

**YAPAY ZEKÂ DESTEKLİ SÜRÜ ROBOTLARI İLE TARIM BİTKİLERİNİN
YETİŞTİRİLMESİ VE ZİRAİ MÜCADELEDE KULLANILMASI**

DOKTORA TEZİ

Muhammed Mustafa KELEK

**Danışman
Prof. Dr. Yüksel OĞUZ**

**İkinci Danışman
Doç. Dr. Uğur FİDAN**

ELEKTRİK–ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

Ekim 2025

AFYON KOCATEPE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DOKTORA TEZİ

YAPAY ZEKÂ DESTEKLİ SÜRÜ ROBOTLARI İLE TARIM
BİTKİLERİNİN YETİŞTİRİLMESİ VE ZİRAİ MÜCADELEDE
KULLANILMASI

Muhammed Mustafa KELEK

Danışman

Prof. Dr. Yüksel OĞUZ

İkinci Danışman

Doç. Dr. Uğur FİDAN

ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

Ekim 2025

TEZ ONAY SAYFASI

Muhammed Mustafa KELEK tarafından hazırlanan “Yapay Zekâ Destekli Sürü Robotları ile Tarım Bitkilerinin Yetiştirilmesi ve Zirai Mücadelede Kullanılması” adlı tez çalışması lisansüstü eğitim ve öğretim yönetmeliğinin ilgili maddeleri uyarınca 31/10/2025 tarihinde aşağıdaki jüri tarafından **oy birliği** ile Afyon Kocatepe Üniversitesi Fen Bilimleri Enstitüsü **Elektrik–Elektronik Mühendisliği Anabilim Dalı’nda DOKTORA TEZİ** olarak kabul edilmiştir.

Danışman: Prof. Dr. Yüksel OĞUZ

İkinci Danışman: Doç. Dr. Uğur FİDAN

- Başkan** : Prof. Dr. Sezai TAŞKIN
Manisa Celal Bayar Üniversitesi, Mühendislik Fakültesi
- Üye** : Prof. Dr. Yüksel OĞUZ
Afyon Kocatepe Üniversitesi, Teknoloji Fakültesi
- Üye** : Prof. Dr. Ahmet ALTUNCU
Afyon Kocatepe Üniversitesi, Teknoloji Fakültesi
- Üye** : Prof. Dr. İsmail KOYUNCU
Afyon Kocatepe Üniversitesi, Teknoloji Fakültesi
- Üye** : Doç. Dr. Kubilay TAŞDELEN
Isparta Uygulamalı Bilimler Üniversitesi, Teknoloji Fakültesi

Afyon Kocatepe Üniversitesi
Fen Bilimleri Enstitüsü Yönetim Kurulu’nun
..... /..... /..... tarih ve
..... sayılı kararıyla onaylanmıştır.

.....
Prof. Dr. Bekir YALÇIN
Enstitü Müdürü

BİLİMSEL ETİK BİLDİRİM SAYFASI
Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- Görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- Başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- Atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- Kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- Ve bu tezin herhangi bir bölümünü bu üniversite veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

31 / 10 / 2025

Muhammed Mustafa KELEK

ÖZET

Doktora Tezi

YAPAY ZEKÂ DESTEKLİ SÜRÜ ROBOTLARI İLE TARIM BİTKİLERİNİN YETİŞTİRİLMESİ VE ZİRAİ MÜCADELEDE KULLANILMASI

Muhammed Mustafa KELEK

Afyon Kocatepe Üniversitesi

Fen Bilimleri Enstitüsü

Elektrik–Elektronik Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Yüksel OĞUZ

İkinci Danışman: Doç. Dr. Uğur FİDAN

Bu tez çalışmasında, tarım bitkilerinin yetiştirilmesi ve zararlılardan korunması süreçlerinde kullanılmak üzere yapay zekâ destekli sürü drone sistemi geliştirilmiş ve Gazebo simülasyon ortamında modellenmiştir. Çalışmanın temel amacı, tarımsal üretim süreçlerinde otonom olarak görev yapabilen sürü robotlarının görev planlama, nesne tespiti ve rota optimizasyonu bileşenlerini bütünlük bir mimaride bir araya getirmektir. Geliştirilen sistem hem veri toplama hem de ilaçlama işlevlerini eş zamanlı olarak gerçekleştirebilen çoklu drone yapısına sahiptir.

Sürüde görev alan kâşif drone, tarım arazisini sensör verileriyle tarayarak görüntü toplamakta; bu görüntüler YOLO (You Only Look Once) tabanlı derin öğrenme modeliyle işlenmekte ve tespit edilen hedef konumlar ilaçlama drone'larına aktarılmaktadır. Böylece bitkilerin bulunduğu alanlara hassas ve hedef odaklı ilaçlama yapılabilmektedir. Sistem mimarisi; Robot İşletim Sistemi (ROS), ArduPilot, MAVLink ve MAVROS protokollerinin entegrasyonu ile çoklu drone haberleşmesini, veri akışını ve görev senkronizasyonunu sağlamaktadır. Görev planlamasında Karınca Kolonisi Optimizasyonu (Ant Colony Optimization – ACO) ve En Yakın Komşu (Nearest Neighbor – NN) algoritmaları birlikte kullanılarak rota optimizasyonu yapılmış; uçuş mesafesi, bataryanın şarj durumu (State of Charge – SOC) değeri ve görev süresi parametrelerinde anlamlı iyileşmeler elde edilmiştir. Optimizasyon uygulanmayan senaryoda toplam uçuş mesafesi 960 m iken NN algoritmasıyla bu değer %10 azalarak

865 m'ye, ACO algoritmasıyla ise %13.4 azalarak 831 m'ye düşürülmüştür. Buna paralel olarak batarya SOC değerleri sırasıyla %63, %65 ve %66 olarak ölçülmüş ve ACO ile %4.54'lük bir avantaj elde edilmiştir. Aynı şekilde uçuş süreleri de sırasıyla 491.39 s, 470.34 s ve 447.28 s olarak kayıt edilmiş olup ACO ile yaklaşık %9'a varan bir iyileşme sağlanmıştır.

Nesne tespiti modülünde YOLOv3 ve YOLOv5 modelleri karşılaştırmalı olarak değerlendirilmiş; en yüksek doğruluk YOLOv5x modelinde (%99.5 mAP@0.5) elde edilmiştir. Ancak sistemin gerçek zamanlı çalışabilmesi için hesaplama verimliliği yüksek olan YOLOv3 mimarisi tercih edilmiştir. Böylelikle, donanımsal kaynak kullanımı azaltılırken nesne tespitinde yeterli doğruluk korunmuştur. Ayrıca bir drone'un devre dışı kaldığı acil durum senaryolarında görevlerin kalan drone'lara otomatik olarak devredilmesiyle sistemin güvenilirliği ve sürekliliği artırılmıştır.

Tez kapsamında geliştirilen yapay zekâ destekli sürü drone mimarisi, tarımsal ilaçlama, bitki izleme ve veri toplama süreçlerinde hız, doğruluk, enerji verimliliği ve görev güvenliği açısından önemli kazanımlar sağlamıştır. Nesne tespiti, görev paylaşımı ve rota optimizasyonu süreçlerinin bütünleşik biçimde yürütülmesi, tarımsal üretim süreçlerinde insan müdahalesini en aza indiren, ölçeklenebilir ve otonom bir çözüm ortaya koymuştur. Elde edilen sonuçlar, sürü robotlarının koordineli hareket kabiliyetinin hem kaynak kullanımını optimize ettiğini hem de operasyonel sürekliliği artırdığını göstermektedir. Ayrıca sistemin açık kaynaklı simülasyon ortamında geliştirilen modüler yapısı, farklı donanım platformlarına kolaylıkla uyarlanabilmesine olanak tanımaktadır. Bu çalışma, gelecekte otonom tarım sistemlerinde sürü robotlarının yaygın kullanımına yönelik güçlü bir mühendislik altyapısı sunmaktadır.

2025, xiv + 134 sayfa

Anahtar Kelimeler: Sürü drone, Yapay zekâ, Tarımda robotlar, YOLO, Zirai ilaçlama, Optimizasyon.

ABSTRACT

Ph.D. Thesis

ARTIFICIAL INTELLIGENCE-BASED SWARM ROBOTS FOR GROWING AGRICULTURAL PLANTS AND USAGE IN AGRICULTURAL CONTROL

Muhammed Mustafa KELEK

Afyon Kocatepe University

Graduate School of Natural and Applied Sciences

Department of Electrical and Electronics Engineering

Supervisor: Prof. Dr. Yüksel OĞUZ

Co-Supervisor: Assoc. Prof. Dr. Uğur FİDAN

In this Study, an artificial intelligence-supported swarm drone system was developed and modeled in the Gazebo simulation environment for use in cultivating agricultural plants and their protection against pests. The study's main objective is to integrate the components of mission planning, object detection, and route optimization of swarm robots capable of performing autonomous tasks in agricultural production processes within a unified architecture. The developed system features a multi-drone structure capable of simultaneously performing data collection and spraying tasks.

The explorer drone scans the agricultural field within the swarm using sensor data to collect images. These images are processed by a deep learning model based on YOLO (You Only Look Once), and the detected target locations are transmitted to the spraying drones. Thus, precise and target-oriented pesticide application is achieved in areas where plants are located. The system architecture integrates the Robot Operating System (ROS), ArduPilot, MAVLink, and MAVROS protocols to ensure multi-drone communication, data flow, and mission synchronization. For mission planning, Ant Colony Optimization (ACO) and Nearest Neighbor (NN) algorithms were employed together for route optimization, resulting in significant improvements in flight distance, battery SOC value, and mission duration. In the non-optimized scenario, the total flight distance was 960 m, which decreased by 10% to 865 m with the NN algorithm and by 13.4% to 831 m with the ACO algorithm. In parallel, the battery SOC values were measured at 63%, 65%, and

66%, respectively, and a 4.54% advantage was achieved with ACO. Similarly, flight times were recorded as 491.39 s, 470.34 s, and 447.28 s, respectively, with an improvement of approximately 9% achieved with ACO.

In the object detection module, YOLOv3 and YOLOv5 models were comparatively evaluated, and the highest accuracy was achieved with the YOLOv5x model (99.5% mAP@0.5). However, to ensure real-time operation of the system, the computationally efficient YOLOv3 architecture was preferred. This way, hardware resource utilization was reduced while maintaining sufficient detection accuracy. Moreover, in emergency scenarios where one drone becomes inoperative, the automatic reassignment of tasks to the remaining drones enhances the system's reliability and continuity.

The artificial intelligence-supported swarm drone architecture developed within this thesis's scope has significantly improved speed, accuracy, energy efficiency, and operational safety for agricultural spraying, crop monitoring, and data collection processes. The integrated execution of object detection, task allocation, and route optimization has produced an autonomous, scalable solution that minimizes human intervention in agricultural production. The results demonstrate that the coordinated motion capability of swarm robots optimizes resource utilization and enhances operational continuity. Furthermore, the modular design developed in an open-source simulation environment enables easy system adaptation to different hardware platforms. This study provides a robust engineering foundation for the widespread use of swarm robots in future autonomous agricultural systems.

2025, xiv + 134 pages

Keywords: Swarm drone, Artificial intelligence, Robots in agriculture, YOLO, Agricultural spraying, Optimization

TEŞEKKÜR

Bu araştırmanın konusu, deneysel çalışmaların yönlendirilmesi, sonuçların değerlendirilmesi ve yazımı aşamasında yapmış olduğu büyük katkılarından dolayı tez danışmanlarım Sayın Prof. Dr. Yüksel OĞUZ ve Doç. Dr. Uğur FİDAN'a, araştırma ve yazım süresince yardımlarını esirgemeyen ve her konuda öneri ve eleştirileriyle yardımlarını gördüğüm hocalarıma ve arkadaşlarıma teşekkür ederim.

Ayrıca bu araştırma boyunca maddi ve manevi desteklerinden dolayı eşime ve kızıma teşekkür ederim.

Muhammed Mustafa KELEK

Afyonkarahisar, 2025

İÇİNDEKİLER DİZİNİ

	Sayfa
ÖZET	i
ABSTRACT	iii
TEŞEKKÜR	v
İÇİNDEKİLER DİZİNİ.....	vi
SİMGELER ve KISALTMALAR DİZİNİ	ix
ŞEKİLLER DİZİNİ	xi
ÇİZELGELER DİZİNİ.....	xiii
RESİMLER DİZİNİ	xiv
1. GİRİŞ.....	1
2. LİTERATÜR BİLGİLERİ.....	4
2.1 Sürü Robotiği.....	4
2.1.1 Sürü Robotiği ve Temel Özellikleri	4
2.1.2 Sürü Robotlarında İletişim ve Koordinasyon Yöntemleri	6
2.1.3 Davranış Modelleri ve Kontrol Yaklaşımları.....	8
2.1.4 Sürü Robotlarının Uygulama Alanları	9
2.2 İnsansız Hava Araçları.....	11
2.2.1 İnsansız Hava Araçlarının Temel Bileşenleri	11
2.2.1.1 Gövde ve Pervane Sistemleri	12
2.2.1.2 Sensörler.....	12
2.2.1.3 Haberleşme Sistemleri.....	14
2.2.1.4 Uçuş Kontrolcüler ve Otomatik Pilot Sistemleri.....	15
2.2.2 Sabit Kanatlı İnsansız Hava Araçları	16
2.2.2.1 Aerodinamik Yapısı ve Uçuş Özellikleri	17
2.2.2.2 Kullanım Alanları.....	18
2.2.3 Döner Kanatlı İnsansız Hava Araçları	18
2.2.3.1 Quadrotor, Hexacopter ve Octocopter Yapıları	19
2.2.3.2 Uçuş Dinamikleri ve Manevra Kabiliyeti	19
2.2.3.3 Kullanım Alanları.....	21
2.3 Yapay Zekâ	22
2.3.1 Yapay Zekâ ve Tarihsel Gelişimi.....	22

2.3.2 Makine Öğrenmesi.....	24
2.3.3 Yapay Sinir Ağları	24
2.3.4 Derin Öğrenme.....	25
2.3.4.1 Evrimsel Sinir Ağları.....	26
2.3.5 Nesne Tespiti Algoritmaları.....	29
2.3.5.1 You Only Look Once	30
2.4 Literatür Taraması.....	36
2.4.1 Ulusal Çalışmalar	36
2.4.2 Uluslararası Çalışmalar	40
2.4.3 Literatürün Genel Değerlendirilmesi	49
3. MATERYAL ve METOT.....	52
3.1 Sürü Robotiğinin Simülasyon Platformu.....	52
3.1.1 Gazebo Simülasyon Programı.....	52
3.1.2 ArduPilot.....	55
3.1.3 Yazılım Döngüsünde Simülasyon.....	58
3.1.4 Mikro Hava Aracı İletişim Protokolü	59
3.1.5 DroneKit.....	62
3.1.6 Robot İşletim Sistemi Tabanlı MAVLink.....	63
3.1.7 Robot İşletim Sistemi.....	64
3.1.8 Kullanılan Sensörler.....	66
3.1.9 YOLO'nun Entegrasyonu	67
3.1.10 Kullanılan Optimizasyon Yöntemleri	68
3.1.10.1 En Yakın Komşu Algoritması	68
3.1.10.2 Karınca Kolonisi Optimizasyonu	69
3.2 Yapay Zekâ Destekli Sürü Robot Sisteminin Mimarisi.....	70
3.2.1 Genel Mimari	71
3.2.2 Gazebo Simülasyon Ortamı	72
3.2.3 Tasarlanan Sistemin Yazılım ve Kod Mimarisi.....	78
3.2.4 Sensör Verilerinin İşlenmesi ve Veri Akış Yapısı	83
3.2.5 Mesajlaşma Altyapısı.....	84
3.2.6 Tarama Modülü.....	86
3.2.7 Oluşturulan Veri Seti	87

3.2.8 YOLO Tabanlı Nesne Tespitinin Eğitimi ve Entegrasyonu	88
3.2.9 Görev Planlama Modülü	89
3.2.10 Acil Durum Senaryoları	91
3.2.11 Görselleştirme ve Harita Üzerinde Görev Takibi	92
3.3 Sistemin Değerlendirme Kriterleri.....	92
4. BULGULAR.....	94
4.1 Sürü Davranışı ve Acil Durum Senaryoları.....	94
4.2 Drone Görev Akışlarının Harita Tabanlı Görselleştirilmesi.....	96
4.3 Ziraî Mücadele için Tasarlanan Simülasyon Ortamı Sonuçları.....	101
4.4 Nesne Tespit Performansı.....	103
4.5 Görev Planlama ve Optimizasyon	107
5. TARTIŞMA ve SONUÇ.....	111
6. KAYNAKLAR	114
EKLER	124
ÖZGEÇMİŞ.....	131

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler

cm	santimetre
m	metre
ms	milisaniye
ns	nanosaniye
s	saniye

Kisaltmalar

ABC	Yapay Arı Kolonisi (Artificial Bee Colony)
ACO	Karınca Kolonisi Optimizasyonu (Ant Colony Optimization)
BCE	İkili Çapraz Entropi (Binary Cross Entropy)
BM	Bulanık Mantık
SOC	Şarj Durumu (State Of Charge)
CIoU	Complete IoU
CNN	Evrişimsel Sinir Ağları (Convolutional Neural Network)
ConvLSTM	Evrişimsel LSTM
CUDA	Compute Unified Device Architecture
cuDNN	CUDA Deep Neural Network
DBSCAN	Gürültülü Uygulamaların Yoğunluk Tabanlı Mekansal Kümelenmesi (Density-Based Spatial Clustering of Applications with Noise)
DIoU	Distance IoU
DOPS	Drone Optimized Performance Score
DÖ	Derin Öğrenme
FPS	Saniyedeki Görüntü Sayısı (Frame Per Second)
GA	Genetik Algoritma
GFLOPs	Giga Floating Point Operations per Second
GIoU	Generalized Örtüşme Oranı
GNSS	Küresel Konum Belirleme Sistemi (Global Navigation Satellite Systems)
GPS	Küresel Konumlama Sistemi (Global Positioning System)
GPU	Grafik İşlemci Ünitesi
GSM	Mobil İletişim İçin Küresel Sistem (Global System for Mobile Communications)
IMU	Ataletsel Ölçüm Birimi (Inertial Measurement Unit)
IoU	Örtüşme Oranı (Jaccard index)
ISM	Sınai, Bilimsel ve Tıbbi cihaz (Industrial, Scientific and Medical)
İHA	İnsansız Hava Aracı
İKA	İnsansız Kara Aracı
LiDAR	Işın Algılama ve Mesafe Ölçme (Laser Imaging Detection and Ranging)
LR	Lineer Regresyon
LSTM	Uzan – Kısa Vadeli Bellek (Long Short Term Memory)
mAP	Ortalama Hassasiyet (Mean Average Precision)

MAVLink	Micro Air Vehicle Link
MAVROS	Robot İşletim Sistemi Tabanlı MAVLink (MAVLink on ROS)
MLC	Askeri Yük Sınıflandırması (Military Load Classification)
MÖ	Makine Öğrenmesi
MSE	Hataların karesinin ortalaması (Mean Squared Error)
NMS	Non-Maximum Suppression
NN	En Yakın Komşu (Nearest Neighbor)
OBIA	Nesne Tabanlı Görüntü Analizi
OSRF	Open Source Robotics Foundation
PSO	Parçacık Sürü optimizasyonu (Particle Swarm Optimisation)
RA	Rendezvous Algoritmaları
ReLU	Rectified Linear Unit
RF	Radyo Frekansı (Radio Frequency)
RGB	Kırmızı, Yeşil, Mavi (Red, Green, Blue)
RL	Pekiştirmeli Öğrenme (Reinforcement Learning)
ROS	Robot İşletim Sistemi (Robot Operating System)
RTK	Gerçek Zamanlı Kinematik (Real Time Kinematic)
RTL	Eve Dönüş (Return to Launch)
SAO-CNN	Self-Aktivasyon Optimization Convolutional Neural Network
SAS	Solar Agro Savior
SGD	Stochastic gradient descent
SITL	Yazılım Döngüsünde Simülasyon (Software in the Loop)
SSD	Single Shot MultiBox Detector
SVM	Destek Vektör Makinaları (Support Vector Machine)
TCP	İletim Kontrol Protokolü (Transmission Control Protocol)
TSP	Gezgin Satıcı Problemi (Travelling Salesman Problem)
UDP	Kullanıcı Datagram Protokolü (User Datagram Protocol)
VTOL	Dikey İniş Kalkış Yapan Sabit Kanatlı Drone (Vertical Take-Off and Landing)
YOLO	You Only Look Once

ŞEKİLLER DİZİNİ

	Sayfa
Şekil 2.1 Sürü robotlarının örnek kullanım alanları.	10
Şekil 2.2 Döner kanatlı İHA’larda kullanılan çeşitli sensörler.	13
Şekil 2.3 Döner kanatlı İHA’larda kullanılan haberleşme yöntemlerinden örnekler.	14
Şekil 2.4 Sabit kanatlı İHA’ların kullanım alanları.	18
Şekil 2.5 Pervane sayısına göre İHA çeşitleri.	19
Şekil 2.6 Drone pervane yönlerinin gösterimi.	20
Şekil 2.7 Döner kanatlı İHA’ların kullanım alanları.	22
Şekil 2.8 Dört adet girişi, üç adet gizli katmanı ve iki adet çıkışı olan YSA modeli.	25
Şekil 2.9 CNN’in katmanlı yapısının şematik gösterimi.	27
Şekil 2.10 Görüntü üzerinde YOLO mimarisinin adımları.	36
Şekil 3.1 Gazebo simülasyon arayüzü.	54
Şekil 3.2 ArduPilot’un kullanım alanları.	56
Şekil 3.3 SITL simülasyon çıktıları.	59
Şekil 3.4 MAVLink protokolünün temel bileşenleri.	60
Şekil 3.5 MAVROS protokolünün temsili gösterimi.	64
Şekil 3.6 Sürü drone sistem mimarisi ve bileşenleri.	72
Şekil 3.7 Tasarlanan tarım arazilerinin üstten görünümü.	73
Şekil 3.8 Gazebo’da kullanılan domates bitki modeli.	73
Şekil 3.9 Gazebo’da drone konumlarının üstten görünümü.	74
Şekil 3.10 Kâşif drone üzerinde bulunan kameranın 90° açıdaki görüntüsü.	74
Şekil 3.11 İlaçlama drone’unun ilaçlama öncesi ve ilaçlama anındaki görüntüleri.	75
Şekil 3.12 LiDAR lazer görüntüleri aktif olan drone’ların üstten görünümü.	75
Şekil 3.13 Zirai ilaçlamada kullanılmak üzere tasarlanan sürü drone sisteminin akış diyagramı.	78
Şekil 3.14 Yer istasyonunun görev akış diyagramı.	79
Şekil 3.15 Kâşif drone’un görev akış diyagramı.	80
Şekil 3.16 İlaçlama drone’larının görev akış diyagramı.	81
Şekil 3.17 Eylem fonksiyonunun akış diyagramı.	83
Şekil 3.18 Hedef – Eylem – Veri sıralamasına dayalı mesajlaşma akışı.	84
Şekil 3.19 Sürü drone sistemindeki tarama modülünün akış diyagramı.	86

Şekil 3.20 Domates bitkisinden oluşan veri setinde 5 ve 10 m yükseklikten elde edilen veri seti görüntülerinin etiketlenmiş halleri.	88
Şekil 3.21 Tasarlanan sürü drone sistemindeki görev planlama modülü akış diyagramı.	90
Şekil 4.1 Drone’ların GPS tabanlı canlı konumlarının haritası.	96
Şekil 4.2 Tarama modülü ile oluşturulan görev noktalarının haritası.	97
Şekil 4.3 Arazi 2’de kâşif drone’un tarama işlemi sırasında görev noktalarının turkuaz renge dönüştürülmesi.	98
Şekil 4.4 Tarama sonrası kümeleme ile oluşturulan ilaçlama noktalarının drone’lara göre renklerle ayrıştırılması.	98
Şekil 4.5 İlaçlama işlemi tamamlanan noktaların yeşil renge dönerek gösterilmesi.....	99
Şekil 4.6 Sürü drone sisteminde hem taramanın hem de ilaçlamanın aynı anda yapılması.	99
Şekil 4.7 Acil durumda görev dağılımı.	100
Şekil 4.8 Arazi türlerine göre drone’ların uçuş mesafeleri.	102
Şekil 4.9 Arazi türlerine göre drone’ların havada kalma süreleri.	102
Şekil 4.10 Arazi türlerine göre drone’ların batarya SOC değerleri.....	103
Şekil 4.11 YOLOv3-Tiny modelinin eğitim sürecine ait kayıp grafikleri ve Precision–Recall eğrileri.....	106
Şekil 4.12 Arazi türlerine göre drone’ların uçuş mesafeleri.	109
Şekil 4.13 Arazi türlerine göre drone’ların havada kalma süreleri.	109
Şekil 4.14 Arazi türlerine göre drone’ların ortalama batarya SOC değerleri.....	110

ÇİZELGELER DİZİNİ

	Sayfa
Çizelge 2.1 Literatürdeki sürü drone sistemlerine ilişkin çalışmaların karşılaştırmalı özeti	50
Çizelge 3.1 Gazebo simülasyon programının avantaj ve dezavantajları.....	55
Çizelge 3.2 MAVLink mesajlarının paket formatı.....	61
Çizelge 3.3 Gazebo simülasyon ortamı parametreleri.....	77
Çizelge 3.4 Kullanılan sensörlerin rostopic adresleri.....	84
Çizelge 3.5 Tasarlanan sürü drone sisteminde tanımlanan hedef komutları.....	85
Çizelge 3.6 Tasarlanan sürü drone sisteminde tanımlanan eylemlerin ve varsa verilerin açıklamaları.	85
Çizelge 3.6 (Devam) Tasarlanan sürü drone sisteminde tanımlanan eylemlerin ve varsa verilerin açıklamaları.	86
Çizelge 3.7 Nesne tespiti için YOLOv3 – YOLOv5 mimarilerinde kullanılan eğitim parametreleri.	89
Çizelge 4.1 Acil durum senaryosunda elde edilen veriler.....	94
Çizelge 4.2 Optimizasyon algoritması pasif iken elde edilen sonuçlar.....	101
Çizelge 4.3 Kullanılan YOLO modellerinin eğitim sonuçları.	104
Çizelge 4.4 NN algoritması aktif iken elde edilen sonuçlar.....	107
Çizelge 4.5 ACO optimizasyon algoritması aktif iken elde edilen sonuçlar.....	108

RESİMLER DİZİNİ

Sayfa

Resim 2.1 Baykar TB2 sabit kanatlı İHA.	17
---	----



1. GİRİŞ

Robotik bilimi, son yıllarda hem akademik hem de endüstriyel çevrelerde en hızlı gelişen disiplinlerden biri hâline gelmiştir. Sensör teknolojilerindeki ilerlemeler, gömülü sistemlerin işlem kapasitelerinin artması, yapay zekâ algoritmalarındaki ve iletişim altyapılarındaki gelişmeler, robotların daha karmaşık görevleri otonom biçimde yerine getirebilmesini mümkün kılmaktadır. Modern robotlar, yalnızca önceden programlanmış işlemleri tekrarlayan mekanik sistemler olmaktan çıkarak çevrelerini algılayabilen, durum değerlendirmesi yapabilen ve değişen koşullara uyum sağlayabilen akıllı sistemler hâline gelmiştir. Bu dönüşüm, robotik alanını çok geniş uygulama alanlarına taşımış; endüstriyel otomasyon, sağlık hizmetleri, lojistik, savunma sanayi, tarım, arama-kurtarma ve çevresel izleme gibi alanlarda yaygın biçimde kullanılmasını sağlamıştır (Pietrantoni vd. 2024, Mohammed vd. 2025).

Tek bir robotun kullanıldığı sistemler birçok uygulamada faydalı olmakla birlikte, özellikle geniş alanların taranması, uzun süreli görevlerin yürütülmesi ya da karmaşık ortamların kontrol edilmesi gibi durumlarda yetersiz kalabilmektedir. Tekil robotların sınırlı kapsama alanı ve enerji kısıtları yaptığı görevlerdeki performansı sınırlayan temel unsurlardır. Bu nedenle araştırmacılar, birden fazla robotun eş zamanlı ve koordineli biçimde görev alabileceği sistemlere yönelmişlerdir. Doğadan ilham alınarak geliştirilen sürü robotları yaklaşımı, bu ihtiyaca güçlü bir yanıt sunmakta; görevlerin daha kısa sürede, daha geniş kapsama alanı ile ve daha yüksek güvenilirlik düzeyinde yerine getirilmesine sağlamaktadır (Alqudsi ve Makaraci 2025a).

Sürü robotları, çok sayıda basit bireyin yerel kurallara dayalı olarak hareket etmesiyle ortaya çıkan kolektif bir zekâ anlayışına sahiptir. Her bir robot kendi sensörlerinden elde ettiği bilgileri değerlendirerek karar alırken sürünün bütünsel davranışı bu yerel etkileşimler sonucunda ortaya çıkar. Bu yaklaşım, doğadaki karınca kolonileri, balık sürüleri veya kuş topluluklarından esinlenerek geliştirilmiştir. Sürü robotlarının en önemli avantajı, görevlerin tek bir robota kıyasla çok daha kısa sürede ve daha geniş bir kapsama alanında tamamlayabilmesidir. Sürü içindeki herhangi bir robotun devre dışı kalması görev bütünlüğünü bozmaz çünkü diğer robotlar kalan işi üstlenebilir. Bu

özellikler, sürü robotlarını özellikle dinamik ve belirsiz ortamlarda tekil sistemlere kıyasla çok daha güvenilir, esnek ve verimli hâle getirmektedir. Günümüzde bu nedenle sürü robotları, askeri operasyonlardan arama-kurtarma faaliyetlerine, çevresel izleme çalışmalarından tarımsal uygulamalara kadar geniş bir alanda kullanılmaktadır (Nguyen 2024).

Tarımsal üretim, sürü robotlarının en fazla öne çıktığı uygulama alanlarından biri olarak dikkat çekmektedir. Artan dünya nüfusu ve tarıma ayrılan iş gücündeki azalma, gıda üretiminde verimliliğin artırılmasını zorunlu kılmaktadır. İklim değişikliğinin getirdiği çevresel baskılar da bu ihtiyacı güçlendirmektedir. Ayrıca aşırı kimyasal kullanımının toprak ve su kaynakları üzerindeki olumsuz etkileri, tarımsal süreçlerde daha kontrollü ve sürdürülebilir yöntemlere olan ihtiyacı her geçen gün artırmaktadır. Bu çerçevede, otomasyona dayalı çözümler sadece verimlilik açısından değil aynı zamanda çevresel etkilerin azaltılması bakımından da kritik öneme sahiptir. Özellikle drone'ların sürü hâlinde kullanımı, geniş tarım arazilerinin kısa sürede taranmasına, hastalık ve zararlıların erken tespitine ve yalnızca gerekli bölgelere hassas ilaçlama yapılmasına olanak vermektedir. Böylece hem üretim verimliliği artırılmakta hem de çevreye verilen zarar minimum düzeye indirilmektedir.

Bu tez çalışmasında, tarımsal alanlarda kullanılmak üzere sürü drone sistemine yönelik bütünleşik bir mimari geliştirilmiştir. Önerilen yapıda üç drone görev almaktadır: bunlardan biri keşif amacıyla arazide tarama yaparken diğer ikisi tarama sırasında belirlenen hedefler üzerinde ilaçlama gerçekleştirmektedir. Sistem, Küresel Konumlama Sistemi (Global Positioning System – GPS), Işın Algılama ve Mesafe Ölçme (Laser Imaging Detection and Ranging – LiDAR) ve kamera sensörleriyle desteklenmiş olup engel algılama ve çarpışma önleme kabiliyeti kazandırılmıştır. Kamera verileri, YOLO (You Only Look Once) tabanlı nesne tespit algoritması ile işlenmekte ve elde edilen koordinatlar görev planlama modülüne aktarılmaktadır. Görev paylaşımı sürecinde tasarlanan fonksiyon ile drone'lara dengeli görev dağılımı yapılmaktadır. Görev sürecinde en uygun rotaların belirlenmesi için optimizasyon yöntemlerinden yararlanılmış, böylece drone'ların havada kalma süresi ve yol aldıkları mesafe azaltılmıştır. Ayrıca bir drone'un devre dışı kalması gibi durumlarda kalan görevlerin diğer drone'a aktarılabilmesiyle sistemin güvenilirliği artırılmıştır.

Bu alıřmanın nemi, tarama, nesne tespiti, ilalama, grev planlama ve optimizasyon gibi farklı bileřenleri tek bir modler mimaride birleřtirmesidir. Sistem Gazebo simlasyon ortamında modellenmiř ve tarımsal uygulamaları temsil eden senaryolarda test edilmiřtir. Elde edilen bulgular, sr drone kullanımının tekil drone uygulamalarına gre grevlerin tamamlanma sresini kısalttıđını, kaynak kullanımını daha dengeli hle getirdiđini ve grev gvenilirliđini artırdıđını ortaya koymuřtur.

Bu tez alıřması beř ana blmden oluřmakta olup tezin ikinci blmde sr robotiđi, İnsansız Hava Araları (İHA), Yapay Zek (YZ) ve Derin đrenme (D) konularında teorik bilgilere yer verilmiř; ayrıca ulusal ve uluslararası alıřmalardan rnekler sunulmuřtur. nc blmde simlasyon ortamı ve sistem mimarisi ayrıntılı biimde ele alınarak, yapay zek destekli sr drone yapısının tasarımı anlatılmıřtır. Drdnc blmde elde edilen bulgular sunulmuř, son blmde ise bu bulgular deđerlendirilerek sonu ve neriler ortaya konulmuřtur.

2. LİTERATÜR BİLGİLERİ

Bu başlıkta sürü robotiği, İHA ve YZ'nin tarihsel gelişimi hakkında bilgilendirme yapılmıştır. Ayrıca konu ile ilgili ulusal ve uluslararası alanlarda yapılan akademik literatür çalışmaları incelenerek çalışmanın literatürdeki yeri değerlendirilmiştir.

2.1 Sürü Robotiği

Bu bölümde öncelikle sürü robotiğinin temel kavramları ve özellikleri ele alınmakta, ardından iletişim ve koordinasyon yöntemleri üzerinde durulmaktadır. Daha sonra sürü robotlarında kullanılan davranış modelleri ve kontrol yaklaşımları incelenerek farklı senaryolara uygun stratejilerin avantaj ve sınırlılıkları tartışılmaktadır. Son olarak sürü robotlarının askerî operasyonlardan tarımsal faaliyetlere kadar uzanan geniş uygulama alanları örneklerle açıklanmaktadır. Böylece sürü robotiğinin kavramsal çerçevesi, teknik altyapısı ve pratik kullanım alanları bütüncül bir bakış açısıyla ortaya konulmaktadır.

2.1.1 Sürü Robotiği ve Temel Özellikleri

Doğada canlıların hayatta kalabilmeleri için kendi aralarında yapmış oldukları çeşitli etkileşimler bulunmaktadır. Bu etkileşimler canlı türlerine göre farklılık göstermektedir. Karıncaların besin bulmak için kendi aralarında feromonlar ile hareket etmeleri, balıkların bir arada hareket ederek kendilerini tehlikelerden korumaları gibi birçok örnek verilebilir (Mısırs ve Gökrem 2020). Bilim insanları bu gibi çeşitli örneklerden esinlenerek bu davranışları robotik alana taşımışlar ve sürü robotiğini oluşturmuşlardır (Sumpter 2010). Sürü robotları basit fiziksel yapıya sahip olan çok sayıdaki robotun kendi aralarında yerel haberleşme yaparak bir bütün halinde hareket edebilmesi şeklinde tanımlanabilir. Bu robotların basit, uygun maliyetli, esnek ve sağlam yapıları en önemli özellikleri arasındadır. Ayrıca kendi kendini örgütleme, özerklik, iş birliği ve koordinasyon yetenekleri de bulunmaktadır.

Sürü robotlarının tek bir robot kullanımına kıyasla en belirgin avantajı, görevlerin daha kısa sürede ve daha geniş bir alanda tamamlanabilmesidir. Birden fazla robotun aynı anda

farklı bölgelerde çalışabilmesi, özellikle büyük ölçekli tarım arazilerinde, arama-kurtarma çalışmalarında ya da keşif görevlerinde kullanılması bu görevlerin yerine getirilmesi sırasında ciddi bir zaman kazancı sağlamaktadır. Bunun yanı sıra, sürü içindeki herhangi bir robotun arızalanması ya da devre dışı kalması durumunda görevlerin diğer robotlara aktarılabilmesi sistemin güvenilirliğini artırmaktadır. Böylece hem karmaşık ve hem de değişken koşullarda daha esnek çözümler üretilebilmektedir.

Doğadan esinlenerek oluşturulan davranışların sergilenmesi için sürü robotlarında bulunması gereken çeşitli özellikler vardır. Bu özelliklerden bazıları her sürü robotunda olması gerekmekte iken bazı özellikler ise verilecek olan görevlere göre robotlara entegre edilmektedir. Her sürü robotlarında olması gereken temel özellikler (Dorigo vd. 2021);

Esneklik, robotun verilen görevi basit bir şekilde yerine getirebilmesi, bir görev değişikliği olduğunda ona uyum sağlayarak yeni görevini yerine getirebilmesi olarak tanımlanabilir. Arızalanan ya da bozulan robotun yerine yeni bir robotun gelmesini veya mevcut drone'lara görev yükünü tekrar dağıtarak hareket etmeleri buna örnek olarak verilebilir.

Ölçeklenebilirlik, farklı sayıdaki robotların beraber çalışabilme özelliğidir. Robot sayısındaki değişiklik sürünün performansını olumsuz etkilememelidir. Robot sayısı az olduğunda verilen görevin yerine getirilmesi gerekmekte, çok olduğunda da görev paylaşımı yapılarak yine verilen görevler de yerine getirilmelidir.

Sağlamlık, çevresel koşullar karşısında robotun bu koşullara direnç göstererek verilen görevi devam ettirebilme yeteneğine denir. Bu koşullar çevrenin değişiklik göstermesi, ortamdaki engel sayısının artması, hava değişimi gibi çeşitlendirilebilir.

Bu temel özelliklerin dışında sürü robotlarında yaygın olarak kullanılan çeşitli özellikler de bulunmaktadır. Ancak bu özellikler her robota entegre edilen özellikler olmayıp verilecek olan göreve göre entegre edilebilen özelliklerdir. Bunlar;

Otonom özelliği sayesinde robotların çevresindeki durumlara göre kendilerinin karar vererek otonom bir şekilde hareket etme özelliğidir. Bu özellikleri sayesinde merkezi bir kontrol biriminden yönetilmelerine gerek kalmamaktadır.

Kendi kendine organize olma özelliği sayesinde sürü robotiğine verilen bir görevde robotlar kendi aralarında organize olarak hareket edebilmektedirler. Dışarıdan herhangi bir müdahale olmadan robotlar bu süreci kendi kendine yürütmektedirler.

Kendi kendine birleşme özelliği sayesinde yapılacak olan göreve göre robotların kendi kendine birleşerek bir bütün olarak hareket etmesidir. Tek bir robotun başarılı olamayacağı görevlerde bu özelliğe sahip olan sürü robotlarının birleşerek hareket etmesi ve verilen görevi yerine getirmesi örnek olarak verilebilir.

Merkezi kontrolü olmayan özelliği sayesinde robotların herhangi bir merkezden kontrolü yapılmamaktadır. Bunun yerine robotlar o anki çevre koşullarına ve göreve göre kendileri karar vererek görevi tamamlamaktadırlar. Bu özellik sayesinde herhangi bir kontrol istasyonuna ihtiyaç duyulmamakta ve olası insan hatalarının önüne geçilebilmektedir.

Stigmerji özelliği ise robotların dolaylı iletişimini ifade etmektedir. Karıncaların besin kaynaklarına giderken bıraktıkları feromonlar gibi robotların da kendi aralarında kurmuş oldukları dolaylı iletişim özelliğidir.

2.1.2 Sürü Robotlarında İletişim ve Koordinasyon Yöntemleri

Sürü robotlarının ortak bir hedef doğrultusunda verimli biçimde görev yapabilmeleri için en önemli gerekliliklerden birisi de iletişim ve koordinasyon mekanizmalarıdır. Her robotun yalnızca kendi sensör verisine dayanarak karar alması çoğu zaman yeterli değildir. Robotların bilgi paylaşımı ve hareketlerinin uyumlu olması, görevin başarısını doğrudan artırmaktadır. Bu nedenle sürü robotik sistemlerinde kullanılan iletişim stratejileri, sistemin ölçeklenebilirliği, hata toleransı ve adaptasyon yeteneği üzerinde etkili bir rol oynamaktadır. Robotların haberleşmesinde doğrudan iletişim yöntemleri, dolaylı iletişim yöntemleri ve lider – takipçi model yöntemleri kullanılmaktadır.

Doğrudan iletişim yöntemleri, sürü robotlarının en yaygın kullandığı haberleşme yöntemi olup koordinasyonun daha kolay yapıldığı yöntemdir. Bu yöntemde robotlar, kablosuz haberleşme teknolojileri (Wi-Fi, ZigBee, LoRa vb.) aracılığıyla konum, hız, batarya durumu ya da görev bilgisi gibi verileri birbirine iletirler. Bu iletim esnasında mesaj tabanlı protokollerin kullanımı (örneğin Robot İşletim Sistemi (Robot Operating System – ROS)’un yayıncı/izleyici altyapısı veya Micro Air Vehicle Link (MAVLink) protokolü) bu süreci kolaylaştırır. Doğrudan iletişim yöntemi, anlık ve kesin bilgi aktarımı sağlaması açısından önemli avantaj sağlamaktadır. Ancak iletişim hattında yaşanabilecek kesintiler, paket kayıpları veya oluşan yoğun ağ trafiği dezavantaj oluşturabilmekte ve sürünün davranışını olumsuz etkileyebilmektedir.

Dolaylı iletişim (stigmerji) yönteminde robotların doğrudan birbirleriyle iletişim kurmalarına gerek yoktur. Robotlar çevreye bıraktıkları işaretler üzerinden haberleşirler (Hunt vd. 2019). Bu yaklaşım, doğadaki karınca kolonileri veya arı topluluklarından esinlenerek geliştirilmiştir. Örneğin bir robot keşfettiği hedef konumunu ortak bir harita üzerine işaretleyebilir ve diğer robotlar bu işareti takip ederek koordineli hareket edebilir. Dolaylı iletişim tabanlı yöntemler, sınırlı haberleşme altyapısı gerektirdiğinden sistemler daha dayanıklı olmaktadır. Ancak bilgi aktarımı dolaylı gerçekleştiğinden tepki süresi doğrudan iletişime kıyasla daha yavaştır.

Sürü robotlarında kullanılan bir diğer önemli yöntem ise lider – takipçi modelidir. Bu modelde tek bir robot (lider robot) sürünün yönelimini belirler ve diğer robotlar bu lideri takip ederek hareket ederler. Bu yöntem koordinasyonu basitleştirerek sürünün kontrolünü kolaylaştırmaktadır. Ancak olası bir lider robotun arızalanması durumunda sistemin bütünlüğü bozulabilmektedir. Sonuç olarak sürü robotlarında iletişim ve koordinasyon yönteminin seçimi robotlara verilecek olan görevin gerekliliklerine, görev yapılacak olan çevrenin karmaşıklığına ve robotlarda kullanılan donanım altyapısına bağlı olarak değişmektedir. Gerçek zamanlı, yüksek hassasiyet gerektiren görevlerde doğrudan iletişim tercih edilirken geniş ölçekli ve dinamik ortamlarda dağıtık yapıların avantajları ön plana çıkmaktadır. Bu tez kapsamında geliştirilen sistemde ise drone’lar arası koordinasyon, doğrudan iletişim yaklaşımı çerçevesinde ROS ve MAVLink tabanlı mesaj protokolleri kullanılarak gerçekleştirilmiştir.

2.1.3 Davranış Modelleri ve Kontrol Yaklaşımları

Sürü robotik sistemlerinde, robotların uyum içinde çalışabilmesi için uygun davranış modelleri ve kontrol stratejilerinin geliştirilmesi gerekmektedir. Tek bir merkezden yönetim yerine, bireylerin yerel bilgilerini kullanarak ortak bir amaca yönelmeleri, sürü sistemlerinin esnekliğini ve hata toleransını artırmaktadır. Bu bağlamda literatürde, farklı biyolojik sistemlerden esinlenen ya da matematiksel olarak modellenen çeşitli yaklaşımlar geliştirilmiştir. İşaret işleme tabanlı yöntemler (Hunt vd. 2019, Talamali 2020, Sadeghi Amjadi vd. 2022) ve öz-örgütlenebilir kontrol yöntemleri öne çıkmaktadır. Öz-örgütlenebilir kontrol çerçevesinde olasılıksal modeller (Katada 2018, Firat vd. 2020), kontrol algoritmaları (Parhizkar vd. 2020, Salimi 2021), yapay fizik tabanlı yaklaşımlar (Khaldi vd. 2019) ve evrimsel algoritmalar (Soysal ve Şahin 2006) sürü robotlarının davranışlarını şekillendirmede öne çıkan başlıca yöntemlerdir. Bu yöntemlerin her biri, sürü içi koordinasyonun nasıl sağlandığı ve görev başarımının hangi ilkeler üzerinden yürütüldüğü konusunda farklı avantajlar ve sınırlılıklar sunmaktadır.

İşaret işleme tabanlı yaklaşımda, robotlar birbirleri ile belirli sinyaller veya mesajlar üzerinden doğrudan bilgi paylaşırlar. Kablosuz haberleşme, konum verileri veya durum bilgileri kullanılarak sürüde koordinasyon sağlanır. Hızlı ve net bilgi aktarımı yöntemin avantajı iken haberleşme altyapısına bağımlılık, bant genişliği sınırlamaları ve iletişim kesintileri gibi sorunlar dezavantajdır.

Öz-örgütlenebilir kontrol yaklaşımı ise, robotların basit yerel kurallar doğrultusunda hareket ederek merkezi bir kontrol olmadan küresel ölçekte istenilen düzenin ortaya çıkmasını sağlar (Khaldi vd. 2019). Bu yaklaşım doğadaki karınca, kuş veya balık sürülerinden esinlenerek geliştirilmiştir. Yüksek esneklik ve ölçeklenebilirlik en önemli avantajlarıdır. Ancak sistem davranışının çoğu zaman deterministik olmaması ve tahmin edilememesi dezavantaj oluşturmaktadır. Öz-örgütlenebilir kontrolün altında farklı alt yaklaşımlar bulunmaktadır. Bunlar;

Olasılıksal yaklaşımda, robotlar kararlarını olasılık dağılımlarına dayalı olarak verir. Rastgelelik kullanılarak sürünün farklı bölgelerde keşif yapması veya görev paylaşımı

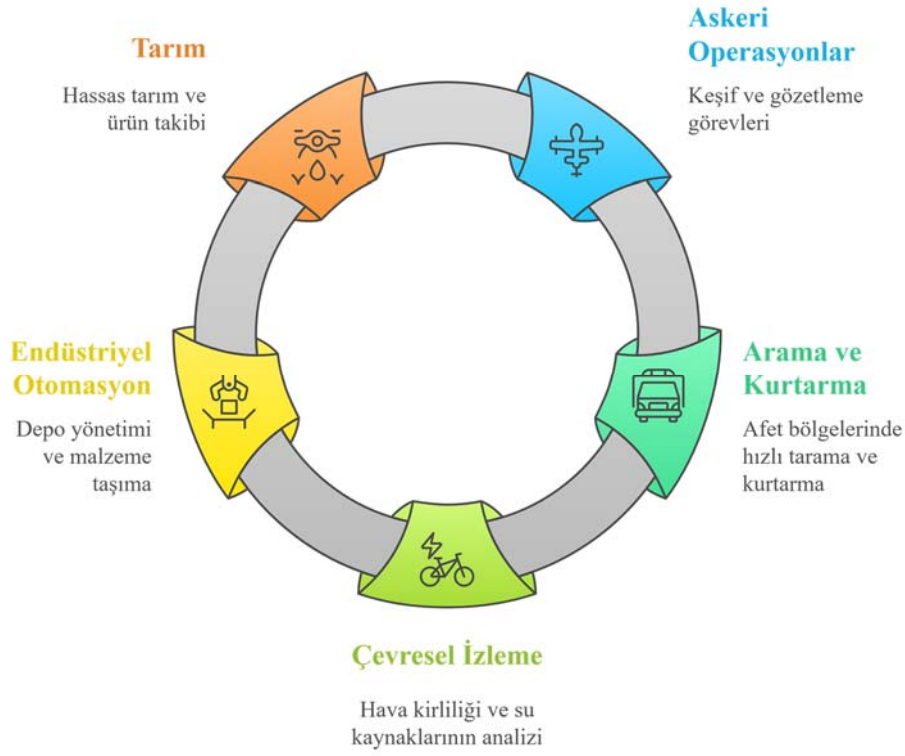
çeşitlenir. Özellikle arama-kurtarma veya keşif görevlerinde yaygın olarak kullanılmaktadır. Ancak yöntemin performansı her zaman optimum olmayıp olasılıklı yapı nedeniyle verimlilik değişkenlik gösterebilmektedir.

