with the images taken from the camera located on the front side of the robot during operation. Finally, a model for flocking and formation using robots' position information and fuzzy logic has been proposed. In this model, the robots arrive at the target area using the fuzzy flocking algorithm derived from the Boids algorithm until the region where the pattern is defined, and then perform a specific target point action that creates the geometric pattern. With the control system designed as a result of the thesis, flight control of different types of aircraft was realized. The proposed positioning methods are simulated and run on actual images and demonstrated their applicability. Multiple robots have been investigated individually or collaboratively to perform tasks and the advantages and disadvantages of the developed fuzzy flocking algorithm have been revealed.

**KEYWORDS: Multicopter, Multi-robot system, Swarm behavior, Image processing**

# İÇİNDEKİLER

Sayfa

<b>ÖZET</b> .....	<b>i</b>
<b>ABSTRACT</b> .....	<b>ii</b>
<b>İÇİNDEKİLER</b> .....	<b>iii</b>
<b>ŞEKİL LİSTESİ</b> .....	<b>v</b>
<b>TABLO LİSTESİ</b> .....	<b>vii</b>
<b>SEMBOL LİSTESİ</b> .....	<b>viii</b>
<b>ÖNSÖZ</b> .....	<b>ix</b>
<b>1. GİRİŞ</b> .....	<b>1</b>
1.1 Tezin Amacı .....	1
1.2 Bilimsel Yazın ve Temel Kavramlar .....	2
1.2.1 Çoklu Robot Kavramı ve Diğer Çoklu Sistemler .....	2
1.2.2 Biyolojik Sürü Davranışları ve Uygulamaları .....	4
1.2.2.1 Yiyecek Arama ve Kapsama .....	4
1.2.2.2 Akın Etme ve Düzen Alma .....	5
1.2.2.3 Nesnelere Etki Etme .....	7
1.2.2.4 Çoklu Hedef Gözleme ve Takip .....	7
1.2.2.5 Sürü Trafığı ve Yol Planlama .....	7
1.2.3 Çoklu Robotik Sistemlerde Robotlar Arası İşbirliği .....	8
1.2.3.1 Çoklu Robot Sistemlerinde Haberleşme .....	8
1.2.3.2 Çoklu Robotikte Algılama .....	9
1.2.3.2.1 İki Boyutlu Ortamda Hareketler ve Algılama .....	11
1.2.3.2.1 Üç Boyutlu Ortamda Hareketler ve Algılama .....	12
1.2.4 Multikopterler ve Temel Kavramlar .....	15
1.2.4.1 Temel Hareketler .....	16
1.2.4.2 Dört Rotorlu Hava Araç Modeli .....	18
1.2.4.3 Dört Rotorlu Hava Araçlarında Denetim Çalışmaları .....	20
1.3 Problem ve Bilimsel Yazına Katkı .....	21
<b>2. MATERYAL VE YÖNTEM</b> .....	<b>23</b>
2.1 Denetim Kartı Tasarımı .....	23
2.2 Yazılım Tasarımı .....	28
2.2.1 Uçuş Denetimi .....	28
2.2.1.1 Durum Denetimi .....	29
2.2.1.2 İrtifa Denetimi .....	31
2.2.2 Konumlandırma .....	32
2.2.2.1 GPS Tabanlı Konumlandırma Sistemi .....	32
2.2.2.2 Görsel Eşleme Tabanlı Konumlandırma Sistemi .....	33
2.2.2.2.1 Görsel Öznitelik Eşleme Algoritması .....	34
2.2.2.3 Tek Gözlü Etkin Poz Takibi Sistemi .....	40
2.2.3 Çoklu Robot Denetim Sistemi .....	41
2.2.3.1 Haberleşme ve Komut Denetimi .....	42
2.2.3.2 Desen Düzeni Alma ve Akın Denetimi .....	44
2.2.3.2.1 Akın Etme Denetimi ve Bulanık Mantık Yaklaşımı .....	47
<b>3. UYGULAMALAR VE BULGULAR</b> .....	<b>51</b>
3.1 Uçuş Denetim Uygulamaları .....	51
3.2 Konumlandırma Uygulamaları .....	53



3.2.1	Öznitelik Eşleme Tabanlı Konumlandırma .....	53
3.2.2	Tek Gözlü Etkin Poz Takibi Tabanlı Konumlandırma Uygulaması .....	61
3.3	Çoklu Robot Denetim Uygulamaları.....	63
<b>4.</b>	<b>SONUÇ VE ÖNERİLER.....</b>	<b>68</b>
<b>5.</b>	<b>KAYNAKLAR.....</b>	<b>70</b>
<b>6.</b>	<b>ÖZGEÇMİŞ.....</b>	<b>79</b>

## ŞEKİL LİSTESİ

### Sayfa

Şekil 1.1: Boids Algoritması Bölgeleri .....	6
Şekil 1.2: I-SWARM.....	12
Şekil 1.3: Kilobot .....	12
Şekil 1.4: Dört Rotorlu Uçan Robot Sürüsü (GRASP).....	13
Şekil 1.5: Swarmanoid (Soldan sağa el-bot, ayak-bot ve göz-bot).....	14
Şekil 1.6: Multikopter Tipleri .....	16
Şekil 1.7: Temel Hareketler: a) İrtifa b) Yunuslama c) Yuvarlanma d) Sapma ...	17
Şekil 1.8: Dört Rotorlu Hava Aracı Çerçevesi.....	19
Şekil 1.9: Geleneksel PID denetimi .....	21
Şekil 1.10: Hava Aracı Denetimi .....	21
Şekil 2.1: Otonom Hava Aracı Denetim Sistem Tasarımı .....	24
Şekil 2.2: Hava Araçlarının Montajlı Halleri: a) Dört Rotorlu b) Altı Rotorlu ..	26
Şekil 2.3: Denetim Kartı Baskı Devre Çizimi.....	26
Şekil 2.4: Denetim Kartı Şematik Çizimi .....	27
Şekil 2.5: Genel Algoritma .....	28
Şekil 2.6: Uçuş Denetimi .....	29
Şekil 2.7: Durum Denetim Algoritması .....	30
Şekil 2.8: İrtifa Denetim Algoritması .....	31
Şekil 2.9: Hava Araçları Arası Durumlar a) Çeviri b) Ölçekleme c) Dönme ve d) Birleşim .....	34
Şekil 2.10: Görsel Konumlandırma Algoritması .....	36
Şekil 2.11: Eşleme Tablosu Oluşturma Algoritması.....	38
Şekil 2.12: Eşleme Tablosu Örneği.....	39
Şekil 2.13: Tek Gözlü Görsel Etkin Poz Takibi Algoritması.....	41
Şekil 2.14: Çoklu Robot Denetim Algoritması .....	42
Şekil 2.15: Robot Haberleşmesi ve KSA .....	43
Şekil 2.16: Komut Mesajı Örneği .....	44
Şekil 2.17: Robotlar Arası Mesaj Örneği .....	44
Şekil 2.18: Desen Düzeni Alma ve Akın Etme Algoritması.....	45
Şekil 2.19: Türetilmiş Boids Algoritması .....	48
Şekil 2.20: Bulanık Akın Etme Algoritması .....	49
Şekil 2.21: Giriş Üyelik Fonksiyonu.....	50
Şekil 2.22: Çıkış Üyelik Fonksiyonu .....	50
Şekil 3.1: Altı Rotorlu Hava Aracı Uçuş Görüntüleri.....	51
Şekil 3.2: İrtifa Denetim Ölçümü.....	52
Şekil 3.3: Yuvarlanma Denetim Ölçümü.....	52
Şekil 3.4: Yunuslama Denetim Ölçümü.....	53
Şekil 3.5: Çeviri Durum Hesaplamaları .....	55
Şekil 3.6: Ölçekleme Durumu Hesaplamaları.....	56
Şekil 3.7: Dönme Durumu Hesaplamaları .....	57
Şekil 3.8: Birleşim Durumu Hesaplamaları .....	59
Şekil 3.9: Tek Gözlü Görsel Etkin Poz Hesaplama Sonucu .....	61
Şekil 3.10: Görsel Etkin Poz Hesaplama Uygulama Ara Sonuçları .....	62
Şekil 3.11: V-REP Benzetim Ortamı .....	63
Şekil 3.12: Bireysel Robotların Yarışarak Hedeflere Ulaşması.....	64

Şekil 3.13: Bireysel Robotların Önceden Atanmış Hedeflere Ulaşması.....	64
Şekil 3.14: İşbirlikli Robotların Yarışarak Hedeflere Ulaşması .....	65
Şekil 3.15: İşbirlikli Robotların Önceden Atanmış Hedeflere Ulaşması .....	65
Şekil 3.16: Bulanık İşbirlikli Robotların Yarışarak Hedeflere Ulaşması.....	66
Şekil 3.17: Bulanık İşbirlikli Robotların Yarışarak Hedeflere Ulaşması.....	66

## TABLO LİSTESİ

### Sayfa

Tablo 2.1: Malzeme Listesi.....	25
Tablo 2.2: Öznitelik Algılama ve Tanımlama Algoritmaları.....	35
Tablo 2.3: Düzen Kodları.....	44
Tablo 2.4: Geometrik Desen Üretici Fonksiyonları .....	46
Tablo 2.5: Bulanık Akın Etme Kural Tablosu .....	47
Tablo 3.1: Durum Karşılaştırması.....	54
Tablo 3.2: Çeviri Duru Başarım Tablosu.....	55
Tablo 3.3: Ölçekleme Başarım Tablosu.....	56
Tablo 3.4: Dönme Performans .....	58
Tablo 3.5: Birleşim Durumu Performans Tablosu.....	59
Tablo 3.6: Performans Karşılaştırma Tablosu .....	60
Tablo 3.7: Düzen Alma Tamamlama Tekrar ve Süreleri.....	67

## SEMBOL LİSTESİ

<b>İHA</b>	:	İnsansız Hava Aracı
<b>GPS</b>	:	Küresel Konumlandırma Sistemi (Global Positioning System)
<b>RF</b>	:	Radyo Frekans
<b>PID</b>	:	Oransal İntegral Türev (Proportional Integral Derivative) Denetim
<b>KSA</b>	:	Kablosuz Sensör Ağı
<b>UART</b>	:	Evensel asenkron alıcı verici (Universal Asynchronous Receiver/Transmitter)
<b>I2C</b>	:	Entegreler Arası Devre (Inter-Integrated Circuit) Protokolü
<b>DAC</b>	:	Sayısal analog çevirici (Digital Analog Converter)
<b>ESC</b>	:	Elektronik Hız Denetleyicisi (Electronic Speed Controller)
<b>BLDC</b>	:	Fırçasız Doğru Akım Motoru (Brushless DC motor)
<b>PWM</b>	:	Darbe Genişlik Modülasyonu (Pulse Width Modulation)
<b>VSLAM</b>	:	Görsel Eşzamanlı Yer Belirleme ve Haritalama (Visual Simultaneous Localization and Mapping)
<b>RANSAC</b>	:	Rastgele Örnekleme Fikir Birliği (Random Sample Consensus)

## ÖNSÖZ

Öncelikle gerek doktora öncesinde gerekse doktora çalışmam esnasında beni yönlendiren ve destekleyen danışmanım Prof. Dr. Ceyhun Karpuz ve Yrd. Doç. Dr. Ahmet Özek'e minnettar olduğumu belirtmek isterim. Bu günlere gelmemde en büyük pay sahibi olan başta annem ve babam olmak üzere tüm aile büyüklerime, desteğini ve sevgisini sürekli hissettiğim sevgili eşim Mine Seçkin'e ve huzur ve mutluluk kaynağım oğlum Oğuz Alp'e şükranlarımı sunarım. Ayrıca manevi desteklerinden ötürü çalışma arkadaşlarım Doç. Dr. Adem Sezer, Yrd. Doç. Dr. Yusuf İnel, Yrd. Doç. Dr. Mehmet Deniz, Arş. Gör. Ufuk Uluçınar, Öğr. Gör. S. Anıl Yücesoy ve Öğr. Gör. Fatih Bozkurt'a teşekkür ederim.

# 1. GİRİŞ

## 1.1 Tezin Amacı

Bu tezde, özdeş olmayan multikopter tiplerini ve çoklu robot uygulamalarını çalıştırabilecek uçuş denetim, konumlandırma ve çoklu robot denetim sistemleri tasarımı amaçlanmıştır. Amaç doğrultusunda öncelikle otonom olarak çalışabilecek uçan robot sistemi tasarımı sunulmuştur. Tasarlanacak elektronik denetim kartı ile görüntü işleyebilen, kablosuz sensör ağları ile haberleşebilen, farklı tipteki multikopterlerde uçuş denetimi gerçekleştirebilen bir sistem olması ve mevcut sistemlere yerli bir alternatif sunulması hedeflenmiştir. Tasarlanan sistem üzerinde 3 boyutlu çoklu robot uygulamalarında çalıştırılmak üzere görüntü işleme tabanlı konumlandırma ve etkin poz takibi yöntemleri geliştirilmesi hedeflenmiştir. Konumlandırma için görsel eşleme ile robotlar arası konumlandırma ve tek gözlü etkin poz takibi yöntemleri oluşturulmuştur. Bu yöntemlerden görsel eşleme tabanlı konumlandırmada robotlar aşağı yönde bakan kameralardan aldıkları eşsiz görüntüleri birbirleriyle paylaşmakta ve dağıtık olarak konum tahmini yapmaktadırlar. Tek gözlü etkin poz takibinde ise robotların çalışma esnasında ön tarafında bulunan kameradan aldıkları görüntüler ile her robot kendi konum değişimini hesaplamaktadır. Son olarak, robotların konum bilgileri ve bulanık mantık kullanılarak akın etme ve düzen alma için bir model önerilmiştir. Bu modelde robotlar düzen deseninin tanımlı olduğu bölgeye kadar Boids algoritmasından türetilen bulanık akın algoritmasını kullanarak ulaşmakta ve sonrasında geometrik deseni oluşturan belirli hedef noktalara gitme eylemini gerçekleştirmektedir.

Tez çalışması dört ana kısımdan meydana gelmektedir. İlk kısımda bilimsel yazın ve temel tanımlar verilmiştir. Bu tezin konusu olan çoklu uçan robot sisteminde uçan robot olarak multikopter tipi araçlar tercih edilmiştir, bu tercihin sebebi ise multikopterlerin manevra kabiliyetlerinin yüksek, üretimlerinin kolay ve mekanizmalarının nispeten ucuz olmasıdır. Bu sebepten ilk kısımda sırasıyla çoklu robot sistemleri, çoklu robot sistemlerinde kullanılan yöntemler ve multikopterlerin

çalışma prensipleri hakkında bilgiler sunulmuştur. İkinci kısımda tez kapsamında tasarlanan uçuş denetim sistemi, konumlandırma ve çoklu robot denetimi için kullanılan materyal ve yöntemler sunulmuştur. Üçüncü kısımda ise tasarlanan sistemlerle ilgili saha ve benzetim uygulamaları sonucu elde edilen bulgular sunulmuştur. Son olarak ise elde edilen bulgular hakkında sonuç ve öneriler verilmiştir.

## **1.2 Bilimsel Yazın ve Temel Kavramlar**

### **1.2.1 Çoklu Robot Kavramı ve Diğer Çoklu Sistemler**

Günümüzde robotlar fabrika uygulamaları haricinde askeri, sağlık, güvenlik vb. birçok alanda yoğun şekilde faaliyet göstermeye başlamıştır. Fabrika uygulamaları haricindeki uygulamaların en büyük farklılığı bu robotların daha karmaşık işlemlerde ve daha zorlu çevre koşullarında çalışmaya zorlamasıdır. Dolayısıyla bu robotlar daha karmaşık ve bazen de hantal hale gelmektedir. Karmaşık ve hantal sistemlerde güvenilirlik azalmakta ve maliyetler büyük oranda artmaktadır.

Elektronik ve robotik sistemlerde görev yükünün artmasına bağlı olarak karmaşıklaşma ve hantallaşma problemlerinin ortaya çıkmasıyla iş yükünün dağıtılması ve riskin azaltılması için çoklu sistemler geliştirilmeye başlanmıştır. Robotikte ise canlı sürülerinin (karınca, termit, arı, kuş, bakteri vb.) yapısından ve davranışlarından ilham alınarak çoklu robotik sistemleri kavramı 1980'lerin başında ortaya çıkmıştır (Parker 2008). Temel olarak çoklu robot sistemleri tasarımında tek bir robotun yapacağı görevi birden çok ve daha basit yapıdaki robotlara paylaşarak amaçlanmaktadır. Bu sayede istenen işi yapabilecek yeteneğe sahip bir grup robot işe sürülmekte ve gruptaki bir veya birkaç robotun devre dışı kalması durumunda dahi sistem sağlamlığını koruyup işlevini yerine getirmeye devam edebilmektedir. Çoklu robot sistemlerinde robotlar birbirlerinin yerlerini alabildiği durumlarda gürbüzlük (robustness), esneklik (flexibility) ve ölçeklenebilirlik (scalability) avantajlarını sağlamaktadır. Bu avantajlar Bayindir ve Şahin tarafından sürü robotlar için aşağıdaki şekilde açıklanmıştır (Bayindir ve diğ. 2007):



- Esneklik: Sistemin yeni, farklı veya deęişen çevre koşullarına uyum sağlayabilmesidir.
- Gürbüzlük: Sistemin kısmi arıza veya anormal durumlarda görevini veya işlevini sürdürebilmesidir.
- Ölçeklenebilirlik: Sistemin kendi kendine organize olabilen bir mekanizmadaki birey/bileşen sayısının artırılmasının veya azaltılmasının sistemin performansını önemli ölçüde etkilememe yeteneğidir.

Çoklu robot sistemleri kendi içinde homojen ve heterojen olmak üzere ikiye ayrılmaktadır. Homojen sistemleri oluşturan her robot aynı tip ve aynı görev tanımına sahiptir. Homojen yapıda olan sistemler sürü robotik veya kolektif (toplu/ortaklaşa) robotik olarak da anılmaktadır. Sürü robotlar, çok sayıda aynı ve basit robottan oluşan çoklu robot sistemlerinin koordinasyonu için yeni bir yaklaşım olarak tanımlanmıştır (Kube ve dię. 1993, Tan ve dię. 2013). Kolektif robotik olarak adlandırılmasının sebebi ise robotlara basit yerel kontrol kurallarıyla takım ölçeğinde ve küresel ölçekte ortak davranış kazandırılmasıdır. Heterojen sistemlerde ise böyle bir zorunluluk yoktur yani bireylerin görev tanımı veya birey tipi farklı olabilir. Heterojen sistemler, homojen sistemlere göre daha karmaşıktır çünkü bu sistemleri oluşturan robotlar arasında görev paylaşımı, yeteneęe/kapasiteye göre iş atama gibi problemler ortaya çıkmaktadır. Karmaşıklık sebebiyle heterojen sistemlerde yapılan çalışmalarda daha az bireyle çalışmalar yürütüldüğü görülmüştür. Hatta bazı kaynaklarda heterojen sistemler farklı tip homojen sistemlerin bir arada çalışması olarak tanımlanmıştır (Parker 2008, Yan ve dię. 2013).

Çoklu robot sistemleri çoęu zaman birlikte kullanıldığı veya ilişkili olduęu çoklu ajan ve dağıtık/paralel hesaplama sistemleri ile karşılaştırılmaktadır. Temel olarak çoklu robot sistemleri gerçek ortamda, fiziksel olarak çalışan gerçek birçok robottan oluşan bir sistemdir. Çoklu robot sistemleri konusundaki çalışmalarda aynı (homojen) veya farklı (heterojen) tipte fiziksel robotların birlikte çalışması konu edilmektedir.

Günümüzde robotlar sadece gerçek ortamlarda deęil aynı zamanda sanal ortamda da olabilmektedir. Ajan tabanlı sistemler herhangi bir sistemin eylem, davranış veya etkileşimlerinin benzetimini yapmak için geliştirilmiş yazılımsal modellerdir. Çoklu ajanlar ise bu türde birden fazla ajanın bir araya gelmesiyle

oluşmaktadır. Bu ajanlardan her biri bir görevin anlamlı bir parçası olarak çalıştırılmıştır. Ancak iş birliği olmaksızın da sadece kendilerine atfedilen etki ve tepkileri yerine getirmek üzere çalışabilmektedirler. Çoklu ajanlar karmaşık modellerin benzetimi, veri madenciliği, güvenlik-casusluk vb. amaçlarla kullanılmıştır ve daha çok bir İnternet teknolojisi olarak anılmaktadır (Cao ve diğ. 1995, Ferber 1999).

Robotların ve ajanların yardımlaşması ile ortaya çıkan çoklu yapılar aynı zamanda büyük hesaplama görevlerinde de görülmektedir. Bu sistemler işlenecek verinin veya hesaplama işlemlerinin birden çok bilgisayara küçük parçalar halinde dağıtık veya paralel şekilde aktarılması prensibiyle çalışmaktadır. Bu sayede hesaplama görevi geleneksel olarak seri şekilde akmak yerine birden fazla işlem biriminde dağıtık veya paralel aktılır ve işlemin tamamı için harcanan süre azaltılmış olur (Attiya ve diğ. 2004).

## **1.2.2 Biyolojik Sürü Davranışları ve Uygulamaları**

Çoklu robot sistemlerinde biyolojik sürü davranışlarından esinlenilmiştir ve bu davranışların benzetimi ve saha uygulaması olarak pek çok çalışma vardır. Temel olarak sürü davranışları Parker ve diğerleri tarafından beş gruba ayrılmıştır (Parker 2008):

- Yiyecek Arama (Foraging) ve Kapsama (Coverage)
- Akın Etme (Flocking) ve Düzen Alma (Formation)
- Nesnelere Etki Etme
- Çoklu Hedef Gözleme ve Takip
- Sürü Trafiği ve Yol Planlama.

### **1.2.2.1 Yiyecek Arama ve Kapsama**

Yiyecek aramada, çevreye dağılmış hedeflerin veya yiyeceklerin robotlar tarafından bir veya bir kaç noktaya (yuvaya veya depoya) taşınıp biriktirilmesi amaçlanmaktadır (Balch 1999). Bu çalışmada robotların çok sıkı şekilde senkronize

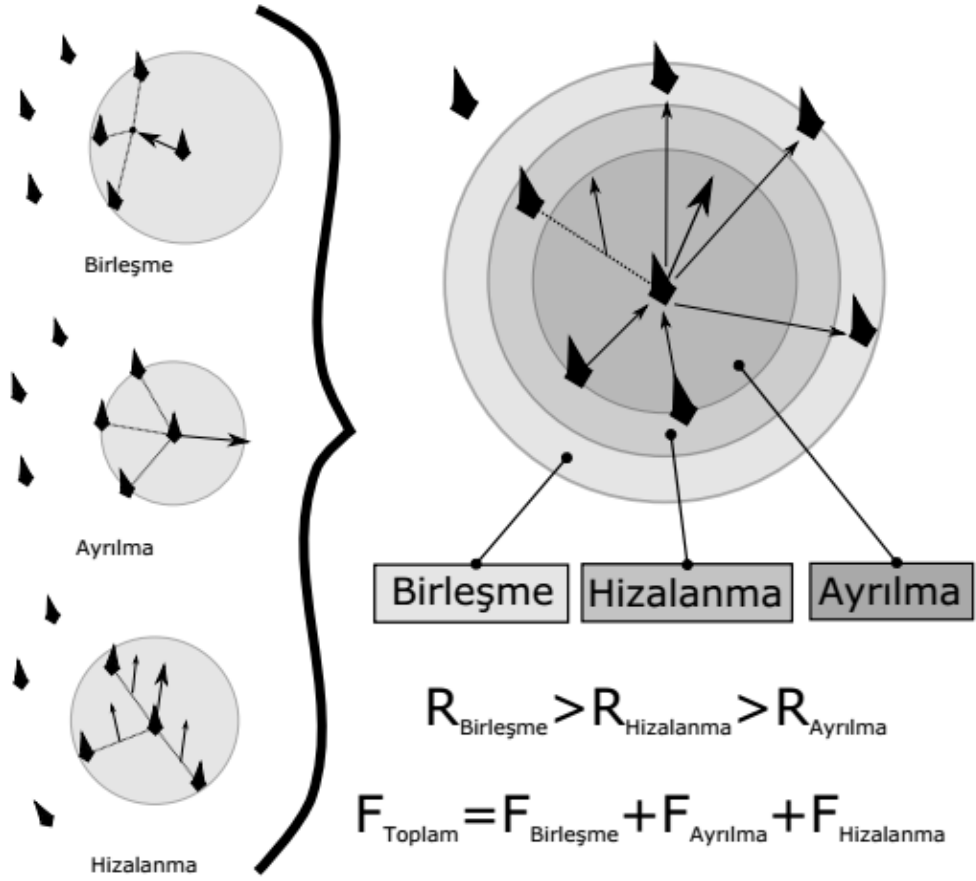
çalışması gerekmez. Zararlı (atık/mayın/bomba) temizleme, arama ve kurtarma gibi uygulamalar bu kapsamda değerlendirilebilir. Bunun yanında robotların hedefleri bulması için bölgelerini keşfetmesi gerekmektedir.

Kapsama davranışı, robotların bir bölgeye algılama veya denetleme kapasitelerine göre erişebildikleri tüm çevreyi kapsayacak şekilde yayılmalarıdır. Bu davranış yiyecek aramaya benzemektedir fakat robotların kendi çevrelerinde tüm alanı ziyaret etmesi gerekir (Butler ve diğ. 2000). Mayın temizleme, bahçe bakımı ve çevre haritalama gibi uygulamalar kapsama davranışına örnek olarak verilebilir.

Yiyecek arama ve kapsama davranışlarında temel problem robotun çevresini hızlı ve hareket tekrarı yapmadan veya birbirine çarpmadan nasıl keşfedeceğidir (Berhault ve diğ. 2003). Bunun için yapılmış dolaylı haberleşme (Beckers ve diğ. 1994), zincir oluşturma (Drogoul ve diğ. 1993), heterojen robotlar kullanma ve diğer yöntemler (Passino 2002) verilmiştir.

### **1.2.2.2 Akın Etme ve Düzen Alma**

Sürü Halinde Hareket Etme robotların bir yönde birlikte hareket etmesi olarak tanımlanmaktadır (Mataric 1993). Çoklu ajanlardan oluşan sürülerin hareketi için ilk çözüm algoritması Reynolds tarafından hazırlanan boids algoritmasıdır. Boids algoritması Şekil 1.1’de gösterilmiş olup birleşme (cohesion), ayrılma(dispersion) ve hizalanma (align) kurallarının oluşturduğu vektörlerin toplanmasıyla elde edilmiş bir algoritmadır (Reynolds 1987). Birleşme kuralı, birleşme alanı içinde algılanan komşuların konumlarına göre merkezlerine doğru hareketi sağlayacak bir çekme kuvvet vektör oluşturmaktır. Bu kural sürüye dâhil olma veya yaklaşma için kullanılmaktadır. Ayrılma kuralı, ayrılma için tanımlanmış bölgede algılanan komşu bireylerden uzaklaşma amacıyla oluşturulan itme kuvvet vektörüdür. Hizalanma kuralı ise birleşme ve ayrılma bölgeleri arasında tanımlı olan bir alanda sürünün aynı yönde veya aynı hedefe doğru hareket etmesi için komşuların hız vektörüne uyum sağlaması için oluşturulan bir kuvvet vektörüdür. Ayrılma, birleşme ve hizalanma bölgeleri robotun algılama, hareket ve sürü içi sıklık gibi parametrelere bağlı olup özelleştirilebilir. Tüm bu kuvvetlerin vektör toplamı sayesinde boids algoritması olarak bilinen sürü halinde hareket eylemi gerçekleştirilebilmektedir.



Şekil 1.1: Boids Algoritması Bölgeleri

Düzen alma, robotların, o anki bağıl konumlarına göre farklı geometrik şekillerde veya desenlerde dağılmaları veya şekli koruyarak hareket etmeleridir. Düzen alma davranışında anahtar problem yerel kontrol kurallarının belirlenmesi, yol planlama, hedef konumlandırma ve birey konumlandırmasıdır (Kloder ve diğ. 2006, Mourikis ve diğ. 2006, Vásárhelyi ve diğ. 2014) . Tüm kurallar ve yol planlama düzen alma için geçerli olan desene bağlıdır. Desenlerin hareket esnasında kullanılması, denklemleri bilinen basit geometrik şekiller için sıkıntı değildir. Fakat bu desenler karmaşıktıkça homografi tekniği ile desenin döndürülmesi, büyütülüp küçültülmesi veya çarpıtılması mümkün olmaktadır (López-Nicolás ve diğ. 2013, Aranda ve diğ. 2015).

### **1.2.2.3 Nesnelere Etki Etme**

Nesnelere etki etme davranışının birçok pratik uygulaması olacağı için çok ilgi çeken bir konudur. Bu davranışta genellikle robotların birbirine temas etmesi ve çok düzenli olarak çalışması gerekmektedir. Swarm Bots projesinde kendi kendine organize olan ve kendiliğinden birleşen robot tasarımı, uygulaması ve simülasyonu yapılmıştır (Dorigo ve diğ. 2004). Bu çalışmada robotların kol ve elleri birbirlerine ve etki edecekleri nesneye tutunmalarını sağlamaktadır. Robotlar elleri ile diğer robotlarla birleşip zincirler oluşturmak suretiyle organize olmakta ve bağlandıkları nesnelere taşıyabilmektedir.

### **1.2.2.4 Çoklu Hedef Gözleme ve Takip**

Çoklu hedef gözlemede Robotların birden çok hareketli hedefi gözlemesi veya takip etmesi amaçlanmaktadır. Bu görevde sorunlar hedef sayısının robot sayısından fazla olması durumunda ortaya çıkmaktadır. Bu duruma örnek olarak (Parker 1999, Beard ve diğ. 2002) verilebilir.

