



T.C.

TOKAT GAZİOSMANPAŞA ÜNİVERSİTESİ

LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

MEKATRONİK MÜHENDİSLİĞİ ANABİLİM DALI

**DERİN ÖĞRENME TEKNİKLERİ KULLANILARAK GERÇEK
ZAMANLI SALDIRI TESPİTİ**

DOKTORA TEZİ

Ahmet ER

Danışman: Doç. Dr. A. Hakan YAVUZ

TOKAT- 2024

ETİK SÖZLEŞME

Tokat Gaziosmanpaşa Üniversitesi Lisansüstü Eğitim Enstitüsü tez yazım kılavuzuna göre, Doç. Dr. A. Hakan YAVUZ danışmanlığında hazırlamış olduğum “Derin Öğrenme Teknikleri Kullanılarak Gerçek Zamanlı Saldırı Tespiti” adlı Doktora tezinin bilimsel etik değerlere ve kurallara uygun, özgün bir çalışma olduğunu, aksinin tespit edilmesi halinde her türlü yasal yaptırımını kabul edeceğimi beyan ederim.

.../.../2024

Ahmet ER

JÜRİ KABUL VE ONAY

Ahmet ER tarafından hazırlanan "**Derin Öğrenme Teknikleri Kullanılarak Gerçek Zamanlı Saldırı Tespiti**" adlı tez çalışmasının savunma sınavı **08.03.2024** tarihinde yapılmış olup aşağıda verilen Jüri tarafından **Oy Birliği** ile Tokat Gaziosmanpaşa Üniversitesi Lisansüstü Eğitim Enstitüsü Mekatronik Mühendisliği Anabilim Dalı'nda Doktora Tezi olarak kabul edilmiştir.

Jüri Üyeleri (Unvanı, Adı Soyadı)	İmzası
Üye (Başkan): Doç. Dr. Levent GÖKREM	
Üye (Danışman): Doç. Dr. A. Hakan YAVUZ	
Üye: Doç. Dr. Bülent TURAN	
Üye: Dr. Öğr. Üyesi Canan ORAL	
Üye: Dr. Öğr. Üyesi Ayşe AYDIN YURDUSEV	

ONAY

...../...../2024

Lisansüstü Eğitim Enstitüsü Müdürü

TEŞEKKÜR

Yoğun iş temposuna rağmen doktora çalışmamın her aşamasında değerli katkılarıyla çalışmayı yönlendiren kıymetli hocam Sayın Doç. Dr. A. Hakan YAVUZ'a,

Bu günlere gelmemi sağlayan ve maddi-manevi desteklerini hiçbir zaman esirgemeyen babam Mehmet ER ve annem Gülsüm ER'e, her zaman olduğu gibi çalışma süresince de desteğini her daim hissettiğim değerli eşim Pınar ER ve biricik kızım Elif Gülce ER'e sonsuz teşekkür ederim.

AHMET ER

Mart, 2024

ÖZET

DERİN ÖĞRENME TEKNİKLERİ KULLANILARAK GERÇEK ZAMANLI SALDIRI TESPİTİ

Er, Ahmet

Doktora Tezi, Mekatronik Mühendisliği Anabilim Dalı

Tez Danışmanı: Doç. Dr. A. Hakan YAVUZ

Mart 2024, xiv + 102 sayfa

Güvenlik, varoluşundan beri insan için en önemli olgulardan biri olmuş ve tüm zamanlar boyunca da önemini artırarak korumuştur. Bu kapsamda bireysel ve kamusal güvenliğin sağlanması için güvenlik kameralarının kullanılması veya görevlilerce nöbet tutulması bir zorunluluk haline gelmiştir. Ancak insan operatörlerin gün boyu ekran başında güvenlik kamerasını veya bir bölgeyi, görev süresi boyunca dikkatini en yüksek seviyede tutarak gözlemesi mümkün değildir. Güvenliğin tesis edilmesi adına görüntülerden elde edilen verilerin, daha az insan operasyonu ile otomatik olarak analiz edilmesini sağlayan güvenlik sistemlerine olan ihtiyaç giderek artmaktadır. Güvenlik uygulamaları göz önüne alındığında gözetleme, hedef tespiti ve hedef takibi gibi çalışmalarda hareket tespiti ve bu hareketin sınıflandırılması önem arz etmektedir. Yürütülen çalışma ile “*drone*” üzerindeki kamera ile elde edilen görüntülerden gerçek zamanlı saldırı tespitinin yapılması amaçlanmıştır. Bu kapsamda görüntülerden hareket tespiti için; sırasıyla optik akış, kamera hareketinin telafisi ve YOLOv8 segmentasyon algoritması, insan aktivitesinin tanınmasında ise; YOLOv8 segmentasyon ve nesne tespiti algoritmaları kullanılmıştır. Ayrıca kamera görüntülerinden nesnenin “*drone*”a olan uzaklığının tespiti yapılmaya çalışılmıştır. YOLOv8’e ait tüm varyantlar (Nano, Small, Medium, Large ve Extra Large) bu çalışma kapsamında ayrı ayrı kullanılmış ve performansları kıyaslanmıştır. Hareket tespitinden en başarılı YOLOv8 varyantı; %89 mAP@0.5 değeri ile YOLOv8m olurken nesne tespitinde, %96 mAP@0.5 ile YOLOv8l olmuştur. Nesnenin “*drone*”a olan uzaklığı ise %1.66 bağıl hata ortalaması ile hesaplanmıştır.

Anahtar Kelimeler: Hareketli Nesne Tespiti, Nesne Takibi, Derin Öğrenme, Mesafe Tespiti, YOLOv8

ABSTRACT

REAL-TIME ATTACK DETECTION USING DEEP LEARNING TECHNIQUES

Er, Ahmet

Doctoral Thesis, Department of Mechatronics Engineering

Advisor: Assoc.Prof. Dr. A. Hakan YAVUZ

March 2024, xiv + 102 pages

Security has been one of the most important concepts for humans since their existence, and it has consistently increased in significance over time. In this context, the use of security cameras or the presence of personnel on duty has become a necessity to ensure both individual and public safety. However, it is not feasible for human operators to continuously monitor a security camera or a specific area with the highest level of attention throughout their shift. The need for security systems that can automatically analyze data from images with fewer human operations is increasingly growing to establish security. When considering security applications, tasks such as surveillance, target detection, and target tracking are crucial, where motion detection and classification of this motion play a significant role. The conducted study aims to achieve real-time attack detection from images obtained from a "drone" camera. Optical flow, camera motion compensation, and YOLOv8 segmentation algorithms were sequentially used for motion detection from images, while for human activity recognition, YOLOv8 segmentation and object detection algorithms were employed. Additionally, attempts were made to detect the object's distance from the "drone" using camera images. All variants of YOLOv8 (Nano, Small, Medium, Large, and Extra Large) were individually utilized in this study, and their performances were compared. The most successful YOLOv8 variant in motion detection achieved a value of 89% mAP@0.5 with YOLOv8m, while in object detection, it was 96% mAP@0.5 with YOLOv8l. The object's distance to the drone was calculated with a relative error average of 1.66%.

Keywords: Motion Object Detection, Object Tracking, Deep Learning, Distance Detection, YOLOv8

İÇİNDEKİLER

Sayfa

ETİK SÖZLEŞME	i
JÜRİ KABUL VE ONAY.....	ii
ÖZET	iv
ABSTRACT.....	v
İÇİNDEKİLER.....	vi
ŞEKİLLER LİSTESİ	ix
SİMGE ve KISALTMALAR	xii
1. GİRİŞ.....	1
2. KURAMSAL TEMELLER VE LİTERATÜR ÖZETLERİ.....	6
2.1. Kuramsal Temeller.....	6
2.1.1. Hareket tespiti.....	6
2.1.2. Nesne tespiti	20
2.1.3. Mesafe tespiti	25
2.1.4. Nesne takibi.....	28
2.2. Literatür Özetleri	28
3. MATERYAL VE YÖNTEM.....	39
3.1. Materyal	39
3.2. Yöntem.....	41
3.2.1. Kamera kalibrasyonu.....	44
3.2.2. Etiketleme işlemleri	46
3.2.3. Derin öğrenme modelinin seçimi.....	48

3.2.4. Saldırı tespitiine yönelik geliştirilen yöntem	54
4. BULGULAR VE TARTIŞMA	68
4.1. Hareket Tespitine İlişkin Elde Edilen Bulgular.....	68
4.2. Nesne Tespitine İlişkin Elde Edilen Bulgular	70
4.3. Mesafe Tespitine İlişkin Elde Edilen Bulgular	83
5. SONUÇ VE ÖNERİLER.....	86
6. KAYNAKLAR.....	90
ÖZGEÇMİŞ	102

TABLÖLAR LİSTESİ

Tablo No	Sayfa
Tablo 3.1. Kullanılan veri setleri	39
Tablo 3.2. DJI Phantom 3 PRO 4K Drone teknik özellikleri	40
Tablo 3.3. Modelin kullanıldığı bilgisayarın teknik özellikleri	41
Tablo 3.4. Etiketleme yapılan sınıflar.....	47
Tablo 3.5. YOLOv8 ile YOLOv5 kıyaslaması (Rath, 2023).....	50
Tablo 3.6. YOLOv8 modeline ait varyantların performansları	51
Tablo 3.7. Görüntü sayısını artırma için kullanılan yöntemler.....	59
Tablo 4.1. Hareket tespiti için YOLOv8 varyantlarının kullanımı.....	69
Tablo 4.2. Hareket tespiti için YOLOv8 varyantlarının performansı.....	70
Tablo 4.3. YOLOv8 nesne tespiti varyantlarının kullanımı	71
Tablo 4.4. Nesne tespiti için YOLOv8 varyantlarının performansı	72
Tablo 4.5. YOLOv8 görüntü segmentasyonu varyantlarının kullanımı	77
Tablo 4.6. Görüntü segmentasyonu için YOLOv8 varyantlarının performansı	78
Tablo 4.7. Mesafe tespiti performansı	84

ŞEKİLLER LİSTESİ

Şekil No	Sayfa
Şekil 1.1. Maslow'un ihtiyaçlar hiyerarşisi (Maslow, 1943).....	1
Şekil 2.1. Sabit kameradan hareket tespiti yöntemleri.....	6
Şekil 2.2. Ardışık kare farkı algoritmasının akış diyagramı	7
Şekil 2.3. Arka plan çıkarımı (Anonim, 2015)	8
Şekil 2.4. Optik akış problemi (Kuklin, 2021a).....	9
Şekil 2.5. FlowNetSimple mimarisi (Fischer ve ark., 2015)	11
Şekil 2.6. FlowNetCorr mimarisi (Fischer ve ark., 2015)	12
Şekil 2.7. Kaba öznitelik haritalarının yüksek çözünürlüklü hale getirilmesi (Fischer ve ark., 2015)	13
Şekil 2.8. RAFT kodlayıcı (Kuklin, 2021b)	14
Şekil 2.9. RAFT Korelasyon Hesabı (Kuklin, 2021b).....	14
Şekil 2.10. RAFT görüntü çıktısı (Kuklin, 2021b)	15
Şekil 2.11. Hareketli kameradan hareket tespiti yöntemleri	16
Şekil 2.12. Panoramik arka plan oluşturma (Anonim, 2020)	17
Şekil 2.13. Yörüngelerin bir alt uzayda (sağda) ve görüntüdeki sonuçta (solda) kümelenmesine bir örnek (Elqursh ve Elgammal, 2012)	18
Şekil 2.14. Sol üstteki görüntüde örnek hareket segmentasyonu. Diğer üç görüntü, görüntüde gözlemlenen üç hareketin optik akışını temsil eder (Zhu ve Elgammal, 2017)	19
Şekil 2.15. Nesne sınıflandırma, nesne tespiti (Elgendy, 2020) ve görüntü segmentasyonu	20
Şekil 2.16. Nesne tespiti yaklaşımları.....	21
Şekil 2.17. CNN yapısı	22
Şekil 2.18. SSD algoritmasının şeması.....	24
Şekil 2.19. YOLO algoritması	25
Şekil 2.20. Uzaklık Tespiti Yaklaşımları (Zhao ve ark., 2020)	26

Şekil 2.21. Üçgenleme (Akarun, 2020)	26
Şekil 3.1. DJI Phantom 3 PRO 4K Drone.....	40
Şekil 3.2. Geliştirilen yazılımın blok diyagramı.....	43
Şekil 3.3. Kamera kalibrasyonu öncesi ve sonrası (Lee ve ark., 2019)	44
Şekil 3.4. Satranç tahtası görseli kullanılarak kamera kalibrasyonu	45
Şekil 3.5. “yaml” formatında kalibrasyon parametreleri	46
Şekil 3.6. Görüntülerin etiketlenmesi	48
Şekil 3.7. YOLO uygulamalarının bibliyometrik ağ görselleştirmesi (Terven ve ark., 2023)	50
Şekil 3.8. YOLOv8 performans karşılaştırması (Jocher ve ark., 2023).....	52
Şekil 3.9. Karışıklık matrisi	52
Şekil 3.10. IoU hesaplanması	54
Şekil 3.11. Ardışık kare kuyruğu	55
Şekil 3.12. Farklı yüksekliklerden elde edilen hareketli nesnelere ait normalize edilmiş görüntüler	56
Şekil 3.13. Hareket tespiti için önerilen yöntemin blok diyagramı	57
Şekil 3.14. Görüntü artırımı işlemi	58
Şekil 3.15. Kamera çapraz görüş alanı (FOV), dikey görüş alanı (VFOV), yatay görüş alanı (HFOV)	61
Şekil 3.16. “Drone” kamerasından mesafe tespiti	62
Şekil 3.17. Mesafe tespitini saldırı tespitinde kullanımı.....	62
Şekil 3.18. Saldırı tespitinde kullanılan sınıflara ait görseller	65
Şekil 3.19. Örnek, saldırgan davranış grafiği	66
Şekil 4.1. YOLOv8 ile hareket tespiti.....	70
Şekil 4.2. Nesne tespiti modelinin öğrenme döngüsü sürecine ait a) kesinlik, b) duyarlılık, c) mAP@0.5, d) doğrulama kaybı grafikleri.....	74
Şekil 4.3. YOLOv8 nesne tespiti varyantlarıyla eğitilen modellerin karışıklık matrisleri	76
Şekil 4.4. Elinde bir şey taşıyan (carrying_something) sınıfına ait örnek çıktı.....	76
Şekil 4.5. Hedef alan (aiming) sınıfına ait örnek çıktı.....	77

Şekil 4.6. Çalışmada kullanılan görüntü segmentasyonu modelinin öğrenme döngüsü sürecine ait a) kesinlik, b) duyarlılık, c) mAP@0.5, d) doğrulama kaybı grafikleri ..	80
Şekil 4.7. YOLOv8 görüntü segmentasyonu varyantlarıyla eğitilen modellerin karışıklık matrisleri	82
Şekil 4.8. Koşan kişiyi ifade eden “running” ve ayakta bekleyen kişiyi temsil eden “standing” sınıflarına ait örnek çıktı görselleri	82
Şekil 4.9. Silahla hedef alan kimseyi ifade eden “aiming”, gizlenen kimseyi temsil eden “hiding” ve araçları ifade eden “vehicle” sınıflarına ait örnek çıktı görselleri ..	83
Şekil 4.10. Futbol sahasında mesafe ölçümü testi	84
Şekil 4.11. Mesafe tespiti için örnek çıktı	85

SİMGE ve KISALTMALAR

Simgeler

Açıklama

%	Yüzde
°	Derece (Açı)
dk	Dakika
g	Gram
h	Yükseklik
m	Metre
ms	Milisaniye
s	Saniye
sa.	Saat
α	Alfa
δ	Gama
φ	Teta

Kısaltmalar

Açıklama

3B	3 Boyutlu
AP	Average Precision (Ortalama Kesinlik)
AR	Augmented Reality (Artırılmış Gerçeklik)
AUC	Area Under The Curve (Eğrinin Altındaki Alan)
CNNs	Convolutional Neural Networks (Evrişimsel Sinir Ağları)
CPU	Central Process Unit (Merkezi İşlem Birimi)
DARPA	The Defense Advanced Research Projects Agency (İleri Savunma Araştırma Projeleri Kurumu)
DCNN	Deep Convolutional Neural Networks (Derin Evrişimsel Sinir Ağları)
DDD	Dull, Dangerous, Dirty (Sıkıcı, Tehlikeli, Kirli)

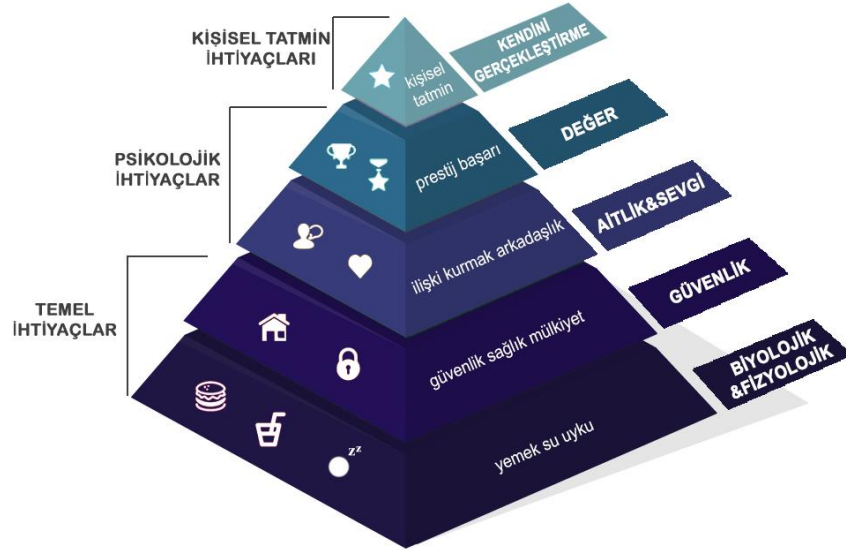
Faster RCNN	Faster Region-based Convolutional Neural Network (Daha Hızlı Bölge Tabanlı Evrişimli Sinir Ağı)
FN	False Negative (Yanlış Negatif)
FOV	Field of View (Görüş Açısı)
FP	False Positive (Yanlış Pozitif)
FPS	Frame Per Second (Saniyedeki Kare Sayısı)
GAN	Generative Discriminative Network (Üretken Çekişmeli Ağ)
GB	Gigabyte
GHz	Gigahertz
GPS	Global Positioning System (Küresel Konumlama Sistemi)
GPU	Graphics Processing Unit (Grafik İşlem Birimi)
GRU	Gated Recurrent Unit (Geçitli Tekrarlayan Birim)
HD	High-Definition (Yüksek-Çözünürlük)
HFOV	Horizontal Field of View (Yatay Görüş Açısı)
HOG	Histogram of Oriented Gradients (Yönlendirilmiş Gradyanların Histogramları)
Hz	Hertz
IMU	Inertial Measurement Unit (Eylemsiz Ölçüm Birimine)
IoU	Intersection over Union (Kesiştirilmiş Bölgeler)
İHA	İnsansız Hava Aracı
KITTI	Karlsruhe Institute of Technology and Toyota Technological Institute
LIDAR	Light Detection and Ranging (Işık Tespiti ve Uzaklık Tayini)
LSTM	Long Short-Term Memory (Uzun Kısa Süreli Bellek Ağları)
mAP	Mean Average Precision (Ortalama Kesinlik Ortalaması)
MHz	Megahertz
MLP	Multi Layer Perceptron (Çok Katmanlı Algılayıcı)
MOT	Multiple Object Tracking (Çoklu Nense Takibi)
MSE	Mean Square Error (Ortalama Kare Hata)
NMS	Non-maximum Suppression (Maksimum Olmayan Bastırma)

OCSVM	One Class Support Vector Machine (Tek-Sınıf Destek Vektör Makineleri)
RAFT	Recurrent All-Pairs Field Transforms (Tekrarlayan Tüm Çiftler Alan Dönüşümleri)
R-CNN	Region-based Convolutional Neural Network (Bölge Tabanlı Evrişimli Sinir Ağı)
RGB-D	Red Green Blue-Depth (Kırmızı Yeşil Mavi-Derinlik)
RNN	Recurrent Neural Network (Yinelemeli Sinir Ağı)
SIFT	Scale Invariant Feature Transform (Ölçek Değişmez Unsur Dönüşümü)
SSD	Single Shot Multibox Detector (Tek Atış Çoklu Kutu Algılama)
SVM	Support Vector Machine (Destek Vektör Makinesi)
TDK	Türk Dil Kurumu
TN	True Negative (Doğru Negatif)
TP	True Positive (Doğru Pozitif)
UCF	University of Central Florida
VFOV	Vertical Field of View (Dikey Görüş Açısı)
VIRAT	Video Image Retrieval and Analysis Tool (Video Görüntü Alma ve Analiz Aracı)
YOLO	You Only Look Once (Sadece Bir Kez Bakarsın)

1. GİRİŞ

Güvenlik, varoluşundan beri insan için en önemli olgulardan biri olmuş ve tüm zamanlar boyunca da önemini artırarak korumuştur. Türk Dil Kurumu (TDK) sözlüğünde güvenlik; “toplum yaşamında yasal düzenin bozulmadan yürütülmesi, kişilerin korkusuzca yaşayabilmesi durumu, emniyet” olarak tanımlanmaktadır (TDK, 2020).

Abraham Maslow’un ihtiyaçlar hiyerarşisi teorisinde güvenliğin önemi vurgulanarak insanın temel ihtiyaçları arasında gösterilmiştir (Şekil 1.1). Yemek yeme, su içme gibi fizyolojik ihtiyaçlardan sonra yer alan güvenlik olgusu; ait olma, sevgi ihtiyacı, değer görme ve kendini gerçekleştirme ihtiyacının giderilmesi için bir ön koşul bir temel olarak belirtilmiştir. İnsanoğlu bu ihtiyacını karşılamak adına geleneksel güvenlik anlayışı doğrultusunda silahlı kuvvetler gibi kendini güven içinde hissedeceği unsurlar oluşturmuştur. Ancak teknolojik gelişmeler insanın geleneksel güvenlik anlayışının yetersiz olduğunu göstermiş ve bambaşka bir boyuta evirmiştir.



Şekil 1.1. Maslow’un ihtiyaçlar hiyerarşisi (Maslow, 1943)

Son yirmi yılda dünya, farklı sosyo-ekonomik sektörlerde, bireylerin emniyeti ve güvenliği de dahil olmak üzere yaşamın farklı açılarında bir gelişmeye tanık olmuş dolayısıyla güvenlik altına alma ve izleme ihtiyacı bir zorunluluk haline gelmiştir. Kamusal ve özel alanlara kurulan güvenlik kameraları yardımıyla güvenliğin sağlanması uygun bir çözüm olmuştur. Genel olarak insan kaynakları bu kameraları günün 24 saati izlemeye adanmış ve bu da sürekli izlemeyi çok pahalı hale getirmiştir. Bu nedenle, güvenlik kameralarının kullanıldığı farklı konumlardaki olayları izlemek için otomatikleştirilmiş bir sisteme büyük ihtiyaç duyulmaktadır. Gözetim yoluyla sağlanan verilerin daha az insan operasyonu ile otomatik olarak analiz edilmesini sağlayan güvenlik sistemlerine olan ihtiyaç giderek artmaktadır (Elharrouss ve ark., 2021). Bunun yanında askeri amaçlı güvenli alanların oluşturulması göz önünde bulundurulduğunda nöbet tutan askerlerin olası saldırılara doğrudan maruz kalabilecek olması durumu, bu tür sistemlerin kullanımını daha anlamlı kılan diğer bir gerekçedir.

Geleneksel video gözetim sisteminde güvenlik görevlisinin şüpheli etkinliği izlemek ve incelemek için tüm gün oturması gerekmektedir. Güvenlik görevlisi sürekli olarak yüzlerce farklı kameraya bakmak zorunda kalabilmektedir. Tek bir güvenlik görevlisinin tüm gün kamera izlemesi mümkün değildir. Görevlilerin sürekli değişmesi ve farklı görevlinin şüpheli faaliyetlere karşı farklı algıları olabilmektedir. İnsanlar iki ekrandan fazlasını etkili bir şekilde izleyememekte ve her şeyi sürekli olarak denetleyememektedir. Kişi tek bir ekranı 30 dakika boyunca izlese bile insan gözü aktivitenin yalnızca %20'sini algılamaktadır. İnsan gözü bir süre sonra yorulduğundan birçok bilgi kaybı yaşanabilmektedir. Bu, güvenlik sistemi açısından akıllı, güvenilir ve etkili bir yaklaşım olmadığı gibi kişinin sağlığını da etkileyebilmektedir (Ghode ve Bhagat, 2019).

Güvenlik amaçlı gözetleme görevlerinde sabit kameralardan elde edilen görüntülerin yanında kamera ile donatılmış “*drone*”lardan elde edilen hareketli kameraya ait görüntüler de kullanılmaktadır. Geçtiğimiz yıllarda, küresel İnsansız Hava Araçları (İHA) pazarı; trafik izleme, arama kurtarma operasyonları, gözetleme ve biyokimyasal

algılama gibi geniş sivil ve askeri uygulamalar nedeniyle artmakta, hükümetlerin ve ticari endüstrilerin daha fazla ilgisini çekmektedir (Hayat ve ark., 2016; Gupta ve ark., 2015; Motlagh ve ark., 2016). Günümüzde İHA olarak da bilinen “*drone*” lar herkes tarafından ulaşılabilir duruma gelmiş olup sivil ve askeri alanlarda kullanımı sıkça karşımıza çıkmaktadır. Güvenlik uygulamaları düşünüldüğünde gözetleme, hedef tespiti ve hedef takibi gibi geniş çaplı uygulamaları mevcuttur (Delibaşoğlu, 2021; Mekdad, 2023).

Uluslararası literatürde “*drone*” veya “İnsansız Hava Aracı (İHA)” olarak tanımlanan araçlar, üzerinde yolcusu ve pilotu olmayan, uzaktan kontrol edilebilen amaca uygun kamera, lazer tarama cihazı gibi ekipmanlarla donatılan bir çeşit hava aracı ve uçan robot olarak tanımlanmaktadır. İHA’ların ilk kullanımı 1.Dünya Savaşı’na kadar dayanmakta olup günümüzde birçok farklı amaç için kullanılabilmektedir (Kahveci ve Nazlı, 2017). Genellikle DDD (Dull, Dangerous, Dirty) olarak ifade edilen insan tarafından yapılması *sıkıcı, tehlikeli ve kirli* olan görevler için İHA kullanımı tercih edilmektedir (Austin, 2011). Bu araçlar operasyonel basitliği ve hareketliliğinden dolayı arama kurtarma ve gözetleme gibi uygulamalarda konuşlandırılabilir (Perera ve ark., 2018).

Otomatize edilmiş gözetleme sistemleri güncel bir araştırma konusudur. Artan güvenlik talebiyle birlikte ülke sınırları ve askeri üsler gibi güvenlik açısından kritik öneme sahip alanların insan gözlemciler tarafından sürekli bir şekilde izlenmesi ve analiz edilmesi mümkün değildir (Sincan ve ark., 2018). Bu kapsamda İHA’lar üzerinden görüntüleme teknolojileri ile gerçek zamanlı insan tespiti, yörünge tespiti, şüpheli hareket tespitine yönelik çeşitli çalışmalar yapılmıştır (Yang ve ark., 2019).

