



**İHA KULLANILARAK ORMAN YANGINLARININ  
TESPİTİ VE GÖRÜNTÜLENMESİ İÇİN DERİN  
ÖĞRENME TABANLI GÖZETLEME SİSTEMİNİN  
GELİŞTİRİLMESİ**

**2023  
YÜKSEK LİSANS TEZİ  
MEKATRONİK MÜHENDİSLİĞİ**

**İbrahim SHAMTA**

**Tez Danışmanı  
Dr. Öğr. Üyesi Batıkan Erdem DEMİR**

**İHA KULLANILARAK ORMAN YANGINLARININ TESPİTİ VE  
GÖRÜNTÜLENMESİ İÇİN DERİN ÖĞRENME TABANLI GÖZETLEME  
SİSTEMİNİN GELİŞTİRİLMESİ**

**Ibrahim SHAMTA**

**Tez Danışmanı**

**Dr. Öğr. Üyesi Batıkan Erdem DEMİR**

**T.C.**

**Karabük Üniversitesi**

**Lisansüstü Eğitim Enstitüsü**

**Mekatronik Mühendisliği Anabilim Dalında**

**Yüksek Lisans Tezi**

**Olarak Hazırlanmıştır**

**KARABÜK**

**Ağustos 2023**

Ibrahim SHAMTA tarafından hazırlanan “İHA KULLANILARAK ORMAN YANGINLARININ TESPİTİ VE GÖRÜNTÜLENMESİ İÇİN DERİN ÖĞRENME TABANLI GÖZETLEME SİSTEMİNİN GELİŞTİRİLMESİ” başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Dr. Öğr. Üyesi Batıkan Erdem DEMİR .....  
Tez Danışmanı, Mekatronik Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından Oy Birliği ile Mekatronik Mühendisliği Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir. 25.08.2023

<u>Ünvanı, Adı SOYADI (Kurumu)</u>	<u>İmzası</u>
Başkan : Dr. Öğr. Üyesi Rıfat HACIOĞLU (BEÜ)	.....
Üye : Dr. Öğr. Üyesi Batıkan Erdem DEMİR (KBÜ)	.....
Üye : Dr. Öğr. Üyesi Şafak ALTAY AÇAR (KBÜ)	.....

KBÜ Lisansüstü Eğitim Enstitüsü Yönetim Kurulu, bu tez ile, Yüksek Lisans derecesini onamıştır.

Doç. Dr. Zeynep ÖZCAN .....  
Lisansüstü Eğitim Enstitüsü Müdürü



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

Ibrahim SHAMTA

## ÖZET

Yüksek Lisans Tezi

# İHA KULLANARAK ORMAN YANGINLARININ TESPİTİ VE GÖRÜNTÜLEMESİ İÇİN ZEKİ GÖZETLEME SİSTEMİNİN GELİŞTİRİLMESİ

**Ibrahim SHAMTA**

**Karabük Üniversitesi**

**Lisansüstü Eğitim Enstitüsü**

**Mekatronik Mühendisliği Anabilim Dalı**

**Tez Danışmanı:**

**Dr. Öğr. Üyesi Batıkan Erdem DEMİR**

**Ağustos 25, 64 sayfa**

Bu çalışmada derin öğrenme ağları kullanılarak orman yangınlarının erken tespiti için bir hava gözetleme sistemi oluşturuldu. Bu sistem, insansız hava aracı ve uçuş sırasında iletilen verileri görüntülemek için geliştirilen bir yer istasyonu uygulaması içermektedir. YOLOv8 ve YOLOv5'nin orman yangınlarını tanıma performansı incelenmiş, görüntülerin yangın içerip içermediğini sınıflandırmak için CNN-RCNN ağı tasarlanmış ve YOLOv8 ağı sınıflandırması ile sonuçlar karşılaştırılmıştır. NVIDIA Jetson Nano, bu derin öğrenme ağlarının gerçek zamanlı analizine olanak sağlanması için insansız hava aracına entegre edilmiştir. Aynı zamanda bu çalışma kapsamında geliştirilen bir yer istasyonu yazılımı, yangına ilişkin verileri almak ve görüntülemek amacıyla kullanılmıştır. Bu program, tespit edilen alanın yangın görüntülerini ve koordinatlarını göstermektedir. Algılama yöntemlerinin performans ölçümü için, CNN-RCNN ağının sınıflandırma etkinliği, YOLOv8 modeli tarafından

elde edilen sonuçlarla karşılaştırmalı bir değerlendirmeye tabi tutulmuştur. Deneysel sonuçlar, çeşitli algılama yöntemlerinde yüksek derecede doğruluk göstermiştir. CNN-RCNN ağı için %96, YOLOv8 tabanlı sınıflandırma için %96, YOLOv5 tabanlı nesne algılama için %89 ve YOLOv8 tabanlı nesne algılama için %89'dur. Genel olarak, bu araştırma orman yangınlarının erken tespiti için derin öğrenme teknolojilerinin uygulanmasına katkı sağlamakta ve bu tür doğal felaketlerin yıkıcı etkilerini hafifletebilecek kapsamlı bir sistem sunmaktadır.

**Anahtar Sözcükler:** Dört Rotorlu İHA, Orman Yangını Tespiti, Derin Öğrenme, Görüntü İşleme, YOLO, Evrimsel Sinir Ağı.

**Bilim Kodu:** 92906

## **ABSTRACT**

**M. Sc. Thesis**

### **DEVELOPMENT OF A DEEP LEARNING BASED SURVEILLANCE SYSTEM FOR FOREST FIRE DETECTION AND MONITORING USING UAV**

**Ibrahim SHAMTA**

**Karabük University**

**Institute of Graduate Programs**

**Mechatronics Engineering**

**Thesis Advisor:**

**Assist. Dr. Batikan Erdem DEMİR**

**August 2023, 64 pages**

In this study, an aerial surveillance system was developed using deep learning networks for the early detection of forest fires. This system encompasses an unmanned aerial vehicle and a ground station application developed for visualizing data transmitted during flight. The performance of YOLOv8 and YOLOv5 in detecting forest fires was examined, and a CNN-RCNN network was designed to classify whether the images contained fires or not, with the results compared to the YOLOv8 network's classification. The NVIDIA Jetson Nano was integrated into the unmanned aerial vehicle to enable real-time analysis of these deep learning networks. Additionally, a ground station software developed within the scope of this study was used to receive and display data related to fires. This program displays images and coordinates of the detected fire area. For performance evaluation of the detection methods, the classification effectiveness of the CNN-RCNN network was subjected

to a comparative assessment with the results obtained by the YOLOv8 model. Experimental results demonstrated a high degree of accuracy across various detection methods, with 96% for the CNN-RCNN network, 96% for YOLOv8-based classification, 89% for YOLOv5-based object detection, and 89% for YOLOv8-based object detection. Overall, this research contributes to the application of deep learning technologies for the early detection of forest fires, offering a comprehensive system that could mitigate the destructive effects of such natural disasters.

**Key Word:** Four-Rotor UAV, Forest Fire Detection, Deep Learning, Image Processing, YOLO, Convolutional Neural Network.

**Science Code :** 92906

## TEŐEKKÜR

Bu akademik alıőmanın yürütölmesinde bana rehberlik eden, alıőmanın araştırma ve uygulama safhalarında kritik öneriler ve yönlendirmeler sađlayan, kapsamlı deneyim ve bilgi birikimi ile genel anlamda bana yardımcı olan kıymetli danışmanım Dr. Öğr. Üyesi Batıkan Erdem DEMİR'e derin őükranlarımı sunuyorum.

Bu araştırma, Karabük Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimi tarafından KBÜ-BAP-23-YL-055 proje kodu ile finanse edilmiştir. Bu desteđi sađlayan KBÜ BAP Koordinasyon Birimi'ne içten teşekkürlerimi iletiyorum.

alıőmamın yazım sürecinde, projemin her aşamasında katkı sađlayan diđer kişilere, süreç boyunca gösterdikleri kesintisiz destek için kalpten teşekkür ederim.

Türk devleti ve Türk toplumuna, karmaşık bir savaş döneminde Suriyeli bireylere sađladıkları destek için duyduğum minnet ve takdiri ifade etmek isterim. Ayrıca, akademik eğitimime devam etme fırsatı tanıyan Karabük Üniversitesi'ne özel bir teşekkür sunmak isterim.

Hayatımın her aşamasında, maddi ve manevi açıdan beni destekleyen sevgili aileme, bu destekleri için sonsuz őükranlarımı sunuyorum.

## İÇİNDEKİLER

	<b><u>Sayfa</u></b>
KABUL.....	ii
ÖZET.....	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER .....	ix
ŞEKİLLER DİZİNİ.....	xii
SİMGELER VE KISALTMALAR DİZİNİ .....	xv
BÖLÜM 1 .....	1
GİRİŞ .....	1
BÖLÜM 2 .....	4
LİTERATÜR ARAŞTIRMASI .....	4
BÖLÜM 3 .....	7
İNSANSIZ HAVA ARAÇLI SİSTEM DÜZENLEMESİ.....	7
3.1 DÖRT ROTORLU İNSANSIZ HAVA ARACI.....	7
3.1.1. Dört Rotorlu İnsansız Hava Aracının Yer İstasyonu.....	9
3.2. NVIDIA JETSON NANO GÖMÜLÜ YAPAY ZEKA BİLGİSAYARI .....	10
3.3. NVIDIA JETSON NANO İÇİN UYUMLU SIM7600-H MODÜLÜ .....	11
BÖLÜM 4 .....	15
ORMAN YANGINLARININ DERİN ÖĞRENME ALGORİTMALARI İLE TESPİT EDİLMESİ.....	15
4.1. YAPAY ZEKA.....	16
4.2. DERİN ÖĞRENME VE MAKİNE ÖĞRENMESİ .....	19
4.2.1. Derin Öğrenme Tabanlı Görüntü İşleme .....	20
4.2.2. Derin Öğrenme Mimarileri .....	20

	<b><u>Sayfa</u></b>
4.2.3. Evrişimli Sinir Ağı.....	21
4.2.3.1. Giriş Katmanı ve Çıkış Katmanı.....	22
4.2.3.2. Evrişimli Katman.....	23
4.2.3.3. Doğrusallaştırılmış Birim.....	24
4.2.3.4. Havuzlama Katmanı.....	24
4.2.3.5. Tam Bağlantılı Katman.....	24
4.2.4. Tekrarlayan Sinir Ağı.....	25
4.2.5. Uzun Kısa Süreli Bellek Ağları.....	26
4.2.6. Bölge Tabanlı Evrişim Sinir Ağı.....	27
4.2.7. Hızlı Bölge Tabanlı Evrişimli Sinir Ağı.....	29
4.2.8. Daha Hızlı Bölge Tabanlı Evrişimli Sinir Ağı.....	30
4.2.9. YOLO Ağı.....	32
4.2.10. YOLOV5 Ağı.....	32
4.2.10.1. YOLOv5 Mimarisi.....	33
4.2.11. YOLOv8 Ağı.....	35
4.2.11.1. YOLOv8 Mimarisi.....	35
4.2.12. CNN-RCNN Ağı Tasarımı.....	37
4.3. VERİ SETİ OLUŞTURULMASI.....	40
4.3.1. Veri Setinin Etiketlenmesi.....	41
4.4. GELİŞTİREN YER İSTASYONU ARAYÜZÜ İLE ORMAN YANGINININ İZLENMESİ.....	43
BÖLÜM 5.....	45
DENEYSEL ÇALIŞMALAR.....	45
5.1. DÖRT ROTORLU İNSANSIZ HAVA ARACI İLE UÇUŞ TESTLERİ.....	46
5.1.2. Dört Rotorlu İnsansız Hava Aracının Otonom Uçuş Testleri.....	46
5.2. DERİN ÖĞRENME ALGORİTMALARI VE İNSANSIZ HAVA ARACI KULLANILARAK ORMAN YANGINLARINI ALGILAMA DENEYLERİ ...	47
5.2.1. Orman Yangınlarını Tespit Etmek İçin Kullanılan Algoritmaların Performansının Karşılaştırılması Ve Değerlendirilmesi.....	49
5.2.1.1. Sınıflandırma Algoritması Eğrileri.....	49
5.2.1.2. Eğriler Nesne Algılama Algoritmaları.....	50

	<b><u>Sayfa</u></b>
5.2.1.3. Performans Deęerlendirmesi .....	52
BÖLÜM 6 .....	56
SONUÇLAR VE DEęERLENDİRME .....	56
KAYNAKLAR .....	58
ÖZGEÇMİŞ .....	64



## ŞEKİLLER DİZİNİ

### Sayfa

Şekil 3.1. Geliştirilen sistemin blok şeması. ....	7
Şekil 3.2. Kullanılan dört rotorlu insansız hava aracı. ....	9
Şekil 3.3. Mission Planner programı, arayüzü. ....	10
Şekil 3.4. Nvidia Jetson NANO gömülü yapay zeka bilgisayarı. ....	11
Şekil 3.5. SIM7600-H Modülü ....	12
Şekil 3.6. Nvidia Jetson NANO ile SIM 7600-H modülü. ....	13
Şekil 4.1. Yapay zeka, makine öğrenmesi ve derin öğrenme arasındaki farklar. ....	18
Şekil 4.2. Derin öğrenme makine öğrenmesi diyagramı. ....	19
Şekil 4.3. Evrişimli Sinir Ağı. ....	22
Şekil 4.4. Evrişimli katman. ....	23
Şekil 4.5. Tekrarlayan Sinir Ağı. ....	26
Şekil 4.6. LSTM Sinir Ağı. ....	27
Şekil 4.7. R-CNN Mimarisi. ....	28
Şekil 4.8. Kesiştirilmiş Bölgeler ....	29
Şekil 4.9. Maksimum olmayan bastırma uygulama öncesi ve sonrası. ....	29
Şekil 4.10. Daha Hızlı Bölge Tabanlı Evrişimli Sinir Ağı. ....	31
Şekil 4.11. Bölge Öneri Ağı. ....	31
Şekil 4.12. YOLOv5 ....	34
Şekil 4.13. YOLOv8 mimarisi. ....	36
Şekil 4.14. YOLOv8 ve eski sürümlerin karşılaştırılması. ....	37
Şekil 4.15. CNN R-CNN Ağı mimarisi ....	38
Şekil 4.16. a) Orman Yangını b) Karabük Üniversitesi'nde Hazırlanan Görüntüleri	40
Şekil 4.17. Veri seti açıklaması. ....	41
Şekil 4.18. Roboflow veri setinin etiketleme platformu. ....	42
Şekil 4.19. Veri setinin etiketlemesi. ....	42
Şekil 4.20. Etiket dosyası. ....	43
Şekil 4.21. Yer istasyonu arayüzü. ....	44
Şekil 4.22. Yangının çıktığı yerin tespiti ....	44

	<b><u>Sayfa</u></b>
Şekil 5.1. Test amaçlı oluşturulan kontrollü yangın. ....	45
Şekil 5.2. Deneilerde kullanılan ekipmanlar. ....	47
Şekil 5.3. Orman yangını tespit testi için oluşturulan otonom uçuş rotası.....	48
Şekil 5.4. Otonom uçuş rotası parametreleri. ....	48
Şekil 5.5 YOLOV8 sınıflandırma modelinin eğitim sonuçları. ....	49
Şekil 5.6 CNN-RCNN sınıflandırma modelinin eğitim sonuçları. ....	50
Şekil 5.7 YOLOv5 ve YOLOv8 nesne algılama modelinin eğitim sonuçları. ....	51



## ÇİZELGELER DİZİNİ

### Sayfa

Çizelge 3.1. Nvidia Jetson NANO gömülü yapay zeka bilgisayarının teknik özellikleri.....	10
Çizelge 5. 1. Deneysel çalışma sonucu oluşan hata matrisi.....	54
Çizelge 5. 2. Nesne algılama için değerlendirme metrikleri.....	54
Çizelge 5. 3. Değerlendirme Metrikleri .....	55
Çizelge 5. 4. NVIDIA Jetson Nano'da işlem hızı .....	55

## SİMGELER VE KISALTMALAR DİZİNİ

### SİMGELER

- a-Si : Amorf Silisyum  
C : Kapasitenin X Katı Kadar Akım Miktarı  
CdTe : Kadmiyum Tellür  
FT : Toplam İtki Kuvveti  
GaAs : Galyun Arsenik  
 $H_i$  : Hub Kuvveti  
 $I_{xx}$  : X Eksenindeki Atalet Momenti  
 $I_{yy}$  : Y Eksenindeki Atalet Momenti  
 $I_{zz}$  : Z Eksenindeki Atalet Momenti  
Li-Ion : Lityum-İyon  
Li-Po : Lityum-Polimer  
mAh : Milimamper Saat  
 $R_i$  : Yuvarlanma Momentini  
 $R_{x,y,z}$  : Rotasyon Matrisi  
Si : Silisyum  
S : Bataryada Kullanılan Hücre Sayısı  
 $T_i$  : Pervanenin Ürettiği İtki Kuvveti  
 $U_1$  : İtki Kuvvetini  
 $U_2$  : Yuvarlanma Kuvveti  
 $U_3$  : Yunuslama Kuvveti  
 $U_4$  : Sapma Kuvveti  
 $\Omega_i$  : Pervanelerin Açısal Hızları  
 $\Theta$  : Yunuslama Açısı  
 $\Phi$  : Yuvarlanma Açısı  
 $\Psi$  : sapma açısı

## KISALTMALAR

BLDC	: Brushless Direct Current (Fırçasız Doğru Akım)
CNN	: Convolution Neural Network (Evrışimli Sinir Ağı)
CPU	: Central Process Unit (Merkezi İşlem Birimi)
DAE	: Deep Auto Encoder (Derin Oto-Kodlayıcılar)
DBN	: Deep Belief Network (Derin İnanç Ağları)
DCNN	: Deep Convolution Neural Network (Derin Evrışimli Sinir Ağı)
ESC	: Electronic Speed Controller (Elektronik Hız Denetleyicileri)
EVA	: Etilen Vinil Asetat
EU	: European Union (Avrupa Birliği)
ESA	: European Space Agency (Avrupa Uzay Ajansı)
FC	: Fully Connected (Tam Bağlı Katman)
GPS	: Global Position System (Küresel Konumlandırma Sistemi)
GPU	: Graphics Processing Unit (Grafik İşlem Birimi)
IMU	: Inertial Measurement Unit (Ataletsel Ölçü Birimi)
IoU	: Intersection Over Union (Birlik Üzerinden Kesişme)
İHA	: İnsansız Hava Aracı
LTSM	: Long Short-Term Memory Neural Network (Uzun Kısa Vadeli Hafıza Ağları)
OHEM	: Online Hard Example Mining (Çevrimiçi Zor Örnek Madenciliği)
mAP	: Mean Average Precision (Ortalama Hassasiyetin Ortalama Değeri)
NMS	: Non-Maximal Suppression (Maksimum Olmayan Baskı Eşiği)
PV	: Fotovoltaik (Photovoltaic)
PVF	: Polyvinyl Fluoride Film (Polivinil Florid Film)
PWM	: Pulse Width Modulation (Darbe Genişlik Modülasyonu)
RBM	: Restricted Boltzmann Machines (Sınırlı Boltzmann Makineleri)
ReLu	: Rectified Linear Unit (Doğrutulmuş Doğrusal Lineer Birim)
RNN	: Recurrent Neural Network (Tekrarlayan Sinir Ağı)
RCNN	: Recurrent Convolution Neural Network (Tekrarlayan Evrışimli Sinir Ağı)
R-CNN	: Region-Based Convolutional Neural Networks (Bölge Tabanlı Evrışimsel Sinir Ağları)

RPN	: Region Proporsal Network (Bölge Öneri Ağı)
SSD	: Single Shot MultiBox Detector(Tek Atışlı Çok Kutulu Dedektör)
UAV	: Unmanned Aerial Vehicle (İnsansız Hava Aracı)
USB	: Universal Serial Bus (Evrensel Seri Veriyolu)
WSN	: Wireless Sensor Networks (Kablosuz Sensör Ağları).
YOLO	:You Only Look Once (Sadece Bir Kez Bakarsınız)



## BÖLÜM 1

### GİRİŞ

Sanayi devrimi'nin getirdiği antropojenik etmenlerin etkisi altında, küresel çapta yangın aktivitesinde belirgin bir düşüş gözlenmiştir. Bu süreci etkileyen unsurlar arasında, arazi kullanımında yaşanan dönüşümler ve yangın söndürme yöntemlerinin uygulanması yer almaktadır. Bununla beraber, son 50 yıl içerisinde dünyanın belirli bölgeleri artan yangın sıklığı ile karşı karşıya kalmıştır. Bu tespiti doğrular nitelikte olan kayıtlar ve yanmış alanların tarihsel dinamiklerini modellere yansıtan analizler de bulunmaktadır[1], [2].

Avrupa Birliği (European Union - EU), 2030 için bir strateji oluştururken yangın önleme konusunu Avrupa ormanlarının korunması ve dayanıklılığının artırılması için temel bir unsur olarak açıkça kabul etmektedir. Birleşmiş Milletler Çevre Programı'nın orman yangınları üzerine raporu (2022), reaktif yaklaşımlardan, önceden önlem ve hazırlık odaklı yaklaşımlara geçmek için kaynakların önleme, destek ve yangın koruma önlemlerinin geliştirilmesine tahsis edilmesini acil bir gereklilik olduğunu vurgulamaktadır [3], [4].

Orman yangınlarını erken evrelerde saptamak, özellikle denetim noktalarının dışındaki ormanlarda artan yangın olayları göz önüne alındığında, çevresel bir felaketi önleme noktasında kritik bir rol oynamaktadır. Bu tür erken tespitler, yangın kontrolden çıkmadan önce müdahale edilebilmesi için en uygun zaman dilimini sağlar, çünkü bu aşamada yangının yayılma hızını nispeten yavaştır ve etkilediği alan daha sınırlı tutar [5], [6].

Son dönemlerde, dört rotorlu insansız hava araçları (İHA) sivil amaçlar için çeşitli uygulamalarda (örneğin; yük taşımacılığı, film çekimi) giderek daha popüler hale gelmektedir [7]. Bu araçlar, askeri alanda da özellikle hedef izleme gibi amaçlar için

kullanılmaktadır. İnsansız hava araçlarının sunduğu diğer avantajlar arasında, daha düşük işletme maliyetleri, insanlı uçaklarla karşılaştırıldığında daha az can kaybı riski ve insan yorgunluğu ya da çalışma saatleri gibi faktörlerin sınırlayıcı etkisinin olmaması sayılabilir [8].