### **1.2.2.5 Sürü Trafığı ve Yol Planlama**

Sürü eğer kendisi için dar bir yerde çalışıyorsa, belirli geçitler varsa (kapı, yol, köprü, boğaz vb.) veya çevresini başka sistemlerle paylaşıyorsa çarpışma durumu ortaya çıkabilir. Bu durumda robotların çevreyi çarpışmadan veya kilitlenmeden verimli şekilde kullanması gerekmektedir. Robotlar arasında geçiş önceliklerinin ve izlenecek yolda geçiş sıralaması gibi problemleri çözmek için trafik kontrolü ve yol planlama işlemleri yapılmaktadır. Bu konudaki ilk çalışma Reynolds (Reynolds 2000) tarafından yapılmıştır. Reynolds robotların çarpışmasını önlemek amacıyla hazırladığı dümen algoritmaları ile robotlar arası çarpışmayı önlemeyi ve trafiği yönetmeyi başarmıştır. Kilitlenmeyi önleme konusunda Grossman (Grossman 1988), karşılıklı dışlama için de J. Wang (Wang 1991) çalışmalar yapmıştır.

### **1.2.3 Çoklu Robotik Sistemlerde Robotlar Arası İşbirliği**

Çoklu robot sistemlerinde robotlar arası verimli bir iş birliği için robotların birbirinin farkında olması, birbirleriyle etkileşime geçebilmesi, birbirlerinin hareketlerine uyum sağlaması ve örgütlenme eylemi gerçekleştirmesi gerekmektedir. Örgütlü yapının yönetimi kendi içinde merkezi ve dağıtık olabilir Merkezi sistemlerde çoklu robot sisteminin makro kararları bir merkezden yönetilir. Yönetim için tek bir merkez veya hiyerarşik bir yapı söz konusudur ve sistemin genel eylemleri merkezi olarak yönetilir (Farinelli ve diğ. 2004). İşbirliği kavramını oluşturan farkında olma, etkileşim, uyum sağlama ve örgütlenme için robotikte kullanılan sistemler haberleşme, algılama ve denetim konuları açısından incelenmelidir.

#### **1.2.3.1 Çoklu Robot Sistemlerinde Haberleşme**

Robotlar arası haberleşme doğrudan haberleşme, dolaylı haberleşme veya haberleşmesiz olarak üç şekilde gerçekleştirilmektedir. Doğrudan haberleşmede robotlar birbirleri ile radyo frekans (RF) veya kızıl ötesi gibi farklı yöntemlerle haberleşmektedir. Bu haberleşme eşler arası olabildiği gibi yayın halinde de olabilmektedir (Winfield ve diğ. 2008). Kızıl ötesi haberleşme kullanımı özellikle yer robotlarında kullanılmaktadır ve yakın mesafeler için etkilidir (Arvin ve diğ. 2009). Sürü robotığında robotların yönetilmesi ve durum gözleme işlemlerinde kullanım kolaylığı açısından RF haberleşme daha çok tercih edilmektedir. Kablosuz haberleşme protokolleri olan WiFi ve ZigBee çok sayıda noktanın haberleşmesini sağlayabilmeleri, mesh bağlantıyı desteklemeleri ve yaygın kullanımları sebebiyle birçok çalışmada robotlar arası haberleşme, yer belirleme ve yönetim amacıyla kullanılmıştır (Cianci ve diğ. 2006, Chen ve diğ. 2006, Winfield ve diğ. 2008, Couceiro ve diğ. 2012).

Dolaylı haberleşmede robotlar çevreyi kullanarak birbirlerine işaretler bırakma yoluyla iletişim kurmaktadır. Bu çalışmalarda karınca veya arı gibi canlıların feromon ile iletişime geçmesinden ilham alınmıştır. Uygulamada ise sanal feromonlar ile (basit fenerler ve yönlü sensörler) basit düzeyde haberleşme ve küçük işlem kabiliyetiyle gerçekleştirilmiştir (Payton ve diğ. 2001). Haberleşmesiz

sistemlerde ise robotlar kendilerinden olan robotları algılamakta ve sadece diğer robotların hareketlerine göre hareket etmektedir.

### 1.2.3.2 Çoklu Robotikte Algılama

Çoklu robot sistemlerinde algılama için kullanılan donanımlar tek robotlu sistemlerdekinden pek farklı değildir. Asıl fark bu sistemlerde kullanılan yöntemlerdedir. Geleneksel sensör birleştirme benzeri yöntemler, fikir birliği, oy birliği ve sürü davranışları gibi teknikler sayesinde çoklu robot sistemlerindeki algılama doğruluk, gürbüzlük, bağışıklık gibi avantajlar kazanmaktadır (Luo ve diğ. 1988, Stroupe ve diğ. 2001). Çoklu robotik sistemlerde bir bireyin çevredeki engelleri algılamaktan çok diğer robotların konumunu algılaması ve ona göre uyum sağlaması daha önemlidir. Robotikte yer belirleme işlemleri kendi arasında konumlandırma ve çevre algılama olarak ikiye ayrılmaktadır. Konumlandırma, robotun bir noktaya göre nerede olduğunu tespitidir. Çevre algılama ise robotun çevresindeki nesnelerin konumlarını kendisine göre belirlemesidir.

Konum belirlemede kullanılan sistemler göreceli ve mutlak olmak üzere ikiye ayrılmaktadır (Borenstein ve diğ. 1997, Liu ve diğ. 2007). Göreceli sistemler robot üzerinde bulunan basit sensör ile etkin poz takibi (odometry) yapılması ilkesine dayanmaktadır. Bu sistemler mutlak sistemlere göre daha kolay uygulanabilir ve ucuzdur ancak hata birikimlidir. Yani uzun süreli çalışma durumunda hataların birbirine eklenmesi ve son hatanın sürekli büyümesi söz konusudur bu sebeple hassas çalışmalarda tercih edilmemektedirler.

Mutlak sistemlerde konum bulma, sabit referans noktalardan alınan çeşitli bilgiler ile konum hesaplama ilkesine dayanmaktadır. Mutlak sistemler kendi içinde üçgenleştirme (triangulation ve trilitreatment) ve model eşleme tabanlı yöntemler olmak üzere ikiye ayrılır. Üçgenleştirme yöntemlerinde, konumu bilinen en az üç referans noktasından alınan açı veya mesafe bilgisi veren sensör verilerinden konum tahmini yapılır. Bu sistemlerin en bilineni GPS'dir. Mutlak sistemlerde doğruluk yüksektir ancak altyapı kurulumu zor ve pahalıdır. Ayrıca konumlandırma işlemi ortam bağımlıdır. Ortam bağımlılığı özellikle RF sinyaller ile çalışan GPS gibi yaygın sistemlerde kendini göstermektedir. Bu sistemlerde ortamdaki kaynaklı sinyal

zayıflaması veya yansımalar sebebiyle hatalı sonuçlar veya sonuç alamama problemleri ortaya çıkmaktadır. Model eşleme tabanlı konumlandırma sistemleri diğer sistemler arasında en gelişmiş ve karmaşıktır. Model eşleme bazı kaynaklarda sahne eşleme veya parmak izi (fingerprinting) analizi olarak da anılmaktadır (Siciliano ve diğ. 2008). Mevcut model eşleme sistemleri merkezidir. Konumu hesaplanacak nokta sayısı arttığında merkezin hesaplama gücünün artması gerekmektedir. Ayrıca merkez genellikle taşınabilirliği düşük büyük bir bilgisayardır. Bu da ortam bağımlılığı getirmektedir. Model eşleme çevrimdışı (offline) ve çevrimiçi (online) olmak üzere iki kısımdan meydana gelmektedir. Geleneksel model eşleme sistemlerinde çevrimdışı kısımda çevreden, robota özgün ve parmak izi de denilen veri toplama işlemi yapılır. Çevrimiçi kısımda ise parmak izi verisi bir merkeze gönderilir. Parmak izi verisinin merkezde işlenmesi esnasında merkezin veri tabanında desen tanıma teknikleri yardımıyla eşleştirme yapılarak konum bilgisi üretilir. Son olarak üretilen konum bilgisi robota aktarılır. Model eşleme ile yapılan çalışmalarda veri azaltma amacıyla öznitelik adı verilen gerçek verinin özeti veya önemli noktaları kullanılmaktadır bu şekildeki konumlandırma işlemlerinde öznitelik eşleme ile gerçekleştirilmektedir. Öznitelik algılama ve tanımlama algoritmaları görüntü işlemede sıklıkla kullanılan algoritmalar olup özellikle robotikte görsel eşzamanlı yer belirleme ve haritalama (VSLAM) amacıyla kullanılmaktadır (Se ve diğ. 2002, Davison 2003, Karlsson ve diğ. 2005, Steder ve diğ. 2008). Bu sistemler yüksek işlem gücü gerektirmektedir. VSLAM sistemleri özellikle robot üzerine monte edilebilmeleri ve üç boyutlu hareketleri algılayabilmeleri sebebiyle birçok robotik uygulamada kullanılmaktadır.

Çoklu robotikte konumlandırma için kullanılan sistemler taşınabilirlikleri açısından dâhili konumlandırma ve harici konumlandırma sistemleri olarak ikiye ayrılır. Dâhili sistemlerde, konumlandırmada sadece robotlar üzerlerinde bulunan donanımlar yardımıyla gerçekleştirilir ve göreceli konumlandırma sistemlerinin tamamı dâhili sensörlerle çalışmaktadır. Dahili sistemler ile gerçekleştirilen çoklu robot uygulamaları kendi içinde işbirlikçi veya bireysel olarak ikiye ayrılır (Fox ve diğ. 1999). İşbirliği içinde yapılmakta olan konumlandırmada robotlar kendi aralarında veri paylaşımı veya oylama gibi yöntemler kullanarak daha iyi ve genel sonuçlar üretmektedir. Bireysel konumlandırmada robotlar sadece kendi çevrelerindeki diğer robot ve nesnelere göre hareket etmektedir. Bu konumlandırma

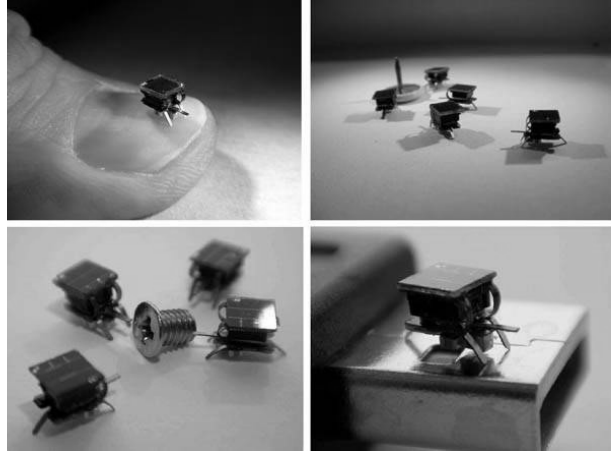


türünde robotlar hiçbir şekilde diğer robotlara konum ile ilgili bilgi paylaşmamaktadırlar. Harici sistemler ise konumlandırma için dışarıdan referans noktalarına, sensörlere veya işlem birimlerine ihtiyaç duyulan sistemlerdir. Bu sistemlere GPS, VICON ve Leica sistemleri örnek olarak verilebilir.

### 1.2.3.2.1 İki Boyutlu Ortamda Hareketler ve Algılama

Fiziksel bir robot sürüsü oluşturmanın hem donanımsal hem de yazılımsal açıdan büyük zorlukları vardır. Bundan dolayı sürü robotu çalışmaları iki boyutlu hareket edebilen yer robotları ile başlamıştır. Sürü robotiği konusundaki ilk çalışma sayılabilecek olan Mataric'in çalışmasına 20 robot kullanılmıştır (Matarić 1995) Bu sistemde toplanma, ayrılma ve akın davranışları gerçekleştirilmiştir. Daha sonra Swarm Bot adlı çalışmada 100 robotla daha kapsamlı bir çalışmaya gerçekleştirmiştir (McLurkin 2004). Bu çalışmada ayrılma, lideri izleme, grup yörüngesinde kalma, kümelenme gibi çeşitli davranışlar için yazılımsal katalog oluşturulmuştur. Daha sonraları Avrupa Birliği tarafından çoklu robot teknolojileri konusunda çeşitli projeler kabul edilmiştir ve bunlardan I-SWARM projesinde milimetre büyüklüğündeki 1000 adet robotla tüm ortaklaşa algılama işlemleri yapılmıştır (Seyfried ve diğ. 2005). Şekil 1.2'de görülen bu robotlarda tüm donanımsal gereksinimler sağlanmıştır (güç, algılama, denetim, haberleşme). Güneş enerjisiyle çalışan ve birbirleriyle iletişim halinde olan bu araçların mikro montaj, biyolojik, tıbbi ve temizlik işleri için elverişli oldukları öngörülmüştür. Bu robotlar tamamen otonomdur ve anlamlı ortaklaşa davranış sergileyebilmektedir. Robotlar 3x3x2mm boyutunda ve 1.5mm/s hızla hareket edebilmektedir. Proje sonucunda donanım ve yazılım açısından önemli ilerlemeler kaydedilmiştir. Diğer bir önemli sürü robotiği araştırması Şekil 1.3'de gösterilen Kilobot'lardır. Harvard Üniversitesi tarafından tasarlanan Kilobot'lar bozuk para büyüklüğünde titreşen bacakları, parlayan ışıkları ve birbirleriyle iletişim kurabilme özellikleriyle oldukça basit işlevselliğe sahip robotçuklardır. Projede işbirlikçi algoritmaları denemek amacıyla ve binlerle ölçülen rakamlarda robot sürüleri oluşturulmuştur (Rubenstein ve diğ. 2012). Ülkemizde de sürü robotiği konusunda çeşitli projeler gerçekleştirilmiştir. Bunların başında *Kobot Projesi* sayılabilir. Sürü robotik için tasarlanmış yeni bir robot kullanılmıştır.

Kızılötesi ile kısa mesafe algılama yapılarak nesnelere ve komşu robotlar algılanmıştır (Turgut et al. 2007).



Şekil 1.2: I-SWARM



Şekil 1.3: Kilobot

### 1.2.3.2.1 Üç Boyutlu Ortamda Hareketler ve Algılama

Son yıllarda üç boyutta hareket eden robotlara ilgi artmaktadır ve çeşitli hava aracı tipleriyle çoklu robot çalışmaları yapılmıştır. Yapılan çalışmalarda ilk olarak GPS'li sistemler tercih edilmiştir. GPS verisi ve basınç veya mesafe sensörlerinden alınan irtifa verisi birleştirilerek üç boyutta konumlandırma yapılmıştır. GPS ve yardımcı sensörler ile yapılan bu tip uygulamalarda dört rotorlu veya sabit kanatlı tip hava araçları kullanılmıştır (Quintero ve diğ. 2013, Vásárhelyi ve diğ. 2014).

Pennsylvania Üniversitesi GRASP laboratuvarlarında dört rotorlu hava araçlarının küçülterek nano hava aracı boyutuna getirildiği çalışmada algılama işlemi için harici kameralar kullanılmıştır. Kamera görüntüleri makine görüşü metotları ile robotların konum bilgileri çıkarılmakta ve bu konum bilgileri robotlara aktarılarak seyrüsefer, düzen alma, engel algılama ve yapı inşa etme gibi çeşitli uygulamalar gerçekleştirilmiştir. Şekil 1.4'de gösterilmiş olan sistemde kullanılan VICON hareket yakalama sistemi laboratuvar sabit kızıl ötesi kameralardan ve merkezi hesaplama biriminden oluşan bir sistemdir. Bu sistem duvar gibi sabit yapılara monte edildiği için taşınabilirliği yoktur. Bunun yanında sistemin çok hassas ve doğru sonuçlar ürettiği belirtilmiştir (Michael ve diğ. 2010, Kushleyev ve diğ. 2013).



Şekil 1.4: Dört Rotorlu Uçan Robot Sürüsü (GRASP)

Üç boyutta çalışan çoklu robotlara bir diğer alternatif de Şekil 1.5'de gösterilen Swarmanoid projesidir. Farklı işlevlere, fiziksel yeteneklere ve görüntüleme sahip olan Swarmanoid bir düzine robottan oluşan heterojen çoklu robot sistemidir. Bu sistemdeki robotlar sırasıyla şöyledir: ayak-botlar (foot-bot) iki boyutlu olarak hareket edebilirler ve diğer robotlara bağlanabilirler, el-botlar (hand-bot) manipülatörlere ve kendilerini diklemesine oynatabilen manyetik bir kancaya sahiptirler ve göz-botlar (eye-bot) uçabilir, tavana yapışabilir ve kameraları sayesinde el-botların ve ayak-botların hareketlerini düzenleyebilmektedirler (Dorigo 2009, Dorigo ve diğ. 2013). Swarmanoid sistemi konumlandırma sistemi olarak göz-botlardan alınan verileri kullanmaktadır. Bu sistem aslında iki boyutta hareket eden el ve ayak robotların hareketlerini yönlendirmeyi amaçlar ancak üç boyutlu

hareketler için tasarlanmış bir konumlandırma sistemi değildir. Ayrıca Swarmanoid sistemden göz-bot çıkarıldığında bu robotun yerini el veya ayak robot doldurmadığından sistem gürbüz değildir. Bu sistemle yapılan bir diğer çalışmada ise sadece göz-botlar ile çalışılmış olup GRASP laboratuvarlarındaki çalışmasında VICON sistemine benzer bir sistem olan Leica TS30 takip sistemi kullanılmıştır (Stirling ve diğ. 2012). Bu sistemde sabit yapılara monte edilmiş olduğu için taşınabilirliği yoktur.



Şekil 1.5: Swarmanoid (Soldan sağa el-bot, ayak-bot ve göz-bot)

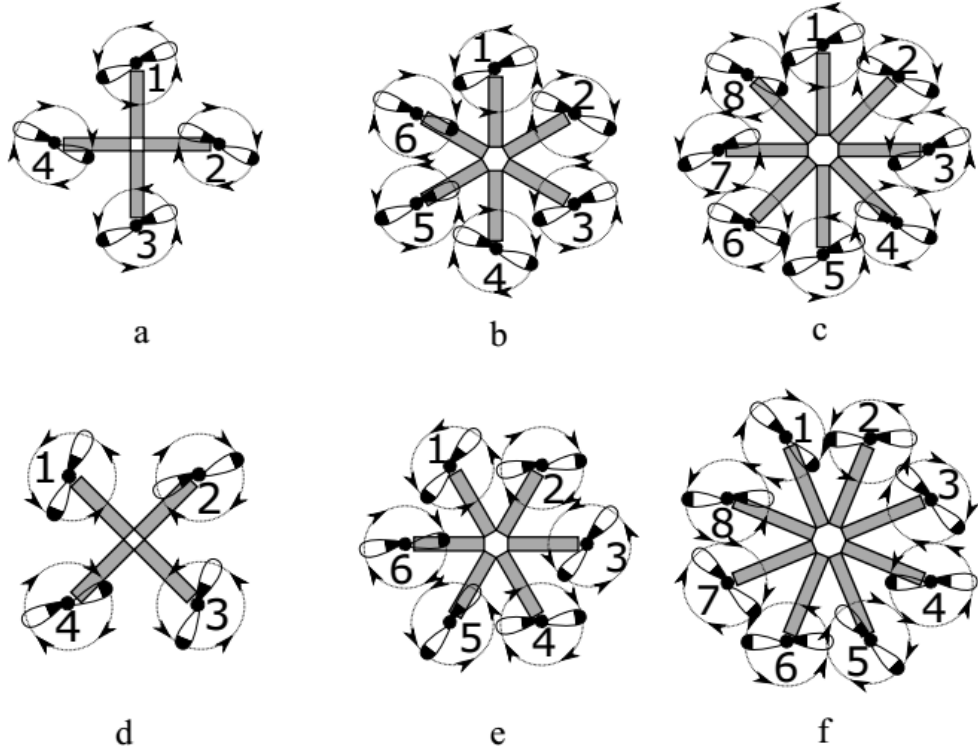
Uçan sürü sistemleri açısından Avrupa’da yapılan en büyük proje olan sFly projesi toplamda altı kuruluştan oluşan bir birlik tarafından gerçekleştirilmiştir. Bu proje kapsamında yapılan çalışmalarda farklı özelliklere sahip multikopterler tasarlanmış, üretilmiş ve arama-kurtarma, haritalama ve konumlandırma uygulamaları yapılmıştır (Achtelik ve diğ. 2012). Bu çalışmadaki konumlandırma işlemleri için temelde RF haberleşme sinyallerinin güç belirteçlerinden faydalanılmıştır (RSSI) Bunun yanında görsel konumlandırma çalışmaları da gerçekleştirilmiştir. Sinyal güç seviyesi mesafe ile ters orantılı olarak değiştiği için farklı zamanlarda alınan sinyal seviyeleri ile robotlar arası mesafe tahmini gerçekleştirilmiştir. Projede üç boyutlu haritalama (Heng ve diğ. 2011), bilinmeyen bölgelerin kaplanması (Renzaglia ve diğ. 2012) ve GPS engelli ortamlarda navigasyon (Weiss ve diğ. 2011) gibi uygulamalar yapılmıştır.

Üç boyutlu hareket eden sistemler sadece uçan sürü robotlardan değil aynı zamanda sualtı çalışan sistemler içinde geçerlidir. Deakin Üniversitesinde yapılan araştırmalarda bir su altı sürüsü için merkezi haberleşme sistemi üzerine çalışılmış ve konumlandırma için RF sinyallerin varış süresinden faydalanılarak bireyler arası mesafe tahmini yapılmıştır (Joordens et al. 2009, Shaneyfelt et al. 2008, Joordens et

al. 2016). Bahsi geçen sistemlerin haricinde LIDAR, stereo görme ve çok-yönlü (omnidirectional) görme gibi teknikler üç boyutlu seyrüsefer için kullanılmaktadır. Bu sistemler henüz sadece haritalama ve engel algılama amaçlı olarak kullanılmaktadır (Kriegman ve diğ. 1989, Murray ve diğ. 1997, Boutteau ve diğ. 2010, Douillard ve diğ. 2011) .

#### **1.2.4 Multikopterler ve Temel Kavramlar**

Çok rotorlu helikopterler Multikopter olarak anılmaktadır. Multikopterler üzerlerinde bulundurdukları rotor sayısına ve rotor denetim şekli göre adlandırılmaktadır. Rotor sayısı ve yerleşimleri genellikle simetrik şekilde seçilmektedir multikopter çalışmalarına bakıldığında temel iki tip rotor denetim şekli benimsenmiştir, bunlar artı (+) ve çarpı (x) şeklindedir. Artı şeklindeki yapılarda rotorların eksenler üzerinde, çarpı şeklindeki yapılarda ise rotorların eksen aralarında olması istenmiştir. Şekil 1.6-a, b ve c’de artı Şekil 1.6-d, e ve f’de çarpı yapıdaki dört (quadkopter veya quadrotor), altı (heksakopter) ve sekiz (oktokopter) rotorlu multikopter yapıları verilmiştir. Multikopter yapılarının denetimi ve modellenmesi çok benzer olup sadece kontrol denklemlerinden sonra hesaplanan motor hız değerlerinin multikopter tipine göre uyarlanması gerekmektedir. Bu açıdan multikopterlerin çalışma prensipleri çarpı tipinde bir dört rotorlu hava aracına ait model üzerinden verilecektir.



Şekil 1.6:Multikopter Tipleri

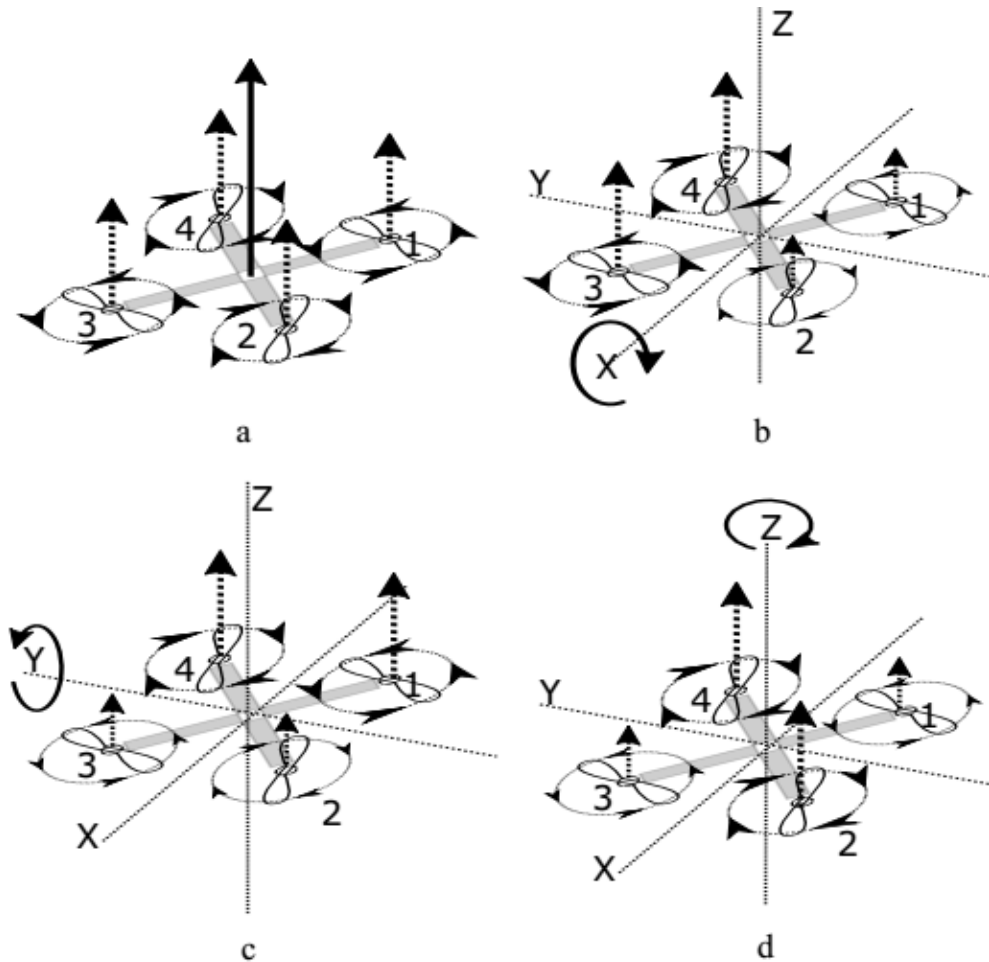
#### 1.2.4.1 Temel Hareketler

Dört rotorlu hava aracını üç boyutlu uzayda yönlendirebilmek ve hareket ettirmek için dört motorun ürettiği kuvvetlerden faydalanılır. Bir hava aracının havdaki temel hareketleri yuvarlanma (roll), yunuslama (pitch), sapma (yaw) ve irtifa kavramlarıyla açıklanmaktadır. Bu kavramları çeşitli kaynaklarda açıklanmış (Bouabdallah 2007, Erginer et al. 2007, Bouabdallah et al. 2007, Bresciani 2008) ve çarpı tipi hava aracı için bu temel hareketler Şekil 1.7’de gösterilmiştir.

- İrtifa (Yükselme/Alçalma): Bu hareket tüm rotor hızlarının aynı oranda artırılması veya azaltılmasıyla sağlanmaktadır. Yere göre z ekseninde alçalıp yükselme eylemi bu şekilde gerçekleştirilmektedir. Bu hareket Şekil 1.7-a’da gösterilmiştir.
- Yunuslama (Pitch): Öne veya arkaya gitmek için yapılan bir eylemdir. Şekil 1.7-b’de gösterilen bu eylem esnasında öne gitmek için öndeki rotorların (1 ve 2) hızları azaltıldığı oranda arkadaki rotorların (3 ve 4)

hızları arttırılmaktadır bu sayede x ekseninde dönme gerçekleşecektir ve yeterli eğimde devam edilmesi durumunda araç öne doğru ilerleyecektir.

- Yuvarlanma (Roll): Sağa veya sola hareket etmek için gerçekleştirilen eylemdir. Sağdaki rotorların (3 ve 4) hızlarının azaltıldığı oranda soldaki (4 ve 1) rotorların hızlarının arttırılması ile araç Şekil 1.7-c'de gösterildiği üzere y ekseninde dönme gerçekleştirecek ve araç sağa doğru hareket edecektir.
- Sapma (Yaw): Aracın z ekseninde dönmesi yani pozunu değiştirmesi için kullanılan bir eylemdir. saat yönünde dönen rotorların (4 ve 2) hızının arttırıldığı oranda saat yönünün tersinde dönen rotorların (1 ve 3) hızlarının azaltılması sonucu Şekil 1.7-d'de gösterildiği üzere saat yönünde dönme eylemi gerçekleştirilir.



Şekil 1.7: Temel Hareketler: a)İrtifa b)Yunuslama c)Yuvarlanma d)Sapma

#### 1.2.4.2 Dört Rotorlu Hava Araç Modeli

Dört rotorlu hava aracının dinamik ve kinematik modeli genellikle Newton-Euler modeli ile açıklanmıştır (Bouabdallah 2007, Bresciani 2008, Salih et al. 2010, Amir et al. 2011, do Nascimento et al. 2012, Honglei et al. 2013). Bu modellere göre 3 serbestlik derecesine sahip gövde ve 3 serbestlik derecesine sahip referans çerçevelerinden meydana gelen 6 serbestlik dereceli bir model tanımlanmıştır. Ayrıca gimbal kilitlemesini önlemek için geliştirilmiş olan quaternion'lar ile de modelleme yapılmıştır (Fresk ve diğ. 2013).