Yapay fizik tabanlı yaklaşımda robotlar, birbirlerine veya çevredeki nesnelere sanal çekim/itme kuvvetleri uyguluyormuş gibi modellenir. Çarpışma önleme, sürü bütünlüğünün korunması ve formasyon hareketleri için oldukça kullanışlıdır. Hesaplama açısından basittir. Fakat karmaşık görevlerde tek başına yeterli gelmemekte ve parametre seçimi önemli rol oynamaktadır.

Evrimsel algoritmalar, doğadaki evrim sürecinden esinlenerek geliştirilmiş optimizasyon ve problem çözme yöntemleridir. Bu algoritmaların temelinde, bir çözüm kümesinin "popülasyon" olarak ele alınması ve her bir aday çözümün "birey" olarak değerlendirilmesi yaklaşımı vardır. Çözümler, seçim, çaprazlama ve mutasyon gibi biyolojik evrimi taklit eden işlemler aracılığıyla yeni nesillere aktarılır. Amaç her yeni nesilde daha iyi çözümlere ulaşmak ve küresel optimuma yaklaşımdır. Evrimsel algoritmaların en önemli avantajı, çözüm uzayında geniş bir arama yapabilmeleri ve karmaşık, doğrusal olmayan problemlerde bile etkin sonuçlar üretebilmeleridir. Ancak yüksek hesaplama maliyeti ve uygun parametre ayarlarının belirlenmesindeki zorluklar, bu yöntemlerin uygulamada karşılaşılan temel sınırlamalarıdır.

2.1.4 Sürü Robotlarının Uygulama Alanları

Sürü robotları, sahip oldukları ölçeklenebilirlik, hata toleransı ve esneklik özellikleri sayesinde pek çok farklı alanda kullanılabilmektedir. Bu uygulama alanları arasında askerî operasyonlar, arama-kurtarma görevleri, tarımsal faaliyetler, çevresel izleme ve sürdürülebilirlik çalışmaları, endüstriyel otomasyon, lojistik, denizcilik ve su altı araştırmaları sayılabilir (Nguyen 2024, Alqudsi ve Makaraci 2025b). Bu alanların her birinde sürü robotlarının farklı özelliklerinden yararlanılmakta, tek robotun yetersiz kaldığı durumlarda sürü yapılarının avantajları ön plana çıkmaktadır. Şekil 2.1’de sürü robotlarının örnek kullanım alanlarının görseli verilmiştir.



Şekil 2.1 Sürü robotlarının örnek kullanım alanları.

Askerî uygulamalarda sürü robotları, keşif ve gözetleme görevlerinden mayın temizleme ve hedef takip operasyonlarına kadar geniş bir yelpazede kullanılmaktadır. Çoklu robotların eş zamanlı hareket edebilmesi, riskin dağıtılmasını sağlayarak görev güvenliğini artırmaktadır. Afet durumlarında ise sürü robotlarının arama-kurtarma faaliyetlerindeki önemi büyüktür. Deprem, yangın veya sel gibi felaketlerde, geniş alanların hızlı taranması ve kritik noktalara kısa sürede ulaşılması sürü robotları sayesinde mümkün olmaktadır. Çevresel izleme uygulamalarında sürü robotları, hava kirliliğinin ölçümü, orman yangınlarının tespiti veya su kaynaklarının analizi gibi görevlerde kullanılmaktadır. Lojistik ve endüstri sektöründe, depo yönetiminden otonom malzeme taşımaya kadar pek çok alanda sürü sistemlerinin kullanımı giderek yaygınlaşmaktadır.

Tarımsal alanda ise bu teknolojiler, özellikle hassas tarım uygulamalarında öne çıkmaktadır. Tarımsal uygulamalar, sürü robotlarının en hızlı gelişim gösterdiği ve pratikte en fazla fayda sağladığı alanlardan biridir. Geleneksel yöntemlerle uzun zaman ve yoğun iş gücü gerektiren ilaçlama, gübreleme, tohum ekimi ve ürün takibi gibi işlemler, sürü hâlinde çalışan drone ve kara robotları sayesinde kısa sürede ve minimum

insan müdahalesi ile gerçekleştirilebilmektedir. Bu sayede hem zamandan hem de maliyetten tasarruf sağlanırken aynı zamanda gereksiz su veya kimyasal kullanımının önüne geçilerek sürdürülebilir tarım anlayışına katkı sunulmaktadır. Özellikle sürü drone sistemleri, geniş tarım arazilerinin eş zamanlı olarak taranmasına, hastalık veya zararlıların erken dönemde tespit edilmesine ve yalnızca ihtiyaç duyulan noktalara ilaçlama yapılmasına imkân tanımaktadır. Böylece ürün verimliliği artırılmakta, çevresel zararlar azaltılmakta ve artan nüfusun gıda ihtiyacına karşı daha etkin çözümler üretilmektedir.

2.2 İnsansız Hava Araçları

İHA, günümüzde hem sivil hem de askerî alanlarda en hızlı gelişen teknolojilerden biridir. Uçuş denetleyicileri, sensörler, haberleşme sistemleri ve görev yükleriyle donatılmış bu araçlar, uzaktan kumandalı ya da tamamen otonom olarak görev icra edebilmektedir. Enerji verimliliğinden manevra kabiliyetine kadar farklı tasarımlar sunan sabit kanatlı ve döner kanatlı modeller, uzun menzilli keşif ve haritalama çalışmalarından noktasal tarımsal ilaçlamaya, afet yönetiminden lojistiğe kadar geniş bir uygulama yelpazesine sahiptir. Bu bölümde İHA'ların temel bileşenleri, farklı aerodinamik yapıları ve kullanım alanları ayrıntılı biçimde ele alınmıştır.

2.2.1 İnsansız Hava Araçlarının Temel Bileşenleri

Bir İHA'nın performansını ve görev başarısını belirleyen en kritik unsur sahip olduğu donanım bileşenleridir. Gövde yapısı, pervane sistemi, sensörler, haberleşme modülleri ve uçuş kontrolcüsü gibi parçalar, aracın uçuş süresinden manevra kabiliyetine, çevresel algılamadan veri aktarımına kadar tüm işlevlerini doğrudan etkilemektedir. Bu bileşenlerin her biri farklı görevlerde farklı avantajlar sunmakta, sistemin genel mimarisine uygun olarak seçilip entegre edilmektedir. Aşağıda İHA'ların gövde ve pervane sistemlerinden başlayarak sensörler, haberleşme sistemleri ve uçuş kontrolcüləri ayrıntılı biçimde ele alınmaktadır.

2.2.1.1 Gvde ve Pervane Sistemleri

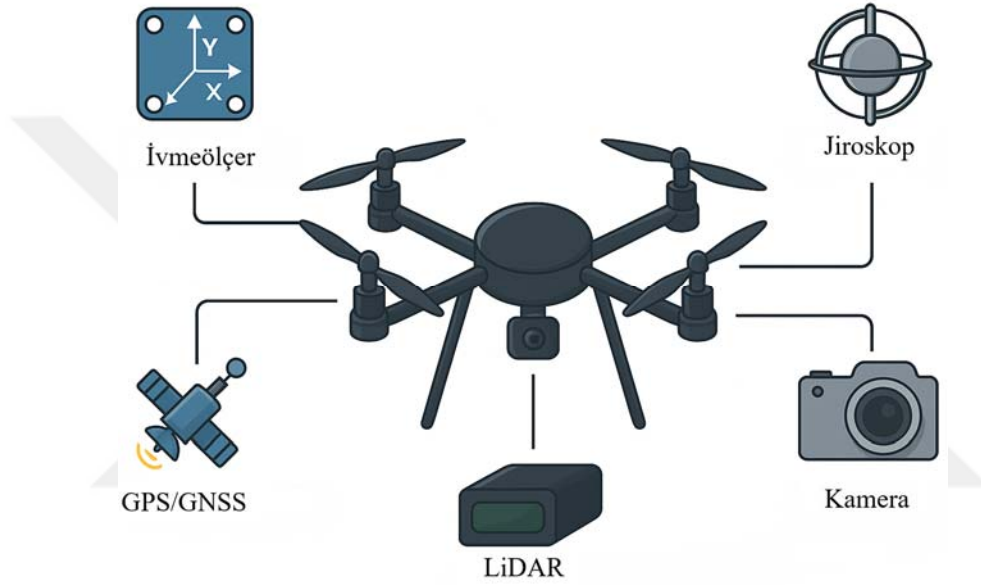
İHA'ların gvde yapısı ve pervane sistemleri, uçuş performansını doğrudan etkileyen en temel unsurlardan biridir. Gvde malzemesi genellikle hafiflik ve dayanıklılığı bir arada sunan karbon fiber, kompozit veya alüminyum alaşımlarından seçilmektedir. Hafif gvde, batarya kapasitesinin daha verimli kullanılmasını sağlarken sağlam yapılar da rüzgâr, titreşim ve ani manevralardan kaynaklanabilecek deformasyonlara karşı direnç kazandırmaktadır (Andries ve Zaharia 2025). Pervaneler ise İHA'nın kaldırma kuvvetini ve itiş gücünü üreten kritik parçalardır. Pervane sayısı ve boyutu, aracın manevra kabiliyeti, taşıyabileceği yük ve uçuş süresi üzerinde belirleyici rol oynamaktadır. Quadcopterler (dört pervaneli) dengeli bir yapı sunarken hexacopterler (altı pervaneli) ve octocopterler (sekiz pervaneli) daha ağır yükleri taşıma ve bir motor arızasında dahi uçuşa devam edebilme avantajına sahiptir. Ayrıca pervane geometrisi ve kullanılan malzeme, uçuş sırasında oluşan gürültü seviyesini, titreşim miktarını ve enerji tüketimini doğrudan etkilemektedir (Chakchouk vd. 2023). Dolayısıyla gvde ve pervane seçimi, İHA'nın görev gereksinimlerine uygun şekilde optimize edilmesi gereken temel bir tasarım aşamasıdır.

2.2.1.2 Sensörler

İHA'ların güvenli ve otonom şekilde görev icra edebilmesi, sahip oldukları sensörlerin sağladığı verilerin doğru ve güvenilir biçimde işlenmesine bağlıdır. Sensörler, aracın çevresini algılamasına, konumunu belirlemesine, uçuş stabilitesini korumasına ve görev sırasında gerekli verileri toplamasına olanak tanır. İHA'lar genellikle farklı sensörlerin birlikte kullanıldığı (Şekil 2.2) bir yapıya sahiptir ve bu sayede sensör füzyonu ile daha doğru veriler elde ederler (Sun vd. 2024).

Ataletsel Ölçüm Birimi (Inertial Measurement Unit – IMU), içerisinde ivmeölçer, jiroskop ve manyetometre sensörlerini barındırarak aracın ivmesini, açısal hızlarını ve yönelimini ölçer. Bu veriler uçuş denetleyicisi tarafından sürekli işlenerek aracın dengede kalmasını ve pilot tarafından veya otonom olarak verilen komutlara hızlı tepki vermesini sağlar. Konum belirleme için kullanılan GPS/ Küresel Konum Belirleme Sistemi (Global

Navigation Satellite Systems – GNSS) alıcıları, aracın dünya üzerindeki koordinatlarını m altı hassasiyetle belirler. Gerçek Zamanlı Kinematik (Real Time Kinematic – RTK) gibi gelişmiş GPS çözümleri ile bu hassasiyet cm'lere düşmektedir. Hassas konum bilgisine ihtiyaç olunan alanlarda sıklıkla kullanılmaktadır (Pire vd. 2019). Barometre sayesinde irtifa bilgisi elde edilmektedir. Basınç değişimlerine duyarlı olan bu sensörler, aracın deniz seviyesine göre yüksekliğini belirlemede kullanılır. Manyetometre ise dünya manyetik alanını referans alarak aracın yönelim açısını belirler.

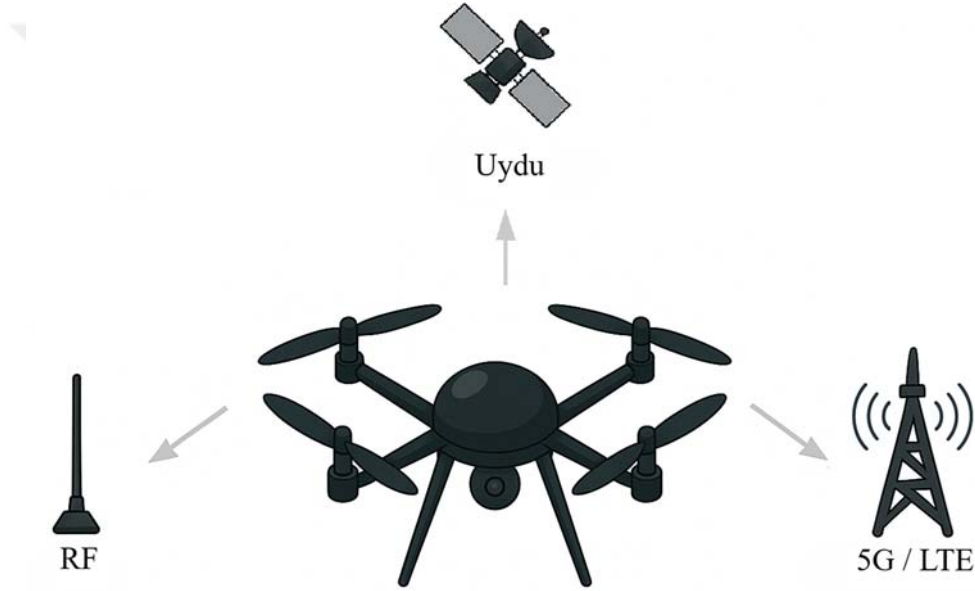


Şekil 2.2 Döner kanatlı İHA’larda kullanılan çeşitli sensörler.

Çevresel algılamada kameralar ve LiDAR sensörleri öne çıkmaktadır. Kameralar, yalnızca görsel bilgi sağlamakla kalmaz, aynı zamanda nesne tespiti, görüntü işleme tabanlı navigasyon ve haritalama görevlerinde kullanılır. Multispektral ve termal kameralar, özellikle tarımsal uygulamalarda bitki sağlığının izlenmesi ve su stresinin belirlenmesinde büyük avantaj sunmaktadır (Guebsi vd. 2024). LiDAR sensörleri ise lazer darbeleri ile çevrenin üç boyutlu haritasını çıkarır, engellerin mesafesini yüksek hassasiyetle ölçer ve çarpışma önleme algoritmalarına veri sağlar. Dolayısıyla farklı sensörlerin sağladığı bu veriler uçuş kontrol algoritmaları ve görev planlama sistemleri için temel girdiyi oluşturur. Otonom görevlerin başarısı, bu sensörlerden elde edilen verilerin doğruluğu ve gerçek zamanlı işlenebilme kapasitesiyle doğrudan ilişkilidir.

2.2.1.3 Haberleşme Sistemleri

İHA'ların haberleşme sistemleri, uçuş güvenliği ve görev başarımı açısından kritik öneme sahiptir. Haberleşme altyapısı, hem yer kontrol istasyonu ile İHA arasındaki komut ve telemetri verilerinin aktarımını hem de görev sırasında elde edilen sensör verilerinin anlık olarak iletilmesini sağlar. Haberleşme sisteminin güvenilirliği, kapsama alanı, veri iletim hızı ve gecikme süreleri, aracın otonom veya yarı otonom operasyon kabiliyetlerini doğrudan etkiler. Bir İHA'da kullanılan haberleşme yöntemlerinden örnekler Şekil 2.3'te gösterilmiştir.



Şekil 2.3 Döner kanatlı İHA'larda kullanılan haberleşme yöntemlerinden örnekler.

Kısa ve orta mesafeli görevlerde en yaygın kullanılan haberleşme altyapısı Radyo Frekans (RF) tabanlı sistemlerdir. 2.4 GHz ve 5.8 GHz gibi Sınai, Bilimsel ve Tıbbi cihaz (Industrial, Scientific and Medical – ISM) bantlarında çalışan bu sistemler, düşük gecikme süreleri ile anlık kontrol imkânı sunar. Telemetri modülleri sayesinde uçuş parametreleri, pil durumu, GPS koordinatları ve sensör verileri yer istasyonuna gerçek zamanlı olarak aktarılır. RF tabanlı sistemlerin en önemli avantajı düşük gecikme süreleri olsa da menzil sınırlamaları ve parazitlenme riski dezavantaj oluşturmaktadır (Boccardo vd. 2021).

Son yıllarda mobil iletişim teknolojilerinin gelişmesiyle birlikte 4G/5G tabanlı haberleşme sistemleri İHA’larda giderek daha fazla kullanılmaya başlanmıştır. Bu sistemler, geniş kapsama alanı ve yüksek bant genişliği sayesinde özellikle gerçek zamanlı video aktarımı ve bulut tabanlı veri işleme gibi görevlerde önemli avantajlar sunar (Qasim ve Jawad 2024). 5G teknolojisi, ultra düşük gecikme süresi ve çoklu cihaz bağlantısı desteği ile sürü drone uygulamaları için oldukça uygun bir altyapı sağlamaktadır (Singh vd. 2022).

Daha uzun menzilli operasyonlarda ise uydu haberleşme sistemleri devreye girmektedir. Uydu tabanlı bağlantılar, küresel kapsama alanı sağlayarak İHA’ların Mobil İletişim İçin Küresel Sistem (Global System for Mobile Communications – GSM) altyapısının bulunmadığı bölgelerde dahi güvenli bir şekilde çalışmasına imkân tanır. Ancak bu sistemler, yüksek gecikme süreleri ve maliyet dezavantajına sahiptir. Özellikle askeri ve sivil keşif görevlerinde, deniz aşırı operasyonlarda veya afet bölgelerinde uydu haberleşmesi kullanılmaktadır.

Sonuç olarak İHA’larda kullanılacak haberleşme altyapısı görev senaryosuna göre seçilmektedir. Kısa mesafeli düşük maliyetli görevlerde RF tabanlı çözümler yeterli olurken yüksek veri aktarımı gerektiren veya geniş alanlarda gerçekleştirilen operasyonlarda 4G/5G ve uydu tabanlı sistemler tercih edilmektedir.

2.2.1.4 Uçuş Kontrolcüler ve Otomatik Pilot Sistemleri

İHA’ların güvenli, kararlı ve otonom şekilde uçabilmesinde uçuş kontrolcüler en kritik donanımlardan biridir. Uçuş kontrolcüsü, IMU, GPS, barometre ve manyetometre gibi sensörlerden gelen verileri gerçek zamanlı olarak işleyerek aracın konumunu, hızını ve yönelimini tahmin eder (Hashim vd. 2023). Bu veriler ışığında motorlara gerekli komutlar üretilir ve aracın stabil bir şekilde uçuşması sağlanır. Uçuş kontrolcüler kararlılık, yönelim ve konum kontrolü olmak üzere üç ana kontrol döngüsünü yönetirler.

Modern uçuş kontrol sistemlerinde hem manuel kullanım hem de otonom kullanım imkânı bulunmaktadır. Otonom uçuş modunda önceden tanımlanan görevler uçuş

kontrolcüsü tarafından otomatik olarak gerçekleştirilebilmektedir. Bu kontrol sistemlerinde görev planlama, belirli GPS noktalarına yönelme, belirli irtifada sabit kalma, Eve Dönüş (Return to Launch – RTL) ve önceden tanımlı rotaları izleme gibi fonksiyonlar standart hâle gelmiştir. Ayrıca güvenlik amacıyla bağlantı kesildiğinde otomatik iniş veya RTL gibi failsafe mekanizmaları devreye girerek aracın kaybolması ya da kontrol dışı kalmasının önüne geçilebilmektedir.

Otomatik pilot sistemlerine örnek olarak verilebilecek olan ArduPilot ve PX4 açık kaynaklı otomatik pilot yazılımları, farklı drone tipleri için uçuş kontrol algoritmaları sunmaktadır. Dünya çapında geniş kullanıcı toplulukları sayesinde sürekli geliştirilmekte olan bu sistemler hem simülasyon ortamlarında (Yazılım Döngüsünde Simülasyon (Software in the Loop – SITL), Gazebo) hem de gerçek donanım üzerinde aynı yapı ile çalıştırılabilmektedir.

2.2.2 Sabit Kanatlı İnsansız Hava Araçları

Sabit kanatlı İHA'lar, uçaklara benzer aerodinamik yapıya sahip olup kaldırma kuvvetini sabit kanat yüzeylerinden elde ederler. İleri doğru itiş, genellikle pervaneler ile sağlanır. Bu tasarım, sabit kanatlı sistemlere yüksek hız, uzun menzil ve düşük yakıt/batarya SOC değeri açısından avantaj sağlar (Ribeiro vd. 2025). Özellikle geniş alanların hızlı bir şekilde taranması gereken görevlerde, sabit kanatlı İHA'lar döner kanatlı sistemlere kıyasla daha verimli bir çözüm sunmaktadır. Ancak sabit kanatlı araçların dikey kalkış-iniş yeteneklerinin olmaması ve dar alanlarda manevra kabiliyetlerinin kısıtlı olması önemli bir dezavantajdır. İniş ve kalkış için geniş pist veya uygun arazi ihtiyacı, operasyonel esnekliği sınırlandırmaktadır. Ayrıca sabit kanatlı sistemler havada sabit kalamadığından düşük hızda hassas konumlama gerektiren görevlerde döner kanatlı platformlara göre yetersiz kalmaktadır. Resim 2.1'de Bayraktar TB2 modeli sabit kanat İHA modeli gösterilmiştir.



Resim 2.1 Baykar TB2 sabit kanatlı İHA (İnt. Kyn. 1).

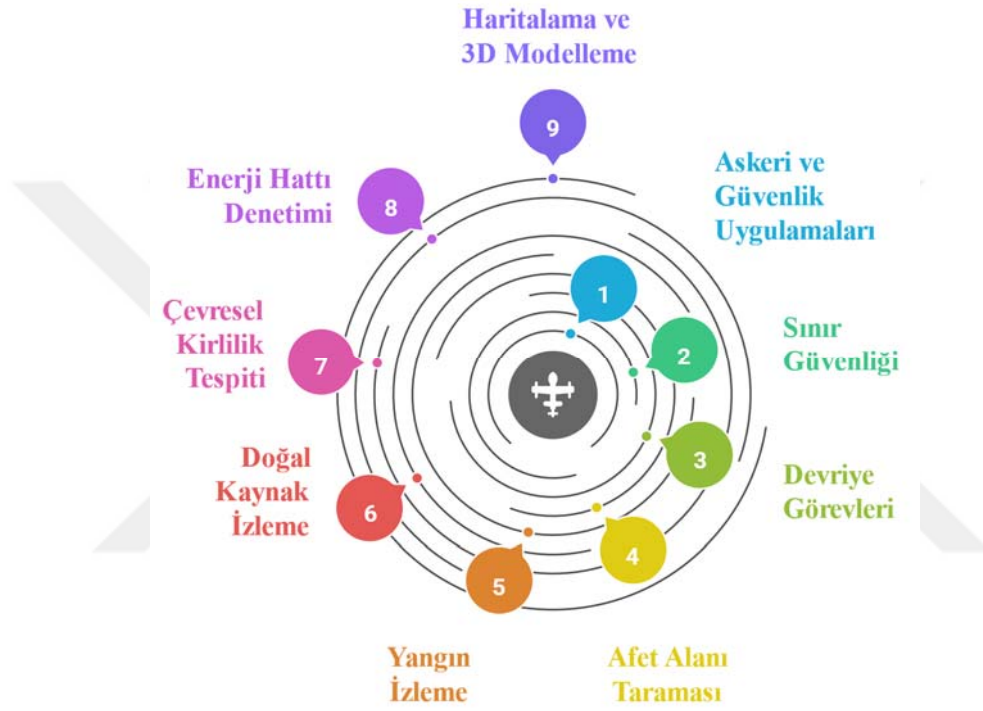
2.2.2.1 Aerodinamik Yapısı ve Uçuş Özellikleri

Sabit kanatlı İHA'lar, temel olarak aerodinamik prensiplere dayalı olarak tasarlanmıştır. Bu sistemlerde kaldırma kuvveti, kanat yüzeylerinin üzerine ve altına etki eden hava akımı farkından doğar. Kanat profili, hava akımını yukarıdan daha hızlı, aşağıdan daha yavaş yönlendirecek şekilde tasarlandığından Bernoulli prensibine göre üst yüzeyde basınç azalır, alt yüzeyde basınç artar ve bu sayede taşıma kuvveti elde edilir. Bu yapı sayesinde sabit kanatlı İHA'lar, enerji tüketiminde verimli olup uzun süreli uçuşlara imkân tanır (Xiao vd. 2021).

Uçuşta itiş gücü pervane veya jet motorlarıyla sağlanırken yön kontrolü kuyruk yüzeylerine bağlı hareketli parçalar aracılığıyla gerçekleştirilir. Aileron, rudder ve elevator gibi kontrol yüzeyleri uçağın yatış, yunuslama ve sapma hareketlerini düzenler. Bu sayede sabit kanatlı sistemler yüksek hızlarda stabil uçuş gerçekleştirebilir ve geniş menzillerde rota takibi yapabilir. Ancak sabit kanatlı araçların yapısal özellikleri, düşük hızlarda ve dar alanlarda manevra kabiliyetlerini sınırlandırır. Bu araçlar havada sabit kalamaz dolayısıyla düşük irtifa gözlemleri, noktasal iniş/kalkış veya hassas konumlama gerektiren görevlerde döner kanatlı platformlara göre dezavantajlıdır. İniş ve kalkış için uzun pistlere ya da uygun açık arazilere ihtiyaç duymaları, operasyonel esnekliği sınırlayan en önemli faktörlerden biridir (Blamis vd. 2021). Sonuç olarak sabit kanatlı İHA'ların aerodinamik yapısı, onları uzun menzilli, yüksek hızlı ve enerji açısından verimli görevler için ideal hâle getirirken; dar alanlarda esneklik ve hassasiyet gerektiren görevlerde kısıtlı kalmalarına neden olmaktadır.

2.2.2.2 Kullanım Alanları

Sabit kanatlı İHA'lar, yüksek hız ve uzun menzil avantajları sayesinde gözetleme, uzun mesafeli keşif ve veri toplama, haritalama ve arazi modelleme gibi görevlerde öne çıkmaktadır. Kullanım alanlarından bazı örnekler Şekil 2.4'te verilmiştir (Rufino vd. 2024).



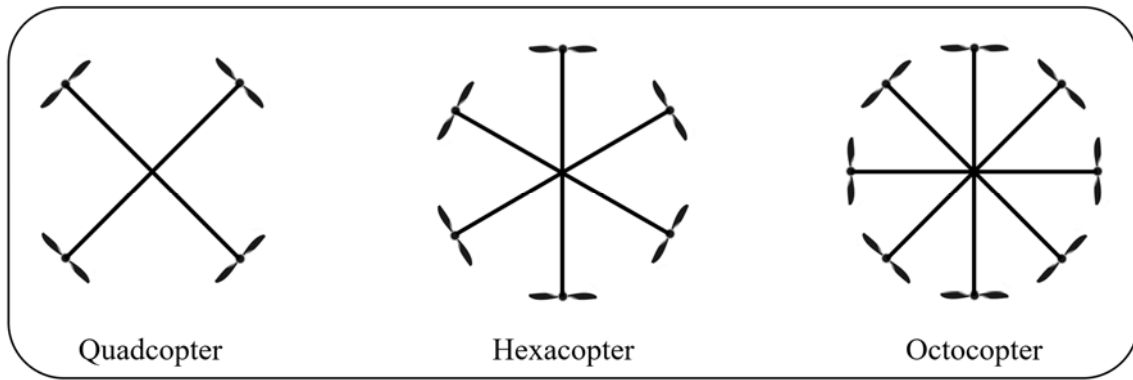
Şekil 2.4 Sabit kanatlı İHA'ların kullanım alanları.

2.2.3 Döner Kanatlı İnsansız Hava Araçları

Döner kanatlı İHA'lar, sabit kanatlı modellere kıyasla daha kısa menzile ve sınırlı uçuş süresine sahiptirler. Ancak sundukları operasyonel esneklik sayesinde birçok alanda tercih edilmektedir. Dikey kalkış ve iniş yapabilmesi, havada sabit kalabilme özelliği ve yüksek manevra kabiliyetleri sayesinde dinamik çevre değişkenlerine hızlı bir şekilde adapte olabilmektedirler. Dikey iniş ve kalkış yapabilmeleri sayesinde herhangi bir pist alanına ihtiyaç duymamaktadırlar. Ayrıca düşük hızlarda hareket edebilmeleri sayesinde görevlerini hassas bir şekilde yapabilmektedirler.

2.2.3.1 Quadrotor, Hexacopter ve Octocopter Yapıları

Drone'lar pervane sayılarına göre farklı sınıflara ayrılmaktadırlar. Drone'un görevi, ağırlığı, faydalı yük kapasitesi ve uçuş süresine bağlı olarak tasarım aşamasında pervane sayısı belirlenir (Sdoukou vd. 2025). En çok kullanılan modeller ise dört adet pervaneye sahip olan quadcopter, altı adet pervaneye sahip olan hexacopter ve sekiz adet pervaneye sahip olan octocopterdir. Şekil 2.5'te quadcopter, hexacopter ve octocopter'in üstten görünümünü simgeleyen tasarımlar verilmiştir.



Şekil 2.5 Pervane sayısına göre İHA çeşitleri.

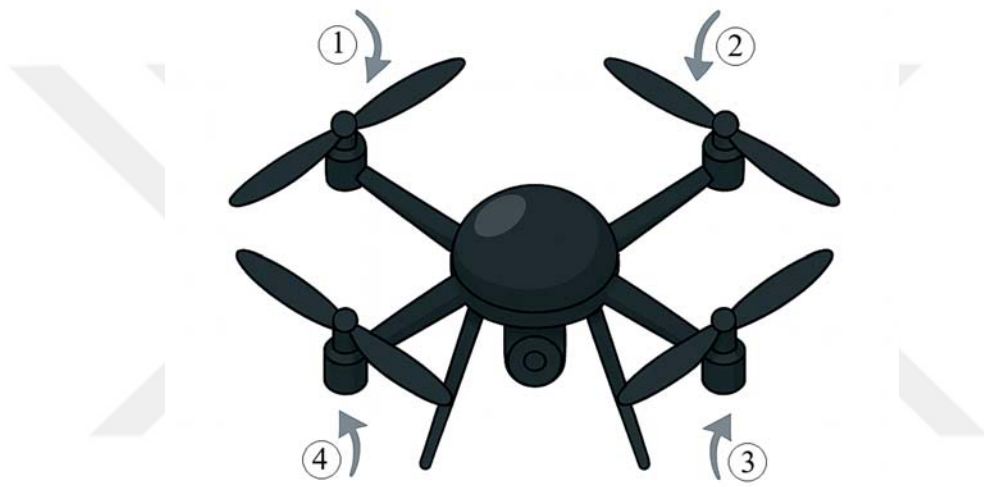
Quadcopterler basit yapıları, boyutlarının küçük olması ve diğerlerine göre daha düşük maliyetleri ile en çok tercih edilen drone çeşididir. Hexacopterler daha dengeli bir uçuş imkânı sunmakta ve quadcopterlere göre daha çok yük taşıma kapasitesine sahiptirler. Ancak daha çok motor olması sebebi ile enerji tüketimi de fazladır. Octocopterler ise en çok yük taşıma kapasitesine sahip olan çeşittir. Kargo taşımacılığı ve tarımsal ilaçlama gibi alanlarda kullanılmaktadır. Yük taşıma kapasitesi ile doğru orantılı olarak enerji tüketimleri ve maliyetleri diğer türlere göre daha yüksektir.

2.2.3.2 Uçuş Dinamikleri ve Manevra Kabiliyeti

Drone'ların manevra kabiliyeti en önemli özelliklerinden birisidir. Her motorun hız kontrolü ayrı ayrı yapılabildiği için drone'un üç ekseninde de (yaw, pitch, roll) hareket etmelerini kolaylaşmaktadır (Dutta 2021). GPS ve IMU desteği sayesinde hassas bir şekilde konumlandırma yapabilmekte ve ani yön değişimlerine hızlı bir şekilde uyum sağlayabilmektedirler. Havada asılı kalabilme özellikleri sayesinde istenilen bölgeden

hassas uygulama gerçekleştirebilirler. Ancak sabit kanatlı İHA'lardaki gibi havada süzülme imkanları olmadığı için enerji tüketimleri daha fazla olmaktadır.

Drone'ların uçuş dinamiklerinin temelinde, her bir pervanenin ürettiği itki kuvveti ve tork bulunmaktadır. Dört pervaneli bir drone üzerinden örnek verilecek olursa çapraz pervaneler aynı yönde, diğer iki ise zıt yönde dönecek şekilde (Şekil 2.6) tasarlanır. Böylelikle gövde üzerinde oluşan toplam torkun dengelenmesini ve aracın stabil kalmasını sağlar (Debele vd. 2022).



Şekil 2.6 Drone pervane yönlerinin gösterimi.

Pervanelerin bağlı olduğu motorların hız kontrolüne göre drone'un yönelimi gerçekleşmektedir. İrtifa kontrolü için bütün motorların hızları artırılarak ya da azaltılarak yükselme ya da alçalma yapılabilir. Drone'un öne doğru gitmesi için arka pervane hızları (3 ve 4 numaralı) ön pervane hızlarına (1 ve 2 numaralı) göre daha hızlı dönerek drone'un öne doğru gitmesi sağlanır. Tam tersi durumda da arkaya doğru hareket eder. Drone'un sağa ve sola doğru eğimlerinde sağ taraftaki pervanelerin hızları (2 ve 3 numaralı) daha yüksek olursa drone sola doğru eğimli bir şekilde hareket ederken tam tersi durumda sağa doğru hareket gerçekleştirir. Dikey eksen etrafında dönüşlerde ise zıt yönlü pervanelerin hızları değiştirilerek saat yönünde ya da saat yönünün tersine olacak şekilde drone'un dönmesi sağlanmaktadır. Uçuş denetleyicisinin hız kontrollerini hızlı bir şekilde yaparak (0.5 ms) drone'un üç ekseninde hızlı ve stabil uçuş gerçekleştirebilmektedir.

2.2.3.3 Kullanım Alanları

Döner kanatlı İHA'lar, dikey kalkış ve iniş yapabilme kabiliyetleri, havada sabit kalabilmeleri ve yüksek manevra kabiliyetleri sayesinde çeşitli alanda tercih edilmektedir. Döner kanatlı İHA'ların kullanım alanlarından (Şekil 2.7) bazılarını aşağıda değinilmiştir (Lyu vd. 2023, Unde vd. 2025).

- Tarımsal Uygulamalar
 - Hassas tarım (ilaçlama ve gübreleme)
 - Hastalık ve zararlı tespiti
 - Verimlilik analizi
 - Bitki sağlığı takibi
 - Tarım arazisi haritalandırma
- Lojistik ve Taşımacılık
 - Kısa mesafelerde kargo taşımacılığı
 - Medikal malzeme ve acil yardım taşımacılığı
- Askeri Uygulamalar
 - Gözetleme ve istihbarat toplama
 - Kamikaze saldırı sistemleri
 - Hedef tespiti ve takibi
 - Sınır güvenliği
- Arama ve Kurtarma
 - Kaybolan kişilerin aranması
 - Yüksek riskli olan bölgelerin taranması
- Çevre İzleme ve Doğa Koruma
 - Orman yangınlarının tespiti
 - Su kaynaklarının takibi
 - Biyolojik çeşitlilik ve habitat izleme
 - Yaban hayatın izlenmesi ve takibi
- Acil Durum ve Afet Yönetimi
 - Doğal afet sonrası hasar tespiti
 - Enkaz altında kalanlar için arama ve kurtarma



Şekil 2.7 Döner kanatlı İHA'ların kullanım alanları.

Döner kanatlı İHA'lar bu avantajları sayesinde tarımdan askeri uygulamalara, afet yönetiminden çevre izlemeye kadar geniş bir alanda etkin biçimde kullanılmaktadır.

2.3 Yapay Zekâ

YZ, günümüzde yalnızca bilgisayar bilimlerinin değil pek çok mühendislik ve uygulama alanının da merkezinde yer almaktadır. Özellikle büyük veri miktarlarının işlenebilir hâle gelmesi ve donanım kapasitesindeki gelişmeler, uzun yıllar teorik düzeyde kalan pek çok yaklaşımın uygulamaya aktarılmasını mümkün kılmıştır. YZ, robotik sistemlerin çevrelerini algılaması, karar verebilmesi ve karmaşık görevleri bağımsız olarak yürütebilmesi açısından kritik bir role sahiptir. Drone teknolojilerinde ise rota planlama, engel algılama, nesne tanıma ve görev optimizasyonu gibi işlevlerde yaygın olarak kullanılmakta, böylece insan müdahalesine duyulan ihtiyaç en aza indirilirken dinamik ortamlarda daha güvenilir ve verimli görev icrası sağlanmaktadır. Bu bölümde YZ'nin temel kavramları, tarihsel gelişimi ve alt dalları özetlenecek; özellikle DÖ tabanlı yöntemlerin nesne tespiti ve drone uygulamalarındaki önemi üzerinde durulacaktır.

2.3.1 Yapay Zekâ ve Tarihsel Gelişimi

YZ kavramı 1950'li yıllara dayanmaktadır. Bir bilgisayar bilimcisi olan Alan Turing tarafından 1950 yılında ortaya atılan "Makine Düşünür mü?" başlıklı makale ile makinelerin insan gibi düşünüp düşünemeyeceği sorusu gündeme gelmiştir. Turing'in geliştirdiği Turing Testi, bir bilgisayarın insan gibi bir zekâ sergileyip sergilemediğini ölçmek için kriter haline gelmiştir (Turing 2007). İlk YZ araştırmaları 1956 yılında

gerçekleştirilen konferansta John McCarthy, Marvin Minsky, Nathaniel Rochester ve Claude Shannon tarafından bilim dünyasına sunulmuştur. İlk dönemlerde YZ’da mantıksal çıkarımlar ve problem çözme algoritmaları sınırlı işlem yeteneğine sahipti. Bu durum, sistemlerin yalnızca önceden tanımlanmış sınırlar içinde çalışmasına ve sınırlı başarılar elde etmesine yol açmıştır.

1970’li yıllarda YZ alanındaki çalışmalar, özellikle “uzman sistemler” adı verilen ilk gerçek uygulamalara odaklanmıştır. Uzman sistemler, belirli bir konuda derinlemesine bilgi barındıran ve bu bilgiye dayanarak insan benzeri kararlar verebilen yazılım programlarıdır. Ancak bu dönemde geliştirilen sistemler, sınırlı bilgi tabanına sahip olmaları ve öğrenme yeteneklerinden yoksun olmaları nedeniyle kapsam açısından oldukça kısıtlı kalmıştır.

1980’lerde YZ alanındaki çalışmalar daha da arttı ve Makine Öğrenmesi (MÖ) ile bağlantılı kavramlar ortaya çıktı. Bu süreçte algoritmaların ve veri setlerinin daha etkili kullanılmasıyla birlikte belirli görevler insan gibi düşünerek hareket etmeye daha yakın hale getirildi. Ancak yine de bu teknolojiler çoğunlukla kurallara dayalıydı ve sınırlı uygulama alanları vardı.

1990’lı yıllardan sonra YZ, özellikle DÖ algoritmalarının gelişmesiyle birlikte büyük bir ivme kazanmıştır. Bu dönemde, büyük veri setlerinin analiz edilmesi ve daha güçlü işlem kapasitesine sahip bilgisayarların kullanılması sayesinde, YZ uygulamaları daha yaygın hale gelmiştir. 1997’de IBM’in Deep Blue adlı satranç programı, dünya şampiyonu Garry Kasparov’u yenerek YZ’nin büyük bir dönüm noktasına ulaşmasını sağladı (İnt. Kyn. 2).

2000’li yıllardan sonra ise teknolojinin daha hızlı gelişmesiyle birlikte YZ alanındaki gelişmeler de artmıştır. Özellikle MÖ ve DÖ yöntemleriyle YZ alanı şekillenmeye başlamıştır. Veri miktarındaki artış, işlemci güçlerinin ve depolama kapasitesinin iyileşmesi, YZ alanında yapılan gelişmelere büyük katkı sağlamıştır. Özellikle büyük veri analizlerinin yaygınlaşmasıyla birlikte YZ sistemlerinin daha büyük veri setleri üzerinde işlem yapmasına imkân sağlayarak algoritmaların daha iyi sonuçlar vermesini sağlamıştır.

2.3.2 Makine Öğrenmesi

YZ'nin alt dallarından birisi olan MÖ veriler üzerinden öğrenme odaklıdır. Yani bir MÖ algoritması, referans veriler üzerinden modelini geliştirir ve yeni veriler üzerinde tahminde bulunur. İnsan müdahalesi minimum düzeyde tutularak algoritmanın kendi kendine öğrenmesi ön plandadır. MÖ, denetimli öğrenme (supervised learning), denetimsiz öğrenme (unsupervised learning) ve Pekiştirmeli Öğrenme (Reinforcement Learning – RL) olmak üzere üç ana alt kategoriye ayrılır (Sarker 2021). Bunlar;

Denetimli öğrenme: Verilerin önceden etiketlendiği ve böylece modelin eğitildiği öğrenme yöntemidir. Bu yöntemde elde edilen sonuçlar ile doğru sonuçlar karşılaştırılarak başarı oranı belirlenmektedir. Genellikle sınıflandırma ve regresyon problemlerinde kullanılır. Lineer Regresyon (LR), Destek Vektör Makinaları (Support Vector Machine – SVM), NN algoritmaları örnek olarak verilebilirler.

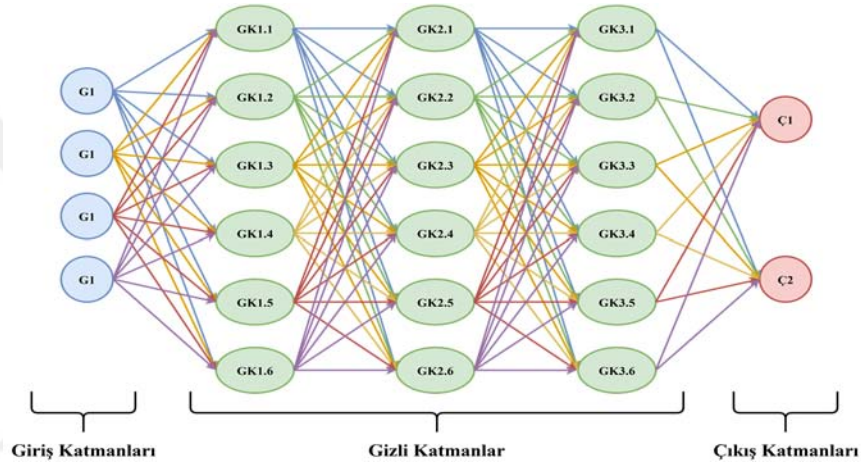
Denetimsiz öğrenme: Veri setleri etiketlenmeden çalışılır ve model verilerin içindeki yapıları kendi kendine keşfederek sonuçların tahminini yapmaktadır. Bu öğrenme yönteminde verilerin sınıflandırılması veya gruplandırılması yapılmakta olup sonuçların tahmin edilmesi için etiketli veriye ihtiyaç duyulmaz. K-Means Clustering, Principal Component Analysis (PCA), Hierarchical Clustering algoritmaları örnek olarak verilebilir.

Pekiştirmeli öğrenme: Bir ajan (YZ) çevresiyle etkileşime girer ve ödül ya da ceza alarak öğrenir. Ajan, belirli bir hedefe ulaşmak için nasıl hareket etmesi gerektiğini keşfeder. Bu öğrenme türü, genellikle oyun, robotik, otonom sistemler ve stratejik karar verme problemlerinde kullanılır. Q-Learning, Deep Q-Networks (DQN), Proximal Policy Optimization (PPO) algoritmaları örnek olarak verilebilir.

2.3.3 Yapay Sinir Ağları

Yapay Sinir Ağları (YSA), insan beyninin çalışmasından esinlenerek geliştirilen matematiksel modellerdir. Öğrenme yeteneğine sahip olan bu ağlarda girdi verileri

işlenerek sonuçlar elde edilmektedir. Sırasıyla giriş, gizli ve çıkış katmanlarından oluşur. Giriş katmanında veriler bulunmakta olup her giriş verisi bir nöron tarafından temsil edilir. Gizli katmanda ise veriler işlenmektedir. Birden fazla gizli katman olabilmektedir. Bu katmanda her nöron çeşitli kombinasyonlarla girişlerdeki verilerden özellikler öğrenirler. Çıkış katmanında ise modelin tahmin ettiği sonuçlar verilir. Ağlar eğitim sürecinde ağırlıklarını güncelleyerek doğru tahminlerde bulunmayı hedefler (Kelek vd. 2025). Şekil 2.8’de 4 adet girişi, 3 adet gizli katmanı ve 2 adet çıkışı olan örnek bir YSA modeli verilmiştir.



Şekil 2.8 Dört adet girişi, üç adet gizli katmanı ve iki adet çıkışı olan YSA modeli.

2.3.4 Derin Öğrenme

DÖ, YZ ve MÖ’nün en önemli alt alanlarından biridir. Büyük veri kümelerindeki karışık ilişkilerin çıkarılmasında kullanılan güçlü bir yöntemdir. DÖ, temelde YSA’nın geliştirilerek çok katmanlı yapıya dönüştürülmüş bir versiyondur. Klasik YSA’da birkaç katman kullanılırken DÖ’de yüzlerce katmanlı ağ yapıları kullanılabilmektedir. Bu sayede yüksek düzeyde özellik çıkarımı yapabilen DÖ, görsel verilerin, ses sinyallerinin veya doğal dil gibi karmaşık yapıya sahip verilerin işlenmesinde büyük avantaj sağlamaktadır (Mienye ve Swart 2024).

DÖ’nün en önemli farklılıklarından biri, özelliklerin insan tarafından önceden tanımlanmasına gerek duymamasıdır. Geleneksel MÖ yöntemlerinde veriden elde edilecek özellikler elle seçilerek algoritmaya aktarılmaktadır. Ancak DÖ’de ham veriden

anlamalı özellikler kendiliğinden oluşturulmakta ve algoritma bu özellikleri kendisi öğrenmektedir. Sonuç olarak karmaşık görevlerde hem daha yüksek doğruluk elde edilmekte hem de insan müdahalesine olan ihtiyaç azalmaktadır. Teknolojinin gelişmesiyle birlikte DÖ'nün başarı oranları da artmıştır. Bu artışta büyük verilerin oluşturulması ve bu verilerin etiketlenmiş olmasının etkisi, yüksek performanslı Grafik İşlemci Ünitesi (Graphics Processing Unit – GPU)'lar sayesinde paralel işlemlerin yapılabilmesi, işlem sürelerinin kısalması ve açık kaynaklı kütüphane desteği (TensorFlow, PyTorch, Keras vb.) sayesinde sürekli geliştiriliyor olması önemli rol oynamaktadır. Bu etkenler sayesinde DÖ'nün kullanım alanları genişlemiş ve çeşitli alanlarda kullanımı artmıştır. Bu alanlara örnek olarak nesne tespiti, otonom sistemler, konuşma tanıma, doğal dil işleme, biyomedikal görüntü işleme gibi birçok örnek verilebilir.

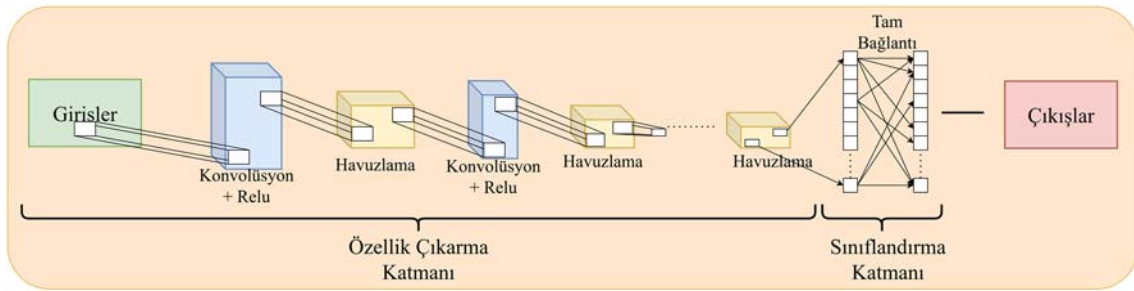
Robotik alanlarda da DÖ oldukça yaygın olarak kullanılmaktadır. Özellikle algılama ve karar verme mekanizmalarının geliştirilmesinde kritik bir öneme sahiptir. Görüntü tabanlı nesne tespiti, çevresel durum analizi, engel algılama ve rota planlama gibi işlemlerde DÖ tabanlı modeller kullanılarak yüksek doğruluk elde edilmekte ve görevler başarılı bir şekilde tamamlanmaktadır. Drone teknolojilerinde ise tarım arazilerinin analizi, hastalık ve zararlıların tespiti, hassas ilaçlama bölgelerinin belirlenmesi ve nesne takibi gibi uygulamalarda DÖ yöntemleri etkili bir şekilde kullanılmaktadır. DÖ'de kullanım alanına göre geliştirilen bazı mimariler aşağıda verilmiştir.