Görüntüden hareket tespitinin yapılması çoğu güvenlik uygulaması için temel araştırma konusu niteliğindedir. Akıllı video gözetimi çalışmalarında hareketli nesne tespiti önemli bir rol oynamaktadır. Özellikle kameranın hareketli olduğu durumlarda bu hareketli nesnelerin tespiti zorlu bir problem haline gelmektedir (Zhu ve ark., 2020).

Bir bölgenin güvenliğini sağlamak amacıyla gözetlenmesi işleminde hareket tespiti yanında bu harekete ilişkin anormallik tespiti veya insan hareketindeki şüpheli durumu algılamak da önemlidir. Bu bağlamda yapılan çalışmalarda silah veya anormallik tespitine yönelik araştırmalar yapılmıştır. Anormal davranış tespiti daha genel bir ifade ile insan eylemi tanıma ve sınıflandırma eylemi ile ilişkilidir (Mehmood, 2021).

Bu tez çalışmasında güvenlik açısından kritik öneme sahip olan veya gözetlenmek istenen alanların otomatik olarak izlenmesi, görüntülerin görüntü işleme ve derin öğrenme yöntemleri yardımıyla analiz edilmesi ve olası bir saldırıdan önce uyarı verecek bir sistemin tasarlanması hedeflenmiştir. Burada şüpheli bir hareket veya durumun tespit edilmesi halinde sistem bunu ilgili kişiye bildirecek ve gerekli aksiyonların alınmasına yardımcı olacaktır. Çalışma kapsamında saldırının tanımlanması; hareket tespiti, nesne tespiti ve mesafe tespiti başlıkları altında ele alınmıştır.

Tez çalışmasında:

- Farklı arazi şartlarında elde edilen “*drone*” görüntülerinden oluşan bir veri setinin oluşturulması,
- Oluşturulan veri seti ve daha önce yapılan benzer çalışmalarda kullanılmış veri setleri kullanılarak “*drone*” görüntüleri üzerinden hareketli nesne tespiti yapılması,
- Nesnenin “*drone*”a göre tahmini uzaklığının hesaplanması,
- Tespiti yapılan nesnenin sınıflandırılması ve insan olup olmadığının anlaşılması,
- Nesne eğer insansa, insan hareketlerinin tehlike arz edip etmediğine karar verilmesi,
- Son olarak hareketlerin tehlike arz etmesi durumunda söz konusu insanın görüntüde takibini yapılması

amaçlanmıştır.

Bu bağlamda tez çalışması beş bölümden oluşmaktadır. Birinci bölüm olan “Giriş” bölümünde tezin amacı ve içeriğine ait genel bilgilere yer verilmiştir. İkinci bölümde ise video görüntülerinden hareket tespiti, nesne sınıflandırma, nesne takibi ve mesafe ölçümüne yönelik literatür araştırmasına değinilmiştir. Üçüncü bölüm olan “Materyal ve Yöntem” bölümünde ise çalışmada kullanılan algoritma ve modellerden bahsedilmiştir. Dördüncü bölüm “Bulgular ve Tartışma” da, kullanılan veri setleri üzerinde geliştirilen hibrit modelin başarısı değerlendirilmiş ve tartışılmıştır. Beşinci ve son bölümde ise çalışmanın sonuçları ortaya konulmuş ve gelecek çalışmalar için önerilerde bulunulmuştur.

2. KURAMSAL TEMELLER VE LİTERATÜR ÖZETLERİ

Bu bölümde saldırının tanımlanmasında kullanılan hareket tespiti, nesne tespiti, mesafe tespiti ve nesne takibine yönelik tanımlara ile gerçekleştirilmiş örnek çalışmalara ilişkin bilgilere yer verilmektedir.

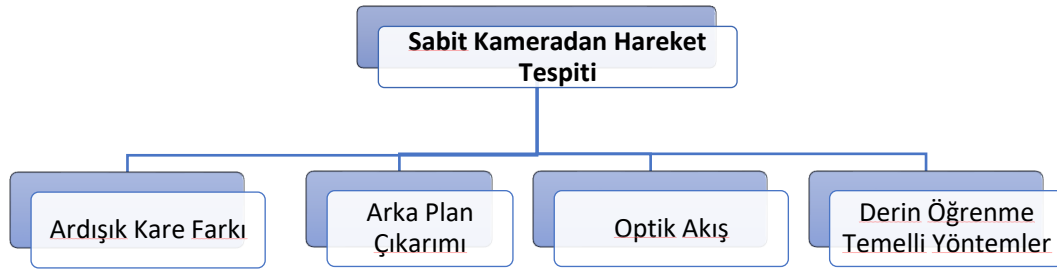
2.1. Kuramsal Temeller

2.1.1. Hareket tespiti

Kamera görüntüleri üzerinden hareket tespitine yönelik yapılan çalışmalar, kameranın sabit veya hareketli olması durumuna göre iki ana gruba ayrılmaktadır. Kameranın hareket durumuna göre araştırma yaklaşımları farklılaşmaktadır.

Sabit kamera görüntülerinden hareket tespiti

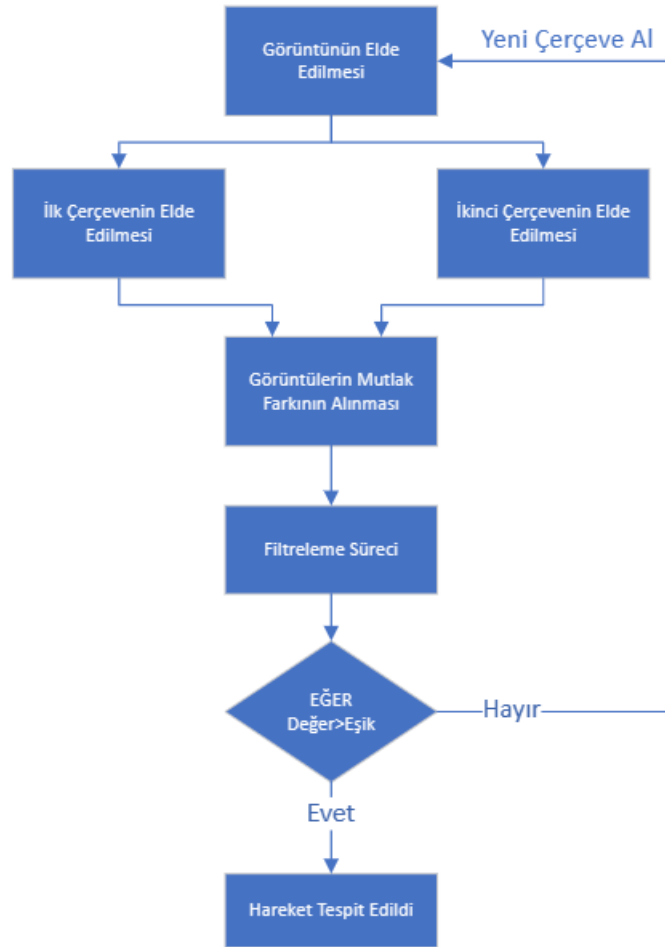
Hareketli nesnelerin tespiti için kameranın sabit olduğu durumlarda kullanılan üç ana yaklaşım vardır. Bunlar video görüntülerinde ardı ardına gelen karelerin (frame) farkını almak, arka plan çıkarımı ve optik akış yöntemleridir (Şekil 2.1).



Şekil 2.1. Sabit kameradan hareket tespiti yöntemleri

Ardışık kare farkı (consecutive frame difference)

Ardışık kare farkı yöntemi, nesneleri algılamaya yönelik olarak kullanılan uygulaması çok basit ama zorluklara karşı bir o kadar hassas olan bir metottur. Bu yöntemde videodaki kare dizisinde önceki kareden güncel kare çıkarılmaktadır (Şekil 2.2). Böylece hareketli nesneye ait pikseller tespit edilebilmektedir (Ghode ve Bhagat, 2019). Burada esas kilit nokta kameranın sabit olduğunu varsaymaktır. Kamera hareketli olduğunda ise bu hareketi telafi edecek farklı yöntemlere başvurulması gerekmektedir. Kamera sabit olsa dahi bu yöntemde görüntüde yer alan gölgeler yanlış tespitlere sebebiyet verebilmektedir.

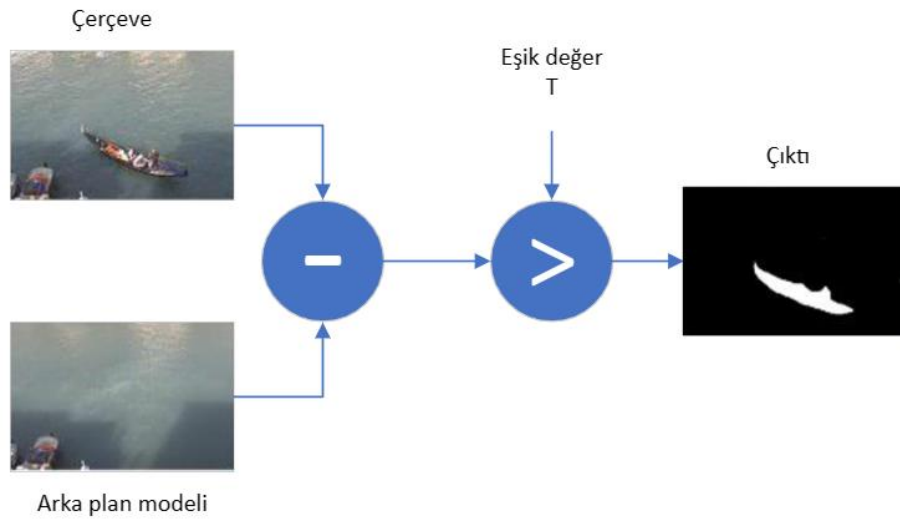


Şekil 2.2. Ardışık kare farkı algoritmasının akış diyagramı

Arka plan çıkarımı (background subtraction)

Yaklaşık otuz yıl boyunca birçok araştırma ekibi farklı ortamlarda nesnelerin tespiti konusu üzerine çalışmalar yürütmüştür. Görüntüden hareket tespiti için kullanılan geleneksel yöntemlerden biri olan arka plan çıkarımı (background subtraction) yönteminde kameranın sabit olduğu durumda hareketli nesne başarılı bir şekilde tespit edilmektedir. Ancak bu yöntem kameranın hareketli olduğu durumda kamera hareketinden kaynaklı gürültüden dolayı yanlış sonuçlar doğurabilmektedir (Das ve Saharia, 2014).

Hareketli nesneleri tespit etmek için en popüler yöntem olan arka plan çıkarma, güçlü tespit ile gerçek zamanlı gereksinim arasındaki en iyi uyuma sahip yaklaşımdır. Bu yöntemle geliştirilen algoritmalar bir videonun hareketli kısımlarını arka plan görüntüsü ve ön plan görüntüsüyle karşılaştırarak çalışmaktadır. Şekil 2.3.'te görüleceği üzere ön plandaki nesneleri, hiçbir nesnenin olmadığı kareyle karşılaştırarak aralarındaki farkları bulacak ve bir mesafe matrisi oluşturacaktır (Verma ve ark., 2022).



Şekil 2.3. Arka plan çıkarımı (Anonim, 2015)

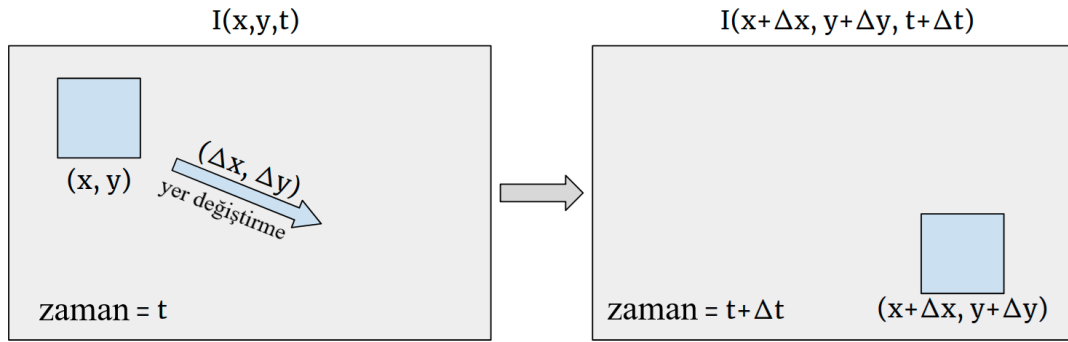
Optik Akış (Optical Flow)

Bu yöntem diğerlerine kıyasla daha güçlüdür ancak yöntemin, gerçek zamanlı çalışmalarda halen kaynak ihtiyacı çok yüksektir.

Video işlemede kareler (frame) arası değişimi tespit etmek önemli bir konudur (Kızrak, 2020). Optik akış (optical flow) art arda gelen video karelerinden hareketli nesneleri tespit etmek için kullanılan bir modeldir (Lin ve ark., 2019). Bu model ile değişimin yönü ve hızı elde edilebilmekte olup akışın ilk karede sabit olduğu varsayılmaktadır (Keçeli ve Kaya, 2018).

Optik akışın (optical flow) çalışma mekanizması şu şekilde özetlenebilir:

Gri tonlamalı bir görüntüye sahip olduğumuz durumda görüntü yoğunluğu (I) ile ifade edilmektedir. Bu fonksiyonu Şekil 2.4.'te görüldüğü üzere $I(x, y, t)$ şeklinde tanımlanmaktadır. Burada (x, y) piksellerin koordinatlarını, t ise kare numarasını (frame numarasını) temsil etmektedir.



Şekil 2.4. Optik akış problemi (Kuklin, 2021a)

Optik akışta ilgili objeye ait piksellerin yoğunluğunun/parlaklığının değişmediği varsayılmaktadır. Bu varsayım sonucu $I(x, y, t) = I(x + \Delta x, y + \Delta y, t + \Delta t)$ eşitliği elde edilmektedir. Söz konusu eşitliğe birinci dereceden Taylor Açılımı yapıldığında;

$$\begin{aligned}
I(x + dx, y + dy, t + dt) &= I(x, y, t) + \frac{\partial I}{\partial x} u + \frac{\partial I}{\partial y} v + \frac{\partial I}{\partial t} t + \dots \\
&= \frac{\partial I}{\partial x} dx + \frac{\partial I}{\partial y} dy + \frac{\partial I}{\partial t} dt = 0
\end{aligned} \tag{1}$$

Eşitlik 1'e ulaşılmaktadır. Bu denklem dt 'ye bölüldüğünde Eşitlik 2 ve Eşitlik 3'teki optik akış eşitliği elde edilmektedir (Lin, 2019).

$$= \frac{\partial I}{\partial x} u + \frac{\partial I}{\partial y} v + \frac{\partial I}{\partial t} t = 0 \tag{2}$$

$$I_x u + I_y v = -I_t \tag{3}$$

Optik akış çoğu uygulama alanı için oldukça kullanışlıdır. Örneğin; trafik analizi, araç takibi, nesne tespiti, nesne takibi ve daha fazlası için optik akış ile başarılı sonuçlar elde edilebilmektedir. Tek problemi yavaş olması ve çok fazla bilgi işlem kaynağı gereksinimidir (Teed ve Deng, 2020).

Derin Öğrenme Yaklaşımlı Yöntemler

Evrşimsel Ağlarla Optik Akış Öğrenme (FlowNet)

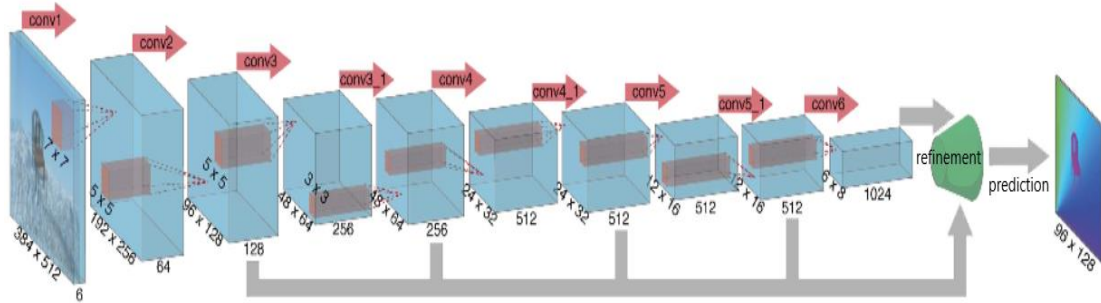
Evrşimli sinir ağları (Convolutional Neural Networks- CNNs) çoğu alanda olduğu gibi bilgisayarlı görüde de oldukça başarılı olmuştur. Ancak optik akış CNN'nin başarı sağladığı bu alanlardan olmamıştır. Optik akış problemine bir denetimli öğrenme görevi olarak yaklaşılmış ve çözüm üretilmiştir. FlowNet ile; Sintel ve KITTI (Karlsruhe Institute of Technology and Toyota Technological Institute) gibi veri setleri üzerinde başarılı genellemeler elde edilmiş ve 5 fps ile 10 fps değerlerinde rekabetçi sonuçlar elde edilmiştir (Fischer ve ark., 2015).

Geri beslemeli evrşimli sinir ağları büyük ölçekli görüntüleri sınıflandırmada iyi performans göstermiştir (Krizhevsky ve ark., 2012). Bundan dolayı bilgisayarlı görüde

CNN'nin kullanımında bir artış gerçekleşmiştir. CNN kullanımına sahip uygulamalara örnek olarak görüntü bölütleme (semantic segmentation) (Farabet ve ark., 2012; Hariharan ve ark., 2015; Girshick ve ark., 2014), derinlik tahmini (Eigen ve ark., 2014), anahtar nokta tahmini (keypoint prediction) (Hariharan ve ark., 2015) ve sınır tespiti (Ganin ve ark. 2014) verilebilir. Bu çalışmalar, her bir piksel için tahminler içerdiğinden optik akış tahminine benzerlik göstermektedir.

Fischer ve ark. (2015), yaptığı çalışmada iki mimari önermekte ve karşılaştırmaktadır. Birincisi genel bir CNN mimarisi, ikincisi ise görüntünün farklı konumlarındaki özellik vektörlerini ilişkilendiren bir katman içeren mimaridir. Bu mimariler FlowNetS ve FlowNetCorr olarak isimlendirilmiştir. Her iki mimari de kodlayıcı (encoder) ve kod çözücü (decoder) bölüme sahiptir. Kodlayıcı ardışık iki görüntüden özellikleri çıkarırken kod çözücü özellik haritalarını yükseltir ve optik akışı tahmin etmektedir (Kuklin, 2021a).

FlowNetSimple;



Şekil 2.5. FlowNetSimple mimarisi (Fischer ve ark., 2015)

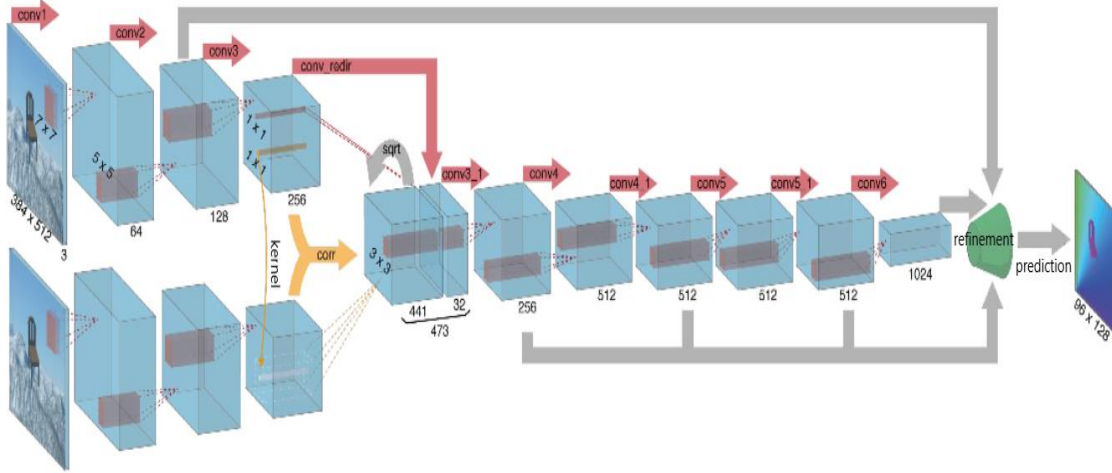
Şekil 2.5.'te gösterilen bu mimari girdi olarak art arda gelen iki karenin (frame) birleşimini almaktadır. Bu iki kare 6 kanallı bir tensörü - ilk 3 (w, h, c) kanal ilk kareye ait, ikinci 3 (w, h, c) kanal ikinci kareye ait - oluşturmaktadır. Kodlayıcı kısmı aktivasyon fonksiyonları tarafından takip edilen birkaç evrişim katmanından oluşmaktadır ve optik akışı tahmin etmeyi öğrenebilmektedir (Fischer ve ark., 2015).

FlowNetCorr;

FlowNetCorr mimarisinde girdi olarak her seferinde bir kare alınmaktadır. Bu ağıda art arda gelen iki kareden ayrı ayrı özellik çıkarmak için CNN'nin ilk aşamaları kullanılmaktadır. Sonraki adımda ise elde edilen iki özellik haritası birleştirilmektedir. Bunun sağlanması için FlowNetSimple ile FlowNetCorr arasındaki en büyük fark olan Korelasyon Katmanı (Correlation Layer) kullanılmıştır. Bu kısımda eğitilebilir çekirdekler yoktur, bir özellik haritası diğeri ile evrilmektedir. Korelasyon formülü Eşitlik 4'teki gibidir.

$$c(x1, x2) = \sum_{o \in [-k, k] \times [-k, k]} \langle f1(x1 + o), f2(x2 + o) \rangle \quad (4)$$

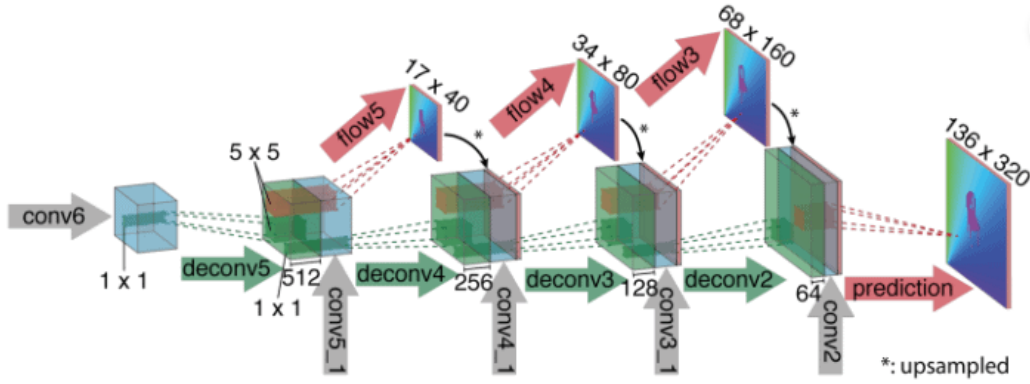
Korelasyon sonucu üst düzey nitelikleri çıkarmak için evrişim katmanlarının geri kalanına iletilmektedir. FlowNetCorr mimarisi Şekil 2.6.'da gösterilmiştir.



Şekil 2.6. FlowNetCorr mimarisi (Fischer ve ark., 2015)

Kodlayıcıdan elde edilen sonuçlar girdi boyutuna göre 64 kat daha küçük olduğundan bu çıktıyı yükseltmeye ihtiyaç duyulmaktadır. Şekil 2.7'de bu çıktıyı yüksek

çözünürlüğe yükseltme için, hesaplama açısından daha maliyetsiz bir yöntem olan çift doğrusal yukarı örnekleme kullanmak daha iyi sonuç elde etmeye olanak sağlamıştır (Kuklin, 2021).



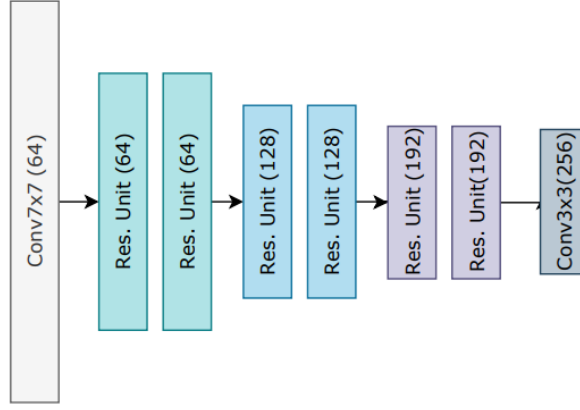
Şekil 2.7. Kaba öznetelik haritalarının yüksek çözünürlüklü hale getirilmesi (Fischer ve ark., 2015)

RAFT (Recurrent All-Pairs Field Transforms)

RAFT (Recurrent All-Pairs Field Transforms) optic flow için 2020 yılında geliştirilmiş bir derin öğrenme mimarisidir. Teed ve Deng (2020) tarafından geliştirilen bu model ile optik akışın (optical flow) eksikliklerinin giderilmesinin yanında hareket tespitinde çok daha fazlasını başarmıştır. Bu çalışmada ardışık bir çift RGB renk uzayına sahip görüntüden (I_1, I_2), I_1 'deki her pikseli (u, v) olarak haritalanarak I_2 'de bu pikseller (u', v') olarak ifade edilmektedir. Yer değiştirme alanının (f_1, f_2) tahmin edilebilmesi için I_2 'de karşılık gelen her piksel $(u', v') = (u + f_1(u), v + f_2(v))$ şeklinde formüle edilmektedir. Model temel olarak 3 ana bileşenden oluşmaktadır: (1) özellik çıkarımı, (2) benzerlik hesaplama, (3) tüm aşamaların türevlenebilir olduğu ve uçtan uca eğitilebilir bir mimaride oluşturulduğu yinelemeli güncelleme kısmıdır.

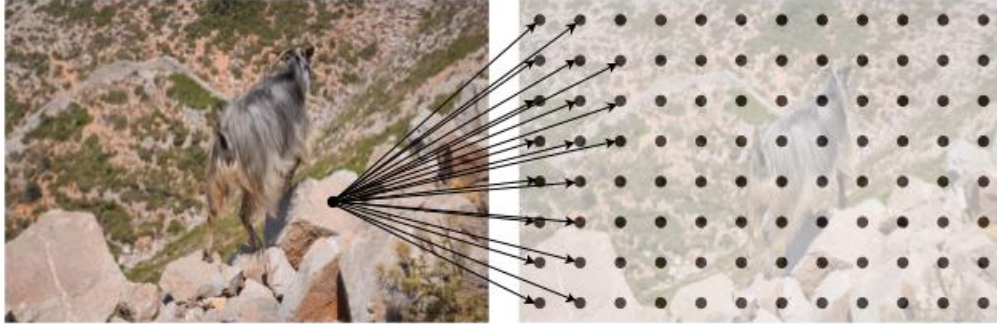
Özellik Çıkarımı: Görüntüdeki bağlamı anlamak amacıyla özellikleri yalnızca ilk kareden ayıklayan başka bir kodlayıcı ile birlikte art arda gelen iki farklı kareden piksel

başına özellikleri çıkaran bir “encoder”dır. Özellikler konvolüsyonel bir ağ kullanılarak elde edilmektedir. Şekil 2.8.’de RAFT encoder yapısı gösterilmektedir:



Şekil 2.8. RAFT kodlayıcı (Kuklin, 2021b)

Benzerlik Hesaplama: Özellik çıkarımının ardından tüm bu özellik vektör çiftlerini, her iki karenin genişliğini ve yüksekliğini kullanarak korelasyon hacmi denilen 4 boyutlu bir tensör oluşturulmaktadır. Bu tensör büyük ve küçük piksel yer değişimlerinin bilgisini taşımaktadır. Burada korelasyon, iki karenin tüm özellik haritası arasında hesaplanmaktadır (Şekil 2.9).