Newton-Euler model denklemleri göre Şekil 1.8'de gösterilen çarpı yapıdaki bir dört rotorlu için uyarlandığında denetim için kullanılacak irtifa eşitlik (1.1)'de, yunuslama (1.2)'de, yuvarlanma (1.3)'de ve sapma (1.4)'de verilmiştir. Bu denklemlerdeki  $\omega_1$ ,  $\omega_2$ ,  $\omega_3$  ve  $\omega_4$  alt indislere ait rotorların dönüş hızlarıdır. Tasarım parametreleri  $b$ ,  $d$  ve  $l$  sırasıyla itki, sürüklenme ve rotor ile merkez arası kol uzunluğudur. Dört rotorlu hava aracı için pervane hızları sırasıyla denklem (1.5), (1.6), (1.7) ve (1.8)'de verilmiştir. Pervane hız denklemlerinde paydadaki bölüm pervane sayısı ve yeri ile ilişkilidir.

$$U_{irtifa} = b(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \quad (1.1)$$

$$U_{yunustama} = lb(-\omega_1^2 + \omega_2^2 + \omega_3^2 - \omega_4^2) \quad (1.2)$$

$$U_{yuvarlanma} = lb(-\omega_1^2 - \omega_2^2 + \omega_3^2 + \omega_4^2) \quad (1.3)$$

$$U_{sapma} = d(-\omega_1^2 + \omega_2^2 - \omega_3^2 + \omega_4^2) \quad (1.4)$$

$$\omega_1 = \frac{U_{irtifa}}{4b} + \frac{U_{yunustama}}{4bl} + \frac{U_{yuvarlanma}}{4bl} - \frac{U_{sapma}}{4d} \quad (1.5)$$

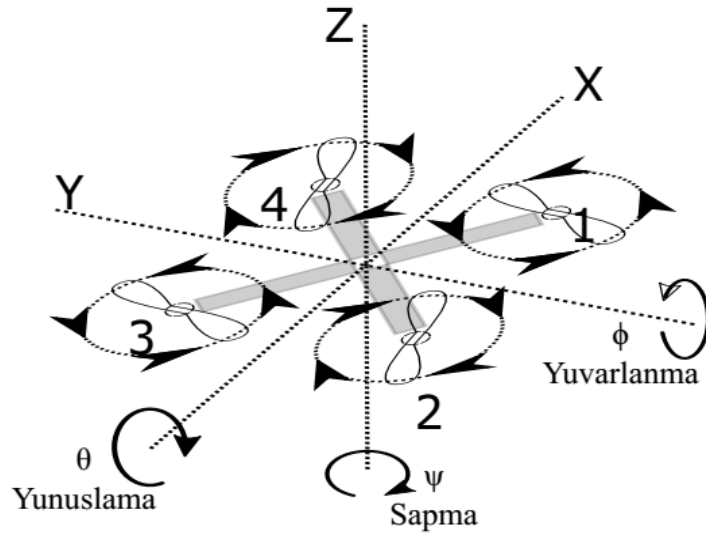
$$\omega_2 = \frac{U_{irtifa}}{4b} + \frac{U_{yunustama}}{4bl} - \frac{U_{yuvarlanma}}{4bl} + \frac{U_{sapma}}{4d} \quad (1.6)$$

$$\omega_3 = \frac{U_{irtifa}}{4b} - \frac{U_{yunustama}}{4bl} - \frac{U_{yuvarlanma}}{4bl} - \frac{U_{sapma}}{4d} \quad (1.7)$$

$$\omega_4 = \frac{U_{irtifa}}{4b} - \frac{U_{yunustama}}{4bl} + \frac{U_{yuvarlanma}}{4bl} + \frac{U_{sapma}}{4d} \quad (1.8)$$



Temel hareket denklemleri açısal ivme eşitlikleri halinde (1.10), (1.11) ve (1.12)'de verilmiştir. Eşitlik (1.9) tüm pervane dönüş yönlerine göre oluşan torku hesaplamada kullanılan bir parametredir. Bu denklemlerdeki  $\phi$ ,  $\theta$  ve  $\psi$  sırasıyla yuvarlanma, yunuslama ve sapma eylemlerini gerçekleştirmedeki açısal dönmesidir.  $J_{tp}$  ile ifade edilen motor atalet momentidir. Eksenlere göre atalet momentleri I harfi ve indisler ile ifade edilmiştir. Eşitlik (1.13), (1.14) ve (1.15) sırasıyla X, Y ve Z eksenlerindeki doğrusal ivmelerdir. Bu denklemlerdeki m hava aracının ağırlığı g ise yer çekimi ivmesidir.



Şekil 1.8: Dört Rotorlu Hava Aracı Çerçevesi

$$\omega = -\omega_1 + \omega_2 - \omega_3 + \omega_4 \quad (1.9)$$

$$\ddot{\phi} = \frac{I_{yy} - I_{zz}}{I_{xx}} \dot{\theta} \dot{\psi} + \frac{U_{yuvarlanma}}{I_{xx}} - \frac{J_{tp}}{I_{xx}} \dot{\theta} \omega \quad (1.10)$$

$$\ddot{\theta} = \frac{I_{zz} - I_{xx}}{I_{yy}} \dot{\phi} \dot{\psi} + \frac{U_{yunuslama}}{I_{yy}} - \frac{J_{tp}}{I_{yy}} \dot{\phi} \omega \quad (1.11)$$

$$\ddot{\psi} = \frac{I_{xx} - I_{yy}}{I_{zz}} \dot{\phi} \dot{\theta} + \frac{U_{sapma}}{I_{zz}} \quad (1.12)$$

$$\ddot{X} = (\sin \psi \sin \phi + \cos \psi \sin \theta \cos \phi) + \frac{U_{irtifa}}{m} \quad (1.13)$$

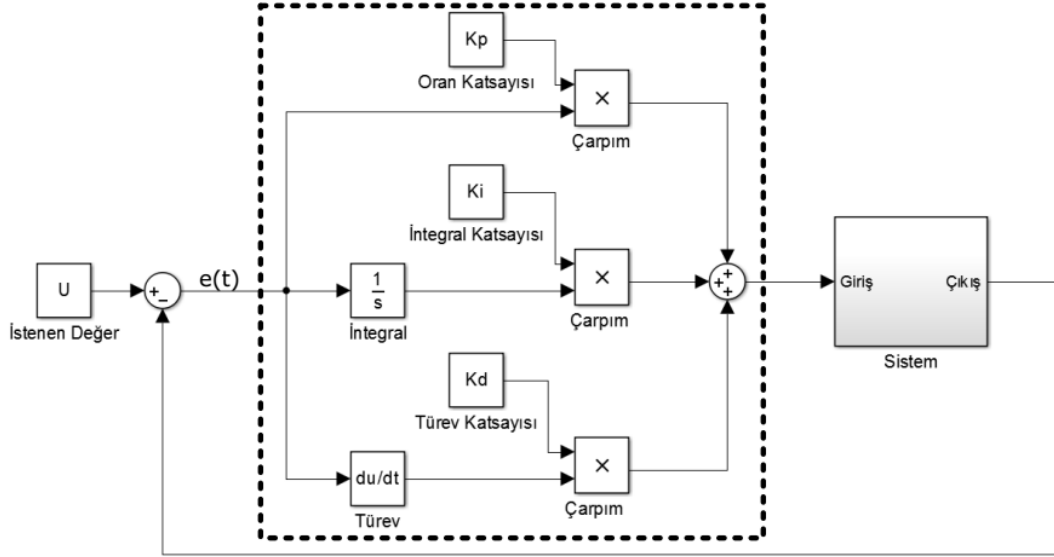
$$\ddot{Y} = (\cos \psi \sin \phi + \sin \psi \sin \theta \cos \phi) + \frac{U_{irtifa}}{m} \quad (1.14)$$

$$\ddot{Z} = -g + (\cos \theta \cos \phi) + \frac{U_{irtifa}}{m} \quad (1.15)$$

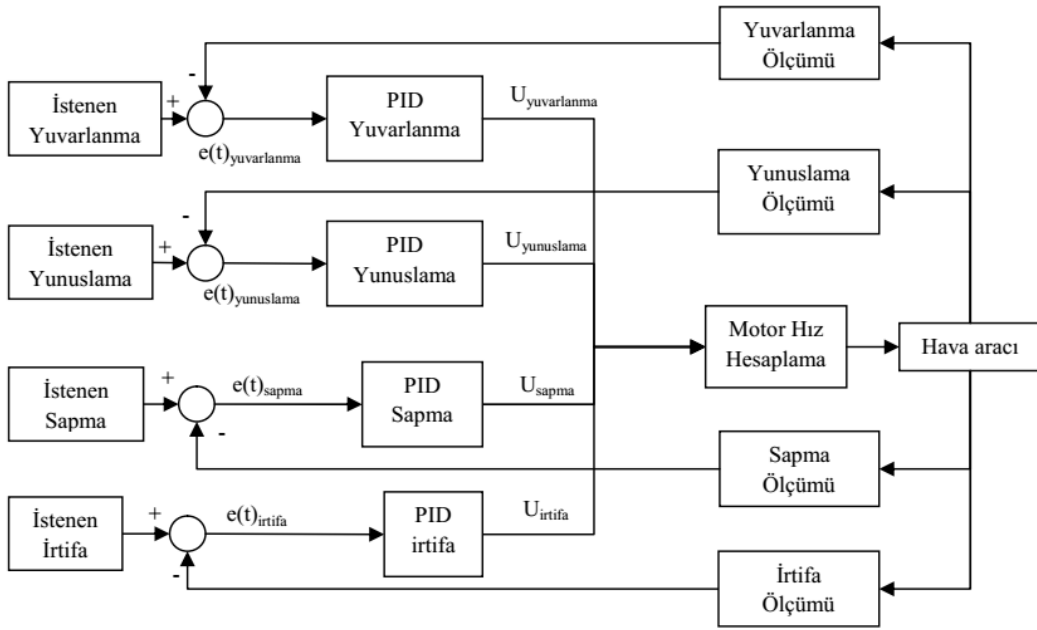
### 1.2.4.3 Dört Rotorlu Hava Araçlarında Denetim Çalışmaları

Dört rotorlu hava araçlarının denetimi konusunda yapılan bilimsel çalışmalar ağırlıklı olarak irtifa ve durum denetimi konusundadır. Bu çalışmalarda PID (Bouabdallah 2007, Bresciani 2008, Salih et al. 2010, Efe 2011, Li et al. 2011, Bolandi et al. 2013), bulanık mantık (Santos ve diğ. 2010, Chen ve diğ. 2011), backstepping (Madani ve diğ. 2007, 2008, Fang ve diğ. 2011), kayan kip (sliding-mod) (Madani ve diğ. 2007, Efe 2010) ve yapay sinir ağları (Efe 2011, Nicol 2011) yöntemleri kullanıldığı görülmüştür. Bu çalışmaların büyük çoğunluğu benzetim ve saha uygulaması şeklindedir. Geleneksel olarak PID Şekil 1.9'da gösterildiği gibidir. Şekilde hata değeri istenen değerden sistem çıkış değerinin çıkarılmasıyla bulunmaktadır. PID denetim işlemi hata girdisine göre Şekil 1.9'da kesikli çizgi içinde bloklarla ve denklem (1.16)'de verilen eşitlikle gösterildiği şekilde uygulanmasıyla bir PID çıkış değeri üretmektedir ve bu çıkış değeri sistem girişine uygulanmaktadır. Bir hava aracındaki durum ve irtifa denetimi için PID uygulaması Şekil 1.10'da gösterilmiştir bu şekle göre durum denetiminde kullanılan yuvarlanma, yunuslama ve sapma denetimi çıkışları ile irtifa denetimi motor hızlarının belirlenmesi için motor hız hesaplamasına tabi tutulur. Motor hız hesaplama bir dört rotorlu hava aracı için denklem (1.5), (1.6), (1.7) ve (1.8)'de verilmiştir bu hız değeri ve hesaplamadan sonra seçilen motorlara göre ölçekleme veya ihmaller yapılabilir.

$$PID = K_p * e(t) + K_i \int_0^t e(t) d\tau + K_d \frac{de(t)}{dt} \quad (1.16)$$



Şekil 1.9: Geleneksel PID denetimi



Şekil 1.10: Hava Aracı Denetimi

### 1.3 Problem ve Bilimsel Yazına Katkı

Tezin konusu hakkında bilimsel çalışmalar incelendiğinde üç boyutlu hareket eden çoklu robot sistemlerinde:

- Çoklu uçan robotlarda konumlandırma işlemleri için kullanılan sistemler mutlak sistemlerdir. Mutlak sistem kullanan çoklu robot uygulamalarında sistemler ortam bağımlıdır, kurulum maliyetlidir,
- Mutlak sistemler ile çalışan robotlarda konumlandırma işlemi için kullanılan asıl donanım harici olduğu için robotlar oldukça düşük işlem gereksinimi duymaktadır. Mevcut çalışmalar incelendiğinde de genellikle robotların işlem kapasitelerinin düşük tutulduğu ve merkezi denetim ile çoklu robot sistemlerinin yönetildiği görülmektedir,
- Çoklu robot sistemlerinde işbirliği ile çalışan sistemlerin bireysel sistemlere göre daha avantajlı olduğuna dair pek çok kaynakta ifadeler kullanılmıştır fakat aynı görevi yerine getirecek bireysel çalışan bir grup robot ve işbirlikli çalışan bir grup robotun çalışmalarını karşılaştıran bir çalışmaya rastlanmamıştır.

Bu çalışmada tasarımı amaçlanan sistem ile bilimsel yazından farkları ve sağlayacağı katkılar:

- Üç boyutlu hareket edebilen çoklu robot sistemi için daha yüksek işlem kapasitesine sahip alternatif bir yapı sunulması ve farklı tipte hava araçları ile çalışabilecek bir sistem geliştirilecek olması açısından mevcut sistemlerden farklı olup kendisinden sonra yapılacak çalışmalara altyapı sunması açısından katkı sağlayacaktır.
- Model eşleme tabanlı konumlandırma ve tek gözlü etkin poz takibi yöntemleri ile geleneksel mutlak konumlandırma yöntemleri gibi merkezi değil dağıtık olarak çalışabilecek olması açısından Mevcut uygulamalardan farklı olarak hem kapalı hem açık alanda çalışabilmesi beklenmektedir. Bu açıdan birden fazla destek konumlandırma sistemi içermesi ile mevcut sistemlere alternatif sunulmaktadır.
- Bu çalışma ile aynı görevi yerine getirecek çoklu robot sistemlerinde bireysel ve işbirlikçi davranışlar arasındaki farkı ortaya koyması açısından bilimsel yazına veri sağlayarak katkı sunacaktır. Ayrıca çalışmada akın etme ve düzen alma açısından bulanık mantık yaklaşımı ile yeni bir model sunulmuştur.

## 2. MATERYAL VE YÖNTEM

Bu kısımda çoklu robotik uygulamalarında kullanılmak üzere tasarlanan uçan robot için sistemi oluşturan bileşenler, bileşenler arası bağlantılar, denetim için kullanılan yöntem ve algoritmalar sunulmaktadır.

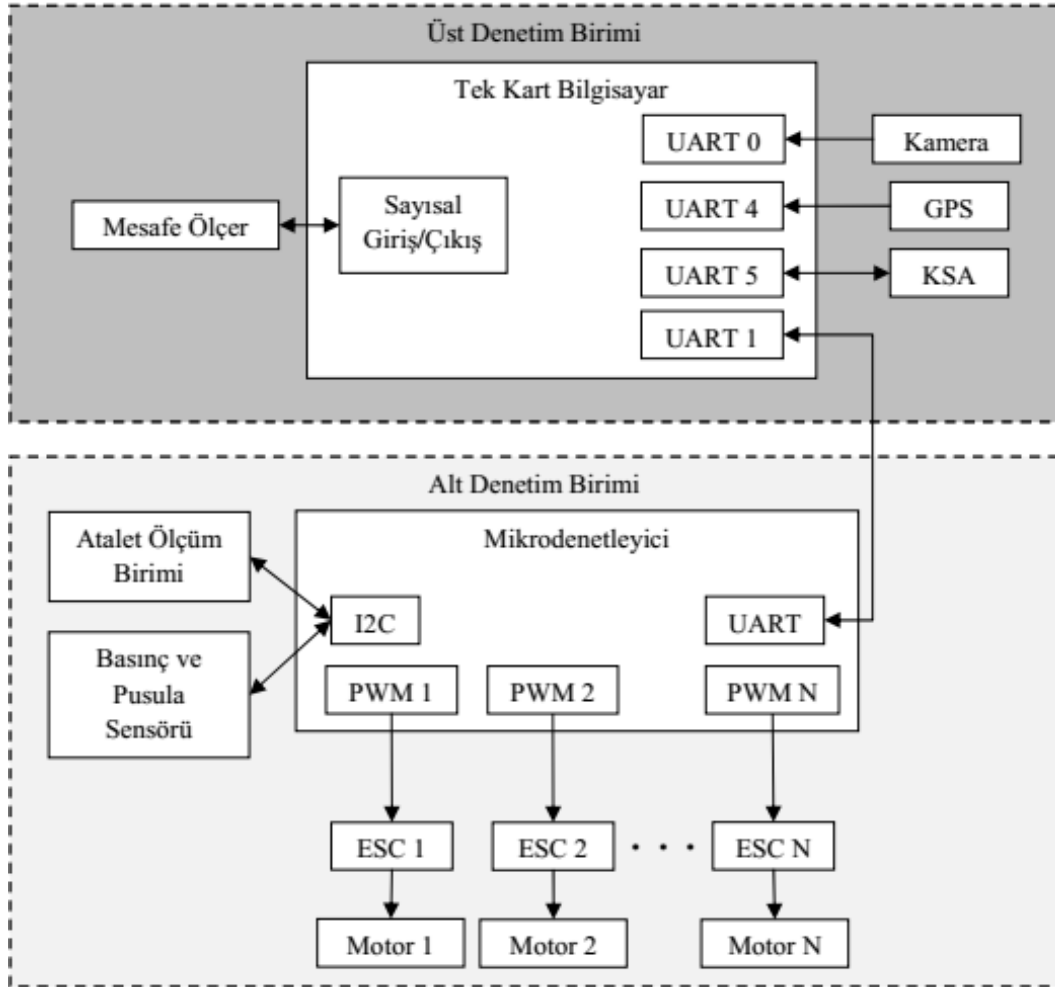
### 2.1 Denetim Kartı Tasarımı

Denetim sistemi farklı tipte hava araçlarında rahatça çalıştırılabilmesi ve görev yükünün paylaşılması amacıyla Şekil 2.1’de gösterildiği gibi üst ve alt denetim birimleri olarak iki kısımdan meydana gelmektedir. Uçan robot tasarımda kullanılan bileşen türleri, markaları, kullanım amacı ve hava aracına göre adetleri Tablo 2.1’de verilmiştir.

Üst denetim birimi kullanıcı denetimi, sürü denetimi, yol planlama, görüntü işleme, irtifa denetimi, uçuş komutu üretme ve veri depolama işlemlerini gerçekleştiren ana yönetim kısmıdır. Bu birime kamera, GPS, Kablosuz Sensör Ağı (KSA) Modülü, mesafe sensörlü tarayıcı ve kısa mesafe irtifa ölçümü için mesafe sensörü doğrudan bağlanmıştır. GPS, KSA modülü, kamera ve alt denetim kısmı üst denetim birimine Evrensel Asenkron Alıcı Verici (UART) protokolü ile bağlıdır yani seri haberleşme kullanılmaktadır. Servo motor ve mesafe ölçerler genel giriş çıkış pinlerine bağlıdır. Servo motor sinyal girişi PWM sinyal çıkışı bulunan P8\_14 pinine bağlıdır. Mesafe ölçerler ise sinyal tetiği ve yansıyan sinyal süresini ölçmek için her biri sayısal bir giriş ve bir çıkış bağlantısına ihtiyaç duymaktadır.

Alt denetim birimi hava aracının temel uçuş işlemlerini gerçekleştiren birimdir. Bu kısımda motor hızlarının üst kısımdan gelen komutlara göre durum denetimini gerçekleştirmektedir. Alt denetim kısmında atalet ölçüm birimi, basınç ölçer ve pusula uçuş denetiminde kullanıldığı için doğrudan bu birime bağlanmıştır. Atalet ölçüm birimi, pusula ve basınçölçer I2C bağlantı protokolü ile bağlanmıştır. Fırçasız motor sürme işlemleri için elektronik hız ayarlama (ESC) devreleri mikrodenetleyici PWM çıkışları ile sürülmektedir.

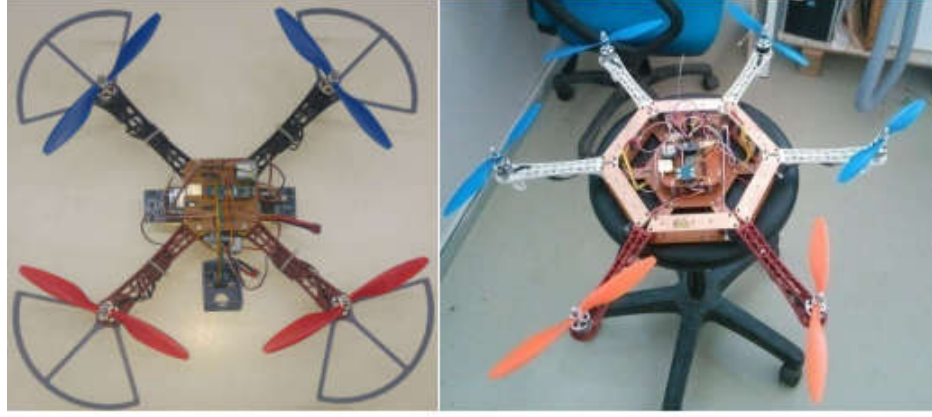
Tasarlanan uçan robotun dört rotorlu hava aracı için montajlı hali Şekil 2.2’de gösterilmiştir. Denetim kartı için devre kartı tasarımı Autodesk-EAGLE programı ile yapılmıştır ve devre üretimi LPKF-S103 cihazı ile gerçekleştirilmiştir ilgili devrenin baskı devresi Şekil 2.3’de bağlantı şeması Şekil 2.4’ de gösterilmektedir.



Şekil 2.1: Otonom Hava Aracı Denetim Sistem Tasarımı

Tablo 2.1: Malzeme Listesi

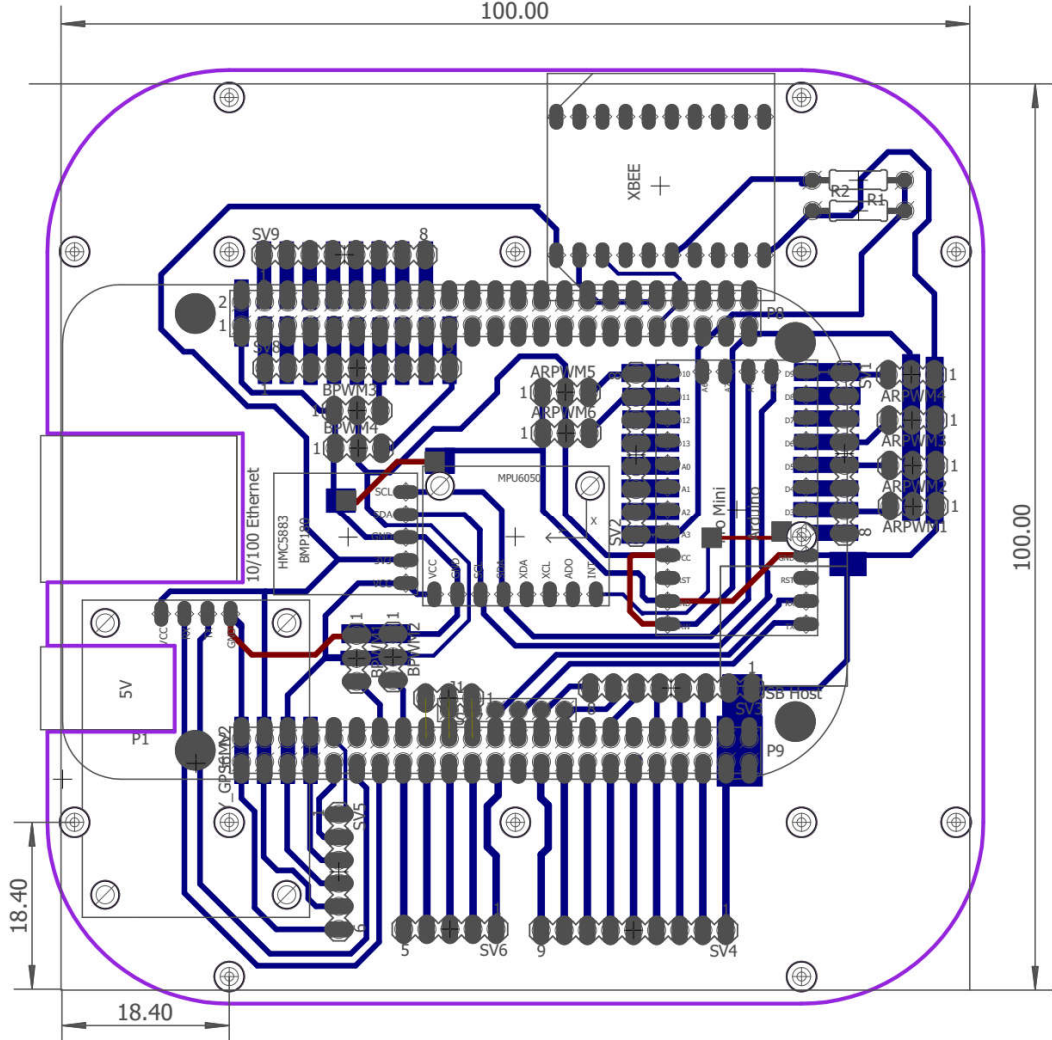
Adı	Açıklama	Amaç	Adet
Tek Kart Bilgisayar	BeagleBone Black Rev. C	Üst denetim sistemi yazılımını çalıştıracak birim.	1
Mikrodenetleyici	Arduino Pro Mini	Alt denetim sistemi işlevlerini yerine getirmek.	1
Kablosuz Sensör Ağı (KSA) Modülü	XBee 1mW Kablo Antenli	Hava araçları arasında ve kullanıcı komutlarını almak amacıyla haberleşme birimi.	1
GPS	GY-NEO6Mv2 GPS Modülü NMEA Destekli	GPS konumlandırma bilgisi sağlamak.	1
Kamera	640x480 Renkli 30fps CMOS Webcam	Görsel uygulamalar için görüntü alma.	1
USB Çoklayıcı	4 Port USB 2.0 Mini Hub.	Kamera ve diğer usb arayüzü aygırlarını tek kart bilgisayara bağlama.	1
Mesafe Sensörü	HC-SR04 ultrasonik mesafe sensörü	İki tanesi kısa mesafe çevre tarama diğeri de kalkış esnasında kısa mesafe irtifa ölçümü için.	1
Atalet Ölçüm Birimi (IMU)	MPU6050 İvme Ölçer ve Jiroskop Sensörlerini barındıran kart.	Mikrodenetleyiciye uçuş denetimini sağlama için üç ekseninde açısal hareketler ve ivme verisi sağlamak.	1
Pusula ve Atmosferik Basınç Sensör Kartı	HMC5883L ve BMP085 Sensör Kartı	Üst ve alt denetim sistemlerine yön ve irtifa bilgisi sağlamak	1
Batarya	Li-Po 11.1V 2600mAh 25C	Hava aracının elektriksel güç ihtiyacını karşılamak.	1
Motor	1400KV Fırçasız Doğru Akım Dış Döner Motor	Elektriksel enerjiyi pervaneleri döndürme yoluyla itme kuvvetine dönüştürmek.	Hava aracının rotor sayısı kadar
ESC	20A ESC	Mikrodenetleyiciden gönderilen PWM sinyallerine göre BLDC motorların dönüş hızını ayarlamak	
Pervane	10x4.5 ölçüsünde Pervane çifti	Motordan gelen dönme kuvvetini itme kuvvetine çevirme.	Hava aracının rotor sayısının yarısı kadar



(a)

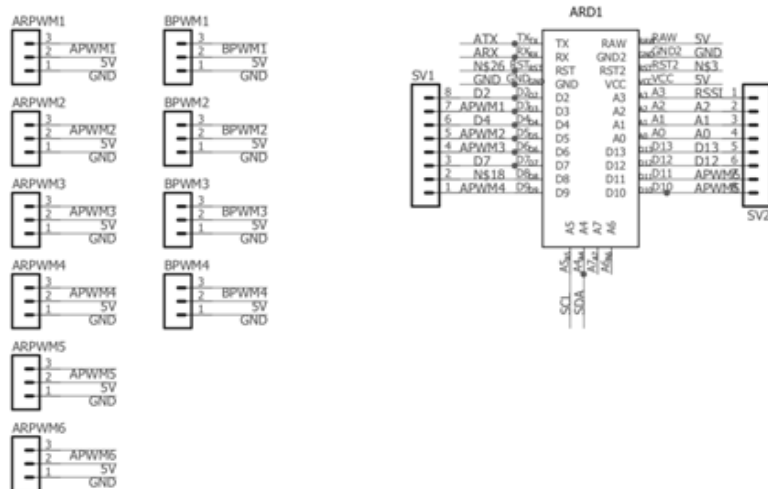
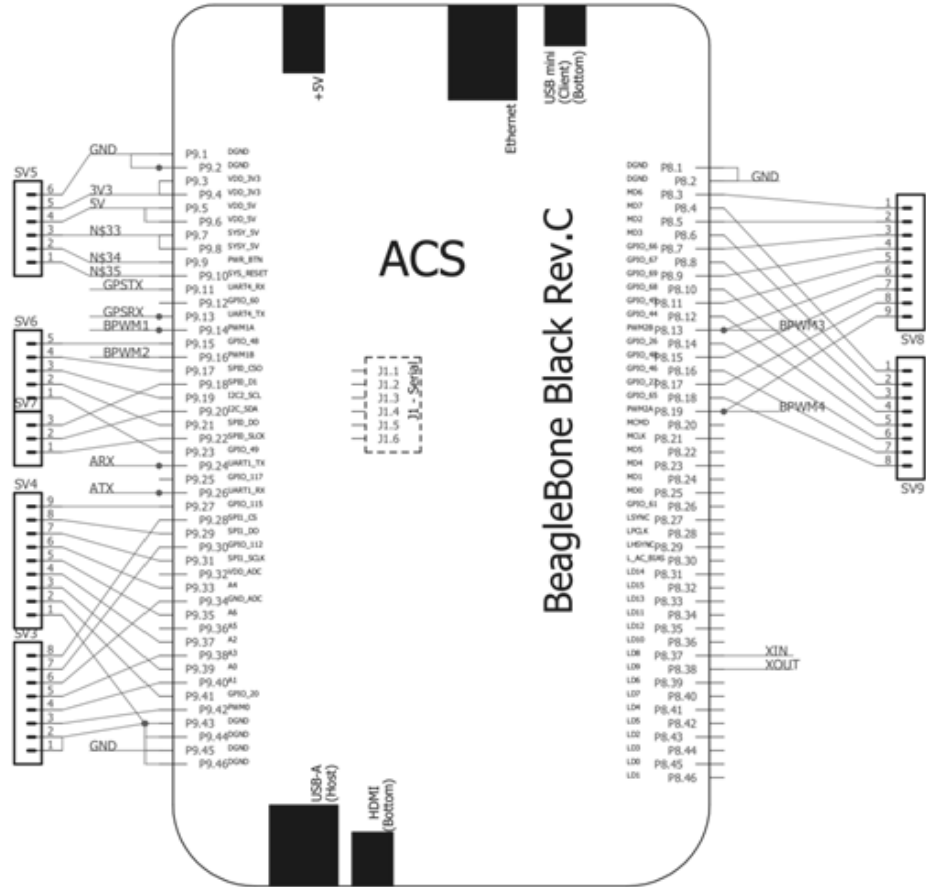
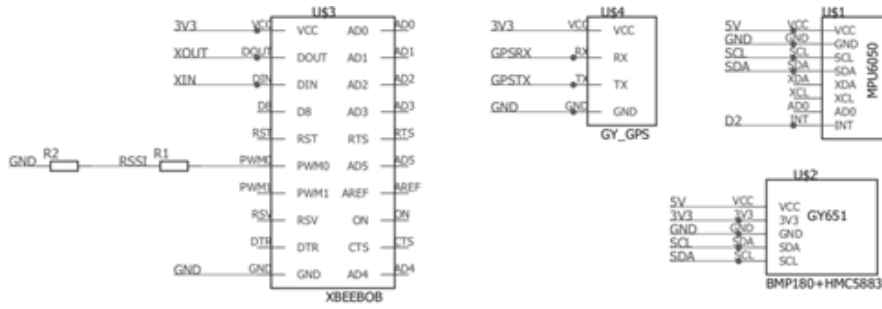
(b)

Şekil 2.2:Hava Araçlarının Montajı Halleri: a) Dört Rotorlu b) Altı Rotorlu



Şekil 2.3: Denetim Kartı Baskı Devre Çizimi

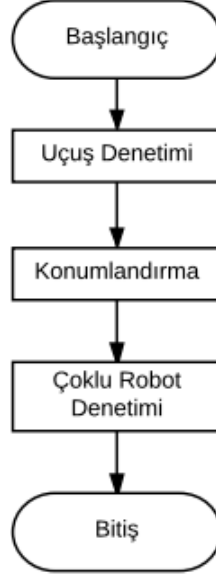




Şekil 2.4: Denetim Kartı Şematik Çizimi

## 2.2 Yazılım Tasarımı

Bu kısımda sisteminde kullanılan ana algoritma ve bileşenleri açıklanmaktadır. Tasarlanan genel algoritma Şekil 2.52'te gösterilmiştir. Algoritma uçuş denetimi, konumlandırma ve çoklu robot denetimi kısımlarından oluşmaktadır.