2.3.4.1 Evrişimsel Sinir Ağları

Evrişimsel Sinir Ağları (CNN), DÖ alanındaki en önemli dönüm noktalarından birini oluşturmaktadır. İlk kez 1980'lerin sonlarında Yann LeCun tarafından geliştirilen LeNet-5 mimarisi, el yazısı rakamların tanınması için kullanılmış ve bu alandaki öncü çalışmalardan biri olmuştur. Daha sonraki yıllarda donanım gücünün ve büyük veri setlerinin artmasıyla birlikte CNN'ler, Krizhevsky ve arkadaşlarının 2012'de geliştirdiği AlexNet modeli ile büyük bir çıkış yapmıştır. AlexNet'in ImageNet yarışmasında elde ettiği başarı, CNN'lerin günümüzde görüntü işleme ve bilgisayarla görü uygulamalarında standart hâle gelmesinin önünü açmıştır (Krizhevsky vd. 2012). Bu tarihsel gelişim,

CNN'in yalnızca teorik bir model değil aynı zamanda endüstride yaygın kullanılan pratik bir çözüm olduğunu göstermektedir.

CNN, giriş katmanı, konvolüsyon katmanı, havuzlama katmanı ve tam bağlantı katmanlarından oluşmaktadır (Yamashita vd. 2018). Şekil 2.9'da CNN modelinin yapısı gösterilmiştir.



Şekil 2.9 CNN'in katmanlı yapısının şematik gösterimi.

Konvolüsyon katmanında bir filtre görüntü matrisi üzerinde kaydırılır. Bu kaydırma işlemi esnasında matematiksel toplama – çarpma gibi işlemlerle özellikler çıkarılmaktadır. İki boyutlu konvolüsyon işleminin matematiksel formülü denklem 2.1'de verilmiştir. Her filtre görüntünün farklı özelliklerini ortaya çıkarmaktadır. Ağ derinleştikçe ilk katmanlarda basit kenarlar ortaya çıkarken ilerleyen katmanlarda karmaşık nesneler ortaya çıkmaktadır (Cengiz vd. 2022).

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(i + m, j + n) \cdot K(m, n) \quad (2.1)$$

Burada;

I: giriş görüntüsünü,

K: Filtre boyutunu,

i,j: filtre koordinatlarını,

S(i,j): özellikleri temsil etmektedir.

Konvolüsyon işleminin ardından Rectified Linear Unit (ReLU) aktivasyon fonksiyonu uygulanmaktadır. Denklem 2.2'de matematiksel formülü gösterilmiştir. Burada, negatif

değerler sıfırlanmakta ve ağ sadece pozitif aktivasyonları dikkate alarak işlem yapmaktadır. Bu da öğrenmeyi kolaylaştırmaktadır (Nwankpa 2018).

$$f(x) = \max(0, x) \quad (2.2)$$

Burada;

f(x): Aktivasyon fonksiyonunun çıktısı,

x: bir nörona giren toplam girdi değerini ifade etmektedir.

Havuzlama katmanı, özellik haritalarının boyutlarında küçültme, parametre sayılarında azaltma ve ağın hesaplama yükünü azaltmak için kullanılmaktadır. En yaygın olarak kullanılan maksimum havuzlama yöntemi (denklem 2.3) olup pencere içerisindeki en yüksek değeri seçerek daha baskın özelliklerin öne çıkmasını sağlar. Bu sayede önemli bilgiler korunmakta ve gereksiz bilgiler ayrıştırılmaktadır. Ortalama havuzlama yönteminin kullanıldığı zamanlarda pencere içerisindeki değerlerin ortalamaları alınarak işlem yapılmaktadır. Bu çeşitlilik sayesinde ağın farklı seviyelerde genelleme yapabilmesine imkân sağlanmaktadır. Havuzlama işlemi sayesinde parametre sayıları azalmakta ve aşırı öğrenme riski düşürülmektedir (Stanković ve Mandić 2023).

$$y = \max(x_1, x_2, x_3, \dots, x_n) \quad (2.3)$$

Burada;

y: seçilen en büyük değerın çıktısı,

x: havuzlama işleminin uygulandığı filtre içerisindeki değerdir.

Tam bağlantılı katmanlarda ise çıkarılan özellikler vektörleştirilerek tam bağlantılı katmanlara aktarılmakta ve YSA ile sınıflandırma yapılmaktadır. Çıkış katmanında genellikle Softmax fonksiyonu (denklem 2.4) kullanılmaktadır. Böylelikle her sınıfa ait olasılık dağılımı sağlanmaktadır.

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad j = 1, 2, 3, \dots, K \quad (2.4)$$

Burada;

z_j : Aktivasyon fonksiyonu uygulamadan önceki son katmanın ürettiği gerçek sayılar,

K: Toplam sınıf sayısı,

$\sigma(z)_j$: Softmax fonksiyonun j. Sınıf için hesapladığı olasılık değerini ifade etmektedir.

CNN özellikle görüntü işleme alanında sağladığı yüksek başarı oranları, parametrik verimliliği ve katmanlı yapısı sayesinde YZ uygulamalarının temelini oluşturmaktadır. Konvolüsyon, aktivasyon ve havuzlama katmanlarının birlikte çalışması sayesinde ağ, görsel verilerden anlamlı öznitelikleri çıkararak sınıflandırma ve nesne tespiti gibi görevlerde üstün performans göstermektedir (Albawi vd. 2018).

2.3.5 Nesne Tespiti Algoritmaları

Bilgisayarlı görü alanında nesne tespiti en çok çalışılan konular arasında bulunmaktadır. Nesne tespiti algoritmalarında amaç, bir görüntü içerisindeki nesnelerin tespit edilerek sınıflandırılması ve konumlandırılmasıdır. Sınıflandırma işleminin yanında aynı zamanda konumlandırma yapması klasik görüntü sınıflandırma algoritmalarına göre daha karmaşık olmasına sebep olmaktadır. Nesne tespit algoritmaları iki ana grupta incelenmektedir. Bunlar iki ve tek aşamalı yöntemler şeklindedir (Carranza-García vd. 2020, Mohammed 2025).

İki Aşamalı Yöntemler (Two-Stage Detectors): Bu yöntem iki aşamadan oluşmaktadır. İlk aşamada görüntü içerisinde nesnenin tespit edildiği bölge belirlenmektedir. İkinci aşamada ise tespit edilen bölge içerisinde sınıflandırma yapılmaktadır. Bu yaklaşımların doğruluk oranları daha yüksektir ancak iki aşamalı olması sebebiyle işlem süreleri de daha yüksek olmaktadır. Bu yöntemlere R-CNN, Fast R-CNN, Faster R-CNN gibi modeller örnek olarak verilebilirler.

Tek Aşamalı Yöntemler (One-Stage Detectors): Bu yöntem türünde ise görüntü üzerinde nesne tespiti ve sınıflandırması aynı anda yapılmaktadır. Bu özelliği sayesinde işlem süreleri iki aşamalı yöntemlere göre daha az olduğu için gerçek zamanlı uygulamalarda genellikle bu yöntem türü tercih edilmektedir. Bu yöntemlerde YOLO, Single Shot

MultiBox Detector (SSD), RetinaNet gibi mimariler ön plana çıkmaktadır.

İki aşamalı yöntemler yüksek doğruluk oranlarıyla öne çıkarken işlem sürelerinin uzun olması gerçek zamanlı uygulamalarda kısıtlayıcı bir faktör hâline gelmektedir. Bu nedenle tek aşamalı yöntemler daha çok tercih edilmektedir. SSD ve RetinaNet gibi yöntemler hız ve doğruluk arasında dengeli çözüm sunar. YOLO ise görüntüyü tek seferde işleyerek nesne tespitini hem hızlı hem de yeterli doğrulukla gerçekleştirmektedir. Bu özelliği sayesinde tarımsal uygulamalar gibi geniş ölçekli ve dinamik ortamlarda etkin bir şekilde kullanılabilir.

2.3.5.1 You Only Look Once

Nesne tespitinde klasik yöntemler öncelikle görüntü üzerinde nesne olan bölgeleri bulmakta, sonra bu bölgeleri sınıflandırmaktadır. Bu yaklaşım, doğruluk açısından başarılı olsa da işlem süresi nedeniyle gerçek zamanlı uygulamalarda sınırlı kalmaktadır. YOLO algoritması ise görüntüyü tek bir kez işleyerek hem nesnenin sınıfını hem de konumunu aynı anda tahmin edebilmektedir (Redmon vd. 2016).

YOLO'nun temel çalışma mantığında giriş görüntüleri öncelikle $S \times S$ boyutunda ızgaralara bölünür. Her bir ızgara hücresi içerisindeki nesneler için tahminler yaptırmaktır. Her hücre, bir veya birden fazla kapsayıcı kutu (bounding box) üretir ve bu kutular için hem nesneye ait güven puanı (c) hem de sınıf olasılıkları ($p(c_i|object)$) hesaplanır. Daha sonra bu kutular birleştirilerek nesnelerin hem konumu hem de türü belirlenir. Bir nesneyi bulmak için oluşturulan kapsayıcı kutunun matematiksel ifadesi denklem 2.5'te verilmiştir. Verilen matematiksel denklemdeki değerler normalize edilmektedir. Böylelikle farklı çözünürlükte görüntülerde modelin genellenebilirliği korunmaktadır.

$$B = (x, y, w, h) \quad (2.5)$$

Burada;

B: kapsayıcı kutuyu,

x, y: kutu merkezinin ızgara hücresine göre normalize edilmiş koordinatlarını,
w, h: bütün görüntüye göre normalize edilmiş kutunun genişliği ve yüksekliğini temsil eder.

Kutular belirlendikten sonra bu kutuların içerisinde bir nesne olup olmadığı ve ne kadar doğru tahminde bulunduğu güven puanı hesaplanarak belirlenmektedir. Güven puanının matematiksel formülü denklem 2.6'da verilmiştir. C değerinin 0 olması o hücrede bir nesnenin olmadığını, 0 ile 1 arasında olması ise Örtüşme Oranına (Intersection over Union – IoU) bağlı olarak doğru tahmin ettiğini gösterir.

$$C = P(object) \cdot IoU(B, B^{gt}) \quad (2.6)$$

Burada;

C: güven puanını,

$P(object)$: belirlenen hücrede nesne olup olmadığını,

$IoU(B, B^{gt})$: tahmin edilen kutu (B) ile gerçek kutu (B^{gt}) arasındaki örtüşme oranını temsil etmektedir.

Güven puanı belirlendikten sonra her bir hücre içerisinde nesne olup olmadığı ve bu nesnenin sınıflandırma olasılıkları gerçekleştirilmektedir. Sınıflandırma olasılığının formülü denklem 2.7'de verilmiştir.

$$p(c_i|object) \quad i = 1, 2, 3, \dots, K \quad (2.7)$$

Burada;

$p(c_i|object)$: sınıf olasılığını,

K: sınıf sayısını ifade etmektedir.

Nesnenin hangi sınıfa ait olduğunu belirlemek için güven puanı ile sınıf olasılıkları çarpılarak, her kutu için nihai nesne tahmin skoru (denklem 2.8) elde edilir.

$$p(c_i) \cdot IoU(B, B^{gt}) \quad (2.8)$$

CNN'lerde genellikle kayıp fonksiyonları tahmin edilen ve gerçek değer arasındaki fark ile ölçülmektedir. YOLO'nun kayıp fonksiyonu kutu (konum) kaybı, güven puanı kaybı ve sınıf tahmini kaybı olmak üzere 3 temel bileşenden oluşmaktadır. Denklem 2.9'da YOLO'nun toplam kayıp fonksiyonu verilmiştir.

$$L_{class} = L_{coord} + L_{conf} + L_{class} \quad (2.9)$$

Burada;

L_{coord} : kutu kayıp fonksiyonunu,

L_{conf} : güven puanı kayıp fonksiyonunu,

L_{class} : sınıf tahmini kayıp fonksiyonunu temsil etmektedir.

YOLO'nun versiyonlarına göre kayıp fonksiyonlarının matematiksel formülü değişmektedir. YOLO'nun ilk versiyonu olan YOLOv1'de hataların karesinin ortalaması (Mean Squared Error – MSE) temelli kayıp fonksiyonlar denklem 2.10 – 2.12 arasında verilmiştir (Redmon vd. 2016).

YOLOv1 mimarisinin kutu kayıp fonksiyonunun matematiksel gösterimi denklem 2.10'da verilmiştir.

$$L_{coord} = \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} \left[\frac{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2}{(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2} \right] \quad (2.10)$$

Burada;

S: ızgara boyutunu,

B: her ızgaradaki kapsayıcı kutu sayısını,

1_{ij}^{obj} : i. hücrede j. kutunun nesne içerip içermediğini gösteren gösterge fonksiyonu,

λ_{coord} : koordinat hatasına verilen ağırlığı,

x, y: gerçek koordinatlar noktalarını,

$\hat{x}, \hat{y}$: tahmin edilen koordinat noktalarını,

w, h: nesnenin gerçek genişliğini ve yüksekliğini,

$\hat{w}, \hat{h}$: nesnenin tahmin edilen genişliğini ve yüksekliğini temsil etmektedir.

YOLOv1'in güven puanın kayıp fonksiyonunun matematiksel formülü denklem 2.11'de verilmiştir.

$$L_{conf} = \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} (C_i - \hat{C}_i)^2 + \lambda_{noobj} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{noobj} (C_i - \hat{C}_i)^2 \quad (2.11)$$

Burada;

C_i : gerçek güven puanını,

$\hat{C}_i$: tahmin edilen güven puanını,

λ_{noobj} : nesne olmayan kutulardan gelen katsayıların modelin öğrenmesini bozmaması için verilen küçük bir katsayıyı temsil eder.

YOLOv1'in sınıf tahmini kaybının fonksiyonu denklem 2.12'de verilmiştir.

$$L_{class} = \sum_{i=0}^{S^2} 1_i^{obj} \sum_{c=0}^C (p_i(c) - \hat{p}_i(c))^2 \quad (2.12)$$

Burada;

$p_i(c)$: i. hücredeki c sınıfının gerçek değerini,

$\hat{p}_i(c)$: i. hücredeki c sınıfının tahmin edilen olasılığını ifade etmektedir.

YOLO'nun versiyonları geliştirildikçe hata fonksiyonunda da düzenlemeler ve iyileştirmeler olmuştur. YOLOv3'e ait olan hata fonksiyonları denklem 2.13 – 2.15 aralığında verilmiştir. Bu versiyonda sınıf tahminlerinde İkili Çapraz Entropi (Binary Cross Entropy – BCE) kayıp fonksiyonu kullanılmıştır. Küçük nesnelerin daha iyi tahmin etmek için çok ölçekli tahmin kullanılmıştır (Redmon ve Farhadi 2018). Ayrıca koordinatlar ve güven puanı için de lojistik regresyon yöntemleri kullanılmıştır. YOLOv3'ün kutu kayıp fonksiyonu denklem 2.13'te, güven puanı kaybı denklem 2.14'te ve sınıf tahmini kaybı denklem 2.15'te verilmiştir (Murat ve Kiran 2025).

$$L_{coord} = \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^B 1_{ij}^{obj} \left[\frac{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2}{(\sqrt{w_i} - \sqrt{\hat{w}_i})^2 + (\sqrt{h_i} - \sqrt{\hat{h}_i})^2} \right] \quad (2.13)$$

$$L_{conf} = -\lambda_{conf} \sum_{i=0}^{S^2} \sum_{j=0}^B [C_{ij} \log(\hat{C}_{ij}) + (1 - C_{ij}) \log(1 - \hat{C}_{ij})] \quad (2.14)$$

$$L_{class} = -\lambda_{class} \sum_{i=0}^{S^2} 1_i^{obj} \sum_{c=1}^C [p_i(c) \log \hat{p}_i(c) + (1 - p_i(c)) \log(1 - \hat{p}_i(c))] \quad (2.15)$$

Burada;

$C_{ij} = 1$: kutu nesne içeriyorsa,

$C_{ij} = 0$: kutu nesne içermiyorsa,

$\hat{C}_{ij}$: modelin nesne tahminini,

$\lambda_{coord}, \lambda_{conf}, \lambda_{class}$: her bileşene verilen ağırlığı temsil etmektedir.

YOLOv5 ve daha güncel versiyonlarında kayıp fonksiyonlarında kullanılan yöntemler de değişkenlik göstermiştir. Kutu kaybı fonksiyonunda IoU kutuların birbirleri ile çakışmasından kaynaklı olarak tek başına yetersiz kaldığı durumlar oluşmaktaydı. Bu sebeple Generalized IoU (GIoU), Distance IoU (DIoU) ve özellikle Complete IoU (CIoU) kayıpları kullanılmaya başlanmıştır. Güven puanı hesaplamasında logitli BCE kaybı kullanılmaktadır. Sınıf tahmini fonksiyonunda ise BCE kayıpları ve Focal kayıplar kullanılmaktadır.

Kutu kayıplarında CIoU yöntemi kullanılarak yalnızca IoU oranlarına değil aynı zamanda kutu merkezleri arasındaki mesafeye ve şekilsel benzerliğin oranına bakılmaktadır. Böylelikle daha iyi sonuçlar elde edilmektedir. Denklem 2.16'da YOLOv5 mimarisinin CIoU kullanarak elde edilen kutu kayıpları verilmiştir.

$$L_{coord} = L_{CIoU} = 1 - IoU + \frac{\rho^2(c, c^{gt})}{d^2} + \alpha \cdot v \quad (2.16)$$

Burada;

ρ : kutu merkezleri arasındaki mesafeyi,

d : tüm kutuları kapsayan dikdörtgenin köşegen uzunluğunu,

v : en – boy oranını gösteren ceza terimini,

α : dengeleyici katsayısı temsil etmektedir.

Güven puanı hesaplamasında logitli BCE yöntemi kullanılmaktadır. YOLOv5 mimarisinin güven puanının matematiksel formülü denklem 2.17’de verilmiştir.

$$L_{conf} = -[t \cdot \log \sigma(o) + (1 - t) \cdot \log(1 - \sigma(o))] \quad (2.17)$$

Burada;

t : hedef değeri,

o : tahmin edilen logiti (çıktıyı),

σ : sigmoid aktivasyonunu temsil etmektedir.

YOLOv5 sınıf tahmininde softmax yerine her sınıf için multi-label uyumlu bağımsız lojistik tahminler yapar. Dolayısıyla birden fazla sınıfa ait olabilecek nesnelerde daha iyi sonuçlar verir. Denklem 2.18’de sınıf tahmini kayıplarının matematiksel formülü verilmiştir.

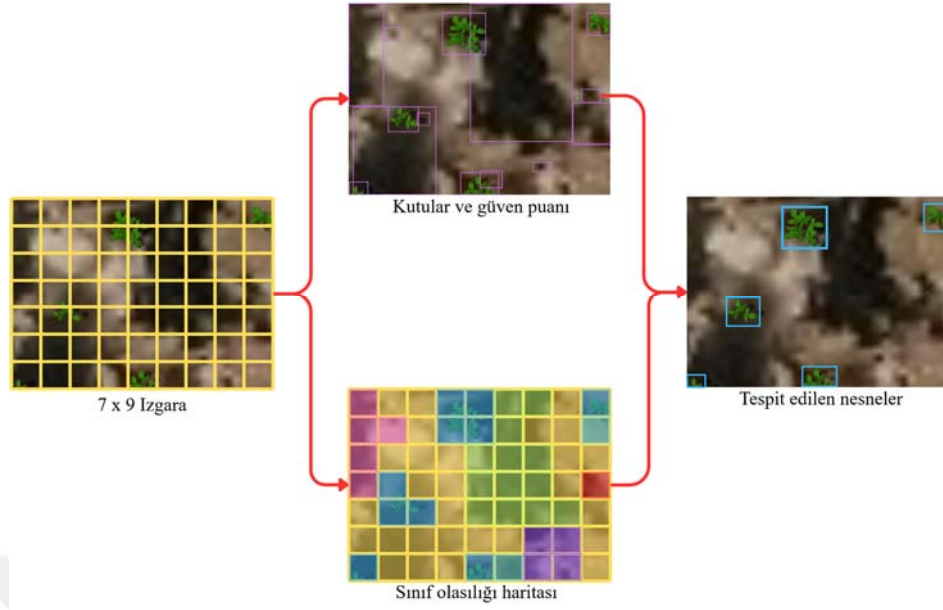
$$L_{class} = - \sum_{c=1}^C [y_c \cdot \log \sigma(s_c) + (1 - y_c) \cdot \log(1 - \sigma(s_c))] \quad (2.18)$$

Burada;

y_c : gerçek etiketi,

s_c : modelin sınıf logitini temsil etmektedir.

Verilen bu açıklamalar doğrultusunda bir görüntü üzerinde YOLO mimarisi tarafından sınıflandırma yapılacağına yapılan işlemler Şekil 2.10’da gösterilmiştir.



Şekil 2.10 Görüntü üzerinde YOLO mimarisinin adımları.

2.4 Literatür Taraması

Bu bölümde hem ulusal hem de uluslararası düzeyde yapılmış çalışmalar incelenerek, tarım alanında yapılan sürü robotları, insansız hava araçları ve yapay zekâ temelli nesne tespiti konularında ne tür yaklaşımlar benimsendiği irdelenmiştir. Ayrıca bu tez çalışmasının mevcut literatürdeki konumunun belirlenmesi ve benzer çalışmalarda izlenen yöntemlerin karşılaştırılması amaçlanmıştır.

2.4.1 Ulusal Çalışmalar

Karabay ve Çavaş seracılıkta ve akıllı tarım teknolojilerindeki gelişmeler üzerine bir çalışma gerçekleştirmişlerdir. Çalışmalarında seracılıkta ve akıllı tarımda kullanılan çeşitli sensörlerden bahsetmişler ve bu alanda kullanılan YZ ve DÖ algoritmalarını incelemişlerdir. Çalışma sonucunda ürün gelişimini, hastalık tespitini, sulama sisteminin otomatikleştirilmesini ve erken uyarı sisteminin çiftçiler açısından faydalı olacağını vurgulamışlardır (Karabay ve Çavaş 2024).

Özgüven ve arkadaşları tarım alanında kullanılan otonom araçların özelliklerinin incelenmesini gerçekleştirmişlerdir. Geçmişten günümüze kadar olan süreç içerisinde

tarım alanında kullanılan cihazların özelliklerinde karşılaştırmalar yapılmış ve otonom olarak kontrol edilen çeşitli makineler (traktör, İnsansız Kara Aracı (İKA) vb.) incelenmiştir. Yapılan incelemeler sonucunda dijital tarım uygulamaları kullanıldığında daha verimli bir şekilde üretim olacağı öngörülmektedir (Özgüven vd. 2024).

Başaran yapmış olduğu çalışmada kırmızı ve Siirt fıstık çeşitlerinin DÖ yöntemi ile birbirinden ayırt edilmesi üzerine gerçekleştirmiştir. Çalışmada yaklaşık 2150 adet fıstık görüntüleri üzerinden dalgacık görüntü saçılımı ve DarkNet53 ile özellik çıkarımlarını gerçekleştirmiştir. Ardından SVM ile sınıflandırma yapmıştır. Çalışma sonucunda dalgacık görüntü saçılımı ile %85.27, DarkNet53 ile % 95.34 ve iki yöntemden elde edilen özelliklerin birleştirilerek yapılan sınıflandırmada ise %97.98'lik bir doğruluk oranı ile fıstıkların ayırt edilmesini gerçekleştirmiştir. Çalışma sonucunda iş gücü maliyetlerinin bu şekilde azaltılarak kalite kontrolün DÖ ile yapılabileceğini vurgulamıştır (Başaran 2022).

Terzi ve arkadaşları üzüm çeşitlerinin (Corint, Merlot, Tayfi, Michele Palieri ve Narince) DÖ yöntemleri ile sınıflandırılmasını incelemişlerdir. Üzüm, yaprak, salkım ve sürgün gibi özellikler incelenerek oluşturulan 15 katmanlı CNN modeli ile 1028 adet görüntüden oluşan veri seti üzerinde eğitim – test işlemlerini gerçekleştirmişlerdir. Çalışma sonucunda %96.10'luk bir başarı oranı ile 5 çeşit üzümün sınıflandırmasını gerçekleştirmişlerdir (Terzi vd. 2023).

Ergün, tarım arazilerindeki yabancı otların tespiti üzerine bir çalışma gerçekleştirmiştir. Çalışmasında 3975 adet görüntüden oluşan veri setini ResNet, DenseNet ve VggNet modelleri ile karşılaştırmıştır. Karşılaştırma sonucunda en yüksek başarı oranını %99.21'lik doğruluk ile VggNet ile elde etmiştir. Böylelikle tarım arazilerinde DÖ'ye bağlı yabancı ot tespitinin tarımsal uygulamalarda ölçeklenebilir ve daha verimli sonuçlar elde etmesine olanak sağladığını göstermiştir (Ergün 2025).

Boyar ve Yıldız çalışmalarında fındıklarda yaygın olarak görülen külleme hastalığının tespitini gerçekleştirmişlerdir. İnternet üzerinden alınan 424 adet görüntüyü etiketleyerek veri setini oluşturmuşlar ve veri artırma yaparak YOLOv5 modeli üzerinde eğitim

yapmışlardır. Sonuçlar incelendiğinde külleme hastalığının yaprak üzerinde yaklaşık %90 doğrulukla tespit edilebildiği gösterilmiştir. Arka planı karmaşık olan görüntülerde sağlıklı bölgelerin tespitinde ise %82'lik bir başarı oranı elde edilmiştir. Kullanılan model ile özellikle küçük ölçekli hastalık belirtilerini doğru şekilde belirlenmiş ve hastalığın erken evrede tespit edilmesine olanak sağlamıştır (Boyar ve Yıldız 2022).

İren ve Yeşilay çalışmalarında tarım alanında yaygınlaşan drone'ların önemini ele almaktadır. Çalışmalarında drone teknolojisinin başlangıçta askeri alanında kullanıldığı, ancak zamanla tarım alanında (ilaçlama, gübreleme, ürün ekimi, hastalık tespiti, yabancı ot tespiti), hayvancılıkta sürü yönetimi ve hasar tespiti gibi çeşitli alanlara kullanılmaya başladığını belirtmişlerdir. Araştırmada sahadan elde edilen verilerle, drone'ların özellikle ilaçlama ve sulamada kaynak kullanımını optimize ettiği, zaman ve iş gücünden tasarruf sağladığı, pestisit miktarını azalttığı ve böylece hem çevresel hem de ekonomik fayda sunduğu görülmüştür. Ayrıca drone destekli görüntüleme sistemlerinin hastalık teşhisi ve erken müdahalede kritik rol oynadığı, YZ ve DÖ yöntemleriyle birleştiğinde daha doğru sonuçlar elde edilebildiği vurgulanmaktadır. Drone'ların sürdürülebilir tarım vizyonu açısından vazgeçilmez bir araç olduğu ve gelecekte Tarım 4.0 kapsamında daha yaygın kullanılacağı belirtilmiştir (İren ve Yeşilay 2025).

Özgüven ve arkadaşları çalışmalarında drone bileşenlerinden, avantajlarından, dezavantajlarından ve kullanılan çeşitli sensörlerden bahsetmişlerdir. Günümüzde tarımda drone kullanım alanlarına örnekler vermişler ve gelecekte bu alanda yapılacak uygulamalara öneride bulunmuşlardır (Özgüven vd. 2022). Bir diğer çalışmada ise bahçe bitkileri yetiştiriciliğinde kullanılan teknolojik gelişmeleri detaylı bir şekilde değerlendirmiştir. İncelemesinde hassas tarım, görüntü işleme, YZ, kullanılan sensörler, drone'lar, nesnelerin interneti, uzaktan algılama gibi konulara değinmiştir. Özellikle drone'lar ile bitkilerin izlenebilmesi, YZ yardımıyla hastalık tespitinin erken aşamalarda belirlenebilmesi, sensörler ile toprak (su, nem vb.) ve çevre durumunun belirlenebilmesi gibi uygulamaların etkinliğinin arttığını vurgulamıştır. Ayrıca gübreleme ve pestisit işlemlerinde bu teknolojik donanımların kullanılmasıyla tasarruf sağlandığı ve çevresel kirlenmenin de aynı şekilde azaldığını vurgulamıştır (Özgüven 2023).

Karahanlı ve Taşkın ayçiçeği bitkisinin gelişim evrelerinin tespiti için DÖ kullanmışlardır. Drone ile 43 farklı günde elde edilen yaklaşık 6500 adet görüntü bölünerek toplamda 12800 adet görüntüden oluşan bir veri seti oluşturmuşlardır. Bu veri seti üzerinden AlexNet, InceptionV3, ResNet101, DenseNet121, MobileNet, Xception olmak üzere 6 farklı CNN modeli ile performans sonuçlarını karşılaştırmışlardır. Sonuçların karşılaştırmasına bakıldığında özellikle InceptionV3 ve DenseNet121 modelleri en başarılı performansı göstermişlerdir. Bu çalışma ile otonom sistemlerde hassas tarım uygulamalarına yönelik sulama, gübreleme ve ilaçlama gibi süreçlerin bitkinin doğru evresinde uygulanmasına katkı sağlayabileceğini vurgulamışlardır (Karahanlı ve Taşkın 2024).

Başer yaptığı çalışmada, konumlandırma temelli sistemlerin hassas tarım uygulamalarındaki rolünü incelemiştir. GPS tabanlı konumlama teknolojilerinin tarımsal faaliyetlerde yaygın olarak tercih edilmesine rağmen yüksek maliyet ve istenilen doğruluğa her zaman ulaşamaması gibi nedenlerle tek başlarına yeterli olmadığı belirtilmiştir. Bu sorunun üstesinden gelmek amacıyla sabit referans noktalarına dayalı bir yersel koordinat sistemi ile GPS destekli hata düzeltme yaklaşımları bir arada kullanılarak daha hassas bir konum hesaplama altyapısı geliştirilmiştir. Kurulan sistemin, özellikle hassasiyetin kritik olduğu ilaçlama ve gübreleme uygulamalarında etkili sonuçlar verdiği ifade edilmiştir. Tasarlanan sistem, mobil robot üzerine entegre edilerek sensör füzyon yapısı ve RTK destekli GPS birimi sayesinde test edilmiş ve yapılan denemelerde ortalama konumlama sapması yaklaşık 11 cm olarak ölçülmüştür (Başer 2022).

İnan ve Karcı tarafından gerçekleştirilen çalışmada, tarımda ağaç ilaçlaması amacıyla otonom olarak görev yapabilen bir drone sistemi geliştirilmiştir. Geliştirilen sistemde, ilaçlama sürecinin verimliliğini artırmak ve insan müdahalesine olan ihtiyacı en aza indirmek hedeflenmiştir. Çalışmada, drone'ların sahadan aldığı görüntüler üzerinden ResNet50 tabanlı DeepForest ile ağaçların konumu belirlenmekte ve bu konum bilgilerine göre Kmax algoritması ile rota oluşturulmaktadır. Elde edilen sonuçlara göre, geleneksel yöntemlerle yapılan ilaçlamaya kıyasla %35-40 oranında ilaç tasarrufu sağlanabileceği ifade edilmiştir. Aynı zamanda, drone sisteminin zorlu arazilere erişimde avantaj

sağladığı, daha homojen ilaçlama sunduğu ve insan sağlığı üzerindeki riskleri azalttığı da vurgulanmıştır (İnan ve Karci 2021).

Karaçal çalışmasında drone'ların sürü kontroölünü ve kümelenmesi incelemiştir. Sürülerin matematiksel modellenmesinde Parçacık Sürü Optimizasyonu (PSO), formasyon kontrolü için PD tabanlı bir kontrol algoritması ve sürüdeki bireylerin birbirleri ile çarpışmalarını engellemek için potansiyel fonksiyon kullanmışlardır. Çalışmasında ayrıca kuş sürüsü, Yapay Arı Kolonisi (Artificial Bee Colony – ABC), yarasa algoritması ve ateş böceği algoritmaları ile de simülasyon ortamında modelleme yapmış ve minimum arama performansını karşılaştırmıştır. Çalışmasında sürünün merkezi bir kontrolü olmayıp lider tarafından sürünün hareketleri gerçekleştirmişlerdir. Lider için YSA tabanlı bir model geliştirerek görevin türüne göre lider belirlemişlerdir. Belirlenen geometrik şekle drone'ların gitmesi için Matlab üzerinden analiz gerçekleştirmişlerdir. Ayrıca Gazebo simülasyon programında da drone'ların uçuş testlerini gerçekleştirmişlerdir (Karaçal 2022).

2.4.2 Uluslararası Çalışmalar

Uluslararası literatürde tarım alanında robotik çalışmaların hızla geliştiğı ve bu alandaki araştırmaların farklı odak noktaları etrafında şekillendiğı görölmektedir. Bu kapsamda yapılan çalışmalar, öncelikle tarım uygulamalarında kullanılan tekil drone teknolojilerinden başlayarak sürü drone sistemlerine kadar uzanan geniş bir yelpazeyi kapsamaktadır. Tarımda tek drone uygulamaları, genellikle ilaçlama, gübreleme ve hastalık tespiti gibi görevlerde etkinlik sağlarken sürü drone yaklaşımları ise daha geniş alanlarda koordineli görev paylaşımı ve yüksek verimlilik sağlamaktadır. Robotik teknolojilerin tarım alanındaki genel kullanımı, yapay zekâ destekli algılama sistemleri, otonom görev planlama algoritmaları ve farklı sensör entegrasyonlarıyla desteklenerek üretim süreçlerinde verimliliğı artırmakta ve iş gücü maliyetlerini azaltmaktadır. Bu kapsamda literatür incelemesi; tarım alanındaki robotik çalışmaları, drone'ları, tekil drone'ların tarım alanında kullanımını, sürü drone'ları ve tarım alanında kullanılan sürü drone'lar ile yapılan çalışmaları içermektedir.

Tarımda robotik sistemlerin kullanımına yönelik bazı çalışmalar özellikle hasat süreçlerine yoğunlaşmaktadır. Farklı ürün türleri için geliştirilen robotik toplama sistemleri, sensör ve görüntü işleme teknikleri ile desteklenerek otomatik hasadı mümkün kılmıştır. Bu kapsamda literatürde yer alan bazı örnek çalışmalar şu şekildedir:

- Elma (Silwal vd. 2017, Onishi vd. 2019),
- Portakal (Muscato vd. 2005),
- Çilek (Xiong vd. 2019, Xiong vd. 2020),
- Kiraz (Tanigaki vd. 2008),
- Kavun (Zion vd. 2014),
- Karpuz (Rong vd. 2022),
- Domates (Yaguchi vd. 2016, Lili vd. 2017, Feng vd. 2018, Ling vd. 2019),
- Salatalık (Fernandez vd. 2018, Mao vd. 2020) ,
- Biber (Lehnert vd. 2017, Arad vd. 2020) örnek olarak verilebilir.

Li ve Duan yapmış oldukları çalışmada drone'ların engellerden kaçınmaları için engel kaçınma ve rota planlama algoritmalarını incelemişlerdir. İncelemelerde, klasik yöntemlerin (Dijkstra, A*, Yapay Potansiyel Alan yöntemi) basit çevre koşullarında başarılı olduklarını ancak karmaşık durumlarda yeterli olmadıklarına değinmişlerdir. Geliştirilmiş yöntemlerin (PSO, Karınca Kolonisi Optimizasyonu (Ant Colony Optimisation – ACO), Genetik Algoritma (GA), RL karmaşık ve dinamik ortamlarda daha çok başarılı olduklarına ama arka planındaki iş yükü zaman aldığı için gerçek zamanlı uygulamalarda sınırlılıklar getirdiklerine değinmişlerdir (Li ve Duan 2025).

Mesías-Ruiz ve arkadaşları domates ve mısır tarlalarındaki yabancı ot tespiti için veri seti oluşturmuşlardır. Oluşturulan veri setinde domates tarlalarında karşılaşılan *Cyperus rotundus*, *Portulaca oleracea* ve *Solanum nigrum*, mısır tarlalarında ise *Atriplex patula*, *Chenopodium album*, *Convolvulus arvensis*, *Datura ferox*, *Lolium rigidum* ve *Sorghum halepense* gibi yabancı otlar bulunmaktadır. Drone ile 11 m yükseklikten elde edilen 67558 adet görüntü üzerinden etiketleme yaparak veri seti oluşturulmuştur. Araştırmacılar veri setinin erken dönemlerde yabancı otların tespit edilerek kimyasal kullanımının azaltılmasında önemli etkisi olduğunu vurgulamışlardır (Mesías-Ruiz vd. 2025).

Apolo-Apolo ve arkadaşları drone ile elde edilen görüntülerden portakal ağacındaki meyvelerin verim tahminini ve meyve boyut tahminini gerçekleştirmişlerdir. 3 yıllık veri toplama süreci sonucunda Faster R-CNN kullanarak meyvelerin tespiti gerçekleştirilmiştir. Modelin tahmini ile hasat gerçekleştirildikten sonra elde edilen meyve sayısı karşılaştırılmış ve modelin % 6.59 hata oranıyla tahmin ettiği belirlenmiştir. Ayrıca meyve boyutlarını da %4.53'lük bir hata oranı ile tahmin etmişlerdir. Çalışmada ayrıca Uzan – Kısa Vadeli Bellek (Long Short Term Memory – LSTM) yöntemi ile ağaç başına ve toplam verim tahminleri yapılmıştır. Bu tahminler uzman kişilerin tahminleri ile karşılaştırılmıştır. Tahminlerdeki hata oranlarında uzmanların %13.74 ve önerilen yöntemde ise %7.22 hata oranı bulunmuştur. Bu sonuçlara bakılarak drone tabanlı ve derin öğrenme tabanlı yöntemlerin geleneksel yöntemlere göre daha verimli sonuçlar elde ettiği gözlemlenmiştir (Apolo-Apolo vd. 2020).

Altaş ve arkadaşları yaptıkları çalışmalarında şeker pancarında görülen Cercospora yaprak leke hastalığının durumunu incelemişlerdir. 12 farklı zamanlarda drone ile elde edilen veriler görüntü işleme teknikleri yardımıyla işlenmiş ve K-means algoritmasıyla sağlıklı/hastalıklı tespiti yapılmıştır. Aynı görüntülerin uzmanlar tarafından incelenmesi ile elde edilen sonuçların yaklaşık olarak birbirine eşit çıkmasıyla tasarlanan sistemin güvenilirliğini vurgulamışlardır. Böylelikle şeker pancarı yaprağında oluşan bir yaprak leke hastalığının varlığını ve şiddetini drone görüntüsü üzerinden öğrenmek mümkün olmuş ve insan işgücüne olan ihtiyacın azaldığı vurgulanmıştır (Altaş vd. 2018).

Kerkech ve arkadaşları üzüm bağlarında drone ile alınan görüntülerden mildiyö hastalığının tespitini ve haritalandırmasını gerçekleştirmişlerdir. 25 m irtifadan RGB ve multispektral kameralarla elde edilen görüntüler öncelikle image registration algoritması ile birleştirilmiştir. Birleştirme sonucunda görüntü kaymaları optimize edilmiş veri çoğaltma yöntemi ile çoğaltılmıştır. Yarı otomatik etiketleme yöntemi ile veriler etiketlenmiş ve SegNet mimarisinde eğitim gerçekleştirmişlerdir. Eğitim sonucunda yapraklarda %92'lik ve asmalarda ise %87'lik bir doğruluk oranı elde etmişlerdir. Sonuçların tek bir kamera ile yapılan çalışmalarda göre daha başarılı olduğunu belirtmişlerdir (Kerkech vd. 2020).

Chu ve Bao çalışmalarında tarımsal alanlarda ortaya çıkan zararlıların tespiti için yeni bir yaklaşım sunmuşlardır. Çalışmalarında Self-Aktivasyon Optimization Convolutional Neural Network (SAO-CNN) isimli modellerini tanıtmışlardır. Model, öz-denetimli öğrenme ve ConvLSTM yapıları ile güçlendirilerek hem zamansal özellikler hem de mekânsal özellikler etkin şekilde çıkarılarak değişken ışık koşulları ve karmaşık arka planlarda daha kararlı sonuçlar elde etmişlerdir. Ayrıca rota planlamasında ACO ve PSO kullanarak drone'ların enerji tüketimlerini minimum seviyede tutmayı amaçlamışlardır. Deneysel çalışma sonuçlarına göre önerilen yaklaşım, mevcut yöntemlere kıyasla %29.2 daha kısa uçuş süresi, %32 daha düşük enerji tüketimi sağlamıştır. Ayrıca %91.2 sınıflandırma doğruluğu ve 0.89 duyarlılık elde etmişlerdir (Chu ve Bao 2025).

Usman ve arkadaşları domates bitkisindeki yaprak hastalıklarının gerçek zamanlı tespitini gerçekleştirmişlerdir. PlantVillage veri setinden seçilen 9 hastalık sınıfı ve sağlıklı yaprak görüntüleri kullanılarak EfficientNet-B5 modelinde eğitimi gerçekleştirmişlerdir. Laboratuvar ortamında %99.8'lik bir başarı oranı elde ederken gerçek zamanlı uygulama da %96'lık bir başarı oranı elde etmişlerdir. Ayrıca tasarladıkları arayüz programından kullanıcıların drone kontrolü sağlamasını, canlı video gösterimini ve anlık hastalık tespitini görmelerini sağlamışlardır (Usman vd. 2024).

Chen ve arkadaşları meyve ağaçlarına büyük zararlar veren Tesseractoma papillosa'ların tespitine ve ilaçlanmasına yönelik çalışma gerçekleştirmişlerdir. Keşif drone'u ile zararlıların görüntüleri toplanmış ve YOLO mimarisi ile farklı yaşam evreleri tespit edilmiştir. Ayrıca tespit edilen konumlar buluta aktararak hem takip edilmesi hem de ilaçlama yapacak olan drone'un en uygun rota üzerinden ilaçlama yapmasına imkân sağlamıştır. Böylelikle sadece ilaçlama yapılacak olan alanlar belirlenerek pestisit kullanımında %70'e yakın azaltıldığı ve iş gücü olarak da yaklaşık %50'ye yakın azalma olduğunu vurgulamışlardır. Tasarlanan sistemin kullanılmasıyla %95 oranında zararlıların kontrol edildiğini belirterek tarım alanında drone sistemlerinin YZ entegrasyonu ile verimli bir şekilde kullanılabileceğini belirtmişlerdir (Chen vd. 2021).

Mattivi ve arkadaşları mısır tarlalarında ortaya çıkan yabancı otların tespit edilmesi için drone tabanlı bir çalışma gerçekleştirmişlerdir. Çalışmalarında Askeri Yük

Sınıflandırması (Military Load Classification – MLC), YSA ve Nesne Tabanlı Görüntü Analizi (OBIA) yöntemlerini karşılaştırmışlardır. Elde edilen sonuçlara göre YSA yöntemi %99.55, MLC yöntemi %99.50 ve OBIA yöntemi %99.38 doğrulukla başarı ile yabancı otların haritalandırmasını yapmışlardır (Mattivi vd. 2021).

Catala-Roman ve arkadaşları portakal bahçesinde ortaya çıkan yabancı otların (*Araujia sericefera* ve *Cortaderia selloana*) tespiti ve haritalandırmasını drone ile gerçekleştirmişlerdir. Çalışmalarında YOLO mimarisi kullanarak yapılan sınıflandırmada *Araujia* yabancı otunu %73 ve *Cortaderia* yabancı otunu %75 doğruluk oranıyla tahmin etmişlerdir. Yapılan uçuşlarda tespit edilen bitkilerin koordinatları GPS ve RTK üzerinden kayıt altına alınmakta ve ilaçlama drone'ları ile ilgili alanlar ilaçlanmaktadır. Bu çalışma sonucunda yabancı otun tespit edilmesi ve yok edilmesinin klasik yöntemlere göre hektar başına yaklaşık 2500€'luk tasarruf sağladığını maliyet analizleri sonucunda ifade etmişlerdir (Catala-Roman vd. 2024).

Mundappat Ramachandran ve arkadaşları tarım arazilerinde sürdürülebilir su yönetimi ve bitki sağlığı üzerine Solar Agro Savior (SAS) sistemini tanıtmışlardır. Sistem, drone ile tarım arazilerindeki bitkilerin sağlığını izleme, nem, sıcaklık ve toprak verilerine dayanarak güneş enerjisi ile çalışan sulama sistemini kontrol etmektedir. SAS sisteminde LSTM ve dikkat mekanizmaları kullanarak hem zaman serilerinde uzun süreli bağımlılıklar göz önüne alınabilmekte hem de çevresel faktörlere ağırlıklar verilerek doğru sınıflandırma yapılabilmektedir. Bitki sağlığının tespitinde YOLO ve LSTM'nin ardışık veri işleme kabiliyetleri birleştirilmiştir. Deneysel çalışmalar sonucunda %99 doğruluk, %97.8 kesinlik, %98.4 duyarlılık ve %98.1 F1 skoru elde etmişlerdir (Mundappat Ramachandran vd. 2025).

Roque-Claros ve arkadaşları tarım uygulamalarında kullanılan drone'ların rota planlamaları için yeni bir yöntem önermişlerdir. Önerdikleri yöntem iki katmandan oluşmakta olup bunlar küresel ve yerel yol planlama katmanlarıdır. Küresel yol planlama katmanında geliştirilmiş GA ve Bulanık Mantık (BM) kullanmışlardır. Böylelikle enerji verimliliği ve yol uzunluğu optimize edilerek en uygun rotalar elde edilmiştir. İkinci katmanda ise geliştirilmiş D* algoritması ile dinamik engellere ve ani değişen çevre

koşullarına adaptasyonu sağlanmaktadır. MATLAB ortamında gerçekleştirilen simülasyon çalışmasında PSO'ya göre %51.7, ACO'ya göre 47.5, GA'ya göre %47.5 daha kısa rotalar elde etmişlerdir. Rotanın optimizasyonunun ise ortalama 5.54 s gibi bir sürede tamamlandığını belirtmişlerdir. Sonuç olarak makine öğrenmesi ve sürü zekâsı ile geliştirilen hibrit rota planlama modelinin, akıllı tarımda drone kullanımını daha güvenli, verimli ve sürdürülebilir hale getirebileceğini vurgulamışlardır (Roque-Claros vd. 2024).

Rexhaj ve arkadaşları gerçek zamanlı tarımsal uygulamalarda daha kapsamlı bir şekilde değerlendirme sunmak için Drone Optimized Performance Score (DOPS)'u önermişlerdir. Çalışmalarında domateslerin olgunluk seviyelerinin tespitini drone üzerinden gerçekleştirmişlerdir. DOPS, klasik ölçüm değerlendirmelerinden farklı olarak doğruluk, Saniyedeki Görüntü Sayısı (Frame Per Second – FPS) ve gecikme parametrelerini dikkate almaktadır. Çalışmalarında MobileNetV2 ve ResNet50 derin öğrenme mimarileri karşılaştırılmıştır. Başlangıçta MobileNetV2 mimarisi düşük gecikme ve yüksek hız avantajı sayesinde drone uygulamalarına daha uygun görülse de ResNet50 üzerinde yapılan optimizasyonlarla hem yüksek doğruluk (%98–99) elde edilmiş hem de DOPS skorunu artırarak gerçek zamanlı uygulamalarda kullanılabilir hâle gelmiştir. Yapılan test işleminde ResNet50 mimarisi 20 FPS üzerinde ve 45 ms gibi düşük bir gecikme ile MobileNetV2'yi geride bırakmıştır. Sonuç olarak drone tabanlı görüntüleme yapılan çalışmalarda derin öğrenme modellerinin entegrasyonunda yalnızca doğruluğun değil hız ve gecikmenin de kritik parametreler olduğunu vurgulanmaktadır (Rexhaj vd. 2025).

Luo ve Oracevic otonom hareket eden sürü drone'larında anomali tespitine yönelik Bi-LSTM tabanlı bir çalışma gerçekleştirmişlerdir. Çalışmalarında drone'ların çeşitli alanlardaki yaygın kullanımlarında oluşabilecek çarpışma, siber saldırı ya da sensör arızası gibi durumlara değinmişlerdir. Bu gibi durumlardan olumsuz etkilenmemek için Bi-LSTM tabanlı ve dikkat mekanizması entegrasyonlu anomali tespiti modeli önermişlerdir. Modelde zaman serilerinin ileri ve geri yöndeki analizleri incelenerek sensörlerden özellik çıkarımları yapılmaktadır. Gazebo simülasyon ortamında gerçekleştirilen çalışmada normal uçuş, rüzgârlı havada uçuş, siber saldırılı bir uçuş ve GPS sensör arızası olan bir uçuş senaryoları oluşturulmuştur. Bu senaryolarda anomali

tespitinde %99.3, siber saldırılarda %99.8 ve sensör arızası tespitinde ise %99.9'luk bir başarı oranı elde etmişlerdir. Ayrıca çalışmanın simülasyon ortamında yapılma sınırlılığından ve önerilen modelin yüksek hesaplama maliyetinden de bahsetmişlerdir (Luo ve Oracevic 2024).

Marek ve arkadaşları sürü robotlarının kontrol algoritmalarının geliştirildiği ve test edildiği bir simülasyon tabanlı yaklaşım önermişlerdir. Çalışmalarında SITL yöntemi kullanılarak drone'ların verimliliğini, uyum içinde hareketlerini ve çarpışma önleme performanslarını incelemişlerdir. Simülasyonda konumlama hatalarına, hız değişimlerine ve gecikme hatalarına karşı sistemin değerlendirmesini yapmışlardır. RTK GPS'li drone'un 100 ms altındaki bir gecikme süresinde 5 m/s hızla hareket edebileceğini, normal GPS ile yapılan testte ise güvenli hız limitinin 2 m/s olduğunu göstermişlerdir. Lider – takipçi sürü senaryolarında bu hataların birikerek artması sebebiyle RTK GPS'li drone hızının 2 m/s'nin altında düştüğünü, normal GPS'li drone'un ise 1 m/s hızın altına düştüğünü belirtmişlerdir. Ayrıca iletişim gecikmelerindeki aksaklıkların sensör verilerinin işlenmesinde de problem oluşturabileceğini, bu sebeple gerçek donanımlarla yapılacak çalışmalarda bunların göz önünde bulunması gerektiğini vurgulamışlardır (Marek vd. 2024).