Şekil 2.9. RAFT Korelasyon Hesabı (Kuklin, 2021b)

Güncelleme: Son kısım ise yinelemeli bir şekilde optik akış güncelleyen bir güncelleme operatördür. Bu bölüm GRU (Gated Recurrent Unit) bloğunun olduğu kısımdır. Önceki

korelasyon hacimlerinden değerler olarak akış alanını yinelemeli olarak güncellemektedir. Teed ve Deng (2020) optical flow akışlarını yinelemeli olarak geliştirmek için bir LSTM (Long-Short Term Memory) benzeri olan bir GRU bloğu kullanmıştır. Bu blok, eğitim sırasında sabit akış alanlarını kullanarak yakınsamaya izin verirken, bu yinelemeler arasındaki ağırlıkları paylaşmalarına olanak tanımaktadır.

Ağın pikseller arasında bir gömme kaybı kullanılarak eğitilmesini gerektirmek yerine doğrudan optik akış üzerinden eğitilebilir, bu da onu çok daha verimli hale getirmektedir. Ardından, akış tahmini ile ilgili olarak mevcut yöntemler, onu bir çift kare arasında doğrudan tahmin etmektedir. RAFT algoritmasının örnek çıktı görüntüsü Şekil 2.10'da gösterilmektedir.

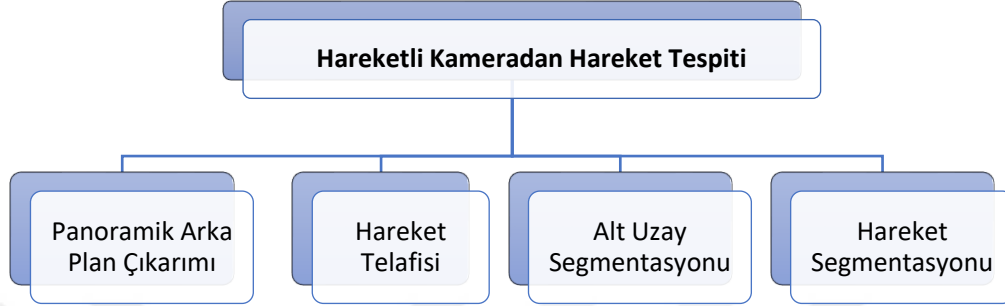


Şekil 2.10. RAFT görüntü çıktısı (Kuklin, 2021b)

Hareketli kamera görüntülerinden hareket tespiti

Hareket tespitini yönelik çalışmalar önceleri, sabit kameralardan hareket tespiti üzerine yoğunlaşırken sonrasında yapılan araştırmalar hareketli kameralar üzerine olmuştur. Bu doğrultuda farklı yöntemler ve metotlar geliştirilmiştir. Hareketsiz kamera görüntülerinden hareket tespiti için önerilen yöntemler hareketli bir kamera ile elde edilen görüntüler üzerinde doğrudan uygulanamamaktadır. Hareketli kamera için

önerilen hareket tespiti yöntemlerinin çoğu arka plan çıkarma fikrinin uygulandığı ya da ilham alındığı yöntemlerdir (Chapel ve Bouwmans, 2020).

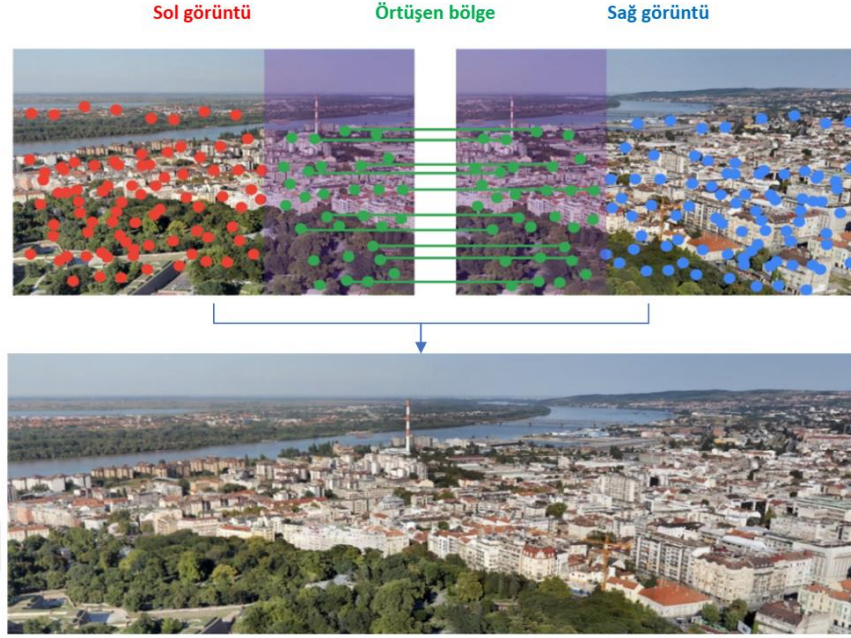


Şekil 2.11. Hareketli kameradan hareket tespiti yöntemleri

Şekil 2.11’de görüldüğü üzere hareketli kameradan hareket tespiti için başlıca kullanılan yöntemler panoramik arka plan çıkarmı, hareket telifisi, alt uzak segmentasyonu ve hareket segmentasyonu yöntemleridir.

Panoramik Arka Plan Çıkarımı

Hareketli bir kamera tarafından çekilen görüntüler, panorama veya mozaik adı verilen daha büyük bir görüntüyü oluşturmak üzere bir araya getirilmektedir (Şekil 2.12). Bu panorama, arka planı modellemek ve statik bir kamerada olduğu gibi hareketli nesneleri algılamak için kullanılabilir. Panoramik görüntünün oluşturulması yüksek doğruluk gerektiren önemli bir adımdır. Geçerli karede ön plan tespitini hesaplamak için görüntünün arka planının kaydedilmesi gerekmektedir. Mozaik oluşturmada görüntüleri hizalamak için üç teknik kullanılmaktadır. Bunlar; kareden kareye: hizalama parametrelerinin ardışık kare çiftlerinden hesaplanmasıdır. Kareden mozaik: mozaik bir kareden daha büyük olduğundan, bir kareyi mozaik hizalamak için büyük yer değiştirme işleminin yapılması gerekmektedir ve önceki kare ile mozaik arasında parametre hesaplaması yapılmaktadır. Mozaikten kareye: diğer iki yöntemin aksine burada mozaik kareye göre hizalanmaktadır.



Şekil 2.12. Panoramik arka plan oluşturma (Anonim, 2020)

Hareket Telifisi

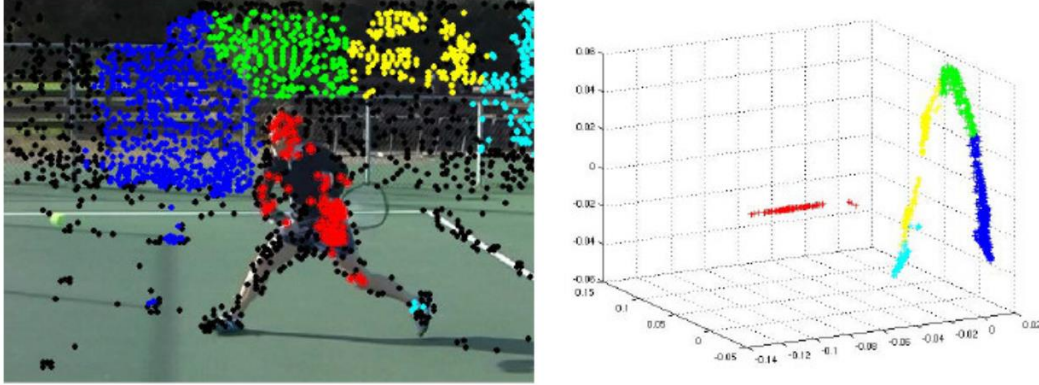
Hareket telifisi hesaplamada, kameranın ve nesnelerin hareketini hesaba katarak önceki ve gelecek karelere göre videodaki bir kareyi tahmin etmek için kullanılan algoritmik bir tekniktir. Arka plan çıkarma yöntemini hareketli bir kameraya uyarlamanın en basit tekniklerinden biridir ve amaç sabit bir kamera durumunda olduğu gibi ardışık kareleri birbirilerinden çıkarma işlemini gerçekleştirmek için kameranın hareketini telafi etmektir. Bir kareden seçilen özellik noktalarının mevcut karedeki konumları kullanılarak kamera hareketini ifade eden homografi matrisi (dönüşüm matrisi) hesaplanmaktadır. Bu matris ile kareler hizalanarak ve birbirlerinden çıkarılarak görüntüdeki hareketli kısım arka plandan ayrıştırılmaya çalışılmaktadır (Delibasoglu, 2021).

Kameranın hareket ettiği durumda, arka planın görünür kısmı bir kareden diğerine zamanla değişmektedir. t zamanındaki arka plan görüntüsü, kameranın görüş alanında

hala görülebilen önceki $t-1$ zamanındaki sahne parçalarından ve mevcut görüntüde görünen yeni sahne parçalarından oluşmaktadır. Bu durumda, birden fazla görüntüden oluşan bir panorama oluşturmanın aksine arka plan modeli kare boyutunda olacaktır. Bu, tüm çıkarma işlemi için gereken hesaplama süresini ve bellek tahsisini azaltmaktadır.

Alt Uzak Segmentasyonu

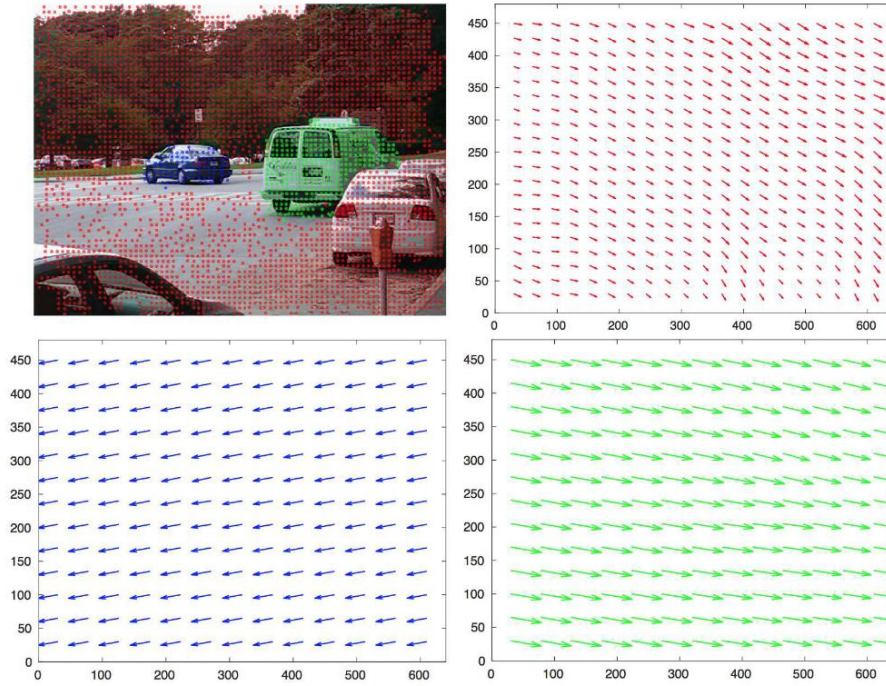
Hareketli nesne algılama yöntemlerinde, arka planı ve ön planı ayırmak için özellik noktalarının yörüngeleri kullanılmaktadır. Önceki yaklaşımların aksine, arka plan çıkarma tekniği uygulamak için görüntüler arasında herhangi bir kayıt yoktur. Özellik noktaları, yörüngelerinin analizine göre etiketlenir ve etiket bilgisi, piksel bazında bir segmentasyon elde etmek için görüntünün tamamına yayılmıştır. Gömülü alt uzaydaki yörüngeler daha sonra kümelenir ve birden fazla ipucunu birleştiren bir fonksiyon ile ön plan veya arka plan olarak etiketlenmektedir (Şekil 2.13).



Şekil 2.13. Yörüngelerin bir alt uzayda (sağda) ve görüntüdeki sonuçta (solda) kümelenmesine bir örnek (Elqursh ve Elgammal, 2012)

Hareket Segmentasyonu

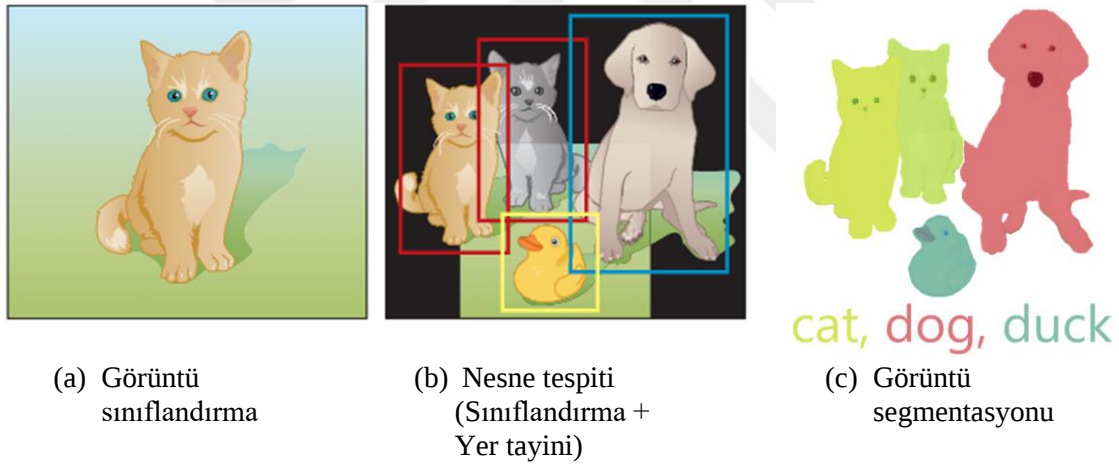
Hareket segmentasyonu yöntemlerinde, videonun her karesini statik veya hareketli olarak bölümlere ayırmak için özellik noktalarının yörüngelerini kullanılmaktadır ancak bir alt uzay kullanılmaz (Şekil 2.14). Bu yöntemlerde, görüntüyü görünen hareketlere göre bölümlere ayırmak amaçlanmaktadır. Hareket segmentasyonu geniş bir literatüre sahip olup sahnenin 2 boyutlu hareketini veya sahnede görüntülenen nesnelerin 3 boyutlu hareketini karakterize etmek için kullanılabilir. Bu kapsamda yapılan çalışmalar, bir arka plan/ön plan etiketlemesi önererek videonun her karesini 2 boyutlu hareketlerle segmentlere ayırmanın ötesine geçmektedir (Chapel ve Bouwmans, 2020).



Şekil 2.14. Sol üstteki görüntüde örnek hareket segmentasyonu. Diğer üç görüntü, görüntüde gözlemlenen üç hareketin optik akışını temsil eder (Zhu ve Elgammal, 2017)

2.1.2. Nesne tespiti

Nesne tanıma bugünlerde hemen hemen her yerde kullanıma sahiptir. Nesne takibi, yaya tespiti, sürücüsüz otomobil ve yüz algılama gibi alanlarda kullanılan nesne tespiti uygulamalarına sayısız örnek verilebilir (Remanan, 2019). Görüntü sınıflandırma bir görüntüye etiket atamayı içerirken nesnenin yerini tayin etmek görüntüdeki bir veya daha fazla nesnenin etrafını sınırlayıcı kutu ile çizmeyi içermektedir. Nesne algılama bu ikisini birleştirir ve daha zorlu bir görevdir (Brownlee, 2019). Görüntü segmentasyonu ise, piksellerin anlamsal etiketlerle sınıflandırılması (semantik segmentasyon), nesnelerin bölümlenmesi (örnek segmentasyonu) veya her ikisinin (panoptik segmentasyon) problemi olarak formüle edilmektedir. Şekil 2.15.'te görüntü sınıflandırma, nesne tespiti ve görüntü segmentasyonu arasındaki fark gösterilmektedir.

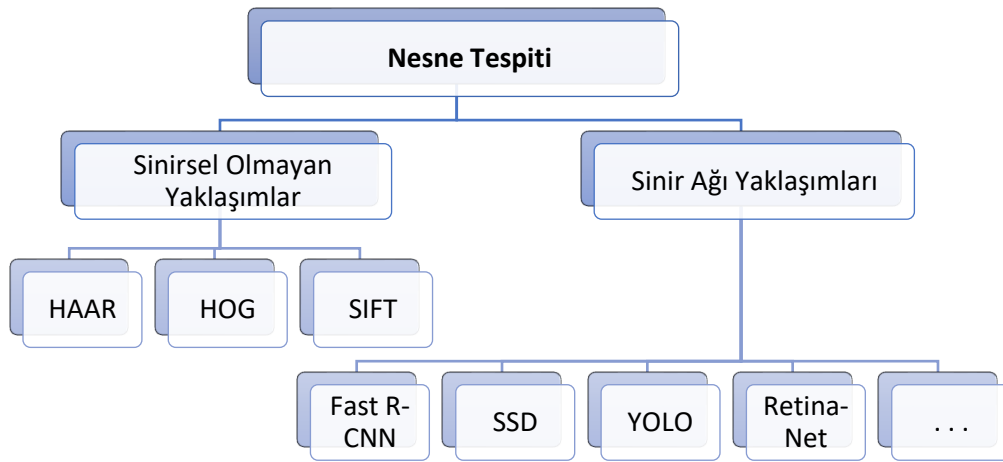


Şekil 2.15. Nesne sınıflandırma, nesne tespiti (Elgendy, 2020) ve görüntü segmentasyonu

Görüntü segmentasyonunda aynı sınıfla ilişkili tüm pikseller bir segmentasyon maskesi oluşturmak için birlikte gruplandırılmaktadır. Anlamsal bölümleme, tüm görüntü pikselleri için bir dizi nesne kategorisiyle (örneğin insan, araba, ağaç, gökyüzü) piksel düzeyinde etiketleme gerçekleştirir; dolayısıyla görüntünün tamamı için tek bir etiket öngören tam görüntü sınıflandırmasından genellikle daha zorlu bir girişimdir. Örnek bölümleme, görüntüdeki ilgi duyulan her nesneyi (örneğin kişiler) tespit edip

tanımlayarak anlamsal bölümlenmenin kapsamını genişletmektedir. Literatürde eşikleme (Otsu, 1979) histogram tabanlı paketleme, bölge büyütme (Nock ve Nielsen, 2004), K-ortalama kümeleme (Dhanachandra ve ark., 2015), havza algoritması (watersheds) (Najman ve Schmitt, 1994) gibi en eski yöntemlerden aktif konturlar (Kass ve ark., 1988) grafik kesmeler (graph cuts) (Boykov ve ark., 2001) ve seyreklik tabanlı (Minaee ve Wang, 2019) yöntemlere kadar çok sayıda görüntü bölütleme algoritması geliştirilmiştir. Ancak son yıllarda, derin öğrenme modelleri, dikkat çekici performans iyileştirmelerine sahip yeni nesil görüntü bölümlendirme modelleri ortaya çıkarmış ve çoğu zaman popüler kıyaslamalarda en yüksek doğruluk oranlarına ulaşmıştır (Minaee ve ark., 2021).

Nesne tespit yöntemleri Şekil 2.16.'da gösterildiği üzere; sinir ağı ve sinirsel olmayan yaklaşımlar olarak ikiye ayrılmaktadır. Sinirsel olmayan yaklaşımlara bir makine öğrenmesi yaklaşımı olan Haar-like (Arreola ve ark., 2019), Scale Invariant Feature Transform (SIFT) ve Histogram of Oriented Gradients (HOG) algoritmaları verilebilir. Bu algoritmalarda genellikle görüntü üzerindeki nesnelerin ayırt edici özellikleri tespit edilerek nesne tespitinin yapılmasına çalışılmaktadır.

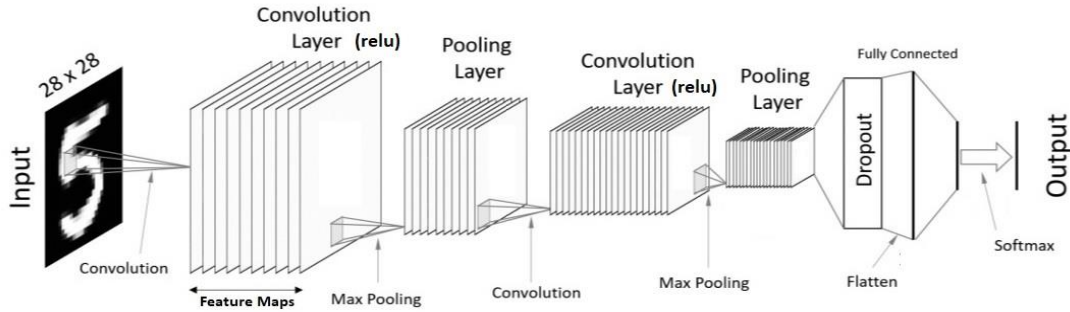


Şekil 2.16. Nesne tespiti yaklaşımları

Nesne tespiti için diğ er bir yaklařım olan sinir ağı yaklařımları tipik olarak evriřimsel sinir ağına dayanmaktadır. Fast R-CNN (Girshick, 2015), Single Shot MultiBox Detector (SSD) (Liu ve ark., 2016), You Only Look Once (YOLO) (Redmon ve ark., 2016) ve Retina-Net (Lin ve ark., 2017) algoritmaları bu grupta yer almaktadır.

Convolutional neural networks (CNN)

CNN  ok katmanlı algılayıcıların (Multi Layer Perceptron- MLP) bir t r d r. CNN hayvanların g rme merkezinden esinlenilerek ortaya  ıkmıř bir mimaridir. Buradaki konvol syon iřlemi, bir n ronun kendi uyarı alanından uyarılara verdiğı cevap olarak d ř n lebilir. CNN yapısı řekil 2.17’de bir veya daha fazla konvol syonel katman, alt  rnekleme (subsampling) katmanı ve bunun ardından standart  ok katmanlı bir sinir ağı gibi bir veya daha fazla, tamamen bağı katmandan oluřmaktadır (LeCun ve ark., 2015).



řekil 2.17. CNN yapısı

B lge bazlı nesne tespit algoritmaları  nce nesne bulunması muhtemel alanları belirleyip ardından oralarda ayrı ayrı CNN sınıflandırıcıları y r tmektedir. Bu y ntem en iyi sonucu vermesine karřın bir resmi iki ayrı iřleme tabi tuttuğı i in resim  zerindeki iřlem sayısını artırmakta ve d ř k bir fps (frames per second) alınmasına sebep olmaktadır.

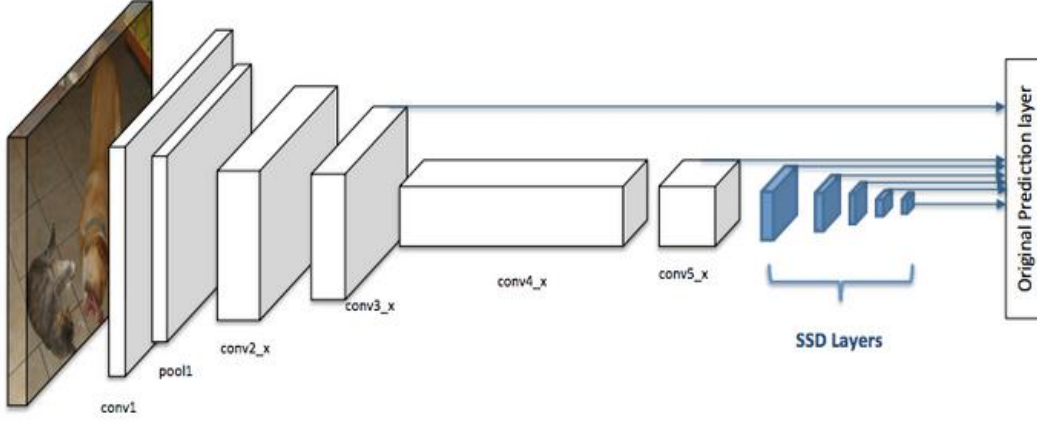
Fast R-CNN

Region Based Convolutional Networks (R-CNN), görüntüdeki nesneye ait bölge önerilerini elde ettikten sonra bölge önerilerine konvolüsyonel sinir ağı uygulayarak nesne tespiti yapmaktadır (Girshick vd., 2016). Sonrasında çıkarılan bölge önerileri Support Vector Machine (SVM) kullanılarak sınıflandırılmaktadır.

Fast R-CNN ise gerçek zamanlı nesne tespitinde daha yavaş olan R-CNN modelinin hız sorununu çözmek amacıyla geliştirilmiştir (Girshick, 2015). Fast R-CNN'nin R-CNN'den farkı öncelikle bölge tavsiyelerine göre görüntüyü ayırmadan görüntünün tamamını ve bölge tavsiyelerini CNN mimarisinde girdi olarak tek bir ileri yaymada almasıdır.

Single shot multibox detector (SSD)

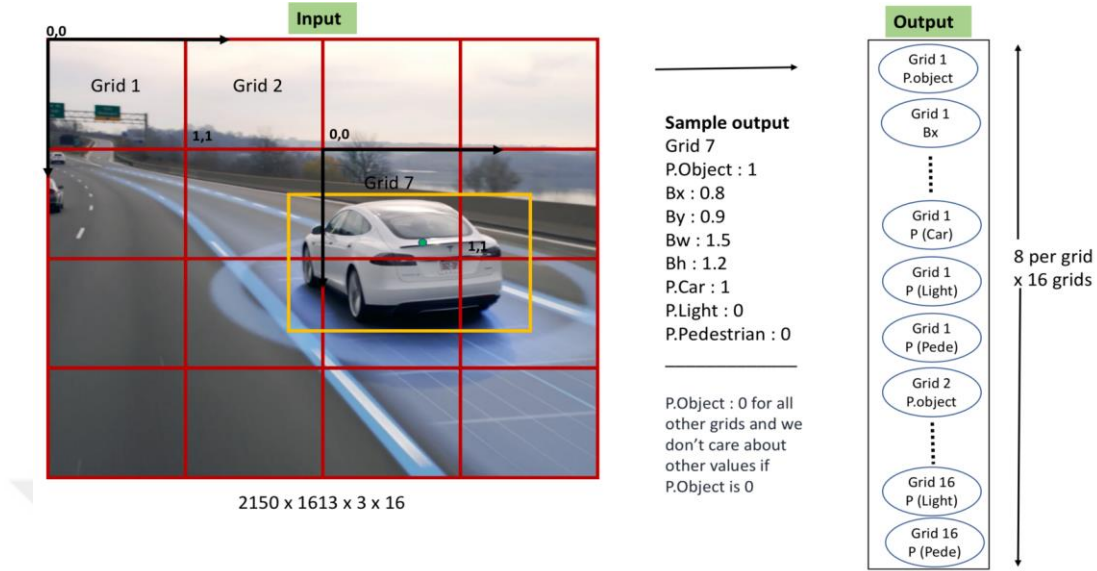
CNN ile bölge önerisini iki farklı aşamada yapılmaktadır. İlk önce resimde nesne olması beklenen bölgeler belirlenmektedir. Daha sonra tam bağlı katman ile sınıflandırma yapılmaktadır. Şekil 2.18'de görüldüğü gibi SSD bu ikisini tek seferde gerçekleştirmektedir. SSD yaklaşımı, sabit boyutlu bir sınırlayıcı kutu koleksiyonu üreten ve bu kutulardaki nesne sınıfı örneklerinin varlığına yönelik puanları üreten, ardından son tespitleri üretmek için maksimum olmayan bir bastırma (non-maximum suppression - NMS) adımının takip ettiği ileri beslemeli evrişimli bir ağa dayanmaktadır (Liu ve ark., 2016). Ayrıca, SSD sadece tek ileri gidiş yaptığı için düşük donanımlı cihazlarda bile hızlı bir şekilde nesne tanıma yapabilmektedir. Ancak SSD ile çok isabetli modeller oluşturulamamaktadır. Bu nedenle, sadece hız önemli ise ve gerçek zamanlı nesne tespiti ve tanıma işlemi gerçekleştirilecekse SSD tercih edilmelidir.