Şekil 2.5: Genel Algoritma

### 2.2.1 Uçuş Denetimi

Uçuş denetimi hava aracının dört temel hareketini dengeli şekilde gerçekleştirebilmesi için gereklidir. Bu amaçla tasarlanan denetim yazılımı bilimsel yazın bölümünde verilen geleneksel PID denetim yönteminden türetilerek tasarlanarak iki kısımdan meydana getirilmiştir. Bunlar durum ve irtifa denetimleridir. Uçuş denetimi için kullanılan genel yöntem Şekil 2.6'da gösterilmiştir. Buna göre yuvarlanma, yunuslama ve sapma hareketlerine göre PID çıkış değerleri ayrı ayrı hesaplanmakta ve sonra irtifa denetiminden gelecek değerlerin eklenmesi ile motor hız hesaplaması gerçekleştirilmektedir. Motor hız hesaplaması için kullanılan eşitlikler dört rotorlu hava aracı için denklem (2.1), (2.2), (2.3) ve (2.4)'de verilmiştir. Buradaki M1, M2, M3 ve M4 motor hızları  $C_{yunuslama}$ ,  $C_{yuvarlanma}$ ,  $C_{sapma}$  ve  $C_{itme}$  ise PID çıkışlarıdır.

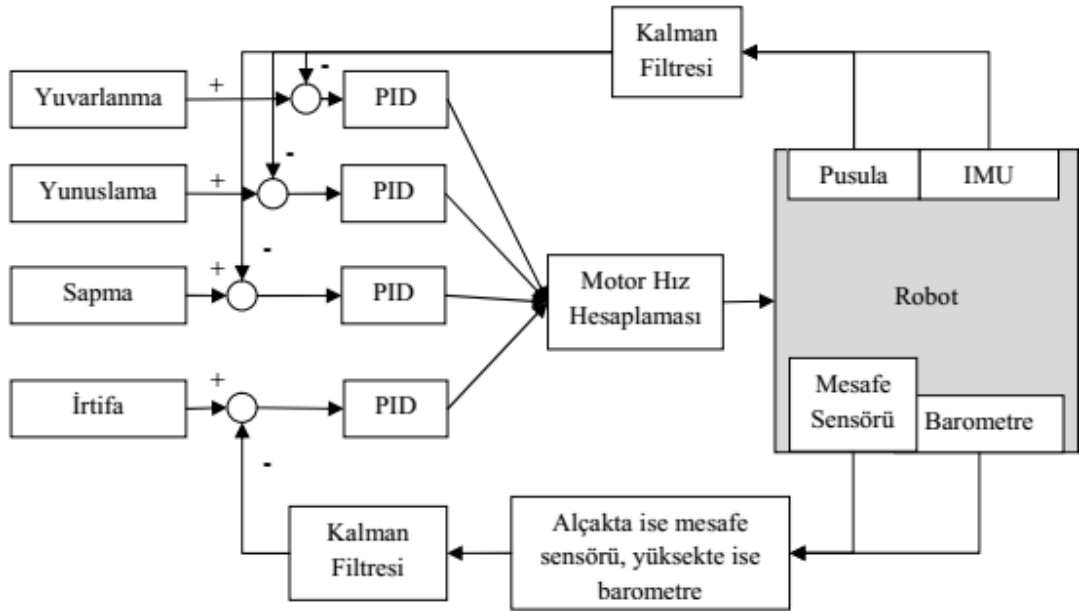
Durum denetimi mikrodenetleyici seviyesindeki alt denetim birimindedir. Bu birimde denetim için gerekli yazılım Arduino programlama dili ve ilgili kütüphaneler ile gerçekleştirilmiştir. İrtifa denetimi ise üst denetim birimi tarafından yönetilmiş gerekli yazılım olarak Python programlama dili ve ilgili kütüphaneler kullanılmıştır.

$$M_1 = C_{yunuslama} + C_{yuvarlanma} + C_{sapma} + C_{itme} \quad (2.1)$$

$$M_2 = C_{yunuslama} - C_{yuvarlanma} - C_{sapma} + C_{itme} \quad (2.2)$$

$$M_3 = -C_{yunuslama} - C_{yuvarlanma} + C_{sapma} + C_{itme} \quad (2.3)$$

$$M_4 = -C_{yunuslama} + C_{yuvarlanma} - C_{sapma} + C_{itme} \quad (2.4)$$

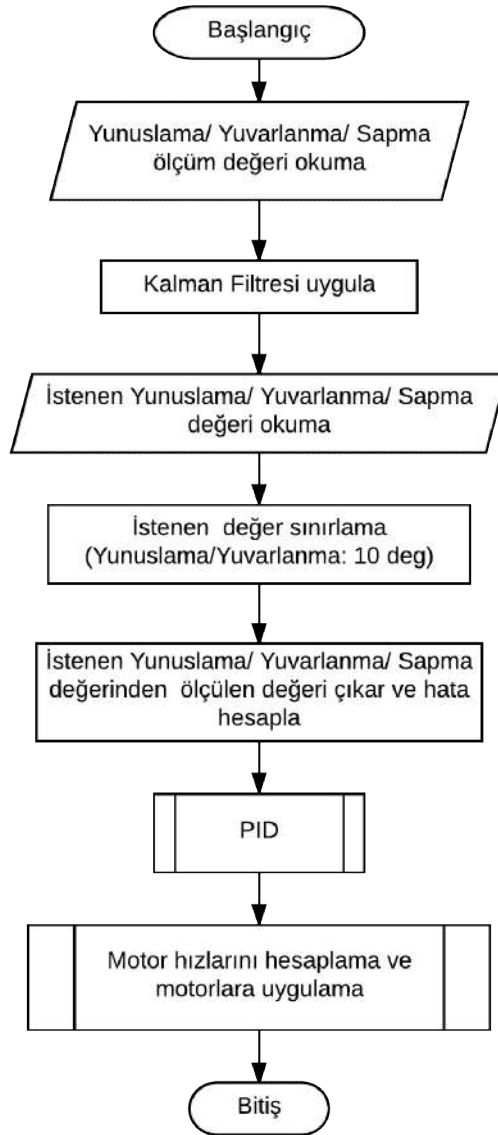


Şekil 2.6: Uçuş Denetimi

### 2.2.1.1 Durum Denetimi

Durum denetimi hava aracının yunuslama, yuvarlanma ve sapma hareketlerinin nasıl yapıldığını yönetmek için kullanılmaktadır. Yuvarlanma, yunuslama ve sapma hareketleri için PID denetimi Şekil 2.7’ de gösterildiği gibi uygulanmıştır. İstenen açı değeri sonraki aşamada robotun devrilmesini önlemek amacıyla yunuslama ve yuvarlanma için  $[-10,10]$  derece aralığında sapma için  $[-180,180]$  derece aralığında sınırlandırılmıştır. Daha sonra, robotun IMU

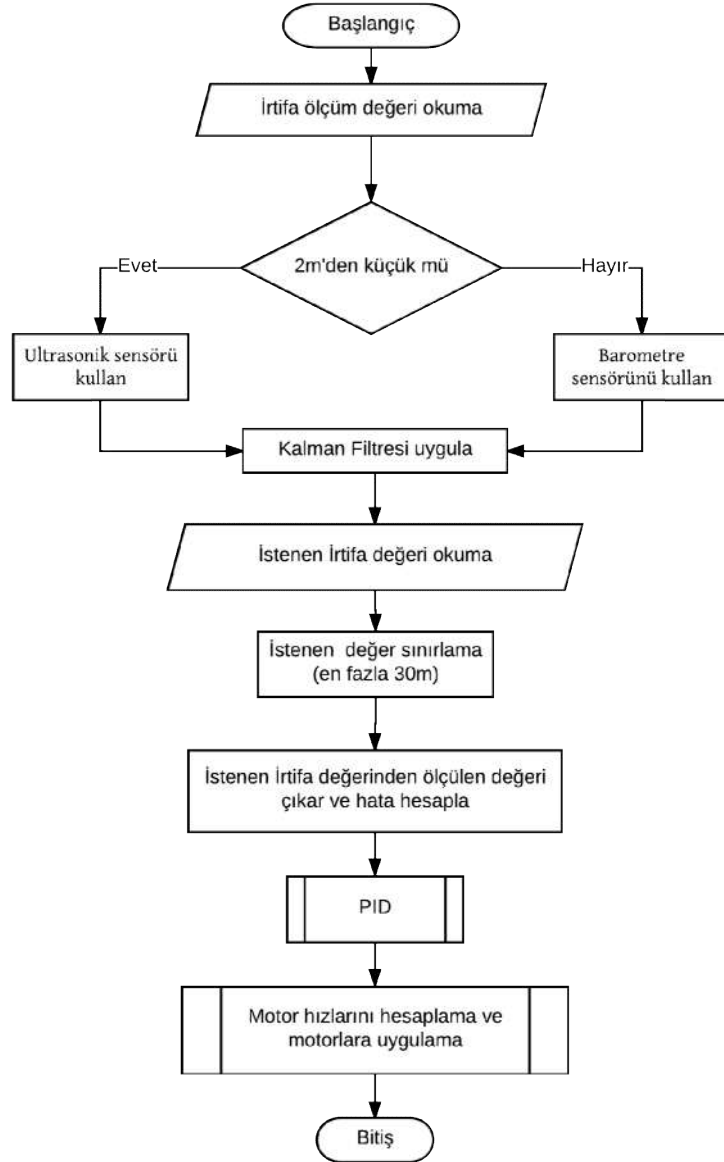
sensöründen ölçülen değerler kalman filtresine sokulur. Böylece eksenlerdeki açısıl dönme değerleri geri besleme olarak alınır. Kalman filtresi için TKJ-Electronics tarafından hazırlanmış “KalmanFilter” kütüphanesi kullanılmıştır. Bu kütüphanede üç eksendeki açısıl hareket ve ivme değerlerinin birleştirilmesi ile yapılmaktadır. Ölçülen değerlerin filtrelenmesi ile elde edilen değer istenen değerden çıkarılarak hata miktarı hesaplanmaktadır. Hata değeri PID hesaplamasına sokularak yuvarlanma, yunuslama veya sapma için çıkışı değeri hesaplanır. Bu çıkış değerleri motor hız hesaplamasına sokulduktan sonra robota uygulanarak uçuş denetimi sağlanmış olur.



Şekil 2.7: Durum Denetim Algoritması

### 2.2.1.2 İrtifa Denetimi

İrtifa denetim algoritması Şekil 2.8’de gösterilmiştir. Buna göre öncelikle robot üzerinden irtifa ölçümü yapılmaktadır. İrtifa ölçüm değeri 2 m’nin altında ise HC-SR04 ultrasonik mesafe ölçer daha hassas şekilde ölçüm alabildiği için öncelikle bu sensörün kullanımı tercih edilmektedir. Ölçüm değeri 2m’den fazla olduğu durumlarda barometre verisinden faydalanılmaktadır. Sonraki adımda irtifa ölçümü kalman filtresine sokularak gürültüsü azaltılmaktadır. Daha sonra irtifa verisi istenen irtifa değerinden çıkarılarak hata değeri hesaplanır ve PID denetim işlemi gerçekleştirilir.



Şekil 2.8: İrtifa Denetim Algoritması

## 2.2.2 Konumlandırma

### 2.2.2.1 GPS Tabanlı Konumlandırma Sistemi

Robotların GPS ile konumlandırmasında GPS modülünden alınan NMEA kodlarına bakılmaktadır. NMEA kodları içinde gelen GPGGA kodu içinden GPS sinyallerinin aktif olup olmadığı ve o anki enlem ve boylam değerleri alınmaktadır. Koordinat verileri enlem ve boylam bilgisi ile hesaplanan merkez koordinatlara göre her hava aracının koordinatları Haversine formülüne göre bakı yönü ve mesafesi hesaplanmaktadır. İki nokta arasındaki mesafeyi hesaplamak için kullanılan Haversine fonksiyonu yer kürenin yarı çapı 6372.8 km ve tüm enlem boylam değerleri radyan olmak üzere (2.8) eşitliği ile verilmiştir. Bu eşitliğe göre  $D_{enlem}$  radyan cinsinden enlemler arasındaki fark,  $D_{boylam}$  radyan cinsinden boylamlar arasındaki fark  $A$  ara hesaplama katsayısı ve  $D$  iki nokta arasındaki km cinsinden mesafedir. Ancak bu mesafe hesaplamasında açı bilgisi yoktur. Bu sebeple yön bilgisini hesaplamak için öncelikle  $P_1$  (2.9) ve  $P_2$  (2.10) açıları hesaplandıktan sonra bu açıların dört kadranlı ters tanjantı denklem (2.11)'de gösterildiği üzere uygulanır.

$$D_{enlem} = E_2 - E_1 \quad (2.5)$$

$$D_{boylam} = B_2 - B_1 \quad (2.6)$$

$$A = \left(\sin \frac{D_{enlem}}{2}\right)^2 + \cos E_1 * \cos E_2 * \left(\sin \frac{D_{boylam}}{2}\right)^2 \quad (2.7)$$

$$D = 6372.8 * \text{asin}(\sqrt{A}) \quad (2.8)$$

$$P_1 = \sin(D_{boylam}) * \cos(E_2) \quad (2.9)$$

$$P_2 = \cos(E_1) * \sin(E_1) - \sin(E_1) * \cos(E_2) * \cos(D_{boylam}) \quad (2.10)$$

$$P = \text{atan2}(P_1, P_2) \quad (2.11)$$

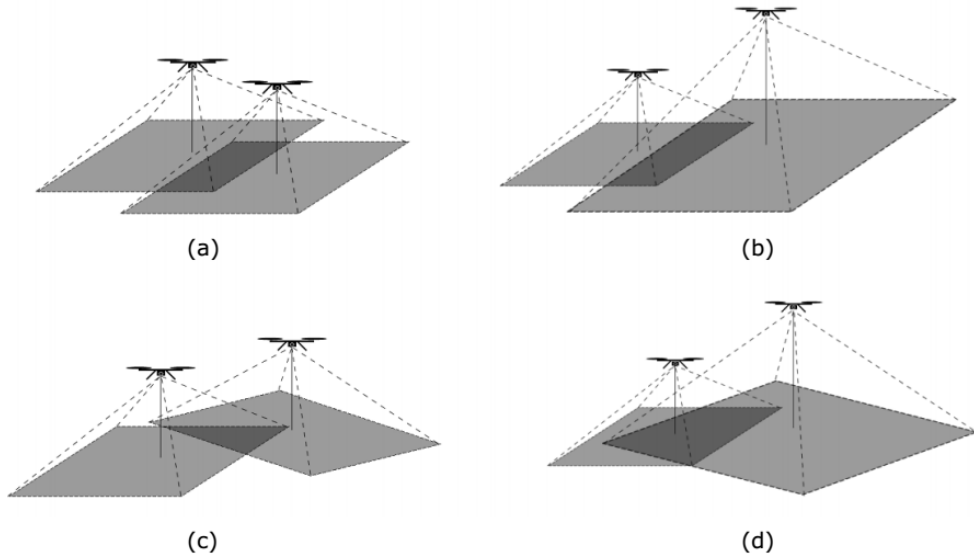
### 2.2.2.2 Görsel Eşleme Tabanlı Konumlandırma Sistemi

Bu bölümde robotların aldığı görüntüleri birbirlerine paylaşmaları ve bu görsellerdeki özniteliklerin eşlenmesi prensibi ile çalışan bir konumlandırma sistemi ve bu sistemin algoritması sunulmuştur. Mevcut konumlandırma sistemleri merkezi, ortam bağımlı, taşınamaz veya yüksek maliyetli olmaları sebebiyle sürü robotiğinin sağladığı avantajları kısıtlamaktadır. Bu sebeple çalışmamızda tasarladığımız konumlandırma sisteminin hem kapalı alanda hem de açık alanda çalışabilmesi, robotlar üzerinde bulunması, dağıtık olması ve kolaylıkla uygulanabilir olması hedeflenmiştir. Böylece sürü robotiğinin sağladığı gürbüzlük, esneklik ve ölçeklenebilirlik avantajlarının korunması amaçlanmıştır.

Sistem çoklu robotlar için tasarlanmış olup sadece robotların etkileşimini sağlamak için birbirlerine göre konumlarını bulmada kullanılmak üzere tasarlanmıştır. Konumlandırma algoritması hareket kabiliyetleri sebebiyle multikopter tipi hava araçları için tasarlanmıştır. Fakat küçük değişikliklerle diğer hava araçlarına da uygulanabilir niteliktedir. Konumlandırma işlemi yaparken model eşleme yönteminden esinlenilmiştir. Model eşlemede en temel gereksinim robota özgün bir parmak izi verisi elde etmektir. Uçan robotlara yerleştirilen kameralardan tekrarlayan veya düz bir desen görüntüsü alınmadığı sürece aynı noktada başka bir robot bulunamayacağından bu görüntüler eşsiz olacaktır. Bu sebeple fingerprint olarak kullanılabilir. Robotlarda tek kamera kullanılmıştır. Tekrarlayan veya düz desenli görüntü alınmasının önüne geçmek amacıyla kamera aşağı yönde bakacak şekilde yerleştirilmiştir.

Robot çiftleri birbirlerine yeterince yakın olduğunda ortaya çıkan temel durumlar Şekil 2.8'de gösterilmiştir. Bu temel durumlarda robot tarafından alınan görüntüler arasında kesişen ve kesişmeyen alanlar bulunmaktadır. Burada görüldüğü üzere kesişim alanları koyu renkle gösterilmiştir. Farklı durumlarda bu alanların büyüklüğü, konumu ve şekli değişmektedir. Tasarımı yapılan bu konumlandırma sisteminde de kesişim alanlarının bu özelliğinden faydalanılarak konum tahmini yapılacaktır. Robotlar aynı yükseklikte ve aynı yöne bakar durumdayken Şekil 2.8.a'da gösterildiği gibidir ve sadece robotların bulunduğu koordinatlar farklıdır. Bu durum çeviri (translation) olarak tanımlanan en basit durumdur. Şekil 2.8 b'de

gösterilen durumda robotlar arası yükseklik farkı vardır. Yükseklik farkı olduğunda iki görüntü arasındaki kesişim alanında bir ölçek (scale) kayması meydana gelmektedir. Şekil 2.8c’de gösterilen durumda robotların bulunduğu yükseklik aynı fakat baktıkları yönler farklıdır. Bu durum dönme (rotation) olarak adlandırılmaktadır. Gerçek bir uygulamada ise Şekil 2.8.d’de gösterildiği gibi çeviri, ölçekleme ve dönme durumlarının hepsi veya bir kısmının birleşimi söz konusu olabilir.



Şekil 2.9: Hava Araçları Arası Durumlar a) Çeviri b) Ölçekleme c) Dönme ve d) Birleşim

### 2.2.2.2.1 Görsel Öznitelik Eşleme Algoritması

Öznitelik eşleme işlemi algılama ve tanımlama olmak üzere iki kısımdan oluşmaktadır. Bir görüntü üzerindeki özel desenlere veya anahtar bölgelere öznitelik, bu bölgelerin bulunması için kullanılan yöntemlere de öznitelik algılama denmektedir. Aynı anahtar noktaları içeren iki görüntüdeki benzer bölgeleri belirlemek için kaynak görüntüdeki özniteliğin, hedef görüntüdeki özniteliklerden hangisine karşılık geldiğini bulmak için ise öznitelik tanımlama algoritmaları kullanılmaktadır. Çalışmada kullanılacak olan başlıca öznitelik algılama ve eşleme algoritması Şekil 2.15’de gösterilmiştir. Burada verilen bazı algoritmalar sadece algılama veya tanımlama işlemi yapmaktadır. SIFT ve SURF ise hem algılama hem tanımlama için kullanılmaktadır. ORB algoritması ise tek başına algılama veya tanımlama olarak kullanılabilirdiği gibi her ikisi yerine de kullanılmaktadır.



Tablo 2.2: Öznitelik Algılama ve Tanımlama Algoritmaları

Algoritma Adı	Kaynak	Öznitelik	
		Algılama	Tanımlama
Efficient Maximally Stable Extremal Region (MSER)	(Donoser ve diğ. 2006)	x	
Center Surround Extremas for Realtime Feature Detection and Matching (CENSURE)	(Agrawal ve diğ. 2008)	x	
Features From Accelerated Segment Test (FAST)	(Rosten ve diğ. 2010)	x	
Binary Robust Independent Elementary Features (BRISK),	(Calonder ve diğ. 2010)		x
Binary Robust Invariant Scalable Keypoints (BRISK)	(Leutenegger ve diğ. 2011)		x
Fast Retina Keypoint (FREAK)	(Alahi ve diğ. 2012)		x
Scale Invariant Feature Transform (SIFT)	(Lowe 2004)	x	x
Speed Up Robust Features (SURF)	(Bay ve diğ. 2006)	x	x
Oriented FAST and Rotated BRIEF (ORB)	(Rublee ve diğ. 2011)	x	x

Öznitelik algılama ve tanımlama algoritmaları işlerini tamamladığında tespit edilen nokta sayısı, noktaların koordinatları ve noktaları tanımlayan bir veri kümesi elde edilir. Nokta koordinatları ve nokta tanımlarından oluşan veri seti Random Sample Consensus (RANSAC) algoritması ile regresyona sokularak homografi matrisi olarak adlandırılan bir matris elde edilmektedir. Homografi matrisinin yapısı (2.12) eşitliğinde gösterilmiştir (Malis ve diğ. 2007) . Buradaki değerlerden  $x$  yönünde ve  $y$  yönünde çeviri değerleri  $t_x$  ve  $t_y$  sırasıyla denklem (2.13)'de ve denklem (2.14)'de verilmiştir. Bu denklemlere göre homografi matrisinden elde edilen  $h_{13}$  ve  $h_{23}$  değerleri  $k_{tx}$  ve  $k_{ty}$  çeviri katsayıları ile çarpılarak  $x$  ve  $y$  yönündeki çeviri miktarları elde edilmektedir. Denklem (2.15)'de  $r$  dönme değerinin nasıl hesaplandığı gösterilmektedir buna göre  $h_{12}$  ve  $h_{11}$  homografi matrisinden gelmektedir,  $k_r$  değeri dönme katsayısıdır. Denklem (2.16)'da ölçekleme değerinin hesaplanması gösterilmektedir. Burada  $s$  ölçekleme değeri  $k_s$  ölçekleme katsayısıdır. Tüm katsayılar kamera ölçümleri ve gerçek değerlere ölçeklenerek uygulanmaktadır.

$$H = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \quad (2.12)$$

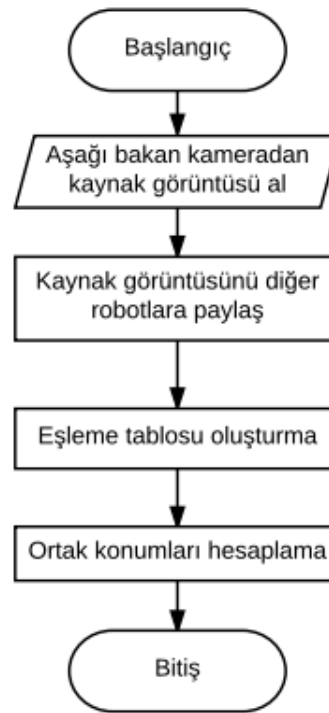
$$t_x = h_{13} * k_{tx} \quad (2.13)$$

$$t_y = h_{23} * k_{ty} \quad (2.14)$$

$$r = \arctan\left(\frac{h_{12}}{h_{11}}\right) * k_r \quad (2.15)$$

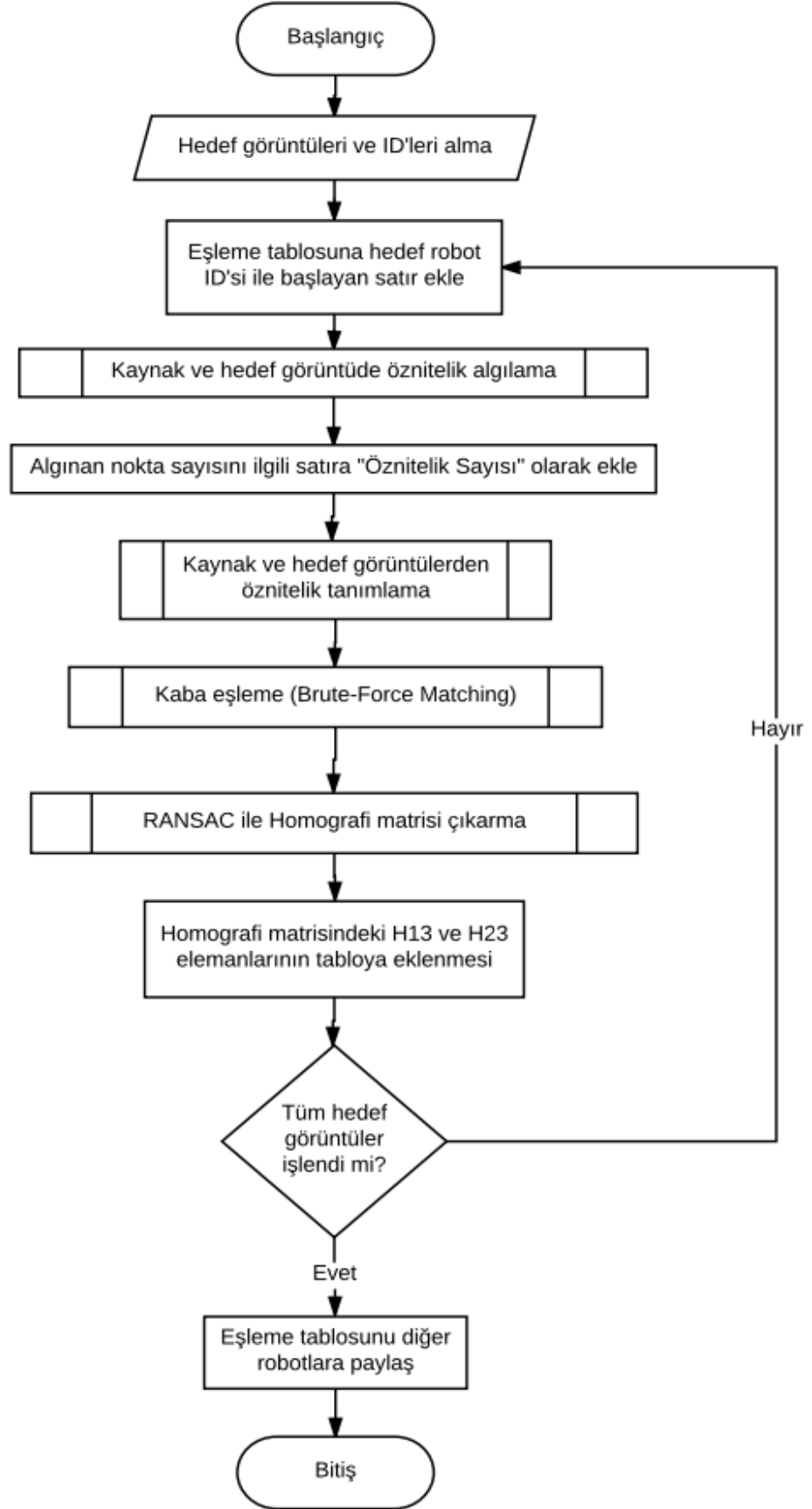
$$s = \left(\sqrt{h_{11}^2 + h_{12}^2}\right) * k_s \quad (2.16)$$

Model eşleme tipi konumlandırma sistemleri merkezidir ancak konumlandırma algoritması dağıtık olarak çalışmaktadır. Böylece sürü robotiğine daha uygun olması hedeflenmiştir. Şekil 2.10'da gösterilen konumlandırma algoritması çevrimdışı ve çevrimiçi kısımlardan oluşmaktadır. Çevrim dışı kısımlarda robot üzerinde yapılan hesaplamalar ve veri toplama işlemleri yapılmaktadır. Çevrimiçi kısımlarda ise robotlar arası veri paylaşımı yapılmaktadır.