Sensör tabanlı duman veya yangın tespit sistemleri kapalı mekanlarda oldukça etkili olmakla birlikte, orman gibi açık alanlarda uygulanabilirliği bir dizi zorluğa sahiptir. Bu zorlukların başında yüksek maliyetler gelmektedir. Ayrıca, bu tür sistemler aracılığıyla yangının görsel olarak tespit edilmesi ve söndürme ekipleri için yangının nasıl ve hangi hızla yayılacağına dair kesin verilerin toplanması genellikle mümkün olmamaktadır [9], [10]. Diğer yandan, termal veya morötesi kameralar gibi görüntüleme araçlarının orman yangınlarını tespit etmek için kullanılması, açık alanlar için genellikle uygun değildir. Bu tür kameralar, çevresel faktörlerin olumsuz etkilerine hızla maruz kalmakta ve kısa tespit mesafeleri nedeniyle açık alanlarda etkili olamamaktadırlar. Uydu temelli uzaktan algılama yöntemleri büyük yangınların başlangıcını tespit etmek için kullanışlı olabilir; ancak, yangınların erken aşamalarında, özellikle küçük ölçekli yangınları tespit etme kapasitesi sınırlıdır [9], [11]–[13].

Bu çalışmada, orman yangınlarının erken evrelerini tespit etmek amacıyla dört rotorlu bir insansız hava aracı kullanılarak derin öğrenme tabanlı gözetleme sistemi tasarlanmıştır. Araştırma kapsamında, bu amaç için özel olarak bir yer istasyonu geliştirilmiştir. İHA'nın üzerinde bulunan kamera aracılığıyla elde edilen görüntüler, NVIDIA Jetson NANO geliştirme kartı ile işlenmektedir.

Sistem, çoklu bileşenlerden meydana gelmektedir. İlk bileşen, orman yangınlarının tespiti için tasarlanmış bir evrişimli sinir ağı- bölge tabanlı evrişimli sinir ağı (Convolutional Neural Network-Recurrent Convolutional Neural Network - CNN-RCNN) algoritmasıdır. İkinci bileşen ise, dört rotorlu İHA'nın tespit ettiği orman yangını verilerini izlemek amacıyla bir yer istasyonu uygulamasının tasarımını kapsamaktadır.

Bu tezin ikinci bölümünde, orman yangınlarının tespiti ile ilgili literatür incelemesi yapılmıştır. Bu kısımda, orman yangınlarının tespiti ile ilgili derin öğrenme algoritmaları ve mevcut akademik çalışmalar üzerinde durulmaktadır. Üçüncü bölüm, kullanılan dört motorlu insansız hava araçları ve kullanılan Nvidia Jetson NANO yapay zeka bilgisayarı hakkında bilgiler sunmaktadır. Dördüncü bölüm, orman yangınlarını tespit ve teşhis etmek için kullanılan derin öğrenme algoritmaları, elde edilen veri seti ve tasarlanan yer istasyonu arayüzünü kapsamaktadır. Beşinci bölüm, deneysel çalışmaların detaylarını sunmaktadır. Son bölümde, orman yangını tespit sisteminin performansı analiz edilmiş ve gelecekteki çalışmalar için çeşitli öneriler sunulmuştur.



## BÖLÜM 2

### LİTERATÜR ARAŞTIRMASI

Orman yangınlarının ciddiyeti ve bu tür felaketlerin dünya genelinde yarattığı zararlar göz önüne alındığında, bu öncelikli alanda çeşitli araştırmalar yürütülmektedir. Bu araştırmaların bir kısmı, orman yangınlarını erken aşamada tespit etmeyi sağlayacak izleme sistemleri geliştirmeye odaklanmaktadır. Diğer bir kısmı ise, yangın ve dumanın etkin bir şekilde tespit edilebilmesi için algoritmalar geliştirmeyi amaçlamaktadır.

Wang ve ekibi, YOLOv5 tabanlı bir duman tespit modeli geliştirmeyi önermektedir. Bu kapsamda, gerçek ve yapay duman görüntülerinden oluşan bir veri seti oluşturulmuş, ardından farklı kayıp fonksiyonları çeşitli YOLOv5 modelleri üzerinde test edilmiştir. Veri setini zenginleştirmek adına mozaik iyileştirme yöntemi kullanılmıştır. Ayrıca, tespit performansını artırmak için bir dikkat mekanizması da eklenmiştir. Deneysel bulgular, bu çalışmada geliştirilen algoritmanın tespit performansının, geleneksel derin öğrenme algoritması ile kıyaslandığında temel modelin mAP (Ortalama Hassasiyeti) değerinden %4,4 daha yüksek olduğunu göstermektedir. Tespit hızı ise 85 saniyedeki kare'ye ulaşmıştır. Bu sonuçlar, modelin mühendislik uygulamaları için gereksinimleri karşılayabilecek daha üstün bir performans sergilediğini açıkça ortaya koymaktadır [14].

Ravishankara ve ekibi, YCbCr renk modelini temel alarak yangın tespiti için kural tabanlı bir algoritma geliştirmişlerdir. İstatistiksel parametrelerin detaylı bir analiziyle, bu algoritma ev sahibi yangın piksellerini etkin bir şekilde ayırt edebilmekte ve yüksek sıcaklıktaki yangın merkezi piksellerini izole edebilmektedir. Gerçek yangın tespit oranının %98,83 olduğu ve düşük hata oranlarına sahip olduğu gözlemlenen bu algoritma, mobil kameralar veya insansız hava araçları ile entegre

edilerek gerçek zamanlı orman yangını tespiti için büyük potansiyel sunmaktadır [15].

Shi ve ekibi, çalışmalarında video gözetimine dayalı bir yangın izleme ve alarm sistemini ele almaktadır. Sistem, YOLOv3 tabanlı yangın ve duman dedektörleri ile donatılmıştır ve bir dizi avantaj sunar. Bunlar hızlı tepki süresi, sıcaklık duyarsızlığı ve detaylı gözetim görüntüleridir. Sistem, internetten toplanan ve LabelImg aracılığıyla etiketlenen yeni bir veri seti üzerinde eğitilmiştir. Örnek dengesizliği sorununu çözmek için, çevrimiçi zor örnek madenciliği (Online Hard Example Mining - OHEM) tekniği uygulanmıştır. Yangın ve duman tespit sonuçlarının entegrasyonu sayesinde, etkin yangın alarmları oluşturulmaktadır. Önerilen algoritmanın etkinliği, spesifik bir veri seti üzerinde yapılan deneysel çalışmalarla doğrulanmıştır [16].

Wu S, Zhang L ve ekibi, bu makalede gerçek zamanlı ve erken orman yangını tespitinin yanı sıra yanlış pozitif tespitleri azaltma konularına odaklanmaktadır. Faster R-CNN, YOLO (tiny-yolo-voc, tiny-yolo-voc 1, yolo-voc 2.0 ve YOLOv3) ve Tek Atış Çoklu Kutu Algılama (Single Shot MultiBox Detecto - SSD) gibi klasik nesne tespit algoritmaları, ilk kez orman yangını tespiti için uygulanmıştır. Bu algoritmalar arasında, SSD üstün gerçek zamanlı performans, yüksek tespit doğruluğu ve erken yangın tespiti potansiyeli ile dikkat çekmektedir. Yanlış tespitleri minimize etmek amacıyla, bir duman kategorisi ve yangın bölgesinin dinamik değişimlerini içeren yeni bir test seti oluşturulmuştur. Ayrıca, YOLO mimarisi modifiye edilerek, yangın tespit doğruluğunu önemli ölçüde artıran yeni bir versiyon olan tiny-yolo-voc1 kullanılmıştır [17].

Kaura ve ekibi bu çalışmasında, orman yangınlarını tespit etmede kullanılan çeşitli yangın tespit tekniklerini analiz etmek ve değerlendirmektir. Çalışma, insan gözlemi, CCD kameralar, IR, uydu sistemleri ve kablosuz sensör ağları (Wireless Sensor Networks - WSN) gibi farklı yöntemleri araştırmaktadır. Dempster-shafer yöntemi, bulanık mantık ve eşikleme yöntemi gibi algoritmalar, topladığı verilerin analizi için detaylı bir inceleme altına alınmıştır. Her bir yöntemin yanlış pozitif ve yanlış negatif oranlarına odaklanmayı hedeflenmekte, bu bağlamda doğruluk ve enerji tüketimi gibi

sınırlamalara yönelik etkili çözümler sunan yeni algoritmaların geliştirilmesi gerektiğini önerilmektedir [18].

Çelik ve ekibi çalışmalarında, görüntü işleme tekniklerini kullanarak yangın ve duman tespiti için inovatif modelleri ele almaktadır. Bu modeller, yangın ve dumanın tespitinde farklı renk uzaylarını kullanmaktadır. Bu renk uzayları, çeşitli video sekansları ve görüntü verilerinden elde edilen istatistiksel analizler aracılığıyla oluşturulmaktadır. Bu şekilde türetilen renk modelleri, hareket analizi ile renk bilgisini bir araya getiren kapsamlı bir yangın ve duman tespit sistemi içerisine entegre edilebilir [19].

Calle ve ekibi tarafından yapılan çalışmada, ENVISAT uydusuna ait AATSR sensörünü kullanarak orman yangınlarının tespitine yoğunlaşmaktadır. Çalışma, 2002 ve 2003 yıllarında Çin'in kuzeydoğusunda meydana gelen orman yangınlarını analiz ederken, yeni çevresel koşullar için yangın tespit algoritmalarını geliştirmeyi hedeflemektedir. Bu araştırma, Avrupa Uzay Ajansı (European Space Agency - ESA) tarafından finanse edilen DRAGON projesi çerçevesinde Valladolid Üniversitesi'nin uzaktan algılama laboratuvarı tarafından yürütülmekte ve Çin Halk Cumhuriyeti'nin Bilim ve Teknoloji Bakanlığı ile işbirliği içinde gerçekleştirilmektedir [20].

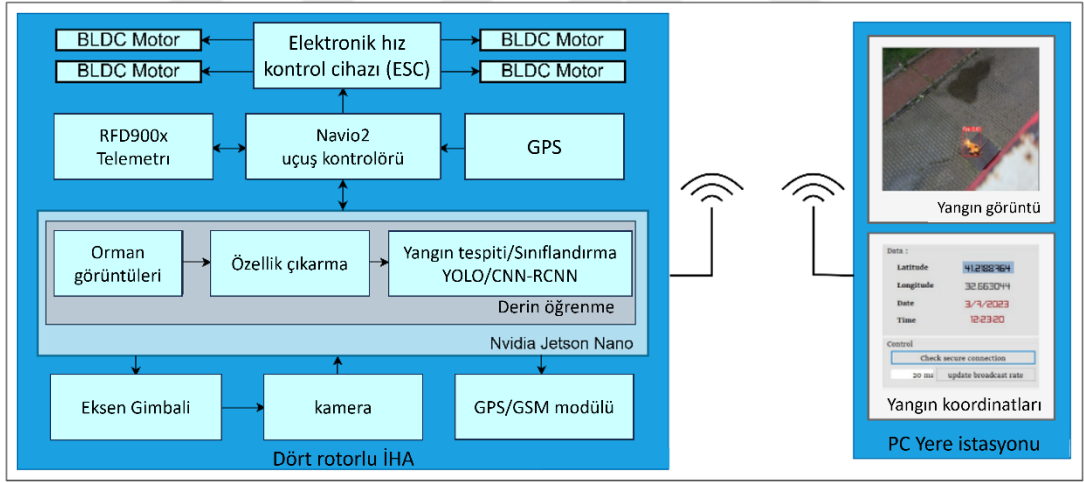
Bu tez çalışmasında, belirli bir bölgede otonom uçuşlar gerçekleştirmek amacıyla dört rotorlu bir İHA kullanılmaktadır. Orman yangınlarının tespiti için CNN-RCNN algoritması kullanılmış ve bu algoritma, YOLOv5 ile YOLOv8 ağlarıyla karşılaştırılmıştır. Tespit işlemleri sırasında, Nvidia Jetson NANO yapay zeka bilgisayarı tercih edilmiştir. Orman yangınlarının yerini ve yayılma alanlarını belirlemek üzere özel bir yer istasyonu yazılımı geliştirilmiştir.

Literatür taraması sonucunda, Nvidia Jetson NANO yapay zeka bilgisayarı kullanarak orman yangınlarını tespit eden ve erken aşama yangın uyarıları ile risk değerlendirmelerini sunan entegre bir izleme sistemine dair çalışmalar oldukça yenidir. Bu yönleriyle, çalışma bilimsel literatüre yeni katkılar sağlamaktadır ve endüstriye yönelik araştırmalar için de faydalı olmaktadır.

## BÖLÜM 3

### İNSANSIZ HAVA ARAÇLI SİSTEM DÜZENLEMESİ

Bu tez çalışmasında, dört rotorlu İHA ile birlikte Raspberry Pi ve Navio2'den oluşan bir kontrol ve otopilot birimini bulunmaktadır. Tespit sistemi ise NVIDIA Jetson Nano geliştirme kartına entegre edilmiştir. Bu kartın seçiminde, CUDA ve cuDNN'i destekleyen NVIDIA TEGRA grafik işlem birimini içermesi gelen verilerin yüksek hızda işlenmesi ve analiz edilmesini sağlaması temel alınmıştır. Sisteme internet bağlantısı ve koordinat bilgisi sağlamak için bir GPS/GSM modülü entegre edilmiştir. Tez çalışmasının blok şeması Şekil 3.1'de verilmektedir.



Şekil 3.1. Geliştirilen sistemin blok şeması.

### 3.1 DÖRT ROTORLU İNSANSIZ HAVA ARACI

Dört rotorlu insansız hava araçları (İHA'lar), bir ana gövde üzerine monte edilmiş çeşitli elektronik bileşenlerden oluşurlar. İtici kuvvetin oluşturulmasında görev alan pervaneler, motorlar tarafından döndürülerek aracın hareketini sağlar. Motorların dönme hızı, itki kuvvetini düzenlemek için elektronik hız kontrol üniteleri (Electronic Speed Controllers-ESC) tarafından ayarlanır. Bu kontrol üniteleri, uçuş

kontrol kartından gelen sinyalleri alır ve motorların dönme hızını buna göre düzenler, böylece İHA'nın istenen hareket profiline uygun bir şekilde manevra yapabilmesi sağlanır.

Uçuş kontrol kartı, dört rotorlu insansız hava araçlarının merkezi işlem birimidir ve genellikle aracın "beyni" olarak kabul edilir. Sensörlerden gelen verileri işleyerek, aracın hareketini düzenler ve istikrarını sağlar. İHA'nın navigasyonu, konumu, yüksekliği ve yönü gibi kritik parametreler, bu kart tarafından sürekli olarak izlenir ve kontrol edilir. Uçuş kontrol kartı, araç içerisinde yer alan batarya sistemi aracılığıyla elektrik enerjisi alır, bu sayede sürekli ve kesintisiz bir operasyon kapasitesi sağlanır.

Dört rotorlu insansız hava aracının tasarımı, maksimum uçuş ağırlığı dikkate alınarak gerçekleştirilir. Maksimum uçuş ağırlığı, aracın üzerinde taşıdığı ekipmanların toplam ağırlığına eşittir. Ekipmanların ağırlığı elektronik tartıda ölçülerek maksimum uçuş ağırlığı 2.45 kg olarak belirlenmiştir.

Dört rotorlu insansız hava aracının gövdesi tasarlanırken, gövde ile motor arasındaki çapa dikkat edilir. Gövde ile motor arası çapın artmasıyla daha geniş pervaneler kullanılır ve daha fazla ağırlık taşınabilir hale gelir. Maksimum uçuş ağırlığı 2.45 kg olan bir hava aracı için en uygun çap 450 mm'dir. Bu tasarım, 12 inç kadar olan pervaneleri destekleyebilir, böylece farklı boyutlarda pervaneler kullanılabilir. Bu çalışmada kullanılan dört rotorlu insansız hava aracı Şekil 3.2'de gösterilmektedir.

Bu İHA'da kullanılan motorlar SunnySky marka X3108S model 720kv FDAM motorlardır. Uçuş kontrol paneli ise Navio2'dir. Kullanılan pil Leopard Power marka 3S 11.1V 6000mAh 30C Li-po bataryadır.



Şekil 3.2. Kullanılan dört rotorlu insansız hava aracı.

### 3.1.1. Dört Rotorlu İnsansız Hava Aracının Yer İstasyonu

Dört rotorlu bir insansız hava aracının uçuş parametrelerinin düzenlenmesi ve izlenmesi için yer istasyonu yazılımı kullanılır. Bu yazılım, İHA'nın anlık uçuş bilgilerini görüntüleme kapasitesine de sahiptir. İHA'da kullanılan Navio2 uçuş kontrol kartı, Ardupilot'a dayalı açık kaynaklı bir yazılımla uyumlu şekilde çalışır. Uçuş parametrelerinin ve ayarlarının yapılandırılabilmesi için Mission Planner adlı bir program kullanılır.

Mission Planner yazılımı sayesinde, İHA'nın gövde tipi, motor sayısı, elektronik hız denetleyici kalibrasyonu ve kumanda kalibrasyonu gibi temel uçuş ayarları kolaylıkla yapılandırılabilir. Ayrıca, bu program otomatik görev planlama, otomatik uçuş rotası oluşturma ve oluşturulan rotanın simülasyonu gibi gelişmiş özelliklere de sahiptir. Mission Planner programının kullanıcı arayüzüne ait bir ekran görüntüsü, Şekil 3.3'te sunulmaktadır.



Şekil 3.3. Mission Planner programı, arayüzü.

Bu yazılımın sunduğu fonksiyonlar, dört rotorlu İHA'nın etkin ve güvenli bir şekilde operasyonunu sağlamak için kritik öneme sahiptir.

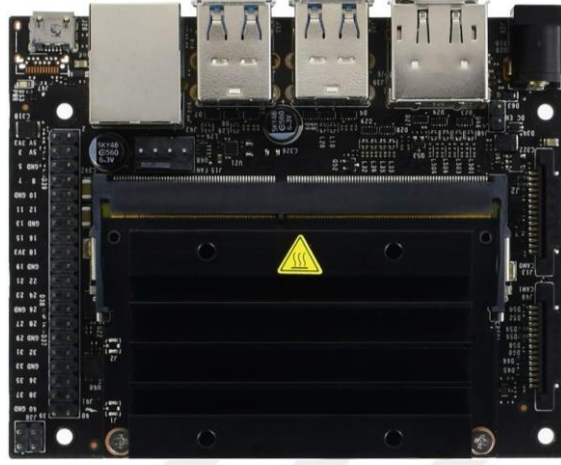
### 3.2. NVIDIA JETSON NANO GÖMÜLÜ YAPAY ZEKA BİLGİSAYARI

Nvidia Jetson Nano gömülü yapay zeka bilgisayarı, Ubuntu 18.04 işletim sistemi temelinde çalışır ve bu platform, kapsamlı bir yapay zeka çözümü sunar. Teknik detayları, Çizelge 3.1'de ayrıntılı bir şekilde listelenmiştir. Çalışmanın ana amacı, yangın tespiti ve orman yangını izleme konusunda bir uygulama geliştirmektir. Bu uygulama, yangınla ilgili verileri anlık olarak analiz edebilecek bir yapay zeka algoritması kullanır.

Çizelge 3.1. Nvidia Jetson NANO gömülü yapay zeka bilgisayarının teknik özellikleri.

CPU	1.43 GHz Arm Cortex-A57 (quad-core)
GPU	128-core Pascal @ 921 MHz & 0.5 TFLOPS (FP16)
Hafıza	4GB 64-bit LPDDR4 @ 1600MHz -25.6 GB/s
Depolama	16GB eMMC 5.1
Ethernet	10/100/1000 Base-T Ethernet

Nvidia Jetson Nano gömülü yapay zeka bilgisayarı Şekil 3.4'te görülmektedir.



Şekil 3.4. Nvidia Jetson NANO gömülü yapay zeka bilgisayarı.

Nvidia Jetson Nano'nun bu konuda tercih edilmesinin sebepleri arasında yüksek işlem kapasitesi CUDA ve cuDNN gibi gelişmiş programlama arayüzlerini desteklemesi yer almaktadır. Bu sayede, karmaşık görüntü işleme ve veri analizi görevleri hızlı ve etkin bir şekilde gerçekleştirilebilmektedir [21].

Bu yapay zeka bilgisayarı, yangın tespiti ve izlemesi için tasarlanmış olan izleme uygulamasıyla entegre edilerek, orman yangınlarına karşı daha hızlı ve etkin bir müdahale kapasitesi oluşturulması hedeflenmektedir.

### 3.3. NVIDIA JETSON NANO UYUMLU SIM7600-H MODÜLÜ

İnternet bağlantısı ve konum belirleme ihtiyacını karşılamak amacıyla, Şekil 3.5'te detaylandırıldığı üzere, Jetson Nano için özel olarak tasarlanmış bir 4G/3G/2G ve GNSS modülü kullanılmıştır. Bu modül, LTE CAT4 standardını destekleyerek, 150 megabit/saniyeye kadar veri indirme hızı sağlar. Aynı zamanda, oldukça düşük enerji tüketimiyle dikkat çekmektedir [22]–[24].

Bu yüksek hızlı 4G bağlantısı sayesinde, kablosuz iletişim imkanı, uzaktan video izleme, telefon görüşmeleri, SMS gönderme ve küresel konum belirleme gibi birçok fonksiyonu kolaylıkla etkinleştirebilirsiniz. Yüksek hızlı bağlantı kapasitesi, özellikle

orman yangını gibi acil durumlar için hızlı ve güvenilir bir iletişim altyapısı oluşturur, bu da müdahalenin hızını ve etkinliğini artırır. Bu modül, sunduğu çok yönlü özellikler ve avantajlar ile dikkat çekmektedir.



Şekil 3.5. SIM7600-H modülü.

- 40PIN GPIO Uzatma Başlığı: Jetson Nano ile kolayca entegre edilebilmesi için kullanılır.
- Çeşitli Protokol Desteği: Dial-up, telefon görüşmesi, SMS, e-posta, TCP, UDP, DTMF, HTTP ve FTP gibi birçok protokolü destekler.
- Çoklu Konumlandırma Sistemi: GPS, BeiDou, Glonass ve LBS baz istasyonu konumlandırma desteği sunar.
- Onboard USB Arabirimi: AT komutlarını test etmek, GPS konum verilerini almak gibi işlemler için mevcuttur.
- UART Kontrol Pinleri: Arduino/STM32 gibi farklı ana kartlarla uyumlu bir şekilde bağlantı kurabilmesi için tasarlanmıştır.
- SIM Kart Uyumluluğu: Hem 1.8V hem de 3V SIM kartları destekler.
- Ses Desteği: 3.5mm ses jakı üzerinden kulaklık ve mikrofon için destek sağlar.
- LED Göstergeleri: Cihazın çalışma durumunu anında izlemeyi kolaylaştırır.
- Ayarlanabilir Gerilim: Onboard gerilim çevirici sayesinde, çalışma gerilimi 3.3V veya 5V olarak seçilebilir.