Şekil 2.18. SSD algoritmasının şeması

You only look once (YOLO)

YOLO algoritması resmi tek bir seferde nöral ağdan geçirerek resimdeki tüm nesnelerin sınıfını ve koordinatlarını tahmin edebilmektedir. Görüntüyü tek bir seferde nöral ağdan geçirerek tahminlemede bulunması YOLO'nun daha hızlı olmasını sağlamaktadır. Bu tahmin işleminin temeli, nesne tespiti tek bir regresyon problemi olarak ele almasında yatmaktadır. Bu yöntemde resim ızgara halinde bölümlere ayrılarak her bir ızgara kendi alanı içinde nesnenin olup olmadığını, varsa orta noktasının içinde olup olmadığını, orta noktasının da içindeyse uzunluğunu, yüksekliğini ve hangi sınıftan olduğunu bulmakla sorumludur (Redmon ve ark., 2016). YOLO algoritmasındaki ızgara yapısı Şekil 2.19'da gösterilmektedir.

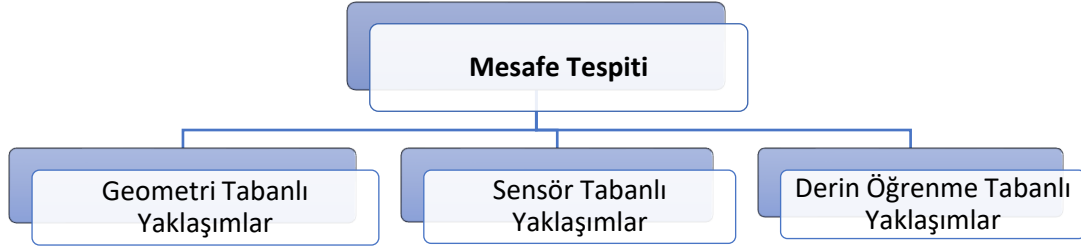


Şekil 2.19. YOLO algoritması

2.1.3. Mesafe tespiti

Bir kamera görüntüsünde mesafe ölçmek zorlu bir görev olmaya devam etmektedir. Ancak otonom araç sürüşü elde etmek, 3B sahne rekonstrüksiyonu ve artırılmış gerçeklik (Augmented Reality - AR) gibi görevleri yerine getirmekte faydalanan uygulamalar arasındandır. Mesafe tespiti, robotikte; derinlik algılama, gezinme ve planlama gibi birçok görev için ön koşuldur (Tan, 2020).

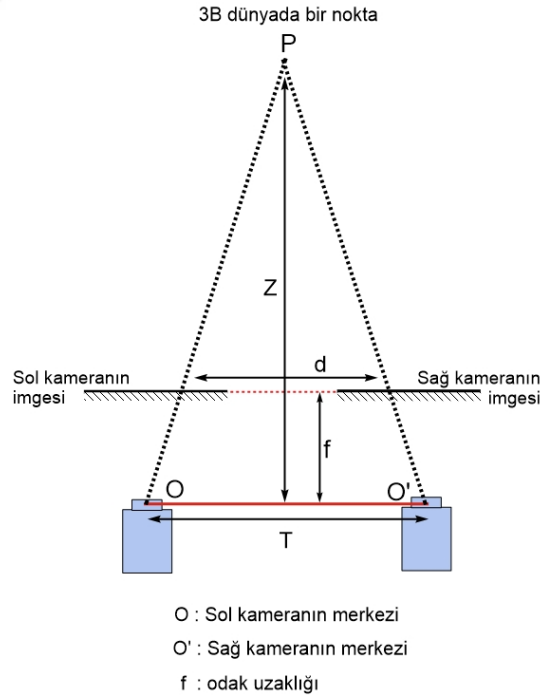
İnsan sağ ve sol gözünden elde ettiği görüntüyü karşılaştırarak derinliği tahmin etmektedir. Her bir görüntüdeki küçük bir değişiklik derinlik hesabı için yeterli olmaktadır. İnsan gözünde olduğu gibi derinlik hesabının yapılması amacıyla genel olarak birkaç farklı yöntem benimsenmiştir. Uzaklık tespitinde kullanılan yaklaşımlar Şekil 2.20’de yer alan üç alt başlıkta incelenebilir.



Şekil 2.20. Uzaklık Tespiti Yaklaşımları (Zhao ve ark., 2020)

Geometri Tabanlı Yaklaşım

Geometri tabanlı derinlik tespitinde en çok bilinen yöntem “Stereo Kamera” kullanılarak elde edilen görüntü çiftinin eşleştirilmesine dayanmaktadır (Zhao ve ark., 2020). Üçgenleme yapılarak x ve y koordinatları bilinen nesnenin z koordinatındaki konumu tespit edilebilmektedir (Akarun, 2020). Stereo kamera geometrisi ile üçgenleme şeması Şekil 2.21’de gösterilmektedir.



Şekil 2.21. Üçgenleme (Akarun, 2020)

Bu yöntemde geometrik özelliklerden faydalanılarak derinlik bilgisi – z eksen koordinatı – Eşitlik 5’teki formül ile hesaplanabilmektedir.

$$Z = \frac{f(T)}{d} \quad (5)$$

Üçgenlemede iki kameradan elde edilen görüntünün eşleştirilmesi bu yöntemde karşılaşılan önemli bir görevdir. Aynı piksel değerlerine sahip bir çift görüntüde derinlik hesabı için görüntü eşleştirme oldukça zordur. Verimli bir derinlik hesabı görüntü çiftlerine bağlıdır (Zhao ve ark., 2020).

Üçgenleme yöntemi ile mesafe tespitinde kamera çözünürlüklerinin yüksek olması ölçümün doğruluğu için önemlidir (Hantao, 2005; Yanık ve ark., 2021).

Sensör Tabanlı Yaklaşım

Sensör tabanlı yaklaşımın genel çalışma prensibinde ortamdaki nesnelere lazer ışınları gönderilerek geri yansımaları arasındaki geçen süreyi hesaplama yatmaktadır. RGB-D (Red Green Blue-Depth) ve LIDAR (Light Detection and Ranging) gibi kameralar kullanılan bu yöntemde derinlik bilgisine doğrudan ulaşılabilir. Ancak sınırlı ölçüm aralığı ve güneş ışığına olan hassasiyet bu yöntemin dezavantajıdır. Buna rağmen insansız sürüş endüstrisinde LIDAR kameraların kullanımı sıkça tercih edilmektedir (Zhao ve ark., 2020).

Derin Öğrenme Tabanlı Yaklaşım

Derin öğrenmedeki hızlı gelişmeler görüntü sınıflandırma, nesne tespiti, anlamsal bölütleme gibi görüntü işleme alanında oldukça başarılı performans göstermiştir. Bunun yanında piksel düzeyinde derinlik haritasının derin öğrenme ile bir görüntüden elde edilebileceği anlaşılmıştır. CNN, RNN (Recurrent Neural Network), GAN

(Generative Discriminative Network) gibi derin öğrenme ağları ile görüntüden derinlik tahmini yapılabilmektedir (Zhao ve ark., 2020).

2.1.4. Nesne takibi

Nesne takibi bir videodaki herhangi bir nesne veya objenin benzersiz özelliklerinin görüntü dizisinin sonraki kısımlarında takibi anlamına gelmektedir. Nesne takibi yöntemleri nokta tabanlı, çekirdek tabanlı ve silüet tabanlı olarak 3 ana başlık altında toplanmıştır. Nokta tabanlı takip yönteminde nesneler noktalar ile ifade edilir ve arka arkaya gelen görüntü dizisinde bu noktaların birbirlerine paralel olmaları beklenmektedir. Kareler arasında bu noktaların benzerlikleri hesaplanarak nesne takibi yapılmaktadır. Bu yöntemde kalman filtresinin kullanımı sıkça tercih edilmektedir. Çekirdek tabanlı yöntemlerde ise; bir geometrik şekil (kare, dikdörtgen vs.) içine alınan nesneden çıkarılan özellikler ardı sıra gelen karelerde aranarak nesne takibi yapılmaktadır. Bu geometrik şekil içinden elde edilen piksellere ait yoğunluk, histogram gibi bilgiler takip edilmektedir. Silüet tabanlı yöntemler; insan gibi belirli bir geometrik şekle sahip olmayan nesnelerin takibinin yapılmasında kullanılmaktadır. Buradaki temel amaç kenar bilgisi veya şekil bilgisi çıkarılarak sonraki karelerde bu özelliklerin takibinin yapılmasıdır. Bu yöntem şekil değişikliklerine karşı çok hassastır (Hanbay ve Üzen, 2017).

2.2. Literatür Özetleri

Tez kapsamında literatür incelendiğinde farklı çalışma alanlarında hareketli kameradan hareket tespiti, nesne tespiti, mesafe tahmini ve şüpheli hareket tespitine yönelik çalışmalar incelenmiştir. Genel amaçlı nesne tespitinde geçmiş yıllarda CNN'e dayalı yaklaşımlar kullanılarak en gelişmiş sonuçlar elde edilmiştir. En iyi bilinen yaklaşımlar arasında bölge önerisine dayalı R-CNN ve YOLO model aileleri yer aldığı görülmüştür.

Liu ve ark. (2015) tarafından müşterilerin baş ve vücut yörüngesini değerlendirerek ürünlere karşı ilgili veya ilgisiz olduklarını tespit etmeye yönelik bir sistem önerilmiştir. Yarı denetimli öğrenme kullanılarak müşterilerin davranışlarının sınıflandırıldığı bu çalışmada davranış tespit başarımı %89.5 olarak gerçekleşmiştir.

Bilgisayarlı görü alanında videoların analizi oldukça zorlu bir görevdir. Aitfares ve ark. (2016), yürüttükleri çalışmada hareketli insanlara ait şüpheli aktiviteleri gerçek zamanlı olarak tespit etmeye çalışmıştır. Görüntüdeki ani yön değişimleri izlenmiş ve görüntü segmentlere ayrılarak şüpheli objenin takibi gerçekleştirilmiştir. Ardışık kareler (frame) içindeki nesnelerin tanımlanması için SIFT algoritması kullanılmıştır.

Kalabalık yönetimi ve kalabalığın anormal davranışlarının tespiti bilgisayarlı görüde önemli bir yere sahiptir. Anormal kalabalık davranışının tespiti, insanların hayatlarını kurtarmakta doğrudan yardımcı olabilmektedir. Lahiri ve ark. (2018), yürüttükleri çalışmada hareket geçmişi görüntüsü ve optik akış teknikleri kullanılarak kalabalığın anormal davranışı tespit etmeye çalışmıştır. Bu kapsamda düşük maliyetli ve pratikte uygulanabilir bir sistem tasarlanmış olup sistemin başarısı %87.5 olarak gerçekleşmiştir.

İnsansız hava araçları (İHA), hareket kabiliyeti ve operasyonel basitliğinden yararlanılarak çeşitli arama kurtarma ve gözetleme uygulamalarında kullanılmaktadır. Bazı durumlarda, bir İHA'nın bir insan deneğin eylemlerini tanıma ve ardından tepki veren eylemlerde bulunma yeteneği arzu edilmektedir. Perera ve ark. (2018) tarafından yürütülen çalışmada İHA görüntülerinden insan, insana ait duruş (postür) ve yörüngenin tahmin edilmesine odaklanılmıştır. Bölge Tabanlı Evrişimli Sinir Ağı (R-CNN) kullanılarak insan algılama modeli eğitilmiştir. Her video karesi, yansıtımlı dönüşüm kullanılarak perspektif için düzeltilmiştir. Silüetlerin Yönlendirilmiş Gradyanların Histogramları (HOG) özellik olarak kullanılarak ve tespit edilen insan figürleri daha sonra duruşlarına göre sınıflandırılarak ileri doğru yürüyüşü ve dönüş yürüyüşünün sırasını tahmin etmek için dinamik bir sınıflandırıcı geliştirilmiştir.

Tahmin edilen duruşlar, insan deneğin kat ettiği yörüngeyi tahmin etmek için kullanılmıştır. Dedektörde ortalama %98 hassasiyet elde edilmiştir. Havadan çekilen videolar üzerinde gerçekleştirilen deneyler, çözümün farklı türde perspektifi bozulan videolar için doğru duruş ve yörünge tahminleri elde edebildiğini doğrulamaktadır. Örneğin, yerden 40 m yükseklikte kaydedilen bir video için perspektif düzeltmesi, duruş ve bakış açısı tahmininde doğruluğu sırasıyla %37.1 ve %17.8 oranlarında artırmaktadır. “Drone” ile yerdeki insan arasındaki açı olan ϕ , 0° ’dan büyük olduğunda dikey perspektif bozulma meydana gelir ve ϕ büyüdükçe bu bozulma kötüleşmektedir. $\phi=90^\circ$ olduğunda bozulma düzeltilememektedir. $60^\circ \leq \phi \leq 90^\circ$ ise perspektif bozulmayı düzeltmek oldukça zor olmaktadır. Bu çalışmada ϕ açısı en fazla 60° olarak belirlenmiştir.

Video gözetimi ve otomatik şiddet tespiti için yeni teknolojilerin geliştirilmesi günlük yaşantımıza daha fazla güvenlik anlayışını getirmektedir. Daha önce yapılan çalışmalarda en son teknolojiyle yayınlanmış çözümler, film sahneleri, spor karşılaşmaları veya kalabalıklardaki şiddeti tespit etmeye yönelik teknikler sunulmaktadır. Soares ve ark. (2019) tarafından yapılan çalışmada, kameradan elde edilen görüntüler üzerinden insan vücudundaki eklemlere ait 17 referans noktası belirlenmiştir. Bu referans noktalarının ikili kombinasyonu hesaplanarak bir vektör elde edilmektedir. Hesaplanan vektörler SVM ile sınıflandırılmıştır. Sonuçlar, yöntemin elleri yukarıda ve yatar durumdaki bireyleri %85'e varan doğruluk oranlarıyla tanımlayabildiğini göstermiştir.

Phyo ve ark. (2019) tarafından, UTKinect Action-3D ve CAD-60 datasetleri üzerinde geliştirdikleri sistem ile insan hareketleri tanımlanmaya çalışılmıştır. Düşük hesaplama maliyeti ve yüksek doğruluk elde edilen bu sistem ile yalnız yaşayan yaşlı insanların günlük aktiviteleri takip edilebilmiş, kuma ve baş ağrısı gibi durumlar tespit edilebilmiştir. DCNN ile geliştirilen bu sistem ile %96.15 oranında doğruluk değerine ulaşılmıştır.

Yang ve ark. (2019) tarafından gerçekleştirilen çalışmada, dinamik arka plan altında daha iyi sağlamlığa ve bölünebilirliğe sahip yeni bir hareketli nesne algılama algoritması sunulmuştur. Dinamik arka plan altında hareketli nesne tespitinde arka plan telafisinin doğruluk gereksinimini azaltmak için kare farkı arama yöntemi önerilmiştir. Spesifik olarak, seçici arama algoritması fikrini hareketli hedef tespit algoritmasına dahil ederek gürültü önleme yeteneğini geliştirmek üzere görüntü piksel düzeyindeki özellikleri dahil etmek için kare farkı yöntemi kullanılmıştır. Ek olarak, seçici arama algoritmasının buluşsal arama avantajından belirli bir hareketli hedefi bölümlere ayırmak için yararlanılmış ve her bir hareketli hedefin tespit kareleri sırasıyla elde edilmiştir. Oluşturulan İHA kampüs veri setinde kare farkı arama yöntemi kullanılarak ortalama %89.2 doğruluk elde edilmiştir.

Derin öğrenme, hareket tespitinin yanında görüntülerden şüpheli veya şiddet içeren hareketleri tespit etmek için de kullanılmaktadır. İnsan davranışları oldukça karmaşık ve doğal bir ortamda fazlaca çeşitlilik gösterebilmektedir. Bordoloi ve ark. (2020), şüpheli hareket içeren videolar üzerinde yaptıkları çalışma ile hırsızlık, kapı kilidi kırma vb. gibi aktiviteleri %95 gibi yüksek bir başarı oranı ile tespit edebilmiştir. Bu çalışmada YOLOv3 kullanılmış olup Faster R-CNN'e göre daha iyi performans sergilediği anlaşılmıştır.

Karantinanın temel amacı, patojenin kuluçka dönemi boyunca sağlık durumlarının gelişimini izlemek amacıyla hastalığa maruz kalma ihtimali olan kişileri ayırmaktır. Dobrea ve Dobrea (2020)'nin yaptığı çalışmanın amacı, karantinayı uygulamak ve COVID-19 gibi bulaşıcı bir hastalığın semptomatik veya asemptomatik taşıyıcısı olarak tanımlanan kişilerin karantinada tutulmasını sağlayan bir sistem geliştirmektir. Bu çalışmada, iki sınıflandırma yaklaşımı arasında, yani klasik bir makine öğrenimi yaklaşımı olarak Destek Vektör Makinesi (SVM) sinir ağı ile derin öğrenme sınıflandırma sistemi arasında karşılaştırmalı bir çalışma yapılmıştır. Bilgisayarlı görmede son teknoloji sistemler; HoverGames quadcopter'ına yerleştirilen Raspberry Pi ve Jetson Nano adlı iki gömülü yardımcı bilgisayar, insanları tanımlamak için

geliştirilen ve uygulanan sınıflandırma sistemlerini desteklemektedir. Bu araştırmanın nihai amacı, önceden programlanmış bir uçuş rotasını takip ederek eş zamanlı olarak insanları tespit etmek ve sistem operatörünü karantina bölgelerini takviye etmesi konusunda uyarma yeteneklerine sahip bir quadcopter (dört pervaneli drone) sistemi geliştirmektir. Elde edilen sonuçlar, derin öğrenme yaklaşımının sağladığı üstün performansları ortaya koymuştur. Klasik yaklaşıma göre altı kattan daha hızlı ve doğrudan video verisi akışında %90'ın üzerinde doğru sınıflandırma performansı bulunduğu gözlemlenmiştir.

Jain ve ark. (2020) CNN tabanlı SSD ve Faster RCNN kullanarak güvenlik uygulamaları için silah tespiti yapmak amacıyla çalışma yürütmüştür. Bu kapsamda SSD ve Faster RCNN algoritmaları, silah tespiti için önceden etiketlenmiş ve oluşturulmuş görüntü veri seti için simüle edilmiştir. Her iki algoritma da etkili ve iyi sonuçlar vermektedir. Ancak gerçek zamanlı uygulamaların hız ve doğruluk arasındaki dengeye dayandığı gözlemlenmiştir. Çalışma kapsamında SSD algoritmasının, 0.736 s/kare ile daha hızlı olduğu, Faster RCNN'nin ise 1.606 s/kare hız değerine sahip olduğu anlaşılmıştır. Doğruluk açısından Faster RCNN ile %84.6 oranında daha iyi doğruluk değeri elde edilmişken SSD'nin %73.8'lik bir doğruluk değerine sahip olduğu görülmüştür. Gerçek zamanlı uygulamalarda SSD daha iyi bir hız değeri sağlarken Faster RCNN çalışma kapsamında daha iyi bir doğruluk değeri sunmuştur.

Zhu ve ark. (2020) çalışmalarında dinamik bir arka plan varlığında hareketli nesnelerin tespiti için simetrik bir yöntem önermektedir. İlk olarak, önerilen hareket bölgesini tespit etmek için bir arka plan telafi yöntemi kullanılmıştır. Daha sonra, hareketli nesnelerin konumunu doğru bir şekilde belirlemek amacıyla, görüntüdeki tüm nesneleri tespit etmek için YOLOv3-SOD adı verilen, hafif ve küçük nesneler için özel olarak tasarlanmış evrişimli sinir ağı tabanlı bir yöntem önerilmektedir. Son olarak hareket algılama ve nesne algılama ile elde edilen sonuçların birleştirilmesiyle hareketli nesneler belirlenmektedir. Kaçırılan tespitler, bitişik karelerdeki zamansal ve mekansal bilgilere göre geri çağırılmaktadır. Özellikle hareketli nesne tespiti ve tanıma için bir

veri seti eksikliğinden dolayı 105 video içeren üç sınıftan oluşan MDR105 veri seti yayınlanmıştır. Yapılan deneylerde, önerilen algoritmanın çeşitli senaryolarda hareketli nesneleri tespitte %91mAP değerine sahip olduğu bildirilmiştir.

Chriki ve ark. (2020) çalışmalarında, videoların bir mobil kamera tarafından alındığı İHA tabanlı gözetleme görevine yardımcı olmak için yeni anormallik tespit teknikleri önermiştir. İHA videolarından sağlam özellikler çıkarmak için, önceden eğitilmiş CNN ve iki popüler el yapımı yöntem HOG ve HOG3D olmak üzere üç farklı özellik çıkarma yöntemi kullanılmıştır. Daha sonra denetimsiz sınıflandırma için OCSVM (One Class Support Vector Machine) uygulanmıştır. Bir otoparkı izleyen İHA tarafından çekilen videoları içeren bir veri seti üzerinde gerçekleştirilen kapsamlı deneyler, önerilen tekniklerin verimliliğini kanıtlamaktadır. Spesifik olarak, AUC (Area Under The Curve) değerlendirme metriği ile elde edilen niceliksel sonuçlar, önerilen yöntemin, 0.93'lük AUC değeri ile iyi sonuçlar elde ettiğini göstermektedir.

Tsiktsiris ve ark. (2020) tarafından, çeşitli adi suç türlerinin zamanında, doğru, sağlam ve otomatik tespiti için derin öğrenme tekniklerine dayalı, çevrimiçi, uçtan uca bir çözüm önerilmiştir. Önerilen sistem, vandalizm ve kazalar gibi anormal yolcu davranışlarını tespit edebilmekle beraber saldırganlık, çanta kapma ve vandalizm gibi küçük suçları tespit ederek yolcu güvenliğini de artırmaktadır. Çalışma kapsamında LSTM'yi içeren yeni bir model tasarlanmış ve %99 doğruluk değeri ile anormal durumlar tespit edilebilmiştir.

Küçük hareketli nesnelerin tespiti, uçan böceklerin izlenmesi, yiyecek arama davranışlarının incelenmesi, mahsullerin çiçeklenmesini ve tozlaşmasını izlemek için böcek tozlayıcıların kullanılması, bal arısı kolonilerinin gözetimi ve bal arılarının hareketlerinin izlenmesi gibi uygulamalar önemli bir araştırma alanıdır. Bununla birlikte, küçük nesneler üzerinde ayırt edici şekil ve dokusal detayların bulunmaması nedeniyle, evrişimli sinir ağlarına dayalı modern nesne algılama yöntemlerinin doğrudan uygulanması, önemli ölçüde daha düşük performans göstermektedir. Stojnic

ve ark. (2021) “*drone*” üzerinden bal arılarının tespitine yönelik bir araştırma gerçekleştirmiştir. Bu çalışmada, standart video kameralarla donatılmış insansız hava araçları kullanılarak kaydedilen videolardaki küçük hareketli nesnelerin tespiti için bir yöntem önerilmiştir. Önerilen yöntemin ana adımları video sabitleme, arka plan tahmini ve çıkarma, CNN kullanarak kare bölümlendirme ve bölümlenmiş kareyi eşiklemedir. Ancak bir CNN'nin eğitimi için geniş bir etiketli veri setinin mevcut olması gerekmektedir. Videolardaki küçük hareketli nesnelerin manuel olarak etiketlenmesi çok zor ve zaman alıcıdır. Çalışmanın gerçekleştirildiği dönemde bu tür etiketlenmiş veri kümelerinin mevcut olmamasından dolayı bu sorunu aşmak için, gerçek dünya arka planlarına sahip video dizilerine küçük damla benzeri nesnelerin eklenmesiyle oluşturulan sentetik videoları kullanarak bir CNN'nin eğitilmesi önerilmiştir. Uçan bal arılarının tespitine ilişkin deneysel sonuçlar, klasik bilgisayarlı görme teknikleri ve CNN'lerin yanı sıra sentetik eğitim setlerinin bir kombinasyonunu kullanarak, önerilen yaklaşımın, CNN'lerin verilen probleme doğrudan uygulanmasıyla ilişkili sorunların üstesinden geldiğini ve gerçek dünya videolarında yapılan testlerinde ortalama F1 skorunun 0.86 olduğu görülmüştür.

Elharrouss ve ark. (2021) tarafından gerçekleştirilen çalışmada temel amacı insanların hayatlarını basitleştirmek, otomatik olarak izlenmesini ve daha güvenli hale getirilmesini sağlamak olan güvenlik ve izleme sistemlerinin genel bir incelemesi yapılmıştır. Video gözetleme sistemlerinin yüz tespiti, aksiyon tespiti, kalabalıkta insan sayısı tespiti, insan kimlik tanıma, nesne tespiti ve takibi, anormal olay tespiti ve video özetleme kapsamında kullanıldığı görülmüştür. Çalışmada çevresel değişikliklerin her sistem için bir sınırlama teşkil ettiği ve bunların video gözetim sistemlerinin zorluklarını ortaya çıkardığı belirtilmiştir. Araştırmada, özellikle güvenliği sağlamak için birçok video gözetim görevi üzerinde çalışmalar yapıldığı ve bu görevler arasında nesne tanıma, nesne sınıflandırma, hareketli nesne algılama ve izleme görevlerinin yer aldığı gözlemlenmiştir. Önerilen birçok yaklaşım ve algoritma, birçok sorunu çözme; kabul edilebilir ve etkili sonuçlar sağlamayı başardığı ancak bazı konularda iyileştirmeler yapılmasının gerekliliği ifade edilmiştir. Çalışmaya göre video gözetim

sistemlerinin zorlukları arasında tüm sistemlerde ortak olan iki nokta vardır ve bunlar; yanlış alarm ve ani ortam değişiklikleridir. Bir video gözetim sisteminde yanlış alarm verilmesi istenmeyen bir durumdur ve ortam değişikliği, özellikle beklenmedik hava değişiklikleri olmak üzere sistemde bir arızaya neden olabilmektedir. Daha önce bahsedilen sınırlamaların üstesinden gelmek için, video gözetim sistemlerinin bazı algoritmalarının sağlamlıkları ve doğrulukları artırılarak iyileştirilmesi gerekmektedir.