Şekil 2.10: Görsel Konumlandırma Algoritması

Konumlandırma algoritmasına göre ilk işlem aşağı bakan kameradan görüntü almaktır. İkinci adımda her robottan alınan veri diğer robotlara paylaşılır. Bu sayede sürüdeki her robotta diğer robotlara ait fingerprint görüntüleri toplanmış olur. Üçüncü adımda eşleme tablosunun oluşturulması gelmektedir bu adımda uygulanan algoritma Şekil 2.10'da gösterilmiştir. Eşleme tablosu oluşturma algoritmasında her robot kendine ait parmak izi kaynak görüntü, diğerlerinden gelen parmak izleri hedef görüntü olarak almaktadır. Bu algoritma her robotta gelen her hedef görüntü için ayrıca hesaplanır. Eşleme tablosu oluşturma algoritmasının ikinci adımında kaynak görüntünün sahibine ait ID ile eşleme tablosunda bir satır oluşturulur. Sonraki adımda hedef ve kaynak görüntüler önce öznitelik algılama işlemi uygulanır. Burada kullanılan algoritma ile görüntü içindeki öznitelik noktaları tespit edilir. Tespit edilen nokta sayısı 10'dan az ise algoritmanın son adımına gidilerek değerler sıfır olarak girilir. Böylece iki robot arasında bir eşleme yapılamadığı bildirilmiş olur. Dördüncü adımda bulunan öznitelik sayısı eşleme tablosuna eklenir. Beşinci adımda tanımlama algoritması çalıştırılır. Böylece iki resim arasındaki benzer anahtar noktalar tanımlanmış olur. Altıncı adımda algılanan ve tanımlanan noktalardan oluşan veri kümesi kabaca eşlenir (Brute-force). Yedinci adımda eşlenen veri kümesinden homografi matrisi tahmin etmek için RANSAC algoritması kullanılır. Son adımda homografi matrisindeki  $h_{13}$  ve  $h_{23}$  elemanları eşleme tablosuna eklenir. Sonuçta üretilen eşleme tablosunun yapısı Şekil 2.11'de gösterilmiştir. Eşleme tablosunda robotun kendisiyle eşlemesi yapılmadığı için kendine ait ID ile girilmiş bir satır yoktur yani tablodaki bilgi satırı sayısı toplam robot sayısından bir eksiktir.



Şekil 2.11: Eşleme Tablosu Oluşturma Algoritması

Konulandırma algoritmasında dördüncü adımda eşleme tablosu robotlar arasında paylaşılır. Buradaki paylaşımda robot her satırdaki bilgiyi ilgili robota gönderir. Bu paylaşımın sebebi her robotun algıladığı öznitelik sayısının farklı olması ve RANSAC algoritmasının aynı veri kümesine uygulanması durumunda dahi içinde bulunan rastgele süreç sebebiyle farklı tahminler oluşturabilmesidir. Konulandırma algoritmasının son adımında ortak konum hesaplaması yapılmaktadır. Robot kendisine diğer robotlardan gelen bilgiyi denklem (2.17) ve denklem (2.18)'te uygulayarak diğer robotların konumlarını hesaplar. Buradaki  $x_s$  ve  $x_t$  sırasıyla kaynak eşleme matrisinin ve hedef eşleme matrisinin  $h_{13}$  terimidir.  $y_s$  ve  $y_t$  terimleri de yine aynı şekilde sırasıyla kaynak eşleme matrisinin ve hedef eşleme matrisinin  $h_{23}$  terimidir.  $f_s$  ve  $f_t$  terimleri ise sırasıyla kaynak ve hedef matrislerdeki öznitelik sayısıdır.

$$x = \frac{(x_s * f_s) + (x_t * f_t)}{f_s + f_t} \quad (2.17)$$

$$y = \frac{(y_s * f_s) + (y_t * f_t)}{f_s + f_t} \quad (2.18)$$

ID :	Öznitelik	:H13:H23
001:	5648	:308:207
002:	8740	:482:139
003:	9996	:108:129
004:	1642	:348:2122
005:	32648	:58 :27
⋮		
00n:	_____	:____:_____

Şekil 2.12: Eşleme Tablosu Örneği

### 2.2.2.3 Tek Gözlü Etkin Poz Takibi Sistemi

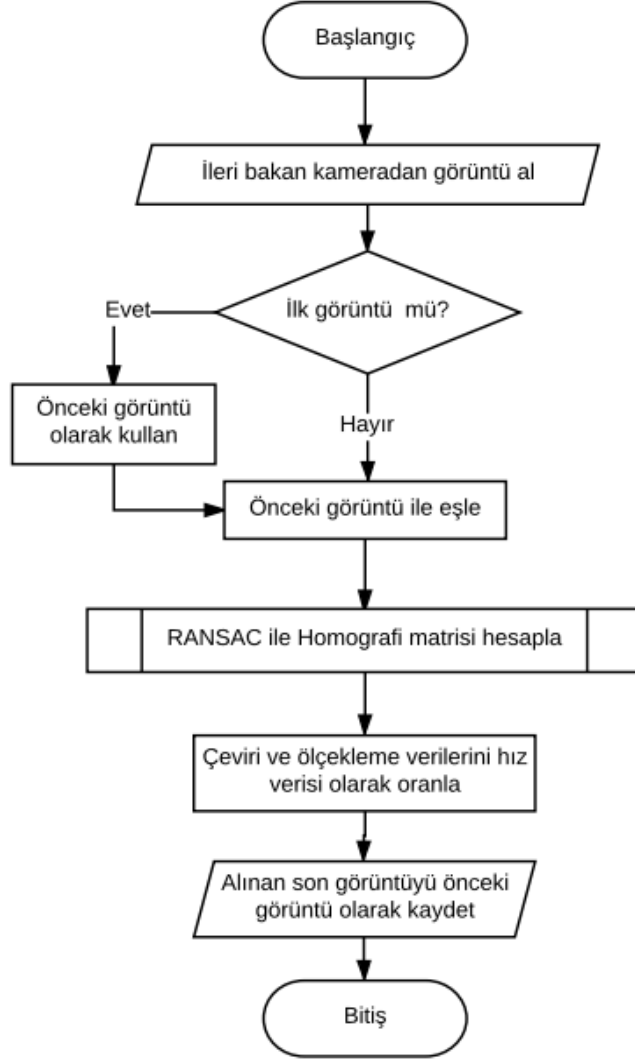
Etkin poz hesaplama (odometry) başlangıç konumuna göre hareket miktarını hesaplama ve gidilen yönü tahmin etme prensibine dayanmaktadır. Etkin poz hesaplama hata birikimli konumlandırma sistemidir. Bu sebeple son hatanın azaltılması amacıyla diğer göreceli algılama sistemleri ile birleştirilerek kullanılmaktadır.

Tasarlanan tek gözlü görsel etkin poz hesaplama sistemi hava aracının ön tarafında bulunan kameradan alınan görüntünün bir önceki görüntü ile eşlenmesi sonucu elde edilen çeviri ve ölçekleme verisidir. Çeviri verisi sağa, sola veya yukarı aşağı dönme verisi, ölçek verisi ise hız verisi üretimde kullanılmaktadır. Algoritma Şekil 2.13’de verilmiştir buna göre öncelikle kameradan görüntü alınır. Daha önce hiç görüntü alınmadıysa bu görüntü önceki görüntü olarak kaydedilir. Sonraki adımda alınan görüntü önceki görüntü ile eşlenir. Eşleme için öncelikle CENSURE algoritması ile öznitelik algılama ile anahtar noktalar tespit edilir ardından bu noktalara FREAK algoritması ile öznitelik tanımlama uygulanır. Eşleme sonrasında RANSAC yöntemi ile homografi matrisinin hesaplanması gerçekleştirilir. Homografi matrisi ile elde edilen ölçekleme (s) ve çeviri ( $t_x$  ve  $t_y$ ) verileri gerçek değerlere göre sırasıyla denklem (2.19), (2.20) ve (2.21)’de gösterildiği üzere oranlanır. Dm ilerleme miktarı, Dx x ekseninde (sola sağa dönme) dönme miktarı ve Dy y (aşağı yukarı) ekseninde dönme miktarıdır. Om, Ox ve Oy değerleri ise ilgili hareketin oran katsayılarını ifade etmektedir. Algoritmanın son adımında alınan son görüntü önceki görüntü olarak kaydedilir.

$$D_m = s * O_m \quad (2.19)$$

$$D_x = t_x * O_x \quad (2.20)$$

$$D_y = t_y * O_y \quad (2.21)$$

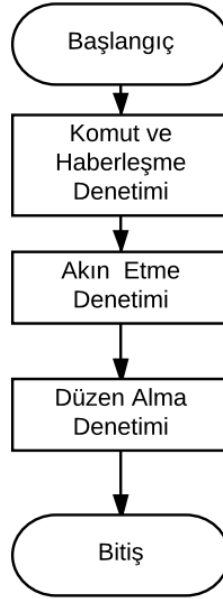


Şekil 2.13: Tek Gözlü Görsel Etkin Poz Takibi Algoritması

### 2.2.3 Çoklu Robot Denetim Sistemi

Çoklu robot denetiminden önce hava aracının uçuş, haberleşmeye başlama ve ilk konumlandırma işlemlerini gerçekleştirmiş olmalıdır. İlk konumlandırma haberleşme başladıktan sonra mevcut konumların robotlar arasında paylaşılması ve ağırlık merkezinin hesaplanıp robotun bağıl konumunun hesaplanması işlemidir. Bu işlem için öncelikle her robotun konumu toplanmakta bu konum verilerinin ortalaması alınıp merkez hesaplanmaktadır. Merkez noktası kartezyen koordinat sisteminde (0,0,0) noktası kabul edilir ve her robot kendi konumunu bu noktaya göre hesaplamaktadır. Desen düzeni için tanımlanan hedef noktaları da bu merkeze göre hesaplanmaktadır. Çoklu robot denetimi Şekil 2.14’te gösteriliği gibi komut ve

haberleşme denetimi, akın denetimi ve düzen denetimi olmak üzere üç kısımdan meydana gelmektedir.



Şekil 2.14: Çoklu Robot Denetim Algoritması

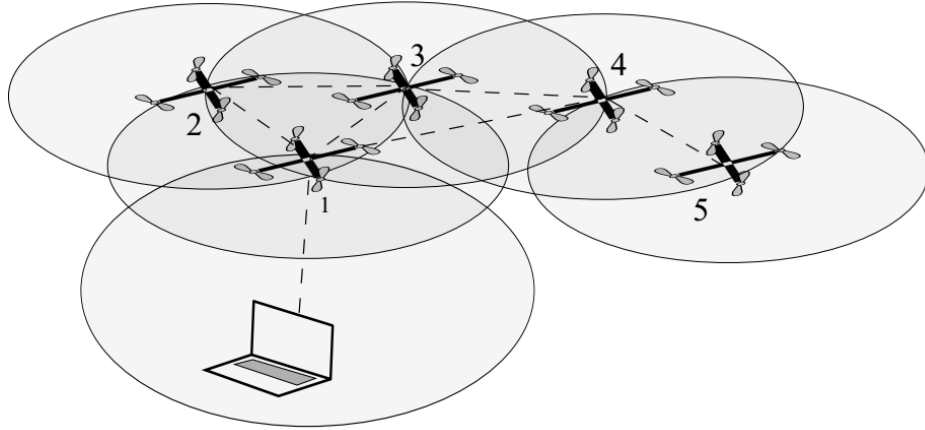
### 2.2.3.1 Haberleşme ve Komut Denetimi

Çoklu robot sisteminde haberleşme komut alma ve robotlar arası haberleşme olmak üzere iki tiptir. Robotların sürü davranışları esnasında birbirleri ile etkileşime geçmek için yaptıkları yayın mesajları robotlar arası haberleşmedir. Kullanıcı komutlarının sistemdeki tüm robotlara yayın (broadcast) olarak yayılması komut alma haberleşmesidir.

Kullanıcı komutları ve robotlar arası haberleşme için XBee modüllerinden meydana gelmiş bir KSA Şekil 2.15'deki gibi oluşturulmuştur. Sistemde kullanılan XBee KSA modülleri ZigBee (IEEE 802.15.4) protokolü ile çalışmaktadır. ZigBee protokolünün kullanılmasının ana sebebi mesh ağ yapısını desteklemesi ve mesh yapıyı destekleyen diğer kablosuz haberleşme sistemlerinden daha uzun menzilli olmasıdır. Mesh ağ yapısı ile merkezin kapsama alanı dışına çıkan Şekil 2.15'deki 4. ve 5. robotlar gibi elemanlar 3. ve 4. robotların mesaj yönlendirmesi sayesinde diğer robotlara ve kullanıcı bağlantısını sürdürebilmektedir. Kablosuz haberleşme protokollerinde ağ yönetimi merkezidir yani yönlendirme, ağa katılma ve adresleme



işlemleri için bir merkezi haberleşme birimine ihtiyaç vardır. Bu durumun XBee modüller içinde geçerlidir fakat modüller AT komutları ile tüm ağ yapılandırması ayarlanabilmektedir. ZigBee protokolünde ağ da yönetici, yönlendirici veya son düğüm haline olmak üzere üç tip ağ elemanı bulunmaktadır. Yönetici ağdaki tüm adresleme, paket yönlendirme gibi işlerin yönetildiği birim olup ve ağda bir adet bulunmalıdır. Yönlendirici düğümler ise ağın kapsama alanını genişletmek için merkezden uzak düğümlere paket yönlendirerek bağlantı sağlamak için kullanılmaktadır. Yönlendirici eleman sayısı istendiği kadar olabilmektedir. Son düğüm'ler ise sadece kendine ait veri alıp gönderen cihazlardır. Tasarlanan sistemde, robotlar içinden herhangi biri yönlendirici olarak atanmakta ve diğer robotların her biri yönlendirici düğüm olmaktadır. Eğer koordinatör robot ağdan düşerse ağda rastgele yeni bir robot koordinatör olarak atanmaktadır. Kumanda cihazında da bir XBee modülü bulunmaktadır ve yönlendirici olarak ayarlıdır. Bu sayede çoklu robot sistemi kullanıcı komutu alma haricinde hiçbir işlemin merkezi olmaması sağlanmaya çalışılmıştır.



Şekil 2.15: Robot Haberleşmesi ve KSA

Haberleşmede kullanılan mesajlar Şekil 2.16'da gösterilen komut mesajı ve Şekil 2.17'de gösterilen robotlar arası haberleşme mesajları halindedir. Komut denetimi için kullanılan mesaj kullanıcı ön eki, ağırlık merkezi ve düzen kodundan meydana gelmektedir. Komut mesajında ön ek olarak “#” karakteri, robotlar arası mesajlaşmada ise ön ek olarak “#” karakteri kullanılmıştır. Komut mesajında hedefin ağırlık merkezi ve alınacak düzen kodu bulunmaktadır. Düzen kodları Tablo 2.2’de verilmiştir. Robotlar arası mesajlaşmada ise robotun kendi koordinat bilgisini, hız bilgisini ve robotun ID’sini içermektedir.

Ön Ek	Ağırlık Merkezi	Düzen Kodu
#	37:29	C

Şekil 2.16: Komut Mesajı Örneği

Ön Ek	Koordinat	Hız	ID
\$	37:29	20:-5	A

Şekil 2.17: Robotlar Arası Mesaj Örneği

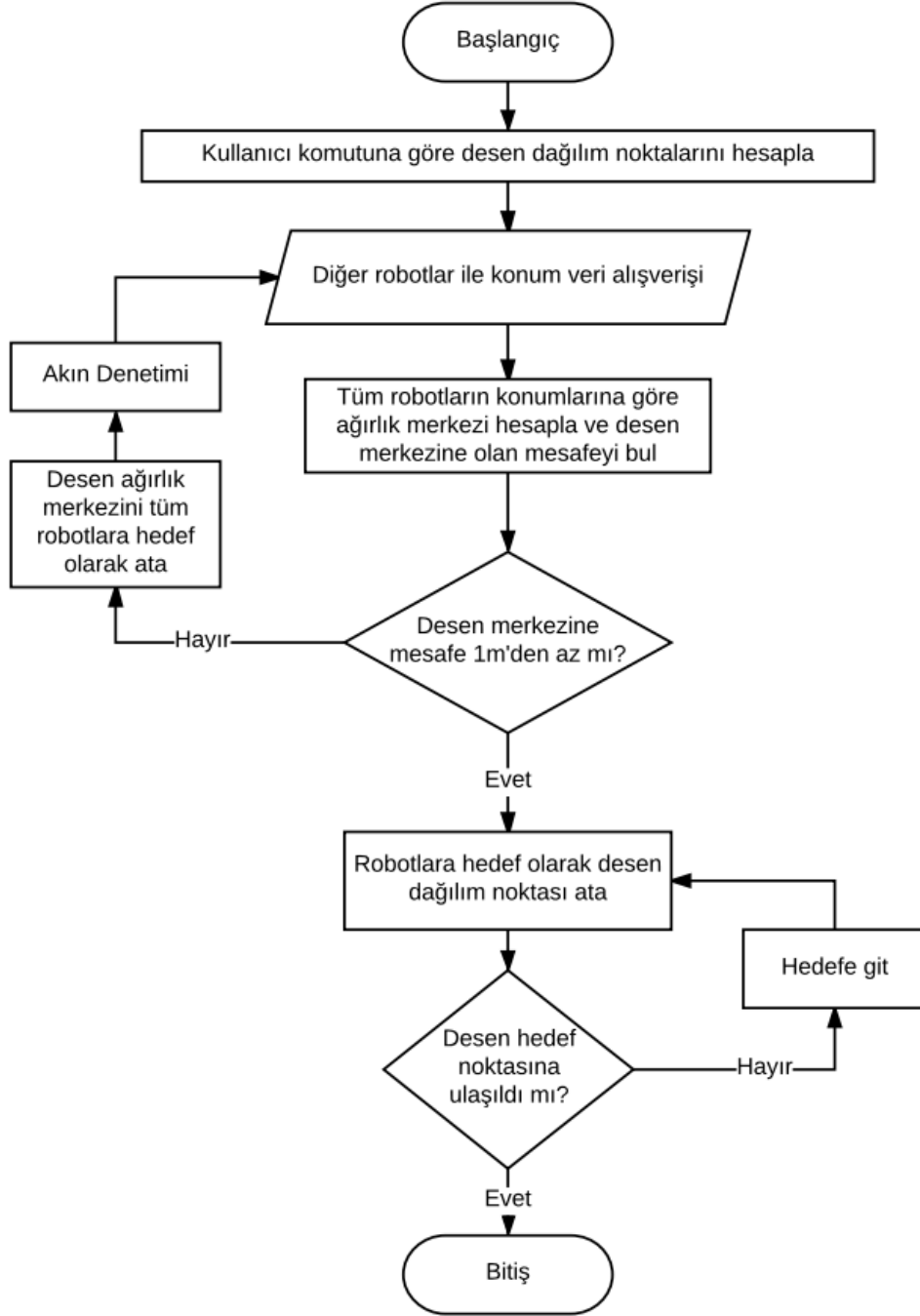
Tablo 2.3: Düzen Kodları

Kod	Düzen
C	çember
D	delta
S	spiral
K	kare

### 2.2.3.2 Desen Düzeni Alma ve Akın Denetimi

Düzen alma bir hedefe toplu şekilde ulaşıldıktan sonra dağılma veya düzen deseninin kısa aralıklarla yakın mesafelerde tanımlanması ile hareket boyunca dağılma şeklinde gerçekleştirilebilmektedir. Bu amaçla düzen deseninin nerede oluşturulacağı, kaç parçadan oluşacağı ve parçalar arasındaki aralığın ne kadar olacağı bilinmelidir. Desen düzeni alma ve akın etme denetimi için kullanılan genel algoritma Şekil 2.18’de gösterilmiştir. Bu algorithmada komut ve haberleşme denetimi ile alınan kullanıcı komutuna göre öncelikle robotların konumuna bağlı desen noktalarının üretilmesi gerekmektedir. Bu noktaların üretimi için çalışmada geometrik desen üretici kullanılmıştır ilgili fonksiyonların Python programlama dili ile yazılmış hali Tablo 2.3’de verilmiştir. Desen üretiminden sonra her robot KSA haberleşmesi ile kendi konum ve hız verisini yayın mesajı olarak paylaşır. Bu süreçte her robot diğer robotların tamamına ait konum ve hız verilerini depolar. Bu işlemin ardından her robotta tüm robotların konumlarına göre ağırlık merkezi hesaplanır. Ardından ağırlık merkezinin desen merkezine uzaklığı sınıranır. Eğer mesafe 1m’den az ise tüm robotlara hedef olarak desenin ağırlık merkezi hedef olarak gösterilmekte ve akın etme denetimi gerçekleştirilmektedir. Akın denetimi bölüm 2.2.3.2.1’de

sunulmuştur. Eğer tüm robotların ağırlık merkezinin desen merkezine olan mesafesi 1m'den az ise robotlara desen hedefi tanımlanır ve hedefe ulaşana kadar işlem devam eder. Desen noktalarının robotlara dağıtılmasında en yakın hedefi ulaşma ve önceden bir desen hedefine tanımlı olma şeklindedir. Bu iki durumun kıyaslaması ve akın denetimi yerine hedefe gitme denetimi ile ilgili uygulamalar ve bulgular bölüm 3'de verilmiştir.



Şekil 2.18: Desen Düzeni Alma ve Akın Etme Algoritması

Tablo 2.4: Geometrik Desen Üretici Fonksiyonları

Şekil	Denklem
Çember	<pre>def circleFormation(noqc, comx, comy, comz, radius):     pd=(2*np.pi)/noqc     px=np.ones(noqc)     py=np.ones(noqc)     pz=np.ones(noqc)*comz     r=radius     for i in range(0, noqc):         px[i]=(np.cos(i*pd)*r)+comx         py[i]=(np.sin(i*pd)*r)+comy     com=[np.mean(px), np.mean(py)]     return px, py, pz, com</pre>
Delta	<pre>def deltaFormation(noqc, comx, comy, comz, mag):     px=np.ones(noqc)     py=np.ones(noqc)     pz=np.ones(noqc)*comz     t=np.ones(noqc)     if noqc%2==0:         for i in range(0, noqc/2):             if i==0:                 t[i]=comx             else:                 t[i]=t[i-1]-mag         for i in range(noqc/2, noqc):             if i==noqc/2:                 t[i]=t[i-1]             else:                 t[i]=t[i-1]+mag     else:         for i in range(0, noqc/2+1):             if i==0:                 t[i]=comx             else:                 t[i]=t[i-1]-mag         for i in range(noqc/2+1, noqc):             t[i]=t[i-1]+mag      px=t     py=np.arange(-noqc/2, noqc/2)     px=px+comx     py=py+comy     com=[np.mean(px), np.mean(py)]     return px, py, pz, com</pre>
Spiral	<pre>def spiralFormation(noqc, comx, comy, comz, radius, sc):     pd=(2*np.pi)/noqc*sc     px=np.ones(noqc)     py=np.ones(noqc)     pz=np.ones(noqc)*comz     r=np.linspace(0, radius, noqc)     for i in range(0, noqc):         px[i]=(np.cos(i*pd)*r[i])+comx         py[i]=(np.sin(i*pd)*r[i])+comy         r=r+1     com=[np.mean(px), np.mean(py)]     return px, py, pz, com</pre>
Kare	<pre>def rectangleFormation(noqc, comx, comy, comz, d1, d2):     d1p=np.ceil((noqc/2.0)*d2/(d1+d2))     d2p=np.floor((noqc/2.0)*d1/(d1+d2))     x1=comx-d1/2.0     y1=comy-d2/2.0     x2=comx+d1/2.0     y2=comy-d2/2.0     x3=comx-d1/2     y3=comy+d2/2     x4=comx+d1/2     y4=comy+d2/2     px=np.array([])     py=np.array([])     pz=np.ones(noqc)*comz     if d1p!=0:         x1x2=np.linspace(x1, x2, d1p)         y1y2=np.ones(len(x1x2))*y1         px=np.append(px, x1x2)         py=np.append(py, y1y2)         x3x4=np.linspace(x3, x4, d1p)         y3y4=np.ones(len(x3x4))*y3         px=np.append(px, x3x4)         py=np.append(py, y3y4)     if d2p!=0:         i=np.abs(y1-y3)/d2p/1.5         y1y3=np.linspace(y1+i, y3-i, d2p)         x1x3=np.ones(len(y1y3))*x1         px=np.append(px, x1x3)         py=np.append(py, y1y3)         y2y4=np.linspace(y2+i, y4-i, d2p)         x2x4=np.ones(len(y2y4))*x2         px=np.append(px, x2x4)         py=np.append(py, y2y4)     print len(px), len(py)     com=[np.mean(px), np.mean(py)]     return px, py, pz, com</pre>

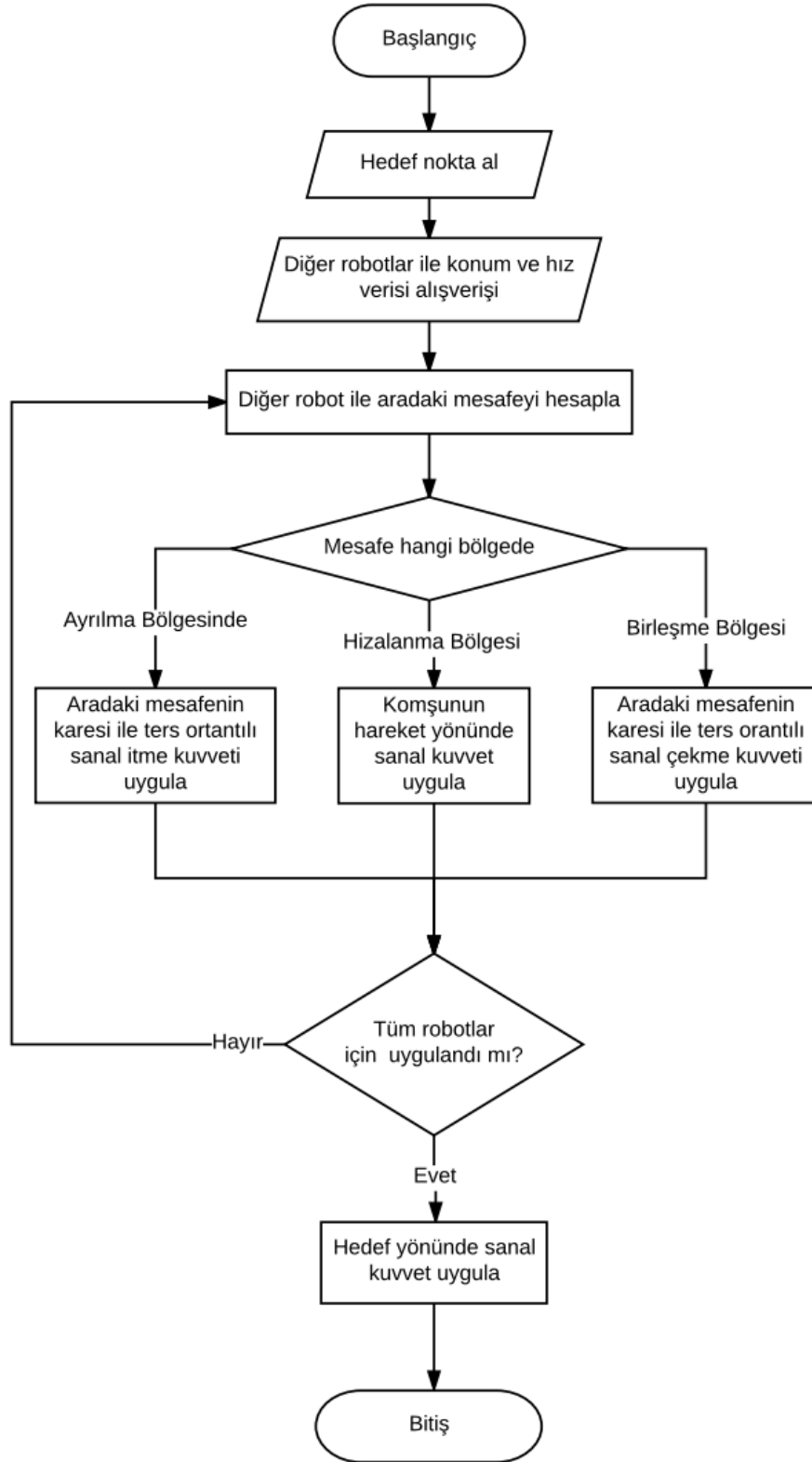
### 2.2.3.2.1 Akın Etme Denetimi ve Bulanık Mantık Yaklaşımı

Robotların akın etme denetiminde Reynolds tarafından belirlenmiş Boids algoritmasından faydalanılarak Şekil 2.19'daki akın denetimi algoritması türetilmiştir. Bu algoritmadaki boids sanal kuvvetleri uzaklığın karesiyle ters orantılı olarak değişecek şekilde uygulanmıştır. Kuvvetlerin toplamı her robota PID ile uygulanmaktadır. Algoritmada öncelikle desen düzen merkezine ait hedef nokta alınmaktadır. Çoklu robot sistemindeki tüm robotlar konum ve hız verilerini paylaşıldıktan sonra her robot diğer robotların kendisine göre uzaklığını hesaplar. Hesaplama sonucu diğer tüm robotların bulunduğu bölgeye göre itme, çekme veya hizalanma kuvvetleri uygulanır. Eğer robotlar belirli bölgeler içinde değil ise hesaba katılmaz. Tüm robotlara ait kuvvetler uygulandıktan sonra robota hedef doğru gitmesi için de bir kuvvet uygulanır. Böylece tüm robotlar hizalanmayı korurken desen düzeninin merkezine birlikte gidebilmektedir.