Belkin ve arkadaşları sürü drone'larının belirsiz ve karmaşık ortamlarda güvenilir bir navigasyon ve kontrol sağlamak için hibrit bir sürü yaklaşımı önermişlerdir. Çalışmalarında iletişim tabanlı dağıtık kontrol algoritmalarının ölçeklenebilir ve koordinasyon gücü ile görsel navigasyon tekniklerinin (özellikle V-SLAM) GPS olmayan ortamlardaki otonom avantajlarını birleştirmişlerdir. Göreve göre değişmekle birlikte 10-50 drone ile test işlemi gerçekleştirmişlerdir. Test sonucunda hibrit yöntemin başarı oranı %92.5 olurken iletişim tabanlı %78.3 ve görsel tabanlı yöntemlerin başarı oranı %83.7 olarak belirlenmiştir. Ayrıca görev süresinin %15 daha kısa sürdüğünü, enerji verimliliğinin de yaklaşık olarak %15 olduğunu belirtmişlerdir (Belkin vd. 2025).

Ganduri ve Pathri yapmış oldukları çalışmada sürü drone'larının karmaşık çevre koşullarında engelden kaçmalarını ve havada istenilen şekilde durabilmeleri üzerine çalışma gerçekleştirmişlerdir. Çalışmaları MATLAB ortamında 8-12 arasında drone ile

farklı senaryolarda gerçekleştirmişlerdir. Velocity Obstacle yaklaşımı ile engellerden kaçınmayı, PSO ile en uygun rotayı bulmayı ve Rendezvous Algoritmaları (RA) ile de drone'ların istenilen şekilde formasyonlara girmelerini sağlamışlardır. Böylelikle verilecek olan göreve göre sürü robotlarının dinamik ortamlarda güvenli navigasyon ve esnek formasyon sağlamalarını önerdikleri yöntemle gerçekleştirmişlerdir (Ganduri ve Pathri 2025).

Edmons ve arkadaşları drone'ların ve İKA'ların sınırlı enerji ile verilen görevlerin yerine getirilmesine yönelik bir çözüm sunmuşlardır. Önerdikleri yöntemde robotların enerji kapasitelerine ve arazi şartlarına bağlı olarak en uygun çözünürlükte drone ile görüntü almışlardır. İKA üzerinde bulunan şarj üniteleri sayesinde drone'lar, enerji ihtiyaçları bir nebze karşılanarak büyük arazilerde kesintisiz görev yapabilmektedirler. Simülasyon ortamında gerçekleştirilen çalışmada drone'ların almış oldukları görüntülerin çözünürlükleri mevcut yöntemlerle karşılaştırıldığında daha kısa sürede daha yüksek çözünürlük elde etmişlerdir (Edmonds vd. 2021).

Olçay ve arkadaşları önceden herhangi bir harita bilgisi olmadan bilinmeyen alanlarda ve birbirine çarpmadan hareket eden sürü robotlarının kontrol algoritmalarını geliştirmişlerdir. Robotlar sensörlerden elde ettikleri verilerle ve komşulardan gelen verilerle eş zamanlı olarak otonom bir şekilde hareket etmektedirler. Çalışmada flocking davranışından esinlenilmiş ve robotların birbirleriyle uyumlu hareket edebilmesi için yakınsama, ayrışma ve hizalanma kuralları uygulanmıştır. Ayrıca potansiyel alan tabanlı yapay kuvvetler sayesinde robotlar hem birbirleriyle güvenli mesafeyi koruyabilmekte hem de sabit engellerden kaçabilmektedir. Simülasyon ortamında çeşitli (zigzag, koridor, yarı dairesel ve çoklu küçük engellerin bulunduğu) senaryolarda 15– 20 robotla test işlemleri gerçekleştirilmiştir. Test sonucunda hiçbir engele çarpmadan grup bütünlüğünü büyük ölçüde koruyarak hedefe ulaşabildiği gösterilmiştir (Olçay vd. 2020).

Guo ve arkadaşları sürü drone'ları için hibrit akıllı algoritmalarla yol planlama yaklaşımını önermişlerdir. Önerilen yöntem, yapay potansiyel alan algoritması ile başlangıçta çarpışmasız global rotalar üretmekte ve RL tabanlı dinamik karar mekanizmaları ile gerçek zamanlı çevresel geri bildirimlere göre bu rotalar

güncellenmektedir. Ayrıca GA ile karmaşık tarım arazilerinde daha güvenilir ve verimli bir rota planlama modeli oluşturulmuştur. Test işlemleri sonucunda 1500 m'ye kadar olan görevlerde klasik yöntemlere göre %18 – 22 aralığında daha kısa sürede rotalama işlemini gerçekleştirmiştir. Heterojen ve engellerin yoğun olduğu tarım arazilerinde algoritmanın kararlılığının devam ettiği ve yaklaşık %87'lik bir başarı oranı elde ettiğini vurgulamışlardır (Guo vd. 2025).

Khaldi ve arkadaşları son yılda yayınlanan toplam 458 akademik makalenin incelemesini yaparak sürü robotlarının çevresel sürdürülebilirlik, ekosistem izleme ve keşif gibi alanlardaki rollerini incelemişlerdir. Yapılan çalışmada sürü robotlarının tekli robotlara göre daha ölçeklenebilir, dayanıklı ve esnek çözümler sunduğunu belirtmişlerdir. Özellikle drone'ların çeşitli alanlarda verilen görevlerin yerine getirilmesinde etkili sonuçlar çıkardığını, tarım alanında (hassas ekim, gübreleme, bitki sağlığı izleme), ekosistem alanında (yangın kontrolü, vahşi yaşam takibi, biyolojik çeşitlilik izleme) ve afet yönetimi (arama-kurtarma, çevresel hasar değerlendirmesi) alanlarında ön planda olduğunu vurgulamışlardır (Khaldi vd. 2025).

Al-Amin ve arkadaşları yapmış oldukları çalışmada yenileyici tarım uygulamalarında sürü robotik sistemlerin bitki üretimine dayalı şerit şeklinde ekimini ve bitki – hayvancılık entegre olan senaryoları geleneksel yöntemlerle karşılaştırmışlardır. Yapılan çalışma sonucunda bitki ekiminde sürü robotlarının geleneksel yöntemlere göre çok daha avantajlı ve düşük maliyetli olduğunu belirtmişlerdir. Ancak hayvancılığın senaryoya dahil olmasında maliyetlerin yüksek olması sebebi ile dezavantajlar ortaya çıkmıştır. Araştırmacılar, sürü robotları ile otonom olarak yapılan işlemlerde iş gücündeki azalma, uzun süre çalışabilme süreleri ve maliyet olarak daha cazip olması tarım alanında sürü robotlarının kullanımında etkili olduğunu vurgulamışlardır (Al-Amin vd. 2025).

Pretto ve arkadaşları hassas tarım uygulaması için hem drone hem de İKA robotları ile bir sürü çalışması gerçekleştirmişlerdir. Çalışmalarında drone ile havadan elde edilen görüntülerin analizi yapılarak yabancı otların tespit edilmesi ve haritalandırılması yapılmaktadır. Sonrasında İKA ilgili bölgeye giderek yabancı otların hem mekaniksel hem de kimyasal kullanarak yok edilmesini sağlamaktadır. Avrupa Birliğinden alından

destekle gerçekleştirilen projenin sonucunda geleneksel yöntemlere göre kullanılan kimyasallarda önemli ölçüde azalma olduğunu ve yapılan müdahalelerin hedefe yönelik olarak gerçekleştirildiğini göstermiştir (Pretto vd. 2020).

Mukherjee ve arkadaşları tarım alanında farklı donanımlara sahip sürü drone'larının iş yükünün dengelenmesi üzerine çalışma gerçekleştirmişlerdir. Çalışmalarında kamera gibi yüksek veri üreten drone'lar ve basit fiziksel sensörleri olan drone'ların iş yüklerini eşit dağıtabilmek için dağıtık veri işleme yöntemi önermişlerdir. Bu yöntemle göre kamera sensörüne sahip drone'lar görüntüleri diğer drone'lara aktararak görüntülerin farklı farklı drone'larda hızlı bir şekilde işlemişlerdir. İş yüklerinin dağıtımında nash pazarlık yöntemi kullanmışlardır. Çalışmayı hem gerçek ortamda (birkaç drone) hem de simülasyon ortamında (100 ve üzeri) gerçekleştirmişlerdir. Elde edilen sonuçlar geleneksel yöntemlerle karşılaştırıldığında daha düşük gecikme ve ölçeklenebilirlik sağladığı görülmüştür. Ayrıca 200 drone ile yapılan simülasyon çalışmasında önerilen yöntem ile ağın işlem hızının 2 katına kadar çıktığı görülmüştür. Bu çalışma ile heterojen yapıya sahip sürü drone'larının tarım alanında etkin bir biçimde kullanılabileceğini göstermektedir (Mukherjee vd. 2020).

2.4.3 Literatürün Genel Değerlendirilmesi

Literatür taraması sonucunda hem ulusal hem de uluslararası çalışmaların tarımsal üretimde robotik sistemlerin kullanımına yönelik önemli katkılar sunduğu görülmektedir. Ulusal düzeyde gerçekleştirilen çalışmalar, genellikle konumlandırma sistemleri, tekil drone uygulamaları ve temel nesne tespit yaklaşımlarına odaklanmış, saha koşullarına uygun maliyetli çözümler geliştirmeyi amaçlamıştır. Ancak bu çalışmaların büyük bir kısmı sınırlı sayıda araç üzerinde denenmiş, sürü yapıları veya otonom görev paylaşımı gibi daha ileri düzey özellikler henüz yeterince ele alınmamıştır.

Uluslararası literatürde ise çoklu robot sistemleri, nesne tespiti için derin öğrenme tabanlı yaklaşımlar ve tarımda hassas ilaçlama konuları ön plana çıkmaktadır. Özellikle YOLO, CNN ve benzeri derin öğrenme modelleri kullanılarak bitki hastalıklarının ya da ürün olgunluğunun tespiti üzerine yoğunlaştığı, ayrıca drone sürülerinin koordineli görev icrası için optimizasyon algoritmalarının yaygın şekilde kullanıldığı dikkat çekmektedir.

Bu çalışmalar, geniş tarım arazilerinde zaman ve iş gücü tasarrufu sağlamanın yanı sıra kimyasal kullanımının azaltılmasına da katkı sunmaktadır. Literatürde yer alan çalışmalara ilişkin özet bilgiler, yöntemsel yaklaşımlar ve kullanılan algoritmalar Çizelge 2.1’de sunulmuştur.

Çizelge 2.1 Literatürdeki sürü drone sistemlerine ilişkin çalışmaların karşılaştırmalı özeti

Çalışma	Konu	Yöntem/Model	S	O	NT
Altaş vd. 2018	Drone ile şekerpancarındaki yaprak leke hastalığı tespiti	K-Means	X	X	✓
Apolo-Apolo vd. 2020	Drone ile portakal ağacındaki meyvelerin verim ve boyut tahmini	Faster R-CNN, LSTM	X	X	✓
Kerkech vd. 2020	Drone ile üzüm bağlarındaki mildiyö hastalığının tespiti	SegNet	X	X	✓
Pretto vd. 2020	Sürü drone ve İKA ile yabancı ot tespiti ve yok edilmesi	RF, FCN, NB, MPC	✓	X	✓
Mukherjee vd. 2020	Sürü drone kontrolü	Dağıtık kontrol	✓	✓	X
İnan ve Karci 2021	Ağaç ilaçlama	ResNet50	X	X	✓
Mattivi vd. 2021	Drone ile mısır arazilerindeki yabancı ot tespiti	MLC, YSA, OBIA	X	X	✓
Karaçal 2022	Sürü kontrolü	ACO, ABC, Yarasa	✓	✓	X
Başaran 2022	Fıstık meyvesinin sınıflandırılması	SVM, DarkNet53	X	X	✓
Terzi vd. 2023	5 farklı üzüm, yaprak salkım tespiti	CNN	X	X	✓
Karahanlı ve Taşkın 2024	Drone ile ayçiçeği bitkisinin gelişim evresinin takibi	AlexNet, InceptionV3, ResNet101, Xception, DenseNet, MobileNet	X	X	✓
Catala-Roman vd. 2024	Portakal bahçelerindeki yabancı otların tespiti ve haritalandırması	YOLO	X	X	✓
Roque-Claros vd. 2024	Tarımda drone’lar için yeni bir rota planlama algoritması	Önerilen algoritma, PSO, ACO, GA	X	✓	X
Luo ve Oracevic 2024	Sürü drone’larda anomali tespiti	Bi-LSTM	✓	X	X
Ergün 2025	Tarım arazisinde yabancı ot tespiti	ResNet, DenseNet, VggNet	X	X	✓
Chu ve Bao 2025	Drone ile tarım alanında zararlıların tespiti	SAO-CNN, ACO, ConvLSTM, PSO	X	✓	✓
Mundappat Ramachandran vd. 2025	Tarımda sürdürülebilir su yönetimi ve bitki sağlığı	YOLO, LSTM	X	X	✓
Belkin vd. 2025	Sürü drone’ların karmaşık ortamlarda güvenli uçuşu ve kontrolü	Dağıtık kontrol, V-SLAM	✓	✓	X
Ganduri ve Pathri 2025	Sürü drone’ların engelden kaçma ve istenilen şekilde durması	VO, PSO, RA	✓	✓	X
Guo vd. 2025	Sürü drone’lar için hibrit rota planlama	RL, GA	✓	✓	X

S: Sürü, O: Optimizasyon, NT: Nesne Tespiti

Genel olarak deęerlendirildięinde, ulusal alıřmaların altyapı oluřturma ve uygulama rnekleri sunma aısından nemli olduęu, uluslararası alıřmaların ise sr robotları, yapay zek destekli nesne tespiti ve grev planlama konularında daha ileri dzeyde olduęu sylenebilir. Bu tez alıřması, ulusal ve uluslararası literatrn eksik ve gl ynlerini dikkate alarak, sr drone sistemlerini tarımsal ilalama zelinde nesne tespiti ve optimizasyon yntemleriyle birleřtiren btnleřik bir mimari ortaya koymaktadır. Bylelikle hem ulusal alıřmalarda sınırlı kalan sr yaklařımı glendirilmiř hem de uluslararası literatrdeki gncel yntemler yerel bir baęlama uyarlanarak zgn bir katkı saęlanmıřtır.



3. MATERYAL ve METOT

Bu bölümde, geliştirilen yapay zekâ destekli sürü drone sisteminin oluşturulmasında kullanılan yöntemsel yaklaşımlar ve materyaller ele alınmaktadır. Öncelikle sistemin geliştirilmesine temel oluşturan simülasyon ortamı, protokoller ve yazılım araçları açıklanmış; ardından mimari yapı, yazılım-donanım entegrasyonu ile veri akış süreçleri ortaya konulmuştur. Son olarak, kurulan sistemin performansının nesnel biçimde değerlendirilebilmesi amacıyla dikkate alınan kriterler tanımlanmıştır.

3.1 Sürü Robotiğinin Simülasyon Platformu

Bu çalışmada, sürü drone sisteminin test edilmesi amacıyla gerçek dünya koşullarını temsil eden bir simülasyon ortamı tasarlanmıştır. Bu sayede, gerçek zamanlı uçuş testlerine ihtiyaç duyulmadan farklı senaryolar güvenli, düşük maliyetli ve tekrar edilebilir şekilde sanal ortamda incelenmiştir. Simülasyon altyapısı; Gazebo, ArduPilot, SITL, MAVLink, Robot İşletim Sistemi Tabanlı MAVLink (MAVLink on ROS – MAVROS) ve ROS yazılımlarının bütünleşik entegrasyonu ile oluşturulmuştur. Ayrıca kullanılan GPS, IMU, LiDAR ve kamera sensörlerinden ve YOLO tabanlı nesne tespit algoritmasının da bu yapıya entegrasyonu sağlanmıştır. Bu altyapının kurulmasında ilk adımı, sanal ortamda drone'ların fiziksel hareketlerinin, sensör tepkilerinin ve çevresel koşulların gerçeğe yakın şekilde modellenmesini sağlayan Gazebo simülasyon programı oluşturmaktadır.

3.1.1 Gazebo Simülasyon Programı

Robotik ve otonom sistemlerin geliştirilmesinde, gerçek donanımlar üzerinde yapılan denemeler çoğu zaman maliyetli, riskli ve zaman alan süreçlerdir. Bu sebeple, son yıllarda robotik topluluklarında simülasyon tabanlı yazılım altyapıları önemli bir araştırma ve geliştirme aracı hâline gelmiştir. Gazebo, bu kapsamda en yaygın kullanılan açık kaynaklı simülasyon ortamlarından birisidir. Özellikle sürü robotları, İHA, İKA ve otonom sistemlerin test edilmesinde, gerçek dünya koşullarına oldukça yakın sonuçlar sunabilmesi nedeniyle tercih edilmektedir.

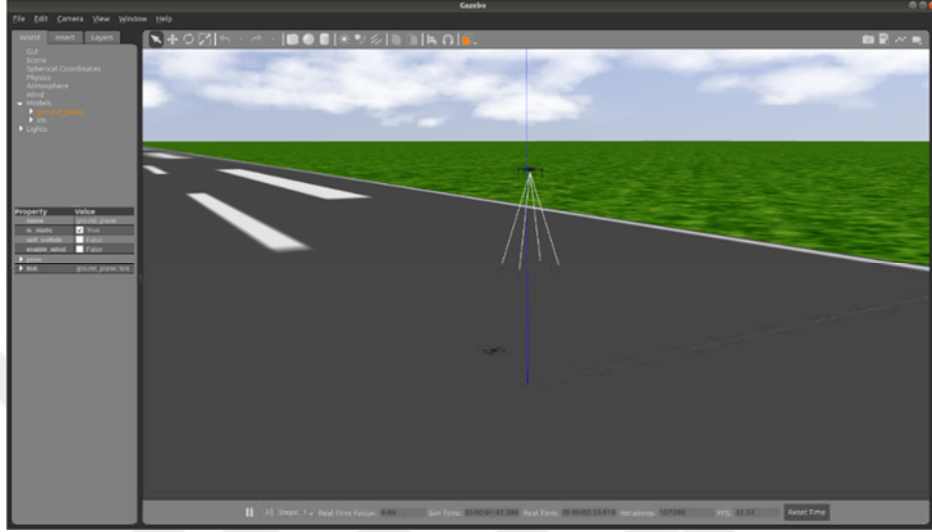
Gazebo, robotların fiziksel ortamlardaki davranışlarını detaylı bir şekilde modelleyebilmek için gelişmiş fizik motoru (ODE, Bullet, Simbody vb.) kullanır. Bu motor sayesinde robotların hareketleri, sensör tepkileri ve çevresel etkiler (yerçekimi, sürtünme, rüzgâr vb.) gerçeğe yakın şekilde simüle edilebilmektedir. Böylece araştırmacılar, farklı senaryoları güvenli bir şekilde deneyebilir, sistem üzerinde değişiklik yaparak performans ölçümleri gerçekleştirebilir ve gerçek testlere geçmeden önce olası hata veya eksiklikleri tespit edebilir.

Gazebo'nun öne çıkan özelliklerinden biri, 3 boyutlu modelleme ve görselleştirme yeteneğidir. Kullanıcılar ister hazır kütüphanelerden seçerek ister kendi tasarımlarını oluşturarak simülasyon ortamına farklı robot, araç, insan, yapı ya da doğal nesne eklemesi yapabilmektedir. Çalışma alanının zemini toprak, çimen veya asfalt gibi farklı yüzeylerle modellenmektedir. Ayrıca hava durumu, ışık koşulları veya engel yoğunluğu gibi çevresel faktörlerde de değişiklik yapılabilir. Bu esneklik, tarımsal robotik uygulamalarda, özellikle farklı arazi koşullarının simülasyonu açısından büyük avantaj sağlar.

Gazebo, çoklu sensör entegrasyonu için de güçlü bir altyapıya sahiptir. Kamera (Fortini vd. 2022, Deng vd. 2024), LiDAR (Giubilato vd. 2020, Udugama 2023), IMU (Manhães vd. 2016, Xiao vd. 2020) ve GPS (Xiao vd. 2020) gibi pek çok sensör simülasyona eklenebilir ve bunların çıktıları gerçek donanımlardaki verilerle benzer şekilde elde edilebilir. Bu çalışmada kullanılan GPS, IMU, LiDAR ve kamera sensörleri Gazebo ortamına entegre edilerek (Şekil 3.1) sürü drone sisteminin sanal testleri yapılmıştır. Örneğin kamera çıktıları YOLO tabanlı nesne tespit algoritmasına aktarılmış, LiDAR verileri ise engel algılama ve çarpışma önleme senaryolarında kullanılmıştır. Böylelikle, gerçek saha testlerinde elde edilebilecek veri akışı önceden modellenmiş ve sistemin işleyişi doğrulanmıştır.

Gazebo'nun bir diğer önemli avantajı ise ROS ile uyumlu çalışmasıdır. ROS, modüler bir yazılım mimarisi sunarak robotik uygulamalarda veri akışını, görev planlamayı ve sensör entegrasyonunu kolaylaştırmaktadır. Gazebo, ROS ile entegre edildiğinde sensörlerden alınan veriler ROS düğümlerine aktarılabilir, kontrol algoritmaları test edilebilir ve görev

planlama modülleri üzerinde simülasyon tabanlı deneyler yürütülebilir. Bu yapı, sürü robotları gibi çoklu araçların koordinasyonunun incelenmesinde kritik bir rol oynamaktadır.



Şekil 3.1 Gazebo simülasyon arayüzü.

Gazebo'nun akademik literatürde oldukça yaygın olarak kullanıldığı görülmektedir. Giubilato ve arkadaşları, Gazebo tabanlı bir simülasyon çerçevesi geliştirerek gezegen yüzeyi benzeri ortamlarda mobil robotların navigasyon ve lokalizasyon performanslarını test etmişlerdir (Deng vd. 2025). Ayala ve arkadaşları ise Gazebo, Webots ve V-REP simülatörlerini performans kriterleri üzerinden karşılaştırmış ve Gazebo'nun özellikle fiziksel gerçekçilik açısından öne çıktığını vurgulamıştır (Ayala vd. 2020). Udugama, ROS ile entegre Gazebo ortamında otonom mobil robotların iç ve dış mekân navigasyon yeteneklerini incelemiştir (Udugama 2023). Bu tür çalışmalar, Gazebo'nun yalnızca teorik değil aynı zamanda farklı alanlarda pratik bir araştırma aracı olduğunu göstermektedir.

Gazebo'nun bazı sınırlılıkları da bulunmaktadır. Öncelikle, yüksek işlem gücü ve grafik kartı gerektirmesi nedeniyle donanım ihtiyacı yüksektir. Gazebo farklı işletim sistemlerinde de kurulabilmekle ancak en verimli ve kararlı çalıştığı ortam Linux tabanlı sistemlerdir. Bu durum, özellikle yeni kullanıcılar için öğrenme sürecini zorlaştırabilmektedir. Bunun yanı sıra açık kaynaklı yapısı ve geniş kullanıcı topluluğu

sayesinde bu zorluklar genellikle topluluk desteği ile aşılabilmektedir. Çizelge 3.1’de Gazebo’nun avantajları ve dezavantajları verilmiştir.

Çizelge 3.1 Gazebo simülasyon programının avantaj ve dezavantajları.

<i>Avantajları</i>	<i>Dezavantajları</i>
Açık kaynaklı ve ücretsizdir.	Yüksek işlem gücü ve güçlü donanım (özellikle GPU) gerektirir.
Gerçeğe yakın fizik motorları ile uçuş ve çevresel etkileşimleri doğru şekilde simüle eder.	Yeni başlayan kullanıcılar için öğrenmesi zor olup, kurulum ve yapılandırma süreci karmaşıktır.
3D ortamda farklı arazi koşulları ve çevresel faktörler kolayca modellenenabilir.	Gerçek donanım koşullarındaki tüm çevresel etkiler (rüzgâr türbülansı, sensör gürültüsü) tam olarak yansıtılamaz.
Kamera, LiDAR, GPS, IMU, barometre gibi çok sayıda sensörün simülasyonu mümkündür.	Yoğun sensör ve çoklu robot simülasyonlarında yüksek sistem kaynağı tüketir, performans düşebilir.
ROS ile tam uyumludur; görev planlama, veri akışı ve sensör entegrasyonu kolaydır.	Bazı ileri düzey eklentilerin manuel düzenleme gerektirmesi, geliştirme süresini uzatabilmektedir.
Tekli robotların yanı sıra sürü robotlarının davranışlarını test etmeye olanak tanır.	Çok büyük ölçekli sürü senaryolarında (50 ve üzeri) gecikme ve senkronizasyon sorunları yaşanabilir.

Bu tez kapsamında Gazebo, sürü drone sisteminin tarımsal senaryolarda modellenmesi için temel simülasyon ortamı olarak seçilmiştir. Sistem, Ubuntu tabanlı bir işletim sisteminde çalıştırılmış ve ArduPilot ile entegre edilerek drone’ların uçuş dinamikleri, görev paylaşımı ve nesne tespiti senaryoları simüle edilmiştir. Gazebo üzerinde oluşturulan sanal tarım arazilerinde, sürü drone’ların keşif, hedef belirleme ve hassas ilaçlama görevleri başarıyla denenmiş; farklı koşullarda sistemin davranışları analiz edilmiştir. Bu kapsamda Gazebo, yalnızca simülasyon ortamı değil aynı zamanda sürü robot sistem mimarisinin geliştirilmesi için bir doğrulama aracı olarak da kullanılmıştır.

3.1.2 ArduPilot

ArduPilot, İHA, İKA ve deniz araçları gibi farklı platformlarda kullanılabilen, açık kaynaklı ve oldukça esnek bir uçuş kontrol yazılımıdır. İlk kez 2009 yılında geliştirilmiş olan ArduPilot, günümüzde dünya çapında geniş bir kullanıcı topluluğuna sahip olup hem

akademik arařtırmalarda hem de endüstriyel uygulamalarda yaygın olarak kullanılmaktadır. Açık kaynaklı yapısı sayesinde sürekli geliştirilmekte, yeni sensör entegrasyonları ve uçuş modları ile güncellenmektedir.

ArduPilot'un en önemli özelliklerinden biri, çoklu araç tiplerini desteklemesidir. Sabit kanatlı uçaklar, döner kanatlı drone'lar (quadcopter, hexacopter, octocopter vb.), Dikey İniş – Kalkış Yapan Sabit Kanatlı Drone (Vertical Take-Off and Landing – VTOL) sistemleri, İKA ve hatta denizaltılar bu yazılım üzerinden kontrol edilebilmektedir (İnt. Kyn. 3). Bu çok yönlülük, ArduPilot'u robotik sistemlerin simülasyonunda tercih edilen en güçlü yazılımlardan biri hâline getirmektedir. Şekil 3.2'te ArduPilot'un kullanım alanlarından örnekler gösterilmiştir.



Şekil 3.2 ArduPilot'un kullanım alanları.

Gazebo simülasyon ortamında drone'ların gerçekçi şekilde test edilebilmesi için ArduPilot yazılımının kurulması ve ayarlanması gereklidir. ArduPilot, terminal üzerinden çalıştırılarak uçuş kontrol kartının işlevlerini yazılım ortamında simüle eder ve motor, sensör ve uçuş komutlarının sanal ortamda denenmesine olanak tanır. Böylece SITL özelliği sayesinde fiziksel donanım kullanmadan yazılım tabanlı uçuş testleri yapılabilmekte ve geliştirme süreci hızlanmaktadır (İnt. Kyn. 4).

ArduPilot ile simülasyon ortamındaki drone'ların kontrolü Python veya C programlama dilleri üzerinden gerçekleştirilebilmektedir. Bu tez çalışmasında, Python dili tercih edilmiştir. Python programı ile uçuş kontrol kartı arasında DroneKit kütüphanesi kullanılarak bağlantı sağlanmış, haberleşme ise MAVLink protokolü üzerinden yürütülmüştür. Bu sayede yazılım tarafında geliştirilen kodlar, simülasyon ortamındaki uçuş kontrol sistemine doğrudan iletilmiş, çift yönlü veri aktarımı mümkün olmuştur.

ArduPilot'un kullanımında temel uçuş adımları belirli bir prosedürü takip etmektedir. Örneğin bir drone'un güvenli şekilde havalanabilmesi için aşağıdaki aşamaların tamamlanması gereklidir:

- Motorlara, motor sürücülerine ve uçuş kontrol kartına enerji verilmesi,
- Uçuş kontrol kartının GPS üzerinden konum doğrulaması yapması,
- Drone'un uçuş modları (Guided, Auto, Land, RTL vb.) arasından uygun olan uçuş moduna (Guided) geçirilmesi,
- Motorlara Arm komutunun gönderilerek en düşük devirlerde motorların düzgün çalıştığının kontrol edilmesi,
- Kalkış komutu ile istenilen irtifaya güvenli kalkışın gerçekleştirilmesi.

Bu adımlar, bir drone'un uçuşa başlaması için temel gerekliliklerdir. Koşullar sağlanmadığı takdirde kalkış işlemi gerçekleşmez. ArduPilot hem bu güvenlik önlemlerini sağlaması hem de uçuş sürecinde esnek kontrol seçenekleri sunması açısından kritik bir yazılımdır.

Gazebo simülasyon ortamında ArduPilot'un entegrasyonu sayesinde, sürü drone sisteminin uçuş dinamikleri ve görev senaryoları test edilmiştir. Özellikle kalkış, rota takibi, görev paylaşımı ve acil durum senaryoları ArduPilot – SITL entegrasyonu kullanılarak modellenmiş, elde edilen veriler sistemin tasarımının doğrulanmasında kullanılmıştır. ArduPilot'un açık kaynak yapısı, Python tabanlı kontrol algoritmaları ve ROS ile uyumluluğu, bu tez çalışmasında sürü robotlarının simülasyonunda tercih edilmesinde temel etken olmuştur.

3.1.3 Yazılım Döngüsünde Simülasyon

Robotik sistemlerin geliştirilmesinde gerçek donanım testleri yapmak, zaman ve maliyet gibi bazı zorlukları beraberinde getirmektedir. Özellikle İHA gibi karmaşık platformlarda yazılım geliştirme sürecinde yapılan küçük hatalar, donanıma zarar verebilir veya ciddi güvenlik riskleri doğurabilir. Bu nedenle, fiziksel donanım gereksinimi olmadan yalnızca yazılım üzerinden uçuş testlerinin yapılabilmesi amacıyla SITL yöntemi geliştirilmiştir.

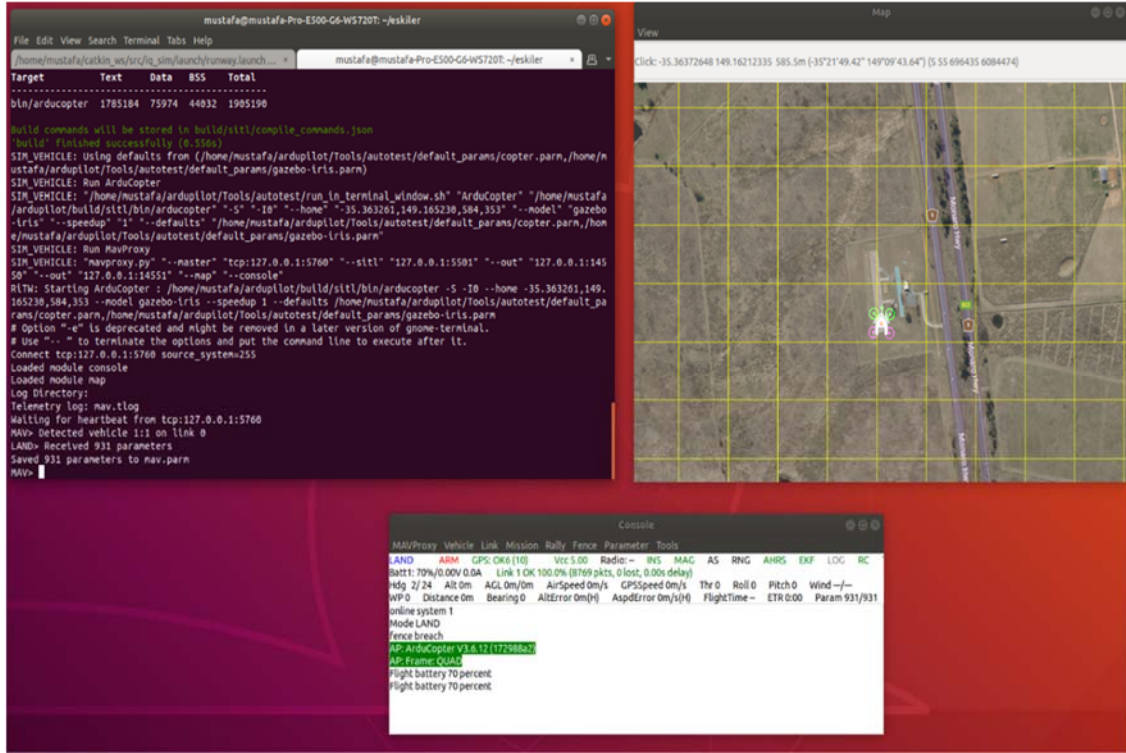
SITL, ArduPilot uçuş kontrol yazılımının doğrudan bilgisayar üzerinde çalıştırılmasına olanak tanıyan bir simülasyon yaklaşımıdır (İnt. Kyn. 5). Bu yöntem sayesinde fiziksel bir donanıma (uçuş kontrol kartına) ihtiyaç duymadan yazılım tabanlı bir uçuş kontrolcüsü çalıştırılabilir ve drone'ların kalkış, iniş, rota takibi, görev planlama ve acil durum senaryoları sanal ortamda test edilebilir. SITL, gerçek uçuş kontrol yazılımının doğrudan çalıştırdığı için simülasyonda elde edilen sonuçlar büyük oranda gerçek donanım ile uyumlu olmaktadır. SITL'in en önemli avantajlarından biri, risk ve maliyeti ortadan kaldırmasıdır. Donanım arızası, kaza veya düşme gibi olumsuzluklar yaşanmadan farklı senaryolar güvenli bir biçimde denenebilir. Ayrıca geliştirilen algoritmaların hızlı bir şekilde test edilmesi, hata ayıklama süreçlerinin kolaylaşması ve farklı parametrelerin kısa sürede denenebilmesi geliştirme süresini büyük ölçüde hızlandırmaktadır.

SITL, yalnızca tekli drone testlerinde değil aynı zamanda çoklu araç senaryolarında da kullanılabilir. Bu özellik, sürü robotik uygulamaları açısından kritik öneme sahiptir. Aynı simülasyon ortamında birden fazla drone'un senkronize olarak uçuşabilmesi, görev paylaşımı ve çarpışma önleme gibi durumların yazılım tabanlı olarak denemesine imkân tanımaktadır. Bu özellikler, SITL'i sürü robotik uygulamaları için vazgeçilmez bir araç hâline getirmektedir (Saavedra-Ruiz vd. 2021, Hiba vd. 2022).

Bu tez çalışmasında SITL, ArduPilot yazılımına entegre edilerek kullanılmıştır. Simülasyon, Ubuntu işletim sistemi üzerinde çalıştırılmış ve Gazebo ortamı ile birlikte değerlendirilmiştir. Drone'ların uçuş kontrolü, Python dili üzerinden yazılan programlar aracılığıyla gerçekleştirilmiş; bu programlar DroneKit kütüphanesi üzerinden MAVLink protokolü kullanılarak SITL ile haberleşmiştir. Böylece yazılım tarafından verilen

komutlar simülasyondaki drone'a aktarılmış, elde edilen telemetri verileri ise tekrar kontrol yazılımına geri beslenmiştir.

SITL'in çalıştırılması sonrasında elde edilen ekran çıktıları Şekil 3.3'te verilmiştir. Görselde, terminal penceresinde ArduPilot'un başlatıldığı, parametrelerin yüklendiği ve telemetri bağlantısının kurulduğu görülmektedir. Sağ taraftaki pencerede drone'un konumu harita üzerinde gösterilmekte, alt kısımda bulunan Console ekranında ise MAVProxy arayüzü üzerinden arm/disarm, GPS, batarya SOC değeri, hız ve irtifa gibi uçuş parametreleri takip edilebilmektedir.



Şekil 3.3 SITL simülasyon çıktıları.

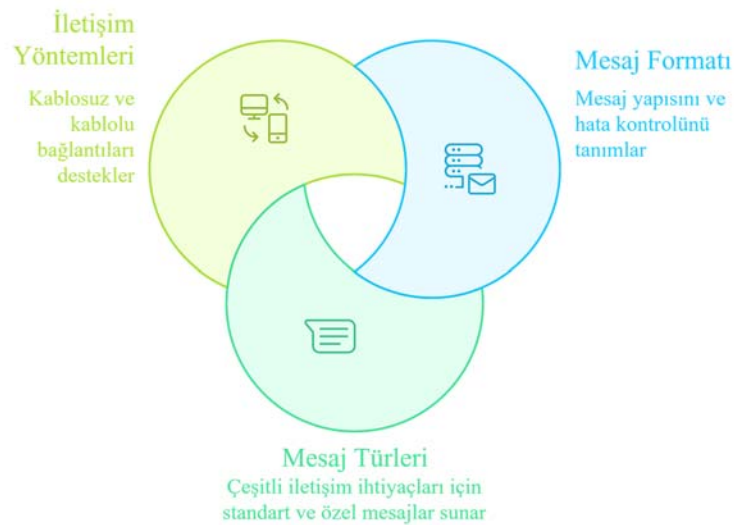
3.1.4 Mikro Hava Aracı İletişim Protokolü

Modern robotik sistemlerde, özellikle İHA ve İKA gibi otonom platformların geliştirilmesinde, araç ile kontrol sistemi arasında güvenilir ve farklı koşullarda uygulanabilir bir iletişim protokolüne ihtiyaç vardır. Bu bağlamda MAVLink, günümüzde en yaygın kullanılan haberleşme protokollerinden biri hâline gelmiştir. MAVLink, yalnızca tek bir araç türüne bağlı kalmadan farklı platformlarda

kullanılabilmesi, kullanıcıların kendi görev ihtiyaçlarına uygun yeni mesaj türleri tanımlayabilmesine olanak sağlaması ve çeşitli haberleşme yöntemleri (seri port, Wi-Fi, RF modem, LTE/5G) üzerinden çalışabilmesi sayesinde farklı uygulama alanlarına kolaylıkla entegre edilebilmesi önemli bir avantaj sunmaktadır. MAVLink, 2009 yılında Lorenz Meier tarafından geliştirilmiş, açık kaynaklı ve bağımsız bir protokoldür (İnt. Kyn. 6). Başlangıçta küçük İHA'lar için tasarlanan protokol, zamanla hem akademik araştırmalarda hem de endüstriyel uygulamalarda standart hâline gelmiştir. MAVLink, uçuş kontrol sistemi ile yer istasyonu arasında çift yönlü iletişimi sağlamakla kalmaz, aynı zamanda araç üzerindeki sensör verilerini toplama, komut gönderme ve telemetri paylaşımı gibi işlevleri de yerine getirmektedir.

MAVLink'in en önemli özelliği, düşük bant genişliği ve hafif mesaj yapısına sahip olmasıdır. Bu sayede mesaj başına yalnızca birkaç byte ek yük gerektirmektedir. Bu ek yük gönderilecek olan mesajın dışında başlık bilgileri, hata denetim verileri ve zaman bilgileri gibi ek verileri temsil etmektedir. Bu verilerin küçük boyutlarda olması sayesinde MAVLink gerçek zamanlı uygulamalarda yüksek performans sunar. Bununla birlikte, farklı araç türleri ve görev senaryoları için genişletilebilir bir yapıya sahiptir. Bu özellikler sayesinde sürü robotiği gibi çoklu robotların bulunduğu sistemlerde, hızlı ve güvenilir bir iletişim altyapısı oluşturulabilmektedir (Koubâa vd. 2019).

MAVLink protokolü, genel olarak üç temel bileşen üzerinden tanımlanabilir (Şekil 3.4):



Şekil 3.4 MAVLink protokolünün temel bileşenleri.

Mesaj Formatı: MAVLink mesajları belirli bir paket yapısı ile tanımlanır. Bu yapı; başlık, mesaj kimliği, yük verisi ve hata kontrol bitlerinden oluşur. Böylece hem mesajın doğru algılanması hem de iletim sırasında olası hataların tespit edilmesi sağlanır. Çizelge 3.2’de MAVLink mesajlarının paket formatı verilmiştir.

Çizelge 3.2 MAVLink mesajlarının paket formatı (İnt. Kyn. 6).

Alan	Açıklama	Veri boyutu (Bayt)
Başlangıç Baytı	Mesajın başladığını belirten sabit bayt değeri (0xFE)	1
Yük Uzunluğu	Mesaj içeriğinde taşınacak veri miktarı (0–255 arasında)	1
Paket Sıra Numarası	Veri sırasını takip etmek için verilen sayaç numarası	1
Sistem Kimliği	Mesajı gönderen aracın ID’si	1
Bileşen Kimliği	Mesajı gönderen bileşenin ID’si (örneğin GPS, IMU vb.)	1
Mesaj Kimliği	Mesaj türünü tanımlayan ID (örneğin, ATTITUDE vb.)	1
Yük	Asıl veri içeriği	0–255
Doğrulama	Verinin bozulup bozulmadığını kontrol eden hata denetimi değeri	2

Mesaj Türleri: Konum bilgisi, hız, batarya SOC değeri, sensör verileri, görev komutları ve hata mesajları gibi farklı amaçlara hizmet eden yüzlerce standart mesaj tipi bulunmaktadır. Ayrıca kullanıcılar özel görevler için yeni mesaj türleri de tanımlayabilir.

İletişim Yöntemleri: MAVLink mesajları kablosuz haberleşme modülleri (RF, Wi-Fi, LTE/5G) veya kablolu bağlantılar üzerinden iletilir. Çoklu bağlantı desteği sayesinde birden fazla yer istasyonu veya kontrol bilgisayarı aynı anda drone’lar ile iletişim kurabilirler. Sistem kimliği tanımlaması sayesinde MAVLink üzerinden 255 adet drone bağlantısı yapılabilir.

MAVLink’in öne çıkan teknik özellikleri şu şekilde özetlenebilir;

- Hafiflik: Mesaj başına yalnızca birkaç bayt ek yük gerektirir.
- Çapraz Platform Desteği: Windows, Linux, macOS gibi platformlarda çalışabilir.
- Düşük Gecikme: Bağlantı türüne göre gecikme oranı değişmekle birlikte yaklaşık olarak Seri (Uart/USB) bağlantıda 5 – 50 ms, Wi-Fi (UDP/TCP)’de 10 – 100 ms, RF modemlerde 50 – 200 ms, 4G/5G LTE’de 20 – 80 ms ve uydu

haberleşmelerinde 0.5 – 1.5 s’dir. Gecikmelerin böyle düşük olması da gerçek zamanlı uygulamalar için uygundur.

- Modülerlik: Farklı araç türleri ve sensörler için genişletilebilir.

Bu tez çalışmasında MAVLink, ArduPilot ve SITL ile Gazebo ortamında çalışan drone’lar arasında iletişim sağlamak için kullanılmıştır. Python tabanlı algoritmalar DroneKit kütüphanesi aracılığıyla MAVLink mesajları göndermekte, aynı zamanda telemetri verileri de yine MAVLink üzerinden alınmaktadır. Böylece sürü robotlarının görev paylaşımı, sensör verilerinin entegrasyonu ve nesne tespiti senaryoları güvenilir bir haberleşme altyapısı üzerinde gerçekleştirilmiştir.

3.1.5 DroneKit

DroneKit, İHA’ların kontrolü ve yönetimi için Python programlama diliyle geliştirilmiş açık kaynaklı bir yazılım kütüphanesidir. MAVLink protokolü üzerinden uçuş kontrol sistemleriyle doğrudan iletişim kurabilmektedir. Böylelikle telemetri verilerinin alınması, araç durum bilgilerinin izlenmesi, parametrelerin değiştirilmesi gibi işlemleri destekler. Ayrıca görev yükleme ve doğrudan hareket komutlarının iletilmesi de mümkündür.

DroneKit, ArduPilot gibi yaygın kullanılan otopilot sistemleriyle uyumlu çalışmaktadır (Pulungan vd. 2024). Bu uyumluluk, gerçek zamanlı kontrol algoritmalarını kolayca test edebilme ve otonom görev senaryoları geliştirebilme imkânı sunmaktadır. DroneKit hem yer kontrol istasyonlarında hem de Raspberry Pi veya NVIDIA Jetson gibi yardımcı bilgisayarlarda kullanılabilir. Böylece bilgisayarla görme, yol planlama, nesne tespiti ve 3D haritalama gibi ileri seviye uygulamalar uçuş kontrol sistemine entegre edilebilmektedir. DroneKit’in sağladığı imkânlardan bazıları şunlardır:

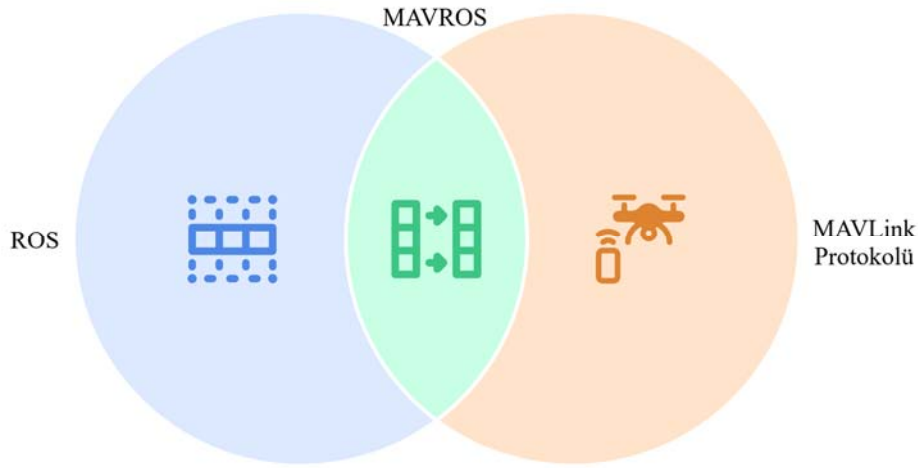
- Telemetri Verisi Erişimi: Konum, hız, batarya SOC değeri, GPS doğruluğu gibi kritik veriler anlık olarak alınabilir.
- Durum ve Parametre Yönetimi: Drone’un uçuş modu, arm/disarm durumunu veya uçuş parametrelerinin okunmasını ve değiştirilmesini mümkün kılar.

- Görev Planlama: Belirli irtifalara kalkış, nokta tabanlı rota takibi, belirlenen koordinatlarda görev icrası gerçekleştirilmesini sağlar.
- Doğrudan Kontrol: Belirli bir hız, yön veya yükseklik komutunun anlık olarak drone'a gönderilmesini sağlar.

Bu özellikleriyle DroneKit hem gerçek donanım üzerinde hem de SITL tabanlı simülasyonlarda çalışabilmekte olup yazılım geliştirme ve test süreçlerinde önemli bir avantaj sağlamaktadır. Bu tez kapsamında DroneKit, Gazebo simülasyon ortamında Python tabanlı kontrol yazılımlarının geliştirilmesi ve test edilmesi amacıyla kullanılmıştır. Geliştirilen programlar, ArduPilot sistemi ile MAVLink protokolü üzerinden iletişim kurmuştur. Böylece drone'ların kalkış, iniş, rota takibi ve görev paylaşımı gibi işlevleri başarıyla gerçekleştirilmiştir. DroneKit, sağladığı kompakt yapı ve programlama kolaylığı ile sürü robotik sistemlerin yazılım tabanlı kontrolünde önemli bir rol oynamıştır.

3.1.6 Robot İşletim Sistemi Tabanlı MAVLink

MAVROS, MAVLink protokolü ile ROS ekosistemi arasında bağlantı sağlayan bir ROS paketi/düğümüdür. Temel işlevi, ArduPilot'tan gelen MAVLink mesajlarını ROS'un kabul ettiği veri yapılarına dönüştürmektir (Jha vd. 2024). Aynı şekilde ROS'tan gelen mesajların da MAVLink formatına çevrilerek uçuş kontrolcüsüne iletilmesini sağlar. Böylece ROS tabanlı bir uygulamada geliştirilen algılama, karar verme ve görev yönetimi modülleri, MAVROS üzerinden bir araç (drone, İKA vb.) ile haberleşebilir (Virginio ve Adiprawita 2021). MAVROS, uçuş kontrol kartına seri, Kullanıcı Datagram Protokolü (User Datagram Protocol – UDP) veya İletim Kontrol Protokolü (Transmission Control Protocol – TCP) üzerinden bağlantı yapabilmektedir. SITL üzerinden yapılan simülasyonlarda yaygın olarak SITL'in açtığı bir UDP uç bağlantı noktası kullanılmaktadır. Aynı araçla eşzamanlı olarak yer kontrol istasyonu da başka bir uç bağlantı noktasından iletişim kurabilmektedir. Şekil 3.5'te, MAVROS'un ROS ve MAVLink arasında köprü görevi görerek iki farklı ekosistem arasındaki iletişimi nasıl sağladığı şematik olarak gösterilmektedir.