Liu ve Szirányi (2021), gerçek zamanlı İHA insan tespiti, vücut ve el kurtarma hareketlerinin tanınması üzerine çalışma yürütmüştür. İnsan tespitine yönelik YOLO3-tiny gibi biyometrik iletişimler kurmak için gövde özellikli çözümler kullanılmıştır. Bir kişinin varlığı tespit edildiğinde, kullanıcı ve kullanıcının hareketinin tanınması aşamasına geçecek şekilde kurgulanan çalışmada; “*drone*”, sesli iletişimin dezavantajlarından kaçınarak kısa ve etkili bir şekilde iletişim kurabilmektedir. On vücut kurtarma hareketinden (tekme, yumruk, çömelme, ayağa kalkma, dikkat, iptal, yürüme, oturma, yönlendirme ve telefon arama) oluşan bir veri seti, İHA'nın yerleşik kamerası tarafından oluşturulmuştur. En önemli iki hareket, sırasıyla ayarlama ve sıfırlama işlevlerini temsil eden *dikkat* (iki elin havada sallanması) ve *iptal* (tek elin havada sallanması) hareketleridir. İnsan vücudunun kurtarma hareketi dikkat olarak algılandığında, “*drone*” yavaş yavaş kullanıcıya el hareketi tanıma için daha büyük bir çözünürlükle yaklaşacaktır. Geliştirilen sistem ile vücut hareketi veri setindeki verileri test etmekte %99.8, el hareketi veri setindeki verileri test etmekte ise %94.71 doğruluk değeri elde edilmiştir. Gerçek zamanlı İHA kameraları üzerinde yapılan deneyler, çözümün beklenen İHA kurtarma amacına ulaşabileceğini doğrulamaktadır.

Sınav, yetenekleri seçmenin bir yoludur ve mükemmel bir gözetmenlik stratejisi, sınavın adilliğini artırmaktadır. Fang ve ark. (2021) tarafından sınav salonundaki anormal davranışların tespitini otomatik olarak gerçekleştirmek için YOLOv3 algoritmasını temel alan bir yöntem önerilmektedir. Algılama sürecini optimize etmek için kare alternatif çift iş parçacığı yöntemi kullanılmıştır. Araştırma sonuçları, geliştirilmiş YOLOv3 algoritmasının hem tespit doğruluğunu hem de tespit hızını

artırabileceğini göstermektedir. Kare alternatifli çift iş parçacığı yöntemi, algılama hızını büyük ölçüde artırma olanağı sunmaktadır. Geliştirilen YOLOv3 algoritmasının test setindeki mAP değeri %88.53'e ve kare alternatifli çift iş parçacıklı algılama yönteminde algılama hızı saniyede 42 fps değerine ulaşmıştır.

Kamusal ve sosyal alanlarda kullanılan güvenlik kameralarında insanların güvenliğinin sağlanması ve sürdürülmesi için insan operatörlerin yerine yapay zeka algoritmalarının kullanılmasına başlanmıştır. Bu kapsamda Ağdaş ve Gülseçen (2022)'nin yaptığı çalışmada güvenlik kameralarının gözetlediği toplumsal alanlarda silah ve bıçağın; görüntülerden bilgisayarlı görü ile tespit edilmesine yönelik çalışma gerçekleştirilmiştir. YOLO algoritmalarının YOLOv4, YOLOv5, YOLOR ve YOLOX modelleri kullanılarak başarı oranları kıyaslanmıştır. 3 000 adet silah ve 2 078 adet bıçak görüntülerinden oluşan veri seti üzerinde en başarılı sonuç %97.6 mAP@0.5 değeri ile YOLOR modeli ile elde edilmiştir.

Srivastava ve ark. (2022)'nin yaptığı çalışmada derin öğrenmeye dayalı geliştirilen güncel teknikler kullanılmakta ve şiddet tespiti için LSTM ile derin öğrenmeye dayalı farklı hibrit modeller kullanarak transfer öğrenme kavramını önermektedir. Şiddetin tespiti ve şiddete karışan bireylerin yüzünün tanınmasının, otomatik video gözetim araştırmasının geliştirilmesinde gözle görülür bir etkisi olduğu değerlendirilmiştir. Toplumda artan riskler ve bunları izleyecek personelin yetersiz olması nedeniyle, “drone”lara ve bilgisayarlı video gözetimine yönelik artan bir talep bulunmaktadır. Şiddet tespiti hızlı yapılabilmektedir ve gözetleme videolarını seçici olarak filtrelemek ve anormalliği yaratan kişiyi tespit etmekte kullanılabilmektedir. Kalabalık bir alanda “drone” gözetleme videolarından bireysel kimlik tespiti, hızlı hareket, örtüşen özellikler ve dağınık arka planlar nedeniyle zordur. Amaç, şiddete bulaşan şiddet yanlısı bireyleri, tanıyan ve bir tehlike sinyali göndererek hızlı yardımın sunulabilmesini sağlayan daha iyi bir “drone” gözetleme sistemi sunmaktır. “Drone” ile çekilen görüntülerden şiddet suçuna maruz kalan kişilerin belirlenmesi, insan yüzünün tanınması önemli sorunları içermektedir. Dolayısıyla bu çalışmada, görüntü

işleme teknikleriyle birleştirilmiş bir CNN modeli kullanılmıştır. Test amacıyla, “*drone*” ile elde edilen video veri seti, kısıtlamasız bir ortam için geliştirilmiştir. Sonuçta, LSTM mimarisi ile başlangıç modülleri ve artık bloklardan oluşan bir hibritten elde edilen özellikler %97.33'lük bir doğruluk sağlamış ve bu sayede dikkat çekici olduğu ve dolayısıyla test edilen diğer modellere göre üstünlüğünü gösterdiği ortaya çıkmıştır. Bireysel tanımlama modülü için veri setimizde elde edilen %99.20'lik en iyi doğruluk, yüz tanımlama için eğitilmiş artık bloklara sahip bir CNN modelidir.

Sambolek ve Ivasic-Kob (2023) tarafından yapılan çalışmada “*drone*” üzerindeki bir monoküler kameradan, GPS alıcısı ve bir eylemsiz ölçüm birimine (IMU) dayalı olarak bir kişiyi tespit etmek ve coğrafi konumunu belirlemek ve kişinin birden fazla fotoğrafta tespit edilmesi durumunda hareketin yönünü ve hızını belirlemek için bir çerçeve geliştirmiştir. İlk olarak, karmaşık sahnelerdeki küçük nesnelerin tespit edilmesindeki verimliliği ve etkinliği nedeniyle, YOLOv4'e dayalı bir yöntem kullanılmıştır. Daha sonra, bir kişinin GPS koordinatlarını hesaplamak için pasif bir coğrafi konum yöntemi sunulmuştur. Sonuçta sistem, elde edilen verilere dayanarak arazi araması için bir segment önermektedir. Önerilen çerçeve, DJI Phantom 4 Advanced ve DJI Mavic 2 Enterprise Advanced olmak üzere iki tür “*drone*” kullanılarak test edilmiştir. Deneysel sonuçlar, önerilen çerçevenin arama alanının dijital yükseklik modeli ile birlikte kullanılmasıyla, eğimli arazi durumunda bile kişinin tatmin edici bir doğrulukla tespit edilebileceğini ve coğrafi konumunun belirlenebileceğini göstermektedir.

Çalışma kapsamında gerçekleştirilen literatür taramasında güvenlik sistemlerinin sürekli geliştirilmesine ve iyileştirilmesine duyulan ihtiyaç sıkça dile getirilmektedir. Buradan yola çıkılarak tez çalışmasında; hareket tespiti, nesne tespiti ve mesafe tespiti yöntemlerinin bir arada, saldırı tespiti amacıyla kullanıldığı hibrit bir model önerilmiştir. “*Drone*” ile elde edilmiş, güvenlik açısından kritik seviyede önemli olan veya gözetlenmek istenen alanlara ait görüntülerin, görüntü işleme ve derin öğrenme yöntemleri yardımıyla anlamlandırılarak otomatize bir şekilde analiz edilmesi ve olası

bir saldırı durumunu haber verecek bir sistemin tasarlanması amaçlanmıştır. Çalışma kapsamında hareket tespitinde literatürde yer alan klasik yöntemlere ek olarak bir derin öğrenme algoritması olan YOLOv8'in segmentasyon algoritması kullanılmıştır. Diğer yandan nesne tespitinde de YOLOv8 algoritmasına ait tüm varyantlar kullanılmış ve başarı oranları karşılaştırılmıştır.



3. MATERYAL VE YÖNTEM

3.1. Materyal

Derin öğrenmede başarı oranına etki eden en önemli faktör modelin geliştirilmesinde kullanılan veri kümesinin büyüklüğü ve kalitesidir. Veri kümesinin zenginliği arttıkça modelin başarısı da artacaktır. Kapsamlı ve zengin bir veri kümesi oluşturmanın zaman sarfiyatını ve maliyeti artıracığı bilindiğinden daha önceden oluşturulmuş veri setlerinin kullanımı, çalışmalarda sıkça tercih edilmektedir. Bu kapsamda Tablo 3.1’de yer alan VIRAT (Video Image Retrieval and Analysis Too) video dataset (Oh ve ark., 2011), UCF (University of Central Florida) aerial action dataset (Anonim, 2009) isimli veri setlerinin yanı sıra DJI Phantom 3 PRO 4K Drone ile elde edilen görüntü örnekleri kullanılmıştır.

Tablo 3.1. Kullanılan veri setleri

Veri Seti İsmi	Çözünürlük	Video Formatı
VIRAT video veri seti	1920 x 1080	.mp4
UCF hava aksiyon veri seti	720 x 480	.mpg
GOP_Drone_Outdoor_Dataset	4096 x 2160	.mov

VIRAT video veri seti, doğal sahnelerin yer aldığı; insanlar ve araçların bulunduğu ve 2–30 Hz kare hızı ve kişilerin 10–200 piksele karşılık geldiği yüksekliklerde video görüntüleri içermektedir.

UCF hava aksiyon veri seti, bir gimbal üzerine monte edilmiş HD kamera ile donatılmış, uzaktan kumanda kontrollü bir keşif balonu kullanılarak elde edilen video dizilerini içermektedir. Koleksiyon, farklı yüksekliklerde ve havadan farklı bakış açılarında öne çıkan çeşitli eylem havuzunu temsil etmektedir. Her eylemin birden fazla

örneđi, 400-450 feet arasında deęiřen farklı uçuř irtifalarında kaydedilmiř ve farklı aktörler tarafından gerçekleştirilmiřtir.

řekil 3.1’te yer alan DJI Phantom 3 PRO 4K Drone kullanılarak açık alanda farklı insan hareketlerinden oluřan ve 24 fps’e sahip GOP_Drone_Outdoor_Dataset bu alıřma kapsamında oluřturulmuř ve kullanılmıřtır. Görüntülerin elde edilmesinde kullanılan “drone”nin teknik özelliklerine Tablo 3.2’de yer verilmiřtir.



řekil 3.1. DJI Phantom 3 PRO 4K Drone

Tablo 3.2. DJI Phantom 3 PRO 4K Drone teknik özellikleri

Özellik	Deęer
Aęırlık	1280 g
Maksimum hız	16 m/s
Batarya	4480 mAh LiPo 4S
Uçuř mesafesi	~25 dk
Kamera	4K
GPS modu	GPS var
Uçuř mesafesi	500m
Diyafram açıklıęı	f/2.8
Görüş açısı	FOV 94°
Sensör	Sony EXMOR 1/2.3" CMOS
Video çözünürlüęü	4096x2160p
Saniyedeki kare hızı	25fps

Hareketli gimbal	3 eksenli gimbal
Kontrol edilebilir aralık	Eğim -90° ile +30° arası

Söz konusu veri setleri üzerindeki görüntüleri anlamlandırmak amacıyla açık kaynak kodlu olan Python programlama dili kullanılmıştır. Bu programlama dilinin yanı sıra OpenCv ve Pytorch kütüphaneleri ile bir derin öğrenme algoritması olan YOLO’dan istifade edilmiştir.

Görüntülerin etiketlenmesinde “*labelImg*” programı ile “*roboflow*” platformundan faydalanılmış, çalışma kapsamında geliştirilen derin öğrenme modeli ise bulut tabanlı Google Colab (Google Colaboratory) üzerinde NVIDIA Tesla T4 16GB GPU kullanılarak eğitilmiştir. Eğitilen modelin kullanıldığı bilgisayarın teknik özellikleri Tablo 3.3’te yer almaktadır.

Tablo 3.3. Modelin kullanıldığı bilgisayarın teknik özellikleri

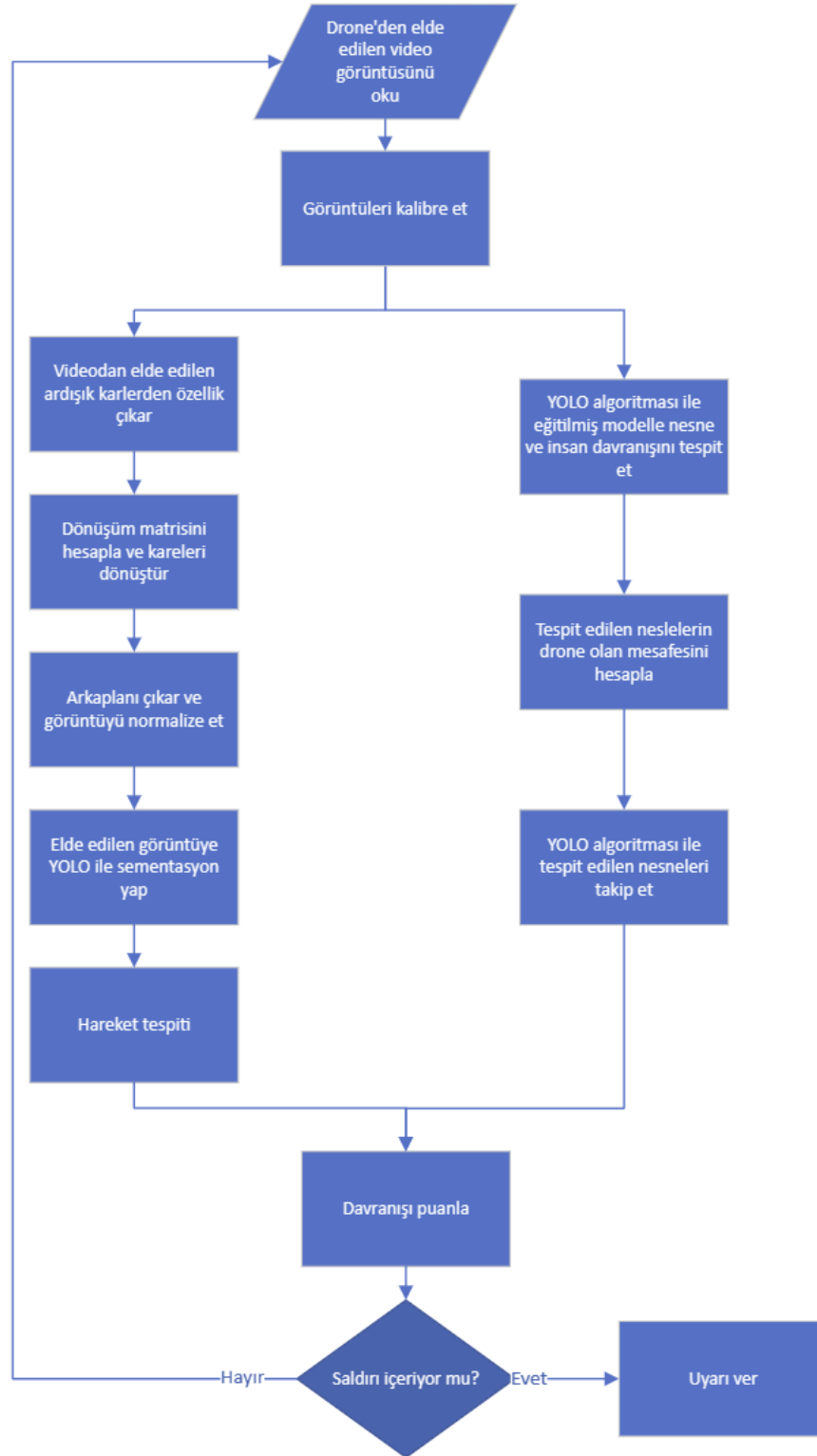
Donanım	Teknik Özellik
Ekran	15.6" FHD (1920 x 1080) IPS 300 Hz
Bellek (RAM)	32 GB DDR4 (16 GB + 16 GB Samsung 3200MHz)
Bellek (SSD)	KIOXIA 1 TB M.2 SSD
Ekran Kartı (GPU)	NVIDIA GeForce RTX 3070 8 GB GDDR6 256 Bit
İşlemci	Intel(R) Core(TM) i7-10870H CPU @ 2.20GHz

3.2. Yöntem

Saldırı tespiti için DJI marka “*drone*” kullanılarak sırası ile 10 m, 20 m, 30 m, 40 m ve 50 m yüksekliklerden farklı arazi koşullarında tespit edilmek istenen insan hareketlerinin yer aldığı video görüntüleri toplanmıştır.

“*Drone*” kamerası ile zemin arasındaki açının bilinmesine mesafe tespiti için ihtiyaç duyulmuştur. Bu açının bilinmesi ve sabit tutulması mesafe tespitinde önemlidir.

Kamera açısının belirli bir değerde tutularak alınan görüntülerden sırasıyla hareket tespiti, nesne tespiti, mesafe tespiti ve saldırı tespiti yapılarak nesne takibi yapılmaya çalışılmıştır. Görüntü analizlerinin doğru yapılabilmesi için her şeyden önce kameranın kalibrasyonu gerçekleştirilmiştir. Ardından eş zamanlı olarak hem nesne tespiti hem de hareket tespiti yapılmaya çalışılmıştır. Optik akış kullanılarak kamera hareketi sönümlenmiş, ardışık kareler farkı yöntemi ile hareketli nesne genel hatları ile ön plana çıkarılmış ve YOLOv8 segmentasyon algoritması ile hareketli pikseller daha detaylı bir şekilde tespit edilmeye çalışılmıştır. İnsan davranışlarının tespiti içinse hem YOLOv8 nesne tespiti algoritması hem de YOLOv8 segmentasyon algoritması kullanılmıştır. Daha sonra nesnenin uzaklığı hesaplanmıştır. Yapılan işlemlerin ardından insan davranışları puanlanarak saldırı tespitinin yapılması sağlanmıştır. Geliştirilen yazılımın blok diyagramı Şekil 3.2’de gösterilmektedir.



Şekil 3.2. Geliştirilen yazılımın blok diyagramı

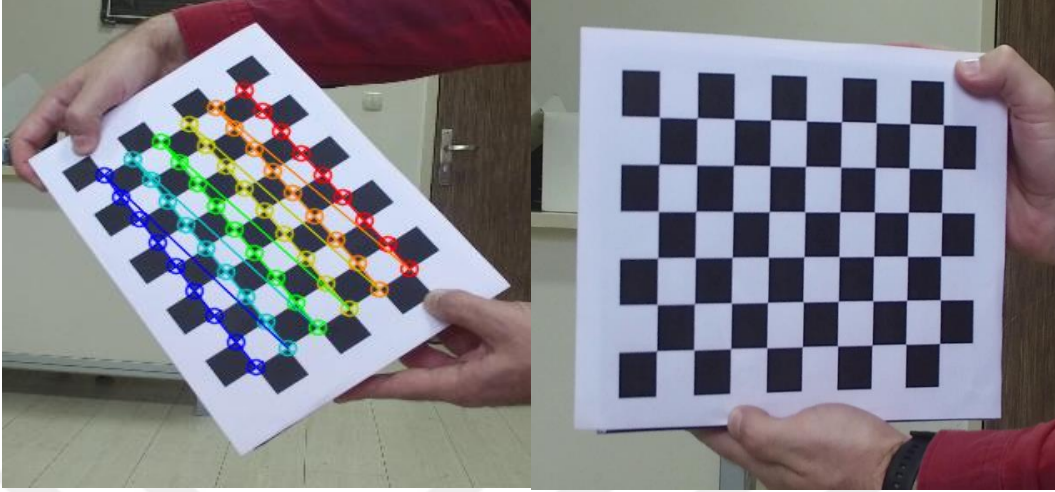
3.2.1. Kamera kalibrasyonu

Kamera kalibrasyonu, elde olmayan dış faktörler nedeniyle bozulmuş kamera görüntülerinin görüntü işlemeye hazır hale getirilmesidir. Kalibrasyonu yapılmamış kamera görüntülerinin görüntü analizi için hatalı olduğu daha doğrusu bir gürültü içerdiği bilinmektedir. Bu durum bazı görüntülerde insan gözüyle anlaşılırken bazılarında anlaşılamamaktadır (Şekil 3.3).



Şekil 3.3. Kamera kalibrasyonu öncesi ve sonrası (Lee ve ark., 2019)

Kamera kalibrasyonu yapmak mesafe ölçümünde en doğru tahmini elde etmek adına önem arz etmektedir. Bu işlemde kalibrasyonu yapılmak istenen kamera kullanılarak elde edilmiş satranç tahtası görselleri sıkça kullanılmaktadır.



a) Satranç tahtası görseli ile OpenCv kullanarak köşelerin tespiti

b) Kalibre edilmiş kameradan elde edilen satranç tahtası görseli

Şekil 3.4. Satranç tahtası görseli kullanılarak kamera kalibrasyonu

Kamera kalibrasyonu işlemi, görüntüdeki bozulmanın kaynağı olan kamera üzerinde yapılmaktadır. Kalibrasyon işlemi kameraya özgüdür. OpenCv içerisinde yer alan kalibrasyon bileşeni görüntü üzerinden referans noktaları belirleyerek (satranç tahtası üzerinden) bozulmanın oranı hesaplanmaktadır (Şekil 3.4). Kamera ile farklı açılardan alınmış satranç tahtasına ait görüntüler kullanılarak bozulmayı giderecek parametrelerin hesaplanması işlemi yapılmış ve bu parametreler “*yaml*” formatında kaydedilmiştir (Şekil 3.5). Hesaplama sonucu elde edilen “*camera_matrix*” ve “*dist_coeff*” değerleri ile kameranın sonraki görüntüleri için kalibrasyonu sağlanmaktadır.

```

1 camera_matrix:
2 - - 7915.040246044395
3   - 0.0
4   - 295.2097203999653
5 - - 0.0
6   - 8650.232686490946
7   - 236.66726302108665
8 - - 0.0
9   - 0.0
10  - 1.0
11 dist_coeff:
12 - - -0.5606518149464288
13   - 9.76834709302872
14   - -0.0420638398229176
15   - -0.048442808548644155
16   - -40.99966228027021

```

Şekil 3.5. “yaml” formatında kalibrasyon parametreleri

3.2.2. Etiketleme işlemleri

Nesne tespiti için etiketleme işlemi, görüntüdeki nesnelerin etrafına sınırlayıcı kutular çizerek nesnenin ait olduğu sınıfın görüntü üzerinde belirtilmesi anlamına gelmektedir. Eğitilecek derin öğrenme modeli tarafından tanınması istenilen nesnenin, doğru bir şekilde özniteliklerinin çıkarılması ve görüntü üzerinde etiketlenmesi model başarısı için elzemdir. Etiketleme işlemi sonucunda nesneleri görüntü üzerinde çevreleyen dikdörtgenlerin x ve y koordinatları ile ait oldukları sınıflara ait bilginin yer aldığı .txt uzantılı dosya elde edilmektedir. Bu dosya model eğitimi için görüntülerin yanında ilave olarak kullanılmaktadır. Her görüntü için ayrı oluşturulan .txt dosyası, görüntü içinde yer alan nesneleri çevreleyecek dikdörtgenin koordinat bilgilerini 0 ile 1 arasında normalize edilmiş halde tutmaktadır. Etiket bilgilerinin yer aldığı .txt dosyasındaki veriler Eşitlik 6’da gösterilen formatta kaydedilmektedir.

(sınıf_id, x, y, genişlik, yükseklik)

(6)

Her görüntü kendi boyutuna göre değerlendirileceği için nesne tespit sisteminde çeşitli boyutlardaki görüntüler kullanılabilir. Çalışma kapsamında 4 830 adet görüntü 14 farklı sınıf için etiketlenmiş ve her bir sınıf için kullanılan örnek sayısı Tablo 3.4'te gösterilmiştir.

Tablo 3.4. Etiketleme yapılan sınıflar

No	Sınıf İsmi	Örnek Sayısı
1	aiming (hedef alan)	349
2	animal (hayvan)	99
3	armed (silahlı)	367
4	carrying_something (elinde bir şey taşıyan)	368
5	hands_up (elleri yukarıda)	265
6	hiding (saklanan)	263
7	hiting (vuran)	317
8	kicking (tekme atan)	348
9	punching (yumruk atan)	280
10	running (koşan)	330
11	squatting (çömelen)	314
12	standing (ayakta duran)	582
13	vehicle (araç)	596
14	walking (yürüyen)	352

Saldırı tespitinde kullanılmak üzere daha önce belirlenmiş sınıflara ait etiketleme işlemleri “*roboflow*” ve “*labelImg*” üzerinde gerçekleştirilmiştir. “*Roboflow*” görüntüleri elle etiketleme için kullanılan bilgisayarlı görü modelleri oluşturmaya yönelik bir “framework”tür. “*labelImg*” ise açık kaynak kodlu bir etiketleme programıdır.



a) Segmentasyon için etiketleme

b) Nesne tespiti için etiketleme

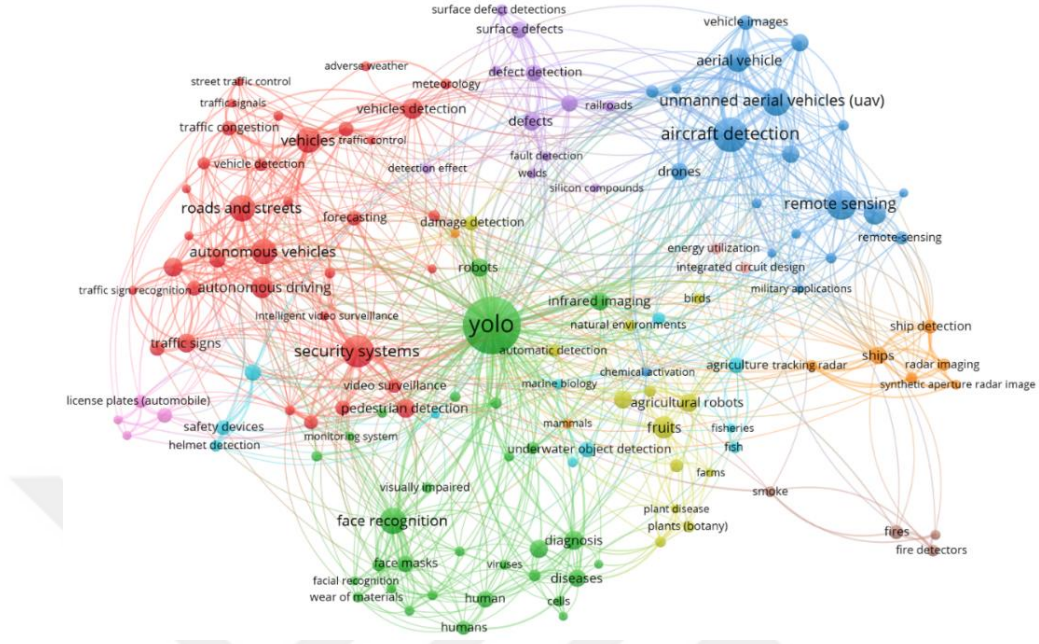
Şekil 3.6. Görüntülerin etiketlenmesi

Nesnelerin etiketleme işlemi, öncelikle “*labelImg*” programı kullanılarak tanınması istenilen nesne etrafına görüntü üzerinde sınırlayıcı kutu çizilerek yapılmıştır. Ancak nesnenin görüntüde etiklendiği anda arka planda istenmeyen objelerin de etikete dahil olması, modelin yanlış pozitif (false positive) tespitlerde bulunmasına neden olduğu gözlemlenmiştir. Bu amaçla “*roboflow*” platformu kullanılarak görüntünün piksel bazında etiketlenmesi sağlanmış ve model eğitiminde bu görüntüler kullanılarak daha doğru tespitlerin yapılmasına çalışılmıştır (Şekil 3.6).