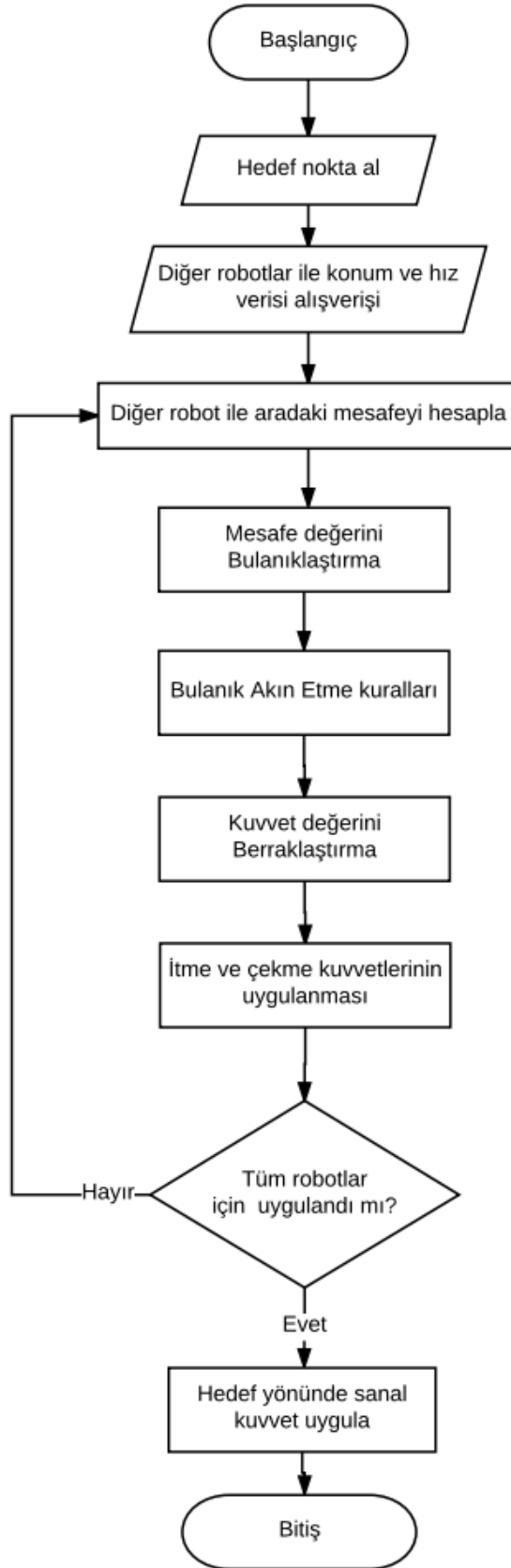
Akın etme algoritmasında tanımlı Boids bölgeleri kesin sınırlardan oluşmaktadır. Boids bölge sınırları robotların hareket yeteneği, sensörlerin algılama mesafesi, kullanıcının istediği sıklık, haberleşme kapsama alanı vb. parametrelere bağlı olarak özelleştirilebilmektedir. Bu bölgelerin kesin çizgiler halinde uygulanması robotlara çok ani kuvvetler uygulanmasına sebep olabilmektedir. Bu durum da özellikle hızlı hareket eden uçan robotlar gibi sistemler için olumsuz sonuçlar doğurabilmektedir. Bu sebeple Boids bölgelerinin bulanıklaştırılması ve uygulanan kuvvetin yön ve şiddetinin daha doğal hale çevrilmesi amaçlanarak Şekil 2.20'deki bulanık akın etme algoritması tasarlanmıştır. Bu algoritmada robotlar arası mesafe değeri öncelikle bulanıklaştırılmaktadır. Bu amaçla tasarlanan giriş üyelik fonksiyonu Şekil 2.21'de çıkış üyelik fonksiyonu ise Şekil 2.22'de gösterilmektedir. Bulanık mantık kuralları ise Tablo 2.5'de verilmiştir.

Tablo 2.5: Bulanık Akın Etme Kural Tablosu

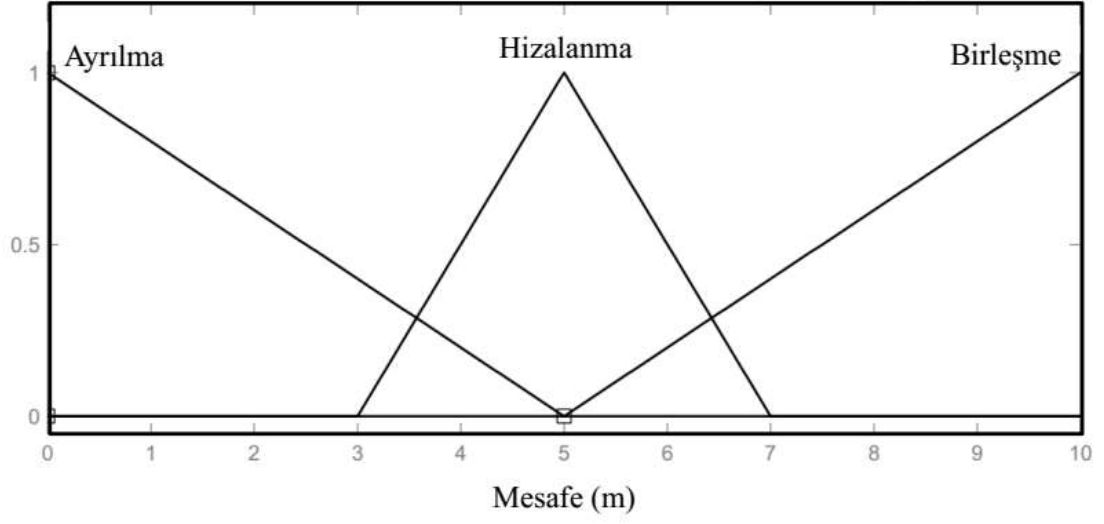
<b>Kurallar</b>
Eğer Mesafe Ayrılma ise Kuvvet İtme
Eğer Mesafe Hizalanma ise Kuvvet Hedefe Gitme
Eğer Mesafe Birleşme ise Kuvvet Çekme



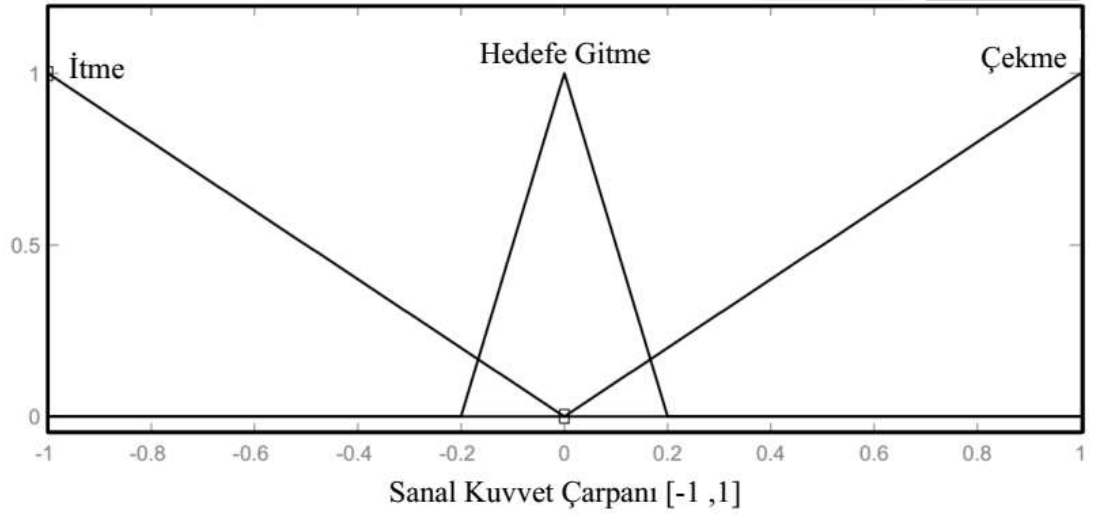
Şekil 2.19: Türetilmiş Boids Algoritması



Şekil 2.20: Bulanık Akın Etme Algoritması



Şekil 2.21:Giriş Üyelik Fonksiyonu



Şekil 2.22:Çıkış Üyelik Fonksiyonu



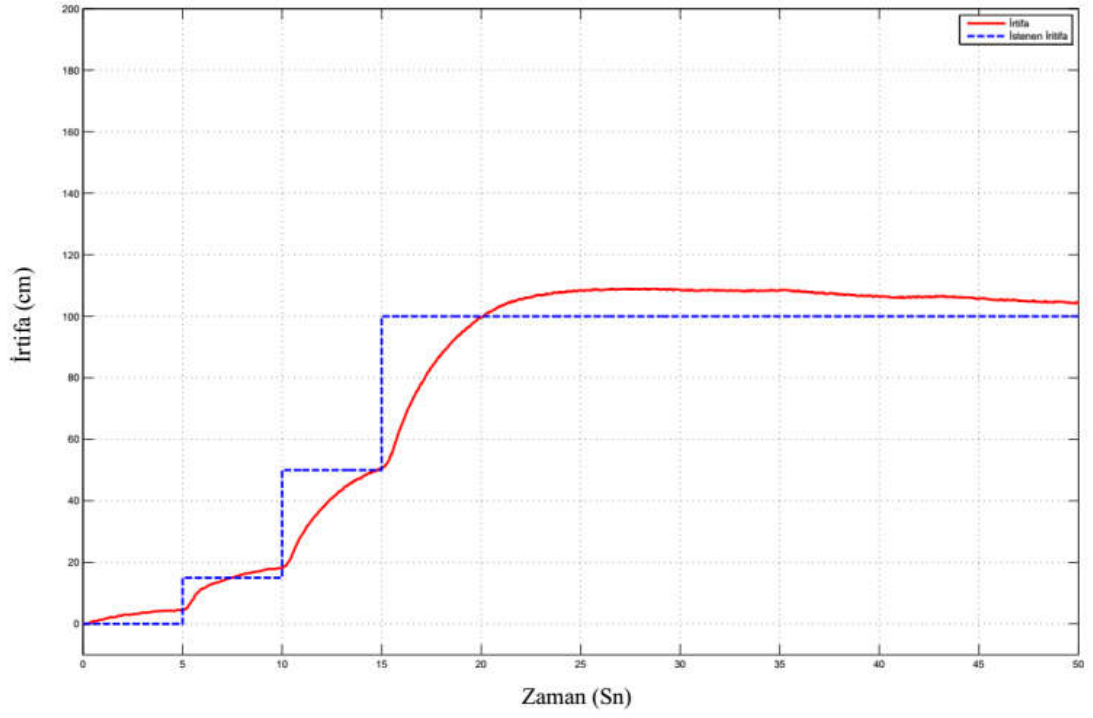
### 3. UYGULAMALAR VE BULGULAR

#### 3.1 Uçuş Denetim Uygulamaları

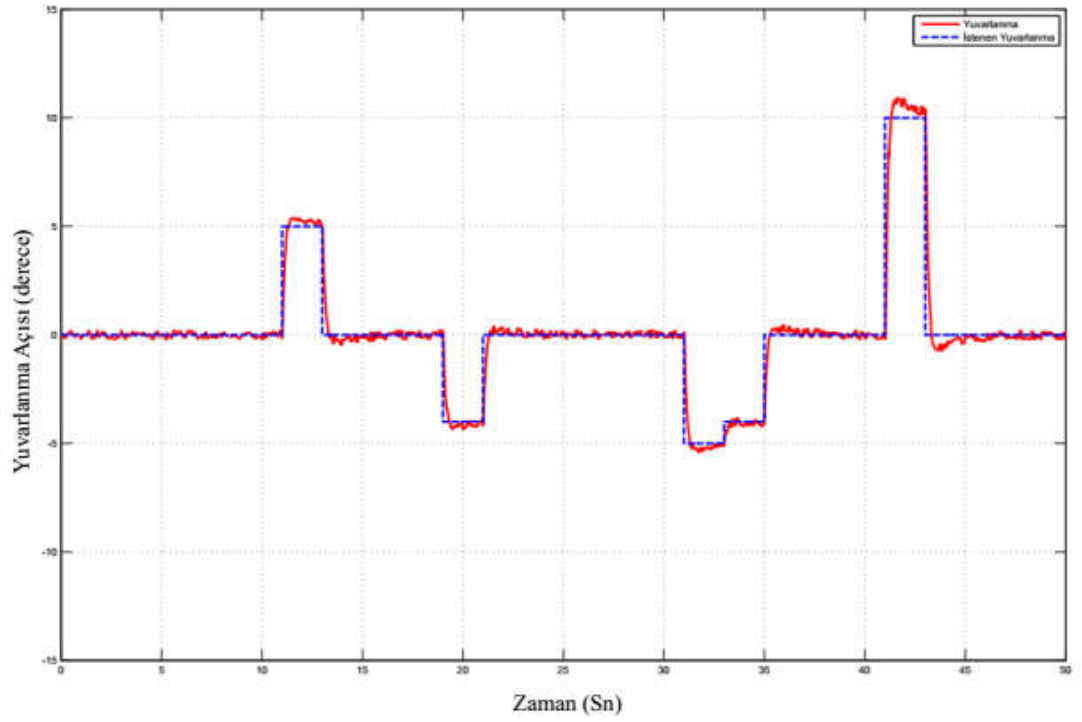
Altı rotorlu hava aracının kapalı ve açık alanda uçuşu esnasındaki görüntüleri Şekil 3.1’de gösterilmiştir. Deneme uçuşları esnasında alınan irtifa denetimi ölçümleri Şekil 3.2’de, durum denetim ölçümlerinden yuvarlanma Şekil 3.3’de ve yunuslama 3.4’de gösterilmiştir. Bu ölçümlere göre Durum denetiminde ivme ölçer ve jiroskop değerlerinde kalibrasyon yapılmış olmasına rağmen ve rastgele eksenlerde ortaya çıkan bir kayma görülmektedir. Bu kayma durumunun yazılımsal olarak önüne geçilemediği için 3m gibi hata payına sahip konumlandırma uygulaması yapılamamıştır.



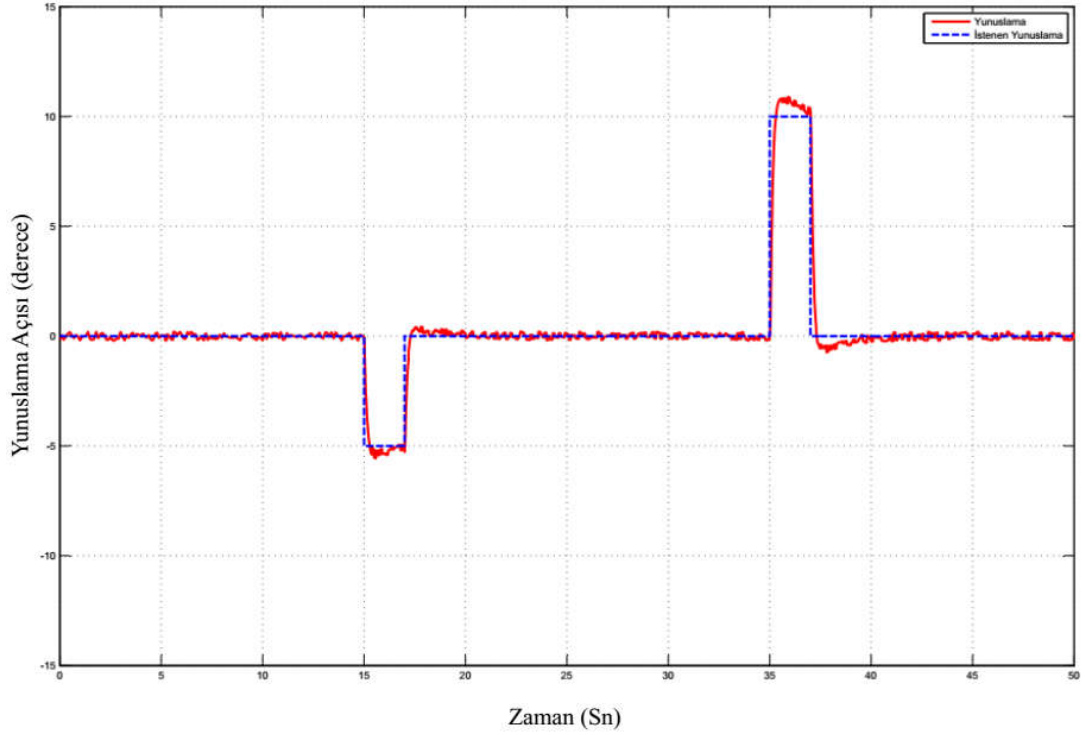
Şekil 3.1: Altı Rotorlu Hava Aracı Uçuş Görüntüleri



Şekil 3.2: İrtifa Denetim Ölçümü



Şekil 3.3: Yuvarlanma Denetim Ölçümü



Şekil 3.4: Yunuslama Denetim Ölçümü

## 3.2 Konumlandırma Uygulamaları

### 3.2.1 Öznitelik Eşleme Tabanlı Konumlandırma

Konumlandırma sisteminde kullanılacak en uygun öznitelik algılama ve tanımlama algoritmasının bulunması amacıyla uydu fotoğraflarında deneme yapılmıştır. Kullanılan uydu fotoğrafları renklidir ve büyüklüğü 750x750 pikseldir. Uydu fotoğrafları ile sistemin benzerini sağlamak amacıyla fotoğraflarla birlikte çeviri için koordinatlar, dönme için yön ve ölçekleme için yükseklik verisi alınmıştır. Denemelerde altı robottan oluşan bir sürü olduğu kabul edilmiştir. Her robot için farklı durumları ifade edecek bir çeviri, bir dönme, bir ölçekleme ve bir de bileşke görüntüsü alınmıştır. Deneyler sonucunda performans tabloları hazırlanmıştır. Bu tablolarda algoritma, zaman,  $C_X$ ,  $C_Y$ ,  $C_{ORT}$  ve skor sütunları vardır. Zaman sütununda verilen değerler her robotun konum hesaplaması için harcadığı sürenin saniye cinsinden toplamıdır. Üretilen konum verilerinin x ve y eksen bileşenlerinin gerçek koordinatların x ve y bileşenleriyle korelasyon katsayıları sırasıyla  $C_X$  ve  $C_Y$  sütunlarında verilmiştir. Konum verilerinin gerçek konum verisine ölçeklenmesi

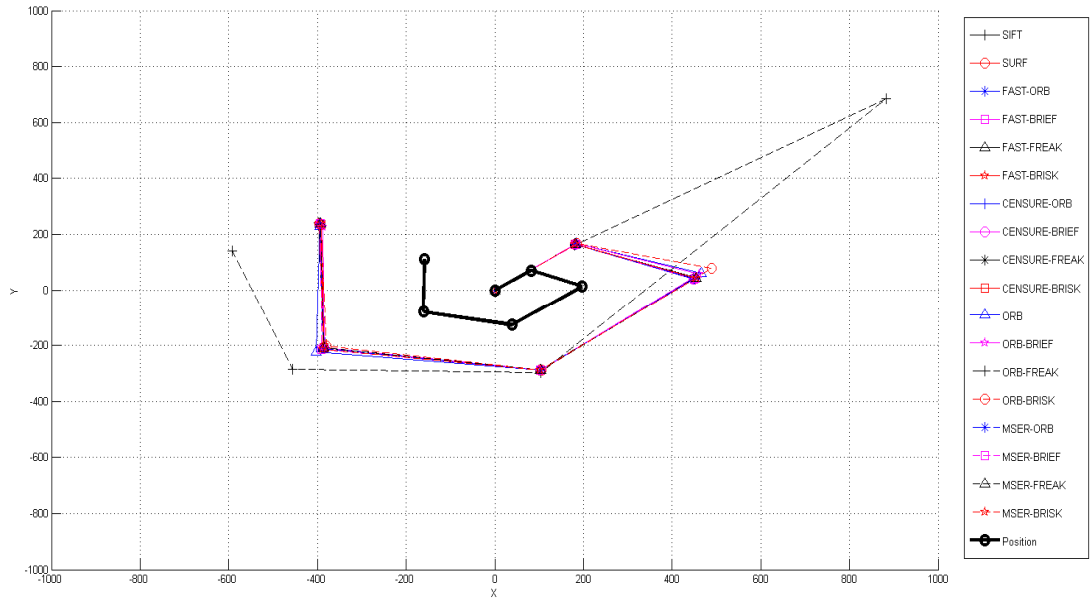
ilişki katsayısı arttıkça artacaktır. Bu sebeple çalışmada performans kıyaslaması için  $C_X$  ve  $C_Y$  değerlerinin ortalaması olan  $C_{ORT}$  başarımlar olarak kabul edilmiştir.  $S_T$  ve  $S_C$  sütunları sırasıyla zaman ve başarımlar değerlerine göre z skorundan üretilmiş t skorudur.

Yapılan deneylerde öncelikle çeviri, ölçekleme, dönme ve birleşim durumlarını kıyaslamak amacıyla Tablo 3.1 oluşturulmuştur. Bu tabloda bahsedilen durumlar için ortalama başarımlar, ortalama hesaplama süresi ve ortalama algılanan öznelik sayısı gösterilmiştir. Bu tabloya göre algılanan öznelik sayısı arttıkça çalışma süresi artmakta azaldıkça azalmaktadır. Ancak algılanan nokta sayısı ile başarımlar arasında bir ilişki görülmemiştir.

Tablo 3.1: Durum Karşılaştırması

Durum	Ortalama Çalışma Süresi	Ortalama Başarımlar	Ortalama Algılanan Öznelik Sayısı
Çeviri	40,2775	0,7009	47108
Ölçekleme	49,1387	0,5854	51223
Dönme	34,2564	0,3162	45881
Birleşim	36,1259	0,2700	46345

Çeviri durumunda her robot aynı yöne bakmaktadır ve hepsinin bulunduğu yükseklik aynıdır. Bu durum için yapılan deneylere göre algoritmaların birinci robot merkezde olacak şekilde ürettiği konumlar Şekil 3.1’de gösterilmiştir. Çeviri en temel durumdur dolayısıyla çoğu algoritma yakın değerler oluşturmuştur. Çeviri durumunda algoritmaların performans verileri Tablo 3.2’de verilmiştir. Bu durumda çoğu algoritmanın başarımlar değeri yüksektir dolayısıyla bu durum için skor değerini asıl etkileyen veri zamandır. Çeviri durumunda algoritmaların hesaplama sürelerinin ortalaması 40,2775 sn’dir ve başarımlar değerlerinin ortalaması ise 0,7009’dur.



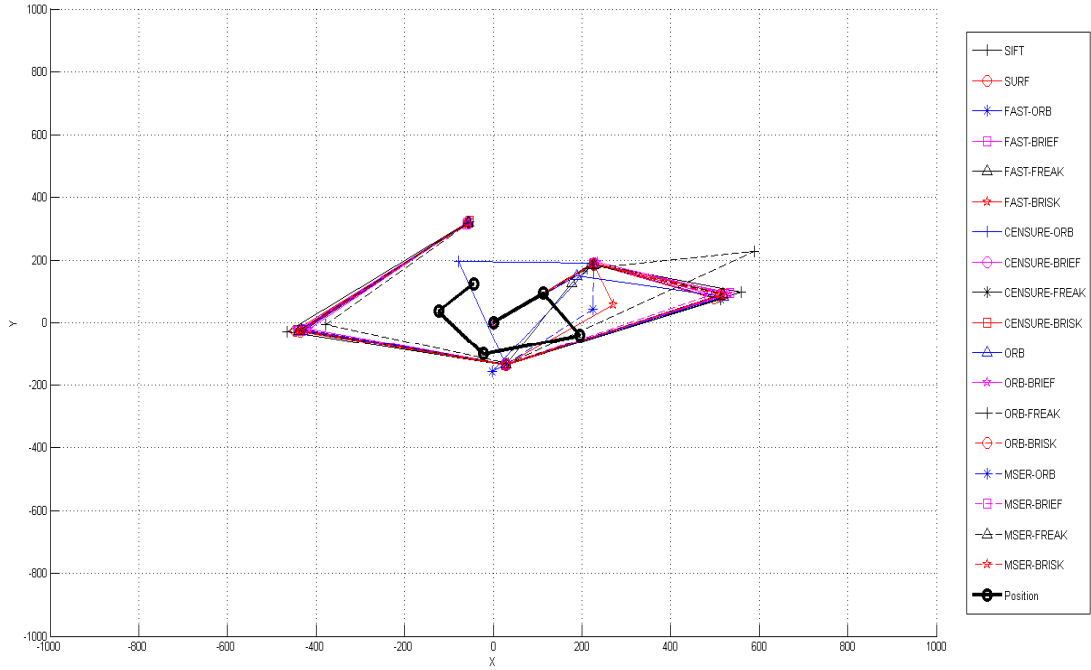
Şekil 3.5: Çeviri Durum Hesaplamaları

Tablo 3.2: Çeviri Duru Başarım Tablosu

Algoritma	T (s)	$S_T$	$C_X$	$C_Y$	$C_{ortlama}$	$S_C$
SIFT	52.2058	53.8154	0.5136	0.9137	0.7137	50.7477
SURF	72.0439	60.1609	0.6952	0.9109	0.8031	55.9699
FAST-ORB	43.8308	51.1366	0.6183	0.0597	0.3390	28.8688
FAST-BRIEF	44.5519	51.3672	0.7969	0.9968	0.8969	61.4464
FAST-FREAK	83.7236	63.8968	0.6592	0.8577	0.7585	53.3658
FAST-BRISK	125.4536	77.2447	0.8085	0.9803	0.8944	61.3028
CENSURE-ORB	9.3588	40.1102	0.7623	0.6449	0.7036	50.1594
CENSURE-BRIEF	9.1270	40.0361	0.7438	0.7962	0.7700	54.0381
CENSURE-FREAK	14.0391	41.6073	0.8071	0.7427	0.7749	54.3246
CENSURE-BRISK	47.4343	52.2892	0.7438	0.6642	0.7040	50.1818
ORB	9.8182	40.2572	0.6957	0.7807	0.7382	52.1820
ORB-BRIEF	8.3058	39.7734	0.8804	0.8053	0.8428	58.2911
ORB-FREAK	13.4039	41.4041	0.0960	0.4137	0.2549	23.9536
ORB-BRISK	47.1063	52.1843	0.7422	0.4092	0.5757	42.6912
MSER-ORB	26.5616	45.6128	0.7535	0.6984	0.7259	51.4649
MSER-BRIEF	26.1461	45.4799	0.7796	0.7649	0.7723	54.1702
MSER-FREAK	30.8726	46.9917	0.7279	0.4001	0.5640	42.0081
MSER-BRISK	61.0111	56.6319	0.6925	0.8747	0.7836	54.8334

Ölçekleme durumundaki denemelerde robotlar farklı yüksekliktedir. Bu durumda uydu fotoğraflarına göre birinci robot için üretilen sonuçlar Şekil 3.2’de gösterilmiştir. Ölçekleme durumunda algoritmaların hesaplama sürelerinin ortalaması 49,1387sn ve başarımlarının ortalaması ise 0,5854’tür. Tablo 3.3’te verilmiş olan ölçekleme durumunun verileri hem zaman hem başarımlar açısından çeviri durumuna göre daha düşüktür. Bu durumda resimlerin ölçeklerinin

değiştirilmesi algılanan nokta sayısını arttırdığı için algoritmaların hesaplama süreleri genel olarak artmıştır.