Şekil 3.5 MAVROS protokolünün temsili gösterimi.

3.1.7 Robot İşletim Sistemi

ROS, robotik yazılım geliştirme sürecini kolaylaştırmak amacıyla tasarlanmış açık kaynaklı bir yazılım platformudur. ROS, ilk olarak 2007 yılında Stanford Üniversitesi Yapay Zekâ Laboratuvarı'nda bir araştırma projesi olarak ortaya çıkmış, 2013 yılından itibaren Open Source Robotics Foundation (OSRF) tarafından desteklenmeye başlanmıştır (Quigley vd. 2015). ROS'un ilk sürümü olan ROS 1, kısa sürede akademik dünyada popülerlik kazanmış ve araştırmacıların standart bir robotik yazılım çerçevesi etrafında birleşmesini sağlamıştır. Ancak endüstriyel robotların gereksinimlerini karşılamak üzere 2014 yılında ROS 2 geliştirilmiş ve gerçek zamanlılık, güvenilirlik ve gömülü sistem desteği gibi yeni özelliklerle geliştirilmiştir (Maruyama vd. 2016).

ROS, modüler bir mimariye sahip olup farklı işlevleri yerine getiren birden fazla yazılım bileşeninden oluşmaktadır. Bu mimaride temel bileşenler şunlardır (Chen vd. 2022):

- Düğüm (Nodes): ROS tabanlı her işlem bir düğüm olarak tanımlanır. Sensör verilerinin işlenmesi, kontrol algoritmalarının yürütülmesi veya kullanıcı arayüzlerinin çalıştırılması gibi belirli görevler düğümler tarafından gerçekleştirilir.
- Mesajlaşma (Messaging): Düğümler arasında veri alışverişi “yayıncı/izleyici” modeli ile gerçekleştirilir. Düğümler belirli konularda veri yayınlar veya bu konuları dinleyerek verileri alır.

- Hizmetler (Services): Anlık istek – cevap tabanlı iletişim için kullanılır. Bir düğüm belirli bir hizmeti sunarken başka bir düğüm bu hizmeti çağırabilir.
- Eylemler (Actions): Daha uzun süren görevler için tasarlanmıştır. Eylemler, işlemin ilerleyişi hakkında geri bildirim sağlar ve gerektiğinde işlemin iptal edilmesine olanak tanır.

ROS, sunduğu standart arayüzler ve hazır sürücüler sayesinde farklı donanımların yazılıma entegre edilmesini kolaylaştırmaktadır. GPS, IMU, LiDAR ve kamera gibi sensörler için hazır paketlerin bulunması entegrasyon sürecini kolaylaştırmakta ve geliştirme hızını artırmaktadır. Bunun yanında veri görselleştirme, simülasyon ve hata ayıklama için kapsamlı araçlar sağlamaktadır. Yazılım bileşenlerinin paketler halinde düzenlenebilmesi, yeniden kullanılabilirliği artırmakta ve modüler geliştirmeyi desteklemektedir. ROS'un kullanım alanlarından bazıları aşağıda verilmiştir.

- Mobil robotlarda: Otonom navigasyon, haritalandırma, engel algılama ve yol planlama uygulamalarında,
- İnsansı robotlarda: Hareket planlama, nesne tanıma, çevre algısı ve insan–robot etkileşiminde,
- Drone'larda: Uçuş kontrol kartı ile iletişim, görev tanımlama, sensör verilerinin işlenmesi ve yüksek seviyeli görev yönetiminde,
- Endüstride: Üretim hatlarında robotların entegrasyonu, çoklu robot iş birliği ve sensör verilerinin merkezi olarak işlenmesinde kullanımları bulunmaktadır.

Bu tez kapsamında ROS, sürü drone sisteminde veri akışının düzenlenmesi, görevlerin yönetilmesi ve sensör verilerinin işlenmesi amacıyla kullanılmıştır. MAVLink protokolünden elde edilen telemetri verileri, MAVROS aracılığıyla ROS ekosistemine aktarılmıştır. Bu veriler üzerine geliştirilen algoritmalar sayesinde drone'ların kontrolü ve sürü koordinasyonu sağlanmıştır. ROS'un sağladığı modüler yapı ve geniş kütüphane desteği, simülasyon ve yazılım geliştirme süreçlerinin güvenli, hızlı ve tekrarlanabilir bir şekilde yürütülmesine imkân tanımıştır.

3.1.8 Kullanılan Sensörler

Sürü drone sistemlerinin hem gerçek hem de simülasyon ortamlarında güvenilir bir şekilde çalışabilmesi, farklı sensörlerden elde edilen verilerin doğru biçimde işlenmesine bağlıdır. Bu sensörler, aracın konumunun belirlenmesinden çevrenin algılanmasına, uçuş kararlılığının sağlanmasından nesne tespitine kadar birçok kritik görevi üstlenmektedir. Bu tez çalışmasında kullanılan başlıca sensörler; GPS, IMU, LiDAR ve kamera olup, her biri sistemin farklı bir yönünü destekleyerek sürü robotlarının daha etkin ve güvenli şekilde görevleri yerine getirmesine katkı sağlamaktadır.

GPS, uydulardan alınan sinyaller aracılığıyla konum, hız ve zaman bilgisi sağlayan küresel bir konumlama sistemidir. Drone'ların en temel sensörlerden biridir çünkü aracın küresel koordinatlarda konumunu belirler (Dlamini ve Ouma 2025). GPS verileri, drone'un kalkış noktasının belirlenmesi, rota takibi, belirlenen koordinatlara yönelme ve acil durumlarda RTL gibi işlevlerde kritik rol oynar. Simülasyonda GPS, drone'un dünya üzerindeki konumunu sağlamak için kullanılır. Gerçek uçuşlarda olduğu gibi, yazılım tabanlı görev planlama ve rota takibi işlemlerinde GPS verileri olmadan drone'un güvenli ve doğru bir şekilde hareket etmesi mümkün değildir.

IMU, ivmeölçer, jiroskop ve çoğu zaman manyetometre bileşenlerinden oluşan bir sensör birimidir (Xu ve Zhang 2021). Drone'larda aracın üç eksenindeki ivmesini ve yönelimini (roll, pitch, yaw) ölçer. Bu sayede drone'un stabil uçuşu sağlanır ve kontrol algoritmaları doğru verilerle çalışır. IMU verileri, uçuş kontrol kartının drone'un hareketlerini dengelemesine, sabit bir irtifada veya belirlenen yönde kalmasına olanak tanır. Simülasyonda, özellikle Gazebo ortamında, IMU çıktıları drone'un hareketlerinin fiziksel doğruluğunu artırır. GPS'in olmadığı durumlarda kısa süreli navigasyon IMU üzerinden sağlanabilir.

LiDAR, lazer ışınları kullanarak çevredeki nesnelere olan mesafeyi ölçen bir algılama sensörüdür. Gönderilen lazer sinyali nesneye çarpar ve geri döner. Geri dönüş süresi ölçülerek mesafe hesaplanır. Drone uygulamalarında LiDAR, engel algılama, çarpışma önleme, haritalandırma ve yükseklik ölçümü için kullanılır (Xu ve Zhang 2021). Bu tezde

LiDAR sensörü, drone'un önündeki engellerin algılanması ve sürü senaryolarında çarpışma ihtimalinin azaltılması amacıyla kullanılmıştır. Ayrıca LiDAR, GPS ve IMU ile birlikte çalışarak daha güvenli ve çevreye duyarlı bir uçuş deneyimi sağlar.

Kamera, drone üzerinde görsel veri sağlayan en kritik sensörlerden biridir. Tek başına ya da görme algoritmaları ile birlikte kullanılabilir. Kamera görüntüleri, çevrenin algılanmasında, nesne tespitinde, hedef takibinde ve haritalandırma çalışmalarında kullanılır. Kamera, sürü drone sistemine yalnızca konum bilgisi değil aynı zamanda çevresel farkındalık da kazandırmaktadır. Görüntü işleme tabanlı görevlerin uygulanabilirliği (örneğin hedef belirleme, ilaçlama yapılacak alanın algılanması) doğrudan kamera sensörüne bağlıdır. Bu tezde kamera sensöründen alınan görüntüler YOLO tabanlı nesne tespit algoritması ile işlenmiş, tarımsal senaryolarda hedef nesnelerin tanımlanması sağlanmıştır.

3.1.9 YOLO'nun Entegrasyonu

Bu çalışmada YOLO algoritması, Gazebo simülasyon ortamındaki kamera sensöründen elde edilen görüntülerin işlenmesi amacıyla kullanılmıştır. Bu entegrasyon için DarkNet_ROS paketi kullanılmış, böylece Gazebo'daki sanal kameradan alınan görüntüler ROS üzerinden YOLO'ya aktarılmıştır. ROS içinde bir düğüm olarak çalışan DarkNet_ROS, her görüntü karesini işleyerek tespit edilen nesnelerin konumlarını, sınıf etiketlerini ve güven skorlarını yayınlamaktadır. Bu veriler, diğer ROS düğümleri tarafından abone olunabilen standart ROS konuları üzerinden paylaşılmakta ve böylelikle sürü robotlarının karar mekanizmalarına girdi sağlamaktadır.

YOLO, yalnızca görüntü işleme için değil aynı zamanda sürü koordinasyonunu destekleyen bir bilgi kaynağı olarak da kullanılmıştır. Örneğin belirlenen bir nesnenin konumu veya belirli bir alandaki yoğunluk bilgisi, sürüdeki drone'lara görev paylaşırma veya rota güncelleme şeklinde yansıtılmıştır. Bu entegrasyon sayesinde, simülasyon ortamındaki drone'lar yalnızca konum ve irtifa bilgisine değil aynı zamanda çevresel farkındalığa da sahip olmuştur. Böylece tarımsal senaryolarda nesneye dayalı görevlerin

(örneğin belirli bir ürünün algılanması veya ilaçlama yapılacak alanın tespiti) daha etkin şekilde planlanabilmiştir.

3.1.10 Kullanılan Optimizasyon Yöntemleri

Hedef noktaları belirlendikten sonra, bu noktaların hangi sırayla ziyaret edileceği toplam görev süresi ve enerji verimliliği açısından kritik öneme sahiptir. Bu nedenle rota planlama problemi, literatürde Gezgin Satıcı Problemi (Travelling Salesman Problem – TSP) olarak ele alınmakta ve çözüm için farklı optimizasyon yöntemleri kullanılmaktadır. Bu çalışmada rota optimizasyonu için En Yakın Komşu (Nearest Neighbor – NN) ve ACO yöntemleri uygulanmıştır.

3.1.10.1 En Yakın Komşu Algoritması

NN algoritması, özellikle rotalama problemleri, örüntü tanıma, sınıflandırma, robot navigasyonu ve makine öğrenmesi gibi birçok alanda kullanılan temel bir yöntemdir (Martínez-Clark vd. 2023). Amaç belirtilen veri noktasına en yakın diğer veri noktasını bulmaktır. Basit ve sezgisel bir algoritma olduğu için hesaplama maliyeti düşüktür. Yerel noktaların belirlenmesinde iyi bir performans sergilerken küresel ölçekte ve büyük veri kümelerinde etkinliği sınırlıdır. Bir nokta kümesi $X = \{x_1, x_2, x_3, \dots, x_n\} \subset R^d$ içerisinde herhangi bir sorgu noktası $q \in R^d$ için, en yakın komşusu $x^* \in X$, denklem 3.1’de gösterildiği gibi tanımlanmaktadır.

$$x^* = \arg \min_{x \in X} d(q, x) \quad (3.1)$$

Burada;

$d = (q, x)$ mesafe fonksiyonunu ifade etmektedir.

Çalışmada öklidyen mesafe fonksiyonu (denklem 3.2) kullanılmıştır.

$$d(q, x) = \|q - x\|_2 = \sqrt{\sum_{i=1}^d (q_i - x_i)^2} \quad (3.2)$$

Rota optimizasyonu algoritmalarında NN kendisine en yakın olan noktaları bularak aç gözlü bir yaklaşım sergilemektedir. Başlangıç noktası seçildikten sonra kendisine en yakın olan noktadan devam edecek şekilde bütün noktaları belirlemektedir (Sales vd. 2023).

3.1.10.2 Karınca Kolonisi Optimizasyonu

ACO, ilk olarak Marco Dorigo tarafından 1992 yılında geliştirilmiş olan, doğadan ilham alan bir meta-sezgisel optimizasyon yöntemidir (Neto ve Godinho Filho 2013). Doğada karıncalar yiyecekleri bulmak için geçtikleri yollara feromon bırakırlar. Kullanılan kısa yollardaki feromon yoğunluğu daha fazla olduğundan diğer karıncalar da bu kısa yolu tercih ederler. ACO, bu doğal süreçten esinlenilerek geliştirilmiştir. Özellikle TSP’de, rota planlama, çizelgeleme, ağ tasarımı gibi ayrık kombinatoriyel problemlerde yaygın olarak kullanılır (Heng vd. 2024). Bir nokta kümesi $V = \{v_1, v_2, v_3, \dots, v_n\}$ ve noktalar arası mesafeleri tanımlayan mesafe matrisi $D = V \times V \rightarrow R_{\geq 0}$ ile amaç, her bir kenar üzerindeki feromon değeri τ_{ij} ve sezgisel bilgi (genellikle mesafe) $n_{ij} = \frac{1}{d_{ij}}$ ile tanımlanır. Bir karıncanın i noktasından j noktasına gitme kararı denklem 3.3’te verilmiştir.

$$P_{ij}^k(t) = \frac{[\tau_{ij}(t)]^\alpha \cdot [n_{ij}]^\beta}{\sum_{l \in N_i^k} [\tau_{il}(t)]^\alpha \cdot [n_{il}]^\beta} \quad (3.3)$$

Burada;

$\tau_{ij}(t)$: t anındaki feromon yoğunluğu,

$n_{ij} = 1/d_{ij}$: sezgisel değeri,

α : feromon etkisi katsayısını,

β : sezgisel bilginin etkisini,

N_i^k : k-karıncanın henüz gitmediği noktaları belirtmektedir.

Tur tamamlandıktan sonra feromonların güncellenme denklemi ise denklem 3.4’te verilmiştir.

$$\tau_{ij}(t + 1) = (1 - \rho)\tau_{ij}(t) + \sum_{k=1}^m \Delta\tau_{ij}^k(t) \quad (3.4)$$

Burada;

$\rho \in (0,1)$: buharlaşma katsayısını,

$\Delta\tau_{ij}^k(t)$: k-karıncaların etkisini göstermektedir.

k-karıncaların etkisinin formülü denklem 3.5'te verilmiştir.

$$\Delta\tau_{ij}^k(t) = \begin{cases} Q/L_k, & (i,j) \text{ karıncaların rotasında} \\ 0, & (i,j) \text{ karıncaların rotasında değil} \end{cases} \quad (3.5)$$

Burada;

Q : Sabit katsayıyı,

L_k : k. karıncanın toplam tur uzunluğunu ifade etmektedir.

ACO'nun mobil robotlarda yol planlama, drone görev rotaları, üretim çizelgeleme, ağ tasarımı ve yönlendirme gibi çeşitli alanlarda yaygın olarak kullanılmaktadır (Heng vd. 2024). ACO ile yapılacak olan bir optimizasyonun paralelleştirilebilir yapıya sahip olması, adaptif, öğrenilebilir ve karmaşık verilerde başarılı olması gibi çeşitli avantajları vardır. Bunun yanı sıra parametre ayarlarının hassas olması, yerel minimum riskinin oluşması gibi çeşitli dezavantajları da bulunmaktadır.

3.2 Yapay Zekâ Destekli Sürü Robot Sisteminin Mimarisi

Bu bölümde yapay zekâ destekli sürü robot sisteminin mimarisi ayrıntılı olarak sunulmaktadır. Sistem, Gazebo simülasyon ortamı, ROS tabanlı mesajlaşma altyapısı ve görev planlama modülleri ile bütünleşik bir yapıda tasarlanmıştır. Mimari, sensör verilerinin işlenmesi, nesne tespiti ve görev dağılımı süreçlerini kapsayan modüler bir yazılım – haberleşme entegrasyonu temelinde oluşturulmuştur.

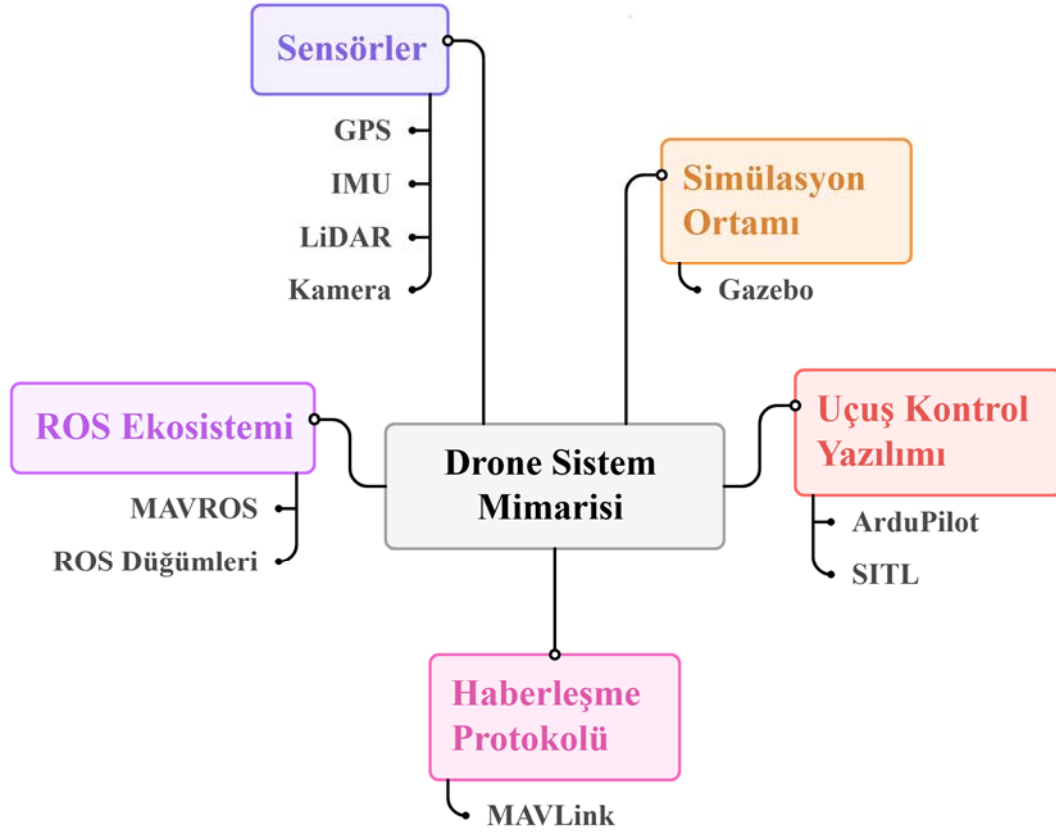
3.2.1 Genel Mimari

Bu tez çalışmasında Gazebo simülasyon ortamında tasarlanan sürü drone sistemi, tarımsal uygulamalarda kullanılmak üzere üç drone'dan oluşmaktadır. Sistemde bir adet kâşif drone'u ve iki adet ilaçlama drone'u görev yapmaktadır. Kâşif drone'u, tarım arazisinde belirlenen koordinat bölgesi içerisinde uçuş gerçekleştirerek kamera sensörü aracılığıyla görüntü toplamaktadır. Bu görüntüler anlık olarak YOLO tabanlı nesne tespit algoritması ile işlenmektedir. Oluşturulan domates bitkisi veri setiyle YOLO modeli eğitilmiş ve uçuş sırasında domates bitkilerinin tespiti sağlanmıştır. Tespit edilen her bitkinin GPS konum bilgisi alınarak hedef noktalar kümesine kaydedilmektedir. Görev yönetim modülü, bu hedef noktaları ilaçlama drone'larına aktarmakta ve ilgili drone'ların ilaçlama görevini yerine getirmesini sağlamaktadır. Böylece sistem, tarımsal alanlarda hem hızlı hem de etkin görev paylaşımı yaparak tekil drone kullanımına kıyasla daha verimli bir çözüm sunmaktadır.

Sistemin mimarisi yalnızca görev akışıyla sınırlı değildir; aynı zamanda bu süreci destekleyen yazılım ve donanım bileşenlerinden oluşan bütünleşik bir altyapıya dayanmaktadır. Şekil 3.6'da sistem mimarisinin görseli verilmiştir. Simülasyon ortamı Gazebo üzerinde oluşturulmuş; drone'ların uçuş kontrolü ArduPilot yazılımı ve SITL özelliği kullanılarak sağlanmıştır. Drone'lar ile kontrol yazılımları arasındaki haberleşme MAVLink protokolü üzerinden yürütülmüş, bu veriler MAVROS aracılığıyla ROS ekosistemine aktarılmıştır. ROS, görev planlama, sensör entegrasyonu ve algoritmaların yönetildiği üst katman olarak kullanılmıştır.

Drone'larda kullanılan sensörler farklı işlevleri yerine getirmektedir: GPS konum belirlemede, IMU uçuş stabilitesinde, LiDAR engel algılamada, kamera ise nesne tespitinde kritik rol üstlenmiştir. Tüm bu sensör verileri ROS düğümleri aracılığıyla işlenmiş ve karar verme modüllerine aktarılmıştır. Kamera görüntülerinden elde edilen YOLO çıktıları koordinat bilgisine dönüştürülmüş, GPS verileri ile eşleştirilmiş ve ilaçlama drone'larına görev tanımları olarak gönderilmiştir. Bu mimari yapı sayesinde, kâşif drone'u tarafından yapılan gözlemler gerçek zamanlı olarak sürü içindeki diğer

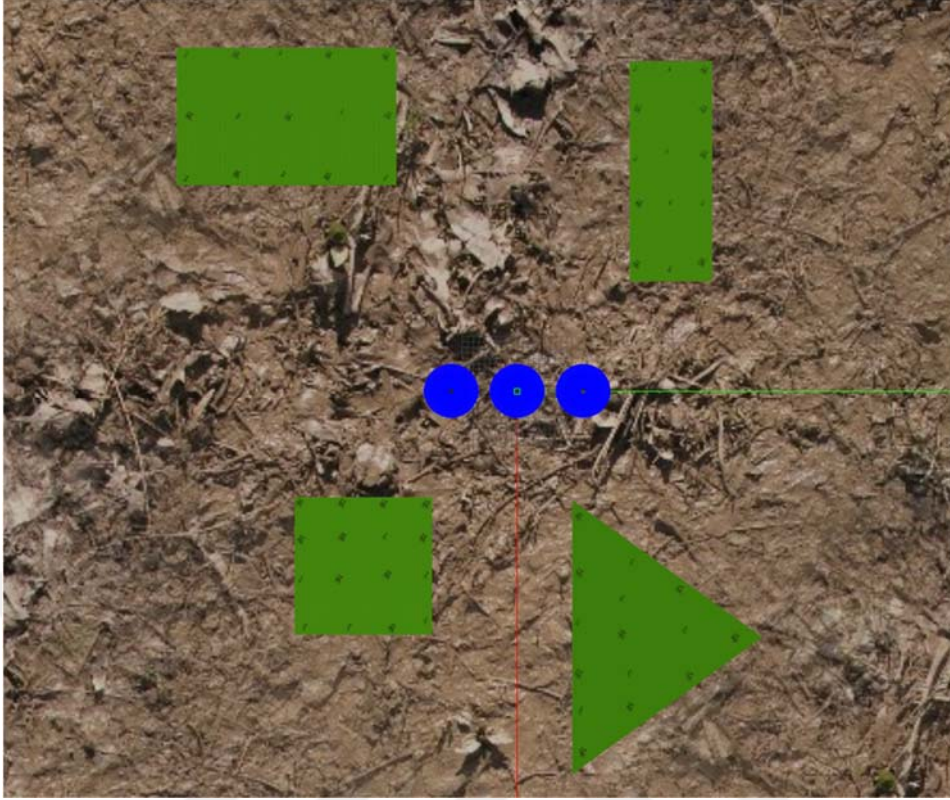
araçlarla paylaşılabilmiş; görev dağılımı, veri akışı ve kontrol süreçleri tek bir bütünleşik çerçevede yönetilmiştir.



Şekil 3.6 Sürü drone sistem mimarisi ve bileşenleri.

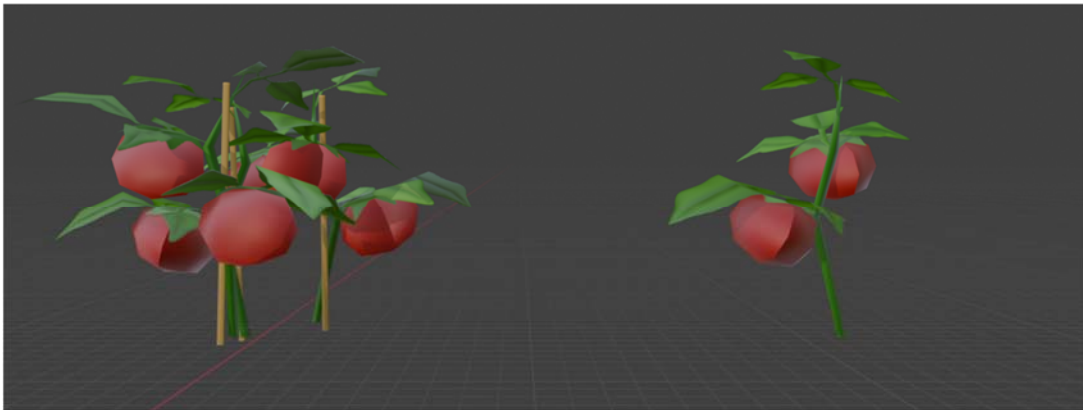
3.2.2 Gazebo Simülasyon Ortamı

Gazebo simülasyon ortamında 2000 m x 2000 m boyutlarında bir zemin oluşturulmuştur. Zemin üzerine toprak modeli eklenerek tarım arazisi görünümü sağlanmıştır. Tasarlanan dünya içerisinde dört farklı tarım arazisi yer almakta olup bu araziler boyut ve şekil bakımından birbirinden farklı olacak şekilde modellenmiştir. Merkez noktada drone'lar konumlandırılmış, araziler ise bu merkez etrafına (Şekil 3.7) yerleştirilmiştir.



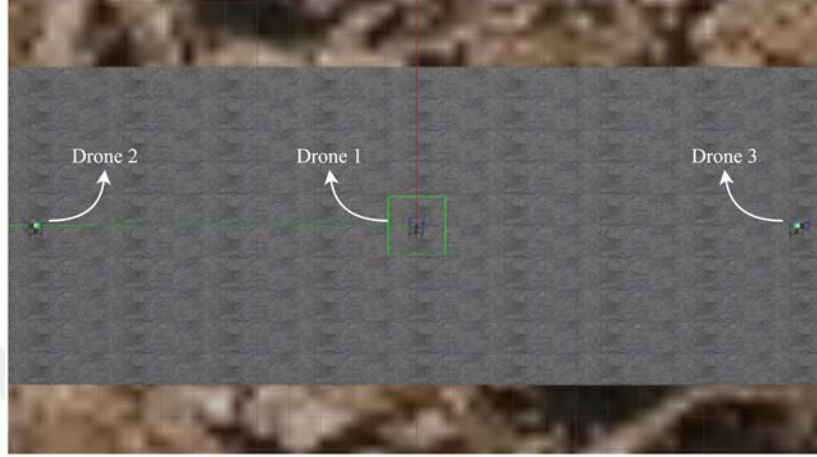
Şekil 3.7 Tasarlanan tarım arazilerinin üstten görünümü.

Gazebo ortamına eklenecek üç boyutlu modellerin “.dae” veya “.obj” formatında olması gerekmektedir. Çalışmada kullanılan 3D domates bitkisi modeli, “Sketchfab” 3D modelleme ve animasyon platformundan elde edilmiş (İnt. Kyn. 7) ve Gazebo ile uyumlu formata dönüştürülmüştür. Bu işlemlerin ardından domates modeli iki farklı boyutta olacak şekilde simülasyon ortamına eklenmiştir. Şekil 3.8’de domates bitkisi modelinin görünümü gösterilmiştir.



Şekil 3.8 Gazebo’da kullanılan domates bitki modeli.

Tasarımı gerçekleştirilen dünyada 1 adet kâşif drone'u ve 2 adet ilaçlama drone'u kullanılmaktadır. Drone'lar bütün arazilere yaklaşık eşit olacak şekilde merkezi bir alanda konumlandırılmışlardır. Şekil 3.9'da drone'ların simülasyon ortamındaki konumlarının üstten görünümü verilmiştir.



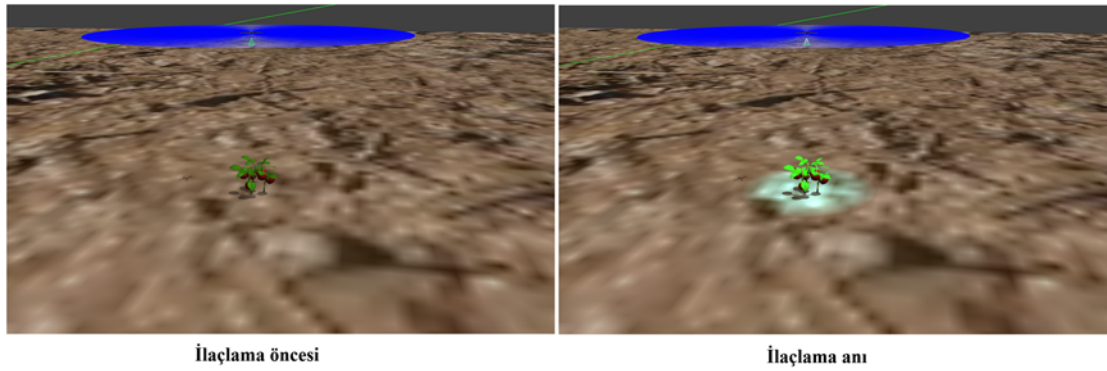
Şekil 3.9 Gazebo'da drone konumlarının üstten görünümü.

Kâşif drone'una kamera entegrasyonu yapılmıştır. Kamera, YOLO tabanlı nesne tespit algoritması için gerekli görüntüleri sağlamaktadır. Kullanılan kamera 720x640 çözünürlüğe ve 80° görüş açısına sahiptir. Kamera, drone üzerine bir gimbal aracılığıyla monte edilmiş olup gimbal sayesinde farklı açılarda (45°, 60°, 75° ve 90°) görüntü alınabilmektedir. Kâşif drone'unun 5 m irtifada iken 90°'lik açıda kameradan elde edilen görüntü Şekil 3.10'da verilmiştir.



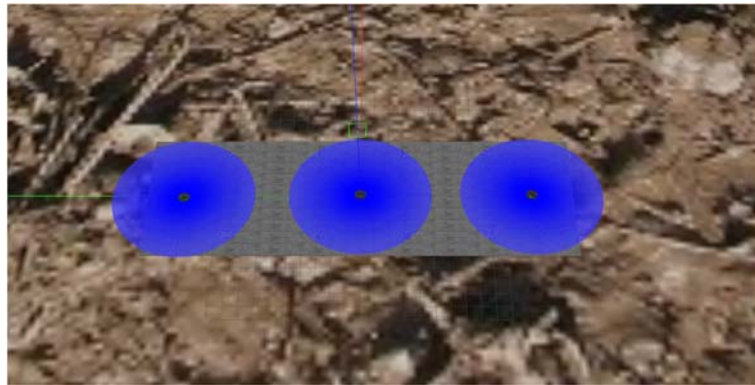
Şekil 3.10 Kâşif drone üzerinde bulunan kameranın 90° açıdaki görüntüsü.

İlaçlama drone'larının görevlerini görsel olarak takip edebilmek için üzerlerine ışık göstergeleri eklenmiştir. Drone'lar belirlenen hedef noktaya ulaştığında, ışık yanarak ilaçlama işleminin başladığını ve ardından söndürülerek işlemin tamamlandığını göstermektedir. Bu yöntem, sistemin çalışma durumunu simülasyon üzerinden gözlemlemeyi sağlamaktadır. Gerçek zamanlı yapılacak olan bir çalışmada ışık göstergeleri yerine ilaç püskürtme mekanizmaları bulunacaktır. Böyle bir durumda ışığın açılıp kapatılması ile aynı mantıkta ilaç püskürtme sistemi açılıp kapatılacaktır. Şekil 3.11'de ilaçlama yapan ve yapmayan bir drone'un görselleri verilmiştir.



Şekil 3.11 İlaçlama drone'unun ilaçlama öncesi ve ilaçlama anındaki görüntüleri.

Oluşturulan sistemde drone'ların engellerle veya birbirleriyle çarpışmasını önlemek için her bir drone üzerine LiDAR sensörü eklenmiştir. LiDAR, 5 m yarıçapında tarama yaparak çevredeki engelleri algılamakta ve geliştirilen engelden kaçınma algoritması sayesinde drone'ların güvenli hareket etmesini sağlamaktadır. Şekil 3.12'de LiDAR lazer taramalarının üstten görünümü verilmiştir.



Şekil 3.12 LiDAR lazer görüntüleri aktif olan drone'ların üstten görünümü.

Her drone’da GPS sensörleri bulunmakta olup bu sayede hem yönelim ve hareketler doğru şekilde gerçekleştirilebilmekte hem de keşif sırasında tespit edilen bitkilerin konum bilgileri kaydedilmektedir. Elde edilen koordinatlar daha sonra ilaçlama drone’larına aktarılmakta ve bu drone’ların ilgili noktalara giderek ilaçlama yapmaları sağlanmaktadır.

Gazebo tabanlı simülasyon ortamında sistem performansının değerlendirilmesi için çeşitli fiziksel ve çevresel parametreler bulunmaktadır. Bu parametreler ve değer aralıkları Çizelge 3.3’te verilmiştir.

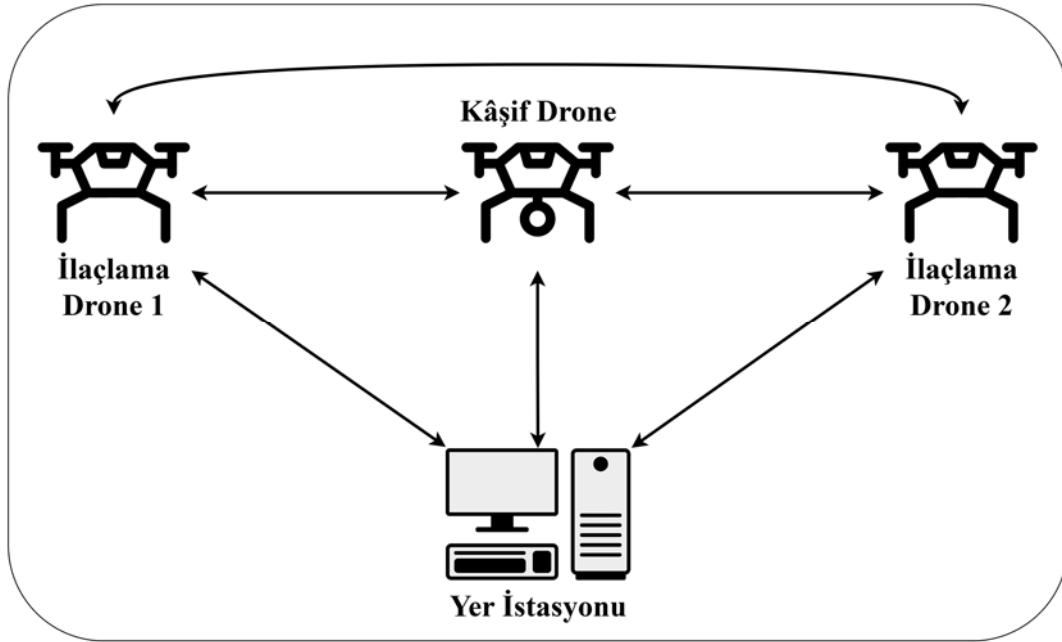


Çizelge 3.3 Gazebo simülasyon ortamı parametreleri.

Parametre Kategorisi	Parametre Adı	Değer / Özellik	Açıklama
Fiziksel ortam	Yerçekimi	9.8 m/s ²	Simülasyonda Dünya koşullarına uygun standart yerçekimi değeri kullanılmıştır.
	Rüzgâr	Aktif değil	Simülasyon ortamında rüzgâr etkisi devre dışı bırakılmış, sabit atmosfer koşulları sağlanmıştır.
	Hava yoğunluğu	1.204 kg/m ³	Gazebo'nun varsayılan atmosfer modeli kullanılmıştır.
Aydınlatma ve görsel ortam	Işık tipi	DirectionalLight	Güneş benzeri yönlü bir ışık kaynağı kullanılarak doğal aydınlatma koşulları oluşturulmuştur.
	Işık rengi	RGB (1, 1, 1)	Beyaz ışık tercih edilerek doğal gün ışığına yakın bir aydınlatma sağlanmıştır.
	Işık yoğunluğu	0.8	Orta parlaklıkta aydınlatma seviyesi belirlenmiş ve sensörler için dengeli görünürlük sağlanmıştır.
	Güneş yönü	(0.5, 0.5, -1.0)	Güneş vektörü doğu–güney yönüne ayarlanarak sahnenin doğal gölgelendirmesi sağlanmıştır.
	Sis	Aktif değil	Görüş mesafesi sabit tutulmuş, atmosferik sis veya bulanıklık etkisi uygulanmamıştır.
Sensör parametreleri	LiDAR menzili	5 m	Drone'un yakın çevresini algılaması için maksimum 5 metre menzil değeri tanımlanmıştır.
	LiDAR yatay çözünürlük	0.4°	Orta seviye tarama hassasiyeti sağlanarak işlem yükü ve doğruluk dengelenmiştir.
	LiDAR güncelleme frekansı	10 Hz	LiDAR sensöründen saniyede 10 ölçüm alınarak gerçek zamanlı engel tespiti yapılmaktadır.
	Kamera çözünürlüğü	640×480 piksel	Görsel algı modülünde kullanılan kameranın çözünürlüğü nesne tespiti performansına uygun seçilmiştir.
	Kamera formatı	RGB	Görüntüler YOLO tabanlı nesne tespit algoritması ile uyumlu renk formatında elde edilmiştir.
	IMU sensörü	Aktif	Drone'un konum ve yönelim stabilitesini sağlamak üzere atalet ölçüm birimi aktif olarak kullanılmıştır.
	GPS sensörü	Aktif	Drone'un konum verilerinin belirlenmesinde ve rota takibinde GPS sensörü kullanılmıştır.
Arazi ve çevresel etkenler	Zemin tipi	Düz	Tarımsal alan senaryosunu temsil eden düz yüzey modeli kullanılmıştır.
	Sürtünme katsayısı	1.0	Drone iniş–kalkış davranışlarında gerçekçi temas kuvveti simülasyonu için varsayılan değer kullanılmıştır.
Sistemsel parametreler	Simülasyon zamanı	Gerçek zamanlı güncelleme oranı: 1000	Simülasyon gerçek zaman ölçeğinde (1× hız) çalışacak şekilde yapılandırılmıştır.
	Fizik motoru	ODE	Gazebo'nun varsayılan fizik motoru kullanılmış, dinamik etkileşimler bu motorla çözülmüştür.
	Zaman adım aralığı	0.001 s	Simülasyon çözünürlüğü artırılarak yüksek doğrulukta zaman adımı sağlanmıştır.

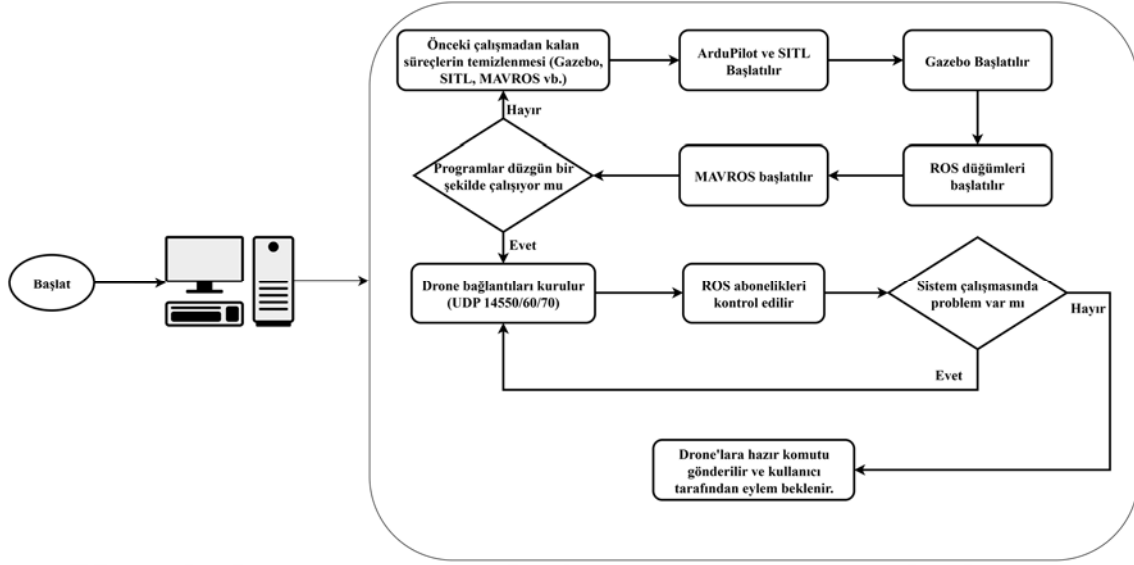
3.2.3 Tasarlanan Sistemin Yazılım ve Kod Mimarisi

Bu tez kapsamında geliştirilen sürü drone sistemi, yalnızca donanımsal bileşenlerden ibaret olmayıp aynı zamanda bu bileşenleri uyum içerisinde çalıştıran yazılım altyapısına sahiptir. Sistem, modüler ve ölçeklenebilir bir yapıda tasarlanmıştır; görev yönetimi, sensör verilerinin işlenmesi, uçuş kontrolü, nesne tespiti ve optimizasyon algoritmaları ayrı modüller hâlinde organize edilmiştir. Böylece hem hata ayıklama hem de yeni işlevlerin sisteme eklenmesi kolaylaştırılmıştır. Şekil 3.13'te, sistemin yazılım mimarisinin genel akış diyagramı sunulmuştur.



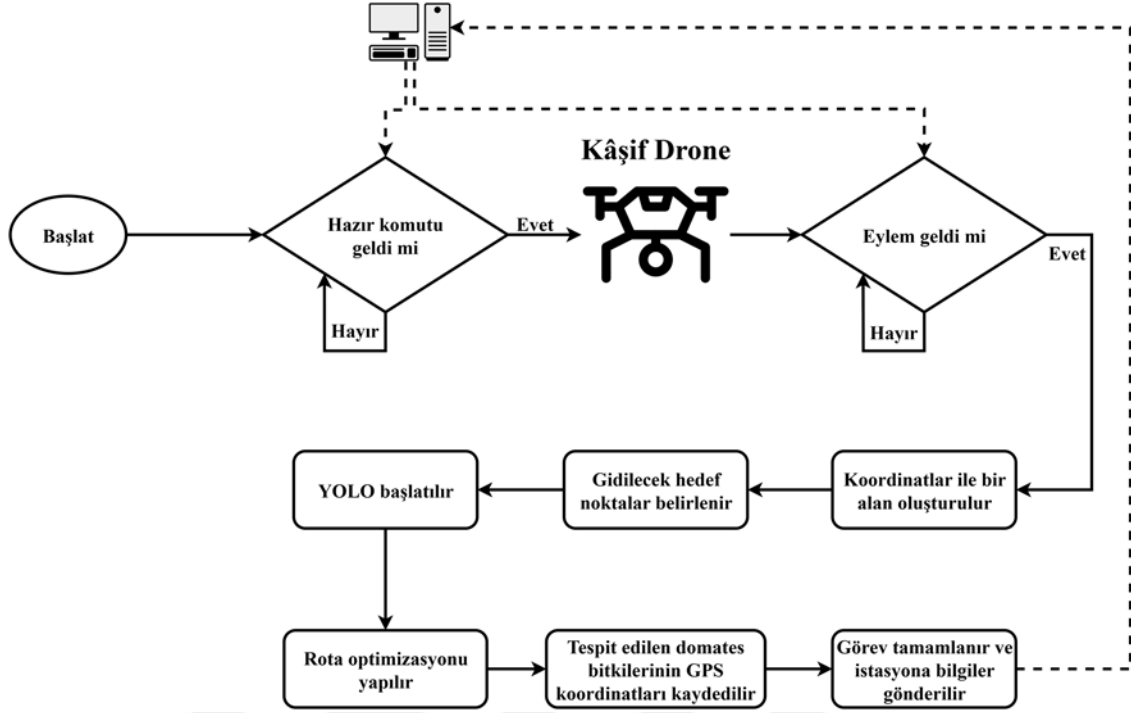
Şekil 3.13 Zirai ilaçlamada kullanılmak üzere tasarlanan sürü drone sisteminin akış diyagramı.

Tasarlanan sistemde, yer istasyonu ile tüm drone'lar arasında çift yönlü bir haberleşme altyapısı bulunmaktadır. Bu iletişim sayesinde yer istasyonundan drone'ların anlık kontrolü ve takibi gerçekleştirilebilmektedir. Sistem yazılımının genel akışını incelemek için yer istasyonu, kâşif drone ve ilaçlama drone'larına ait akış diyagramlarının değerlendirilmesi gerekmektedir. Şekil 3.14'te yer istasyonuna ait akış diyagramı sunulmuştur.



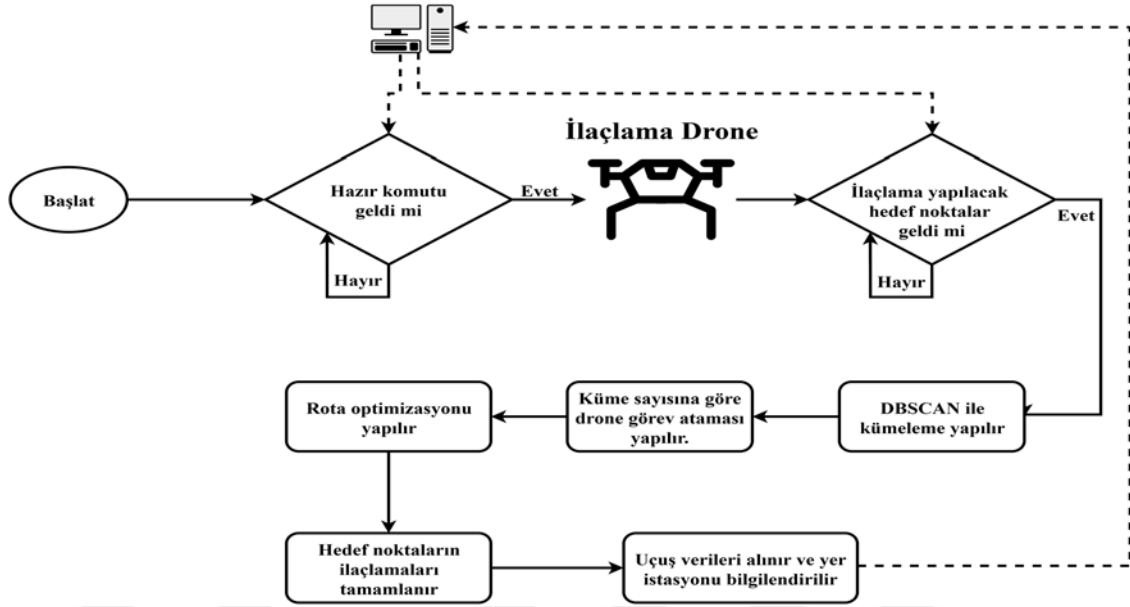
Şekil 3.14 Yer istasyonunun görev akış diyagramı.

Yer istasyonu, sistemin güvenli biçimde çalışmasını ve drone bağlantılarının kurulmasını sağlamaktadır. Sistem başlatıldığında, önceki oturumdan açık kalan bir program varsa öncelikle kapatılmaktadır. Ardından ArduPilot ve SITL programları çalıştırılmakta, bu adımın tamamlanmasından sonra Gazebo ve ROS başlatılmaktadır. Tüm drone'ların kontrol edilebilmesi için MAVROS etkinleştirilir. Bu aşamada sistem, programların doğru şekilde başlatılıp başlatılmadığını denetlemekte; bir hata tespit edilirse süreç otomatik olarak yeniden başlatılmaktadır. Tüm programlar sorunsuz biçimde çalışıyorsa drone bağlantıları kurulmakta ve ROS abonelikleri doğrulanmaktadır. Bağlantı hatası oluşması durumunda yeniden bağlanma işlemi gerçekleştirilir. Herhangi bir sorun tespit edilmediğinde, yer istasyonuna ait işlemler tamamlanır ve drone'lara "hazır" komutu gönderilir. Bu noktadan itibaren kâşif drone'un görev süreci başlamakta olup, bu sürecin akış diyagramı Şekil 3.15'te sunulmuştur.



Şekil 3.15 Kâşif drone’un görev akış diyagramı.