13. Ağ durumu göstergesi

14. Güç göstergesi

15. UART etkinleştirme anahtarı:

ON: SIM7600'ün UART portlarını Jetson Nano'ya bağlar

OFF: SIM7600'ün UART portlarını Jetson Nano'dan ayırır

16. Çalışma gerilimi seçim jumper'ı:

VCC - 3.3V: Çalışma gerilimini 3.3V olarak ayarlar.

VCC - 5V: Çalışma gerilimini 5V olarak ayarlar.

17. PWR kontrol yapılandırması:

PWR - 5V: SIM7600 güç verildiğinde otomatik olarak başlar.

PWR - D6: SIM7600, Jetson Nano'nun D6 pimi tarafından açılıp kapatılır

[22]–[24].

## BÖLÜM 4

### ORMAN YANGINLARININ DERİN ÖĞRENME ALGORİTMALARI İLE TESPİT EDİLMESİ

Bu çalışma oldukça kapsamlı ve entegre bir yaklaşım sunmaktadır. İHA'lar ile orman yangınlarının erken tespiti için derin öğrenme yöntemlerinin bir araya getirilmesi, yangın yönetimi açısından büyük bir ilerleme olabilir. İHA'nın altında yer alan gimbal mekanizması ile desteklenen kamera, ormanları detaylı bir şekilde görüntüleyebilecek kapasitededir. Görüntülerden oluşturulan bu veri seti, özellikle etiketleme işlemleri ile yangın bölgelerini belirginleştirmektedir.

Veri setinin derin öğrenme yöntemleriyle işlenmesi ve farklı sinir ağları aracılığıyla yangın tespiti yapılması, özellikle YOLOv8 ve YOLOv5 gibi gerçek zamanlı sınıflandırma yöntemleri ve CNN-RCNN gibi daha geleneksel evrimsel sinir ağlarını kullanarak, çok yönlü bir analiz sunulmaktadır. Bu sinir ağları, oluşturulan spesifik veri seti üzerinde eğitilerek, yangın tespiti için optimize edilmektedir. Eğitim sürecinde, veri setinde yer alan benzer veya yinelenen görüntülerin sistem performansını olumsuz etkileyecek bir durum oluşturmaması için filtreleme işlemi de gerçekleştirilmektedir.

Son olarak, eğitilmiş modellerin test edilmesi ve farklı ağların karşılaştırılması, bu yaklaşımın etkinliğini ve doğruluğunu değerlendirmek için kritik bir adımdır. Performans metrikleri, tespit doğruluğu, yanlış pozitif oranı ve işlem süresi gibi kriterlerle birlikte sunulduğunda, bu çalışma orman yangınlarının erken tespiti ve müdahalesi için güçlü bir araç oluşturabilir.

#### 4.1. YAPAY ZEKA

John McCarthy, yapay zeka (Artificial Intelligence-AI) alanının kurucularından biri olarak kabul edilir ve bu alanda yaptığı çalışmalar, yapay zeka terminolojisinin ve teorilerinin gelişimine büyük katkı sağlamıştır. McCarthy'nin bu tanımı, yapay zekanın esas amacını belirginleştirir: "akıllı" bilgisayar programları yaratmak ve bu sürecin yalnızca biyolojik gözlemlere dayanmak zorunda olmadığını vurgular.

Bu tanım, yapay zekanın birçok uygulaması için temel oluşturur. Örneğin, doğal dil işleme, görüntü tanıma, oyun teorisi, otomatik planlama ve daha birçok alandaki yapay zeka uygulamaları, McCarthy'nin tanımladığı gibi, akıllı programlar yaratmayı amaçlar. Ayrıca, yapay zekanın insan zekasını anlamak için bir araç olarak kullanılabilmesi, fakat bunun için biyolojik mekanizmaları tamamen taklit etmek zorunda olmadığı fikri, yapay zekanın çok daha geniş bir potansiyele sahip olduğunu gösterir.

Bu geniş potansiyel, günümüzde derin öğrenme, nöral ağlar, takviye öğrenmesi gibi çeşitli yöntemlerle kendini göstermektedir. Örneğin, yukarıda bahsettiğiniz orman yangını tespiti çalışması da, yapay zekanın "akıllı programlar" oluşturmak için kullanılabilmesi uygulamalardan biridir. McCarthy'nin bu kapsamlı tanımı, yapay zekanın farklı alanlarda, farklı uygulamalara sahip olabileceğinin altını çizer [25].

Alan Turing'in "Computing Machinery and Intelligence" makalesi, yapay zeka alanının temellerini atmıştır. Turing, makalenin içerisinde bilgisayarların düşünüp düşünemeyeceği sorusunu gündeme getirerek, yapay zeka çalışmalarının filozofik, teknolojik ve etik boyutlarını tartışmaya açmıştır. Turing testi, bir insanın metin tabanlı bir arayüz aracılığıyla bir bilgisayar ve bir insan arasında ayırım yapamayacağı bir durumu tasvir eder. Eğer bir bilgisayar, bir insan sorgulayıcıyı kandırabilirse, o zaman bilgisayarın "zeki" olduğu kabul edilebilir

Turing testi, yapay zeka araştırmalarının ilk yıllarında önemli bir ölçüm aracı olarak kabul görmüş, ancak zaman içerisinde birçok eleştiri ve sınırlılık ile karşılaşmıştır. Örneğin bir bilgisayarın insan gibi davranabilmesi, onun insan zekasının tüm

yönlerini anladığı veya taklit ettiği anlamına gelmeyebilir. Ayrıca, Turing testi'nin yalnızca doğal dil işleme yeteneklerini ölçmeye yatkın olması, yapay zekanın diğer alanlarını göz ardı etmektedir. [27].

Yine de Turing testi, yapay zeka, dilbilim, felsefe ve hatta etik gibi farklı disiplinler için önemli bir kavramsal araç olarak kalmıştır. Makine öğrenmesi, derin öğrenme, nöral ağlar gibi daha teknik yaklaşımlar geliştirilse de, Turing'in sorduğu temel soru ve sunduğu test, yapay zekanın felsefi ve etik boyutlarını anlamak için hâlâ kullanılan bir çerçeve oluşturur.

Stuart Russell ve Peter Norvig'in "Yapay Zeka: Modern Bir Yaklaşım" kitabı, yapay zeka çalışmalarının bir temel taşıdır ve bu alandaki eğitim için sıklıkla kullanılan bir ders kitabıdır. Kitap, yapay zekanın farklı yönlerini anlayabilmek için dört farklı yaklaşım ya da hedef sunmaktadır [26].

1. İnsan Yaklaşımı İnsan Gibi Düşünen Sistemler; bu yaklaşım, yapay zekanın insan zekasını simüle etmesi gerektiğini savunur. Yani bir yapay zeka sistemi, insanların problem çözme, öğrenme, anlama gibi mental işlevlerini taklit edebilmelidir.

2. İnsan Yaklaşımı İnsan Gibi Hareket Eden Sistemler; bu yaklaşımda amaç, bir yapay zeka sisteminin insan gibi davranabilmesidir. Turing Testi bu kategoride bir ölçüt olarak sıklıkla kullanılır.

3. İdeal Yaklaşım Rasyonel Düşünen Sistemler; burada odak, bir yapay zeka sisteminin mantıklı ve etkin bir şekilde düşünebilmesi üzerinedir. Yani sistem, mevcut bilgilerini kullanarak mantıklı çıkarımlar yapabilmelidir.

4. İdeal Yaklaşım Rasyonel Hareket Eden Sistemler; bu yaklaşım, bir yapay zeka sisteminin belirli bir amacı etkin bir şekilde yerine getirebilmesi için rasyonel davranabilmesi gerektiğini savunur. Yani, sistem, bir dizi eylem arasından en iyi sonucu verecek olanı seçebilmelidir.

Bu dört kategori, yapay zeka arařtırmalarının ve uygulamalarının farklı yönlerini temsil eder ve birçok modern yapay zeka sistemi bu kategorilerden bir veya daha fazlasına hitap edebilir. Örneđin, bir yapay sinir ađı hem görüntüleri tanıyabilir (insan gibi düşünen sistemler) hem de satranç oyununda en iyi hamleyi yapabilir (rasyonel düşünen sistemler).

Yıllar içinde yapay zeka, çeşitli kamusal tanıtım döngülerinden geçmiş olsa da, OpenAI tarafından piyasaya sürülen ChatGPT'nin bile en şüpheli bireyler için bir dönüm noktası olabileceđi anlaşılmaktadır. Daha önceden generatif yapay zekada büyük etkiler yaratmış olsa da, bu atılımlar çoğunlukla bilgisayar görüşü alanında meydana gelmiştir. Ancak şu anki ilerleme, doğal dil işleme odaklıdır. Sadece dil deđil, generatif modeller aynı zamanda yazılım kodları, moleküller, doğal görüntüler ve diđer çeşitli veri türlerinin dilbilgisini de öğrenebilir [25].

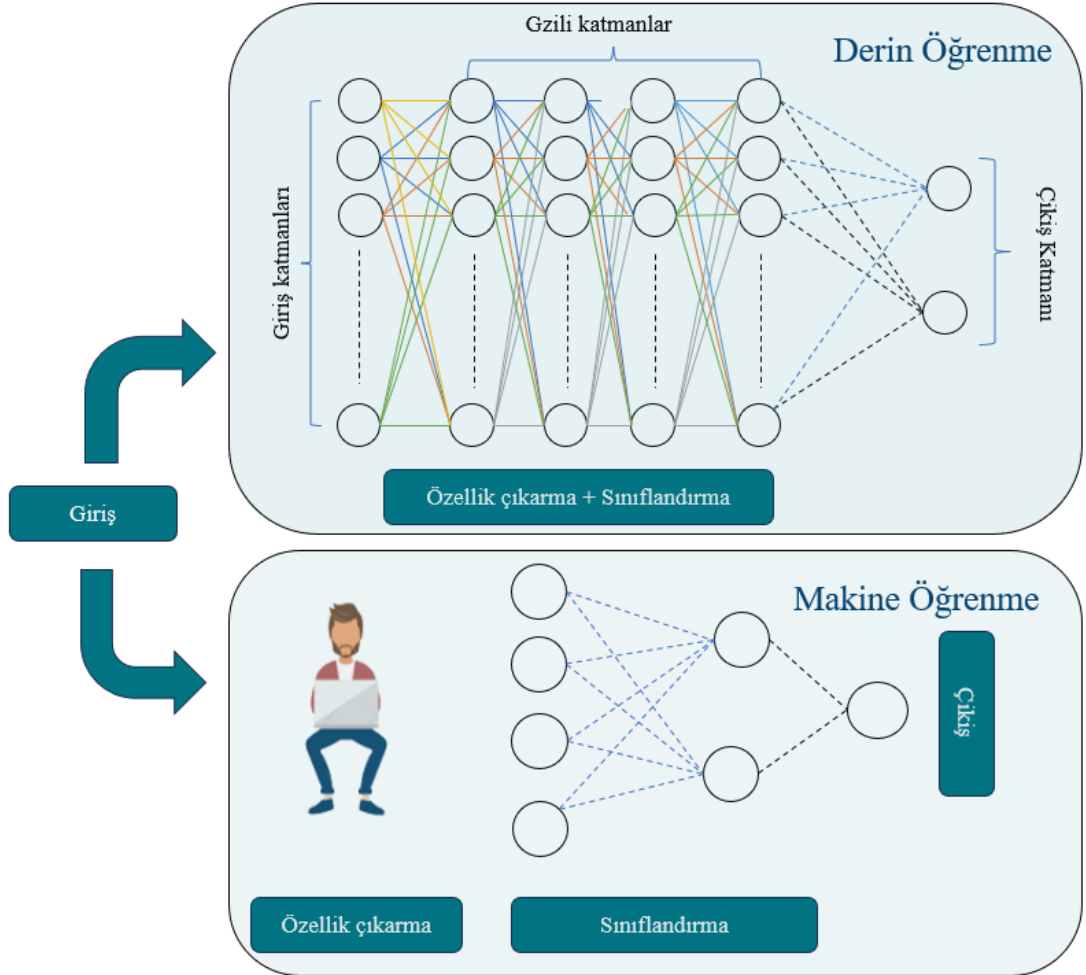
Şekil 4.1'de yapay zeka, derin öğrenme ve makine öğrenmesi arasındaki temel farklılıđı aydınlatmaktadır. Yapay zeka, geniş bir kapsamda akıllı sistemlerin oluşturulmasını amaçlarken, makine öğrenmesi veriye dayalı öğrenme ve karar verme süreçlerine odaklanır. Derin öğrenme ise, makine öğrenmesinin bir alt kategorisi olarak, derin sinir ađlarını kullanarak veri analizi yapar ve karmaşık özelliklerin tanımlanmasına yöneliktir.



Şekil 4.1. Yapay zeka, makine öğrenmesi ve derin öğrenme arasındaki farklar.

## 4.2. DERİN ÖĞRENME VE MAKİNE ÖĞRENMESİ

Derin öğrenme ve makine öğrenmesi sık sık birbiri yerine kullanılsa da, aralarındaki farklar açıklığa kavuşturulmalıdır. Yukarıda da vurgulandığı üzere, her ikisi de yapay zekanın alt dallarıdır; fakat derin öğrenme, makine öğreniminin spesifik bir türüdür. Derin öğrenme, esasen sinir ağlarını temel alır. "Derin" ifadesi, giriş ve çıkış katmanlarını da içerecek şekilde üçten fazla katmana sahip bir sinir ağı anlamına gelmekte olup, bu durum onu bir derin öğrenme algoritması olarak nitelendirilir. Bu kavram Şekil 4.2'de gösterilmektedir.



Şekil 4.2. Derin öğrenme makine öğrenmesi diyagramı.

Derin öğrenme ile makine öğrenmesi arasındaki çarpıcı fark, algoritmaların öğrenme süreçlerinin nasıl işlediğidir. Derin öğrenme, özellik çıkarım aşamalarını büyük

oranda otomatize eder, bu da elle insan müdahalesine olan ihtiyacı azaltır ve daha geniş veri setlerinin kullanılabilirliğini artırır. Lex Fridman'ın da vurguladığı gibi, derin öğrenmeyi "ölçeklenebilir makine öğrenmesi" olarak değerlendirebiliriz. Klasik, yani "derin olmayan" makine öğrenmesi ise, özellik hiyerarşisinin tanımlanması gibi konular için insan uzmanlığına daha fazla bağımlıdır ve genellikle daha yapılandırılmış veri setleri gerektirir.

"Derin" makine öğrenmesi, algoritmasını beslemek için etiketli veri setlerini kullanabilir, bu da denetimli öğrenme olarak adlandırılır. Fakat bu tür bir veri seti şart değildir. İlk haliyle, yani metin veya görüntüler gibi yapılandırılmamış verileri kavrayabilir ve farklı veri türlerini birbirinden ayırt eden özelliklerin hiyerarşisini otomatik olarak saptayabilir. Klasik makine öğrenmesinin aksine, veri işleme aşaması için insan müdahalesi zorunlu değildir. Bu, makine öğrenmesini daha yenilikçi yollarla ölçeklendirmemize imkan tanır [25], [27].

#### **4.2.1. Derin Öğrenme Tabanlı Görüntü İşleme**

Derin öğrenme odaklı görüntü işleme konusunda yürütülen kesintisiz çalışmalar ve araştırmalar, robotik, havacılık ve insansız hava araçları gibi sektörlerde kayda değer ilerlemeleri tetiklemektedir. Derin öğrenme algoritmalarının görüntü işleme sahasına entegrasyonu, makinelerin görsel verileri sofistike bir şekilde anlamasını, yorumlamasını, zeki kararlar almasını ve çevresiyle proaktif bir etkileşimde bulunmasını mümkün kılmaktadır. Bu avantajlı teknoloji, robotların ve İHA'ların işlevselliğini ve güvenilirliğini belirgin bir şekilde güçlendirmektedir. Gelecekte, bu disiplinlerde daha kapsamlı gelişmelerin ve uygulamaların hayata geçirileceği öngörülmekte; bu da gelişmiş robotik sistemler, akıllı havacılık platformları ve diğer yüksek seviyeli uygulamalar için yeni fırsatlar oluşturmaktadır.

#### **4.2.2. Derin Öğrenme Mimarileri**

Görüntü sınıflandırma, bir giriş görüntüsüne dayanarak, bu görüntüyü en uygun biçimde ifade eden bir kategori (örneğin, kedi, köpek vs.) ya da bu kategorilerin olasılıklarını çıktı olarak sunar. Bu, insanlar için, doğumdan itibaren kazandığımız ve

yetişkin oldukça doğal ve çabasız bir biçimde gerçekleştirdiğimiz bir yetidir. İçinde bulunduğumuz ortamı ve çevresel unsurları, düşünmeye dahi gerek kalmadan, hızlı ve kesin bir biçimde tanımlayabiliriz. Bir görüntüyle karşılaştığımızda ya da sadece çevremize göz attığımızda, genellikle sahneleri anında tanır ve nesnelere farkına varmaksızın etiketler atayabiliriz. Hızlı desen tanıma, önceki deneyimlere dayalı genellemeler yapma ve farklı görsel ortamlara kolayca uyum sağlama gibi yetenekler, maalesef ki makine meslektaşlarımızla paylaşmadığımız özelliklerdir [28], [29].

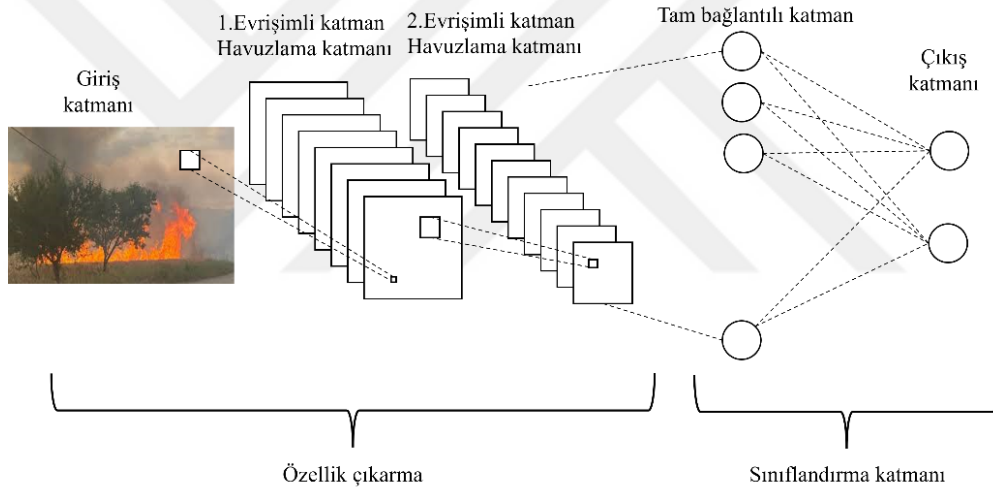
Derin öğrenme ağlarına (Deep Neural Networks - DNN) dair en tanınmış modeller arasında Uzun-Kısa Süreli Bellek Ağları (Long Short-Term Memory - LSTM), Evrişimli Sinir Ağları (Convolutional Neural Networks - CNN) ve Tekrarlayan Sinir Ağları (Recurrent Neural Networks - RNN) yer almaktadır. Ek olarak, görüntüler içerisindeki nesnelere spesifik bir şekilde tanımlamak için tasarlanmış olan ağlar da mevcuttur. Örneğin, belirli bir bölgede yer alan nesnelere saptamak amacıyla, evrişimli sinir ağları temelli bölgesel evrişimli sinir ağları (Region-based Convolutional Neural Networks - R-CNN) ve bu modelin daha gelişmiş varyasyonları olan hızlı R-CNN (Fast R-CNN) ile daha hızlı R-CNN (Faster R-CNN) gibi modeller tasarlanmıştır. Bunlara ek olarak, sınıflandırma, nesne izleme ve regresyon analizi gibi işlemlerde kullanılabilen YOLO (You Only Look Once) algoritması da bulunmaktadır. Ağ yapılandırması, belirli bir amaca hizmet etmek üzere gerçekleştirilmektedir. Bu amaçlar arasında görüntülerin sınıflandırılması, görüntü içindeki nesnelere tanımlanması ve nesnelere öne çıkarılması gibi işlemler bulunmaktadır [29].

Ancak, bu ağların en yaygın ve tartışmasız en popüler uygulama sahası görüntü işlemdir. Görüntü işleme bağlamında, bu tür evrişimli sinir ağlarının (CNN) görüntü sınıflandırması için nasıl kullanılacağı sonraki kısımlarda açıklanmaktadır.

#### **4.2.3. Evrişimli Sinir Ağı**

Evrişimli sinir ağları, bir tür derin öğrenme modeli olarak, sadece ses işleme ve biyolojinin matematikle birleşimi değil, aynı zamanda bilgisayar görüşü alanında

kayda değer yenilikler sunmaktadır. 2012 yılı, sinir ağlarının etkisini gösterdiği dönem olarak kabul edilir. Bu yıl, Alex Krizhevsky ImageNet yarışmasını kazanmak için sinir ağlarını kullanmış ve sınıflandırma hatasını %26'dan %15'e indirerek etkileyici bir gelişme sağlamıştır. Bu dönemden itibaren, çok sayıda şirket temel hizmetlerinde derin öğrenmeyi entegre etmektedir. Örneğin, Facebook otomatik etiketleme, Google görsel arama, Amazon ürün önerileri, Pinterest kullanıcıya özgü içerik ve Instagram görsel arama altyapısında sinir ağlarından faydalanmaktadır [29]–[31]. Bir CNN ağı, şu katmanlardan meydana gelmektedir. Giriş katmanı, evrişimli katman (convolutional), havuzlama katmanı (pooling), tam bağlantılı katman (Fully Connected, FC), Softmax veya lojistik katman ve çıkış katmanı gibi bileşenlerden oluşmaktadır [29]. Şekil 4.3’de bir CNN'nin genel yapısı verilmektedir.



Şekil 4.3. Evrişimli Sinir Ağı.