3.2.3. Derin öğrenme modelinin seçimi

Geliştirilmeye devam eden ekran kartları kapasitesi sayesinde yeni derin öğrenme modelleri ortaya çıkmakta ve bu modellerin başarısı artmaktadır (Glenn ve ark., 2020). Farklı nesne algılama algoritmaları arasında YOLO algoritması, görüntülerdeki nesnelerin hızlı ve güvenilir bir şekilde tanımlanmasına olanak tanıyan gelişmiş hız ve doğruluk dengesiyle öne çıkmaktadır. YOLO ailesi, başlangıcından bu yana, her biri sınırlamaları gidermek ve performansı artırmak için önceki sürümlerin üzerine inşa edilen birden fazla yineleme yoluyla geliştirilmiştir. YOLO modeline ek olarak, nesne algılama ve görüntü işleme alanında başka birçok önemli yöntem ortaya çıkmış, R-CNN (Bölge Tabanlı Evrişimli Sinir Ağları) (Girshick ve ark., 2014) ve onun ardılları olan Fast R-CNN (Girshick, 2015) ve Faster R-CNN (Ren ve ark., 2015) gibi teknikler,

nesne algılama doğruluğunun geliştirilmesinde önemli bir rol oynamıştır. Bu yöntemler, seçici aramanın bölge önerileri ürettiği ve evrişimli sinir ağlarının bu bölgeleri sınıflandırıp hassaslaştırdığı iki aşamalı bir sürece dayanmaktadır. Bir diğer önemli yaklaşım ise, YOLO'ya benzer şekilde, ayrı bir bölge teklifi adımına olan ihtiyacı ortadan kaldırarak hız ve verimliliğe odaklanan Tek Atışlı Çoklu Kutu Dedektörüdür (SSD) (Liu ve ark., 2016). Ek olarak, Mask R-CNN (He ve ark., 2017) gibi yöntemler, örnek bölümlendirmeye yönelik genişletilmiş yeteneklere sahip olup, hassas nesne yerelleştirmesine ve piksel düzeyinde bölümlemeye olanak tanımaktadır. Bu gelişmeler, RetinaNet (Lin ve ark. 2017) ve EfficientDet (Tan ve ark., 2020) gibi diğerlerinin yanı sıra, nesne algılama algoritmalarının çeşitliliğini artırmaya katkıda bulunmuştur. Her yöntem, farklı uygulama ihtiyaçlarını ve hesaplama kısıtlamalarını karşılayarak hız, doğruluk ve karmaşıklık arasında benzersiz dengeler sunmaktadır (Terven ve ark., 2023). Bu araştırmada “*drone*” üzerinden anlık video görüntülerinin işlenmesi amaçlandığı için hız ve doğruluk açısından optimum verim sağlayan YOLO algoritması tercih edilmiştir. YOLO bir nesnenin sınıfını belirlemek ve sınırlayıcı kutusunu koordinatlandırmak için bir kez ileri geçiş yaparak tüm işlemi gerçekleştirmektedir. Bu sayede elde ettiği yüksek hız, gerçek zamanlı uygulamalarda etkili bir performans sunmasına olanak sağlamaktadır. Diğer modellere göre gösterdiği başarı YOLO’yu; güvenlik uygulamalarından otonom araçlara kadar pek çok farklı alanda gerçek dünya uygulamaları için onu ideal kılmaktadır. Bu anlamda akademik çalışmalarda farklı konularda kullanımı sıkça tercih edilmektedir (Şekil 3.7).



Şekil 3.7. YOLO uygulamalarının bibliyometrik ağ görselleştirmesi (Terven ve ark., 2023)

Tez çalışmasına başlarken YOLO ailesinden 2020 yılında kullanıma sunulan YOLOv5 algoritmasının kullanımı öngörülmüştür. Ancak 2023 yılının Ocak ayında kullanıma sunulan YOLOv8'in en güncel model olması, YOLOv5 ve diğer YOLO modellerine kıyasla daha iyi performans sergilemesi tercih sebebi olmuştur (Tablo 3.5).

Tablo 3.5. YOLOv8 ile YOLOv5 kıyaslaması (Rath, 2023)

Model Boyutu	Tespit (%) ¹	Segmentasyon (%) ¹	Sınıflandırma (%) ²
Nano (N)	+33.21	+32.97	+3.10
Small (S)	+20.05	+18.26	+1.12
Medium (M)	+10.57	+10.89	+0.66
Large (L)	+7.96	+6.73	+0.00
Extra Large (X)	+6.31	+5.33	- 0.76

¹ Görüntü boyutu = 640

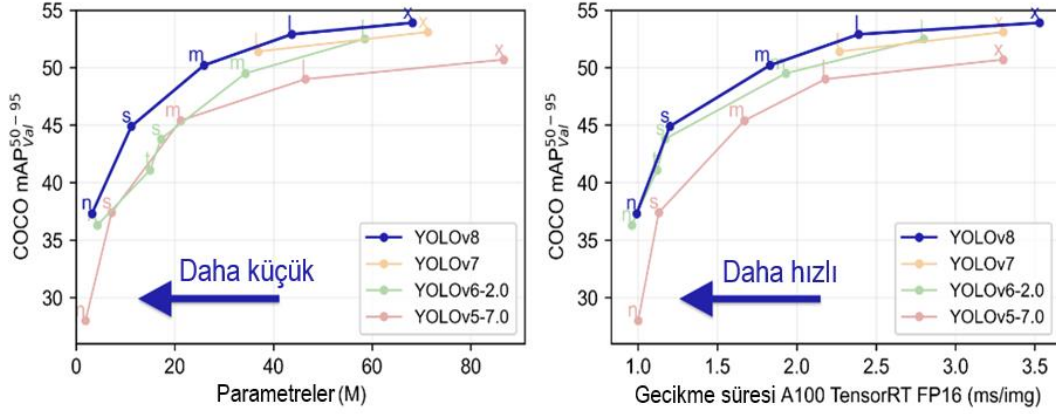
² Görüntü boyutu = 224

Tüm YOLO modelleri, görüntülerin daha hızlı veya küçük nesnelerin daha iyi işlenmesi gibi farklı olanaklar sunan varyantlara sahiptir. YOLOv8'in varyantları arasındaki temel fark, modelin boyutu ve karmaşıklığıdır. Daha büyük ve daha karmaşık modeller yüksek doğruluk değerlerine sahip olurken aynı zamanda daha yavaştır. Daha küçük ve daha az karmaşık modeller ise daha az doğruluğa sahiptir ancak aynı zamanda daha hızlıdır. YOLOv8 varyantları arasındaki farklar özet olarak Tablo 3.6'te verilmiştir. YOLOv8 hem doğruluk hem de yürütme süresi açısından bilinen tüm modellerden daha iyi performans göstermektedir (Şekil 3.8).

Tablo 3.6. YOLOv8 modeline ait varyantların performansları

Model	Size (pixels)	mAPval 50-95	Speed CPU ONNX (ms)	Speed A100 TensorR (ms)	Params (M)	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2
YOLOv8x	640	53.9	479.1	3.53	68.2	257.8

YOLOv8 daha önceki versiyonlarından birkaç durumda pozitif yönde farklılaşarak performans olarak öne çıkmaktadır. Gelişmiş doğruluk oranıyla tespiti zor ve küçük nesnelerde önceki versiyonlarına göre daha öndedir. Ayrıca YOLOv8 daha hızlı olmasına rağmen doğruluk anlamında da diğer versiyonlarına göre daha iyidir (Şekil 3.8).



Şekil 3.8. YOLOv8 performans karşılaştırması (Jocher ve ark., 2023)

Modelin performansının değerlendirilmesi

Karışıklık Matrisi (Confusion Matrix) derin öğrenme ve makine öğrenmesi modellerinin başarısını ölçmek için kullanılan bir tablolama tekniğidir. Bu teknik sınıflandırma modelinin yaptığı doğru ve yanlış tahminlerin detayını vermektedir. Karışıklık matrisine ek olarak doğruluk, kesinlik, duyarlılık ve F1 skor değerleri de başarı ölçütü olarak kullanılmaktadır (Elgendy, 2020).

		Gerçek	
		Pozitif	Negatif
Tahmin Edilen	Pozitif	Doğru Pozitif	Yanlış Pozitif
	Negatif	Yanlış Negatif	Doğru Negatif

Şekil 3.9. Karışıklık matrisi

Şekil 3.9’de yer alan terimler;

Doğru Pozitif (True Positive-TP): Modelin, pozitif değeri pozitif olarak doğru tahmin ettiği,

Doğru Negatif (True Negative-TN): Modelin, negatif değeri negatif olarak doğru tahmin ettiği,

Yanlış Pozitif (False Positive-FP): Modelin, negatif değeri pozitif olarak yanlış tahmin ettiği,

Yanlış Negatif (False Negative-FN): Modelin, pozitif değeri negatif olarak yanlış tahmin ettiği durumları ifade etmektedir.

Eğitilen modelin başarısını ölçmek için kullanılan diğer ölçütler Eşitlik 7, Eşitlik 8, Eşitlik 9 ve Eşitlik 10’da gösterildiği gibi hesaplanmaktadır:

$$\text{Accuracy (Doğruluk)} = \frac{TP + TN}{TP + FP + TN + FN} \quad (7)$$

$$\text{Precision (Kesinlik)} = \frac{TP}{TP + FP} \quad (8)$$

$$\text{Recall (Duyarlılık)} = \frac{TP}{TP + FN} \quad (9)$$

$$F_1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \quad (10)$$

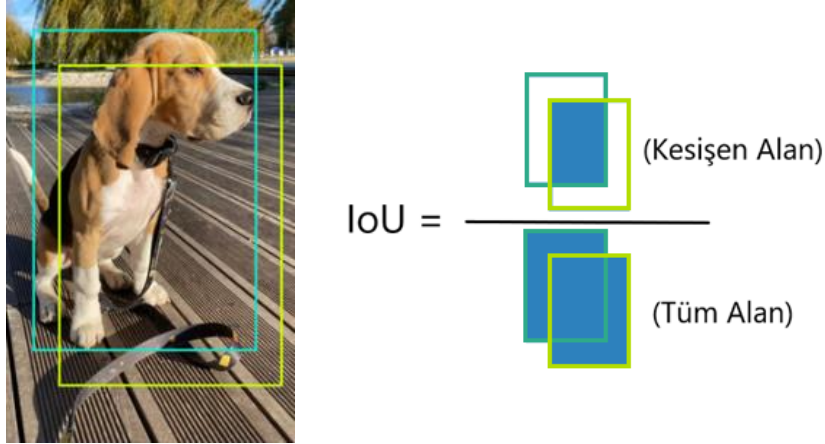
Average Precision (AP), modelin farklı hatırlama seviyelerindeki hassasiyetinin ölçüsüdür. AP, kesinlik-duyarlılık eğrisinin altındaki alandır. Daha yüksek bir AP değeri, modelin daha iyi bir performans gösterdiğini ifade etmektedir.

Mean Average Precision (mAP), bir modelin kesinliği, gerçek pozitiflerin sayısının, gerçek pozitiflerin artı yanlış pozitiflerin sayısına bölünmesiyle tanımlanmaktadır. mAP metriği ise modelin hem kesinliğini hem de duyarlılık değerini hesaba katarak her

sınıf için ortalama kesinliğin ortalaması olarak hesaplanmaktadır. Daha yüksek bir mAP değeri, modelin daha iyi bir performans gösterdiğini ifade etmektedir.

Nesne tespitinde kullanılan diğer bir ölçüt ise Intersection over Union (IoU)'dur. IoU'ya Jaccard indeksi de denmektedir. Nesne tespitinde kullanılan bir metriktir. Makine öğrenmesi veya derin öğrenme modelinin başarısının ölçülmesi için test veri kümesi üzerinden hesaplanmaktadır. Tespit edilmek istenilen nesnenin görüntüdeki hedeflenen konumu Ground Truth (Kesin Referans) olarak adlandırılmaktadır. Ground Truth değeri ile modelin tahmini arasındaki örtüşme Jaccard indeksi ile ölçülmektedir (Şekil 3.10). IoU değeri, Kesin Referans Maskesi (The Ground Truth Mask - gt) ve Tahmin Maskesi (The Predict Mask - pd) kullanılarak Eşitlik 11'de yer alan formül ile hesaplanmaktadır.

$$IoU = \frac{alan(gt \cap pd)}{alan(gt \cup pd)} \quad (11)$$



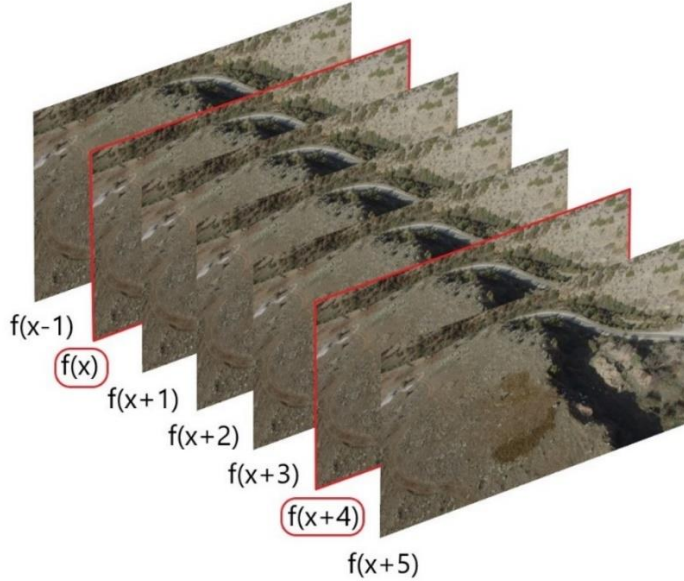
Şekil 3.10. IoU hesaplanması

3.2.4. Saldırı tespitine yönelik geliştirilen yöntem

Çalışma kapsamında saldırı tespiti için öncelikle hareket tespiti yapılmakta ardından nesne tespiti ve son olarak bu nesnenin “drone”a olan uzaklığı hesaplanmaktadır.

Adım 1

Hareket tespiti için ilk olarak hareket telafisi yöntemi kullanılmıştır. Bu yöntemin kullanılması için öncelikle; ardışık kareler arasında kamera hareketinin telafi edilmesi ve buradan kaynaklı bozulmanın söndürülmesi amaçlanmıştır. Arka plan dengeleme, ilk karede elde edilen özelliklerin ardı sıra gelen kareler üzerinde eşleştirilerek dönüşüm matrisinin hesaplanması ve bu matris ile görüntünün arka planının dengelenmesi esasına dayanmaktadır. Karşılaştırılan kareler Şekil 3.11’de görüleceği üzere bir kuyrukta tutularak her bir kare kendinden beş kare öncesi ile kıyaslanmaktadır. Kıyası yapılan kareler birbirlerinden çıkarılıp farkı alınmaktadır. Görüntüde arka planın dengelenmesi sonucu hareketli kısımlar ön plana çıkarılabilmektedir.



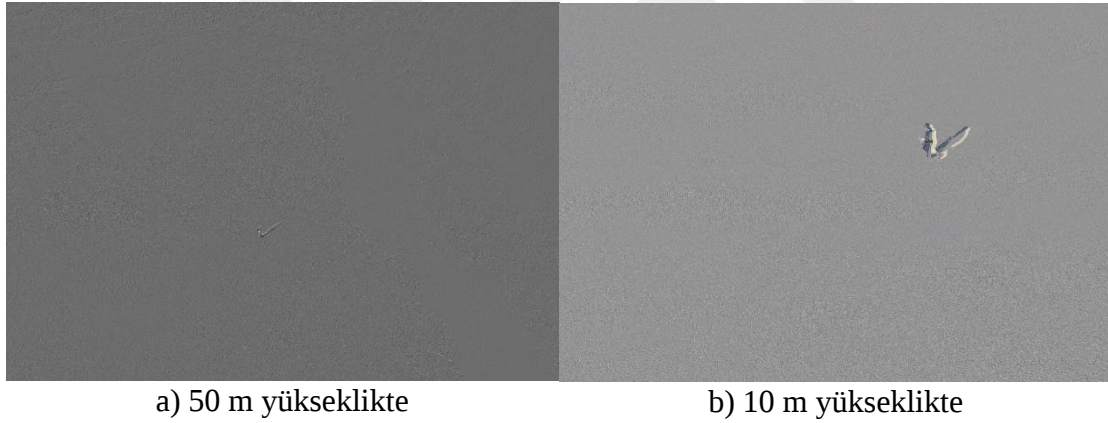
Şekil 3.11. Ardışık kare kuyruğu

Kuyrukta yer alan karelerin arka planının dengelenmesi sonucu çıkarılan özellikler optik akış kullanılarak diğer karede de aranmaktadır. Hesaplanan özellik noktalarına göre sonraki kare önceki kareye homografi matrisi (dönüşüm matrisi) ile

çarpıtılmaktadır. Bu sayede arka plandan hareketli nesnenin genel hatlarıyla ile ayrıştırılması sağlanmıştır.

Ardışık kare farkı ile elde edilen görüntüden kabaca hareketli nesneye ait pikseller tespit edilmektedir. Ancak arka plan; rüzgar ve benzeri çevresel faktörler nedeniyle çimler ve ağaçlar gibi hareketli nesneler içerebilir ve bu da hareketli nesnenin tespitini güçleştirmektedir (Stojnic ve ark.,2021). Bu güçlüğü aşmak için birbirinden çıkarılan kareler ve bu karelerin farkından oluşan yeni kare, piksel bazında Eşitlik 12’de verilen formül ile normalize edilmiştir (Şekil 3.12).

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (12)$$



Şekil 3.12. Farklı yüksekliklerden elde edilen hareketli nesnelere ait normalize edilmiş görüntüler

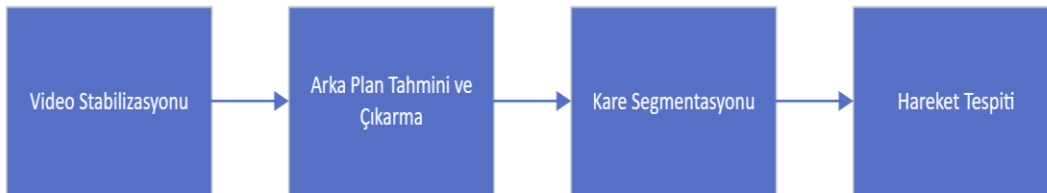
Hareketli nesnelerin tespitine ilişkin yapılan bu işlemlere rağmen basit bir eşikleme çok fazla hatalı pozitif tespite yol açacağı anlaşılmıştır. Hareketli nesneyi segmentlere ayırarak çalışmanın başarısını artırmak amacıyla YOLOv8 modeli kullanılmıştır. Fark görüntülerinden hareketli kısımların etiketlenmesi ile modelin eğitimi için gerekli veri (1 180 adet ikili formatta görüntü) elde edilmiştir.

Diğer taraftan kamera hareketinin fazla olduğu durumlarda arka plan çok hızlı bir şekilde değişeceğinden hareketli nesneye ait görüntü pikselleri tespitin doğru sonuç vermediği gözlemlenmiştir. Ancak ardışık karelerdeki arka plan değişim oranının belirli bir seviyede olduğunda başarılı tespitler yapılabilmektedir. Bu amaçla ardışık kareler için Eşitlik 13’te gösterilen Mean Square Error (MSE) hesaplanmaktadır.

$$MSE = \frac{\sum (Y_i - P_i)^2}{n} \quad (13)$$

640X640 pikseli görüntü için eşik değer MSE≈33 olarak belirlenmiştir. Belirlenen eşik değeri aşıldığında hareket tespiti algoritmasının çalışmayı durdurması sağlanarak hem işlem gücünün verimli kullanılması ve hem de doğru tespitlerin yapılması amaçlanmıştır.

Hareketli nesne tespiti için elde edilen karelerin farklarından oluşan görüntüler ve YOLOv8 segmentasyon algoritması kullanılarak bir model eğitilmiştir. Model, “drone” üzerinden elde edilen ve test verisi olarak ayrılan görüntüler kullanılarak hareketli nesneye ait piksellerin tespiti için test edilmiştir. Hareket tespiti için eğitilen modelde ikili formattaki 1 180 adet görüntü kullanılmıştır. Model; YOLOv8 algoritması, Pytorch “framework”ü, 640x640 çözünürlüğündeki görüntüler ve Google Colab üzerinde NVIDIA Tesla T4 16GB GPU kullanılarak 100 öğrenme döngüsü (epoch) ile eğitilmiştir. Veri setinin %85’i eğitim %15’u ise test işlemleri için ayrılmıştır. Hareket tespiti için geliştirilen yöntemin blok diyagramı Şekil 3.13’te yer almaktadır.



Şekil 3.13. Hareket tespiti için önerilen yöntemin blok diyagramı

Adım 2

Görüntülerden nesnelerin tespiti ve sınıflandırılması için YOLOv8'in **nesne tespit algoritması** ile **görüntü segmentasyon algoritması** ayrı ayrı kullanılmıştır. Hem örnek segmentasyonu hem de nesne algılama modelleri, her bir nesnenin görüntüdeki konumunu tanımlamaktadır. Ancak nesne algılama, dış hatlarını değil yalnızca nesnenin sınırlayıcı kutusunu (box-kutu) tahmin etmektedir. Örnek segmentasyonu ise, nesne tespitinden bir adım daha ileri giderek bir görüntüdeki tek tek nesnelerin tanımlanmasını ve bunların görüntünün geri kalanından bölümlendirilmesini (mask-maske) içermektedir (Jocher ve ark., 2023). Seçilen derin öğrenme modelleri daha önce etiketleme işlemi gerçekleştirilmiş 14 sınıfa ait 4 830 görselden oluşan bir veri seti ile eğitilmiştir. “Drone” üzerindeki kameradan elde edilen görüntülerin sayısını artırmak için *Image Augmentation (Görüntü Artırımı)* yöntemine başvurulmuştur. Görüntü artırımı, veri kümesini yapay olarak genişletmek için kullanılan bir tekniktir. Bu, az veri örneği içeren bir veri seti verildiğinde kullanımı faydalıdır. Görüntü artırımı görüntülerin farklı formlardaki versiyonunun da alınarak modelin beslenmesidir (Şekil 3.14).



Şekil 3.14. Görüntü artırımı işlemi

YOLO modellerine yönelik görüntü sayısı büyütme ayarları, veri kümesinin çeşitliliğini ve boyutunu artırmak için eğitim verilerine uygulanan çeşitli dönüşümleri ve modifikasyonları ifade etmektedir. Çalışma kapsamında YOLO veri artırma yöntemleri kullanılarak eğitim verisi 4 katına çıkarılmıştır (Tablo3.7). YOLO veri artırma ayarları, uygulanan dönüşümlerin türünü ve yoğunluğunu (örneğin, rastgele çevirmeler, döndürmeler, kırpma, renk değişiklikleri), her dönüşümün uygulanma olasılığını ve maskeleri gibi ek özellikleri içermektedir. Artırma sürecini etkileyebilecek diğer faktörler arasında orijinal veri kümesinin boyutu ve bileşimi ile modelin kullanıldığı belirli görevler yer almaktadır. Artırılmış veri kümesinin çeşitli olduğundan ve yüksek performanslı bir modeli eğitmeye yetecek kadar temsili olduğundan emin olmak için bu ayarları dikkatli bir şekilde ayarlamak ve denemeler yapmak önemlidir. Bu ayarlar modelin performansını, hızını ve doğruluğunu etkilemektedir (Jocher ve ark., 2023).

Tablo 3.7. Görüntü sayısını artırma için kullanılan yöntemler

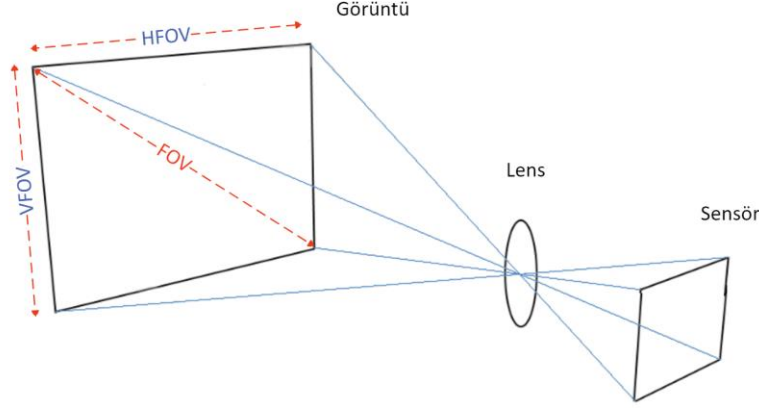
Anahtar	Değer	Tanım
hsv_h	0.015	Görüntü sayısını HSV-Hue ile artırma
hsv_s	0.7	Görüntü sayısını HSV-Saturation ile artırma
hsv_v	0.4	Görüntü sayısını HSV-Value ile artırma
degrees	0.0	Görüntü döndürme (+/- derece)
translate	0.1	Görüntü çevirme (+/- kesir)
scale	0.5	Görüntü skala değiştirme (+/- kazanç)
shear	0.0	Görüntü kırpma (+/- derece)
perspective	0.0	Görüntü perspektifi (+/- kesir), aralık 0-0,001
flipud	0.0	Görüntüyü yukarı aşağı çevirme
fliplr	0.5	Görüntüyü sola sağa çevirme
mosaic	1.0	Görüntüyü mozaikleme
mixup	0.0	Görüntüleri karıştırma
copy_paste	0.0	Segment kopyala yapıştır

auto_augment	randaugment	Sınıflandırma için otomatik büyütme politikası (randaugment, autoaugment, augmix)
erasing	0.4	Sınıflandırma eğitimi (0-1) eğitimi sırasında rastgele silme olasılığı

Model; Google Colab üzerinde NVIDIA Tesla T4 16GB GPU kullanılarak eğitilmiştir. Eğitim; etiketleme yapılmış, görüntülerde tespiti yapılmak istenilen nesnenin sınıf ve koordinat bilgisinin yer aldığı .txt formatında dosyalar ve 640x640 çözünürlüğündeki görüntüler kullanılarak 100 öğrenme döngüsü (epoch) ile gerçekleştirilmiştir. Veri setinin %85'i eğitim %15'u ise test işlemleri için ayrılmıştır.