Tasarlanan sürü drone sisteminde, program başlatıldığında ve yer istasyonundan “hazır” komutu alındığında kâşif drone bekleme moduna geçmektedir. Bu aşamada kullanıcıdan bir görev komutu geldiğinde süreç başlatılmaktadır. Kâşif drone üzerinde kamera modülü bulunduğu için, sistemde nesne tespiti görevinden sorumludur. Bu nedenle drone’a tarama işlemi atanmaktadır. Tarama yapılacak bir alanın oluşturulması için üç veya daha fazla köşe noktasına ait GPS koordinatlarının girilmesi yeterlidir. Bu koordinatlar üzerinden tarama bölgesi tanımlanmakta ve bu alan içinde gidilmesi gereken hedef noktalar belirlenmektedir. Nesne tespiti için kamera modülü etkinleştirilir ve YOLO mimarisi çalıştırılır. Ardından, hedef noktalar arasında en verimli güzergâhın belirlenmesi amacıyla rota optimizasyonu uygulanır. Optimizasyon tamamlandığında tarama süreci başlar; drone hedef noktalar arasında hareket ederken tespit ettiği domates bitkilerinin GPS koordinatlarını kaydeder. Tarama işlemi tamamlandığında bu koordinatlar yer istasyonuna iletilir ve kâşif drone’un görevi sona erer. Bu aşamadan sonra, ilaçlama drone’ları tespit edilen konumlarda ilaçlama görevini yürütmektedir. İlaçlama drone’larının akış diyagramı Şekil 3.16’da sunulmuştur.



Şekil 3.16 İlaçlama drone’larının görev akış diyagramı.

İlaçlama drone’larında, yer istasyonundan “hazır” komutu alındığında ve kâşif drone tarafından gönderilen domates bitkilerinin koordinatları ulaştığında ilaçlama süreci başlatılmaktadır. Bu süreçte, öncelikle ilaçlama yapılacak noktalar Gürültülü Uygulamaların Yoğunluk Tabanlı Mekânsal Kümelenmesi (DBSCAN – Density-Based Spatial Clustering of Applications with Noise) yöntemiyle kümelere ayrılmaktadır. Böylece görev sayısına bağlı olarak kaç drone’un görevlendirileceği belirlenir. Küme sayısı bir olduğunda, kümeye en yakın ilaçlama drone’u görevlendirilir. Küme sayısının birden fazla olması durumunda ise görevler iki drone arasında paylaşılmaktadır. Bu aşamada, kümelerin yoğunluklarına göre katsayılar atanarak görev dağılımının dengeli olması sağlanır. Görev dağıtımının ardından, her drone’un hedef noktalarına en uygun güzergâh üzerinden ilerlemesi için rota optimizasyonu uygulanır. İlaçlama drone’u hedef noktasına ulaştığında, simülasyon ortamında ışık tabanlı işaretleme yöntemi kullanılarak ilgili nokta “ilaçlandı” olarak işaretlenir.

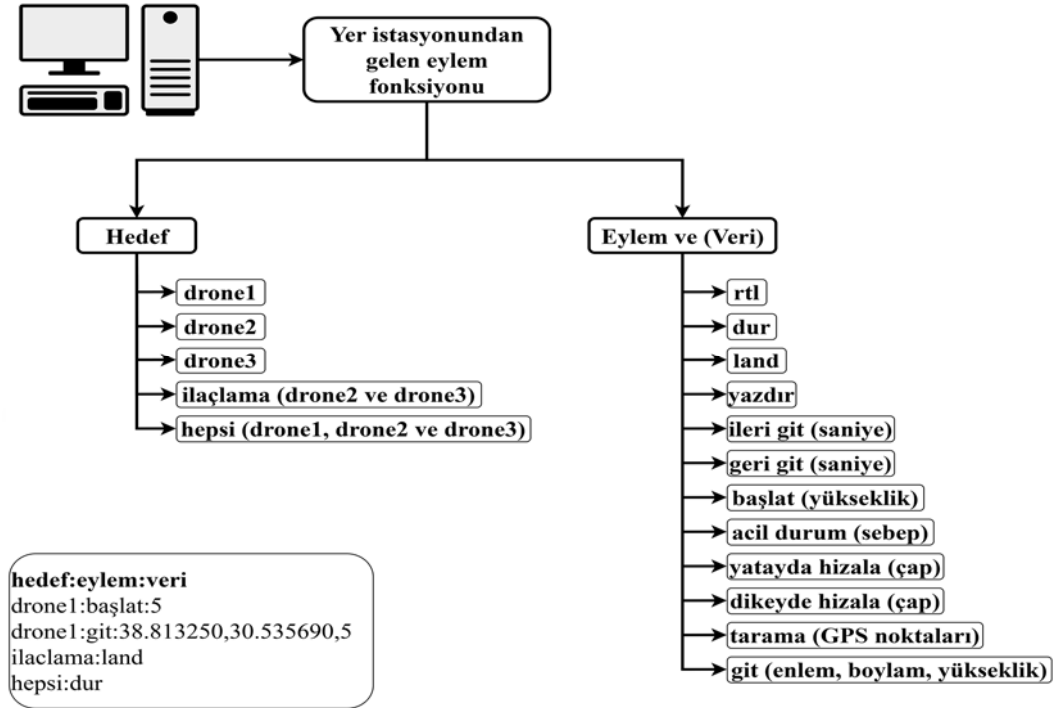
Uçuş boyunca LiDAR tabanlı engelden kaçınma katmanı aktif olarak çalışmaktadır. Drone, 5 m yarıçapında bir engel algıladığında uygun kaçış vektörünü izleyerek çarpışmayı önler. Acil durumlarda (örneğin batarya SOC değerinin %20’nin altında düştüğü durumda, sistem arızası veya operatör müdahalesi) görev güvenli biçimde sonlandırılır ve RTL ya da LAND komutu uygulanır. Bu süreçte kaydedilen uçuş

mesafesi, görev süresi, batarya SOC değeri ve optimizasyon süreleri, sistemin performansını değerlendirmek amacıyla Bölüm 3.3'te tanımlanan kriterler kapsamında analiz edilmiştir.

Tasarlanan sürü drone sisteminde, yukarıda açıklanan akış diyagramlarının uygulanabilmesi için Python programlama diliyle 23 adet fonksiyon geliştirilmiştir. Bu fonksiyonlar aşağıda listelenmiştir.

- Bütün süreçleri durdurma fonksiyonu
- Gazebo, SITL, MAVROS, DarkNet programlarının başlatılma fonksiyonları
- Drone'ların belirlenen irtifaya çıkma fonksiyonu
- Drone'ların RTL fonksiyonu
- LiDAR verileri ile engelden kaçma fonksiyonu
- İstenilen hedef noktaya ulaşmasının kontrol fonksiyonu
- Drone'ların yön değiştirme fonksiyonu
- İstenilen hedefe gitme fonksiyonu
- Drone'ların yatayda ve dikeyde hizalanma fonksiyonları
- Drone'ların ileri ve geri gitme fonksiyonları
- Tarama alanı oluşturma fonksiyonu
- Tarama başlatma fonksiyonu
- İlaçlamaya gitme fonksiyonu
- Drone'ların görsel haritalandırma fonksiyonu
- Görsel haritalandırma üzerinden canlı takip edebilme fonksiyonu
- Tespit edilen nesnenin filtrelenmesi fonksiyonu
- İlaçlama görselinin oluşturulma fonksiyonu
- Drone'ların gittiği mesafelerin belirlenme fonksiyonları
- Hedef noktaların kümelenme fonksiyonu
- Kümelenen noktaların drone'lara görev atama fonksiyonu
- Uçuş metriklerini kayıt altına alma fonksiyonu (mesafe, uçuş süresi, batarya)
- NN ve ACO optimizasyon algoritma fonksiyonları
- Yer istasyonundan gelen eylem fonksiyonu

Bu fonksiyonlardan yer istasyonundan gelen eylem fonksiyonunun kullanımına ait akış diyagramı Şekil 3.17’de sunulmuştur.



Şekil 3.17 Eylem fonksiyonunun akış diyagramı.

3.2.4 Sensör Verilerinin İşlenmesi ve Veri Akış Yapısı

Bu bölümde sistemde kullanılan sensörlerden elde edilen ham verilerin işlenme süreci ve görev yönetiminde anlamlı bilgiye dönüştürülmesi açıklanmaktadır. Drone’lardan alınan GPS verileri, konum ve yönelim bilgilerinin belirlenmesinde kullanılmakta ve hedef koordinatlarıyla eşleştirilerek rota planlamasına aktarılmaktadır. LiDAR sensöründen elde edilen mesafe verileri filtrelenerek engel algılama ve çarpışma önleme algoritmalarına girdi sağlamaktadır. Batarya SOC değerleri, anlık enerji durumunun izlenmesi amacıyla sürekli takip edilmekte ve kritik seviyelerde acil durum modüllerine yönlendirilmektedir. Kamera görüntüleri ise YOLO tabanlı nesne tespit algoritmasına aktarılmakta, bu sayede tarım alanındaki bitkilerin tespit işlemi gerçekleştirilmektedir. Kamera kadrājına giren bir bitkinin tekrar tekrar kayıt altına alınmaması için bir filtreleme işlemi uygulanmaktadır. Bu filtreleme sayesinde kayıt altına alınan hedef bölge koordinatlarında aynı bitki için birden fazla nokta oluşturulmamaktadır. Tüm bu

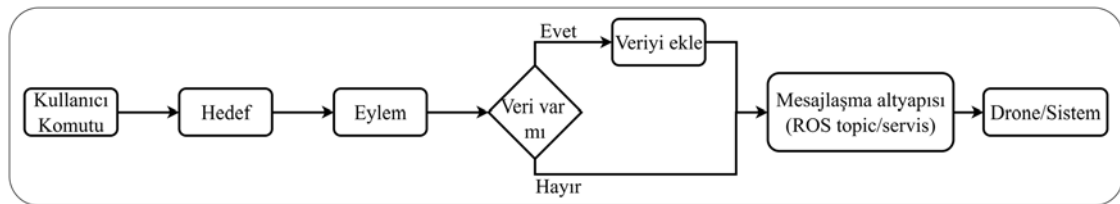
sensörlerden elde edilen veriler ROS ekosisteminde yayıncı/izleyici mekanizmasıyla toplanmakta, senkronize edilmekte ve görev yönetim modülüne aktarılmaktadır. Böylece ham sensör verileri, görev paylaşımı, rota optimizasyonu ve acil durum senaryolarında kullanılabilecek anlamlı bilgiye dönüştürülmektedir. Rostopic list üzerinden elde edilen sensör verilerinin okunması için Çizelge 3.4'teki gibi rostopic adresleri gösterilmiştir.

Çizelge 3.4 Kullanılan sensörlerin rostopic adresleri.

Drone	Sensör	Rostopic adresi
Drone 1	GPS	/drone1/mavros/global_position/global
	LiDAR	/spur/laser/scan
	Batarya	/drone1/mavros/battery
	DarkNet	/darknet_ros/bounding_boxes
Drone 2	GPS	/drone2/mavros/global_position/global
	LiDAR	/spur/laser/scan2
	Batarya	/drone2/mavros/battery
Drone 3	GPS	/drone3/mavros/global_position/global
	LiDAR	/spur/laser/scan3
	Batarya	/drone3/mavros/battery

3.2.5 Mesajlaşma Altyapısı

Bu bölümde öncelikle modüller arası iletişimi sağlayan mesajlaşma altyapısı ele alınmaktadır. Mesajlaşma altyapısı, ROS tabanlı yayıncı/izleyici ve servis mekanizmaları üzerinden gerçekleştirilmekte, böylece görev yönetim modülü ile sensör, kontrol ve optimizasyon bileşenleri arasında senkronize bir veri alışverişi sağlanmaktadır. Bu yapı sayesinde dışarıdan gelen kullanıcı komutları standart bir formata dönüştürülmekte ve sistem bileşenleri arasında güvenilir şekilde iletilmektedir. Komutlar hedef, eylem ve veri sıralamasına göre işlenmekte; ilgili mesajlar görev yönetim modülüne ulaştığında çözümlenerek sürüdeki drone'lara aktarılmaktadır. Mesajlaşma altyapısının akış diyagramı Şekil 3.18'de verilmiştir.



Şekil 3.18 Hedef – Eylem – Veri sıralamasına dayalı mesajlaşma akışı.

Kullanıcı tarafından gönderilen komutlar önce hedefe, ardından eyleme ve gerektiğinde ilgili verilere ayrıştırılmakta; bu bileşenler mesajlaşma altyapısı üzerinden görev yönetim modülüne iletilerek drone veya sürü üzerinde uygulanmaktadır. Mesajlaşma altyapısında ilk adım, kullanıcının göndereceği komutun hangi hedefe yönelik olduğunu tanımlamaktır. Hedef, komutun kapsamını belirler ve sistemin hangi drone veya drone grubunu etkileyeceğini ifade eder. Bu yapı sayesinde hem bireysel hem de toplu operasyonlar kolayca yönetilebilmektedir. Sistemde tanımlanan temel hedef türleri Çizelge 3.5’te verilmiştir.

Çizelge 3.5 Tasarlanan sürü drone sisteminde tanımlanan hedef komutları.

Hedef Adı	Hedefin kapsadığı drone’lar
Drone 1	Drone 1
Drone 2	Drone 2
Drone 3	Drone 3
İlaçlama	Drone 2 ve Drone 3
Hepsi	Drone 1, Drone 2 ve Drone 3

Komut yapısında ikinci aşama, hedef üzerinde gerçekleştirilecek eylemin tanımlanması ve gerekliyse bu eyleme ilişkin verinin sağlanmasıdır. Eylem, sistemin hangi işlevi yerine getireceğini belirtirken; veri, bu işlevin gerçekleştirilmesi için gerekli parametreleri içerir. Böylece bazı komutlar yalnızca eylem ile tanımlanırken bazıları ise eylem ve veri bileşimi ile anlam kazanır. Çizelge 3.6’da eylemler ve varsa eylemlerde kullanılacak olan verilerin açıklamaları verilmiştir. Bu mesajlaşma altyapısı sonraki bölümlerde ele alınacak görev planlama ve tarama süreçlerinin temelini oluşturmaktadır.

Çizelge 3.6 Tasarlanan sürü drone sisteminde tanımlanan eylemlerin ve varsa verilerin açıklamaları.

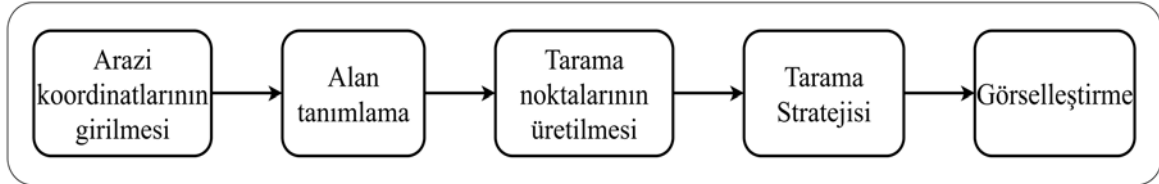
Eylem Adı	Veri	Açıklama
Kalkış_yap	İrtifa	Drone’un istenilen irtifaya çıkmasını sağlar.
Git	Koordinat (x,y,z)	Drone’un istenilen koordinata gitmesini sağlar.
Yatayda_hizala	Referans alınacak drone ve mesafe (m)	Referans alınan drone’a göre x ekseninde belirlenen drone’lar arası mesafeye göre hizalama yapar.
Dikeyde_hizala	Referans alınacak drone ve mesafe (m)	Referans alınan drone’a göre y ekseninde belirlenen drone’lar arası mesafeye göre hizalama yapar.
İleri_git	İleri gitme süresi (s)	Drone’un istenilen süre kadar ileri gitmesini sağlar.
Geri_git	Geri gitme süresi (s)	Drone’un istenilen süre kadar geri gitmesini sağlar.

Çizelge 3.7 (Devam) Tasarlanan sürü drone sisteminde tanımlanan eylemlerin ve varsa verilerin açıklamaları.

Eylem Adı	Veri	Açıklama
Tarama_yap	Arazi Koordinatı	Koordinatları verilen arazi üzerinde kâşif drone'un tarama yapmasını sağlar.
Acil_dur	Acil durma sebebi	Drone'un acil durmasını ve eve dönmesini sağlar.
İniş_yap	-	Drone'un olduğu konuma iniş yapmasını sağlar.
Eve_dön	-	Drone'un kalkış yaptığı yere geri gelip iniş yapmasını sağlar.
Dur	-	Drone'un görevlerini iptal ederek havada asılı kalmasını sağlar.
Yazdır	-	Drone'ların kayıt altına aldıkları verilerin yazdırılmasını sağlar.

3.2.6 Tarama Modülü

Bu bölümde tarama modülü ve kâşif drone'unun görev alanı üzerindeki tarama süreci ele alınmaktadır. Kullanıcı tarafından koordinatları tanımlanan alan, sistem tarafından işlenmekte ve bu alanın eksiksiz şekilde taranabilmesi için gerekli referans noktaları oluşturulmaktadır. Drone, bu noktaları takip ederek alanı sistematik biçimde taramakta ve süreç içerisinde nesne tespiti gerçekleştiğinde ilgili koordinatları kaydetmektedir. Şekil 3.19'da sürü drone sistemindeki tarama modülünün akış diyagramı verilmiştir.



Şekil 3.19 Sürü drone sistemindeki tarama modülünün akış diyagramı.

Tarama sürecinin ilk adımı, kullanıcının vermiş olduğu koordinat bilgileri ile görev alanının sınırlarının belirlemesidir. Kullanıcı tarafından verilen üç veya daha fazla köşe koordinatı sisteme girilmekte, bu koordinatlar bir poligon oluşturularak tarama yapılacak arazi çerçeve içerisine alınmaktadır. Böylece sistem, yalnızca ilgili alan içerisinde işlem yapılmasını garanti altına almaktadır. Belirlenen bu sınırlar içerisinde tarama noktaları oluşturulmaktadır. Kullanıcı tarafından girilen tarama parametrelerine göre grid yapısı oluşturulmaktadır. Bu grid yapısı, satır/sütun mantığında, belirli bir mesafeye göre yerleştirilmiş noktalardan meydana gelmektedir. Grid noktaları, koordinat sistemine

dönüştürülerek drone'un uçuş planına aktarılacak referans konumları oluşturmaktadır. Kâşif drone, üretilen tarama noktaları üzerinden bir rota oluşturarak sistematik bir uçuş gerçekleştirmektedir. Rota oluşturma aşamasında NN ve ACO algoritmaları ile optimizasyon yapılmaktadır. Optimizasyon sonucunda elde edilen en uygun rota noktaları oluşturulan drone harita sistemine aktarılmaktadır. Hazırlanan harita sistemi üzerinde drone'ların canlı konum takibi yapılabilmekte ve drone'ların gidecek olduğu hedef noktalar belirtilmektedir.

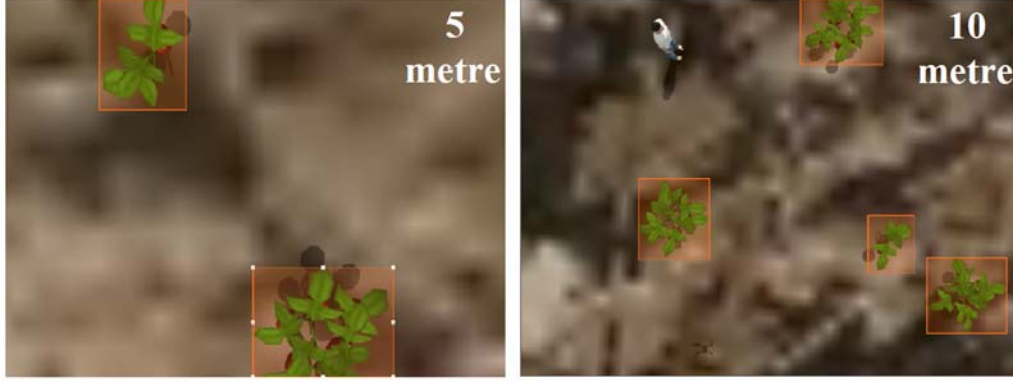
3.2.7 Oluşturulan Veri Seti

Bu çalışmada kullanılan veri seti, Gazebo tabanlı simülasyon ortamında kâşif drone'un farklı irtifalarda gerçekleştirdiği uçuşlardan elde edilmiştir. Kâşif drone, 5 m ve 10 m yüksekliklerde uçurularak video kayıtları alınmış, bu kayıtlar Python/OpenCV kütüphanesi kullanılarak karelere ayrılmıştır. Görseller Kırmızı, Yeşil, Mavi (Red, Green, Blue – RGB) renk formatında ve 640×480 piksel çözünürlükte oluşturulmuş, her biri PNG biçiminde kaydedilmiştir. Elde edilen toplam 2023 görüntü, modelin ölçek değişimlerine karşı dayanıklılığını artırmak ve bitkilerin farklı perspektiflerden tanınabilmesini sağlamak amacıyla iki farklı irtifadan oluşturulmuştur. Görüntülerin yaklaşık %25'i 5 m, %75'i ise 10 m irtifadan elde edilmiştir.

Görsellerin etiketlenmesi sürecinde, çevrimiçi bir etiketleme aracı kullanılmış; etiketler YOLO formatına uygun olarak (sınıf kimliği, x_merkez, y_merkez, genişlik, yükseklik) normalleştirilmiş koordinat bilgilerini içerecek şekilde oluşturulmuştur. Her karede bulunan tüm domates bitkileri tek tek etiketlenmiş, bazı karelerde beş adede kadar domates bitkisi bulunmuştur. Etiketleme süreci yoğun emek gerektirse de modelin doğruluk ve güvenilirlik açısından yüksek performans göstermesi için kritik bir aşama olmuştur.

Veri seti, 1700 görüntü eğitim ve 323 görüntü doğrulama olmak üzere iki alt gruba ayrılmıştır. Eğitim seti, modelin bitkilerin genel görsel özelliklerini öğrenmesini sağlarken; doğrulama seti, modelin eğitimde görmediği örnekler üzerindeki başarımını değerlendirmek amacıyla kullanılmıştır.

Veri setinde farklı ışık koşulları, bitki yoğunlukları, arka plan dokuları ve kamera açıları yer almakta; bu çeşitlilik modelin genellenebilirliğini ve gerçek ortamlarda farklı senaryolara uyum yeteneğini artırmaktadır. Şekil 3.20’de veri setinin yapısını örneklemek amacıyla 5 ve 10 m yüksekliklerden alınmış görüntülerin etiketlenmiş halleri verilmiştir.



Şekil 3.20 Domates bitkisinden oluşan veri setinde 5 ve 10 m yükseklikten elde edilen veri seti görüntülerinin etiketlenmiş halleri.

3.2.8 YOLO Tabanlı Nesne Tespitinin Eğitimi ve Entegrasyonu

Bu çalışmada domates bitkilerinin tespiti için YOLO mimarileri tercih edilmiş hem YOLOv3 hem de YOLOv5 modeline ait toplam sekiz farklı varyant üzerinde eğitim gerçekleştirilmiştir. YOLOv3 modelinden YOLOv3, YOLOv3-Tiny ve YOLOv3-SPP varyantları, YOLOv5 modelinden ise YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l ve YOLOv5x varyantları Ultralytics altyapısı kullanılarak eğitilmiş ve performansları karşılaştırmalı olarak incelenmiştir.

Eğitim süreci, Gazebo simülasyon ortamından elde edilen özel domates veri seti kullanılarak yürütülmüştür. Tüm eğitimler, Intel Core i7-10700 işlemci, 128 GB RAM ve NVIDIA GeForce RTX 3080 GPU donanımına sahip bir bilgisayar üzerinde PyTorch tabanlı Ultralytics altyapısı kullanılarak ve CUDA 11.4 ile cuDNN 8.9.7 kütüphaneleri aracılığıyla GPU hızlandırması sağlanarak gerçekleştirilmiştir.

YOLOv3 ve YOLOv5 mimarilerinde aynı eğitim parametreleri kullanılmış olup Çizelge 3.7’de bu parametreler özetlenmiştir. Tüm modeller aynı eğitim protokolü çerçevesinde çalıştırılmıştır. Mimari farklılıklar nedeniyle modellerin parametre sayıları ve Giga

Floating Point Operations per Second (GFLOPs) değerleri performans üzerinde belirleyici bir etkiye sahip olmuştur. GFLOPs, bir modelin s’de gerçekleştirdiği milyar düzeyindeki kayan nokta işlemlerini ifade etmektedir. Yüksek GFLOPs değerleri daha fazla hesaplama gücü ve güçlü donanım gereksinimi anlamına gelirken düşük GFLOPs değerleri gömülü ve gerçek zamanlı uygulamalarda önemli bir avantaj sağlamaktadır. Dolayısıyla GFLOPs değeri, model seçiminde hız ve donanım uyumluluğu açısından kritik bir kıyaslama ölçütü olarak kullanılmaktadır.

Çizelge 3.8 Nesne tespitinde için YOLOv3 – YOLOv5 mimarilerinde kullanılan eğitim parametreleri.

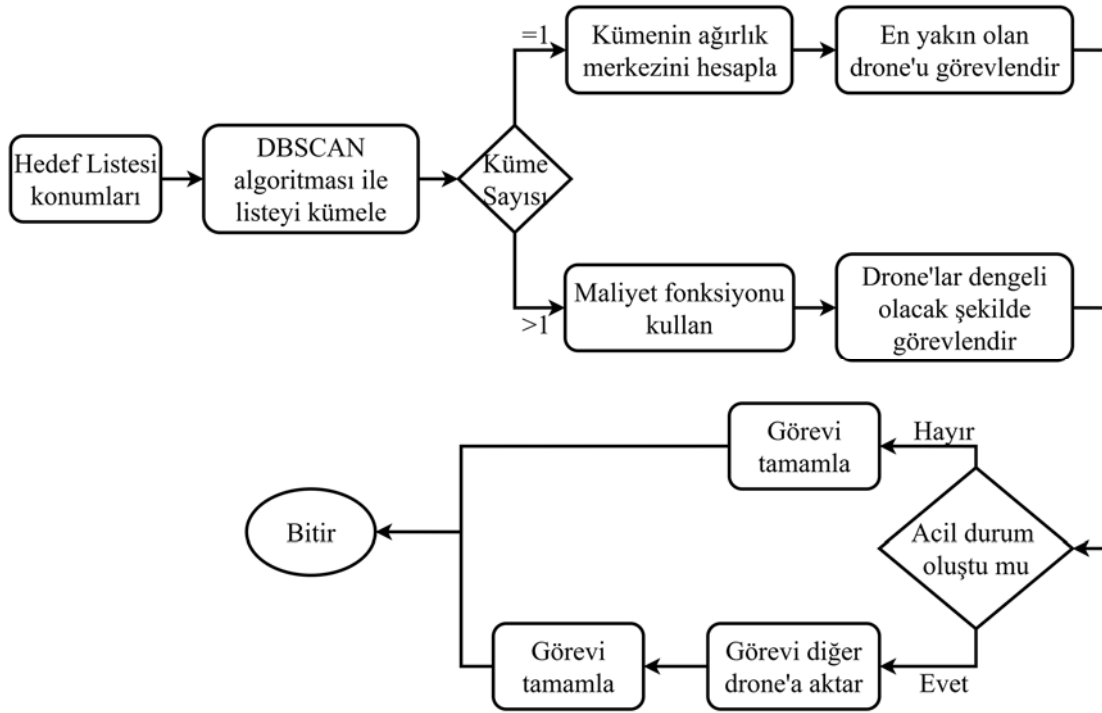
Parametre	Açıklama
Giriş boyutu	640 x 640
Epochs	50
Yığın büyüklüğü (Batch size)	8
Öğrenme oranı (Learning rate ((lr0 / lrf)))	0.001
Momentum	0.9
Ağırlık küçülmesi (Weight decay)	0.0005
Sınıf sayısı	1
Optimizasyon Algoritması	SGD

Yapılan eğitimlerin entegrasyonu aşamasında ise YOLOv3 modelinin varyantları için DarkNet tabanlı sürümleri kullanılarak ROS ve Gazebo simülasyon ortamına entegre edilmiş ve gerçek zamanlı çalıştırılabilmiştir. Buna karşılık YOLOv5 mimarilerinde aynı entegrasyon mümkün olmamıştır. Bunun temel sebebi, kullanılan Ubuntu 18.04 işletim sistemi, ROS Melodic altyapısı ve Python 3.6.9 sürümünün YOLOv5 için gerekli olan en az Python 3.10 gereksinimini karşılamamasıdır. Python sürümünün yükseltilmesi ise yalnızca yazılım güncellemesi ile çözülememekte, ROS sürümünün (Noetic veya ROS 2) ve işletim sistemi altyapısının da değiştirilmesini gerektirmektedir. Bu nedenle YOLOv5 varyantları yalnızca çevrimdışı ortamda test edilmiş, sistem entegrasyonu kapsamında ROS tabanlı gerçek zamanlı çalıştırma yalnızca YOLOv3 varyantları ile sağlanabilmiştir. Yapılan eğitimlerin sonuçları Bulgular bölümünde detaylandırılarak incelenmiştir.

3.2.9 Görev Planlama Modülü

Görev planlama modülü, tarama görevini üstlenen kâşif drone tarafından elde edilen bitki koordinatlarını işleyerek ilaçlama için hedef listesi oluşturmaktadır. Bu hedefler sürüdeki

ilaçlama drone'larına dağıtılırken yalnızca mesafe değil aynı zamanda mevcut konum, batarya SOC değeri ve iş yükü gibi kriterler de göz önünde bulundurulmaktadır. Böylece görevler tek bir drone üzerinde yoğunlaşmamakta, sürü genelinde dengeli ve verimli bir dağılım sağlanmaktadır. Görev planlama modülünün akış diyagramı Şekil 3.21'de verilmiştir.



Şekil 3.21 Tasarlanan sürü drone sistemindeki görev planlama modülü akış diyagramı.

Hedef dağılımında ilk adım, nesne tespiti ile belirlenen konumların DBSCAN algoritması ile mekânsal yoğunluklarına göre kümelenmesidir. Bu yöntem, birbirine yakın konumdaki bitkileri aynı küme altında toplayarak daha düzenli ve bütünlüklü bir planlama yapılmasına imkân verir. Tek bir kümenin oluşması halinde, kümenin ağırlık merkezi hesaplanmakta ve bu merkeze en yakın konumda bulunan drone görevlendirilmektedir. Birden fazla küme oluştuğunda ise, kümelerin drone'lara atanması süreci yalnızca mesafeye dayalı yapılmamakta, aynı zamanda yük dengesi de gözetilmektedir. Bu amaçla, toplam mesafeyi ve iki drone arasındaki görev sayısı farkını birlikte minimize eden bir maliyet fonksiyonu kullanılmaktadır. Böylece hem uçuş mesafeleri azaltılmakta hem de görevlerin adil paylaşımı sağlanmaktadır.

Görev planlama modülü yalnızca başlangıçta atama yapmakla sınırlı değildir; süreç boyunca görevlerin durumunu izlemekte ve gerektiğinde (acil durum senaryolarında) yeniden planlama yapabilmektedir. Bu yapı sayesinde görev planlama modülü, sürünün dinamik koşullara uyum sağlayarak görevlerin sürekli güncellenmesini mümkün kılmaktadır.

3.2.10 Acil Durum Senaryoları

Bu bölümde görev sırasında ortaya çıkabilecek acil durumlar ve bu durumlarda uygulanan çözüm mekanizmaları açıklanmaktadır. Drone'ların görev sırasında batarya SOC değerinin kritik eşiğin (%20) altına düşmesi, donanım arızası yaşaması, iletişim kopuklukları veya çevresel engeller nedeniyle görevini sürdürememesi gibi durumlarda görev planlama modülü devreye girerek acil durum yönetimini gerçekleştirmektedir. Acil durum algılandığında ilgili drone otomatik olarak RTL moduna geçirilmekte ve güvenli bir şekilde kalkış noktasına geri dönmektedir. Bu süreçte drone'un mevcut görevi iptal edilmekte ve kalan hedefler yeniden değerlendirilmek üzere görev planlama modülüne aktarılmaktadır.

İlaçlama hedefleri tek bir küme altında toplanmış ve acil bir durum oluşmuşsa, görev diğer drone tarafından devralınmaktadır. Çok kümeli senaryolarda ise görev planlama modülü devreye girerek tüm kümeleri yeniden birleştirmekte, aynı koordinatlara sahip tekrar eden hedefleri tekilleştirmekte ve kalan drone'a yeni görev ataması yapmaktadır. Böylece görevler dengeli şekilde paylaştırılmakta ve kesintiye uğramadan sürdürülmektedir.

Sonuç olarak acil durum senaryolarına karşı geliştirilen bu mekanizmalar sayesinde bir drone'un görev dışı kalması durumu sistemin bütününe etkilememektedir. Böyle durumlarda sürü dinamik koşullara uyum sağlayarak görevlerin güvenli ve eksiksiz bir şekilde tamamlanmasını mümkün kılmaktadır.

3.2.11 Görselleştirme ve Harita Üzerinde Görev Takibi

Bu bölümde görev icrası sırasında elde edilen konum ve durum bilgilerinin harita üzerinde görselleştirilmesi ve anlık olarak takip edilmesi açıklanmaktadır. Drone'ların GPS verileri sürekli olarak kaydedilmekte ve her bir drone harita üzerinde kendine özgü bir renk ile temsil edilmektedir. Böylece sürüdeki tüm drone'ların anlık konumları kullanıcı tarafından kolaylıkla izlenebilmektedir. Görev noktaları ise sıralı olarak numaralandırılmakta ve harita üzerinde işaretlenmektedir. Bu sayede hem tarama hem de ilaçlama görevlerinin ilerleyişi görsel olarak takip edilebilmektedir.

Görev noktalarının durumları renk kodlaması ile ayırt edilmektedir. İlk işlemler başladığında hedef noktaların renkleri atanan drone'un rengi ile aynı olacak şekilde ayarlanmıştır. Böylelikle harita üzerinden hangi drone'un hangi hedeflere gideceği görülebilmektedir. Kâşif drone'un tarama yapmak için ulaştığı hedef noktalar turkuaz renge dönüşmektedir. İlaçlama drone'larında ise hedefe ulaşıp ilaçlama yapıldığında hedef nokta yeşil renge dönüşmektedir. Böylece görevlerin mevcut ilerleme düzeyi harita üzerinde net bir biçimde görülebilmektedir. Bununla birlikte, drone'ların uçuş performansına ilişkin toplam mesafe, görev süresi ve batarya SOC değerleri de hesaplanmakta ve görev tamamlama sonrasında değerlendirme amacıyla raporlanmaktadır.

Bu görselleştirme altyapısı sayesinde görevlerin tüm aşamaları kullanıcıya anlaşılır ve takip edilebilir bir biçimde sunulmakta, aynı zamanda performans analizleri için gerekli veriler sistematik biçimde kayıt altına alınmaktadır.

3.3 Sistemin Değerlendirme Kriterleri

Bu çalışmada geliştirilen sistemin başarımı, farklı boyutları dikkate alan çeşitli kriterler üzerinden değerlendirilmiştir. Öncelikle nesne tespiti performansı incelenmiştir. YOLO tabanlı nesne tespit algoritmasının doğruluğu Precision, Recall ve Ortalama Hassasiyet (Mean Average Precision – mAP) değerleri ile ölçülmüş, böylece sistemin tespit doğruluğu sayısal olarak değerlendirilmiştir.

Bir diğerk ölçüt drone'ların görev sırasında kat ettikleri toplam uçuş mesafesidir. Mesafeler Haversine yöntemi kullanılarak hesaplanmış, bu değerler hem görev süresinin hem de enerji tüketiminin temel belirleyicisi olarak değerlendirmeye alınmıştır. Bununla bağlantılı olarak sistemin toplam görev süresi de kritik bir kriterdir; tarama, nesne tespiti, rota planlama ve ilaçlama adımlarının tamamı göz önünde bulundurularak görevlerin ne kadar sürede tamamlandığı kaydedilmiştir.

Sistemin enerji verimliliğini ölçmek amacıyla drone'ların batarya SOC değerleri takip edilmiştir. Bu değerler anlık olarak izlenmiş, uçuş sırasında tüketilen enerji miktarı ile görev tamamlama başarımı ilişkilendirilmiştir. Ayrıca rota optimizasyonunda kullanılan yöntemlerin (NN ve ACO) performansı optimizasyon süresi üzerinden karşılaştırılmıştır. Bu ölçüm, algoritmaların gerçek zamanlı kullanım için uygunluğunu değerlendirmede önemli bir ölçüt sağlamaktadır.

Son olarak sistemin görev paylaşım verimliliği ele alınmıştır. Bu kapsamda ilaçlama hedeflerinin drone'lar arasında dengeli bir şekilde dağıtılıp dağıtılmadığı incelenmiş, böylece görev yükünün sürü genelinde adil ve verimli biçimde paylaşılıp paylaşılmadığı değerlendirilmiştir. Belirtilen kriterler doğrultusunda sistemin performansına ilişkin deneysel sonuçlar Bölüm 4'te ayrıntılı olarak sunulmaktadır.

4.BULGULAR

Bu bölümde, YZ destekli sürü drone sistemine ait simülasyon çalışmalarından elde edilen bulgular sunulmaktadır. Bulgular; sürü hareketlerini, görev planlama, nesne tespiti, rota optimizasyon performansı ve acil durum senaryoları kapsamında değerlendirilmiş, geliştirilen mimarinin etkinliği ve güvenilirliği sistem çıktıları üzerinden değerlendirilmiştir.

4.1 Sürü Davranışı ve Acil Durum Senaryoları

Bu bölümde, geliştirilen sürü kontrol yapısının acil durum senaryolarındaki tepkileri incelenmiştir. Simülasyon ortamında kâşif drone’u taramayı tamamladıktan sonra ilaçlama drone’una ya da drone’larına hedef noktalar iletilmekte ve drone’ların o hedef noktaları ilaçlaması sağlanmaktadır. Bu aşamada oluşabilecek olan bir acil durum (batarya SOC değerinin %20’nin altına düşmesi, ilacın bitmesi, haberleşme kaybı) senaryosunda drone’ların istenilen şekilde hareket edip etmedikleri test edilmiştir. Çizelge 4.1’de acil durum senaryosunda elde edilen veriler sunulmuştur.

Çizelge 4.1 Acil durum senaryosunda elde edilen veriler.

Arazi Türü	Acil Duruma Giren Drone	Görevi Devralan Drone	Aktarılan Hedef Sayısı	Drone 2 Uçuş Süresi (s)	Drone 3 Uçuş Süresi (s)	Toplam Uçuş Süresi (s)
Tekli Arazi (Arazi 1)	-	-	0	-	160.80	160.80
Tekli Arazi (Arazi 1)	Drone 3	Drone 2	2	122.28	141.19	263.47
Tekli Arazi (Arazi 1)	Drone 3	Drone 2	4	127.87	133.41	261.28
Tekli Arazi (Arazi 1)	Drone 3	Drone 2	7	109.12	151.43	260.55
İkili Arazi (Arazi 1 ve 2)	Drone 2	Drone 3	0	121.09	154.32	275.41
İkili Arazi (Arazi 1 ve 2)	Drone 2	Drone 3	1	119.33	183.56	302.89
İkili Arazi (Arazi 1 ve 2)	Drone 2	Drone 3	3	98.35	202.76	301.11
İkili Arazi (Arazi 1 ve 2)	Drone 2	Drone 3	5	74.52	214.94	289.46

Çizelge 4.1’de görüldüğü üzere iki farklı acil durum senaryosu test edilmiştir. Birinci senaryoda tek bir arazi üzerinde drone 3 ile ilaçlama yapılırken acil durum oluşturulmuş ve beklemede olan drone 2 havalandırarak görevi devam ettirmiştir. Toplam 9 hedef noktada ilaçlama yapılmaktadır. İlk olarak acil durum oluşturulmadan önce yapılan ilaçlamada 160.80 s içerisinde ilaçlama tamamlanmıştır. Acil durum senaryosu oluşturulmuş ve drone 2’nin havalandırması sağlanmıştır. Aktarılan hedef sayısına göre toplam uçuş süresi değişmekte olup hedef sayısı ne kadar çok olursa toplam uçuş süresi o kadar düşmektedir. Drone 2’nin beklemede olup acil durum ile ilgili bölgeye gitmesi uzun sürdüğü için toplam uçuş süresi yaklaşık 260 s’ye yükselmiştir. İkinci senaryoda ise iki farklı arazinin aynı anda ilaçlaması gerçekleşmektedir. Bu ilaçlama anında drone 2 için acil durum oluşturulmuş ve drone 3 kalan hedef noktaları devralmıştır. Acil durum oluşturulmadan önce 275.41 s süren ilaçlama süresi acil durum ile yaklaşık 303 s’ye çıkmıştır.

Tekli arazi senaryosunda görev devri süresi önemli bir artışa (yaklaşık +100 s) neden olurken çoklu arazi senaryosunda süre artışı (+28 s) sınırlı kalmıştır. Dolayısıyla birden fazla drone’un eş zamanlı görev yapması, sistemin acil durumlara karşı hata toleransını artırarak görev sürekliliğini güvence altına almaktadır. Yapılan acil durum senaryolarının görsel olarak harita üzerinden görünümü de oluşturulmuştur. Bu görsel görünüm Bölüm 4.2’de detaylı bir şekilde açıklanmıştır.

Bu sonuçlar, tekli arazi senaryosunda görev devrinin toplam süreye etkisi araziye göre değişmekle birlikte arazi 1 için %62 civarında ek yük getirdiğini, ikili arazi (arazi 1 ve 2) senaryosunda ise artışın yalnızca %10 civarında kaldığını göstermektedir.

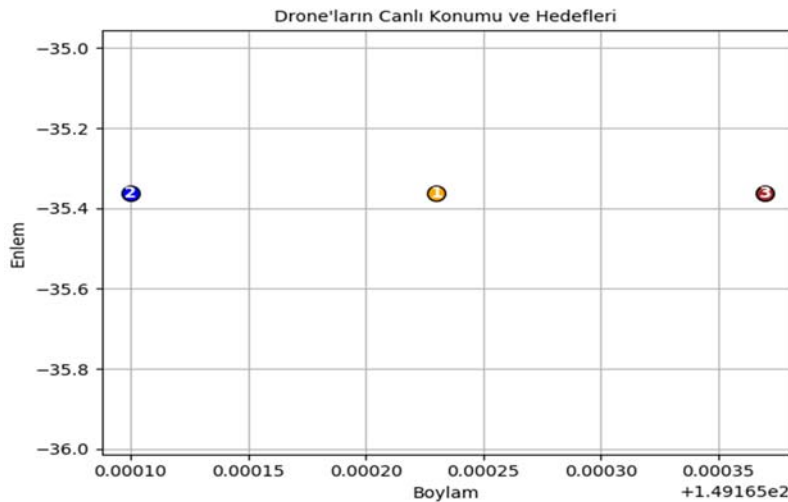
Drone’ların aynı arazi üzerinde olası bir rota planlama durumunda ise çarpışma engelleme algoritması devreye girmekte ve LiDAR sensörü sayesinde 5 m yarıçapında engelleri tespit etmektedir. Oluşturulan engelden kaçınma algoritması sayesinde drone’ların hem birbirleriyle hem de çevresel engellerle çarpışması önlenmektedir. Gerçekleştirilen testler sonucunda, drone’lar arasında herhangi bir çarpışma durumu gözlemlenmemiştir.

Sonuçlar, önerilen sürü yönetim yapısının acil durumlara karşı esnek ve güvenilir olduğunu göstermektedir. Özellikle görev devri senaryosunda hedeflerin kısa sürede yeniden dağıtılması, sürünün görev tamamlamadaki sürekliliğini sağlamıştır. Çarpışma önleme modülü, test edilen tüm koşullarda etkin çalışarak sistemin bütünlüğünü ve görev güvenliğini korumuştur.

4.2 Drone Görev Akışlarının Harita Tabanlı Görselleştirilmesi

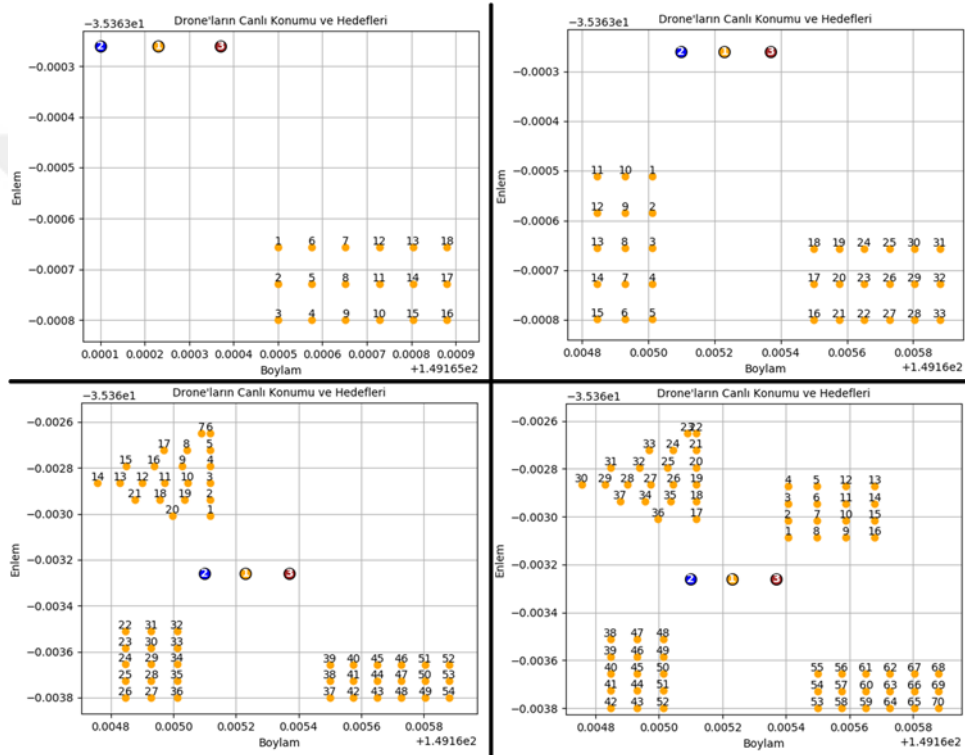
Bu bölümde geliştirilen sistemin görsel çıktıları sunulmuştur. Canlı konum haritası sayesinde drone'ların anlık hareketleri ve görev durumları gerçek zamanlı olarak izlenebilmekte, tarama ve ilaçlama süreçleri ise renk kodları ile görsel olarak takip edilebilmektedir. Ayrıca uçuş esnasında meydana gelen acil durumlarda hedeflerin yeniden dağılımı haritalar üzerinden gösterilerek sistemin esnekliği ortaya konmuştur. Böylece, 4.1'de sunulan sayısal bulguların operasyonel karşılığı görsel düzeyde desteklenmiştir.

Drone'ların GPS tabanlı canlı konumlarının haritası Şekil 4.1'de görülmektedir. Harita üzerinde her bir drone farklı renklerle temsil edilmekte ve anlık olarak konum bilgileri sürekli güncellenmektedir. Bu sayede operatör, tüm sürünün hareketlerini tek bir ekrandan takip edebilmekte ve görevlerin ilerleyişini gerçek zamanlı olarak gözlemleyebilmektedir.



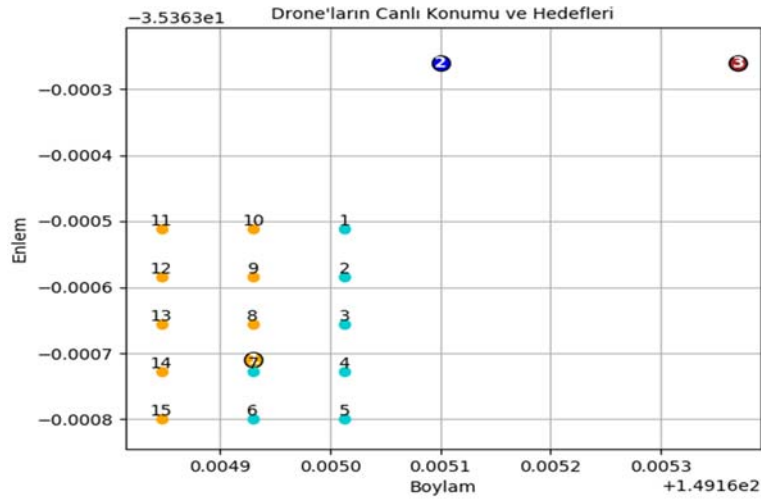
Şekil 4.1 Drone'ların GPS tabanlı canlı konumlarının haritası.

Tarama modülü ile arazilerin kapsama alanlarının belirlenmesi amacıyla görev noktaları oluşturulmuştur. Şekil 4.2’de görüldüğü üzere, her arazi için belirlenen noktalar grid yapısında konumlandırılmış ve böylece tarama sırasında homojen bir kapsama elde edilmesi sağlanmıştır. Farklı senaryolar kapsamında tekli, ikili, üçlü ve dörtlü arazi konfigürasyonları ayrı ayrı planlanarak test edilmiştir. Bu yapı sayesinde sistem, farklı büyüklükteki araziler için esnek bir şekilde görev planlaması yapabilmekte ve sürü içerisindeki drone’lara uygun görev dağılımı gerçekleştirebilmektedir.



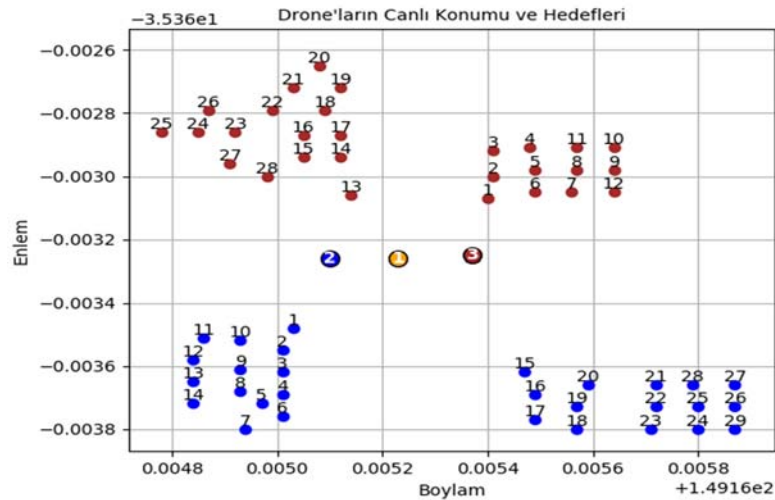
Şekil 4.2 Tarama modülü ile oluşturulan görev noktalarının haritası.