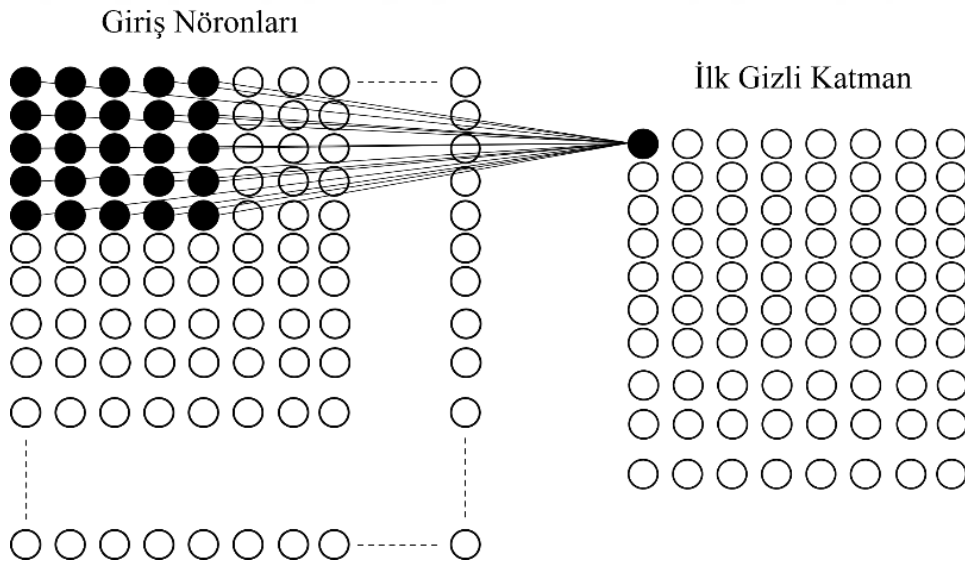
#### 4.2.3.1. Giriş Katmanı ve Çıkış Katmanı

Bilgisayar, bir görüntüyü giriş olarak aldığı anda, bu görüntüyü bir piksel değerleri matrisi olarak ele alır. Görüntünün çözünürlüğü ve boyutlarına bağlı olarak, bu matris genellikle 32x32x3 boyutlarında bir sayılar kümesi olarak değerlendirilir. Burada '3' (RGB - Red-Green-Blue) renk kanallarını temsil etmektedir. Her bir piksel için yoğunluk, 0 ile 255 arasında bir değer olarak atanır. Bu sayılar, insanlar için belki anlamsız olabilir, ancak bilgisayarlar için görüntü sınıflandırma işlemlerinde kullanılabilir temel giriş verileridir. Temel amaç, bu sayılar matrisini bilgisayara

sunmaktır; bilgisayar da karşılığında görüntünün belirli bir kategoriye ait olma olasılığını temsil eden bir dizi sayı ile yanıt verecektir [29].

#### 4.2.3.2. Evrişimli Katman

CNN ağının ilk katmanı daima evrişim katmanıdır. Bu katmana giriş yapılacak olan ilk özellik, girişin boyutlarıdır. Örneğin,  $32 \times 32 \times 3$  boyutlarında piksel değerleri matrisi giriş olarak alınabilir. Buradaki filtre ise, ağırlık ya da parametre olarak adlandırılan sayılar matrisidir. Önemli bir nokta, bu filtre derinliğinin, girişin derinliği ile eşleşmesi gerektiğidir. Bu matematiksel işlemlerin doğru olmasını sağlar. Diyelim ki bu filtre (ya da çekirdek)  $5 \times 5 \times 3$  boyutlarında, bu durum Şekil 4.4'de temsil edilebilir. Filtre, giriş görüntüsü üzerinde kaydırılırken ya da döndürülürken, filtredeki değerleri görüntünün ilgili piksel değerleri ile çarpır (element-wise çarpma işlemi yapılır). Bu çarpımların toplamı alınır. Matematiksel olarak toplamda 75 adet çarpma işlemi yapılacaktır. Sonuç olarak elde ettiğiniz bu sayı, sadece filtrelerin sol üst köşedeki varlığını temsil eder.



Şekil 4.4. Evrişimli katman.

Bu işlem girişin tüm konumları için tekrar edilir (yani filtre bir birim sağa taşınır, ardından bir birim daha sağa taşınır ve işlem bu şekilde devam eder). Her bir benzersiz giriş konumu, bir sayı üretir. Filtreyi tüm konumlar için kaydırduğunuzda,

sonuç olarak 28x28x1 boyutunda bir matris elde edilir. Buna aktivasyon haritası veya özellik haritası adı verilir. 28x28 boyutunda bir matris elde edilmesinin sebebi, 32x32 boyutlu bir giriş görüntüsünde 5x5 boyutunda bir filtre ile 784 farklı konum olabileceğidir. Bu elde edilen sayılar, 28x28 boyutunda bir matrise yerleştirilir.

#### **4.2.3.3. Doğrusallaştırılmış Birim**

Bu katman, evrimsal katmanını takip eder. Aktivasyon katmanı, aynı zamanda ReLu (Rectified Linear Unit - ReLu) olarak da bilinir ve evrim katmanının çıkış matrisindeki negatif değerleri sıfıra çevirirken pozitif değerleri olduğu gibi bırakır. ReLu'nun temel amacı, sinir ağının daha hızlı ve etkili bir biçimde eğitilmesine yardımcı olmaktır. Ek olarak, CNN'ler sıklıkla görüntü işleme amaçlı kullanıldığından, matris değerlerinin 0-255 arasında olması önerilir [32].

#### **4.2.3.4. Havuzlama Katmanı**

Havuzlama katmanı, ReLu katmanının hemen sonrasında yer alır ve bu katman, ReLu katmanı tarafından üretilen çıkış görüntüsünün genişlik ve yükseklik değerlerini küçültür. Ancak bu süreç, ağın derinlik boyutunu etkilemez. Havuzlama katmanı, örnekleme işlemlerini gerçekleştirir ve bu sayede sonraki katmanın yapması gereken hesaplama miktarını azaltır. Ek olarak, bu katman, bilgi kaybını minimize ederek ağın ezberlemesini engeller. Havuzlama işlemleri sırasında çeşitli filtreler uygulanır; bunlar arasında maksimum değeri seçme, minimum değeri seçme ve değerlerin ortalamasını alma gibi yöntemler bulunur. Genellikle, maksimum değeri alma yöntemi daha yüksek performans sunduğu için tercih edilir [30], [31], [33].

#### **4.2.3.5. Tam Bağlantılı Katman**

Yüksek seviyeli özelliklerin tespit edilebildiği bir ağ yapısında, tam bağlantılı bir katmanın eklenmesi işi tamamlar. Bu katman, giriş olarak bir önceki katmandan (bu, bir evrimli, ReLu veya havuzlama katmanı olabilir) gelen hacmi alır ve programın sınıflandırmak istediği sınıf sayısına bağlı olarak N-boyutlu bir vektör çıktısı üretir. Örneğin, eğer rakam sınıflandırma yapmayı amaçlıyorsanız, N değeri 10 olacaktır,

çünkü 10 farklı rakam mevcuttur. Bu N-boyutlu vektör, her bir elemanı ile belirli bir sınıfın olasılığını gösterir. Örneğin, bir rakam sınıflandırıcı için üretilen çıktı vektör [0.1, 0.1, 0.75, 0, 0, 0, 0, 0, 0, 0.05] ise, bu çıktı görüntünün %10 ihtimalle 1, %10 ihtimalle 2, %75 ihtimalle 3 ve %5 ihtimalle 9 olduğunu gösterir. (Not: Çıktıyı farklı şekillerde temsil etmek de mümkündür, ancak burada sadece softmax yaklaşımı ele alınmaktadır.

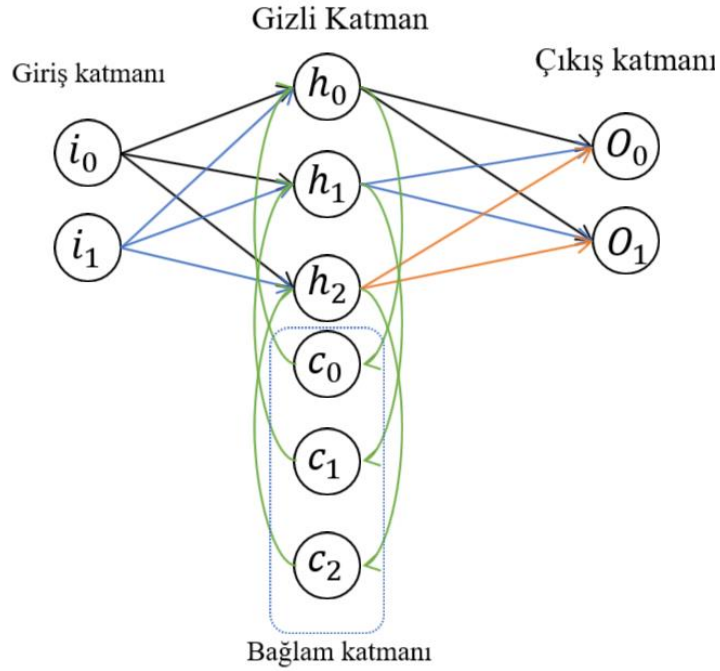
Tam bağlantılı katman, önceki katmandan gelen çıktıyı (yüksek seviyeli özelliklerin aktivasyon haritalarını temsil eder) analiz eder ve hangi özelliklerin belirli bir sınıfla en yüksek ilişkiye sahip olduğunu belirler. Örneğin, eğer program bir görüntünün bir köpek olduğunu tahmin ediyorsa, pençe veya dört ayak gibi özellikleri yüksek aktivasyon değerleriyle gösterecektir. Benzer şekilde, eğer program bir görüntünün bir kuş olduğunu tahmin ediyorsa, kanat veya gaga gibi özellikler yüksek aktivasyon değerlerine sahip olacaktır. Temelde, tam bağlantılı katman, belirli bir sınıf ile en güçlü ilişkili yüksek seviyeli özellikleri tespit etmek için kendi içerisinde özel ağırlıklar kullanır ve bu ağırlıklar ile önceki katmanın çıktısı arasında matematiksel işlemler yapar [29], [34]–[36].

#### **4.2.4. Tekrarlayan Sinir Ağı**

Tekrarlayan sinir ağları (RNN), derin öğrenme mimarilerinin temelini oluşturan önemli ağ yapılarından biridir. Geleneksel çok katmanlı ağlarla döngüsel ağlar arasındaki kritik fark, Şekil 4.5'de gösterildiği üzere döngüsel ağların sadece ileri beslemeli bağlantılar yerine önceki katmanlara (veya aynı katmana) geri beslemeli bağlantılar da içerebilecek olmasıdır. Bu geri beslemeli yapı, RNN'lerin geçmişe yönelik girişlerin hafızasını tutmasına ve zamanla ilişkili problemleri modellemesine olanak tanır [36].

RNN'ler, çeşitli mimari yapıları barındırır. Örneğin uzun-kısa süreli bellek ağları (LSTM). Ağ içerisindeki ana ayırt edici özellik, geri beslemeli yapıdır. Bu, gizli katmandan, çıktı katmanından ya da her ikisinin bir kombinasyonundan kaynaklanabilir [36].

Eđitim s¼recinde, RNN'ler zaman boyutu aılarak standart geri yayılım tekniđi veya onun bir varyasyonu olan zamana yayılan geri yayılım ile eđitilebilir. Uygulama ¼rnekleri arasında konuřma ve el yazısı tanıma yer almaktadır [36].



řekil 4.5. Tekrarlayan Sinir Ađı.

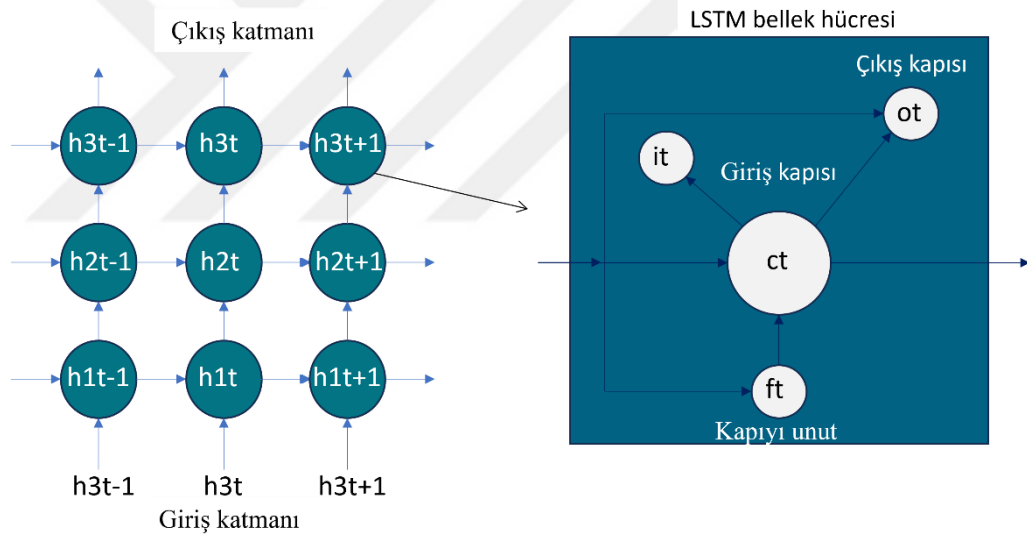
#### 4.2.5. Uzun Kısa S¼reli Bellek Ađları

Uzun kısa s¼reli bellek ađları (LSTM), 1997 yılında Hochreiter ve Schmidhuber tarafından geliřtirilmiř olup, son yıllarda eřitli uygulama alanlarında bir RNN mimarisi olarak pop¼lerliđini artırmıřtır. G¼nl¼k hayatta kullandıđınız cihazlarda, ¼rneđin akıllı telefonlarda, LSTM'lerle karřılařabilirsiniz. IBM Watson® platformu, konuřma tanıma gibi ıđır aan uygulamalarda LSTM teknolojisi kullanmaktadır [37]–[39].

LSTM, klasik sinir ađı mimarilerinden farklı bir yapı sunarak "bellek h¼resi" kavramını getirmiřtir. Bu bellek h¼resi, giriřlere bađlı olarak deđerini kısa veya uzun s¼reliđine saklayabilme kapasitesine sahiptir. Bu sayede h¼re, yalnızca son hesaplanan deđer deđil, aynı zamanda ¼nemli olan diđer bilgileri de hafızasında tutabilir [37], [38].

LSTM'nin bellek hücresi, bilginin hücreye nasıl gireceğini ve çıkacağını kontrol eden üç farklı kapı mekanizması içerir. Giriş kapısı, yeni bilginin hücreye ne zaman kaydedileceğini belirler. Unutma kapısı, mevcut bilgilerin ne zaman silineceğini kontrol eder, böylece hücre yeni verilere odaklanabilir. Çıkış kapısı ise hücrede saklanan bilginin ne zaman kullanılacağını düzenler. Ayrıca, her kapıyı kontrol eden ağırlıklar da bulunmaktadır. Bu ağırlıklar, zamana yayılan geri yayılım adı verilen eğitim algoritmasıyla, elde edilen ağ hatalarına göre optimize edilir [37]–[39].

Son dönemde, CNN'ler ve LSTM'ler bir araya gelerek görüntü ve video altyazılama sistemlerini mümkün kılmıştır. Bu sistemlerde CNN, görüntü veya video işleme görevini üstlenirken, LSTM ise CNN'nin çıktısını doğal dile çevirmek için eğitilmiştir [29], [38]. Şekil 4.6'da LSTM yapısına dair detaylar verilmektedir.

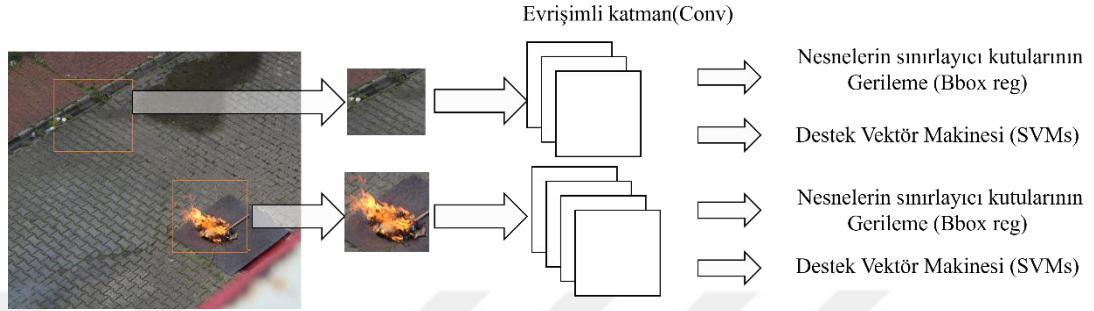


Şekil 4.6. LSTM Sinir Ağı.

#### 4.2.6. Bölge Tabanlı Evrişim Sinir Ağı

Bölge tabanlı evrişimli sinir ağı (R-CNN), görüntülerde yer alan nesnelerin sınıflarını ve bu nesnelerin sınırlayıcı kutularını tespit etmek amacıyla kullanılır. Tek bir görselde birden fazla nesne olduğunda, standart CNN mimarileri yetersiz kaldığı için R-CNN mimarisi geliştirilmiştir [31], [35], [40], [41]. Şekil 4.7'de R-CNN mimarisi detaylandırılmıştır.

R-CNN'nin temel çalışma prensibi iki aşamada gerçekleşir. İlk adımda, nesnelerin yer alabileceği bölgeler "seçici arama" (Selective Search) algoritması ile belirlenir. İkinci adımda ise, her bir belirlenen bölge CNN modeline giriş olarak sağlanır ve sınıf tahmini ile sınırlayıcı kutu oluşturma işlemi yapılır [31], [35], [40], [41].

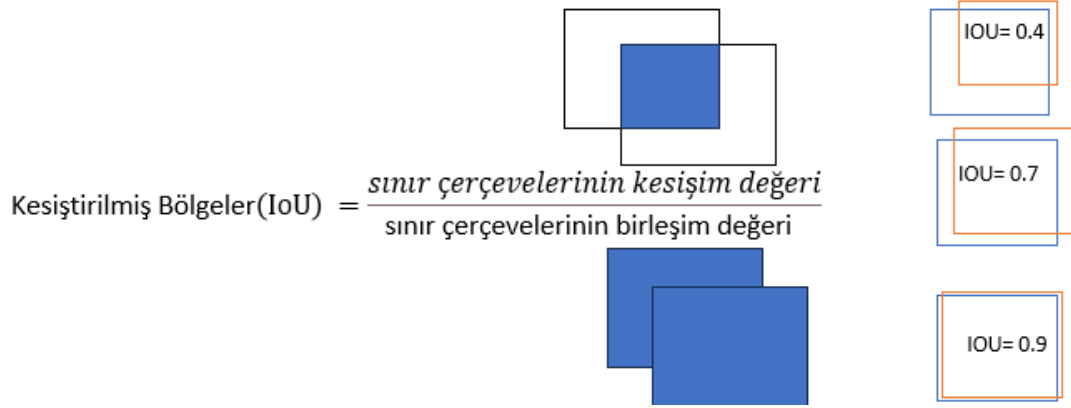


Şekil 4.7. R-CNN Mimarisi.

"Seçici arama" algoritması, görüntüde tespit edilecek bölgeleri saptamak için kullanılır. İlk olarak görselden küçük bölgeler belirlenir, ardından benzer özellikteki bölgeler birleştirilerek daha büyük bölgeler oluşturulur. Bu işlem, görseldeki nesnelerin etrafında kümelenmiş bir yapı oluşturana kadar tekrarlanır [31], [35], [40], [41].

R-CNN, belirlenen bölgelerin adaylarını tespit etmek için "Seçici Arama" metodunu kullanır. Her bir bölge adayı, farklı bir CNN modeline giriş olarak sunulur. Bu işlemler sonucunda yaklaşık olarak 2000 farklı bölge elde edilir ve her biri için farklı bir CNN modeli kullanılır. Nesnelerin sınıfları, bu modellerden elde edilen özelliklerle destek vektör makineleri (Support Vector Machine - SVM) kullanılarak belirlenir ve sınırlayıcı kutular ise regresyon analizi ile tespit edilir [31], [35], [40], [41].

Şekil 4.8'de "Kesiştirilmiş Bölgeler" (IoU) skoru, tahmin edilen sınırlayıcı kutuların doğruluğunu değerlendirmek için kullanılır. Tahmin edilen ve gerçek sınırlayıcı kutuların kesişimi ile birliği oranı olarak hesaplanır[33].



Şekil 4.8. Kesiştirilmiş Bölgeler

"Maksimum olmayan bastırma" (Non-Max Suppression) tekniği, doğru sınırlayıcı kutuları seçmek için uygulanır. Bu yöntem ile IoU skoru 0.5'ten büyük olan sınırlayıcı kutular korunurken, diğerleri bastırılır. Eğer bir nesne için birden fazla 0.5'ten büyük IoU skoru olan sınırlayıcı kutu elde edilirse, en yüksek IoU skoruna sahip olan kutu seçilir[33]. Bu süreç Şekil 4.9'da gösterilmektedir.



Şekil 4.9. Maksimum olmayan bastırma uygulama öncesi ve sonrası

#### 4.2.7. Hızlı Bölge Tabanlı Evrişimli Sinir Ağı

Hızlı bölge tabanlı evrişimli sinir ağı (Fast R-CNN) modelleri, özellikle maliyet açısından daha verimli bir yaklaşım sunar. Klasik R-CNN mimarisinde, her bir görüntü için yaklaşık 2000 aday bölge oluşturulur ve her bir bölge için ayrı bir CNN

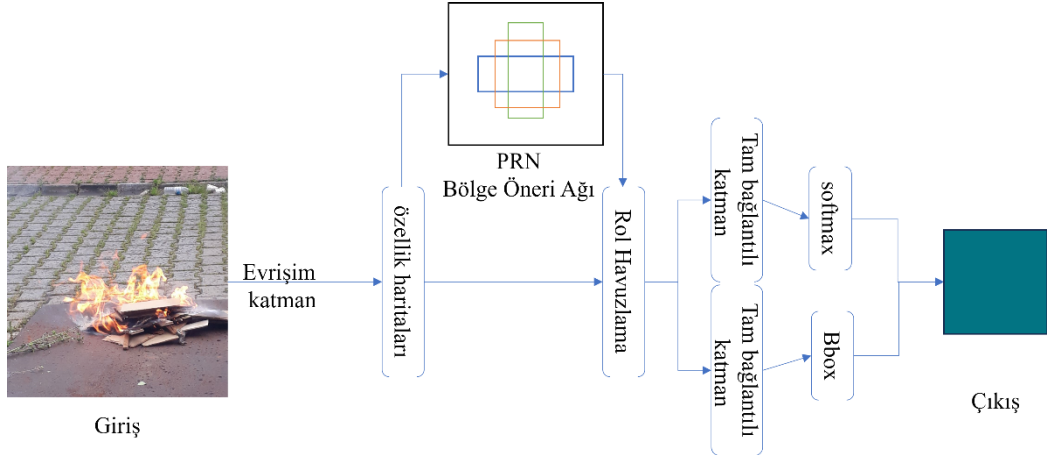
modeli kullanılır. Bu durum, hem yüksek maliyete hem de uzun eğitim sürelerine yol açar. Hızlı R-CNN, bu sorunları çözmek için tek bir CNN modeli kullanarak birden fazla bölgeyi işler. Ayrıca, bu mimaride CNN, SVM ve regresör bileşenleri daha etkin bir şekilde entegre edilmiştir, bu da modelin yüksek performans göstermesini sağlar [42], [43].