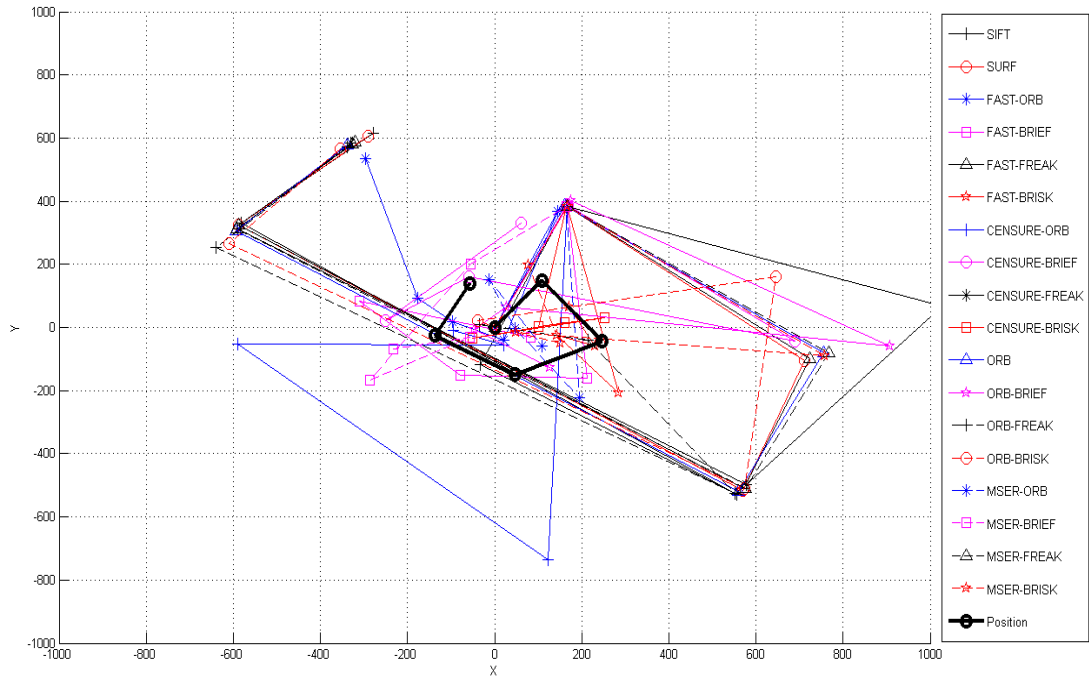


Şekil 3.6: Ölçekleme Durumu Hesaplamaları

Tablo 3.3: Ölçekleme Başarım Tablosu

Algoritma	T	$S_T$	$C_X$	$C_Y$	$C_{ortlama}$	$S_C$
SIFT	110.1076	62.6355	0.3683	0.7988	0.5835	49.9073
SURF	66.9732	53.6961	0.8503	0.8870	0.8687	63.8156
FAST-ORB	66.7329	53.6463	0.2388	0.1163	0.1775	30.1027
FAST-BRIEF	67.0834	53.7190	0.7770	0.8432	0.8101	60.9581
FAST-FREAK	126.5212	66.0371	0.6051	0.8067	0.7059	55.8757
FAST-BRISK	179.8778	77.0950	0.7931	0.3451	0.5691	49.2030
CENSURE-ORB	11.8809	42.2785	0.0168	0.5983	0.3075	36.4432
CENSURE-BRIEF	11.4243	42.1839	0.5974	0.6706	0.6340	52.3689
CENSURE-FREAK	17.2328	43.3877	0.8451	0.7291	0.7871	59.8355
CENSURE-BRISK	55.4727	51.3127	0.7978	0.7033	0.7506	58.0540
ORB	7.6313	41.3978	0.3520	0.7880	0.5700	49.2476
ORB-BRIEF	3.9291	40.6305	0.1839	0.1817	0.1828	30.3608
ORB-FREAK	6.1430	41.0894	0.7569	0.2085	0.4827	44.9865
ORB-BRISK	31.4470	46.3335	0.5867	0.4967	0.5417	47.8660
MSER-ORB	22.4119	44.4610	0.4614	0.5465	0.5040	46.0258
MSER-BRIEF	21.9578	44.3669	0.7139	0.6417	0.6778	54.5049
MSER-FREAK	25.2241	45.0438	0.4759	0.6186	0.5473	48.1373
MSER-BRISK	52.4452	50.6853	0.8251	0.8504	0.8377	62.3071

Dönme durumunda robotlar aynı yüksekliktedir ve yönleri farklıdır. Deneylede algoritmaların birinci robot merkezde olacak şekilde ürettiği konumlar Şekil 3.3’de gösterilmiştir. Dönme durumuna ait performans verileri Tablo 3.4’te verilmiştir. Bu verilere göre dönme durumunda algoritmaların hesaplama sürelerinin ortalaması 34,2564 sn ve başarımlarının ortalaması ise 0,3162’dir. Algoritmalar zaman bakımından ölçekleme ve çeviri durumlarına göre daha hızlıdır. Fakat bu durum bulunan öznelik sayısının az olmasından kaynaklanmaktadır. Başarımların ortalaması ise çeviri ve ölçeklemeden daha düşüktür. Bu durum da doğrudan bulunan öznelik sayısının az olmasından kaynaklanmaktadır. Dönme durumunda görüntülerin yönlerinin değiştirilmesi algoritmaların başarımlarında düşüş sağlamıştır. Algılanan öznelik sayıları düştüğü için hesaplama süresi kısalmıştır.



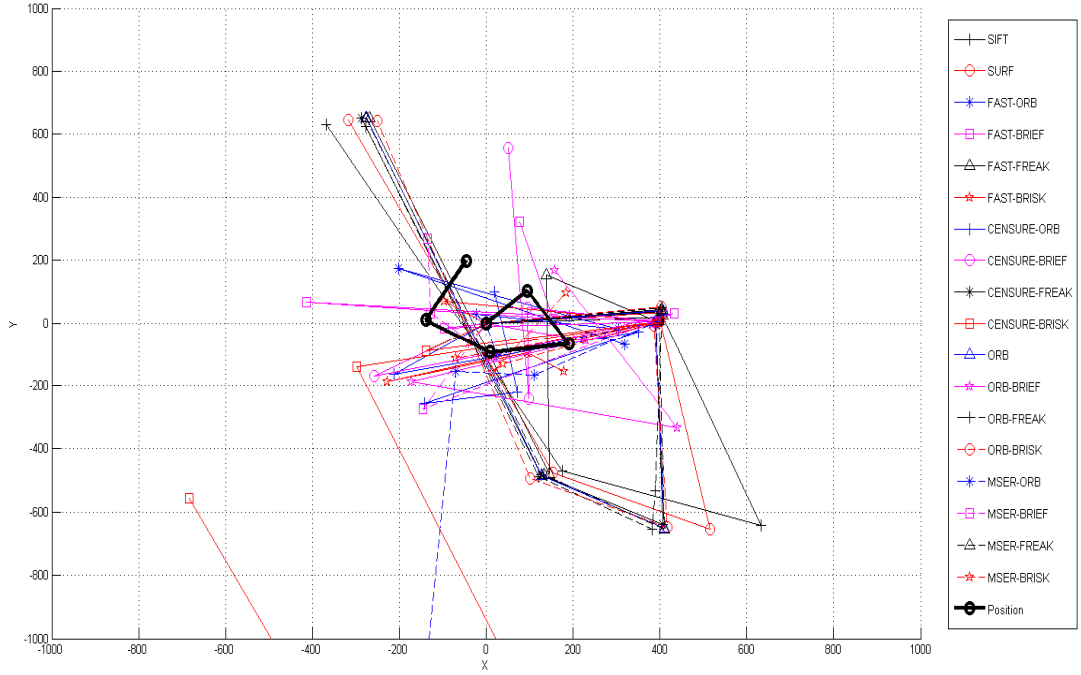
Şekil 3.7:Dönme Durumu Hesaplamaları

Tablo 3.4: Dönme Performans

Algoritma	T	S <sub>T</sub>	C <sub>X</sub>	C <sub>Y</sub>	C <sub>ortalama</sub>	S <sub>C</sub>
SIFT	68.6894	61.8471	0.2450	0.4397	0.3424	51.5578
SURF	72.2596	63.0755	0.5411	0.4732	0.5072	61.3558
FAST-ORB	37.7810	51.2127	0.1181	0.3825	0.2503	46.0804
FAST-BRIEF	40.6459	52.1984	0.1302	0.2369	0.1835	42.1128
FAST-FREAK	77.8763	65.0080	0.6379	0.3973	0.5176	61.9762
FAST-BRISK	104.3537	74.1179	0.4767	0.0727	0.2747	47.5331
CENSURE-ORB	11.0085	42.0012	0.2373	0.3307	0.2840	48.0863
CENSURE-BRIEF	7.4345	40.7716	0.1200	0.6005	0.3603	52.6202
CENSURE-FREAK	9.8347	41.5974	0.4762	0.6118	0.5440	63.5454
CENSURE-BRISK	31.0969	48.9129	0.5267	0.1301	0.3284	50.7239
ORB	7.0466	40.6381	0.1414	-0.0510	0.0452	33.8874
ORB-BRIEF	7.2012	40.6913	0.3162	0.0796	0.1979	42.9692
ORB-FREAK	7.1922	40.6882	0.3137	0.2750	0.2944	48.7035
ORB-BRISK	32.0677	49.2469	0.3176	0.1072	0.2124	43.8284
MSER-ORB	18.4147	44.5495	0.6937	0.4983	0.5960	66.6373
MSER-BRIEF	19.0928	44.7828	0.2274	0.1736	0.2005	43.1214
MSER-FREAK	20.1284	45.1391	0.5232	0.5411	0.5322	62.8429
MSER-BRISK	44.4912	53.5214	0.2870	-0.2460	0.0205	32.4180

Birleşim durumunda çeviri, ölçkleme ve dönme durumlarının hepsi birleştirilmiştir yani her robot farklı yönde ve farklı yüksekliktedir. Birleşim durumunda algoritmaların birinci robot merkezde olacak şekilde ürettiği konumlar Şekil 3.4'te gösterilmiştir. Tablo 3.5'teki performans verilerine göre hesaplanan işlem sürelerinin ortalaması 36,1259sn ve başarımlarının ortalaması ise 0,2700'dir. Birleşim durumunda tüm durumlardaki kötü koşullar birleştiğinden tahminlerdeki başarımlar en düşük seviyededir.





Şekil 3.8: Birleşim Durumu Hesaplamaları

Tablo 3.5: Birleşim Durumu Performans Tablosu

Algoritma	T	$S_T$	$C_X$	$C_Y$	$C_{ortlama}$	$S_C$
SIFT	72.7769	61.4654	0.2416	0.4813	0.3614	56.7649
SURF	73.7615	61.7734	0.2158	0.4349	0.3253	54.0952
FAST-ORB	46.5906	53.2736	0.2359	0.1925	0.2142	45.8765
FAST-BRIEF	46.0478	53.1038	0.0145	0.3303	0.1724	42.7832
FAST-FREAK	83.3889	64.7851	0.5209	0.6083	0.5646	71.7939
FAST-BRISK	116.5835	75.1692	0.2705	0.0211	0.1458	40.8173
CENSURE-ORB	7.7324	41.1178	0.2689	0.3591	0.3140	53.2586
CENSURE-BRIEF	7.7093	41.1106	0.0249	0.4114	0.2182	46.1690
CENSURE-FREAK	9.5393	41.6831	0.2549	0.3815	0.3182	53.5679
CENSURE-BRISK	32.6723	48.9196	0.3367	-0.0220	0.1573	41.6671
ORB	6.6584	40.7818	0.3855	0.4563	0.4209	61.1623
ORB-BRIEF	6.8175	40.8316	0.1459	0.3603	0.2531	48.7511
ORB-FREAK	8.8239	41.4593	0.3110	0.4524	0.3817	58.2647
ORB-BRISK	30.2646	48.1665	0.0619	0.4833	0.2726	50.1972
MSER-ORB	19.5541	44.8159	0.2517	-0.2815	-0.0149	28.9297
MSER-BRIEF	18.3471	44.4384	0.0338	0.1468	0.0903	36.7123
MSER-FREAK	20.9187	45.2428	0.2919	0.4996	0.3958	59.3052
MSER-BRISK	42.0785	51.8621	0.1200	0.4168	0.2684	49.8839

Çeviri, ölçekleme, dönme ve birleşim durumlarına ait sıralı performans tablosu Tablo 3.6’da verilmiştir. Performans hesaplamasında daha önceki tablolarda hesaplanan başarımların zaman skoruna bölünmüştür. Bu değerlerin ortalaması algoritma kombinasyonunun skoru olarak kabul edilmiştir. Tablo 3.6’ya göre en

performanslı algoritma kombinasyonu CENSURE-FREAK'dir. En performanslı öznitelik algılama algoritmaları CENSURE ve ORB'dir. Öznitelik tanımlama algoritmaları arasında en başarılıları FREAK ve BRIEF'dir. Patentli algoritmalar olan SIFT ve SURF algoritmalarının yüksek başarımlı değerleri olmasına rağmen işledikleri nokta sayısı fazla olduğundan yavaş kaldıkları görülmüştür. Benzer şekilde FAST algoritmasının da diğerlerine göre yavaş kaldığı görülmüştür. Bu algoritmalar yavaş çalışmaları için sürü robot konumlandırması için düşük performans puanı almışlardır.

Tablo 3.6: Performans Karşılaştırma Tablosu

Algoritma	Çeviri $S_C/S_T$	Ölçekleme $S_C/S_T$	Dönme $S_C/S_T$	Birleşim $S_C/S_T$	Performans
CENSURE-FREAK	1.3057	1.3791	1.5276	1.2851	1.3744
CENSURE-BRIEF	1.3497	1.2414	1.2906	1.1230	1.2512
ORB	1.2962	1.1896	0.8339	1.4997	1.2049
MSER-FREAK	0.8939	1.0687	1.3922	1.3108	1.1664
CENSURE-ORB	1.2505	0.8620	1.1449	1.2953	1.1382
ORB-BRIEF	1.4656	0.7472	1.0560	1.1940	1.1157
MSER-ORB	1.1283	1.0352	1.4958	0.6455	1.0762
ORB-FREAK	0.5785	1.0948	1.1970	1.4053	1.0689
MSER-BRIEF	1.1911	1.2285	0.9629	0.8261	1.0522
CENSURE-BRISK	0.9597	1.1314	1.0370	0.8517	0.9950
SURF	0.9303	1.1885	0.9727	0.8757	0.9918
FAST-BRIEF	1.1962	1.1348	0.8068	0.8057	0.9859
ORB-BRISK	0.8181	1.0331	0.8900	1.0422	0.9458
MSER-BRISK	0.9682	1.2293	0.6057	0.9619	0.9413
FAST-FREAK	0.8352	0.8461	0.9534	1.1082	0.9357
SIFT	0.9430	0.7968	0.8336	0.9235	0.8742
FAST-ORB	0.5645	0.5611	0.8998	0.8611	0.7217
FAST-BRISK	0.7936	0.6382	0.6413	0.5430	0.6540

### 3.2.2 Tek Gözlü Etkin Poz Takibi Tabanlı Konumlandırma Uygulaması

Bu uygulamada açık alanda toplanan görseller ve GPS verisinden oluşan bir veri tabanı hazırlanmıştır alınan veriler üzerinde tasarlanan görsel etkin poz hesaplama algoritmasında CENSURE-FREAK çalıştırılmıştır. Açık alan için elde edilen etkin poz hesaplama sonucu Şekil 3.9’da verilmiştir. Bu hesaplama esnasında elde edilen bazı ara sonuçlar Şekil 3.10’da sunulmuştur. Buna göre gerçek toplam yol 265m iken hesaplanan toplam yol 302m olup toplam yol açısından hata yüzdesi %13.96’dır. Bitiş noktaları arasındaki uzaklık 16.65 m olarak hesaplanmıştır.



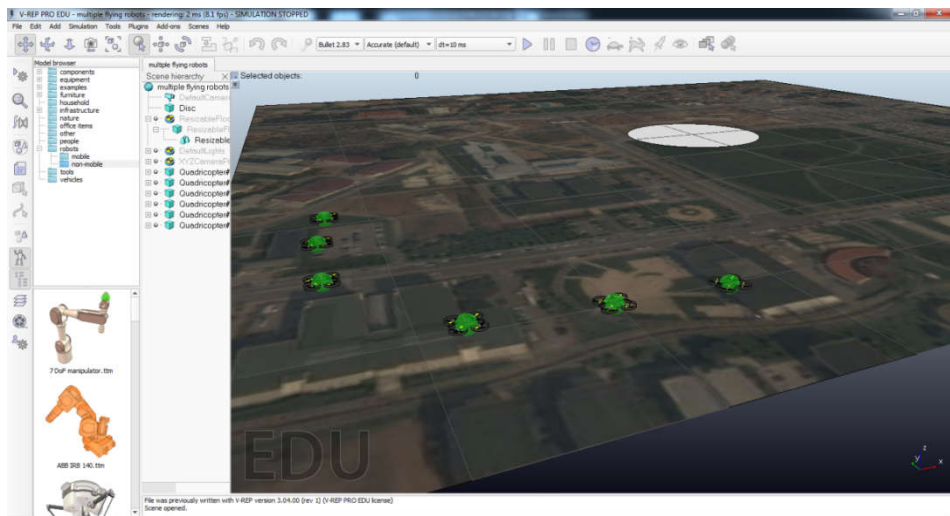
Şekil 3.9: Tek Gözlü Görsel Etkin Poz Hesaplama Sonucu



Şekil 3.10: Görsel Etkin Poz Hesaplama Uygulama Ara Sonuçları

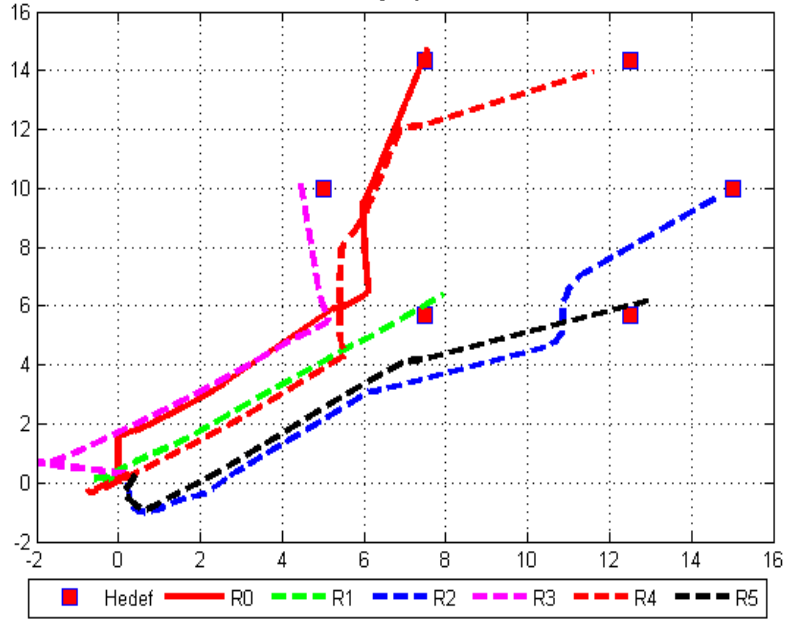
### 3.3 Çoklu Robot Denetim Uygulamaları

Çoklu robot denetimi uygulaması Şekil 3.11’de ekran görüntüsü verilen V-REP (Rohmer ve diğ. 2013) benzetim ortamında yapılmıştır. Altı robot dört rotorlu hava aracı olarak alta bakan ve ileri yönde bakan kameralar eklenerek modellenmiştir. Robotların çoklu denetimi, konumlandırması ve veri günlüklemesi için Python programlama dili kullanılmıştır. V-REP benzetim programına uzaktan erişim sağlama amacıyla Python–Remote API Kütüphanesi kullanılmıştır. Öznitelik eşleme tabanlı konumlandırma işlemi için ise OpenCV 2.7 kütüphanesinin Python diline uygun kütüphanesi kullanılmıştır. Benzetim uygulamasında uydu fotoğrafı yer deseni olarak kullanılmış ve öznitelik eşleme tabanlı konumlandırma sistemi destek amaçlı kullanılmıştır. Çoklu robot benzetiminde robotların çember şeklindeki bir desen şeklini almak için belirli hedeflere gitmeleri işbirliği ve yarışma olmak üzere iki ana grupta ele alınmıştır. İşbirliği açısından çoklu robot çalışması bireysel, işbirlikli ve bulanık işbirlikli olarak üç şekilde incelenmiştir. Yarışma kısmında ise robotların hedef atamasının en yakın noktaya yarışarak kapması ve önceden atanmış olması şeklinde iki şekilde incelenmiştir. Bu kısımda verilen çalışmalar sırasıyla bireysel yarışan, bireysel önceden atanmış, işbirlikli yarışan, işbirlikli önceden atanmış, bulanık işbirlikli yarışan ve bulanık işbirlikli önceden atanmış şeklindedir. Son olarak bu uygulamaların haberleşme tekrar sayıları ve işlem tamamlama süreleri verilmiştir.

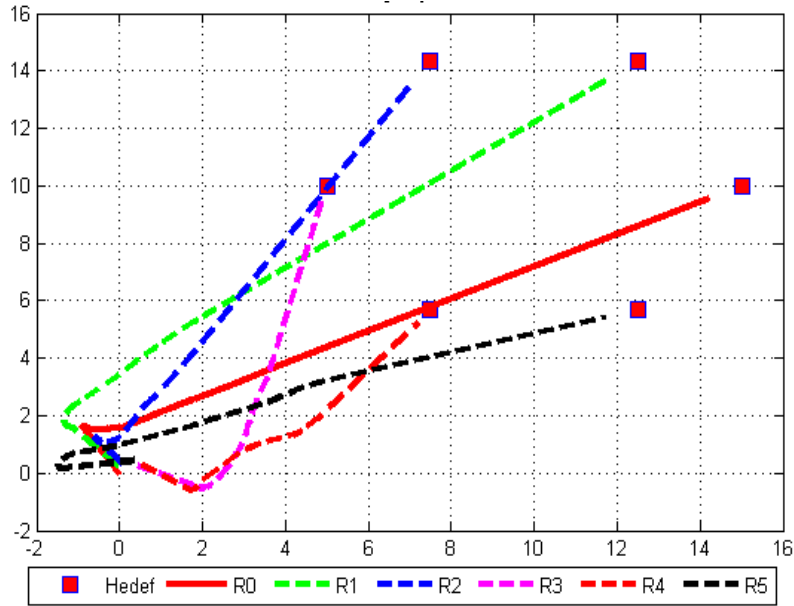


Şekil 3.11: V-REP Benzetim Ortamı

Robotların bireysel olarak çalışması esnasında akın algoritmasının birleşme ve hizalanma kuralları kullanılmadan sadece çarpışma önleme için ayrılma kısmının çalıştırılması ve robotların hedeflere gitmesi gerçekleşmektedir. Robotların bireysel olarak yarışarak hedefe gitme uygulaması sonucu robotların izlediği yollar Şekil 3.12’de gösterilmiştir. Robotların bireysel olarak önceden atanmış hedefe gitme uygulamasında robotların izlediği yollar Şekil 3.13’de gösterilmiştir.

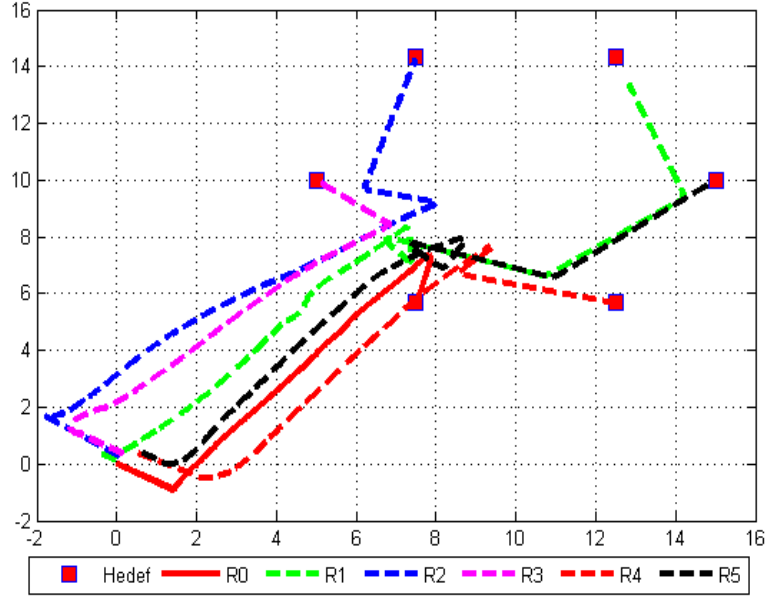


Şekil 3.12: Bireysel Robotların Yarışarak Hedeflere Ulaşması

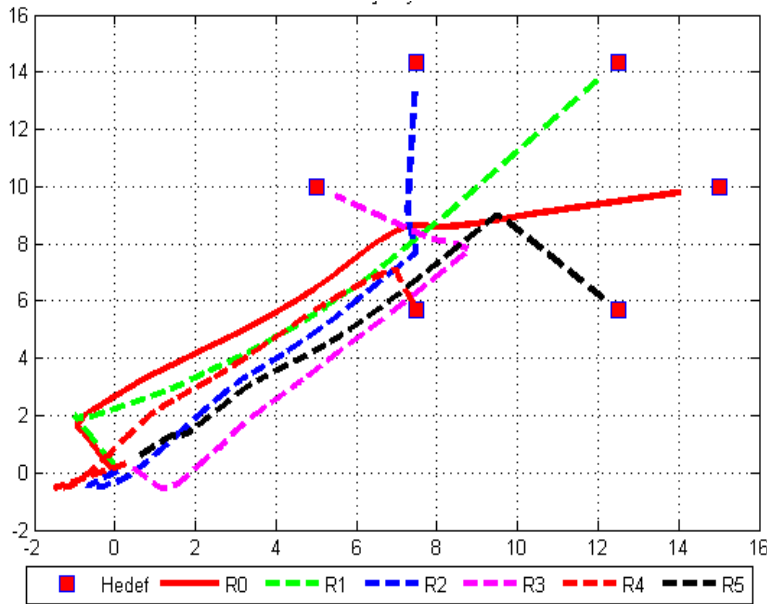


Şekil 3.13: Bireysel Robotların Önceden Atanmış Hedeflere Ulaşması

Robotların işbirlikli olarak çalışması esnasında akın algoritmasının desen merkezine kadar çalıştırılıp daha sonra robotların hedeflere gitmesi gerçekleşmektedir. Robotların işbirlikli olarak yarışarak hedefe gitme uygulaması sonucu robotların izlediği yollar Şekil 3.14’de gösterilmiştir. Robotların işbirlikli olarak önceden atanmış hedefe gitme uygulamasında robotların izlediği yollar Şekil 3.15’de gösterilmiştir.

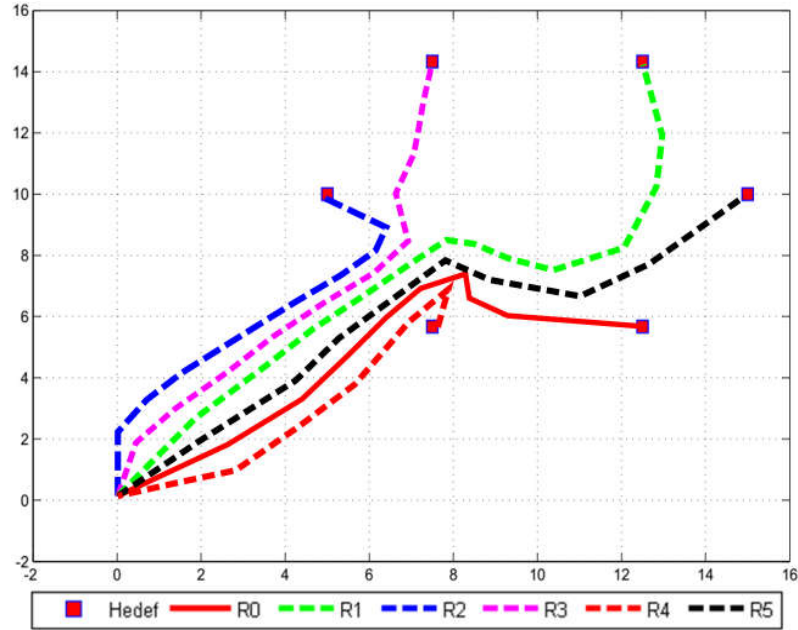


Şekil 3.14:İşbirlikli Robotların Yarışarak Hedeflere Ulaşması

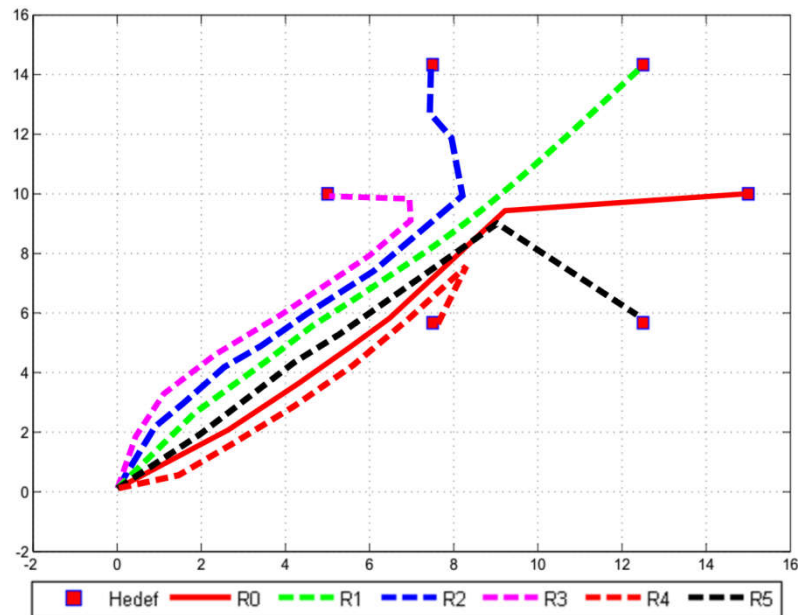


Şekil 3.15:İşbirlikli Robotların Önceden Atanmış Hedeflere Ulaşması

Robotların bulanık akın algoritması ile işbirlikli olarak çalışması esnasında bulanık akın algoritmasının desen merkezine kadar çalıştırılıp daha sonra robotların hedeflere gitmesi gerçekleşmektedir. Robotların bulanık akın algoritması ile işbirlikli olarak yarışarak hedefe gitme uygulaması sonucu robotların izlediği yollar Şekil 3.14’de gösterilmiştir. Robotların işbirlikli olarak önceden atanmış hedefe gitme uygulamasında robotların izlediği yollar Şekil 3.15’de gösterilmiştir.