Adım 3

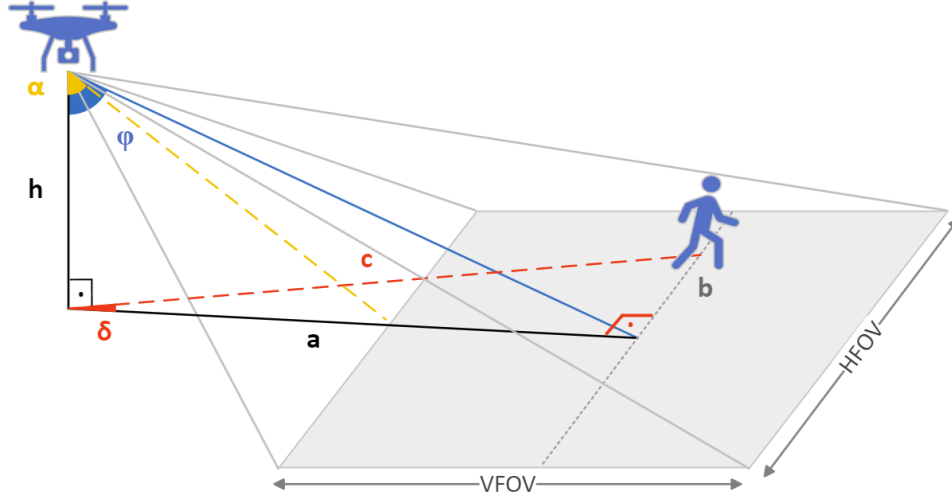
“Drone”un iz düşümü ile nesne arasındaki mesafenin hesaplanması için Şekil 3.16’da α açısının 20° ’de sabit olduğu durumda, kamera görüş açısı (FOV) değeri kullanılmaktadır. Kameraya özgü olan bu değer çalışmada kullanılan “drone” kamerasında 94° ’dir (bkz. Tablo 3.2). FOV değerinden yola çıkılarak ϕ açısı hesaplanmıştır. Çalışmada kullanılan “drone” kamerası 4096x2160 görüntü boyutlarında kayıt yapmaktadır. Bu görüntü çözünürlüğü hesaba katılarak çapraz görüş açısını ifade eden FOV değerinden bir piksele karşılık gelen görüş açısı belirlenmiştir. Ardından görüntü içinde tespiti yapılan nesnenin görüntünün alt sınırına olan mesafesi piksel bazında hesaplanıp bir piksele karşılık gelen görüş açısı ile çarpılarak $\phi - \alpha$ değeri elde edilebilmektedir.



Şekil 3.15. Kamera çapraz görüş alanı (FOV), dikey görüş alanı (VFOV), yatay görüş alanı (HFOV)

Görüntünün yüksekliği ve genişliğine ilişkin verilerden en-boy oranını yani sensörün ne kadarının kullanıldığı belirlenebilmektedir. VFOV'un görüntü yüksekliğine ve HFOV'un görüntü genişliğine bölünmesi, bir görüntü pikselindeki yükseklik ve genişliğe göre açıyı vermektedir (Şekil 3.15). Piksel başına açı değerinin, kişinin bulunduğu konumla çarpıldığında görüntüdeki kişiye ilişkin açı elde edilmiş olacaktır.

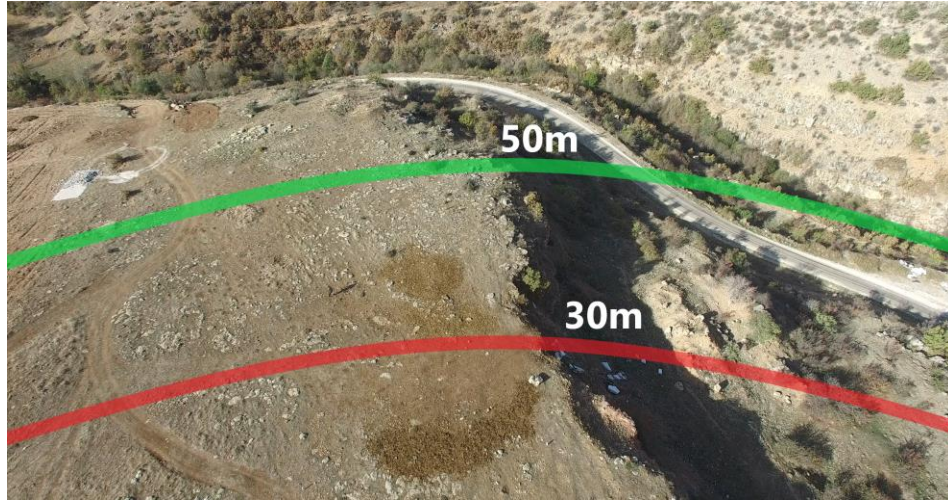
Mesafe tespiti için kullanılan yöntemde “*drone*” üzerinde konumlandırılmış kameranin zeminden yüksekliği (h) ve zemin ile arasındaki açısından (α) yola çıkılarak bir hesaplama yapılmaktadır. Şekil 3.16'daki α ile temsil edilen bu açı 20° 'de sabit tutularak elde edilmiş görüntülerden VFOV değeri kullanılarak ϕ açısı hesaplanmıştır. “*Drone*”un yerden yüksekliğini ifade eden h değeri ve ϕ bilindiğinde Eşitlik 14'te yer alan formül kullanılarak “*drone*” ile nesne arasındaki dikey mesafe olan “ a ” tespit edilebilmektedir. Benzer bir yöntemle ile HFOV değeri ve δ açısı kullanılarak “ b ” hesaplanmaktadır. “*Drone*”nun iz düşümü ile görüntüden tespiti yapılan nesne arasındaki uzaklığı temsil eden “ c ” değeri Eşitlik 15 ile elde edilebilmektedir.



Şekil 3.16. “Drone” kamerasından mesafe tespiti

$$\tan (\varphi) = \frac{a}{h} \quad (14)$$

$$c^2 = a^2 + b^2 \quad (15)$$



Şekil 3.17. Mesafe tespitini saldırı tespitinde kullanımı

Tespit edilen nesnenin mesafe tespitinin yapılması saldırı tespitinde kullanılmaktadır. Tespit edilen nesnenin “drone” olan uzaklığının azalması istenmeyen bir durumdur.

Mesafenin eşik değeri olan 30 m'nin altına düşmesi saldırı tespiti kapsamında değerlendirilmektedir (Şekil 3.17).

Adım 4

“Drone” görüntüleri üzerinden daha önce belirlenen sınıflara ait etiketler kullanılarak eğitimi yapılmış YOLO modeli ile saldırı içeren hareketler tespit edilmeye çalışılmıştır. Bu kapsamda herhangi bir nesneye ait hareketin tespiti ve insan hareketlerinin sınıflandırılması, saldırı tespiti yapılması aşamasında kullanılmıştır. Gerçek zamanlı bir senaryoda, kötü niyetli bir girişim olma ihtimalinin tahmin edilmesi halinde alarma geçilmesi amaçlanmıştır. Bu nedenle, olası hasarı önlemek veya en azından olası zararı azaltmak için kötü niyetli davranışları mümkün olduğunca erken tespit etmeye çalışılmıştır.

Saldırı tespiti için uygulanan yaklaşımın bir benzeri siber güvenlikte de kullanılmaktadır. Bu kapsamda monoton ve monoton-olmayan olmak üzere iki yaklaşım bulunmaktadır. Herhangi bir yazılım, yürütme anında kötü niyetli bir davranış sergilemiş ancak sonrasında zararsız hale geçmiş ya da tam tersi durumda uzun bir süre iyi niyetli davranış sergilerken ana yükünü daha sonra boşaltmış olabilir. Monoton-olmayan yaklaşımda bir hareketin sergilenmesi, şüpheli yazılımı zararsız olarak değerlendirmeye olanak sağlıyorsa zararlı yazılım bu iyi niyetli gibi görünen davranışı sergileyerek modeli atlatması mümkün olabilmektedir. Örneğin standart bir dizinden işlem başlatılması model için zararsız bir özellikse kötü niyetli yazılım bu dizinden ek birkaç işlem daha başlatarak detektörü atlatabilmektedir. Monoton yaklaşımda ise yazılımın çalışma süresi boyunca monoton olarak artan kötü niyetli olma olasılığı bir noktada eşik seviyeyi aşarsa yazılımın kötücül olduğuna karar verilmektedir ve yazılımın çalışma sürecinin sonuna kadar bu durum böyle kalmaktadır. Sonuç olarak zararlı bir yazılımın davranışlarını takip etmeye yönelik yapılan çalışmada zararlı olabileceği düşünülen ve bu yönde bir davranış sergileyen yazılımın her bir davranışı

skorlanmakta ve bu skor belirli bir eřiđi ařtıđında yazılımın zararlı olduđuna karar verilmektedir (Chistyakov ve ark., 2018).

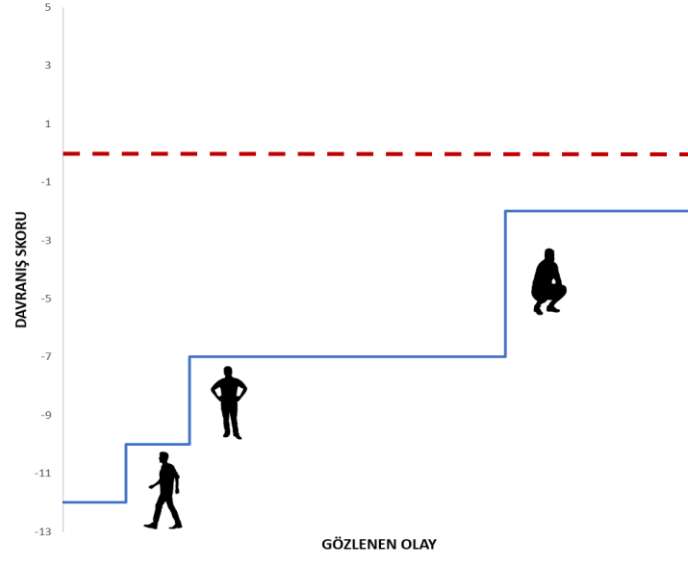
Siber gvenlikte kullanılan monoton yaklařımdan yola ıkılarak gvenlik amalı gzlenen alanda, sergilenen her bir insan hareketi skorlanarak belirli bir eřik seviyeyi gemesi durumunda grevlilerin uyarılmasına ynelik alarm durumuna geilmesi sađlanmıřtır. Bu dođrultuda elinde silah olduđu tespit edilen bir kiřinin silahı sonrasında saklasa dahi sistemin takibini srdrmesi sađlanmaya alıřılmıřtır.

Bu amala tespiti yapılmaya alıřılan davranıř grubuna ait hareketler skorlanmıřtır (řekil 3.18). Kamera grnt alanı iinde yer alacak herhangi bir tehlikeli durumun tespiti yapılmaya alıřılmıřtır. Sz konusu skorlamada: hareket veya nesne tespiti; +1 puan, elini kaldıran veya kořan kimsenin tespiti; +2 puan, elinde bir řey tařıyan, elinde bıak veya sopa savuran, tekme veya yumruk atan kimsenin tespiti; +10 puan ve son olarak silahlı veya silahı ile niřan alan birinin tespit edilmesinde ise +12 puan olarak deđerlendirilmiřtir.

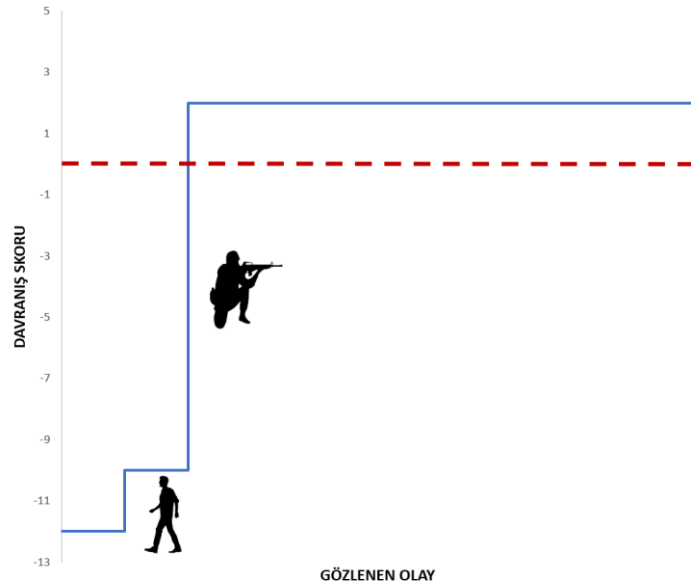


Şekil 3.18. Saldırı tespitinde kullanılan sınıflara ait görseller

Tespiti yapılan insanın davranışlarının güven değerinin 0.85'ten yüksek olduğu durumda skorlanması yapılarak saldırı davranış grafiği elde edilmektedir. Şekil 3.19'da görüleceği üzere gözlenen alana yürüy en biri girdiğinde ve beklemeye başladığında saldırı skoru artış göstermektedir. Buna ek olarak kişi oturduğunda ise saldırı tespiti eşiğine bir adım daha yaklaşılr ancak bir alarm durumu söz konusu değildir. Ne zaman ki tespit edilen insana ait davranış skoru saldırı eşiğini geçerse o zaman görevlilere uyarı vermek amacıyla alarm verilecektir.



a) Saldırı tespiti yapılmayan durum



b) Saldırı tespiti yapılan durum

Şekil 3.19. Örnek, saldırgan davranış grafiği

Çoklu nesne takibinde (MOT), tespit edilen nesnelere kimlik atamak, özellikle nesne hareketinde tıkanmalarla veya ani değişikliklerle karşılaşıldığında zorlu kararlar almayı

gerektirmektedir. YOLO modeli, nesne takibi için Macar algoritması çerçevesinde görünüm ve konum bilgilerinin bir kombinasyonunu kullanmaktadır. Ancak tekrarlanan veya benzer görünen nesneler, hızlı sahne değişiklikleri veya geçici kapanmalar gibi faktörler bazen takip edilen nesnelerin kimliklerinin değişmesine yol açabilmektedir. Çoklu nesne izlemenin karmaşıklığının doğası gereği nesnelerin takibi sırasında atanan kimlik değeri (id); kamera açısı, nesnenin başka bir nesnenin arkasına girmesi vs. durumlarda değişiklik göstermektedir. Bundan dolayı çalışmada insan davranış grafiği o anki kareden elde edilen veriler doğrultusunda oluşturulmaktadır. YOLOv8 algoritması ile nesne takibi yapılmaya çalışılmıştır ancak kareyi bir bütün olarak değerlendirmeye almak daha doğru bir yaklaşım olarak değerlendirilmiştir.

4. BULGULAR VE TARTIŞMA

Çalışmada standart bir video kamera ile donatılmış insansız hava araçlarının belirli bir bölgede konumlandırılması ile elde edilen görüntüler üzerinden saldırı durumunun tespiti yapılmaya çalışılmıştır. Bu kapsamda YOLOv8 derin öğrenme mimarisi ile hareket tespiti ve nesne tespiti yapmak amacıyla bir görüntü işleme yazılımı geliştirilmiştir. “Drone” üzerinden hareket, nesne ve mesafe tespiti yapılması sonucunda çalışmanın başarısı değerlendirilmiştir. Hareket ve nesne tespitinde görüntü segmentasyonu ve nesne tespiti yöntemleri kullanılarak YOLOv8 mimarisinin beş farklı versiyonu için başarı sonuçları; karışıklık matrisi, kesinlik, duyarlılık, F1 skoru ve mAP@0.5 değerleri üzerinden karşılaştırılmıştır.

4.1. Hareket Tespitine İlişkin Elde Edilen Bulgular

Ardışık kareler üzerinden özellik çıkarımı yapılmış, kareler normalize işlemine tabi tutulmuş ve birbirinden çıkarılarak hareketli görüntüye ait pikseller genel hatlarıyla elde edilmiştir. Ardından YOLOv8 ile görüntü segmentasyonu gerçekleştirilerek hareket tespitinin doğruluğunun daha ileriye taşınmasına çalışılmıştır. Bu kapsamda 640x640 boyutundaki 1 180 adet görüntü sayısı, veri artırma yöntemleri yardımıyla (bkz. Tablo 3.7) 4 720’ye çıkarılmıştır. Söz konusu görüntüler kullanılarak 100 öğrenme döngüsü (epoch) ile eğitilen varyantların başarısı Tablo 4.1 ve Tablo 4.2’de verilmiştir.

Tablo 4.1. Hareket tespiti için YOLOv8 varyantlarının kullanımı

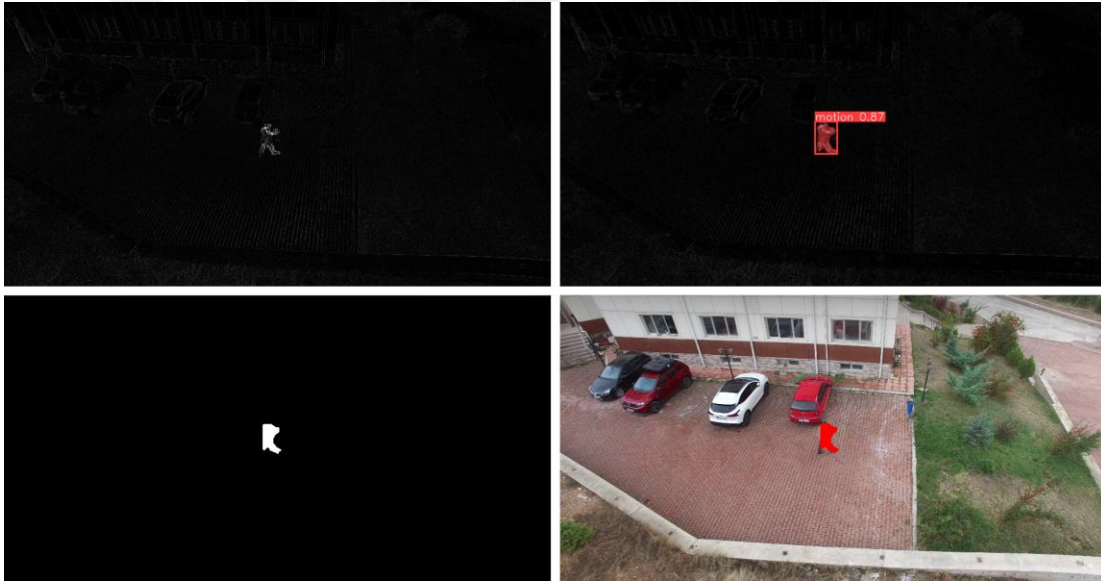
YOLOv8 Varyant İsmi	Katman Sayısı	Parametre Sayısı	Eğitim Süresi (sa.)	Çıkarım Süresi (ms)
Nano (N)	195	3 258 259	0.62	3.9
Small (S)	195	11 779 987	0.77	6.3
Medium (M)	245	27 222 963	1.15	14.6
Large (L)	295	45 912 659	1.8	21.9
Extra Large (X)	295	71 721 619	2.67	34.7

Beklendiği üzere karmaşıklığı ve büyüklüğü artması sebebiyle “*extra large*” varyantı için eğitim ve çıkarım süresi değerleri daha büyüktür (Tablo 4.1). Ancak bu çalışma için eğitim ve çıkarım süresinin artması oranınca başarısında dikkate değer bir farklılaşma söz konusu olmamıştır (Tablo 4.2).

Tablo 4.2’de YOLOv8 varyantlarının hareket tespitinde segmentasyon görevi için nesne algılama doğruluğu değerlerine yer verilmiştir. Tabloda belirtilen *maske (mask)* algılanan nesneye ait piksel grubunu, *kutu (box)* ise algılanan nesnenin etrafındaki sınırlayıcı kutuyu ifade etmektedir (bkz. Şekil 2.14). Tablo 4.2’de yer alan performans sonuçlarına bakıldığında “*medium*” varyantının ön plana çıktığı görülmektedir. Ancak eğitim süresi ve çıkarım süresi göz önünde bulundurulduğunda “*nano*”nun çalışma kapsamında kullanımı tercih edilmiştir. Yazılımın hareket tespitine yönelik görsel çıktılarına Şekil 4.1’de yer verilmiştir.

Tablo 4.2. Hareket tespiti için YOLOv8 varyantlarının performansı

YOLOv8 Varyant İsmi	Kesinlik		Duyarlılık		F1		mAP@0.5	
	Kutu	<i>Maske</i>	Kutu	<i>Maske</i>	Kutu	<i>Maske</i>	Kutu	<i>Maske</i>
Nano (N)	0.80	0.84	0.85	0.88	0.82	0.86	0.85	0.81
Small (S)	0.81	0.75	0.85	0.80	0.82	0.78	0.86	0.82
Medium (M)	0.87	0.84	0.88	0.84	0.85	0.84	0.89	0.87
Large (L)	0.80	0.73	0.85	0.81	0.82	0.77	0.86	0.83
Extra Large (X)	0.85	0.82	0.82	0.79	0.83	0.80	0.87	0.83



Şekil 4.1. YOLOv8 ile hareket tespiti

4.2. Nesne Tespitine İlişkin Elde Edilen Bulgular

Hareket tespiti ile eş zamanlı çalışacak şekilde kurgulanan nesne tespiti, YOLOv8 algoritması kullanılarak hem nesne tespiti hem de görüntü segmentasyonu yöntemleri ile gerçekleştirilmiştir. Bu kapsamda 640x640 boyutundaki 4 830 adet görüntü sayısı, veri artırma yöntemleri yardımıyla (bkz. Tablo 3.7) 19 320'ye çıkarılmıştır. Söz konusu

görüntüler kullanılarak her iki yöntemle ilişkin YOLOv8'in beş ayrı varyantı ile 100 öğrenme döngüsü (epoch) değerine tabi tutulmuş ve başarı sonuçları kıyaslanmıştır. Modellerin eğitime sürecinde öğrenme döngüsü (epoch) değeri 100'den fazla belirlendiğinde anlamlı bir değişiklik olmadığı gözlemlenmiştir. Karışıklık matrisi ile beraber derin öğrenme modellerinin başarısını değerlendirmek için kullanılan metriklerden olan kesinlik, duyarlılık, F1 skoru ve mAP@0.5 değerleri üzerinden değerlendirme yapılmıştır.

Tablo 4.3. YOLOv8 nesne tespiti varyantlarının kullanımı

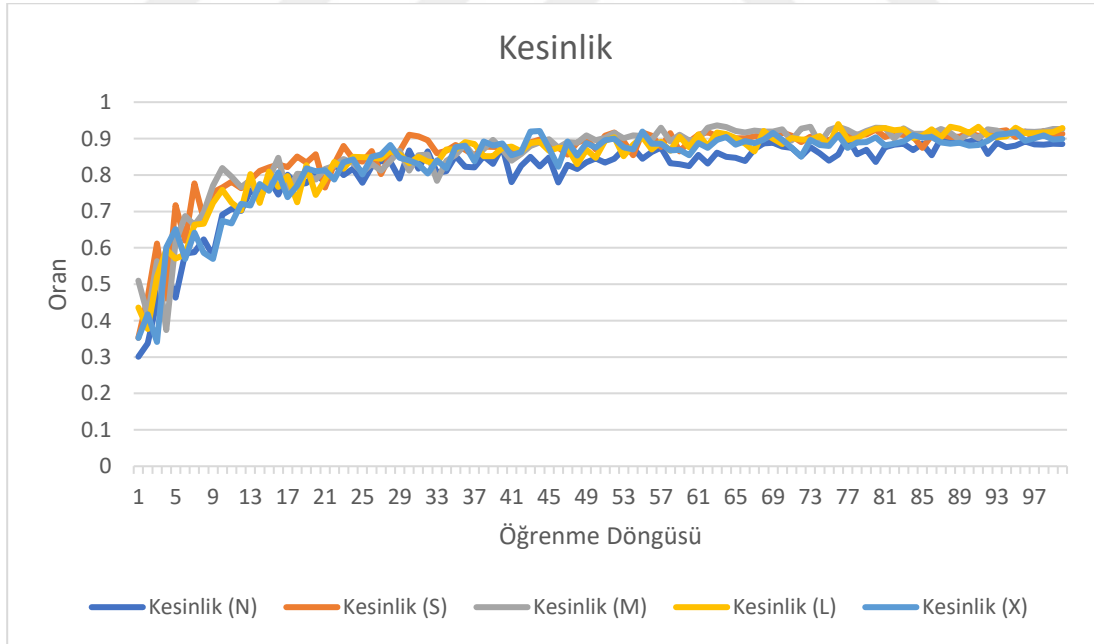
YOLOv8 Varyant İsmi	Katman Sayısı	Parametre Sayısı	Eğitim Süresi (sa.)	Çıkarım Süresi (ms)
Nano (N)	168	3 008 378	2.1	3.8
Small (S)	168	11 131 002	2.26	5.2
Medium (M)	218	25 847 866	3.42	10.5
Large (L)	268	43 617 402	5.3	41.5
Extra Large (X)	295	68 137 050	8.35	26.4

Beklendiği üzere karmaşıklığı ve büyüklüğü artması sebebiyle “*extra large*” varyantı için eğitim ve çıkarım süresi değerleri daha büyüktür (Tablo 4.3). Nesne tespiti için “*medium*” ve “*large*” varyantlarının diğerlerinden çok büyük bir fark ile olmasa da performans anlamında pozitif yönde ayrıştıkları görülmüştür. (Tablo 4.4). Ancak hem işlem süresi hem de doğruluğun belirli bir seviyede sağlanması istendiğinde “*nano*” veya “*small*” varyantlarının kullanımının daha anlamlı olduğu değerlendirilmiştir.