Bu mimaride, tüm görüntü ilk olarak bir CNN modelinden geçirilir ve özellik haritaları elde edilir. Daha sonra, bölge önerileri için gerekli olan özellikler bu haritalardan toplanır. Elde edilen özellik haritalarına, boyutlarını küçültmek için maksimum havuzlama (max pooling) işlemi uygulanır. Bu işleme RoI (Region of Interest Pooling Layer) adı verilir. Boyutu küçültülen özellik haritaları, tek boyutlu bir vektöre dönüştürülür ve bu vektör, CNN modeline giriş olarak sağlanır. Softmax fonksiyonu ile bölgedeki nesnenin sınıf bilgisi belirlenir, aynı zamanda nesnenin sınırlayıcı kutu koordinatları da regresyon analizi ile düzeltilir. Bu yaklaşım, klasik R-CNN mimarisine göre yaklaşık 10 kat daha hızlıdır [42], [43].

#### **4.2.8. Daha Hızlı Bölge Tabanlı Evrişimli Sinir Ağı**

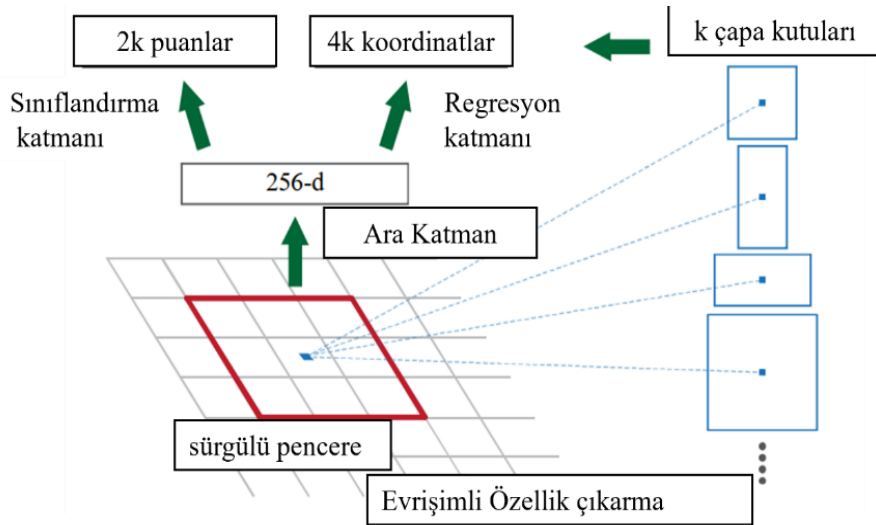
Daha hızlı R-CNN modelinde, RPN (Region Proposal Network) kullanılarak daha hızlı ve hesapsal açıdan daha ekonomik bölge önerileri oluşturulmaktadır. Bu yeni yaklaşım, seçici arama işlemlerinin getirdiği hesapsal yükü önemli derecede azaltır. RPN'nin etkinliği, mevcut çeşitli bilimsel çalışmalarla desteklenmektedir. İşlem sırasında, herhangi bir boyuttaki bir görüntü RPN'ye girdi olarak verilir ve bu görüntü, bir evrişimli sinir ağı (CNN) modeli aracılığıyla işlenir. Eğer seçtiğiniz model VGG16 ya da AlexNet gibi ise, modeli tamamen bağlantılı katmanlar olmadan yeniden yapılandırmanız tavsiye edilir. Bunun nedeni, RPN'nin evrişimli katmanlar tarafından üretilen bir özellik haritasını girdi olarak kullanmasıdır. Bu Şekil 4.10'da gösterilmektedir [5], [35], [41].

RPN'nin kritik bir unsuru olan çapa kutuları (anchor boxes), değişik ölçekler ve en-boy oranlarına sahip kutuları meydana getirir. Bu çapa kutuları, özellik haritası üzerinde kaydırılarak daha etkin bir nesne tespiti gerçekleştirirler [5], [35], [41].



Şekil 4.10. Daha Hızlı Bölge Tabanlı Evrişimli Sinir Ağı.

RPN'de ilk evrişimli katman, varsayılan olarak 3x3 boyutunda bir filtre kullanır ve 512 çıkış kanalına sahiptir. Bu katman, özellik haritasını girdi olarak kabul eder. Bu evrişimli katmanın çıkışı, iki ayrı evrişimli katmana yönlendirilir, ve her biri 1x1 boyutunda filtrelere sahiptir. İlk olarak, sınıflandırma katmanı mevcuttur. Bu katman, kayan pencere üzerinde bir nesnenin olup olmadığını belirler. İkili sınıflandırma işlemi uygulanır. Nesne varsa 1, yoksa 0 olarak işaretlenir. Modelleme sırasında, genellikle sigmoid aktivasyon fonksiyonu ile bir evrişimli katman kullanılır. Bu katmanın çıkış kanal sayısı, genellikle  $2 * 9$  olarak belirlenir. Bu nedenle, sınıflandırma katmanının çıkış kanal sayısı  $2 * k$  olmalıdır. Regresyon katmanı ise, sınıflandırma katmanı tarafından tespit edilen nesnelere için sınırlayıcı kutular çizer.



Şekil 4.11. Bölge Öneri Ağı

Dört koordinat bilgisi sağladığı için, bu katmanın çıkış kanal sayısı genellikle  $4 * 9$  (veya  $4 * k$ ) olarak ifade edilir. Modelleme sırasında, regresyon katmanı için genellikle doğrusal bir aktivasyon fonksiyonu kullanılır. Bütün bu detaylar, Şekil 4.11'de gösterilmektedir [5], [35], [41].

#### **4.2.9. YOLO Ağı**

YOLO algoritması, görüntü ve video analizinde nesne tespiti için en etkili ve yenilikçi yöntemlerden biri olarak kabul edilir. Gerçek zamanlı nesne tespiti konusunda yüksek hızlı tespit kapasitesiyle dikkat çeker ve bu özelliği ile bilgisayar görüşü alanında en hızlı algoritmalarından biri olma özelliğini taşır. YOLO, değişen boyutlara sahip ve yoğun ortamlardaki nesnelere etkin bir şekilde tespit edebilir, fakat çok küçük nesnelere tespitinde bazı kısıtlamalar yaşayabilir. YOLO aynı zamanda, tespit edilen nesnelere konumlarını belirlemek için "Sınırlayıcı Kutu" olarak adlandırılan bir çıkış formatını kullanmaktadır. Bu algoritma, ilk olarak 2016 yılında araştırmacı Joseph Redmon ve ekibi tarafından Washington Üniversitesi'nde tasarlanmış ve o zamandan beri sürekli olarak geliştirilmeye devam etmiştir [44], [45].

#### **4.2.10. YOLOv5 Ağı**

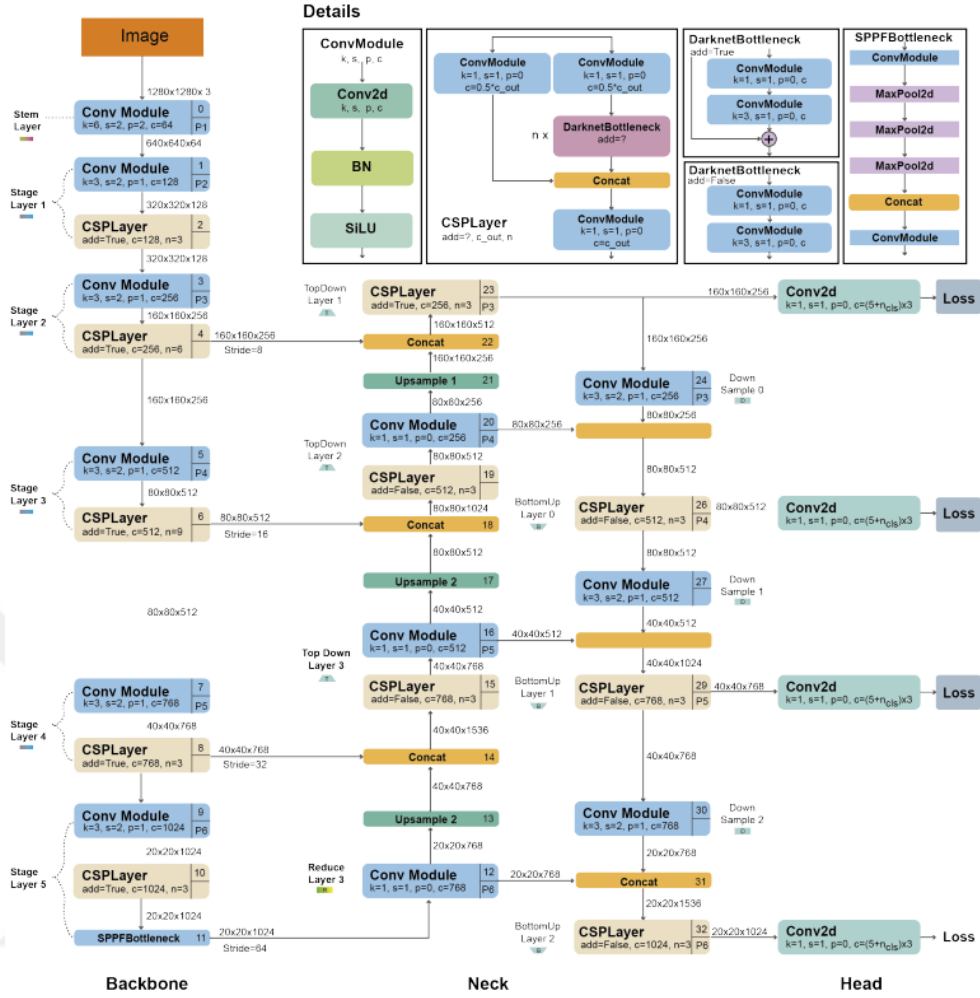
YOLOv5, Ultralytics'in kurucusu ve CEO'su Glen Jocher tarafından 2020 yılında, YOLOv4'ün lansmanından birkaç ay sonra piyasaya sunulmuştur. YOLOv5, YOLOv4'te gerçekleştirilen birçok iyileştirmeyi devralmış fakat yapısını Darknet'in aksine PyTorch üzerinde oluşturmuştur. Bu versiyon, AutoAnchor adı verilen özgün bir Ultralytics algoritmasını barındırır. Bu araç, veri seti ve eğitim parametreleri (örneğin, görüntü boyutları) için en uygun çapa kutularını belirlemek amacıyla tasarlanmıştır. Süreç, veri setindeki etiketlere k-ortalama istatistiksel yöntemini uygulayarak başlar. Bu, genetik evrim algoritmasının başlangıç koşullarını oluşturur. GE algoritması, bu çapa kutularını varsayılan olarak 1000 nesil süresince evrimleştirir. CIOU kaybı ve "En İyi Olası Geri Çağırma" (Best Possible Recall) kriterlerini uygunluk fonksiyonu olarak kullanır [45], [46].

#### 4.2.10.1. YOLOv5 Mimarisi

Sırt bölgesi, adım adım bir evrişim katmanına sahip olan ve büyük pencere boyutu ile hesaplama ve bellek maliyetlerini düşürmeyi hedefleyen Stem katmanı ile başlar. Bu bir modifiye edilmiş CSPDarknet53 yapısını oluşturur. Bu katmanın hemen ardından, giriş görüntüsünden çeşitli özellikleri elde etmek için tasarlanmış evrişim katmanları gelmektedir [46], [47]. SPPF (Spatial Pyramid Pooling Fast) katmanı ve onu takip eden diğer evrişim katmanları, özellikleri farklı ölçeklerde işlerken, yükseltme katmanları özellik haritalarının çözünürlüğünü iyileştirir. SPPF katmanı, farklı ölçeklerdeki özellikleri sabit boyutlu bir özellik haritasına entegre ederek ağır hesaplama hızını artırmayı amaçlamaktadır. Her evrişim işleminin hemen ardından, grup normalizasyonu ve SiLU aktivasyon fonksiyonu kullanılmaktadır [48]. Boyun bölgesi, SPPF ve modifiye edilmiş CSP-PAN teknolojilerini kullanırken, baş kısmının yapısı YOLOv3 ile benzerlik göstermektedir [47].

YOLOv5, çeşitli veri zenginleştirme yöntemleri kullanarak model performansını artırmaktadır. Bu yöntemler arasında mozaik, kopyala-yapıştır [49], rastgele affine dönüşümleri (afin dönüşümler, yani geometrik dönüşümler kategorisine giren bir dizi işlemi ifade eder ve bu işlemler şeklin ölçeklenmesi, döndürülmesi ve taşınması gibi işlemleri içerir), MixUp [50], HSV renk uzayı zenginleştirilmesi, rastgele yatay döndürme ve Albumentations paketinden (Albumentations, bir Python kütüphanesidir ve görüntü veri zenginleştirme işlemleri için kullanılır) diğer çeşitli veri zenginleştirme yöntemlerini içermektedir. Ayrıca, modelin ızgara hassasiyetini iyileştirerek kaçış gradyanlarına karşı daha kararlı bir performans sergilemesini sağlar [47].

YOLOv5, beş farklı ölçekli sürüm sunar: YOLOv5n (nano), YOLOv5s (küçük), YOLOv5m (orta), YOLOv5l (büyük) ve YOLOv5x (çok büyük). Her bir sürümün evrişim modüllerinin genişliği ve derinliği, spesifik uygulama ihtiyaçlarına ve donanım özelliklerine göre ayarlanmıştır. Örneğin, YOLOv5n ve YOLOv5s, kaynak kısıtlı cihazlar için optimize edilmiş hafif modellerdir, oysa YOLOv5x, yüksek performansı hızın maliyetine göre optimize edilmiştir [46], [47]. Şekil 4.12, YOLOv5'in detaylı mimari yapısını göstermektedir



Şekil 4.12. YOLOv5

YOLOv5 projesi açık kaynaklıdır ve Ultralytics tarafından aktif bir şekilde yönetilmektedir. 250'den fazla katkı sağlayan kişi bulunmakta ve proje sürekli olarak yeni iyileştirmelerle güncellenmektedir. YOLOv5'in kullanımı, eğitimi ve dağıtımı oldukça kolaydır. Ultralytics, mobil platformlar için iOS ve android sürümleri dâhil, etiketleme, eğitim ve dağıtım konularında kapsamlı entegrasyonlar sunmaktadır [47].

MS COCO veri kümesinin 2017 test-dev sürümünde yapılan değerlendirmelerde, YOLOv5x modeli, 640 piksel görüntü boyutu için %50.7'lik bir ortalama hassasiyet değeri elde etmiştir. NVIDIA V100 grafik kartı üzerinde, 32'lik bir grup boyutu kullanılarak 200 FPS hıza ulaşılmıştır. 1536 piksel gibi daha büyük bir giriş boyutu ve test zamanı veri artırma ile birlikte, YOLOv5 %55.8'lik bir AP değeri kaydetmiştir [47].

#### 4.2.11. YOLOv8 Ağı

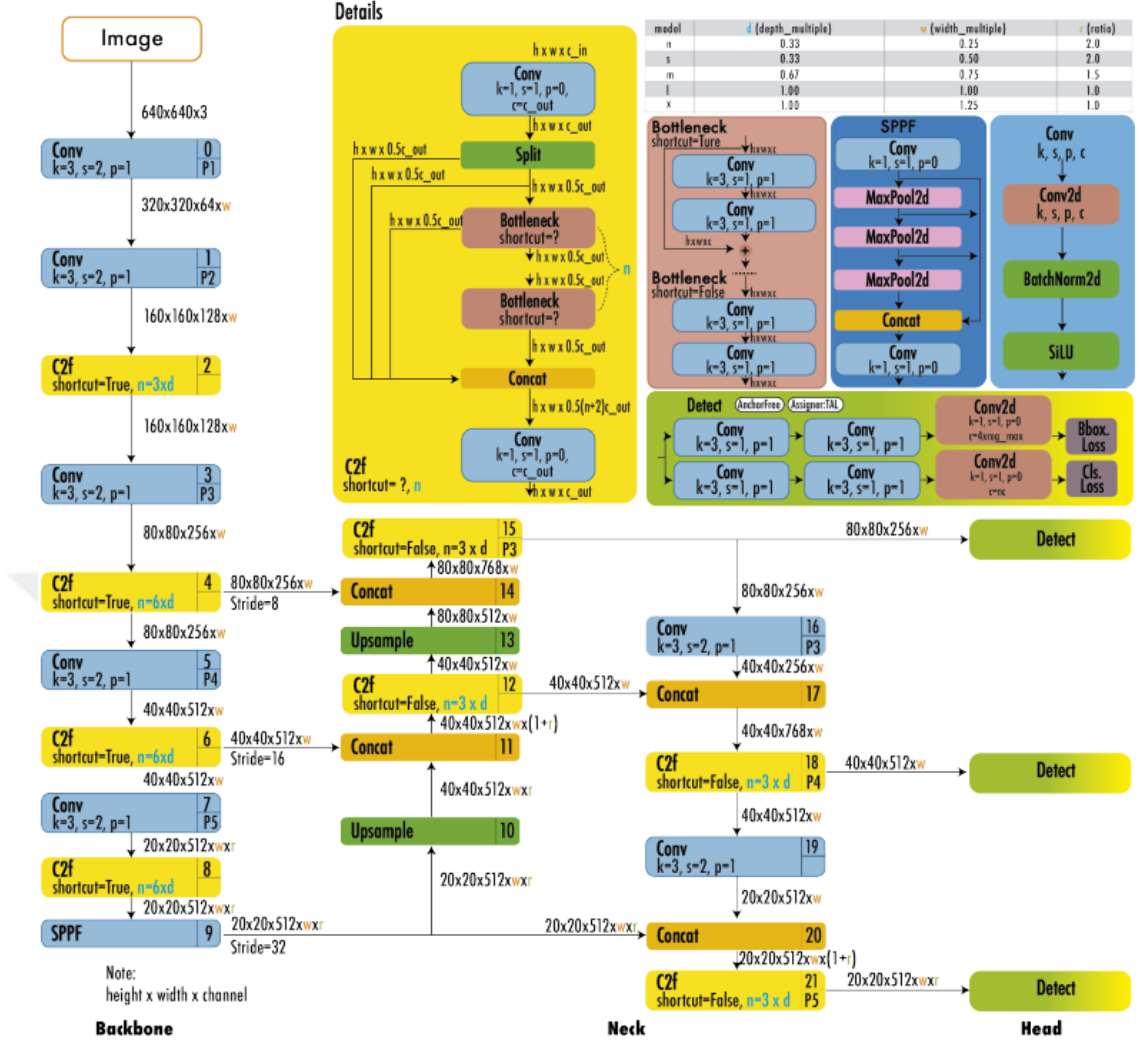
YOLOv8, YOLOv5'i geliştiren Ultralytics şirketi tarafından Ocak 2023 tarihinde piyasaya çıkarılmıştır. Bu sürüm, YOLOv8n (nano), YOLOv8s (küçük), YOLOv8m (orta), YOLOv8l (büyük) ve YOLOv8x (çok büyük) olmak üzere beş farklı boyutta versiyonlar sunar. YOLOv8, nesne tespiti, görüntü bölütleme, pozisyon tahmini, nesne takibi ve sınıflandırma gibi çeşitli görüntü işleme görevlerini desteklemektedir [47].

##### 4.2.11.1. YOLOv8 Mimarisi

Şekil 4.13'de, YOLOv8'in ayrıntılı mimari yapısını sergilenmektedir. YOLOv8, temel yapısını YOLOv5 ile paylaşırken, CSPLayer üzerinde gerçekleştirilen modifikasyonlar sayesinde artık C2f modülü olarak adlandırılmaktadır. Bu C2f modülü (Çapraz Aşamalı Kısmi Darboğazlı Çift Evrişimli (Cross-Stage Partial Bottleneck Dual Convolution)), yüksek seviyeli özellikleri bağlamsal bilgilerle bütünleştirerek algılama doğruluğunu yükseltir [47].

YOLOv8, nesne özelliklerini, sınıflandırmayı ve regresyon görevlerini bağımsız bir şekilde ele alabilen çiftsiz bir başlık modeli kullanmaktadır. Bu yapı, her bir dalın kendi özgü görevine odaklanabilmesine ve bu sayede modelin genel doğruluğunun artmasına imkan tanır. YOLOv8'in çıktı katmanında, nesnenin varlığına dair olasılığı temsil etmek için aktivasyon işlevi olarak sigmoid fonksiyonu tercih edilmiştir. Sınıf olasılıklarını temsil etmek için ise softmax fonksiyonu kullanılmıştır.

YOLOv8, sınırlayıcı kutu kaybı için CIoU [45] ve DFL [51] kayıp fonksiyonlarını kullanmaktadır, ayrıca sınıflandırma kaybı için ikili çapraz entropi yöntemini tercih etmektedir. Bu kayıp fonksiyonları, özellikle küçük ölçekli nesnelerin algılanması konusunda performansı belirgin bir şekilde artırmıştır.