Şekil 3.16: Bulanık İşbirlikli Robotların Yarışarak Hedeflere Ulaşması



Şekil 3.17: Bulanık İşbirlikli Robotların Yarışarak Hedeflere Ulaşması



Robotların düzen alma işlemlerini tamamlama süreleri ve bu sürede yaptıkları haberleşme tekrar sayıları Tablo 3.7’de verilmiştir. Buna göre yarışmacı sistemlerin görevi tamamlaması daha uzun sürdüğü görülmektedir. Yarışmacı sistemlerden bireysel uygulama toplamda 5.15 dk süren 617 tekrar haberleşme kaydı içermektedir. İşbirlikçi yarışmacı sistem ise 652 tekrar haberleşme kaydı içermekte ve toplam 5.43 dk sürmüştür. Önceden atanmış hedefler çalışan uygulamalarda tamamlama süresi daha kısadır ve bireysel sistem için 352 tekrar haberleşme kaydı içermektedir. Bu uygulamada görev tamamlama 2.94 dk sürmüştür. İşbirlikçi önceden tanımlı hedeflerle çalışan sistemde görev tamamlama 295 tekrar haberleşme kaydı içermektedir ve toplam 2.45 dk sürmüştür. İşbirlikçi önceden tanımlı hedeflerle çalışan sistemde görev tamamlama 295 tekrar haberleşme kaydı içermektedir ve toplam 2.45 dk sürmüştür. Bulanık akın algoritması ile çalışan işbirlikli sistemlerde ise önceden atanmış hedeflerle elde edilen sonuç 270 haberleşme tekrarı ve görev tamamlama süresi 2.25’dk olmuştur. Yarışmacı bulanık işbirlikli çalışma sonucu ise 635 haberleşme tekrarı ve 5,30’dk görev tamamlama süresidir.

Tablo 3.7: Düzen Alma Tamamlama Tekrar ve Süreleri

<b>Uygulama</b>	<b>Haberleşme Tekrarı</b>	<b>Süre (dk)</b>
Bireysel Yarışmacı	617	5.15
Bireysel Önceden Atanmış	352	2.94
İşbirlikli Yarışmacı	652	5.43
İşbirlikli Önceden Atanmış	295	2.45
Bulanık İşbirlikli Yarışmacı	635	5,30
Bulanık İşbirlikli Önceden Atanmış	270	2,25

## 4. SONUÇ VE ÖNERİLER

Çoklu robotik, hantal ve çok karmaşık bir robot yerine çok sayıda ve daha basit robotları işbirliği içinde çalıştırarak aynı görevi veya daha karmaşık görevleri yerine getirmeyi amaçlamaktadır. Bu sistemlerde işbirliği davranışlarının gerçekleştirilmesi için robotların etkileşim içinde olması gerekmektedir. Bu amaçla tasarlanan konumlandırma ve düzen alma teknikleri tez çalışmasında sunulmuş ve uygulamalar ile elde edilen bulgular paylaşılmıştır. Tasarlanan denetim kartı ile dört rotorlu ve altı rotorlu hava araçları için uçuş denetim başarı ile gerçekleştirilmiştir. Mekanik sıkıntılardan dolayı meydana gelen rastgele sürüklenme probleminin yazılımsal olarak telafisi mümkün gözükmemektedir. Bu durum çoklu robot uygulamalarında da geçerli olup yüksek hassasiyetli imalat gerektirmektedir.

Konumlandırma destek amacıyla tasarlanan görsel eşleme çalışmasından elde edilen bulgulara göre:

- En yüksek başarımlı katsayıları çeviri durumundadır. Diğer durumlarda başarımlı oranı çok düşmektedir. Bu sebeple konumlandırma algoritmasının çeviri durumunda çalıştırılması için yön ve yükseklik ölçecek sensörler ile desteklenmesi gerekmektedir.
- Performansı en yüksek olan üç algoritmanın çalışma sürelerinin ortalaması 10sn civarındadır. Bu durumda kullanılan görüntülerin büyüklüğü 750x750 pikseldir. Konumlandırma algoritmasının dağıtık olarak çalıştığı düşünülürse bu süre yaklaşık olarak altıda bir olacaktır. Ayrıca görüntü boyutları küçültülürse çalışma süresi bir saniyenin altına düşecektir. Bu durumda konumlandırma algoritmasının sürü robotlarda kullanımı yeterli veri oranını sağlayacaktır.
- Konumlandırma algoritması için en yüksek performansı CENSURE-FREAK algoritma birleşimi elde etmiştir. SIFT ve SURF gibi kendini kanıtlamış ve patentli algoritmalar çalışma süresi açısından alt sırada kalmıştır. CENSURE-FREAK kombinasyonunun kullanılmasının performansı en yüksek yaptığı görülmüştür.

- Yapılan denemeler sonucunda hız ve başarımlar açısından konumlandırma algoritmasının uçan sürü robotlarda kullanılmasının mümkün olduğu görülmüştür. Ayrıca sistem sensör ve işlemci gereksinimleri bakımından oldukça ekonomiktir.
- Algoritmada kullanılan veri paylaşımları yayın mesajı olarak tasarlanmıştır bu sebeple konum hesaplamaları için geçen süre robot sayısına göre değişecektir. Yüksek sayıda robotun dağıtık olarak çalışabilmesi için algoritmanın veri paylaşım kısmının iyileştirilmesi gerekmektedir.

Tek gözlü görsel etkin poz hesaplama algoritması ile yapılan çalışmada yalnızca görsellerden faydalanarak konum hesaplaması yapılmıştır. Elde edilen bulgulara göre:

- Algoritma kısa mesafe ve düz ilerleme durumlarında iyi sonuçlar üretmektedir.
- Her dönemde hata oranlarının arttığı gözlenmiştir. Bu açıdan jiroskop ve pusula verisinden faydalanarak dönüşte oluşan hatanın azaltılmasında gerekmektedir. Bu sayede kapalı alanda kullanımı mümkün olacaktır.

Düzen alma uygulamaları sonucu elde edilen bulgulara göre:

- Yarışmacı sistemler, yarışmayan saha uygulamasında görev tamamlama süresi ve işlem tekrarı açısından yavaş kalmaktadır.
- Çoklu robot yapılarında toplu halde hareket etmede topluluk içinde yarışma olmaz ise yalnız başına çalışan sistemlerden daha hızlı şekilde düzen deseni içinde dağılım sağlanabilmektedir.
- Çoklu robotlarda daha yüksek performans almak için görev paylaşımı daha önceden sistemde fikir birliği içinde çözülmeli veya kullanıcı tarafından atanmalıdır.
- Bulanık akın algoritması, Boids algoritmasından türetilen akın algoritmasından daha kısa görev tamamlama süresi ve daha az haberleşme tekrarı sağlamaktadır. Bu açıdan bakıldığında tasarlanan bulanık akın algoritmasının başarımlarını daha yüksektir.

## 5. KAYNAKLAR

Achtelik, M., Brunet, Y., Chli, M., Chatzichristofis, S., Decotignie, J. dominique, Doth, K. michael, Fraundorfer, F., Kneip, L., Gurdan, D., Heng, L. and others, "SFly: Swarm of micro flying robots". *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on.*, IEEE, pp. 2649–2650,(2012).

Agrawal, M., Konolige, K. and Blas, M. R., "CenSurE: Center Surround Extremas for Realtime Feature Detection and Matching". In: Forsyth, D., P. Torr & A. Zisserman (eds.), *Computer Vision – ECCV 2008.*, Berlin, Heidelberg: Springer Berlin Heidelberg, Vol. 5305pp. 102–115,(2008).

Alahi, A., Ortiz, R. and Vandergheynst, P., "Freak: Fast retina keypoint". *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on.*, Ieee, pp. 510–517,(2012).

Amir, M. Y. and Abbas, V., "Modeling and Neural Control of Quad rotor Helicopter". *Yanbu Journal of Engineering and Science.*, 2, 35–49,(2011).

Aranda, M., López-Nicolás, G., Sagüés, C. and Mezouar, Y., "Formation control of mobile robots using multiple aerial cameras". *IEEE Transactions on Robotics.*, 31(4), 1064–1071,(2015).

Arvin, F., Samsudin, K. and Ramli, A. R., "A short-range infrared communication for swarm mobile robots". *2009 International Conference on Signal Processing Systems.*, IEEE, pp. 454–458,(2009).

Attiya, H. and Welch, J., *Distributed computing: fundamentals, simulations, and advanced topics.*, John Wiley & Sons, Vol. 19,(2004).

Balch, T., "The impact of diversity on performance in multi-robot foraging". *Proceedings of the third annual conference on Autonomous Agents.*, ACM, pp. 92–99,(1999).

Bay, H., Tuytelaars, T. and Van Gool, L., "Surf: Speeded up robust features". *Computer vision–ECCV 2006.*, Springer, pp. 404–417,(2006).

Bayindir, L. and Şahin, E., "A review of studies in swarm robotics". *Turkish Journal of Electrical Engineering & Computer Sciences.*, 15(2), 115–147,(2007).

- Beard, R. W., McLain, T. W., Goodrich, M. A. and Anderson, E. P., "Coordinated target assignment and intercept for unmanned air vehicles". *Robotics and Automation, IEEE Transactions on.*, 18(6), 911–922,(2002).
- Beckers, R., Holland, O. E. and Deneubourg, J. L., "From local actions to global tasks: Stigmergy and collective robotics". *Artificial life IV.*, Vol. 181p. 189,(1994).
- Berhault, M., Huang, H., Keskinocak, P., Koenig, S., Elmaghraby, W., Griffin, P. and Kleywegt, A., "Robot exploration with combinatorial auctions". *Intelligent Robots and Systems, 2003.(IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on.*, IEEE, Vol. 2pp. 1957–1962,(2003).
- Bolandi, H., Rezaei, M., Mohsenipour, R., Nemati, H. and Smailzadeh, S. M., "Attitude Control of a Quadrotor with Optimized PID Controller". *Intelligent Control and Automation.*, 04(03), 335–342,(2013).
- Borenstein, J., Everett, H. R., Feng, L. and Wehe, D., *Mobile robot positioning-sensors and techniques.*, DTIC Document,(1997).
- Bouabdallah, S., a., *Design and control of quadrotors with application to autonomous flying.*, École Polytechnique federale de Lausanne, (2007).
- Bouabdallah, S. and Siegwart, R., "Full control of a quadrotor". *Intelligent robots and systems, 2007. IROS 2007. IEEE/RSJ international conference on.*, IEEE, pp. 153–158,(2007).
- Boutteau, R., Mazari, B., Ertaud, J. Y. and Savatier, X., *A 3D Omnidirectional Sensor For Mobile Robot Applications.*, INTECH Open Access Publisher,(2010).
- Bresciani, T., a., *Modelling, identification and control of a quadrotor helicopter.*, Department of Automatic Control, Lund University,(2008).
- Butler, Z. J., Rizzi, A. A. and Hollis, R. L., "Cooperative coverage of rectilinear environments". *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on.*, IEEE, Vol. 3pp. 2722–2727,(2000).
- Calonder, M., Lepetit, V., Strecha, C. and Fua, P., "Brief: Binary robust independent elementary features". *Computer Vision–ECCV 2010.*, 778–792,(2010).
- Cao, Y. U., Fukunaga, A. S., Kahng, A. B. and Meng, F., "Cooperative mobile robotics: Antecedents and directions". *Intelligent Robots and Systems*

95. 'Human Robot Interaction and Cooperative Robots', *Proceedings. 1995 IEEE/RSJ International Conference on..*, IEEE, Vol. 1pp. 226–234,(1995).

Chen, W. and Meng, X., "A cooperative localization scheme for Zigbee-based wireless sensor networks". *Networks, 2006. ICON'06. 14th IEEE International Conference on..*, IEEE, Vol. 2pp. 1–5,(2006).

Chen, X., Li, D., Xu, Z. and Bai, Y., "Robust control of quadrotor MAV using self-organizing interval type-II fuzzy neural networks (SOIT-IIFNNs) controller". *International Journal of Intelligent Computing and Cybernetics.*, 4(3), 397–412,(2011).

Cianci, C. M., Raemy, X., Pugh, J. and Martinoli, A., "Communication in a swarm of miniature robots: The e-puck as an educational tool for swarm robotics". *International Workshop on Swarm Robotics*, Springer, pp. 103–115,(2006).

Couceiro, M. S., Figueiredo, C. M., Luz, J. M. A., Ferreira, N. M. and Rocha, R. P., "A Low-Cost Educational Platform for Swarm Robotics.". *International Journal of Robots, Education & Art.*, 2(1),(2012).

Davison, A. J., "Real-time simultaneous localisation and mapping with a single camera". *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on..*, IEEE, pp. 1403–1410,(2003).

do Nascimento, D. A., Rufino, C. Z., Oliveira, S. C. and Feitosa, M. A. F., "Modeling and Design of an Autonomous Unmanned Aerial Vehicle Quadrotor Prototype for Applications in Swarm of Robots". *Robotics Symposium and Latin American Robotics Symposium (SBR-LARS), 2012 Brazilian..*, IEEE, pp. 325–328,(2012).

Donoser, M. and Bischof, H., "Efficient maximally stable extremal region (MSER) tracking". *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on..*, IEEE, Vol. 1pp. 553–560,(2006).

Dorigo, M., Tuci, E., Gros, R., Trianni, V., Labella, T. H., Nouyan, S., Ampatzis, C., Deneubourg, J. L., Baldassarre, G., Nolfi, S. and others, "The swarm-bots project". *International Workshop on Swarm Robotics..*, Springer, pp. 31–44,(2004).

Dorigo, M., "Swarm-Bots and Swarmanoid: Two Experiments in Embodied Swarm Intelligence.". *Web intelligence.*, pp. 2–3,(2009).

Dorigo, M., Floreano, D., Gambardella, L. M., Mondada, F., Nolfi, S., Baaboura, T., Birattari, M., Bonani, M., Brambilla, M., Brutschy, A. and

others, "Swarmanoid". *IEEE Robotics & Automation Magazine.*, 1070(9932/13),(2013).

Douillard, B., Underwood, J., Kuntz, N., Vlaskine, V., Quadros, A., Morton, P. and Frenkel, A., "On the segmentation of 3D LIDAR point clouds". *Robotics and Automation (ICRA), 2011 IEEE International Conference on.*, IEEE, pp. 2798–2805,(2011).

Drogoul, A. and Ferber, J., "From tom thumb to the dockers: Some experiments with foraging robots". *From Animals to Animats II.*, 451–459,(1993).

Efe, M. o., "Integral sliding mode control of a quadrotor with fractional order reaching dynamics". *Transactions of the Institute of Measurement and Control.*, 33(8), 985–1003,(2010).

Efe, M. Ö., "Neural network assisted computationally simple PID control of a quadrotor UAV". *IEEE Trans. Ind. Informat.*, 7(2), 354–361,(2011).

Erginer, B. and Altug, E., "Modeling and PD control of a quadrotor VTOL vehicle". *Intelligent Vehicles Symposium, 2007 IEEE.*, IEEE, pp. 894–899,(2007).

Fang, Z. and Gao, W., "Adaptive integral backstepping control of a Micro-Quadrotor". *Intelligent Control and Information Processing (ICICIP), 2011 2nd International Conference on.*, Vol. 2pp. 910–915,(2011).

Farinelli, A., Iocchi, L. and Nardi, D., "Multirobot systems: a classification focused on coordination". *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics).*, 34(5), 2015–2028,(2004).

Ferber, J., *Multi-agent systems: an introduction to distributed artificial intelligence.*, Addison-Wesley Reading, Vol. 1,(1999).

Fox, D., Burgard, W., Kruppa, H. and Thrun, S., "Collaborative multi-robot localization". *Mustererkennung 1999.*, Springer, pp. 15–26,(1999).

Fresk, E. and Nikolakopoulos, G., "Full quaternion based attitude control for a quadrotor". *Control Conference (ECC), 2013 European.*, IEEE, pp. 3864–3869,(2013).

Grossman, D. D., "Traffic control of multiple robot vehicles". *Robotics and Automation, IEEE Journal of.*, 4(5), 491–497,(1988).

Heng, L., Lee, G. H., Fraundorfer, F. and Pollefeys, M., "Real-time photo-realistic 3d mapping for micro aerial vehicles". *Intelligent Robots and*

*Systems (IROS), 2011 IEEE/RSJ International Conference on..*, IEEE, pp. 4012–4019,(2011).

Honglei, A., Jie, L., Jian, W., Jianwen, W. and Hongxu, M., "Backstepping-Based Inverse Optimal Attitude Control of Quadrotor". *International Journal of Advanced Robotic Systems.*, 1,(2013).

Joordens, M. and Champion, B., "Underwater Swarm Robotics: Challenges and Opportunities". *Handbook of Research on Design, Control, and Modeling of Swarm Robotics.*, IGI Global, pp. 718–740,(2016).

Joordens, M. A. and Jamshidi, M., "Underwater swarm robotics consensus control". *Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on..*, IEEE, pp. 3163–3168,(2009).

Karlsson, N., Di Bernardo, E., Ostrowski, J., Goncalves, L., Pirjanian, P. and Munich, M. E., "The vSLAM algorithm for robust localization and mapping". *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on..*, IEEE, pp. 24–29,(2005).

Kloder, S. and Hutchinson, S., "Path planning for permutation-invariant multirobot formations". *Robotics, IEEE Transactions on.*, 22(4), 650–665,(2006).

Kriegman, D., Triendl, E. and Binford, T. O., "Stereo vision and navigation in buildings for mobile robots". *Robotics and Automation, IEEE Transactions on.*, 5(6), 792–803,(1989).

Kube, C. R. and Zhang, H., "Collective robotics: From social insects to robots". *Adaptive Behavior.*, 2(2), 189–218,(1993).

Kushleyev, A., Mellinger, D., Powers, C. and Kumar, V., "Towards a swarm of agile micro quadrotors". *Autonomous Robots*, 35(4), 287–300,(2013).

Leutenegger, S., Chli, M. and Siegwart, R. Y., "BRISK: Binary robust invariant scalable keypoints". *Computer Vision (ICCV), 2011 IEEE International Conference on..*, IEEE, pp. 2548–2555,(2011).

Li, J. and Li, Y., "Dynamic analysis and PID control for a quadrotor". *Mechatronics and Automation (ICMA), 2011 International Conference on.*, pp. 573–578,(2011).

Liu, H., Darabi, H., Banerjee, P. and Liu, J., "Survey of wireless indoor positioning techniques and systems". *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on.*, 37(6), 1067–1080,(2007).



- López-Nicolás, G., Aranda, M. and Sagüés, C., "Multi-robot Formations: One Homography to Rule Them All.". *ROBOT (1)*., pp. 703–714,(2013).
- Lowe, D. G., "Distinctive image features from scale-invariant keypoints". *International journal of computer vision.*, 60(2), 91–110,(2004).
- Luo, R. C., Lin, M. H. and Scherp, R. S., "Dynamic multi-sensor data fusion system for intelligent robots". *IEEE Journal on Robotics and Automation.*, 4(4), 386–396,(1988).
- Madani, T. and Benallegue, A., "Sliding mode observer and backstepping control for a quadrotor unmanned aerial vehicles". *American Control Conference, 2007. ACC'07.*, pp. 5887–5892,(2007).
- Madani, T. and Benallegue, A., "Adaptive control via backstepping technique and neural networks of a quadrotor helicopter". *Proceedings of the 17th World Congress of The International Federation of Automatic Control*,(2008).
- Malis, E. and Vargas, M., "Deeper understanding of the homography decomposition for vision-based control",(2007).
- Mataric, M. J., "Designing emergent behaviors: From local interactions to collective intelligence". *Proceedings of the Second International Conference on Simulation of Adaptive Behavior.*, pp. 432–441,(1993).
- Matarić, M. J., "Issues and approaches in the design of collective autonomous agents". *Robotics and autonomous systems.*, 16(2), 321–331,(1995).
- McLurkin, J. D., *Stupid robot tricks: A behavior-based distributed algorithm library for programming swarms of robots.*, Massachusetts Institute of Technology, (2004).
- Michael, N., Mellinger, D., Lindsey, Q. and Kumar, V., "The grasp multiple micro-uav testbed". *Robotics & Automation Magazine, IEEE.*, 17(3), 56–65,(2010).
- Mourikis, A. I. and Roumeliotis, S. I., "Performance analysis of multirobot cooperative localization". *Robotics, IEEE Transactions on.*, 22(4), 666–681,(2006).
- Murray, D. and Jennings, C., "Stereo vision based mapping and navigation for mobile robots". *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on.*, IEEE, Vol. 2pp. 1694–1699,(1997).
- Nicol, C. E., *A robust adaptive neural network control for a quadrotor helicopter.*, Library and Archives Canada = Bibliothèque et Archives Canada, Ottawa,(2011).

- Parker, L. E., "Cooperative robotics for multi-target observation". *Intelligent Automation & Soft Computing.*, 5(1), 5–19,(1999).
- Parker, L. E., "Multiple mobile robot systems". *Springer Handbook of Robotics.*, Springer, pp. 921–941,(2008).
- Passino, K. M., "Biomimicry of bacterial foraging for distributed optimization and control". *Control Systems, IEEE.*, 22(3), 52–67,(2002).
- Payton, D. W., Daily, M. J., Hoff, B., Howard, M. D. and Lee, C. L., "Pheromone robotics". *Intelligent Systems and Smart Manufacturing.*, International Society for Optics and Photonics, pp. 67–75,(2001).
- Quintero, S., Collins, G. E., Hespanha, J. P. and others, "Flocking with fixed-wing UAVs for distributed sensing: A stochastic optimal control approach". *American Control Conference (ACC), 2013.*, IEEE, pp. 2025–2031,(2013).
- Renzaglia, A., Doitsidis, L., Martinelli, A. and Kosmatopoulos, E. B., "Multi-robot three dimensional coverage of unknown areas". *The International Journal of Robotics Research.*, 0278364912439332,(2012).
- Reynolds, C. W., "Flocks, herds and schools: A distributed behavioral model". *ACM SIGGRAPH Computer Graphics.*, 21(4), 25–34,(1987).
- Reynolds, C. W., "Interaction with groups of autonomous characters". *Game Developers Conference.*, Vol. 2000pp. 449–460,(2000).
- Rohmer, E., Singh, S. P. and Freese, M., "V-REP: A versatile and scalable robot simulation framework". *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on.*, IEEE, pp. 1321–1326,(2013).
- Rosten, E., Porter, R. and Drummond, T., "Faster and better: A machine learning approach to corner detection". *Pattern Analysis and Machine Intelligence, IEEE Transactions on.*, 32(1), 105–119,(2010).
- Rubenstein, M., Ahler, C. and Nagpal, R., "Kilobot: A low cost scalable robot system for collective behaviors". *Robotics and Automation (ICRA), 2012 IEEE International Conference on.*, IEEE, pp. 3293–3298,(2012).
- Rublee, E., Rabaud, V., Konolige, K. and Bradski, G., "ORB: an efficient alternative to SIFT or SURF". *Computer Vision (ICCV), 2011 IEEE International Conference on.*, IEEE, pp. 2564–2571,(2011).
- Salih, A. L., Moghavvemi, M., Mohamed, H. A. and Gaeid, K. S., "Modelling and PID controller design for a quadrotor unmanned air vehicle". *Automation Quality and Testing Robotics (AQTR), 2010 IEEE International Conference on.*, IEEE, Vol. 1pp. 1–5,(2010).

Santos, M., López, V. and Morata, F., "Intelligent fuzzy controller of a quadrotor". *Intelligent Systems and Knowledge Engineering (ISKE), 2010 International Conference on.*, pp. 141–146,(2010).

Se, S., Lowe, D. and Little, J., "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks". *The international Journal of robotics Research.*, 21(8), 735–758,(2002).

Seyfried, J., Szymanski, M., Bender, N., Estana, R., Thiel, M. and Wörn, H., "The I-SWARM project: Intelligent small world autonomous robots for micro-manipulation". *Swarm Robotics.*, Springer, pp. 70–83,(2005).

Shaneyfelt, T., Joordens, M. A., Nagothu, K. and Jamshidi, M., "RF communication between surface and underwater robotic swarms". *WAC'08: Proceedings of the World Automation Congress' 08.*, IEEE, pp. 1–6,(2008).

Siciliano, B. and Khatib, O. (Eds.), *Springer handbook of robotics.*, Berlin: Springer,(2008).

Steder, B., Grisetti, G., Stachniss, C. and Burgard, W., "Visual SLAM for flying vehicles". *Robotics, IEEE Transactions on.*, 24(5), 1088–1093,(2008).

Stirling, T., Roberts, J., Zufferey, J. C. and Floreano, D., "Indoor navigation with a swarm of flying robots". *Robotics and Automation (ICRA), 2012 IEEE International Conference on.*, IEEE, pp. 4641–4647,(2012).

Stroupe, A. W., Martin, M. C. and Balch, T., "Distributed sensor fusion for object position estimation by multi-robot systems". *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on.*, IEEE, Vol. 2pp. 1092–1098,(2001).

Tan, Y. and Zheng, Z. yang, "Research Advance in Swarm Robotics". *Defence Technology.*, 9(1), 18–39,(2013).

Turgut, A. E., Gökçe, F., Celikkanat, H., Bayındır, L. and Sahin, E., "Kobot: Suru robot çalışmaları için tasarlanmış gezgin robot platformu". *TOK'07.*, p. 259,(2007).

Vásárhelyi, G., Virágh, C., Somorjai, G., Tarcai, N., Szorenyi, T., Nepusz, T. and Vicsek, T., "Outdoor flocking and formation flight with autonomous aerial robots". *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on.*, IEEE, pp. 3866–3873,(2014).

Wang, J., "Fully distributed traffic control strategies for many-AGV systems". *Intelligent Robots and Systems' 91. Intelligence for Mechanical*

*Systems, Proceedings IROS'91. IEEE/RSJ International Workshop on.*, IEEE, pp. 1199–1204,(1991).

Weiss, S., Scaramuzza, D. and Siegwart, R., "Monocular-SLAM-based navigation for autonomous micro helicopters in GPS-denied environments". *Journal of Field Robotics.*, 28(6), 854–874,(2011).

Winfield, A. F., Liu, W., Nembrini, J. and Martinoli, A., "Modelling a wireless connected swarm of mobile robots". *Swarm Intelligence.*, 2(2–4), 241–266,(2008).

Yan, Z., Jouandeau, N. and Cherif, A. A., "A survey and analysis of multi-robot coordination". *International Journal of Advanced Robotic Systems.*, 10(12), 399,(2013).

## 6. ÖZGEÇMİŞ

Adı Soyadı : Ahmet Çağdaş SEÇKİN  
Doğum Yeri ve Tarihi : ANKARA 29.07.1985  
Lisans : Süleyman Demirel Üniversitesi  
Y. Lisans : **Pamukkale Üniversitesi**  
Elektronik posta : cagdaseckin@gmail.com  
İletişim Adresi : Fatih Mah. 1. Köksal Sok. No:18  
Daire:17 Merkez UŞAK

### Yayın Listesi

- **Seçkin, A.Ç.**, Karpuz, C., Özek, A., “Feature matching based positioning algorithm for swarm robotics”, Computers and Electrical Engineering, (2016)
- Balcıoğlu, H. E., **Seçkin A. Ç.**, Aktaş, M. "Failure load prediction of adhesively bonded pultruded composites using artificial neural network." Journal of Composite Materials, 50(23) 3267-3281, (2016).
- Sezer, A, İnel, Y., **Seçkin, A.Ç.**, Uluçınar, U., "The Relationship between Attention Levels and Class Participation of First-Year Students in Classroom Teaching Departments." International Journal of Instruction 10(2) 55-68, (2017).

### Konferans listesi

- **Seçkin, A. Ç.**, Karpuz, C., Özek, A., "Sürü Robotiği", Uluslararası Bilgisayar Bilimleri ve Mühendisliği Konferansı, 2016, Tekirdağ
- Kaynak, İ., **Seçkin, A.Ç.**, Yücesoy, S.A., Koç, D., “Eğitim Uygulamaları için İnsan Bilgisayar Arayüz Aygıtı Tasarımı” , Uluslararası Bilgisayar Bilimleri ve Mühendisliği Konferansı, 2016, Tekirdağ

- Sekin, M., Turan, N.Y, **Sekin, A..**, “Comparison of Production Methods in Soft Robotic”, International Conference on Advances in Software, Control and Mechanical Engineering, 2015, Antalya

- Sezer, A, İnel, Y., **Sekin, A..**, Uluınar, U., " An Investigation of University Students' Attention Levels in Real Classroom Settings with NeuroSky's MindWave Mobile (EEG) Device" International Educational Technology Conference, 2015, İstanbul