Tarama işlemi sırasında kâşif drone'un ziyaret ettiği her görev noktası sistem tarafından turkuaz renge dönüştürülmektedir. Böylece harita üzerinde, drone'un geçtiği ve taradığı bölgeler anlık olarak işaretlenmektedir. Şekil 4.3'te, Arazi 2 için gerçekleştirilen tarama sürecinde kâşif drone'un takip ettiği rota boyunca görev noktalarının turkuaz renge dönüştüğü görülmektedir. Böylece taramanın hangi noktalarda tamamlandığı kolaylıkla takip edilebilmektedir.



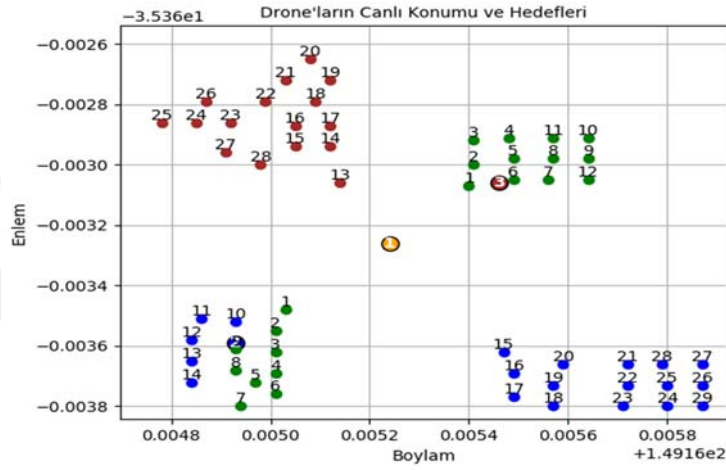
Şekil 4.3 Arazi 2’de kâşif drone’un tarama işlemi sırasında görev noktalarının turkuaz renge dönüştürülmesi.

Tarama süreci tamamlandıktan sonra kâşif drone tarafından belirlenen noktalar, kontrol algoritması ile kümelenecek ilaçlama görevleri oluşturulmuştur. Her küme, ilgili ilaçlama drone’una atanmış ve bu atama harita üzerinde renklerle görselleştirilmiştir. Şekil 4.4’te görüldüğü üzere, drone 2’ye atanan hedef noktalar mavi, drone 3’e atanan hedef noktalar ise kırmızı ile işaretlenmiştir. Bu yöntem sayesinde her bir drone’un sorumlu olduğu bölgeler açıkça ayrıştırılmakta ve görev paylaşımı görsel olarak net biçimde izlenebilmektedir.



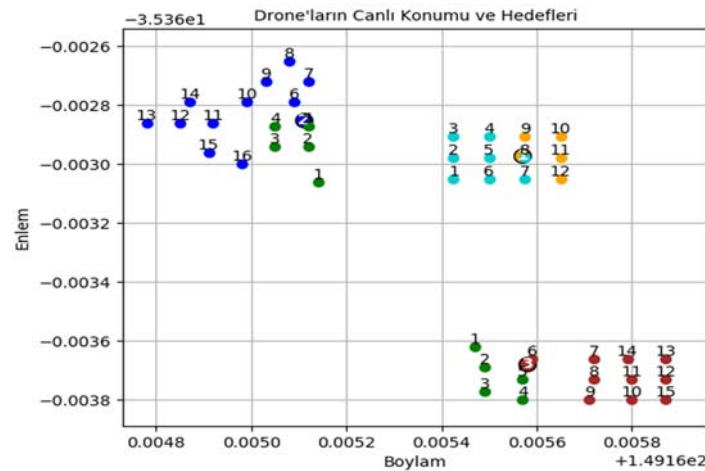
Şekil 4.4 Tarama sonrası kümeleme ile oluşturulan ilaçlama noktalarının drone’lara göre renklerle ayrıştırılması.

İlaçlama süreci ilerledikçe drone'ların sorumluluğunda bulunan hedef noktalar üzerindeki işlem tamamlandığında bu noktaların rengi değiştirilerek görselleştirilmiştir. Şekil 4.5'te görüldüğü üzere, ilaçlanan noktalar yeşil renge dönüşmekte ve böylece hangi hedeflerin başarıyla tamamlandığı kolaylıkla takip edilebilmektedir. Bu yaklaşım sayesinde hem görev ilerleyişi anlık olarak izlenebilmekte hem de operasyon sonunda tüm hedeflerin durumuna ilişkin net bir görünüm elde edilmektedir.



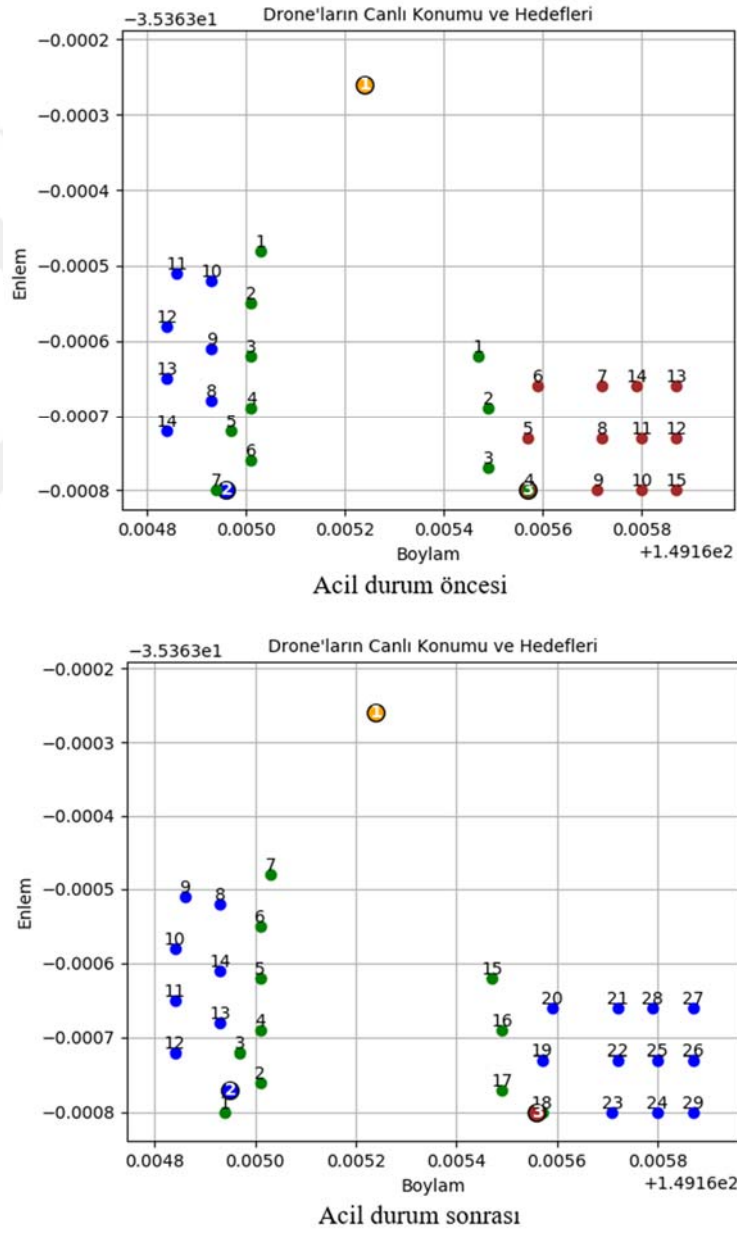
Şekil 4.5 İlaçlama işlemi tamamlanan noktaların yeşil renge dönerek gösterilmesi.

Tasarlanan sistemde arazilerin ilaçlaması yapılırken yeni bir arazinin taraması yapılabilmektedir. Bu sayede sürü drone sisteminin tam anlamıyla kullanımı gerçekleştirilmiş olmaktadır. Şekil 4.6'da kâşif drone ile tarama, drone 2 ve drone 3 ile ise ilaçlama yapığını gösteren harita verilmiştir.



Şekil 4.6 Sürü drone sisteminde hem taramanın hem de ilaçlamanın aynı anda yapılması.

Uçuş esnasında meydana gelen acil durum senaryolarında, görev devri ve hedeflerin yeniden atanması harita üzerinde karşılaştırmalı olarak gösterilmiştir. Şekil 4.7’de acil durum öncesi ve sonrası görev dağılımı yer almaktadır. Görsellerden de görüldüğü üzere, arızalanan (SOC değeri kritik eşiğin altına düşen veya ilacı biten) drone 3’ün sorumluluğunda bulunan hedefler sistem tarafından drone 2’ye aktarılmış ve bu hedefler ilgili drone’un rengine göre yeniden işaretlenmiştir. Böylece görev sürekliliği korunmuş, tüm hedeflerin tamamlanması sağlanmıştır.



Şekil 4.7 Acil durumda görev dağılımı.

4.3 Zirai Mücadele için Tasarlanan Simülasyon Ortamı Sonuçları

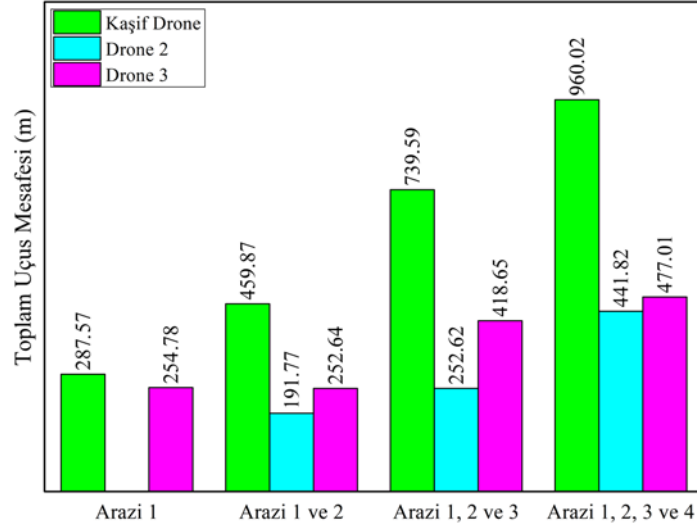
Bu bölümde geliştirilen sistemin performansı, Gazebo tabanlı simülasyon ortamında gerçekleştirilen test senaryoları üzerinden değerlendirilmiştir. Farklı arazi türlerinde (tekli, ikili, üçlü ve dördü arazi) görev senaryoları test edilmiş; toplam uçuş mesafesi, batarya SOC değeri, havada uçuş süresi ve görev başarı oranı gibi temel metrikler ölçülmüştür. Ayrıca tek bir drone ile gerçekleştirilen görevler ile sürü yaklaşımı karşılaştırılmış, böylece önerilen sistemin verimlilik ve hata toleransı açısından sağladığı katkılar sayısal olarak ortaya konmuştur. Çizelge 4.2’de optimizasyon algoritmaları pasif iken elde edilen sonuçlar sunulmaktadır. Bu değerler, sistemin iyileştirme mekanizmaları devreye girmeden önceki temel performansını ortaya koymaktadır.

Çizelge 4.2 Optimizasyon algoritması pasif iken elde edilen sonuçlar.

		Arazi Türleri			
		Arazi 1	Arazi 1 ve 2	Arazi 1, 2 ve 3	Arazi 1, 2, 3 ve 4
Drone Sayısı		2	3	3	3
Toplam	D1	287.57	459.87	739.59	960.02
Uçuş	D2	-	191.77	252.62	441.82
Mesafesi (m)	D3	254.78	252.64	418.65	477.01
Havada	D1	175.78	270.99	386.68	491.39
Kalma	D2	-	166.67	221.01	323.13
Süresi (s)	D3	199.34	205.85	323.45	323.13
Batarya	D1	87	80	81	63
SOC Değeri	D2	-	89	83	76
(%)	D3	85	85	75	76
Görev Başarı Oranı		15/15	29/29	46/46	60/62

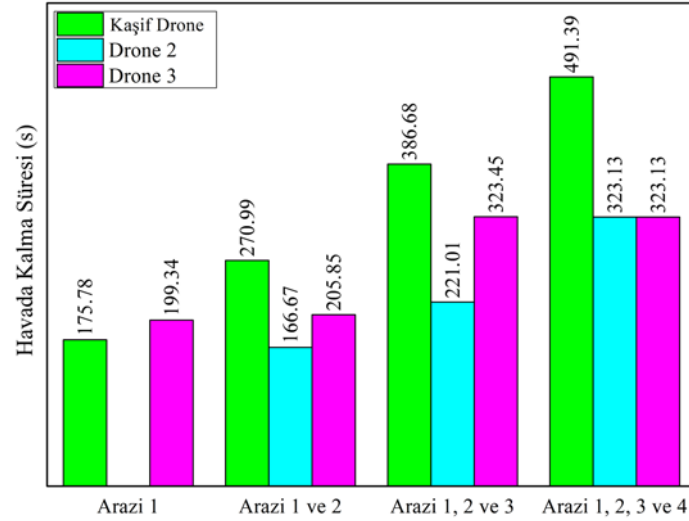
D1: Kâşif drone, D2: Drone 2, D3: Drone 3

Tekli arazi senaryosunda kâşif drone yaklaşık 287.57 m uçarken ilaçlama görevini tek başına üstlenen drone 3 yaklaşık 255 m uçuş gerçekleştirmiştir. Bu senaryoda tek arazi olduğu için drone 2 görevlendirilmemiştir. Arazi sayısı arttıkça kâşif drone’un tarama yükü beklendiği üzere yükselmiş ve dördü arazi senaryosunda 960.02 m’ye kadar çıkmıştır. Buna karşın ilaçlama drone’ları sürü içinde görev paylaşımı yaptığından, drone 2 yaklaşık 441 m ve drone 3 yaklaşık 477 m düzeylerinde kalarak mesafelerini sınırlı tutabilmiştir. Şekil 4.8’de drone’ların uçuş mesafeleri gösterilmiştir.



Şekil 4.8 Arazi türlerine göre drone'ların uçuş mesafeleri.

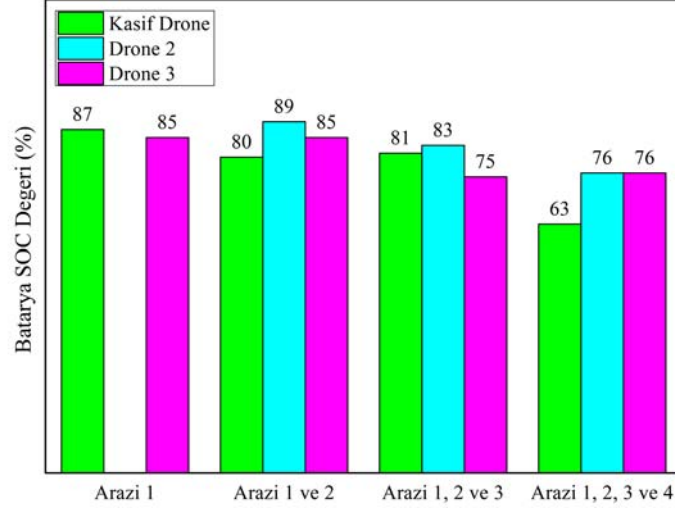
Havada kalma süreleri de uçuş mesafeleri ile benzer eğilimdedir. Dört arazili senaryoda kâşif drone yaklaşık 491 s havada kalırken ilaçlama drone'larının görevleri yaklaşık 323 s seviyesinde tamamlanmıştır. Böylelikle tarama yükü artsa da ilaçlama yükünün sürü sayesinde dengelendiğini görülmektedir. Drone'ların havada kalma süreleri Şekil 4.9'da verilmiştir.



Şekil 4.9 Arazi türlerine göre drone'ların havada kalma süreleri.

Drone'ların batarya SOC değerleri ise havada kalma süreleri ve uçuş mesafesi ile ters orantılı olarak hareket etmektedir. Örneğin dörtlü arazide kâşif drone %73 batarya SOC değerine sahip olurken ilaçlama drone'ları %76'da kalmıştır. Bu iş yükü dengesi de sürü

robotiğinin avantajını göstermektedir. Şekil 4.10'da arazi sayısına göre batarya SOC değerleri verilmiştir.



Şekil 4.10 Arazi türlerine göre drone'ların batarya SOC değerleri.

Yapılan test işlemleri sonucunda, kâşif drone'un tarama yükü arazi ile orantılı artarken ilaçlama drone'larının paralel çalışmayla uçuş mesafesi ve sürelerini makul bir aralıkta tuttuğunu ortaya koymaktadır. Bu sayede toplam görev süresi tek drone yaklaşımına kıyasla belirgin biçimde kısalmakta; sistem görev sürekliliği ve operasyonel verimlilik bakımından avantaj sunmaktadır.

4.4 Nesne Tespit Performansı

Bu bölümde, önerilen sistemin nesne tespit performansı YOLO mimarileri kullanılarak değerlendirilmiştir. YOLOv3 mimarisinden YOLOv3-Tiny, YOLOv3, YOLOv3-SPP ve YOLOv5 mimarisinden YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x varyantları Ultralytics altyapısı kullanılarak eğitilmiş ve karşılaştırmalar bu eğitimler üzerinden gerçekleştirilmiştir. Modellerin seçiminde, hız/doğruluk dengesi sunmaları ve literatürde en yaygın kullanılan nesne tespit mimarileri arasında yer almaları etkili olmuştur. Performans değerlendirmesi için Precision, Recall, F1 skoru, mAP@0.5 ve mAP@0.5:0.95 metrikleri kullanılmıştır. Böylece farklı boyut ve karmaşıklıkta YOLO varyantların nesne tespit performansları karşılaştırılmıştır.

Bununla birlikte, entegrasyon açısından önemli bir fark bulunmaktadır. YOLOv3 modellerinde DarkNet tabanlı sürümler kullanılarak ROS/Gazebo simülasyon ortamına dönüştürülebilmüş ve gerçek zamanlı çalıştırılabilmektedir. Ancak YOLOv5 modellerinde aynı entegrasyon mümkün olmamıştır. Bunun temel sebebi, çalışmalarda kullanılan Ubuntu 18.04 işletim sistemi, ROS Melodic altyapısı ve Python 3.6.9 sürümüdür. YOLOv5 mimarisi için Python 3.6.9'dan daha yeni bir Python sürümü gerektirdiğinden, mevcut sistemle uyumlu çalışmamaktadır. Python sürümünün yükseltilmesi ise yalnızca yazılım güncellemesi ile çözülememekte, ROS sürümünün (Noetic veya ROS 2) ve işletim sistemi altyapısının da tamamen değiştirilmesini gerektirmektedir. Bu nedenle çalışmada YOLOv5 versiyonları yalnızca çevrimdışı ortamda test edilebilmiş, pratik entegrasyonları ise mevcut sistemin yazılım gereksinimleri nedeniyle sınırlı kalmıştır. Yapılan çalışmada kullanılan YOLO modellerinin eğitimlerinin sonuçları Çizelge 4.3'te verilmiştir.

Çizelge 4.3 Kullanılan YOLO modellerinin eğitim sonuçları.

Model	Precision	Recall	F1 Skoru	mAP @0.5	mAP @0.5:0.95	Parametre Sayısı	GFLOPS
YOLOv3-Tiny	0.993	0.990	0.991	0.995	0.651	8.66M	12.9
YOLOv3	0.991	0.991	0.991	0.995	0.654	61.5M	154.5
YOLOv3-SPP	0.991	0.990	0.990	0.995	0.655	62.5M	155.4
YOLOv5-n	0.988	0.99	0.988	0.995	0.650	1.76M	4.1
YOLOv5-s	0.991	0.991	0.991	0.995	0.655	7.01M	15.8
YOLOv5-m	0.985	0.993	0.988	0.995	0.648	20.85M	47.9
YOLOv5-l	0.987	0.989	0.987	0.995	0.653	46.10M	107.6
YOLOv5-x	0.991	0.994	0.992	0.995	0.655	86.17M	203.8

M: Milyon

Sonuçlar incelendiğinde, YOLOv3-Tiny, yaklaşık 8.6 milyon parametre ve 12.9 GFLOPs değeri ile oldukça hafif bir yapıya sahiptir. Bu sayede düşük donanım gereksinimleriyle hızlı çalışabilmektedir. Ancak daha büyük varyantlara göre doğruluk sınırlı kalmaktadır. YOLOv3 ve YOLOv3-SPP varyantları YOLOv3-Tiny'ye kıyasla daha yüksek doğruluk sunmalarına karşın 60 milyonun üzerinde parametre ve 150+ GFLOPs düzeyindeki hesaplama yükü nedeniyle donanım ihtiyacı oldukça yüksektir.

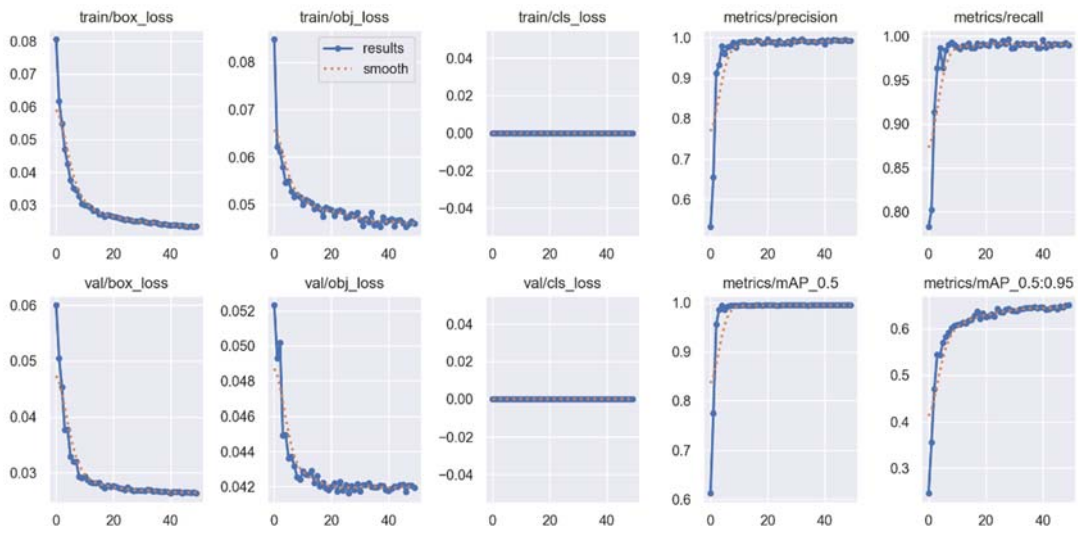
YOLOv5n modeli, yalnızca 1.7 milyon parametre ve 4.1 GFLOPs değeri ile en hafif mimari olup gömülü sistemler için oldukça uygun bir alternatiftir. Ancak daha büyük YOLOv5 modellerine kıyasla doğruluk oranı nispeten daha düşüktür. YOLOv5s, 7 milyon parametre ve 15.8 GFLOPs değeri ile hız ve doğruluk arasında başarılı bir denge kurmaktadır. YOLOv5m varyantı ise doğruluk açısından daha yüksek performans sunarken 21 milyon parametre ve 47.9 GFLOPs ile donanım gereksinimi artmaktadır. YOLOv5l, daha geniş ağ yapısı sayesinde yüksek doğruluk değerlerine ulaşmaktadır. Ancak 46 milyon parametre ve 107 GFLOPs hesaplama maliyeti sebebiyle güçlü GPU donanımı olmadan çalıştırılması zordur. YOLOv5x varyantı ise en yüksek doğruluk değerlerini sunmakta, özellikle Precision ve Recall metriklerinde üstün performans göstermektedir. Bununla birlikte, 86 milyon parametre ve 204 GFLOPs ile hesaplama maliyeti en yüksek modeldir ve bu nedenle gömülü sistemlerde gerçek zamanlı kullanım için uygun değildir.

YOLOv3 ailesi değerlendirildiğinde, düşük donanım gereksinimi ile gömülü sistemlerde gerçek zamanlı çalıştırmaya uygun olan YOLOv3-Tiny varyantı hız açısından avantajlıdır. Ancak doğruluk değerleri sınırlı kaldığından, güçlü donanım imkânı bulunan senaryolarda daha yüksek doğruluk sağlayan YOLOv3-SPP varyantının tercih edilmesi daha doğru olacaktır.

YOLOv5 ailesi ise farklı varyantlarıyla hız ve doğruluk arasında daha esnek bir seçim imkânı sunmaktadır. YOLOv5n ve YOLOv5s varyantları düşük parametre ve GFLOPs değerleri sayesinde gömülü sistemlerde ve donanım kısıtlı ortamlarda kullanılmaya elverişlidir. Buna karşılık YOLOv5m ve YOLOv5l, doğruluk değerlerini artırırken daha güçlü donanım ihtiyacı doğurmaktadır. En yüksek doğruluk değerlerine ulaşan YOLOv5x varyantı ise yüksek hesaplama maliyeti nedeniyle yalnızca güçlü GPU altyapısına sahip sistemlerde tercih edilebilecek bir modeldir. Dolayısıyla pratik uygulamalarda YOLOv5n ve YOLOv5s, hız ve hafiflik avantajları ile öne çıkarken; doğruluk öncelikli ve yüksek işlem gücü gerektiren senaryolarda YOLOv5l ve YOLOv5x daha uygun bir seçim olmaktadır.

Genel olarak değerlendirildiğinde, YOLOv5 mimarisi, YOLOv3'e kıyasla daha düşük parametre ve hesaplama maliyetiyle daha yüksek doğruluk değerleri elde etmiştir. Özellikle YOLOv5n ve YOLOv5s varyantları, hız ve hafiflik avantajları sayesinde gömülü sistemlerde kullanım için öne çıkarken YOLOv5l ve YOLOv5x varyantları yüksek doğruluk gerektiren ancak güçlü donanım altyapısına sahip senaryolar için daha uygun modellerdir. Bu sonuçlar, YOLOv5 ailesinin genel olarak hız/doğruluk dengesi ve donanım verimliliği açısından YOLOv3'e üstünlük sağladığını ortaya koymaktadır.

Elde edilen metriklere ek olarak, bütün modellerin eğitim sürecinde kayıp grafikleri ve Precision – Recall eğrileri görselleştirilmiştir. Sonuçlardan YOLOv3-Tiny modelinin çıktıları Şekil 4.11'de verilmiş olup diğer modellerin grafikleri EK – 1'de verilmiştir. Bu grafikler, her modelin eğitim sürecindeki yakınsama eğilimini ve doğruluk-duyarlılık dengesini görsel olarak ortaya koymaktadır.



Şekil 4.11 YOLOv3-Tiny modelinin eğitim sürecine ait kayıp grafikleri ve Precision-Recall eğrileri

Bu bulgular, nesne tespit performansında YOLOv5 ailesinin, özellikle gömülü sistemler için YOLOv5n ve YOLOv5s varyantlarıyla öne çıktığını, yüksek doğruluk gerektiren senaryolarda ise YOLOv5l ve YOLOv5x'in tercih edilebileceğini göstermektedir.

4.5 Görev Planlama ve Optimizasyon

Bu bölümde görev planlamasında uygulanan optimizasyon algoritmalarının etkisi incelenmiştir. Drone'ların hedef noktalara atanmasının ardından, her bir drone'un izleyeceği rotanın belirlenmesi toplam görev süresi, uçuş mesafesi ve enerji tüketimi açısından kritik öneme sahiptir. Bu amaçla, rota planlama aşamasında NN ve ACO algoritmaları uygulanmış ve elde edilen performans metrikleri karşılaştırılmıştır. Optimizasyon yapılmayan referans senaryonun sonuçları Çizelge 4.2'de verilmişti. Bu süreçte drone'ların uçuş rotaları, hedeflerin atanma sırasına göre belirlenmiş ve herhangi bir rota iyileştirmesi yapılmamıştır. Böylelikle görev sürelerinin ve toplam uçuş mesafelerinin daha yüksek olduğu, dolayısıyla batarya SOC değerinin azaldığı görülmektedir. Bu sonuçlar, optimizasyon algoritmalarının gerekliliğini göstermektedir. NN algoritması aktif iken yapılan rota planlamasına ait sonuçlar Çizelge 4.4'te verilmiştir.

Çizelge 4.4 NN algoritması aktif iken elde edilen sonuçlar.

		Arazi Türleri			
		Arazi 1	Arazi 1 ve 2	Arazi 1, 2 ve 3	Arazi 1, 2, 3 ve 4
Drone Sayısı		2	3	3	3
Toplam Uçuş	D1	292.97	445.12	695.45	865.82
Mesafesi	D2	-	203.96	293.84	418.22
(m)	D3	262.66	250.59	426.52	420.03
Havada	D1	173.39	265.16	373.95	470.34
Kalma Süresi	D2	-	172.03	219.01	319.13
(s)	D3	198.94	192.25	319.25	319.13
Batarya SOC	D1	87	81	71	65
değeri	D2	-	87	83	76
(%)	D3	85	86	75	76
Optimizasyon	D1	45	135	293	491
Süresi	D2	-	105	170	380
(ns)	D3	38	118	250	365
Görev Başarı Oranı		15/15	29/29	46/46	60/62

D1: Kâşif drone, D2: Drone 2, D3: Drone 3

NN algoritmasıyla gerçekleştirilen rota planlamasında, her drone en yakınındaki hedef noktayı takip ederek görevini tamamlamıştır. Bu yaklaşım basit ve hızlı bir çözüm üretse de küresel optimum çözümü garanti etmemektedir. Sonuçlara bakıldığında, optimizasyonsuz senaryoya kıyasla uçuş sürelerinde ve toplam mesafede belirgin bir azalma elde edilmiştir.

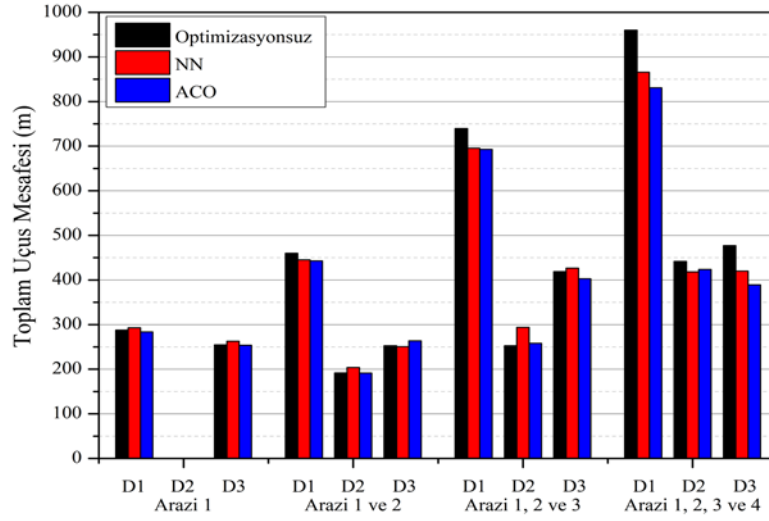
ACO algoritması ile yapılan en iyi rota optimizasyonunda elde edilen sonuçlar Çizelge 4.5'te sunulmuştur. ACO algoritmasıyla yapılan görev planlamasında, karınca kolonilerinden esinlenen sezgisel yaklaşım sayesinde daha dengeli rotalar oluşturulmuştur. NN'ye kıyasla daha yüksek hesaplama maliyeti getirmesine rağmen, toplam uçuş mesafesi ve görev süresinde daha iyi sonuçlar elde edilmiştir. Ayrıca batarya SOC değerinin daha dengeli dağılması ve görev başarı oranının daha yüksek seviyelere ulaşması ACO algoritmasının avantajları arasında yer almaktadır.

Çizelge 4.5 ACO optimizasyon algoritması aktif iken elde edilen sonuçlar.

		Arazi Türleri			
		Arazi 1	Arazi 1 ve 2	Arazi 1, 2 ve 3	Arazi 1, 2, 3 ve 4
Drone Sayısı		2	3	3	3
Toplam Uçuş	D1	283.68	442.74	692.79	830.83
Mesafesi	D2	-	191.56	258.03	423.78
(m)	D3	253.89	263.6	402.8	389.18
Havada	D1	176.46	245.71	378.36	447.28
Kalma Süresi	D2	-	166.14	201.23	304.99
(s)	D3	188.35	193.04	283.11	300.75
Batarya SOC	D1	88	82	71	66
Değeri	D2	-	88	85	77
(%)	D3	87	85	79	78
Optimizasyon	D1	10.08	34.56	89.47	123.14
Süresi	D2	-	6.66	11.13	28.13
(s)	D3	7.3	8.32	23.93	28.13
Görev Başarı Oranı		15/15	29/29	46/46	60/62

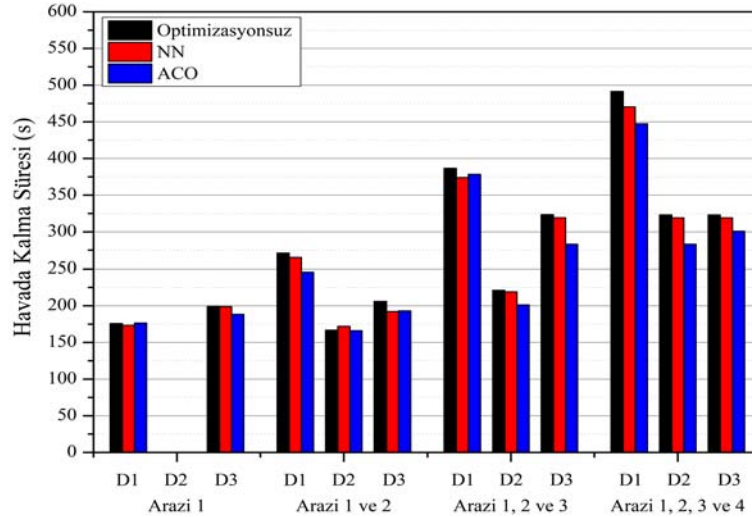
D1: Kâşif drone, D2: Drone 2, D3: Drone 3

Çizelge 4.2, Çizelge 4.4 ve Çizelge 4.5'in dört arazili senaryodaki sonuçları incelendiğinde, optimizasyonsuz durumda toplam uçuş mesafesi 960 m iken NN algoritması ile bu değer 865 m'ye, ACO algoritması ile ise 831 m'ye düşmüştür. Böylece NN yaklaşık %10, ACO ise %13.4 oranında mesafe iyileştirmesi sağlamıştır. Şekil 4.12'de arazi türlerine göre drone'ların toplam uçuş mesafeleri verilmiştir.



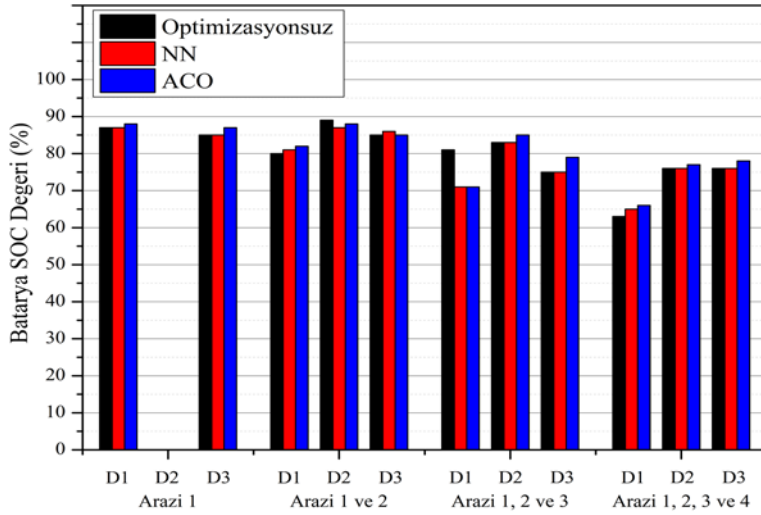
Şekil 4.12 Arazi türlerine göre drone'ların uçuş mesafeleri.

Uçuş mesafesindeki bu azalma havada kalma sürelerine de doğrudan yansımış; optimizasyonsuz durumda D1 için 491 s olan süre NN ile 470 s'ye, ACO ile 447 s'ye kadar gerilemiştir. Bu durum, ACO'nun rotaları daha dengeli dağıtarak enerji verimliliğini artırdığını açıkça göstermektedir. Şekil 4.13'te arazi türlerine göre drone'ların havada kalma süreleri verilmiştir.



Şekil 4.13 Arazi türlerine göre drone'ların havada kalma süreleri.

Aynı şekilde batarya SOC değerlerine de ters orantılı olarak yansımış; optimizasyonsuz durumda %63 olan SOC değeri, NN ile %65, ACO ile ise %66 seviyesine çıkmıştır. Şekil 4.14'te arazi türlerine göre batarya SOC değerleri verilmiştir.



Şekil 4.14 Arazi türlerine göre drone’ların ortalama batarya SOC değerleri.

Bir diğer kriter olan optimizasyon süreleri değerlendirildiğinde, NN algoritmasının ns mertebesinde çözüm ürettiği, ACO’nun ise s ölçeğinde hesaplama yaptığı görülmektedir. Dört arazili senaryoda NN toplamda 400 ns seviyesinde sonuç üretirken ACO yaklaşık 123 s’lik bir işlem süresi gerektirmiştir. Bu nedenle NN gerçek zamanlı anlık yeniden planlama için uygunken ACO daha yüksek hesaplama maliyeti karşılığında enerji verimliliğini ön plana çıkarmaktadır.

Genel olarak, dört arazili senaryo sonuçları göstermektedir ki NN hızlı karar verme kabiliyeti ile operasyonel esneklik sunarken ACO daha dengeli görev dağılımı ile uçuş mesafesi ve batarya SOC değerinde belirgin bir avantaj sağlamaktadır. Bu noktada, hibrit bir çözümün benimsenmesi, yani görev öncesinde ACO ile küresel rota planlamasının yapılması ve görev sırasında NN ile anlık düzeltmelerin gerçekleştirilmesi hem yüksek enerji verimliliği hem de düşük çevrimiçi gecikme açısından en uygun yaklaşım olarak öne çıkmaktadır.

5. TARTIŞMA ve SONUÇ

Bu çalışmada geliştirilen sürü drone sistemi, farklı görev senaryoları altında sürü davranışı, nesne tespiti ve görev planlama performansı açısından kapsamlı biçimde incelenmiştir. Elde edilen sonuçlar, sistemin operasyonel esneklik ve verimlilik bakımından önemli üstünlükler sunduğunu göstermektedir. Literatürde yapılan benzer çalışmaların özeti Çizelge 2.1’de sunulmuştur. Bu tez çalışmasında ise literatürdeki örneklerden farklı olarak sürü yapısı, görev optimizasyonu ve nesne tespiti süreçleri tek bir sisteme entegre edilmiştir. Böylelikle hem operasyonel koordinasyon hem de karar verme süreçleri bakımından daha gelişmiş bir model ortaya konmuştur.

Sürü davranışına yönelik testler, özellikle acil durum senaryolarında sistemin hata toleransını artırdığını göstermiştir. Bir drone’un görev dışı kalması halinde diğer drone’lara görev devrinin hızlı bir şekilde yapılabilmesi, operasyonun sürekliliğini sağlamış; çarpışma önleme modülü ise sürünün güvenli çalışmasına katkı sunmuştur. Sonuç olarak literatürde vurgulanan sürü robotlarının ölçeklenebilirlik ve sağlamlık özelliklerini destekler niteliktedir.

Nesne tespitinde kullanılan YOLO mimarileri değerlendirildiğinde, YOLOv5 modellerinin doğruluk açısından YOLOv3’e kıyasla belirgin üstünlük sağladığı görülmüştür. Özellikle YOLOv5n ve YOLOv5s gibi hafif varyantlar, gömülü sistemlere uygunluklarıyla öne çıkarken; YOLOv5l ve YOLOv5x varyantları yüksek doğruluk sağlamış ancak güçlü donanım gereksinimleri sebebiyle daha sınırlı uygulama alanlarına hitap etmiştir. Bununla birlikte, YOLOv3 modellerinin DarkNet tabanlı sürümleri ROS/Gazebo ortamına entegre edilebilirken YOLOv5 modelleri Ubuntu 18.04 ve ROS Melodic altyapısındaki kısıtlamalar nedeniyle yalnızca çevrimdışı test edilebilmiştir. Bu bulgu, algoritma seçiminin yalnızca doğruluk ve hız kriterleriyle değil aynı zamanda sistem entegrasyonu ile de yakından ilişkili olduğunu göstermektedir. YOLO’nun daha yeni sürümleri (YOLOv6 – YOLOv11) mevcut sistem altyapısının uyumsuzluğu nedeniyle bu çalışmada değerlendirilememiştir.

Rota planlama sonuçları incelendiğinde, optimizasyon yapılmayan senaryolara kıyasla hem NN hem de ACO algoritmalarının görev performansını artırdığı görülmüştür. NN algoritması düşük hesaplama maliyetiyle hızlı çözümler üretirken ACO algoritması daha uzun optimizasyon süreleri karşılığında daha düşük toplam mesafe, süre ve batarya tüketimi sağlamıştır. Özellikle dört arazili senaryoda ACO'nun sağladığı iyileştirme, enerji verimliliği ve dengeli görev dağılımı açısından dikkat çekicidir. Bu noktada hibrit yaklaşımların, yani görev öncesinde ACO ile küresel rota planlaması yapıp görev sırasında NN ile anlık düzeltmelerin uygulanması, gelecekte daha etkin çözümler sunabileceği değerlendirilmektedir. Ayrıca GA, ABC, RL veya deep RL gibi daha gelişmiş optimizasyon tekniklerinin kullanılması da rota planlama ve görev dağılımında hem enerji verimliliğini hem de gerçek zamanlı uygulanabilirliği artırma potansiyeline sahiptir.

Genel olarak, bu çalışmada geliştirilen yöntemler ve elde edilen bulgular, sürü drone sistemlerinin özellikle tarımsal uygulamalarda önemli bir potansiyel sunduğunu göstermektedir. Drone'ların sürü halinde koordineli çalışabilmesi sayesinde geniş ölçekli tarım arazilerinde tarama ve ilaçlama görevleri daha kısa sürede, daha az enerji tüketimiyle ve yüksek doğrulukla gerçekleştirilebilmektedir. Dolayısıyla hem operasyonel süreklilik hem de kaynak verimliliği açısından klasik yöntemlere kıyasla belirgin bir avantaj sağlamaktadır. Ayrıca geliştirilen sistemin hata toleransı yüksek yapısı, acil durum senaryolarında görev devrinin sağlanabilmesi ve çarpışma önleme gibi mekanizmalar, gerçek saha koşullarında da güvenilirlik ve ölçeklenebilirlik açısından kritik katkılar sunmaktadır. Nesne tespiti için kullanılan YOLO mimarilerinin karşılaştırmalı değerlendirmesi ise, farklı donanım koşullarına ve uygulama gereksinimlerine uygun esnek çözümler sunulabileceğini ortaya koymuştur. Tüm bu sonuçlar, sürü drone sistemlerinin yalnızca tarımda değil arama-kurtarma, lojistik ve çevresel izleme gibi geniş bir uygulama yelpazesinde de etkili bir şekilde kullanılabileceğini göstermektedir.

Çalışma bazı kısıtlar içermektedir. Oluşturulan senaryolar Gazebo tabanlı simülasyon ortamında gerçekleştirilmiş olup, gerçek saha koşulları test edilmemiştir. Kullanılan sistem altyapısı (Ubuntu 18.04, ROS Melodic ve Python 3.6.9) nedeniyle YOLOv5

modelleri yalnızca çevrimdışı olarak değerlendirilebilmiş, daha güncel YOLO sürümleri bu çalışmada denenememiştir. Ayrıca ACO algoritmasının hesaplama süreleri s mertebesinde olduğu için gerçek zamanlı görevlerde kullanımında sınırlılık oluşturmaktadır.

İlerleyen çalışmalarda sistemin gerçek zamanlı olarak tarım arazilerinde test edilmesi ve saha koşullarına uygun iyileştirmelerin yapılması planlanmaktadır. Daha güncel yazılım altyapıları (Ubuntu 22.04, ROS 2, Python 3.10+) ile YOLOv6 – YOLOv11 gibi yeni nesil modellerin entegrasyonu mümkün hale getirilebilir. Ayrıca rota planlamada hibrit yaklaşımlar, yani görev öncesinde ACO ile küresel planlama ve görev sırasında NN ile anlık düzeltmelerin birlikte kullanılması, enerji verimliliği ve operasyonel esneklik açısından daha etkin çözümler sunabilecektir. Böylelikle çalışmada oluşan sınırlılıkların da giderileceği düşünülmektedir. Bunun yanında GA, ABC, RL ve DÖ tabanlı optimizasyon yöntemlerinin kullanılmasıyla görev planlama süreçleri daha da geliştirilebilir.

6. KAYNAKLAR

- Al-Amin A A, Lowenberg-DeBoer J, Franklin K, Dickin E, Monaghan J M, Behrendt K, 2025, Autonomous regenerative agriculture: Swarm robotics to change farm economics, *Smart Agricultural Technology*, 11, 101005.
- Albawi S, Bayat O, Al-Azawi S, Ucan O N, 2018, Social touch gesture recognition using convolutional neural network, *Computational Intelligence and Neuroscience*, 2018, 6973103.
- Alqudsi Y, Makaraci M, 2025a, Exploring advancements and emerging trends in robotic swarm coordination and control of swarm flying robots: A review, *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 239, 180-204.
- Alqudsi Y, Makaraci M, 2025b, UAV swarms: research, challenges, and future directions, *Journal of Engineering and Applied Science*, 72, 12.
- Altaş Z, Özgüven M, Yanar Y, 2018, Determination of Sugar Beet Leaf Spot Disease Level (*Cercospora beticola* Sacc.) with Image Processing Technique by Using Drone, *Current Investigations in Agriculture and Current Research*, 5, 621-631.
- Andries V, Zaharia S M, 2025, Generative Design, Simulation, and 3D Printing of the Quadcopter Drone Frame, *Applied Sciences*, 15, 9647.
- Apolo-Apolo O E, Martínez-Guanter J, Egea G, Raja P, Pérez-Ruiz M, 2020, Deep learning techniques for estimation of the yield and size of citrus fruits using a UAV, *European Journal of Agronomy*, 115, 126030.
- Arad B, Balendonck J, Barth R, Ben-Shahar O, Edan Y, Hellström T, vd., 2020, Development of a sweet pepper harvesting robot, *Journal of Field Robotics*, 37, 1027-1039.
- Başaran E, 2022, Image Wavelet Scattering and Densenet Based Pistachio Identification, *Uluslararası Anadolu Ziraat Mühendisliği Bilimleri Dergisi*, 4, 81-87.
- Boccadoro P, Striccoli D, Grieco L A, 2021, An extensive survey on the Internet of Drones, *Ad Hoc Networks*, 122, 102600.

- Boyar T, Yıldız K, 2022, Powdery mildew detection in hazelnut with deep learning, *Hittite Journal of Science and Engineering*, 9, 159-166.
- Carranza-García M, Torres-Mateo J, Lara-Benítez P, García-Gutiérrez J, 2020, On the performance of one-stage and two-stage object detectors in autonomous vehicles using camera data, *Remote Sensing*, 13, 89.
- Catala-Roman P, Segura-Garcia J, Dura E, Navarro-Camba E A, Alcaraz-Calero J M, Garcia-Pineda M, 2024, AI-based autonomous UAV swarm system for weed detection and treatment: Enhancing organic orange orchard efficiency with agriculture 5.0, *Internet of Things*, 28, 101418.
- Cengiz E, Kelek M M, Oğuz Y, Yılmaz C, 2022, Classification of breast cancer with deep learning from noisy images using wavelet transform, *Biomedical Engineering/Biomedizinische Technik*, 67, 143-150.
- Chakchouk M A, El Hami A, Dahoo P R, Lakhlifi A, Gafsi W, Haddar M, 2023, The impact of curved-tip propeller geometry on hovering drone performance for air quality monitoring, *Advances in Mechanical Engineering*, 15, 16878132231206330.
- Chen C J, Huang Y Y, Li Y S, Chen Y C, Chang C Y, Huang Y M, 2021, Identification of fruit tree pests with deep learning on embedded drone to achieve accurate pesticide spraying, *IEEE Access*, 9, 21986-21997.
- Chen X, Hopkins B, Wang H, O'Neill L, Afghah F, Razi A, vd., 2022, Wildland fire detection and monitoring using a drone-collected rgb/ir image dataset, *IEEE Access*, 10, 121301-121317.
- Chu S, Bao W, 2025, Dynamic Agricultural Pest Classification Using Enhanced SAO-CNN and Swarm Intelligence Optimization for UAVs, *International Journal of Cognitive Computing in Engineering*, 6, 588-602.
- Debele Y, Shi H Y, Wondosen A, Kim J H, Kang B S, 2022, Multirotor Unmanned Aerial Vehicle Configuration Optimization Approach for Development of Actuator Fault-Tolerant Structure, *Applied Sciences*, 12, 6781.
- Deng J, Marri S, Klein J, Pałubicki W, Pirk S, Chowdhary G, vd., 2024, Gazebo plants: Simulating plant-robot interaction with cosserat rods, *arXiv preprint*

arXiv:2402.02570.