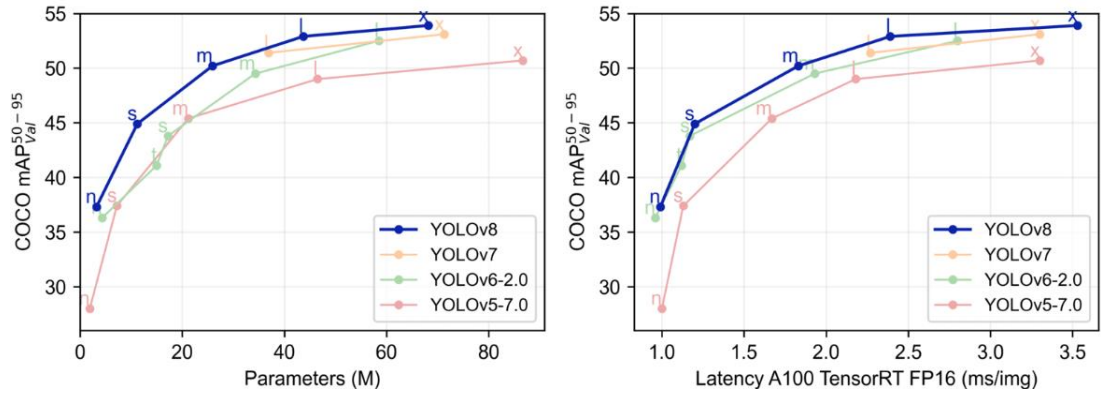
Şekil 4.13. YOLOv8 mimarisi.

YOLOv8, ek olarak YOLOv8-Seg adı verilen bir anlamsal bölütleme modeli de sunmaktadır. Bu modelin temel yapısı, CSPDarknet53 özellik çıkarıcıyı C2f modülü ile güçlendirerek geleneksel YOLO boyun mimarisini yükseltmektedir. C2f modülü sonrasında, giriş görüntüsüne ait anlamsal bölütleme maskelerini tahmin etmeye yönelik iki bölütleme başlığı yer almaktadır. Model, YOLOv8 ile benzer şekilde, beş adet algılama modülü ve bir tahmin katmanına sahiptir. YOLOv8-Seg modeli, yüksek hız ve verimlilik sunarken, çeşitli nesne algılama ve anlamsal bölütleme testlerinde en iyi performansı göstermiştir.

YOLOv8, komut satırı arayüzü (CLI) ile ya da PIP paketi vasıtasıyla yüklenilebilir. Bunun yanı sıra, etiketleme, eğitim ve dağıtım süreçleri için çoklu entegrasyon seçenekleri sunar. MS COCO test-dev 2017 veri kümesinde yapılan

değerlendirmelerde, YOLOv8x modeli, 640 piksel çözünürlüğündeki görüntülerde YOLOv5'in %50.7 AP değerine kıyasla %53.9 AP değerini elde etmiştir. Ayrıca, NVIDIA A100 ve TensorRT kullanılarak 280 FPS hıza ulaşmıştır.

Mimari olarak, YOLOv8 değiştirilmiş bir CSPDarknet53 yapısını benimser. Bu yapıda, C2f modülü, YOLOv5'te yer alan CSPLayer'ı ikame etmektedir. SPPF adlı bir katman, özellikleri sabit boyutlu bir haritaya dönüştürerek hesaplamaların hızlanmasını sağlar. Her bir evrişim işlemi, grup normalizasyonunu ve SiLU etkinleştirmeyi içermektedir. Mimarinin baş kısmı, nesne varlığı, sınıflandırma ve regresyon görevlerini bağımsız bir şekilde işleyecek biçimde ayrılmıştır. Şekil 4.14'de YOLOv8'in önceki sürümlerine kıyasla üstünlüklerini gözler önüne serilmektedir [54].

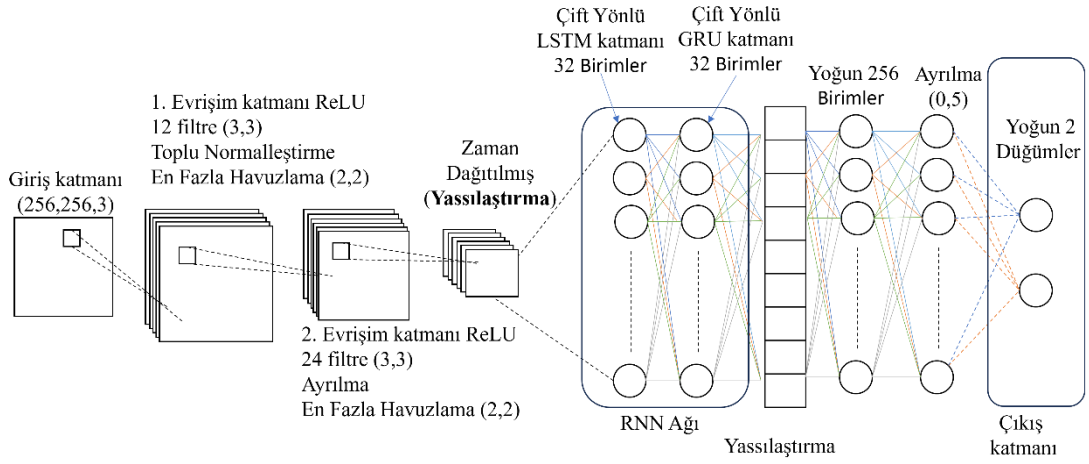


Şekil 4.14. YOLOv8 ve eski sürümlerin karşılaştırılması.

#### 4.2.12. CNN-RCNN Ağı Tasarımı

Şekil 4.15'de, CNN-RCNN'nin detaylı mimarisi gösterilmektedir.

Bu algoritma veya model (Convolutional Neural Network with Recurrent Neural Network - CNN-RCNN)" modelidir. Bu, CNN'lerin ve RNN'lerin güçlerini birleştiren bir derin öğrenme modeli türüdür. CNN'ler, görüntülerden mekansal özellikleri çıkarmak konusunda iyidir, RNN'ler ise ardışık verileri işlemede iyidir. CNN-RCNN modelinin şu katmanları bulunmaktadır:



Şekil 4.15. CNN R-CNN Ağı mimarisi

1. Evrişimli Katman (1.Conv2D): Giriş görüntüsü üzerinde 2D evrişim işlemini gerçekleştirir. 3x3 boyutunda 12 filtre kullanılır. Elde edilen sonuçları etkinleştirmek için ReLU aktivasyon fonksiyonu kullanılır.
2. Topaklama Normalizasyonu (Batch Normalization): Önceki katmanların çıktılarını normalize etmek için kullanılır. Ağın eğitim sürecinin istikrarını ve hızını artırmaya yardımcı olur.
3. En Büyük Havuzlama (Max Pooling): Özellik haritalarının boyutunu belirli bir aralık içinde en büyük değeri seçerek azaltır. Bu modelde 2x2 boyutunda max pooling işlemi uygulanır.
4. 2.Evrişimli Katman (2.Conv2D): 3x3 boyutunda 24 filtre kullanılarak bir kez daha evrişim işlemi gerçekleştirilir.
5. Düşürme (Dropout): Önceki katmanda rastgele olarak belirli bir yüzde (0.2) oranında birimleri atar. Bu, ağın genellemesini iyileştirmek ve aşırı uyumdan kaçınmak için kullanılır.
6. En Büyük Havuzlama (Max Pooling): Bir kez daha 2x2 boyutunda max pooling işlemi uygulanır.

7. Zaman Dağıtılmış Katman (TimeDistributed): Bir sonraki katman olan Flatten katmanına zaman dağıtımı işlemi uygular. Bu, ağın değişken boyutlu verilerle çalışmasını sağlar.
8. İki Yönlü LSTM (Bidirectional LSTM): Zamanlı verilerle çalışan iki yönlü bir LSTM katmanıdır. Önceki ve sonraki durumlar arasındaki ilişkiyi korur. Aşırı uyumu önlemek ve genelleştirmeyi iyileştirmek için %50 dropout oranı kullanılır.
9. İki Yönlü GRU (Bidirectional GRU): İki yönlü bir GRU katmanıdır ve bidirectional LSTM katmanı ile aynı şekilde çalışır. Aşırı uyumu önlemek ve genelleştirmeyi iyileştirmek için %50 dropout oranı kullanılır.
10. Düzleştirme (Flatten): Veriyi düz bir vektöre dönüştürür, bu da tam bağlantılı katmanlara besleme yapmayı mümkün kılar.
11. Tam Bağlantılı Katman 1 (Dense1): Veri üzerinde karmaşık lineer ve non-lineer işlemleri gerçekleştirir. Bu katmanda 256 birim bulunur ve ReLU aktivasyon fonksiyonu kullanılır.
12. Düşürme (Dropout): Rastgele olarak belirli bir yüzde (0.5) oranında birimleri atar.
13. Tam Bağlantılı Katman 2 (Dense2): Veri üzerinde başka karmaşık lineer ve non-lineer işlemleri gerçekleştirir. Bu katmanda 2 birim bulunur ve softmax aktivasyon fonksiyonu kullanılarak olası sınıf dağılımını üretir.

Ardından, modelin kaybı ve performans ölçütü atanır ve Adam optimizer kullanılarak model derlenir. Model ağırlıklarını kaydetmek için bir yol belirlenir ve ModelCheckpoint geri çağırısı kullanılarak ağırlıklar kaydedilir.

Son olarak, model eğitim görüntü seti ve doğrulama görüntü seti verileriyle eğitilir.

### 4.3. VERİ SETİ OLUŞTURULMASI

Farklı türlerdeki orman yangınlarıyla etkin bir şekilde başa çıkabilmek amacıyla, büyük ölçekli orman yangınları, ağaç tepesi yangınları ve yer yangınları gibi çeşitli orman yangını türlerini gösteren geniş bir veri seti oluşturdum. Bu veri seti, çalışmanın hedefleriyle uyumlu olacak şekilde çeşitli kaynaklardan derlenmiş, düzenlenmiş ve ayarlanmıştır. Kullanılan bu veri seti, Kaggle, Mendeley kaynaklarından alınan[52], [53] ve kişisel olarak topladığım 2.654 farklı görüntüyü içermektedir. Şekil 4.16 (a)'da orman yangınları ve yangınsız, sakin orman alanlarına ilişkin modellemeler, bu veri seti üzerinden gösterilmektedir. Ayrıca, kişisel olarak topladığım veriler Şekil 4.16 (b)'de verilmektedir.



(a)

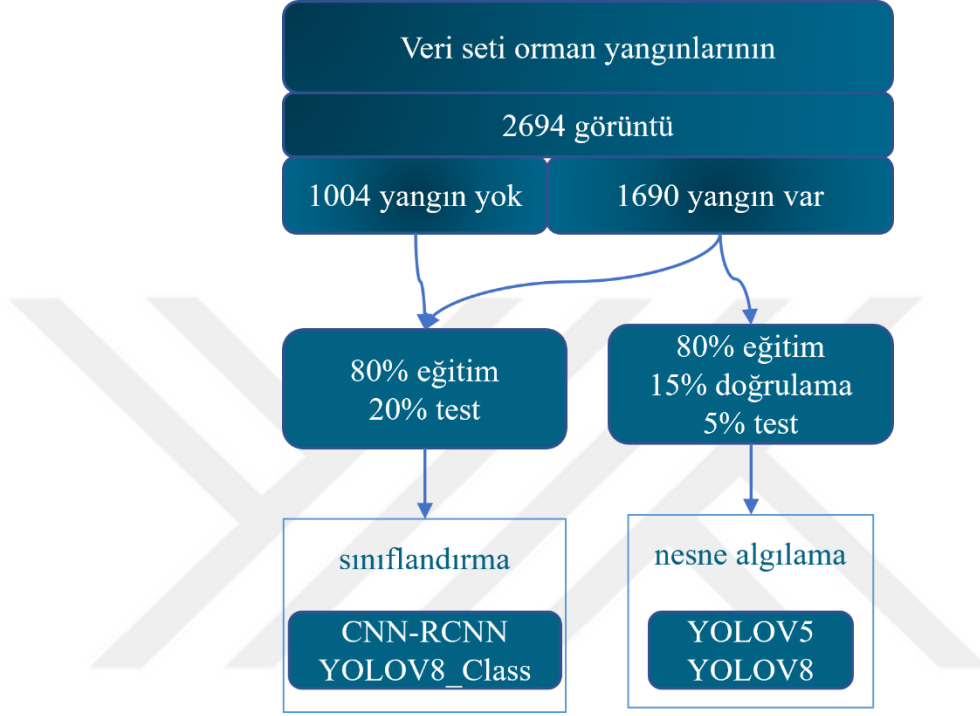


(b)

Şekil 4.16. a) Orman Yangını b) Karabük Üniversitesi'nde Hazırlanan Görüntüleri

Veri seti, orman yangını nesne tespiti amacıyla şu şekilde ayrılmıştır: %80'i eğitim, %5'i test ve %15'i doğrulama için kullanılmıştır. Bu dağılım, görüntülerin doğrulanabilirliğini sağlamak ve tekrarların önüne geçmek için tasarlanmıştır. Orman yangını sınıflandırması için ise veri seti, %80 eğitim ve %20 test olacak şekilde

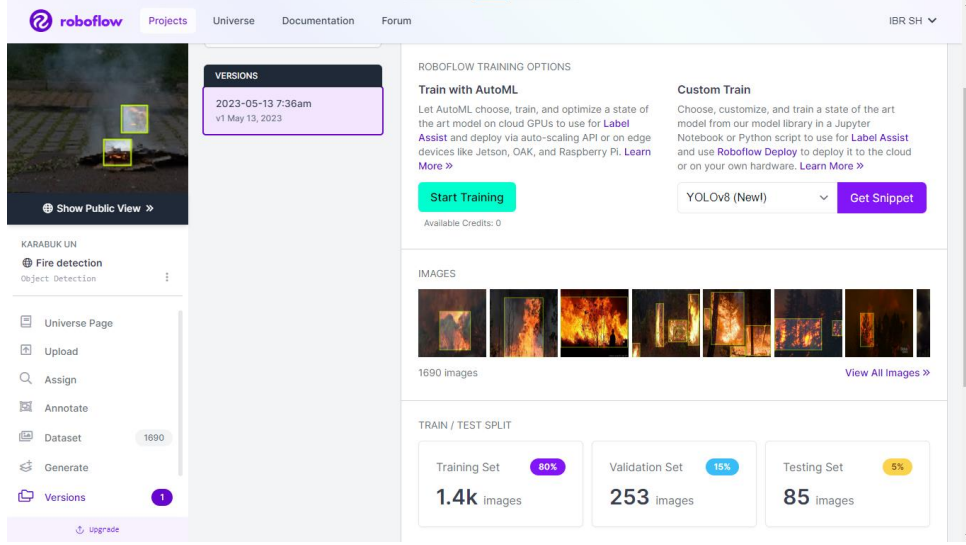
bölünmüştür. Şekil 4.17'de orman yangınlarıyla ilgili veri setinin tanımı ve bu veri setinin, yangın tespiti algoritmaları için eğitim, doğrulama ve test kategorilerine nasıl ayrıldığı detaylı olarak gösterilmektedir. Ayrıca, sınıflandırma algoritmaları için de benzer bir bölümlendirme yapılmıştır.



Şekil 4.17. Veri seti açıklaması.

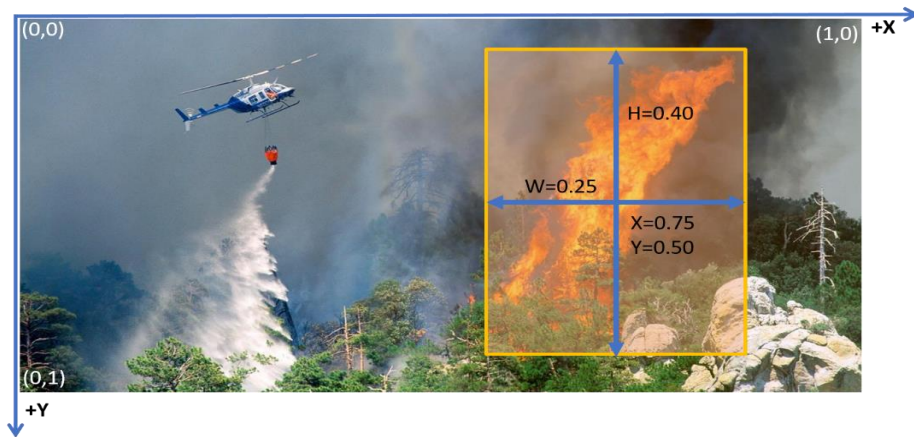
#### 4.3.1. Veri Setinin Etiketlenmesi

YOLOv5 ve YOLOv8 ağlarının veri setinde bulunan görüntüleri doğru bir şekilde tanıyabilmesi için, bu görüntülerin sınıflandırılması zorunludur. Bu amaçla yapılan işlem, tanımlanacak nesnenin etrafına bir kutu ya da dikdörtgen yerleştirmeyi içerir ve bu işleme "etiketleme" adı verilir. Etiketleme işlemi için Roboflow platformu kullanılmıştır. Bu platform sayesinde, belirli algoritmalara uygun bir formatta işlenir. Platformun bir ekran görüntüsü, Şekil 4.18'de sunulmaktadır. Orman yangınlarının belirlenen bölgeleri görüntülerde işaretlendikten sonra, bu görüntüler YOLOv8 ve YOLOv5 formatlarına uygun şekilde dışa aktarılır.



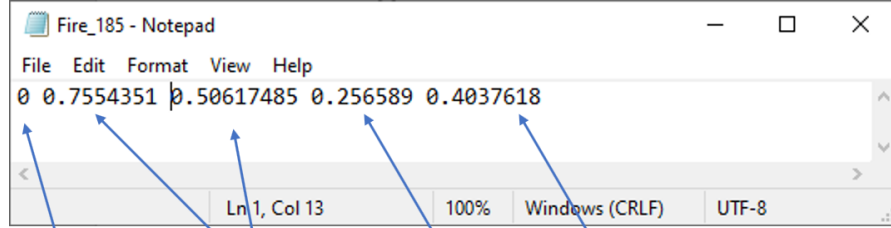
Şekil 4.18. Roboflow veri setinin etiketleme platformu.

Algoritmanın orman yangınlarını tespit etmek için eğitileceği bölgenin koordinatlarını belirlemek adına, görüntüler üzerinde Şekil 4.19'da gösterildiği gibi işaretleme işlemleri yapılır. Her bir görüntü için ilişkilendirilmiş bir etiket dosyası oluşturulur ve bu dosya beş parametre içerir. İlk parametre, sınıfı ifade eder (örneğin, 0, 1, 2, vb.) ve bu, algoritmanın eğitileceği nesne türleri ya da şekillere göre değişiklik gösterir. Bu durumda, sadece "Yangın" olarak adlandırılan bir sınıfımız mevcuttur. Diğer dört parametre ise x, y, genişlik (w) ve yükseklik (h) değerlerini temsil eder.



Şekil 4.19. Veri setinin etiketlemesi.

Belirtildiği gibi, yukarıda anlatılan etiket dosyası sadece bir yangını içerir ve bu yangın 'sınıf 0' olarak etiketlenmiştir. Şekil 4.20’de her bir görüntü için etiketlerin nasıl belirlendiğini ve bu etiketlerin yer aldığı dosyanın nasıl oluşturulduğunu açıklanmaktadır.



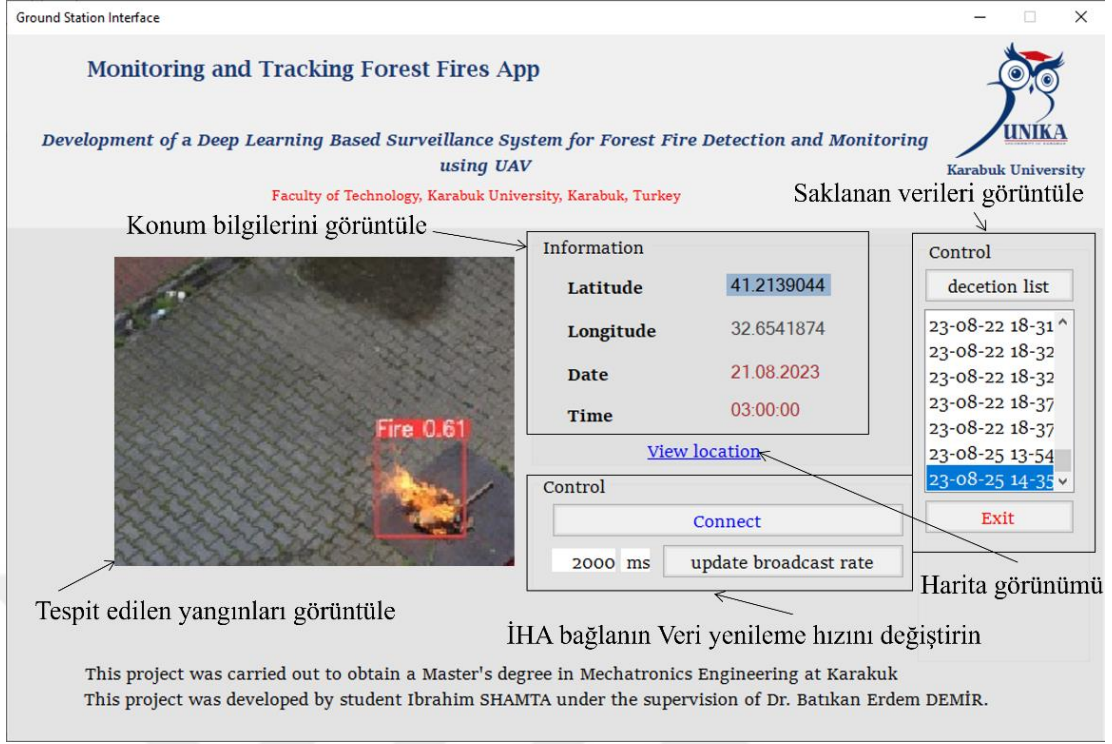
sınıf etiket merkezi (x,y) Yükseklik Genişlik

Şekil 4.20. Etiket dosyası.

#### 4.4. GELİŞTİREN YER İSTASYONU ARAYÜZÜ İLE ORMAN YANGINININ İZLENMESİ

Bu tezde, orman bölgeleri üzerinden hareket eden İHA’yı takip etmek amacıyla özel bir yer istasyonu arayüz programı geliştirilmiştir. Şekil 4.21’de bu yer istasyonu arayüzü verilmektedir.

Bu yazılım, orman yangınlarının tespit edilmesi durumunda, İHA’nın en son elde ettiği görüntüleri sağlamakta ve yangının yerini belirleyen koordinatları sunmaktadır. Ayrıca, yangının keşfedildiği tarih ve saati gibi ek bilgiler de verilmektedir. Program, C# programlama dili ile oluşturulmuştur. Yer istasyonu yazılımı, projeye özgü olarak tasarlanmış bir web sitesiyle entegre edilmiştir. İHA’da bulunan algılama sistemi, elde ettiği verileri bu web sitesine yüklemekte, yer istasyonu yazılımı ise bu web sitesinde bulunan en güncel verileri çekmektedir.



Şekil 4.21. Yer istasyonu arayüzü.

Şekil 4.22’de yangınların tespit edildiği alanların görsel bir temsilini sunulmaktadır ve bu görsellerin sunum metodu için bir örnek olarak işaretlenmiştir.



Şekil 4.22. Yangının çıktığı yerin tespiti.

## BÖLÜM 5

### DENEYSEL ÇALIŞMALAR

Bu çalışmada, dört rotorlu insansız hava aracıyla otonom olarak test uçuşları yapılmış ve bu testler Karabük Üniversitesi kampüsünde gerçekleştirilmiştir. Ek olarak, gerekli önlemler alınarak ve üniversite yetkililerinin izniyle, kontrollü bir yangın oluşturulmuştur. Testler, orman yangınlarını tespit etmek için eğitilen derin öğrenme algoritmaları olan YOLOv5 ve YOLOv8 tabanlı nesne algılama ile görüntüleri sınıflandırmak için tasarlanmış YOLOv8 sınıflandırma ve CNN-RCNN algoritmaları kullanılarak yapılmıştır. NVIDIA Jetson NANO geliştirme kartı kullanılarak bu deneyler gerçekleştirilmiştir. Aynı zamanda, tespit edilen yangınların görüntülerini ve koordinatlarını gösteren bir yer istasyonu uygulaması tasarlanmıştır. Şekil 5.1’de test edilen yangın gösterilmektedir.



Şekil 5.1. Test amaçlı oluşturulan kontrollü yangın.

## **5.1. DÖRT ROTORLU İNSANSIZ HAVA ARACI İLE UÇUŞ TESTLERİ**

Navio2 uçuş kontrol kartı, dört rotorlu insansız hava aracına başarıyla entegre edildi. Bu kart, hava aracının navigasyon kontrolünü yürütmekte ve uçuşun düzenlenmesine katkı sağlamaktadır. Jiroskop, ivmeölçer ve barometre gibi sensörlerin kalibrasyonu, doğruluk ve hassasiyeti gibi konfigürasyon işlemleri, Mission Planner programı aracılığıyla tamamlandı. Bu program, uçuş parametrelerinin ve sensör ayarlarının konfigürasyonunu ve yönetimini kolaylaştırmaktadır.

Konfigürasyon ayarları tamamlandıktan sonra, insansız hava aracı ile toplamda 10 kullanıcı kontrollü ve otomatik pilot modunda 10 uçuş testi yapıldı. Bu testler, uçuş sistemleri ve yazılımlarının doğru ve güvenli bir şekilde işlediğini doğrulamak için kritik öneme sahiptir.

Ek olarak, NVIDIA Jetson Nano geliştirme kartı insansız hava aracına bağlanarak entegre edildi. Bu sayede, uçuş sırasında derin öğrenme tabanlı görüntü işleme süreçleri gerçek olarak yapıldı.

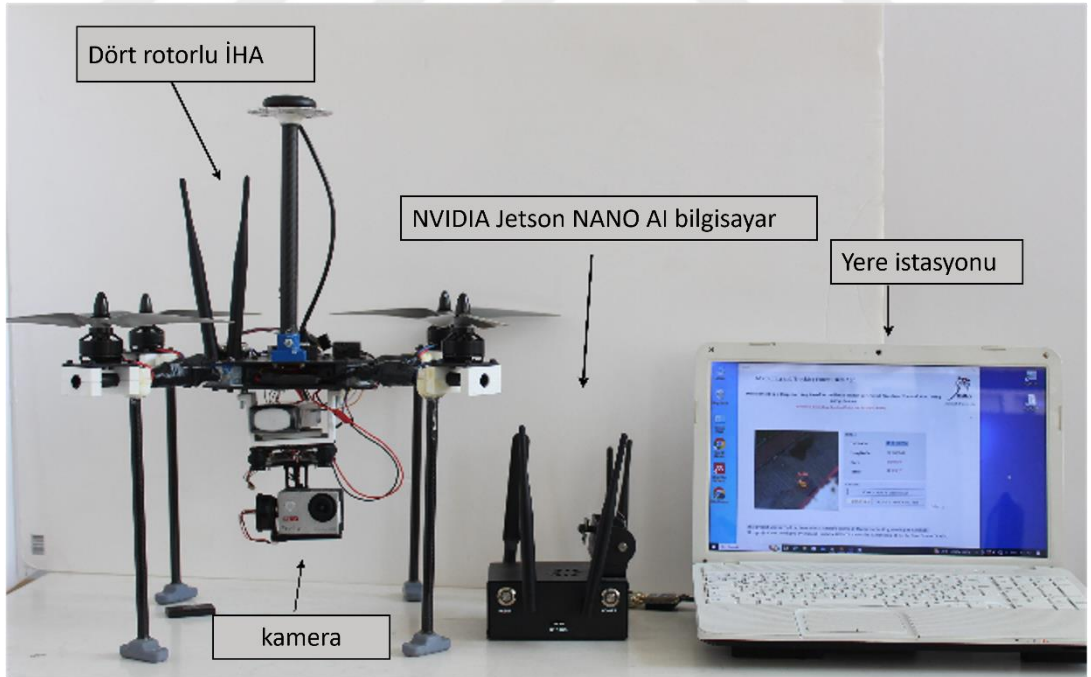
### **5.1.2. Dört Rotorlu İnsansız Hava Aracının Otonom Uçuş Testleri**

Dört rotorlu insansız hava aracı için otonom uçuş testi, önceden tasarlanmış bir görev rotası üzerinde yürütülmüştür. Bu belirlenen rota, Mission Planner yazılımı aracılığıyla oluşturulmuş ve Navio2 uçuş kontrol kartına entegre edilmiştir. Otonom uçuş testi sırasında, Mission Planner yazılımından alınan uçuş verileri ile aracın, tasarlanmış görev rotasına uygun bir seyir izleyip izlemediği dikkatle gözlemlenmiş ve kontrol edilmiştir. Testler, Karabük Üniversitesi Teknoloji Fakültesi'nin arka bahçesinde icra edilmiş olup, uçuş 20 metre yükseklikte ve 45 saniye süreyle tamamlanmıştır.

## 5.2. DERİN ÖĞRENME ALGORİTMALARI VE İNSANSIZ HAVA ARACI KULLANILARAK ORMAN YANGINLARINI ALGILAMA DENEYLERİ

Yapay sinir ağlarının eğitim süreci tamamlandıktan sonra, her bir ağı bağımsız bir şekilde test etmekteyiz. Nvidia Jetson Nano geliştirme kartı, insansız hava aracı içerisinde gerçek zamanlı olarak işlev görmekte ve ormanlık alanlarda meydana gelen yangınları algılayarak ilgili verileri masaüstü uygulamasına iletmektedir. Bu deney serilerinde, belirli bir coğrafi bölgede insansız hava aracının hem kullanıcı kontrollü hem de otonom uçuşları başarıyla gerçekleştirilmiştir.

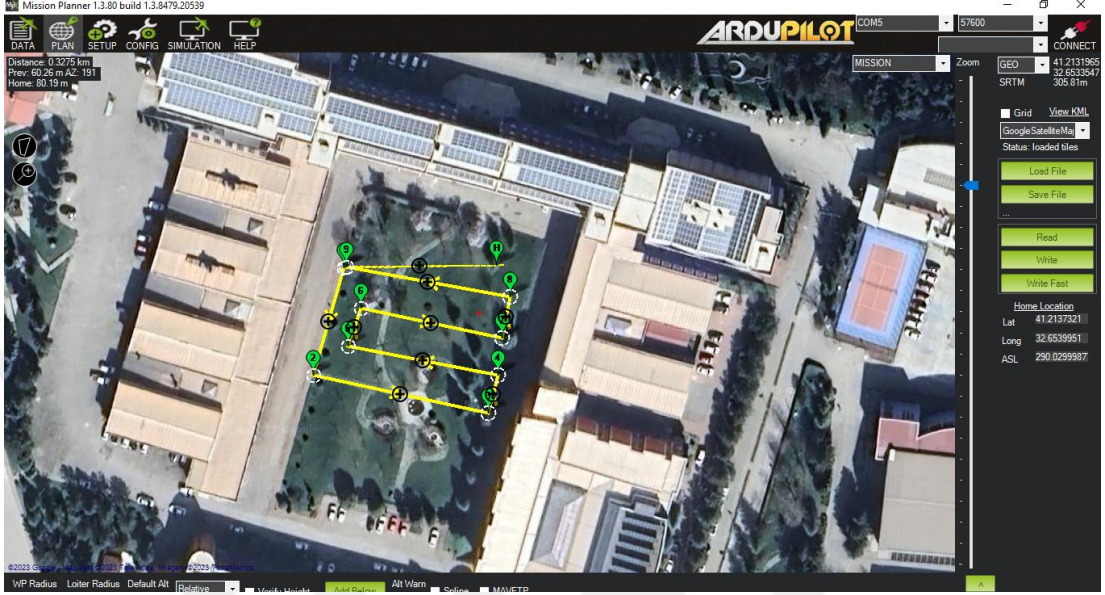
Deneysel için kullanılan ekipmanlar, Şekil 5.2'de detaylı bir şekilde açıklanmıştır. Ormanlık alanları izlemek amacıyla, bir kamera sistemine sahip dört rotorlu bir insansız hava aracı kullanılmıştır. Bu araç, orman bölgeleri üzerinde otonom bir şekilde hareket eder ve algıladığı yangın verilerini yer istasyonu uygulamasına anlık olarak gönderir.



Şekil 5.2. Deneysel ekipmanlar.

Otonom bir uçuş yapabilmek için önceden belirlenmiş bir uçuş rotası oluşturulması gerekmektedir. Şekil 5.3'te, orman yangınlarını tespit etmek amacıyla hazırlanmış

olan bu otonom uçuş rotası gösterilmektedir. Bu hazırlanan rotası, Navio2 uçuş kontrol kartına yüklenmiş ve bu şekilde otonom uçuşlar toplamda 18 defa olmak üzere başarıyla tamamlanmıştır.



Şekil 5.3. Orman yangını tespit testi için oluşturulan otonom uçuş rotası.

Şekil 5.4'te otonom uçuş rotasına dair parametreler sunulmaktadır. Bu rotanın uçuş yüksekliği 20 metre olarak ayarlanmıştır. Yapılan otonom uçuşların ortalama süresi ise 5 dakika 15 saniye olarak kaydedilmiştir. Uçuş sonrası batarya gerilimi de 11.9V olarak ölçülmüştür.

	Command					Lat	Long	Alt	Frame	Delete			Grad %	Angle	Dist	AZ
1	TAKEOFF	0	0	0	0	41.2137251	32.6534888	30	Relative	X			70.8	35.3	51.9	269
2	WAYPOINT	0	0	0	0	41.2134568	32.6533869	30	Relative	X			0.0	0.0	31.0	196
3	WAYPOINT	0	0	0	0	41.2133619	32.6539689	30	Relative	X			0.0	0.0	49.8	102
4	WAYPOINT	0	0	0	0	41.2134568	32.6540011	30	Relative	X			0.0	0.0	10.9	14
5	WAYPOINT	0	0	0	0	41.2135294	32.6535022	30	Relative	X			0.0	0.0	42.5	281
6	WAYPOINT	0	0	0	0	41.2136242	32.6535451	30	Relative	X			0.0	0.0	11.1	19
7	WAYPOINT	0	0	0	0	41.2135496	32.6540145	30	Relative	X			0.0	0.0	40.1	102
8	WAYPOINT	0	0	0	0	41.2136525	32.6540413	30	Relative	X			0.0	0.0	11.7	11
9	LAND	0	0	0	0	41.2137271	32.6534969	0	Relative	X			0.0	0.0	46.3	280

Şekil 5.4. Otonom uçuş rotası parametreleri.

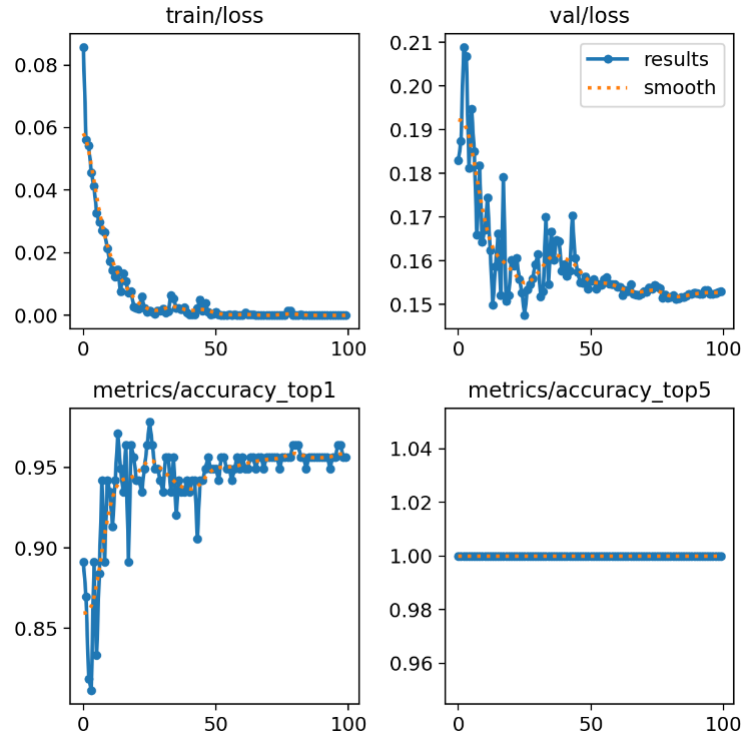
### 5.2.1. Orman Yangınlarını Tespit Etmek İçin Kullanılan Algoritmaların Performansının Karşılaştırılması Ve Değerlendirilmesi

Oluşturulan veri seti, CNN-RCNN ve YOLOv8 sınıflandırma algoritmalarının eğitimi için kullanılmıştır. Bu veri seti, orman yangınlarını gösteren görüntüler ile sağlam orman koşullarını yansıtan görüntüleri içermektedir. YOLOv8 ve YOLOv5 gibi nesne tespiti algoritmaları, yalnızca orman yangınlarını gösteren görüntüler üzerinde özel olarak eğitilmiştir.

Farklı kaynaklardan toplanan verilerin doğruluğu detaylı bir şekilde incelenmiş ve tekrar eden veya benzer özelliklere sahip görüntülerin olmadığı doğrulanmıştır. Bu ön incelemenin ardından, belirtilen tüm algoritmalar 100'er defa eğitilmişlerdir.

#### 5.2.1.1. Sınıflandırma Algoritması Eğrileri

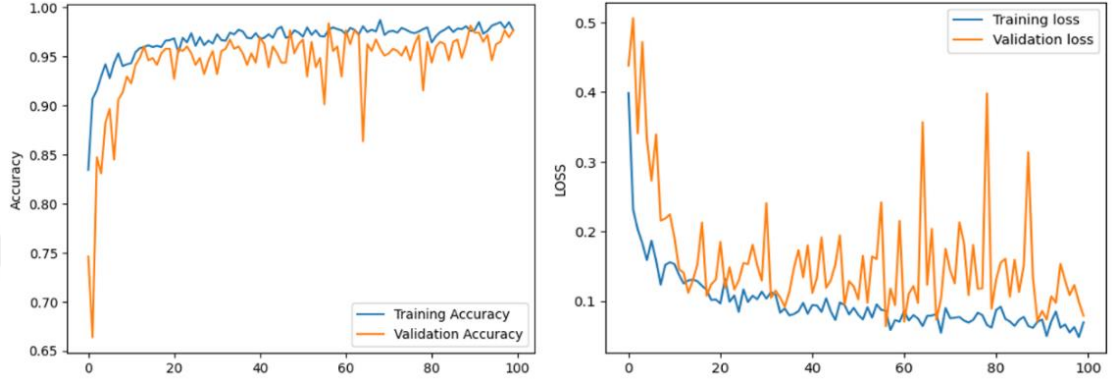
Şekil 5.5'de YOLOv8 sınıflandırma modelinin eğitim sürecine ilişkin ayrıntıları sunmaktadır.



Şekil 5.5 YOLOv8 sınıflandırma modelinin eğitim sonuçları.

İlk grafik, eğitim kaybını (train loss) temsil etmekte ve 100 iterasyon sonucunda bu değer sıfıra ulaşmıştır. Doğrulama kaybı (val loss) ise %0.15 seviyesine inmiştir. Nihai olarak, modelin en yüksek doğruluk oranı %100 olarak kayıt edilmiştir.

Şekil 5.6'da, CNN-RCNN modelinin eğitim sürecine dair detayları gösterilmektedir.

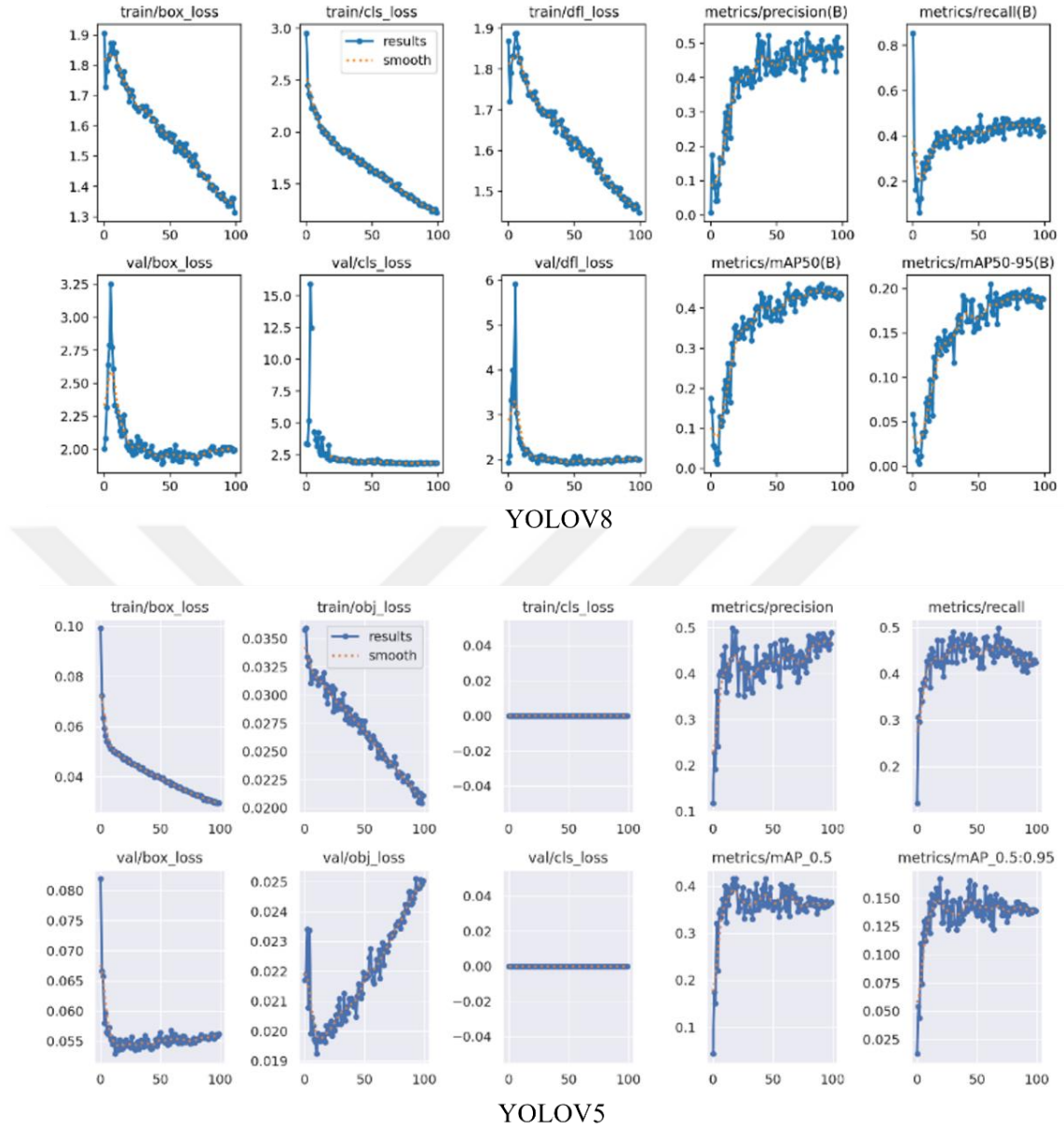


Şekil 5.6 CNN-RCNN sınıflandırma modelinin eğitim sonuçları.

İlk grafik, hem eğitim hem de doğrulama (validation) süreçlerine ait doğruluk oranlarını sergilemektedir. İkinci grafik ise modelin kayıp (loss) değerlerini göstermekte, ve burada belirli dalgalanmalar göze çarpmaktadır. Bu dalgalanmaların nedeni, veri setinin çeşitliliği ve veritabanından seçilen bölgelerin farklı karakteristiklere sahip olmasından kaynaklanmaktadır.

### 5.2.1.2. Eğriler Nesne Algılama Algoritmaları

Şekil 5.7'de, YOLOv8n ve YOLOv5n sınıflandırma modellerinin eğitim süreçlerinden elde edilen sonuçları sıralı bir biçimde sunulmaktadır. Bu grafikler, modellerin performansını anlamamıza yardımcı olan eğrileri içermekte ve modellerin karşılaştırmalı performanslarını görsel bir şekilde değerlendirmeye olanak tanımaktadır.



Şekil 5.7 YOLOv5 ve YOLOv8 nesne algılama modelinin eğitim sonuçları.

YOLO algılama sisteminin performansını değerlendirmek için kullanılan temel metrikler ve bunların açıklamaları şu şekildedir.

- mAP50: Bu metrik, farklı kategoriler arasındaki ortalama doğruluğu temsil eder ve "Intersection over Union" (IoU) değeri %50 ve üzeri olan sınır kutuları için hesaplanır (yani, bir nesne %50 veya daha fazla doğrulukla tespit edildiğinde, bu bir doğru pozitif olarak kabul edilir) [54].

- mAP50-95: Bu değer, IoU eşiği %50 ile %95 arasında değişen sınır kutuları için ortalama doğruluğu ifade eder [54].
- Doğruluk (B): Bu, tespit edilen doğru pozitiflerin (TP) toplam doğru pozitif sayısına oranını temsil eder [54].
- Duyarlılık: Bu metrik, sınır kutusu içinde doğru bir şekilde algılanan pozitiflerin, toplam doğru pozitif hedef sayısına bölünerek elde edilir [54].
- Doğruluk-Duyarlılık (PR) Eğrisi: Bu eğri, algılama ve sınıflandırma modellerinin performansını değerlendirmek için makine öğrenimi sahasında kullanılır. Eğri, farklı güven eşikleri için doğruluk ve duyarlılık değerlerini hesaplayarak oluşturulur. Bu değerler, modelin performansını görselleştirmek için bir eğri olarak sunulur [54].