- Dlamini S M, Ouma Y O, 2025, Large-Scale Topographic Mapping Using RTK-GNSS and Multispectral UAV Drone Photogrammetric Surveys: Comparative Evaluation of Experimental Results, *Geomatics*, 5, 25.
- Dorigo M, Theraulaz G, Trianni V, 2021, Swarm robotics: Past, present, and future [point of view], *Proceedings of the IEEE*, 109, 1152-1165.
- Dutta A, 2021, Robust design of a multirotor aerial vehicle, *Scientific Reports*, 11, 21137.
- Ergün E, 2025, Harnessing deep learning for multi-class weed species identification in agriculture, *Niğde Ömer Halisdemir Üniversitesi Mühendislik Bilimleri Dergisi*, 14, 251-262.
- Feng Q, Zou W, Fan P, Zhang C, Wang X, 2018, Design and test of robotic harvesting system for cherry tomato, *International Journal of Agricultural and Biological Engineering*, 11, 96-100.
- Fernandez R, Montes H, Surdilovic J, Surdilovic D, Gonzalez-De-Santos P, Armada M, 2018, Automatic detection of field-grown cucumbers for robotic harvesting, *IEEE Access*, 6, 35512-35527.
- Firat Z, Ferrante E, Gillet Y, Tuci E, 2020, On self-organised aggregation dynamics in swarms of robots with informed robots, *Neural Computing and Applications*, 32, 13825-13841.
- Ganduri K V, Pathri B P, 2025, Swarm intelligence in action: particle swarm optimization and rendezvous algorithms for swarm robotics, *Journal of Field Robotics*, 42, 1696-1723.
- Guebsi R, Mami S, Chokmani K, 2024, Drones in precision agriculture: A comprehensive review of applications, technologies, and challenges, *Drones*, 8, 686.
- Guo L, Shu X, Zhu Y, 2025, Smart Farming with Swarm Intelligence: Hybrid Algorithm for Optimal Multi Drone Plant Protection Paths, *Pakistan Journal of Agricultural Sciences*, 62, 215-225.
- Hashim H A, Eltoukhy A E, Odry A, 2023, Observer-based controller for VTOL-UAVs tracking using direct vision-aided inertial navigation measurements, *ISA*

transactions, 137, 133-143.

Heng H, Ghazali M H M, Rahiman W, 2024, Exploring the application of ant colony optimization in path planning for Unmanned Surface Vehicles, Ocean Engineering, 311, 118738.

Hiba A, Bauer P, Nagy M, Simonyi E, Kisari A, Kuna G I, vd., 2022, Software-in-the-loop simulation of the forerunner UAV system, IFAC-PapersOnLine, 55, 139-144.

Hunt E R, Jones S, Hauert S, 2019, Testing the limits of pheromone stigmergy in high-density robot swarms, Royal Society open science, 6, 190225.

İnan M, Karci A, 2021, Tarımda ağaç ilaçlamanın drone'larla yapılmasında yeni bir yöntemin geliştirilmesi ve uygulanması, Computer Science, 6, 72-89.

İren Z, Yeşilay R B, 2025, Tarımda Drone Kullanımının Potansiyel Faydaları, Ege Üniversitesi Ulaştırma Yönetimi Araştırmaları Dergisi, 2, 50-64.

Karaçal B, 2022, İnsansız Hava Araçlarında Kümelenme ve Sürü Kontrolü, Karabük Üniversitesi, Lisansüstü Eğitim Enstitüsü, Yüksek Lisans Tezi, 140, Karabük.

Karahanlı G, Taşkın C, 2024, Derin öğrenme yöntemleri kullanılarak ayçiçeği bitkisinin gelişim evrelerinin tespiti, Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 39, 1455-1472.

Katada Y, 2018, Evolutionary design method of probabilistic finite state machine for swarm robots aggregation, Artificial Life and Robotics, 23, 600-608.

Kelek M M, Fidan U, Oğuz Y, 2025, The effect of wavelet transform on the classification performance of different deep learning architectures, Signal, Image and Video Processing, 19, 366.

Kerkech M, Hafiane A, Canals R, 2020, Vine disease detection in UAV multispectral images using optimized image registration and deep learning segmentation approach, Computers and electronics in agriculture, 174, 105446.

Khalidi B, Harrou F, Cherif F, Sun Y, 2019, Flexible and efficient topological approaches for a reliable robots swarm aggregation, IEEE Access, 7, 96372-96383.

Khalidi B, Harrou F, Sun Y, 2025, Collaborative swarm robotics for sustainable

- environment monitoring and exploration: Emerging trends and research progress, *Energy Nexus*, 17, 100365.
- Koubâa A, Allouch A, Alajlan M, Javed Y, Belghith A, Khalgui M, 2019, Micro air vehicle link (mavlink) in a nutshell: A survey, *IEEE Access*, 7, 87658-87680.
- Krizhevsky A, Sutskever I, Hinton G E, 2012, Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, 25, 1-9.
- Lehnert C, English A, McCool C, Tow A W, Perez T, 2017, Autonomous sweet pepper harvesting for protected cropping systems, *IEEE Robotics and Automation Letters*, 2, 872-879.
- Lili W, Bo Z, Jinwei F, Xiaoan H, Shu W, Yashuo L, vd., 2017, Development of a tomato harvesting robot used in greenhouse, *International Journal of Agricultural and Biological Engineering*, 10, 140-149.
- Ling X, Zhao Y, Gong L, Liu C, Wang T, 2019, Dual-arm cooperation and implementing for robotic harvesting tomato using binocular vision, *Robotics and Autonomous Systems*, 114, 134-143.
- Lyu M, Zhao Y, Huang C, Huang H, 2023, Unmanned aerial vehicles for search and rescue: A survey, *Remote Sensing*, 15, 3266.
- Mao S, Li Y, Ma Y, Zhang B, Zhou J, Wang K, 2020, Automatic cucumber recognition algorithm for harvesting robots in the natural environment using deep learning and multi-feature fusion, *Computers and electronics in agriculture*, 170, 105254.
- Marek D, Paszkuta M, Szyguła J, Biernacki P, Domański A, Szczygieł M, vd., 2024, Swarm of drones in a simulation environment—efficiency and adaptation, *Applied Sciences*, 14, 3703.
- Martínez-Clark R, Pliego-Jimenez J, Flores-Resendiz J F, Avilés-Velázquez D, 2023, Optimum k-Nearest neighbors for heading synchronization on a swarm of UAVs under a time-evolving communication network, *Entropy*, 25, 853.
- Mattivi P, Pappalardo S E, Nikolić N, Mandolesi L, Persichetti A, De Marchi M, vd., 2021, Can commercial low-cost drones and open-source GIS technologies be

- suitable for semi-automatic weed mapping for smart farming? A case study in NE Italy, *Remote Sensing*, 13, 1869.
- Mesías-Ruiz G A, Peña J M, de Castro A I, Dorado J, 2025, Drone imagery dataset for early-season weed classification in maize and tomato crops, *Data in Brief*, 58, 111203.
- Mienye I D, Swart T G, 2024, A comprehensive review of deep learning: Architectures, recent advances, and applications, *Information*, 15, 755.
- Mısır O, Gökrem L, 2020, Sürü robotları için esnek ve ölçeklenebilir toplanma davranışı metodu, *Avrupa Bilim ve Teknoloji Dergisi*, 2020, 100-109.
- Mohammed A, Mohammadi M, Sharma S, Sharma U, Rana V S, Kumar M, vd., 2025, Assessment of advancements and applications of robotics, artificial intelligence, and automated technology in the modern food sector, *Applied Food Research*, 5, 101261.
- Mohammed S Y, 2025, Architecture review: Two-stage and one-stage object detection, *Franklin Open*, 12, 100322.
- Mukherjee A, Misra S, Sukrutha A, Raghuwanshi N S, 2020, Distributed aerial processing for IoT-based edge UAV swarms in smart farming, *Computer Networks*, 167, 107038.
- Mundappat Ramachandran M, Fahad Mon B, Hayajneh M, Abu Ali N, Badidi E, 2025, Solar Agro Savior: Smart Agricultural Monitoring Using Drones and Deep Learning Techniques, *Agriculture*, 15, 1656.
- Murat A A, Kiran M S, 2025, A comprehensive review on YOLO versions for object detection, *Engineering Science and Technology, an International Journal*, 70, 102161.
- Muscato G, Prestifilippo M, Abbate N, Rizzuto I, 2005, A prototype of an orange picking robot: past history, the new robot and experimental results, *Industrial Robot: An International Journal*, 32, 128-138.
- Neto R T, Godinho Filho M, 2013, Literature review regarding Ant Colony Optimization applied to scheduling problems: Guidelines for implementation and directions for

- future research, *Engineering applications of artificial intelligence*, 26, 150-161.
- Nguyen L V, 2024, Swarm intelligence-based multi-robotics: a comprehensive review, *AppliedMath*, 4, 1192-1210.
- Nwankpa C, 2018, Activation functions: Comparison of trends in practice and research for deep learning, *arXiv preprint arXiv:1811.03378*.
- Olcay E, Schuhmann F, Lohmann B, 2020, Collective navigation of a multi-robot system in an unknown environment, *Robotics and Autonomous Systems*, 132, 103604.
- Onishi Y, Yoshida T, Kurita H, Fukao T, Arihara H, Iwai A, 2019, An automated fruit harvesting robot by using deep learning, *Robomech Journal*, 6, 1-8.
- Özgüven M M, 2023, Bahçe Bitkileri Yetiştiriciliğinde Kullanılan Dijital Tarım Teknolojileri, *Tarım Makinaları Bilimi Dergisi*, 19, 174-193.
- Özgüven M M, Altaş Z, Güven D, Çam A, 2022, Tarımda drone kullanımı ve geleceği, *Ordu Üniversitesi Bilim ve Teknoloji Dergisi*, 12, 64-83.
- Özgüven M M, Eminoğlu M B, Çolak A, 2024, Tarımda Otonom Araçların Kullanımı, *Tarım Makinaları Bilimi Dergisi*, 20, 218-234.
- Parhizkar M, Di Marzo Serugendo G, Nitschke J, Hellequin L, Wade A, Soldati T, 2020, First-order agent-based models of emergent behaviour of *Dictyostelium discoideum* and their inspiration for swarm robotics: A selection of aggregation phase behaviour with biological illustrations, *Artificial Life and Robotics*, 25, 643-655.
- Pietrantoni L, Favilla M, Fraboni F, Mazzoni E, Morandini S, Benvenuti M, vd., 2024, Integrating collaborative robots in manufacturing, logistics, and agriculture: Expert perspectives on technical, safety, and human factors, *Frontiers in Robotics and AI*, 11, 1342130.
- Pire T, Mujica M, Civera J, Kofman E, 2019, The Rosario dataset: Multisensor data for localization and mapping in agricultural environments, *The International Journal of Robotics Research*, 38, 633-641.
- Pretto A, Aravecchia S, Burgard W, Chebrolu N, Dornhege C, Falck T, vd., 2020, Building an aerial-ground robotics system for precision farming: an adaptable

- solution, *IEEE Robotics & Automation Magazine*, 28, 29-49.
- Pulungan A B, Putra Z Y, Sidiqi A R, Hamdani H, Parigalan K E, 2024, Drone kit-Python for autonomous quadcopter navigation, *JOIV: International Journal on Informatics Visualization*, 8, 1287-1294.
- Qasim N H, Jawad A M, 2024, 5G-enabled UAVs for energy-efficient opportunistic networking, *Heliyon*, 10, e32660.
- Redmon J, Farhadi A, 2018, Yolov3: An incremental improvement, *arXiv preprint arXiv:1804.02767*.
- Rexhaj Y, Kasemi R, Lammer L, 2025, DOPS: Drone Optimized Performance Score for Evaluating Real-Time Tomato Ripeness Detection, *ARW Proceedings*, 25, 43-48.
- Rong J, Fu J, Zhang Z, Yin J, Tan Y, Yuan T, vd., 2022, Development and evaluation of a watermelon-harvesting robot prototype: vision system and end-effector, *Agronomy*, 12, 2836.
- Roque-Claros R E, Flores-Llanos D P, Maquera-Humpiri A R, Sonthi V K, Sengan S, Rangasamy R, 2024, UAV Path Planning Model Leveraging Machine Learning and Swarm Intelligence for Smart Agriculture, *Scalable Computing: Practice and Experience*, 25, 3752-3765.
- Rufino G, Conte C, Basso P, Tirri A E, Donato V, 2024, Mission design and validation of a fixed-wing unmanned aerial vehicle for environmental monitoring, *Drones*, 8, 641.
- Saavedra-Ruiz M, Pinto-Vargas A M, Romero-Cano V, 2021, Monocular visual autonomous landing system for quadcopter drones using software in the loop, *IEEE Aerospace and Electronic Systems Magazine*, 37, 2-16.
- Sadeghi Amjadi A, Raoufi M, Turgut A E, 2022, A self-adaptive landmark-based aggregation method for robot swarms, *Adaptive Behavior*, 30, 223-236.
- Salimi M, 2021, Adaptive and learning-based formation control of swarm robots, University of Colorado, School of Interactive Arts and Technology, Doctor of Philosophy, 232, Denver.
- Sarker I H, 2021, Machine learning: Algorithms, real-world applications and research

- directions, SN computer science, 2, 160.
- Sdoukou E, Milidonis A, Efstathiou K, Voyiatzis I, 2025, A survey of UAV hardware selection, Journal of Engineering and Applied Science, 72, 88.
- Silwal A, Davidson J R, Karkee M, Mo C, Zhang Q, Lewis K, 2017, Design, integration, and field evaluation of a robotic apple harvester, Journal of Field Robotics, 34, 1140-1159.
- Singh R, Ballal K D, Berger M S, Dittmann L, 2022, Overview of drone communication requirements in 5G, Global IoT Summit, 13533, 3-16.
- Stanković L, Mandić D, 2023, Convolutional neural networks demystified: A matched filtering perspective-based tutorial, IEEE Transactions on Systems, Man, and Cybernetics: Systems, 53, 3614-3628.
- Sun N, Qiu Q, Li T, Ru M, Ji C, Feng Q, vd., 2024, GNSS/LiDAR/IMU Fusion Odometry Based on Tightly-Coupled Nonlinear Observer in Orchard, Remote Sensing, 16, 2907.
- Talamali M S, 2020, Simple individual behavioural rules for improving the collective behaviours of robot swarms, University of Sheffield, Department of Computer Science, Doctor of Philosophy, 162, Sheffield.
- Tanigaki K, Fujiura T, Akase A, Imagawa J, 2008, Cherry-harvesting robot, Computers and electronics in agriculture, 63, 65-72.
- Terzi I, Özgüven M M, Yağcı A, 2023, Detection of some grape varieties with deep learning techniques, Turkish Journal of Agriculture – Food Science and Technology, 11, 125-130.
- Udugama B, 2023, Mini bot 3d: A ros based gazebo simulation, arXiv preprint arXiv:2302.06368.
- Unde S S, Kurkute V, Chavan S S, Mohite D D, Harale A A, Chougale A, 2025, The expanding role of multirotor UAVs in precision agriculture with applications AI integration and future prospects, Discover Mechanical Engineering, 4, 1-26.
- Usman B A, Aminu A A, Ridwanullah A, 2024, Real-Time Tomato Leaf Disease Classification Using Efficientnet and UAV, BIMA Journal of Science and

Technology, 8, 227-243.

Xiong Y, Ge Y, Grimstad L, From P J, 2020, An autonomous strawberry-harvesting robot: Design, development, integration, and field evaluation, Journal of Field Robotics, 37, 202-224.

Xiong Y, Peng C, Grimstad L, From P J, Isler V, 2019, Development and field evaluation of a strawberry harvesting robot with a cable-driven gripper, Computers and electronics in agriculture, 157, 392-402.

Xu W, Zhang F, 2021, Fast-lio: A fast, robust lidar-inertial odometry package by tightly-coupled iterated kalman filter, IEEE Robotics and Automation Letters, 6, 3317-3324.

Yamashita R, Nishio M, Do R K G, Togashi K, 2018, Convolutional neural networks: an overview and application in radiology, Insights into imaging, 9, 611-629.

Zion B, Mann M, Levin D, Shilo A, Rubinstein D, Shmulevich I, 2014, Harvest-order planning for a multiarm robotic harvester, Computers and electronics in agriculture, 103, 75-81.

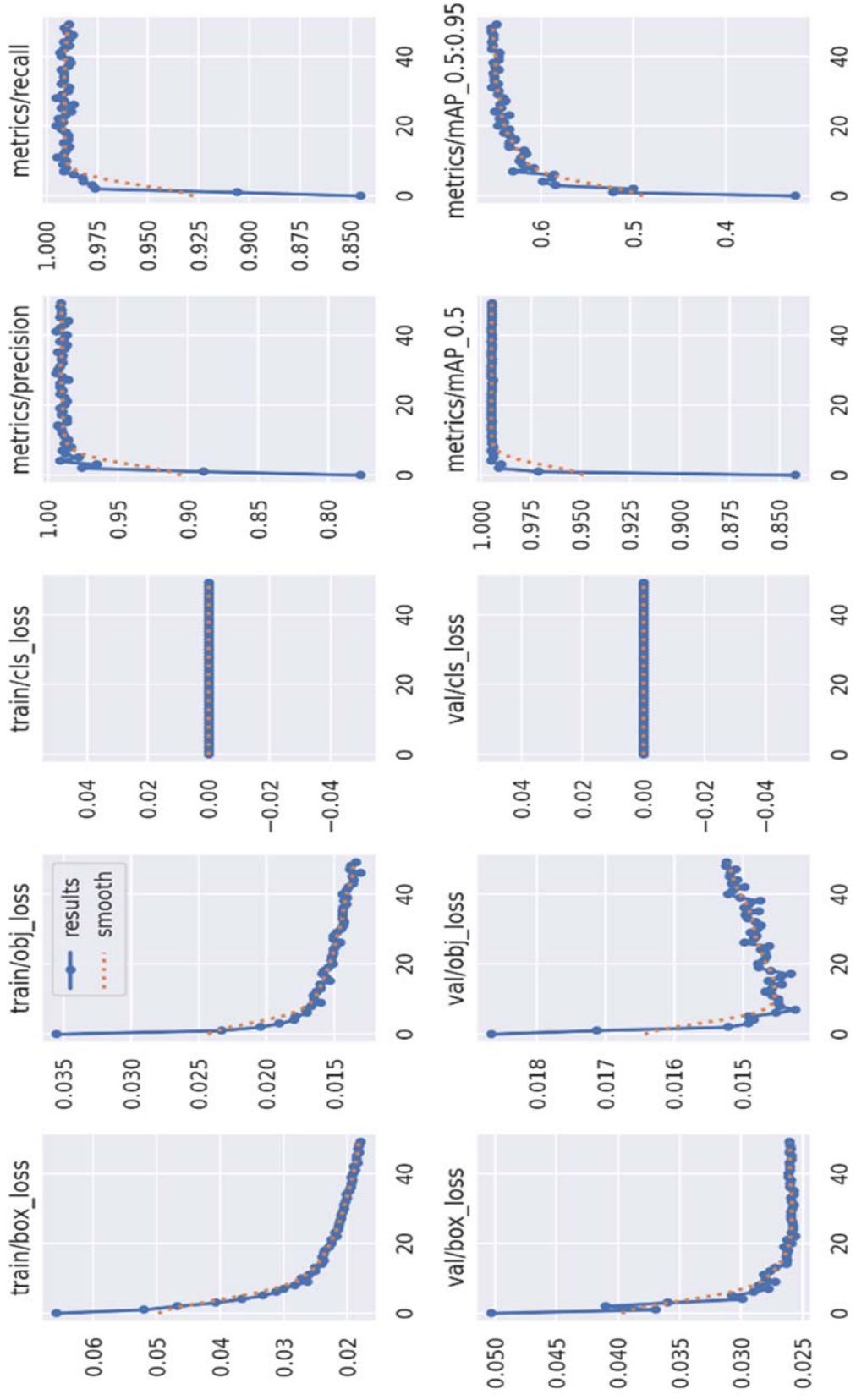
İnternet Kaynakları

1. <https://baykartech.com/tr/uav/bayraktar-tb2/> ,06.11.2025
2. https://en.wikipedia.org/wiki/Deep_Blue_versus_Garry_Kasparov , 06.11.2025
3. <https://ardupilot.org/ardupilot/docs/common-all-vehicle-types.html> , 06.11.2025
4. <https://ardupilot.org/dev/docs/sitl-with-gazebo.html> , 06.11.2025
5. <https://ardupilot.org/dev/docs/sitl-simulator-software-in-the-loop.html> , 06.11.2025
6. <https://mavlink.io/en/> , 06.11.2025
7. <https://sketchfab.com/3d-models/farm-plants-models-mobile-game-ready-lowpoly-9b59e7cf3bcd4feb80ff0cf5fb780055> , 06.11.2025

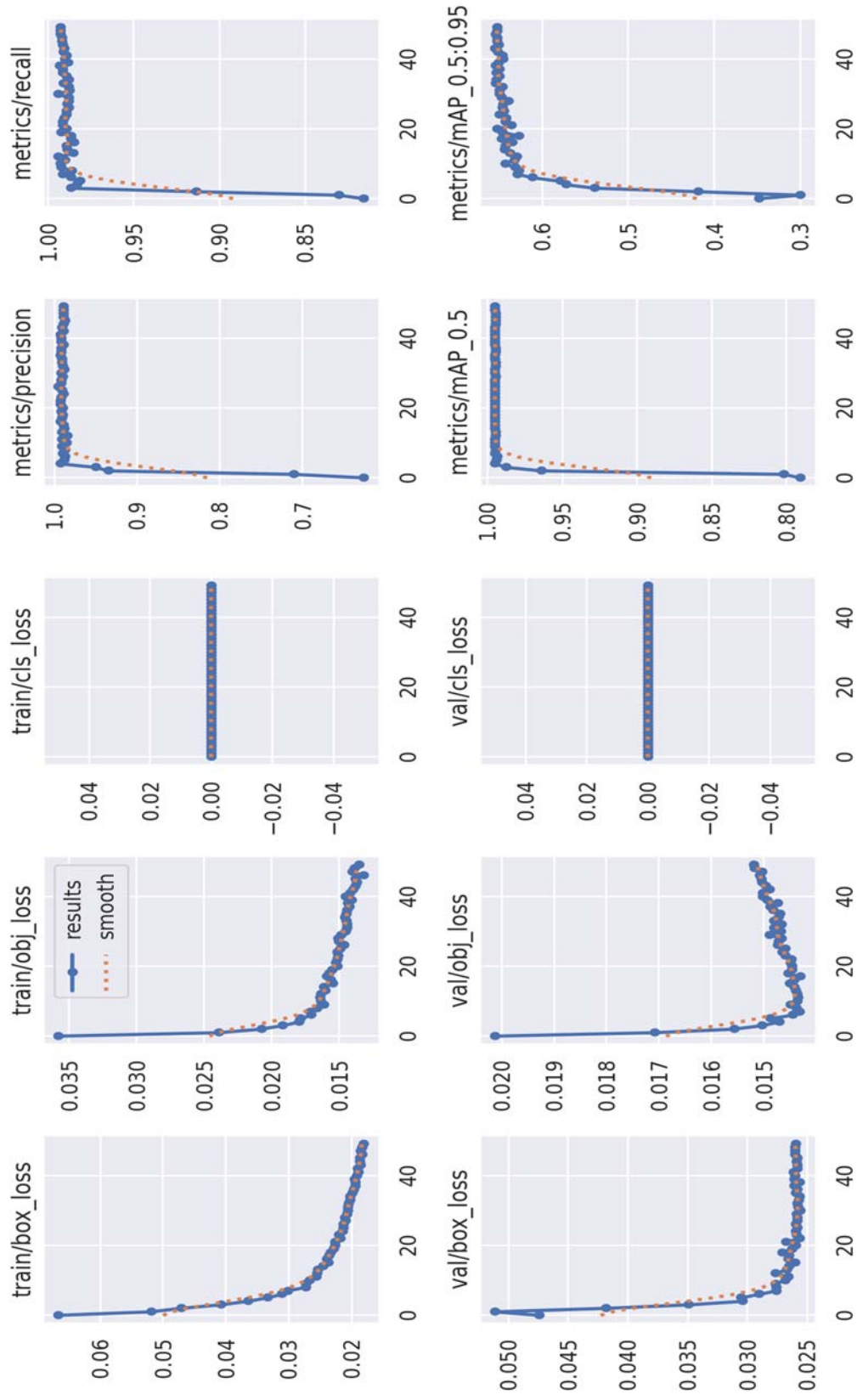
EKLER

EK-1

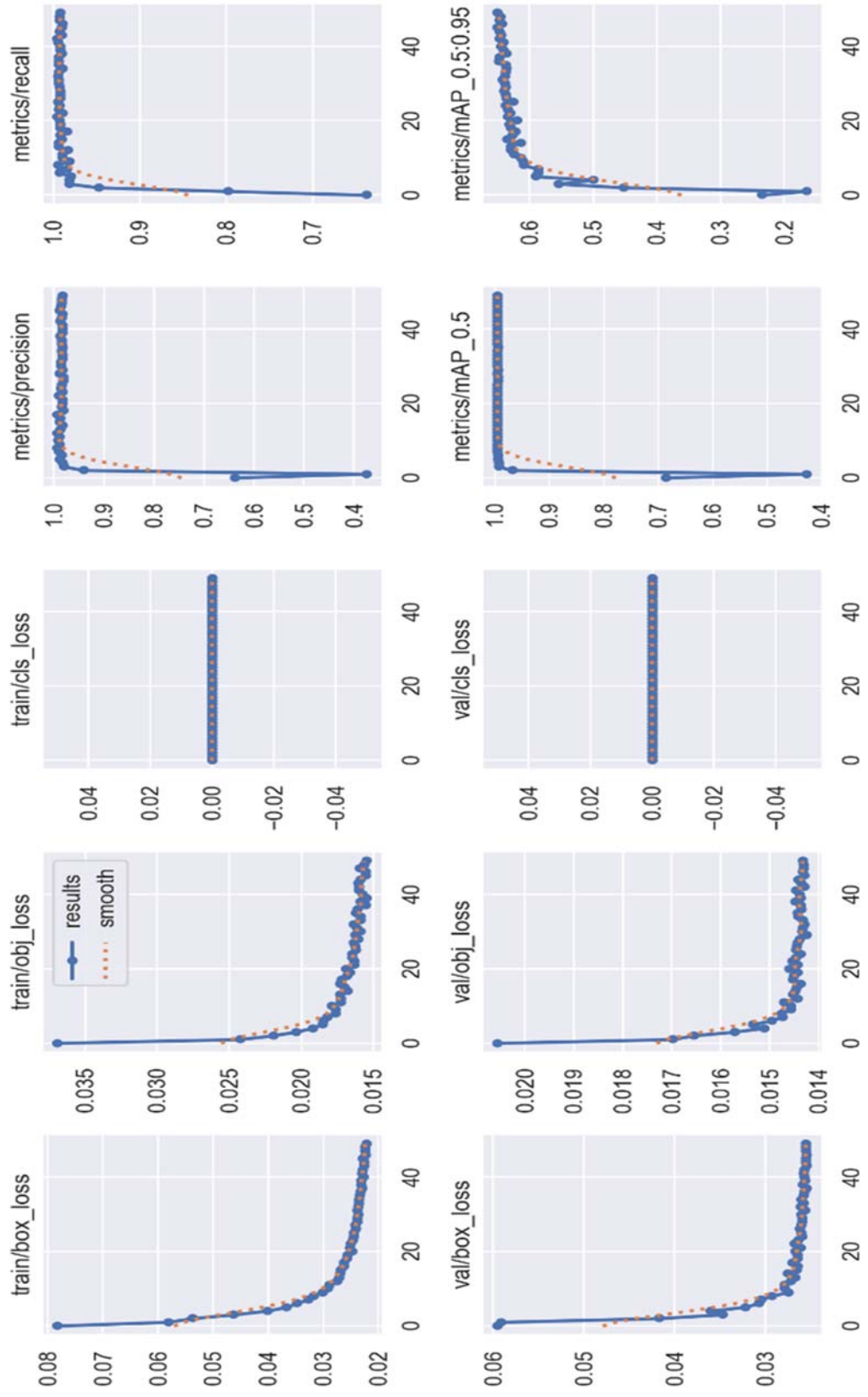
YOLOv3 modelinin eğitim sürecine ait kayıp grafikleri ve Precision–Recall eğrileri.



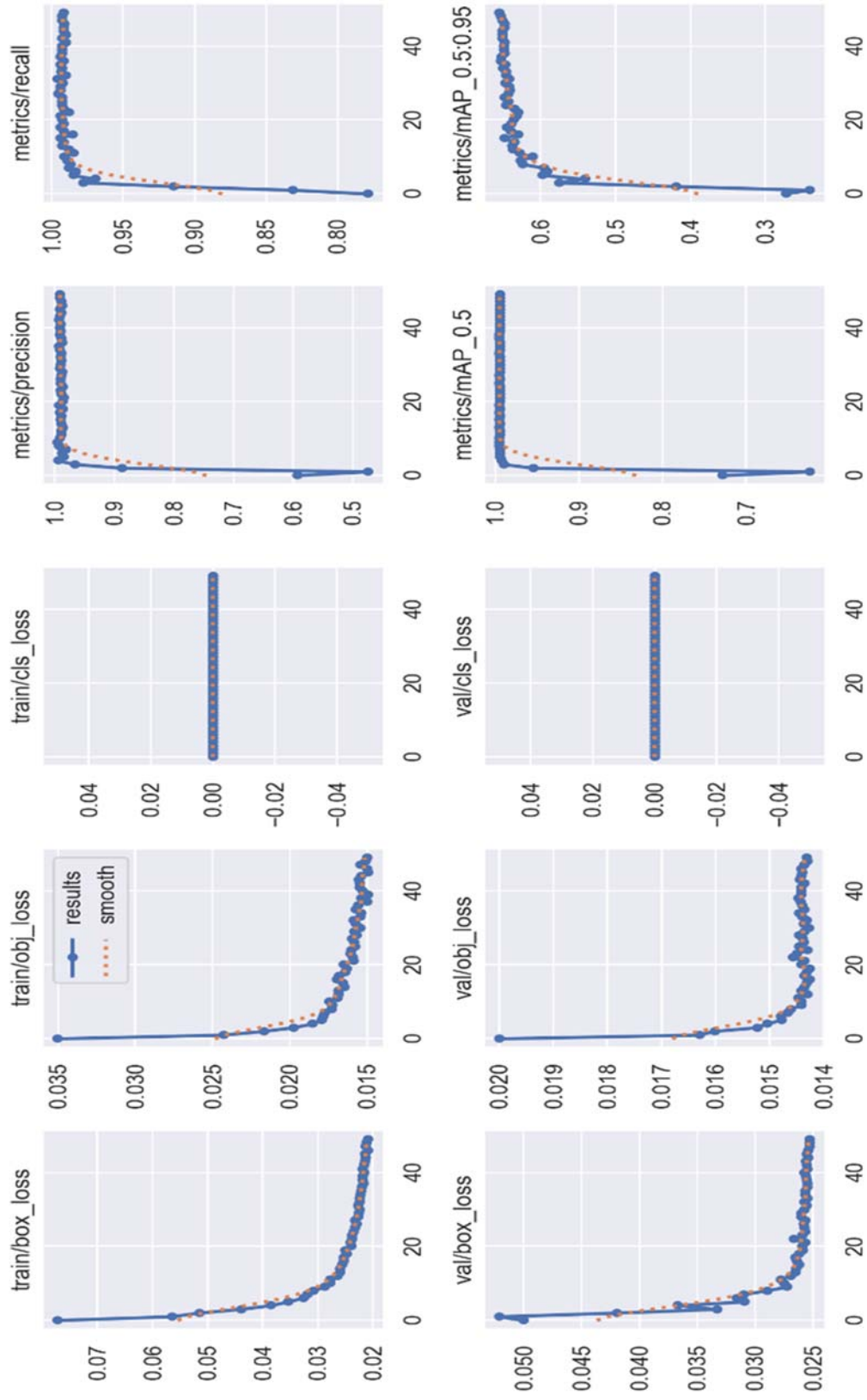
YOLOv3-SPP modelinin eğitim sürecine ait kayıp grafikleri ve Precision-Recall eğrileri.



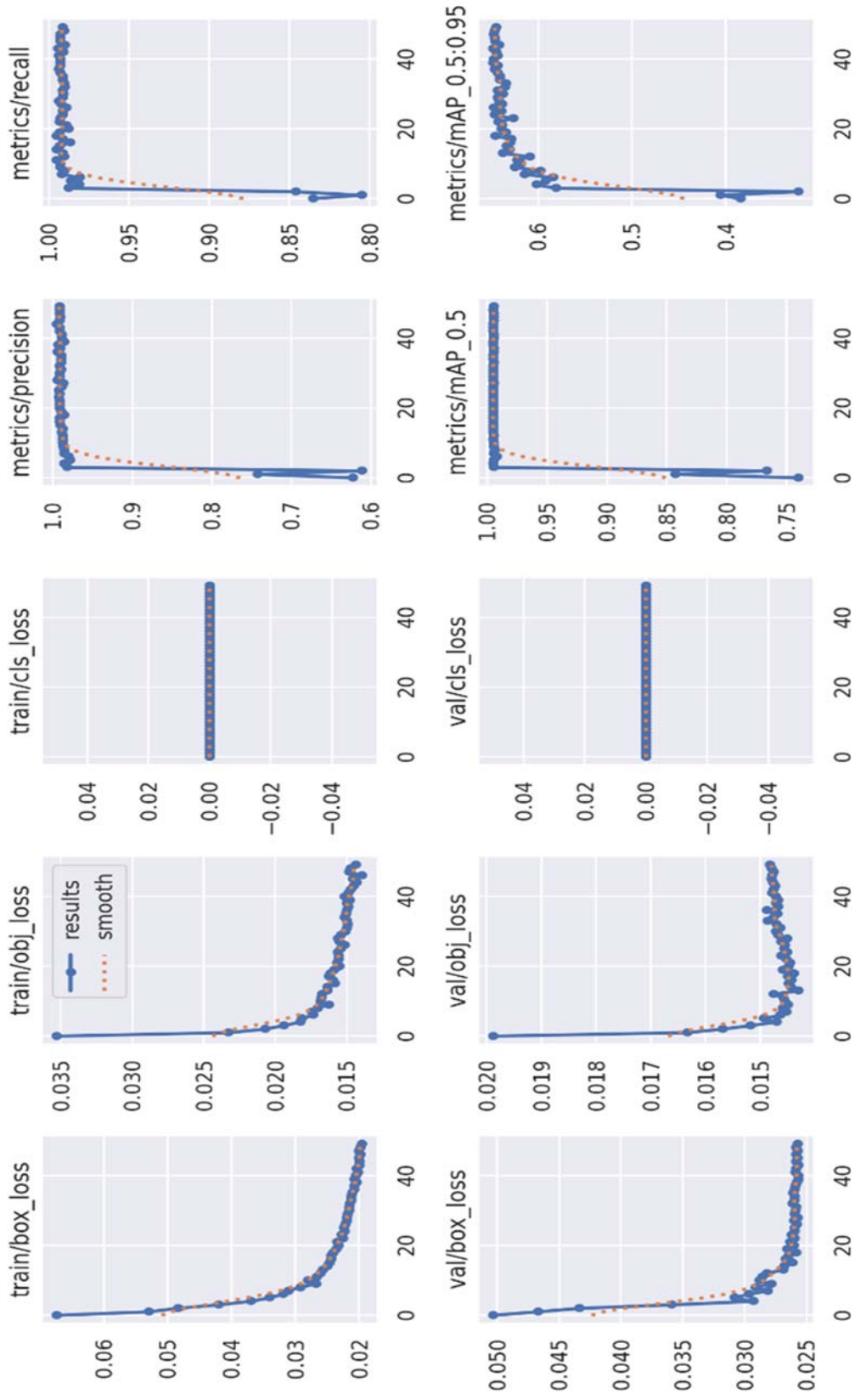
YOLOv5n modelinin eğitim sürecine ait kayıp grafikleri ve Precision–Recall eğrileri.



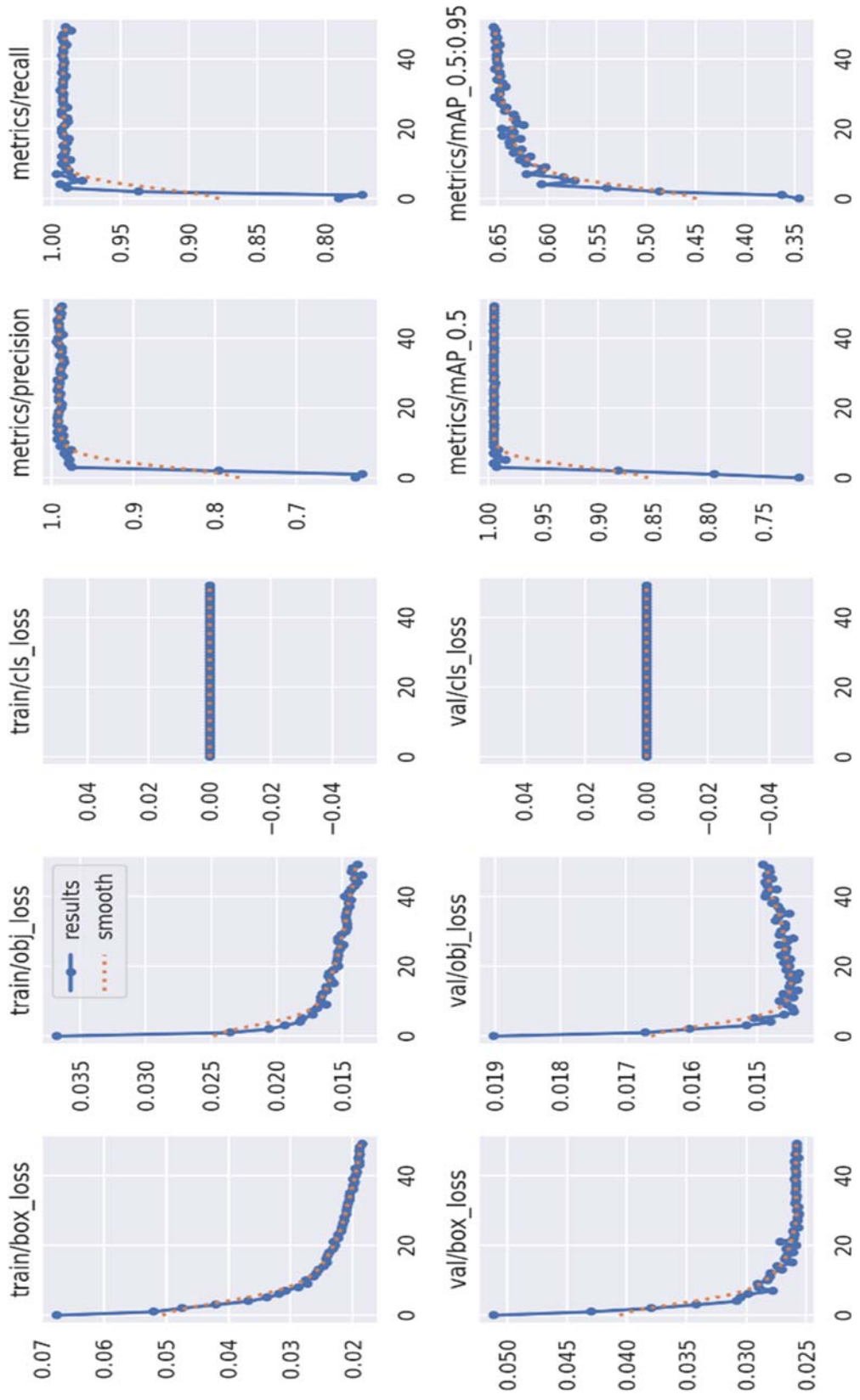
YOLOv5s modelinin eğitim sürecine ait kayıp grafikleri ve Precision–Recall eğrileri.



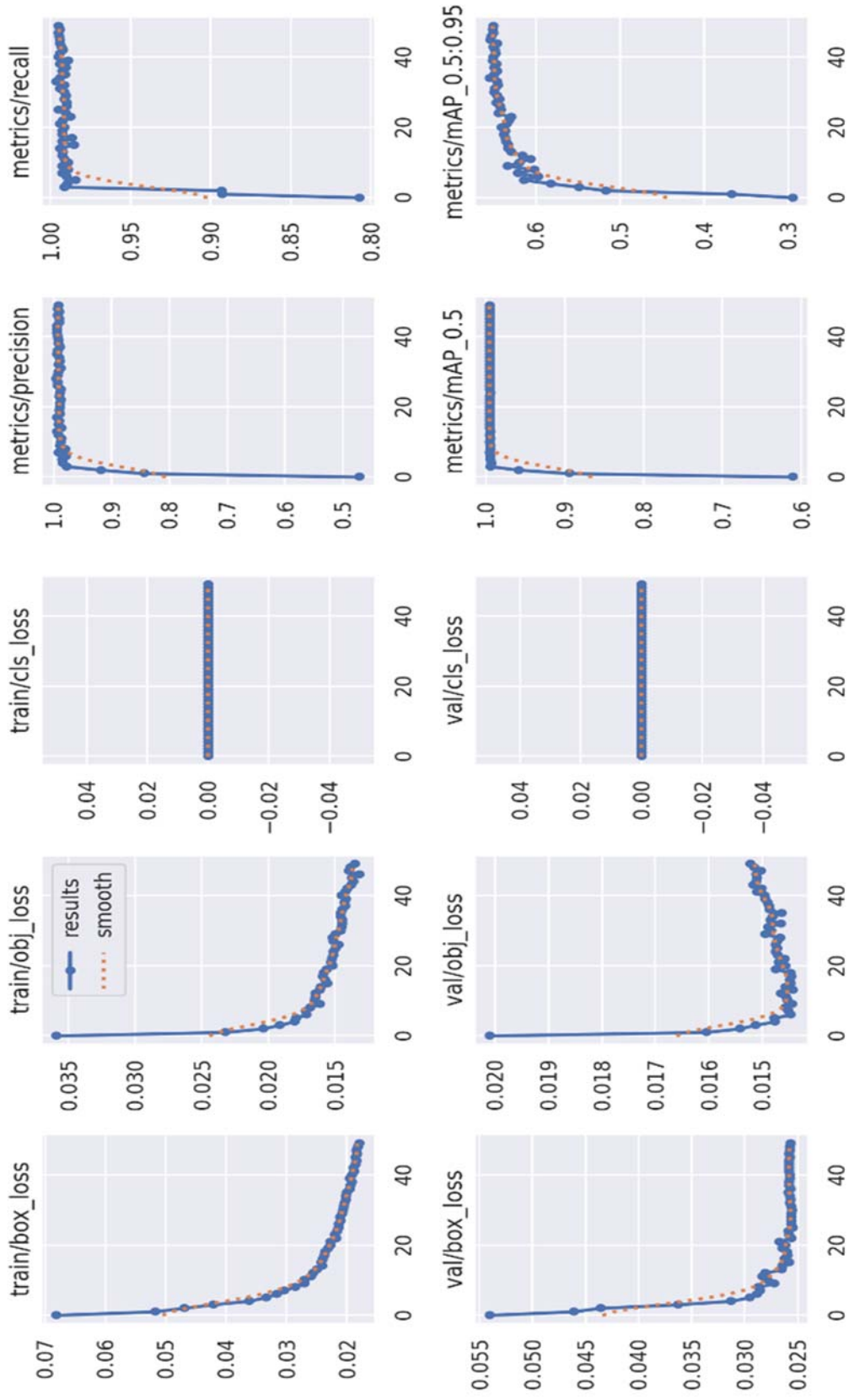
YOLOv5m modelinin eğitim sürecine ait kayıp grafikleri ve Precision-Recall eğrileri.



YOLOv5l modelinin eğitim sürecine ait kayıp grafikleri ve Precision–Recall eğrileri.



YOLOv5x modelinin eğitim sürecine ait kayıp grafikleri ve Precision-Recall eğrileri.



Buildings, 3rd International Congress on Engineering, Architecture and Design, 4-5 Mayıs 2018, Kocaeli, Türkiye.

Katırcıoğlu F, Kelek M M, Şen M, Koyuncu İ, Oğuz Y, 2018, FPGA-Based Design of Gaussian Membership Function for Real-Time Fuzzy Logic Applications, V. International Multidisciplinary Congress of Eurasia, 24-26 July 2018, Barcelona, Spain.

Şahin M, Oğuz Y, Kelek M M, Özer T, 2018, Examination of Semi Direct Lighting Type in Indoor Lighting under the Color Selection, 3rd International Conference on Engineering Technology and Applied Sciences, 17-21 July 2018, Skopje, Macedonia.

Şahin M, Oğuz Y, Özer T, Kelek M M, 2018, Integration and Application of Solar Energy in Urban Illumination, 3rd International Conference on Engineering Technology and Applied Sciences, 17-21 July 2018, Skopje, Macedonia.

Kelek M M, Özer T, Oğuz Y, Şahin M, 2018, DsPic33f Series Microprocessor Based Single-Phase Inverter Design and Implementation, International Journal of Scientific & Engineering Research, 9, 77-80.

Özer T, Kelek M M, Oğuz Y, Kıvrak S, Şahin M, 2018, Double Axis Solar Tracking System Design and Implementation, International Journal of Scientific & Engineering Research, 9, 66-71.

Kelek M M, Çelik İ, Fidan U, Oğuz Y, 2019, The Simulation of Mathematical Model of Outer Rotor BLDC Motor, 4th International Symposium on Innovative Approaches in Engineering and Natural Sciences, 22-24 November 2019, Samsun, Türkiye.

Cetin H, Durdu A, Aslan M F, Kelek M M, 2019, Q-Learning Based Real Time Path Planning for Mobile Robots, International Journal of Trend in Scientific Research and Development (IJTSRD), 4, 631-636.

Özer T, Kıvrak S, Oğuz Y, Kelek M M, 2020, STM32f103C8 Mikroişlemcisi Kullanılarak CAN Haberleşme Protokolü Tabanlı Veri Toplama Sistemi Uygulaması, Afyon Kocatepe Üniversitesi Uluslararası Mühendislik Teknolojileri ve Uygulamalı Bilimler Dergisi, 3, 27-31.

- Kelek M M, Oğuz Y, Fidan U, Özer T, 2021, Yük hücresi temelli iki tekerlekli denge robotunun PID kontrolör kullanarak gerçek zamanlı kontrolü, Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, 27, 597-603.
- Kelek M M, Fidan U, Oğuz Y, Celik I, Özer T, 2021 Load Cell-Based PID Method-Controlled Segway System Modelling and Simulation. International Journal of Engineering Systems Modelling and Simulation, 12, 230-238.
- Kelek M M, Cengiz E, Oğuz Y, Yönetken A, 2021, RLBP Metodu ile Mamografi Görüntülerinin İncelenmesi ve Sınıflandırılması, Afyon Kocatepe Üniversitesi Uluslararası Mühendislik Teknolojileri ve Uygulamalı Bilimler Dergisi, 4, 59-64.
- Özer T, Kıvrak S, Oğuz Y, Kelek M M, Karaca A, 2021, DsPic33fj32mc204 Mikroişlemci Tabanlı Hall Sensörlü Fırçasız Doğru Akım Motor Sürücüsü Tasarımı ve Proteus Simülasyon Uygulaması, Afyon Kocatepe Üniversitesi Fen ve Mühendislik Bilimleri Dergisi, 21, 820-828.
- Kıvrak S, Özer T, Oğuz Y, Kelek M M, 2021, Novel active and passive balancing method-based Battery management system design and implementation. Journal of Power Electronics, 21, 1855–1865.
- Cengiz E, Kelek M M, Oğuz Y, Yılmaz C, 2022, Classification of breast cancer with deep learning from noisy images using wavelet transform, Biomedical Engineering / Biomedizinische Technik, 67, 143-150.
- Özer T, Akdoğan C, Cengiz E, Kelek M M, Yıldırım K, Oğuz Y, Akkoç H, 2022, Cherry Tree Detection with Deep Learning, 2022 Innovations in Intelligent Systems and Applications Conference (ASYU), 7-9 September 2022, Antalya, Türkiye.
- Yildiz E, Kelek M M, Hocaoglu F O, Oğuz Y, 2023, Forecasting the Impact of Vaccination on Daily Cases in Türkiye for Covid-19, Academic Platform Journal of Engineering and Smart Systems (APJESS), 11, 19-26.
- Kelek M M, Özer T, Fidan U, Oğuz Y, 2023, Simulation and Real-time Application of the Load Cell-based Segway, Comptes rendus de l'Académie bulgare des sciences: sciences mathématiques et naturelles, 76, 75-83.
- Kelek M M, Oğuz Y, Fidan U, Özer T, 2024, Load cell-based two-wheeled mobile robot control with proportional–integral–derivative and fuzzy proportional–integral–

derivative methods, *Electrica*, 24, 247-255.

Oğuz Y, Habibi S, Kelek M M, 2025, Integration of Wind and Solar Hybrid Power Generation Systems into The Grid and Control of Power Quality with Fuzzy Logic, *Osmaniye Korkut Ata Üniversitesi Fen Bilimleri Enstitüsü Dergisi*, 8, 211-221.

Kelek M M, Fidan U, Oğuz Y, 2025, The Effect of Wavelet Transform on The Classification Performance of Different Deep Learning Architectures, *SIViP* 19, 366, 1-15.