### 5.2.1.3. Performans Değerlendirmesi

Derin öğrenme görevlerinde kullanılan değerlendirme metrikleri, optimize edilmiş bir sınıflandırıcı elde etmede kritik öneme sahiptir [55]. Bu metrikler genellikle iki ana aşamada, yani eğitim ve test aşamalarında, veri sınıflandırma süreçlerinde kullanılır. Eğitim aşamasında, bu metrikler sınıflandırma algoritmasının optimizasyonu için temel oluşturur. Bu, belirli bir sınıflandırıcı için gelecekteki değerlendirmelerde daha doğru tahminler üretebilecek bir ayırım yapma mekanizması olarak değerlendirme metriğinin kullanılmasını ifade eder. Öte yandan, test aşamasında değerlendirme metrikleri, bilinmeyen veri setleri üzerinde sınıflandırıcının etkinliğini ölçmek için kullanılır [55].

Eşitlik 5.1'de gösterildiği üzere, "Doğru Negatif" (True Negative - TN) ve "Doğru Pozitif" (True Positive - TP), başarılı bir şekilde sınıflandırılmış negatif ve pozitif örnekler olarak kabul edilir. Ayrıca, "Yanlış Negatif" (False Negative – FN) ve "Yanlış Pozitif" (False Positive – FP) ise yanlış bir şekilde sınıflandırılmış pozitif ve negatif örnekler olarak tanımlanır. Sonraki bölümde, en yaygın kullanılan değerlendirme metriklerinden birkaçı listelenmiştir [29], [55].

1- Doğruluk (Accuracy): Doğru tahmin edilen sınıfların toplam değerlendirilen örnek sayısına oranını hesaplar. Denklem 5.1'e göre hesaplanır [29], [55].

$$\text{Doğruluk (Accuracy)} = \frac{TP+TN}{TP+FP+TN+FN} \quad (5.1)$$

2- Duyarlılık veya Gerçek Pozitif Oranı (Recall): Pozitif örneklerin doğru bir şekilde sınıflandırılan oranını hesaplamak için kullanılır. Denklem 5.2'ye göre hesaplanır [29], [55].

$$\text{Duyarlılık (Sensitivity)} = \frac{TP}{TP+FN} \quad (5.2)$$

3- Özgüllük (Specificity): Negatif örneklerin doğru bir şekilde sınıflandırılan oranını hesaplamak için kullanılır. Denklem 5.3'e göre hesaplanır [29], [55].

$$\text{Özgüllük (Specificity)} = \frac{TN}{FP+TN} \quad (5.3)$$

4- Kesinlik (Precision): Pozitif sınıftaki tüm tahmin edilen örnekler arasında doğru bir şekilde tahmin edilen pozitif örnekleri hesaplamak için kullanılır. Denklem 5.4'e göre hesaplanır [29], [55].

$$\text{Kesinlik (Precision)} = \frac{TP}{TP+FP} \quad (5.4)$$

5- F1\_Skoru: Duyarlılık ve kesinlik oranları arasındaki harmonik ortalama hesaplanır. Denklem 5.5'e göre hesaplanır [29], [55].

$$F1_{\text{Skoru}} = 2 \times \frac{\text{Kesinlik} \times \text{Duyarlılık}}{\text{Kesinlik} + \text{Duyarlılık}} \quad (5.5)$$

6- J\_Skoru: Bu metrik aynı zamanda Youden'in J istatistiği olarak da adlandırılır. Denklem 5.6, metriği temsil eder [29], [55].

$$J_{\text{skoru}} = \text{Duyarlılık} + \text{Özgüllük} - 1 \quad (5.6)$$

7- Yanlış Pozitif Oranı (FPR): Bu metrik, yanlış alarm oranının olasılığını ifade eder ve Denklem 5.7'de hesaplanır [29], [55].

$$FPR = 1 - \text{Özgüllük} \quad (5.7)$$

Çizelge 5.1'de bu tez kapsamında ele alınan algoritmaların karmaşıklık matrisini sergilemektedir. Tüm algoritmalar, eğitim sürecinde kullanılmayan aynı veri kümesi üzerinde test edilmiştir.

Çizelge 5. 1. Deneysel çalışma sonucu oluşan hata matrisi.

	Nesne algılama		Sınıflandırma	
	YOLOv5	YOLOv8	YOLOv8	CNN-RCNN
TP	84	85	98	76
FP	6	7	6	5
TN	94	93	94	95
FN	16	15	2	2

YOLOv8 ve YOLOv5 algoritmaları, aynı şartlar ve aynı veri kümesi gözetilerek eğitilmiş, ve bu sürecin sonuçları karşılaştırılmıştır. Elde edilen bulgular, Çizelge 5.2'de detaylandırılmıştır.

Çizelge 5. 2. Nesne algılama için değerlendirme metrikleri.

	(YOLOv8 ve YOLOv5) Nesne algılama için değerlendirme metrikleri				
	mAP@50	mAP@50-95	Kesinlik	Duyarlılık	KD (PR)
YOLOv8n	0.462	0.205	0.503	0.432	0.462
YOLOv5n	0.365	0.138	0.448	0.424	0.416

Çizelge 5.3'de bu tez içerisinde incelenen ağların performans metriklerini sunmaktadır. Bu performans ölçütleri sayesinde, her bir ağın davranışsal özellikleri ve belirlenen görevleri ne derece başarılı bir şekilde yerine getirebildiği anlaşılabilir.

Çizelge 5. 3. Değerlendirme metrikleri.

	Doğruluk (Acc)	Duyarlılık (sn)	Özgüllük (sp)	Kesinlik (p)	F1_Skoru	J_Skoru	FPR
CNN-RCNN sınıflandırma	0.96	0.97	0.95	0.94	0.96	0.92	0.05
YOLOV8 sınıflandırma	0.96	0.98	0.94	0.94	0.96	0.92	0.06
YOLOV8n nesne algılama	0.89	0.85	0.93	0.92	0.88	0.78	0.07
YOLOV5n nesne algılama	0.89	0.84	0.94	0.93	0.88	0.78	0.06

Çizelge 5.4, her model için işlem hızını göstermektedir. Sınıflandırma/nesne algılama hızı, NVIDIA Jetson NANO geliştirme kartı üzerinde test edilmiştir.

Çizelge 5. 4. NVIDIA Jetson Nano'da işlem hızı.

	CNN-RCNN sınıflandırma	YOLOV8 sınıflandırma	YOLOV8n nesne algılama	YOLOV5n nesne algılama
Milisaniye	0,100 ms	0,20 ms	0,45 ms	0,65 ms

## BÖLÜM 6

### SONUÇLAR VE DEĞERLENDİRME

Bu çalışma, orman yangınlarını izlemek amacıyla derin öğrenme ağları ve NVIDIA Jetson Nano geliştirme kartını kullanarak insansız hava araçları ile entegre bir sistem geliştirmenin yenilikçi bir çerçevesini sunmaktadır. Algılanan bir yangın durumunda, insansız hava aracı, yangının genişlemesini gösteren görüntüler çekip bu bilgileri, konum verisiyle birleştirerek orman yangınlarını özel olarak izlemek için tasarlanmış yer istasyonu uygulamasına yüklemektedir.

Çalışma, orman yangını tespiti için özgün bir CNN-RCNN algoritması üzerinde de durmaktadır. Elde edilen sonuçlar, farklı algoritmalara dayalı performansla karşılaştırılmıştır. Tüm modeller, aynı kriterlere tabi tutulan ve toplamda 2694 görüntü içeren aynı veri kümesi üzerinde eğitilmiştir; bu görüntülerden 1650'si orman yangınına, 1004'ü ise sağlam orman koşullarına aittir.

CNN-RCNN ve YOLOv8 sınıflandırma algoritmaları, doğruluk metrikleri açısından öne çıkmaktadır. CNN-RCNN algoritmasının doğruluk oranı %96, YOLOv8 algoritmasının ise %96 olarak kaydedilmiştir. Ek olarak, CNN-RCNN algoritmasının eğitim ve doğrulama hataları %0.06 iken, YOLOv8 algoritmasının doğrulama hatası %0.15 ve eğitim hatası ise sıfırdır. Duyarlılık açısından karşılaştırıldığında, YOLO sınıflandırma algoritması en yüksek hassasiyet değerini göstermiştir. YOLOv8 modelinin hassasiyeti %98 olarak ölçülmüştür.

Son olarak, hiper-parametre seçiminin, evreşimli sinir ağı performansını büyük ölçüde etkileyebileceği belirtilmelidir. Bu nedenle, dikkatli parametre seçimi, optimizasyon sürecinde kritik öneme sahiptir. Grafikselleştirme biriminin (GPU) yüksek işlem kapasitesi, etkili bir evreşimli sinir ağı eğitimi için gereklidir. Bulut bilişim platformlarının, gelecekte derin öğrenme uygulamalarının gelişiminde önemli

bir rol oynaması beklenmektedir. Bulut bilişim, sadece büyük veri setlerini işleme kapasitesi sağlamakla kalmayıp, işlem verimliliğini artırır ve maliyetleri azaltır, ayrıca derin öğrenme mimarilerinin eğitilmesinde esneklik kazandırır.

Bu çalışma, gelecekte uygulanabilecek şu önerileri sunmaktadır: CNN-RCNN algoritmasının hızını, TensorFlow kütüphanesindeki yeni iyileştirmeleri kullanarak artırmak, ayrıca algoritmanın doğruluğunu parametreleri veya katmanları değiştirerek artırmak. NVIDIA'nın Jetson Nano'dan daha yüksek kapasiteli daha yeni bir yapay zeka bilgisayarı kullanmak da önerilmektedir.

Ayrıca, orman yangınlarının yayılma davranışını öngörebilecek bir özelliğin eklenmesi önerilmektedir. Bu, yangınların artışta mı yoksa azalışta mı olduğunu, yayılma yönünü belirlemeyi ve bu verileri yer istasyonu uygulamasına göndermeyi içerir.

Son olarak, birden fazla şehirde filo halinde bir insansız hava araçları topluluğu oluşturmak ve yer istasyonu programını geliştirmek, aynı zamanda bu tüm insansız hava araçlarını izleyebilecek bir web sitesinin tasarlanması önerilmektedir. Bu, daha geniş alanları izleme olanağı sağlayacaktır.

## KAYNAKLAR

- [1] J. T. Abatzoglou, A. P. Williams, and R. Barbero, “Global Emergence of Anthropogenic Climate Change in Fire Weather Indices,” *Geophys Res Lett*, vol. 46, no. 1, pp. 326–336, Jan. 2019, doi: 10.1029/2018GL080959.
- [2] S. Liang and M. D. Hurteau, “Novel climate–fire–vegetation interactions and their influence on forest ecosystems in the western USA,” *Functional Ecology*. British Ecological Society, 2023. doi: 10.1111/1365-2435.14263.
- [3] F. J. Alcasena, A. A. Ager, J. D. Bailey, N. Pineda, and C. Vega-García, “Towards a comprehensive wildfire management strategy for Mediterranean areas: Framework development and implementation in Catalonia, Spain,” *J Environ Manage*, vol. 231, pp. 303–320, Feb. 2019, doi: 10.1016/j.jenvman.2018.10.027.
- [4] A. Karali, K. V. Varotsos, C. Giannakopoulos, P. P. Nastos, and M. Hatzaki, “Seasonal fire danger forecasts for supporting fire prevention management in an eastern Mediterranean environment: The case of Attica, Greece,” *Natural Hazards and Earth System Sciences*, vol. 23, no. 2, pp. 429–445, Feb. 2023, doi: 10.5194/nhess-23-429-2023.
- [5] L. Zhang, M. Wang, Y. Ding, and X. Bu, “MS-FRCNN: A Multi-Scale Faster RCNN Model for Small Target Forest Fire Detection,” *Forests*, vol. 14, no. 3, Mar. 2023, doi: 10.3390/f14030616.
- [6] Y. Zhang, S. Chen, W. Wang, W. Zhang, and L. Zhang, “Pyramid Attention Based Early Forest Fire Detection Using UAV Imagery,” in *Journal of Physics: Conference Series*, Institute of Physics, 2022. doi: 10.1088/1742-6596/2363/1/012021.
- [7] B. E. Demir, R. Bayir, and F. Duran, “Real-time trajectory tracking of an unmanned aerial vehicle using a self-tuning fuzzy proportional integral derivative controller,” *International Journal of Micro Air Vehicles*, vol. 8, no. 4, pp. 252–268, Dec. 2016, doi: 10.1177/1756829316675882.
- [8] *Technology Development for Army Unmanned Ground Vehicles*. National Academies Press, 2002. doi: 10.17226/10592.

- [9] 2009 4th IEEE Conference on Industrial Electronics and Applications : 25-27 May 2009, Xi'an, China. IEEE, 2009.
- [10] Z. Li, S. Nadon, J. Cihlar, and B. Stocks, "Satellite-based mapping of Canadian boreal forest res: evaluation and comparison of algorithms," 2000. [Online]. Available: <http://www.tandf.co.uk/journals>
- [11] R. Xu, H. Lin, K. Lu, L. Cao, and Y. Liu, "A forest fire detection system based on ensemble learning," *Forests*, vol. 12, no. 2, pp. 1–17, Feb. 2021, doi: 10.3390/f12020217.
- [12] Wuhan da xue. and IEEE Communications Society., 2005 *International Conference on Wireless Communications, Networking and Mobile Computing : WCNM 2005 : proceedings : September 23-26, 2005, Wuhan, China*. IEEE, 2005.
- [13] A. M. Fernandes, A. B. Utkin, A. V. Lavrov, and R. M. Vilar, "Development of neural network committee machines for automatic forest fire detection using lidar," *Pattern Recognit*, vol. 37, no. 10, pp. 2039–2047, Oct. 2004, doi: 10.1016/j.patcog.2004.04.002.
- [14] Z. Wang, L. Wu, T. Li, and P. Shi, "A Smoke Detection Model Based on Improved YOLOv5," *Mathematics*, vol. 10, no. 7, Apr. 2022, doi: 10.3390/math10071190.
- [15] Ravishankara K, Ankith K, and Monish PS, "Fire Detection for Surveillance of Forest Using Image Processing," *International Research Journal of Engineering and Technology*, 2020, [Online]. Available: [www.irjet.net](http://www.irjet.net)
- [16] F. Shi, H. Qian, W. Chen, M. Huang, and Z. Wan, "A Fire Monitoring and Alarm System Based on YOLOv3 with OHEM." [Online]. Available: <https://cvpr.kmu.ac.kr/>
- [17] S. Wu and L. Zhang, "Using Popular Object Detection Methods for Real Time Forest Fire Detection," in *Proceedings - 2018 11th International Symposium on Computational Intelligence and Design, ISCID 2018*, Institute of Electrical and Electronics Engineers Inc., Jul. 2018, pp. 280–284. doi: 10.1109/ISCID.2018.00070.
- [18] A. Kaur, R. Sethi, and K. Kaur, "Comparison of Forest Fire Detection Techniques Using WSNs." [Online]. Available: [http://www.geog.ubc.ca/courses/geog570/talks\\_2001/fuzzy.htm](http://www.geog.ubc.ca/courses/geog570/talks_2001/fuzzy.htm)

- [19] T. Çelik, H. Özkaramanlı, and H. Demirel, “Fire and Smoke Detection Without Sensors: Image Processing Based Approach.”
- [20] A. Calle, J. Sanz, C. Moclan, A. Romo, and J. L. Casanova, “Detection and Monitoring of Forest Fires in China through the Envisat-Aatsr Sensor.”
- [21] Nvidia, “DATA SHEET NVIDIA Jetson Nano System-on-Module Maxwell GPU + ARM Cortex-A57 + 4GB LPDDR4 + 16GB eMMC,” 2014. [Online]. Available: [www.khronos.org/conformance](http://www.khronos.org/conformance).
- [22] “4G/3G/2G/GNSS Expansion Board for Jetson Nano, Based on SIM7600G-H, Global Applicable.” <https://www.waveshare.com/sim7600g-h-4g-for-jetson-nano.htm> (accessed Jul. 12, 2023).
- [23] “SIM7600E-H 4G HAT User Manual SIM7600E-H 4G HAT.”
- [24] “SIM7600X-H\_SPEC\_20180124”.
- [25] J. McCarthy, “What Is Artificial Intelligence?,” 2007. [Online]. Available: <http://www-formal.stanford.edu/jmc/>
- [26] S. J. Stuart J. Russell *et al.*, *Artificial intelligence : a modern approach*.
- [27] C. Janiesch, P. Zschech, and K. Heinrich, “Machine learning and deep learning,” *Electronic Markets*, vol. 31, no. 3, pp. 685–695, Sep. 2021, doi: 10.1007/S12525-021-00475-2/TABLES/2.
- [28] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 2015.
- [29] L. Alzubaidi *et al.*, “Review of deep learning: concepts, CNN architectures, challenges, applications, future directions,” *Journal of Big Data 2021 8:1*, vol. 8, no. 1, pp. 1–74, Mar. 2021, doi: 10.1186/S40537-021-00444-8.
- [30] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun ACM*, vol. 60, no. 6, pp. 84–90, Jun. 2017, doi: 10.1145/3065386.
- [31] E. A. Smirnov, D. M. Timoshenko, and S. N. Andrianov, “Comparison of Regularization Methods for ImageNet Classification with Deep Convolutional Neural Networks,” *AASRI Procedia*, vol. 6, pp. 89–94, 2014, doi: 10.1016/j.aasri.2014.05.013.

- [32] G. Altan, "DeepGraphNet: Grafiklerin Sınıflandırılmasında Derin Öğrenme Modelleri," *European Journal of Science and Technology*, pp. 319–327, Oct. 2019, doi: 10.31590/ejosat.638256.
- [33] B. Kayci, B. E. Demir, And F. Demir, "İHA Tarafından Elde Edilen Termal Görüntüler Kullanılarak Fotovoltaik Sistemde Derin Öğrenme Tabanlı Arıza Tespiti ve Teşhisi," *Journal of Polytechnic*, Jun. 2022, doi: 10.2339/politeknik.1094586.
- [34] C.-Y. Wang, H.-Y. M. Liao, Y.-H. Wu, P.-Y. Chen, J.-W. Hsieh, and I.-H. Yeh, "CSPNet: A New Backbone that can Enhance Learning Capability of CNN."
- [35] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," Jun. 2015, [Online]. Available: <http://arxiv.org/abs/1506.01497>
- [36] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu, "CNN-RNN: A Unified Framework for Multi-label Image Classification."
- [37] "Deep learning architectures - IBM Developer." <https://developer.ibm.com/articles/cc-machine-learning-deep-learning-architectures/> (accessed Jul. 16, 2023).
- [38] C. Gao, J. Yan, S. Zhou, P. K. Varshney, and H. Liu, "Long short-term memory-based deep recurrent neural networks for target tracking," *Inf Sci (N Y)*, vol. 502, pp. 279–296, Oct. 2019, doi: 10.1016/j.ins.2019.06.039.
- [39] R. Kumar, "Cerebral LSTM: A Better Alternative for Single- and Multi-Stacked LSTM Cell-Based RNNs," *SN Comput Sci*, vol. 1, no. 2, Mar. 2020, doi: 10.1007/s42979-020-0101-1.
- [40] Institute of Electrical and Electronics Engineers. Turkey Section. and Institute of Electrical and Electronics Engineers, *3rd International Symposium on Multidisciplinary Studies and Innovative Technologies : proceedings : 11-13 October 2019, Ankara, Turkey*.
- [41] B. Cheng, Y. Wei, H. Shi, R. Feris, J. Xiong, and T. Huang, "Revisiting RCNN: On Awakening the Classification Power of Faster RCNN."
- [42] Ö. İnik *et al.*, "Gaziosmanpaşa Bilimsel Araştırma Dergisi (GBAD), Derin Öğrenme ve Görüntü Analizinde Kullanılan Derin Öğrenme Modelleri", [Online]. Available: <http://dergipark.gov.tr/gbad>

- [43] Institute of Electrical and Electronics Engineers. Turkey Section. and Institute of Electrical and Electronics Engineers, *3rd International Symposium on Multidisciplinary Studies and Innovative Technologies : proceedings : 11-13 October 2019, Ankara, Turkey.*
- [44] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," Dec. 2016, [Online]. Available: <http://arxiv.org/abs/1612.08242>
- [45] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression," 2016. [Online]. Available: <https://github.com/Zzh-tju/DIoU>.
- [46] M. Shenoda, "Lighting and Rotation Invariant Real-time Vehicle Wheel Detector based on YOLOv5," May 2023, [Online]. Available: <http://arxiv.org/abs/2305.17785>
- [47] J. Terven and D. Cordova-Esparza, "A Comprehensive Review of YOLO: From YOLOv1 and Beyond," Apr. 2023, [Online]. Available: <http://arxiv.org/abs/2304.00501>
- [48] D. Hendrycks and K. Gimpel, "Gaussian Error Linear Units (GELUs)," Jun. 2016, [Online]. Available: <http://arxiv.org/abs/1606.08415>
- [49] G. Ghiasi *et al.*, "Simple Copy-Paste is a Strong Data Augmentation Method for Instance Segmentation." [Online]. Available: <https://cocodataset.org/>
- [50] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond Empirical Risk Minimization," Oct. 2017, [Online]. Available: <http://arxiv.org/abs/1710.09412>
- [51] X. Li *et al.*, "Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection."
- [52] A. Khan and B. Hassan, "Dataset for Forest Fire Detection," vol. 1, 2020, doi: 10.17632/GJMR63RZ2R.1.
- [53] "FIRE Dataset | Kaggle." <https://www.kaggle.com/datasets/phylake1337/fire-dataset> (accessed Jun. 04, 2023).
- [54] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal Speed and Accuracy of Object Detection," Apr. 2020, [Online]. Available: <http://arxiv.org/abs/2004.10934>
- [55] H. M and S. M.N, "A Review on Evaluation Metrics for Data Classification Evaluations," *International Journal of Data Mining & Knowledge*

*Management Process*, vol. 5, no. 2, pp. 01–11, Mar. 2015, doi:  
10.5121/ijdkp.2015.5201.



## ÖZGEÇMİŞ

Ibrahim SHAMTA 2016 yılında Sham International University Mühendislik Fakültesi Mekatronik Mühendisliği bölümünde öğrenimine başladı. 2020 yılında lisans eğitimini tamamladıktan sonra Karabük Üniversitesi Lisansüstü Eğitim Enstitüsü Mekatronik Mühendisliği Anabilim Dalında Yüksek Lisans eğitimine devam etmektedir.