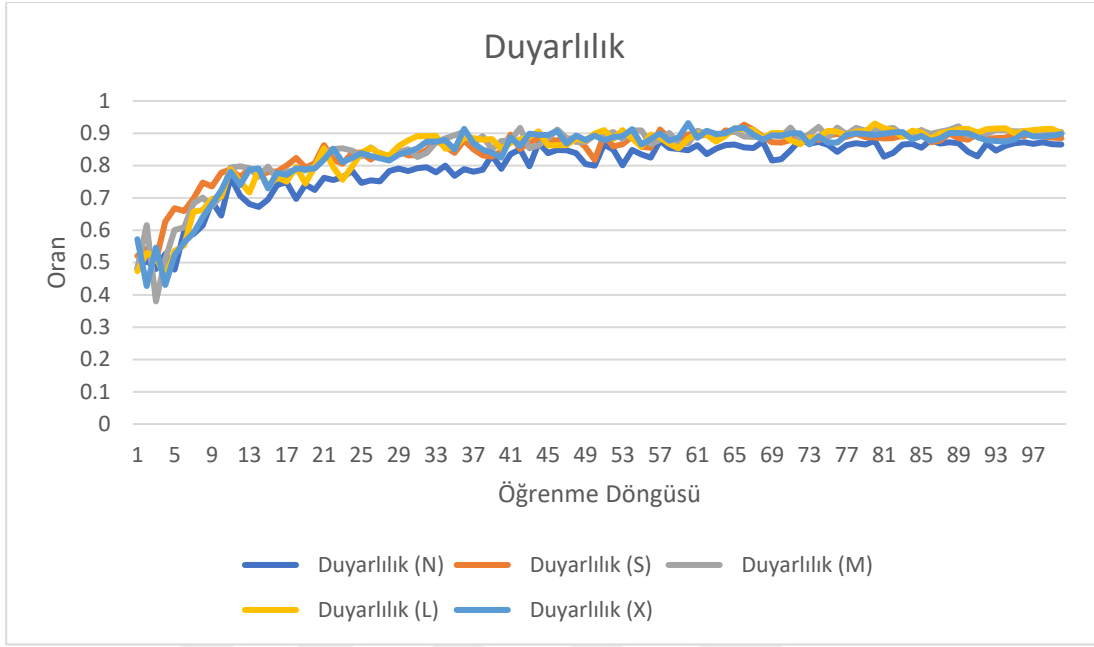
Tablo 4.4. Nesne tespiti için YOLOv8 varyantlarının performansı

YOLOv8 Varyant İsmi	Kesinlik	Duyarlılık	F1	mAP@0.5
Nano (N)	0.89	0.87	0.88	0.93
Small (S)	0.90	0.90	0.90	0.95
Medium (M)	0.92	0.90	0.90	0.96
Large (L)	0.91	0.92	0.91	0.96
Extra Large (X)	0.90	0.89	0.89	0.95

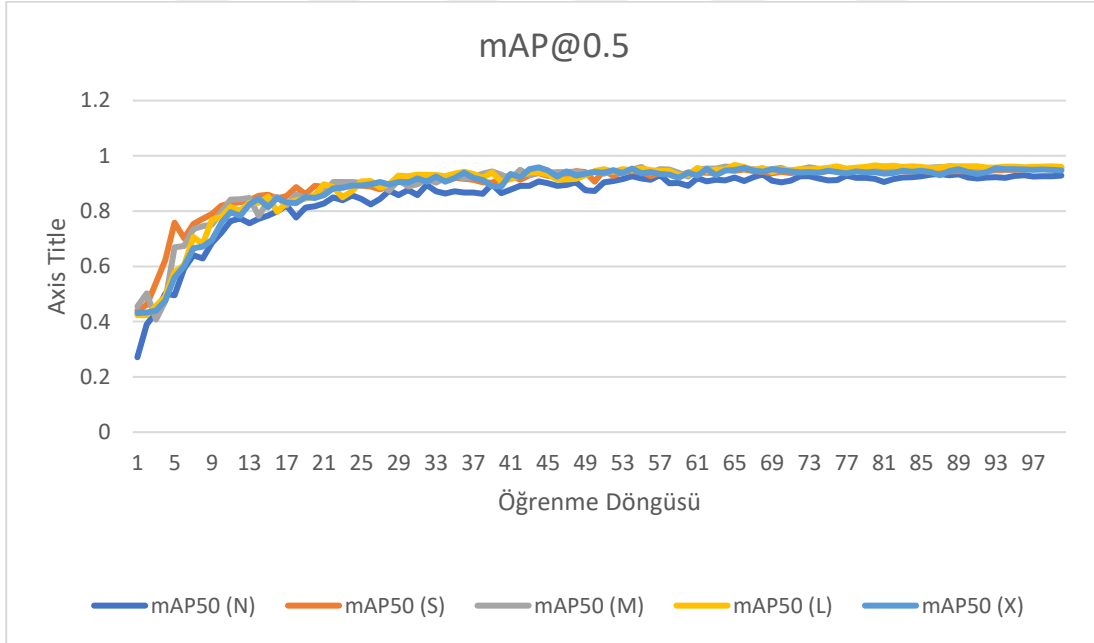
YOLO'nun nesne tespiti görevi için kullanılan “*nano*”, “*small*”, “*medium*”, “*large*” ve “*extra large*” varyantları üzerinden eğitim gerçekleştirildiğinde 100 öğrenme döngüsü (epoch) sürecindeki kesinlik, duyarlılık, F1 ve mAP@0.5 değerlerini gösteren grafikler Şekil 4.2’de yer almaktadır.



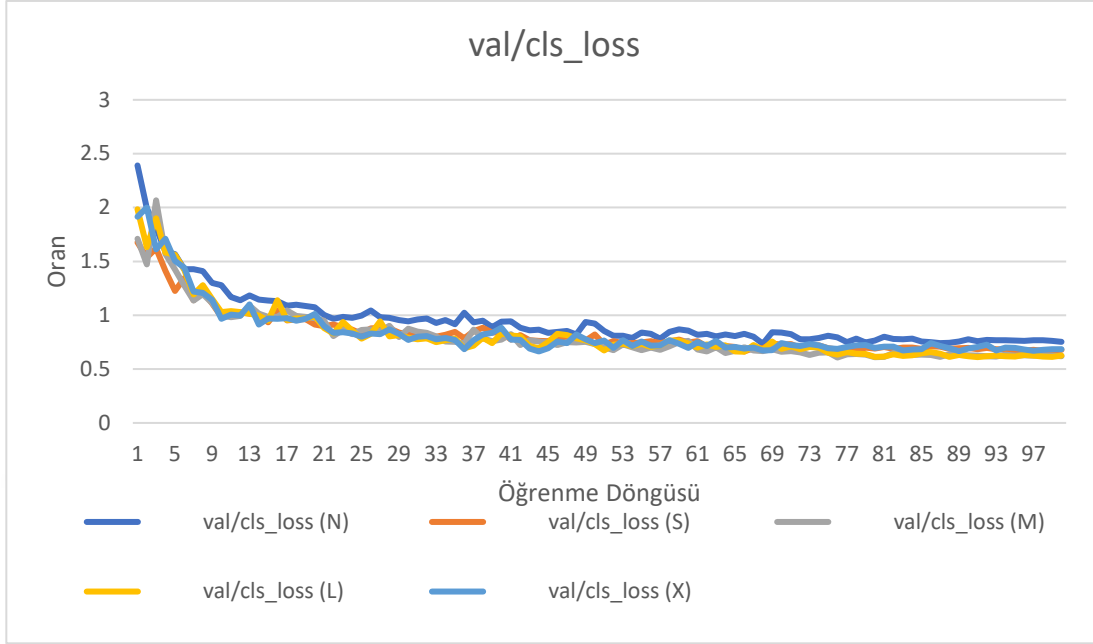
a) Nesne tespiti için kesinlik değeri grafiği



b) Nesne tespiti için duyarlılık grafiği



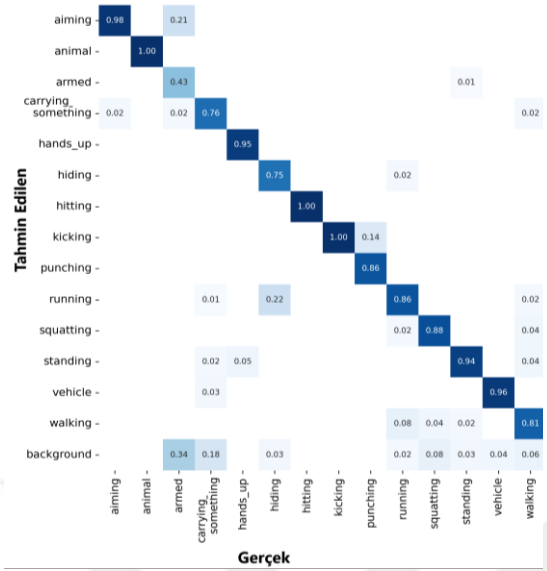
c) Nesne tespiti için mAP@0.5 grafiği



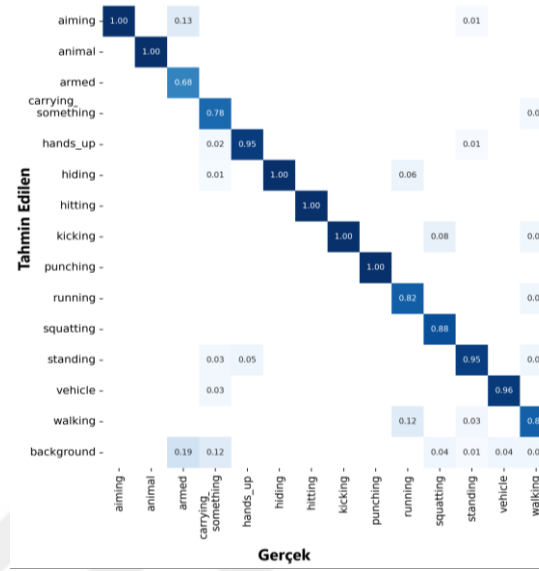
d) Nesne tespiti ilişkin doğrulama kaybı grafiği

Şekil 4.2. Nesne tespiti modelinin öğrenme döngüsü sürecine ait a) kesinlik, b) duyarlılık, c) mAP@0.5, d) doğrulama kaybı grafikleri

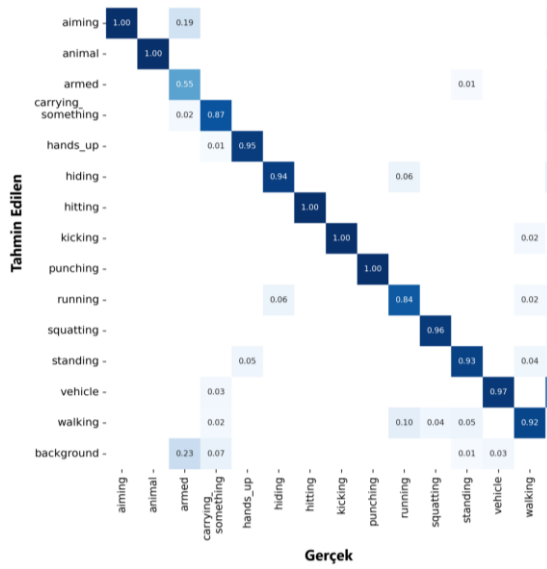
Şekil 4.2’de yar alan grafikler ve YOLOv8 varyantlarına ait Şekil 4.3’teki karışıklık matrisleri incelendiğinde en başarılı varyantın YOLOv8l olduğu görülmüştür. Ayrıca tüm varyantlarda silahlı kişiyi ifade eden “armed” etiketinin en düşük doğru tahmin etme değerine sahip olduğu anlaşılmıştır. Bu sınıfa ait görsellerin en çok silahla nişan alan kimseyi ifade eden “aiming” sınıfı ile karıştırıldığı gözlemlenmiştir. Geliştirilen yazılıma ait çıktı örneklerine ait görseller Şekil 4.4 ve Şekil 4.5’te görülmektedir.



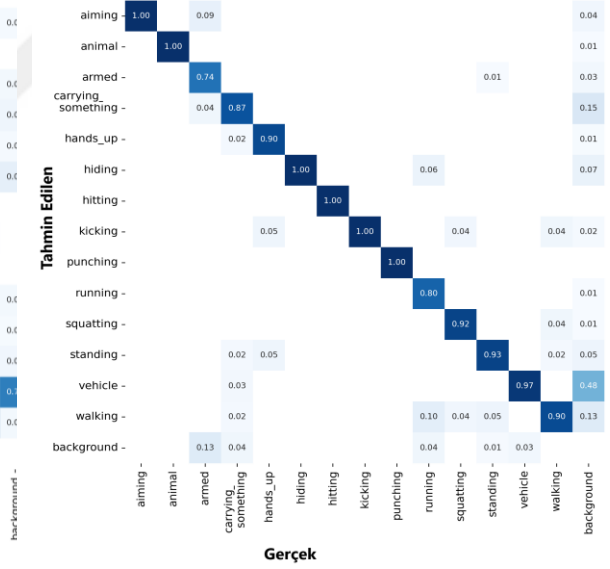
a) YOLOv8n



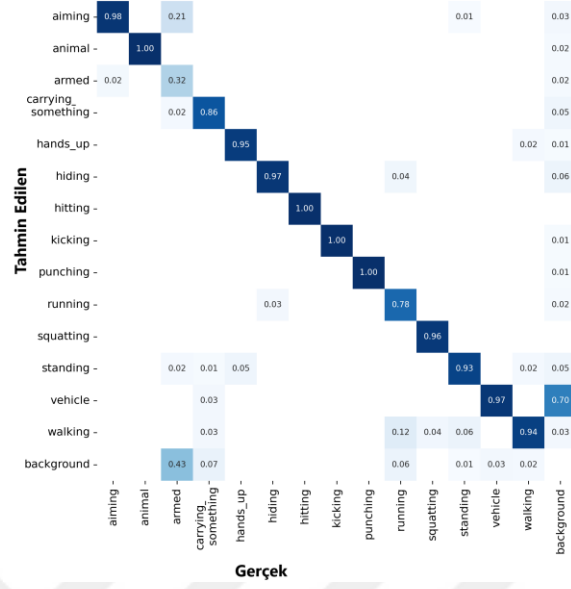
b) YOLOv8s



c) YOLOv8m

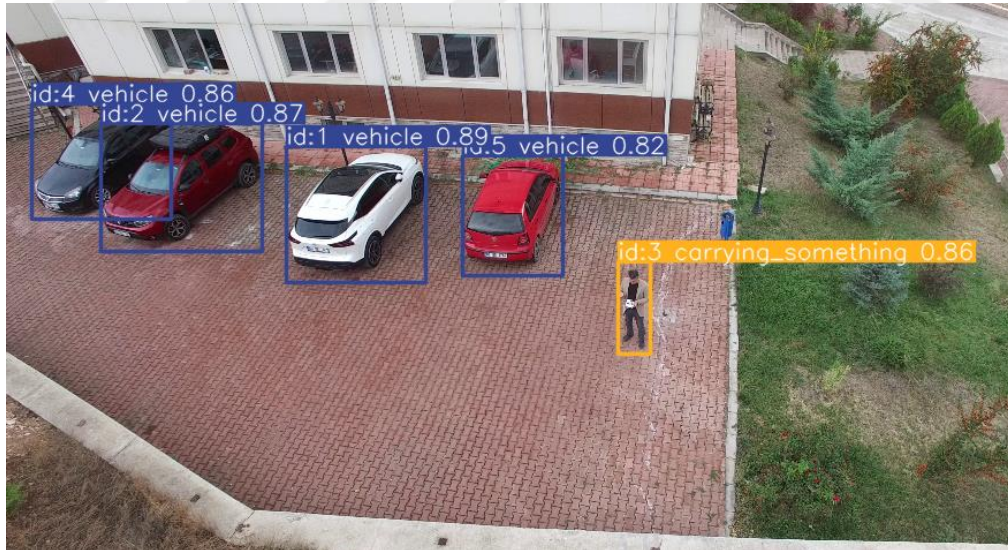


d) YOLOv8l



e) YOLOv8x

Şekil 4.3. YOLOv8 nesne tespiti varyantlarıyla eğitilen modellerin karışıklık matrisleri



Şekil 4.4. Elinde bir şey taşıyan (*carrying_something*) sınıfına ait örnek çıktı



Şekil 4.5. Hedef alan (*aiming*) sınıfına ait örnek çıktı

Nesne tespitinde YOLOv8 görüntü segmentasyonu algoritmasına ait varyantlar kullanıldığında Tablo 4.5 ve Tablo 4.6'daki veriler elde edilmiştir. Tabloda belirtilen *maske (mask)* algılanan nesneye ait piksel grubunu, *kutu (box)* ise algılanan nesnenin etrafındaki sınırlayıcı kutuyu ifade etmektedir (bkz. Şekil 2.14). Tablo 4.6'da yer alan nesne algılama doğruluğu değerlerine bakıldığında, YOLOv8 görüntü segmentasyonu varyantları arasında “*large*” ve “*xlarge*” varyantlarının öne çıktığı görülmüştür. Ancak gerçek zamanlı uygulamalarda hem işlem süresi hem de doğruluk değeri için “*nano*” veya “*small*” varyantlarının kullanımı daha makul bir tercih olmuştur.

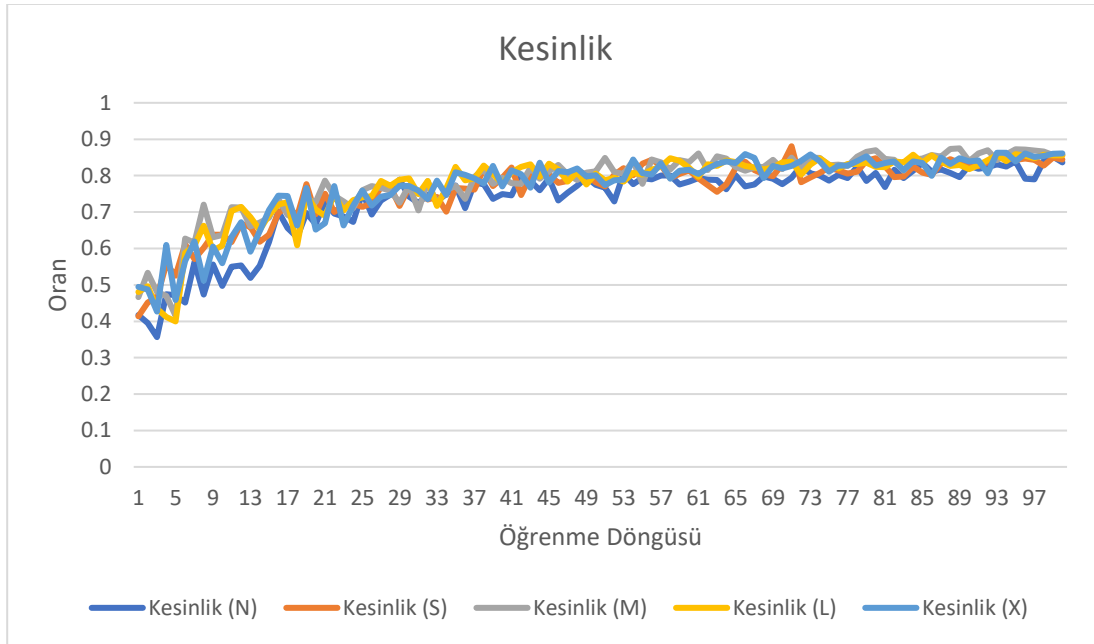
Tablo 4.5. YOLOv8 görüntü segmentasyonu varyantlarının kullanımı

YOLOv8 Varyant İsmi	Katman Sayısı	Parametre Sayısı	Eğitim Süresi (sa.)	Çıkarım Süresi (ms)
Nano (N)	195	3 260 794	3.55	3.3
Small (S)	195	11 785 018	3.71	6.4
Medium (M)	245	27 230 490	5.39	13.4
Large (L)	295	45 922 682	7.87	20.1
Extra Large (X)	295	71 734 138	11.67	32.5

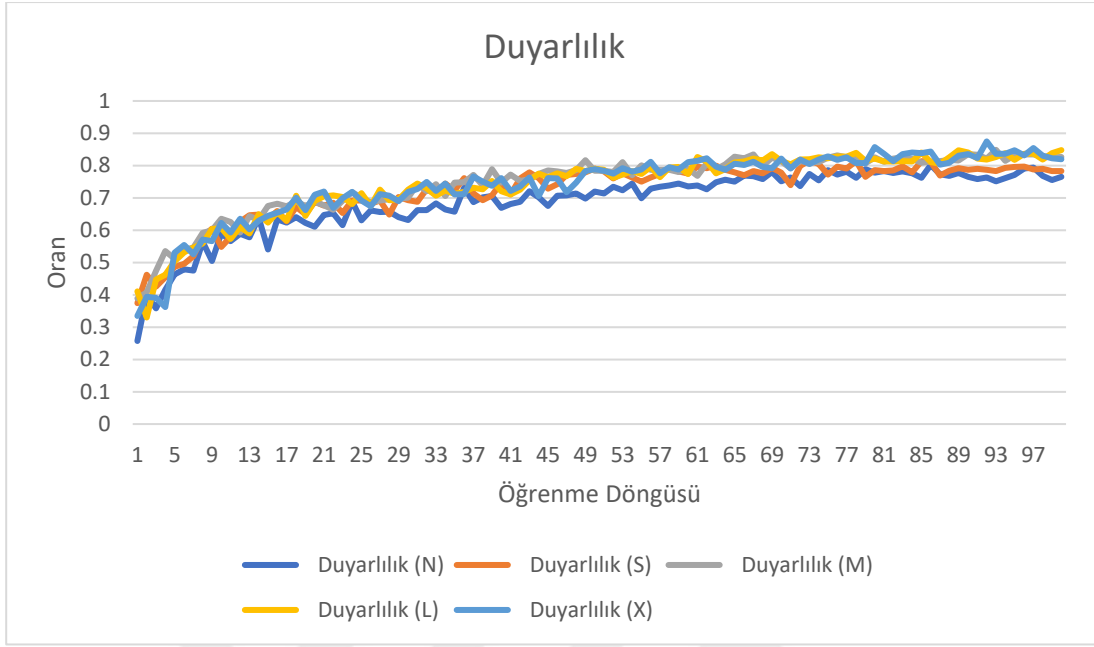
Tablo 4.6. Görüntü segmentasyonu için YOLOv8 varyantlarının performansı

YOLOv8 Varyant İsmi	Kesinlik		Duyarlılık		F1		mAP@0.5	
	Kutu	<i>Maske</i>	Kutu	<i>Maske</i>	Kutu	<i>Maske</i>	Kutu	<i>Maske</i>
Nano (N)	0.83	0.79	0.82	0.77	0.82	0.77	0.87	0.82
Small (S)	0.85	0.85	0.87	0.79	0.85	0.81	0.91	0.84
Medium (M)	0.87	0.87	0.90	0.83	0.88	0.84	0.92	0.87
Large (L)	0.88	0.85	0.88	0.84	0.88	0.84	0.93	0.88
Extra Large (X)	0.89	0.86	0.88	0.84	0.88	0.84	0.93	0.89

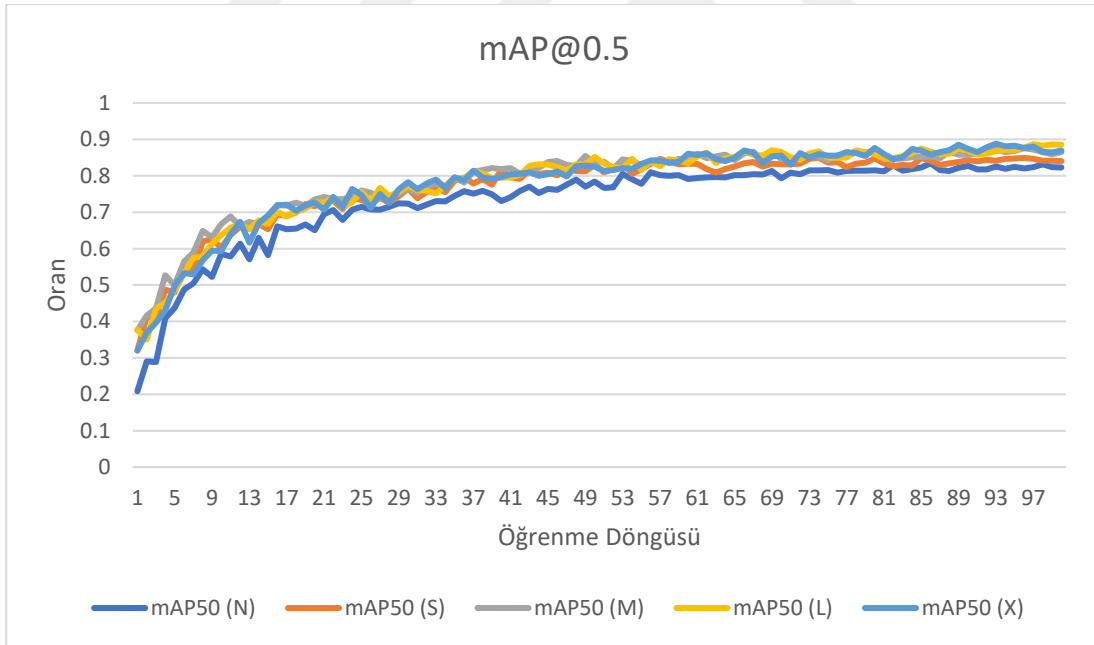
Görüntü segmentasyonu ile nesne tespiti görevi için kullanılan “nano”, “small”, “medium”, “large” ve “extra large” varyantları üzerinden eğitim gerçekleştirildiğinde 100 öğrenme döngüsü (epoch) sürecindeki kesinlik, duyarlılık, F1 ve mAP@0.5 değerlerini gösteren grafikler Şekil 4.6’de yer almaktadır.



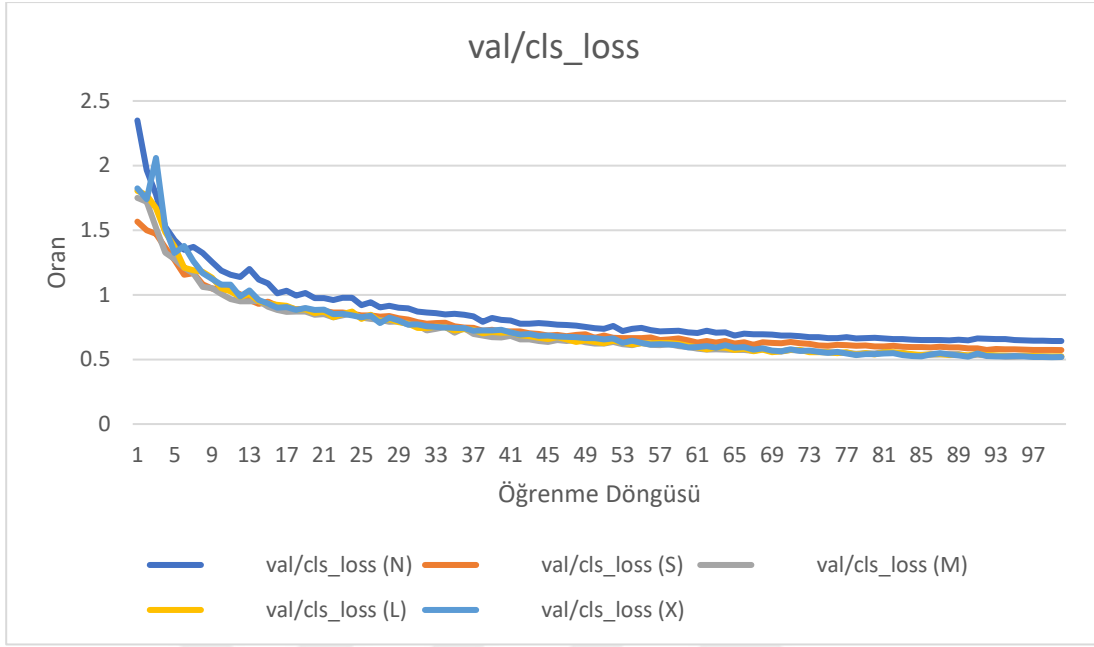
a) Görüntü segmentasyonuna ilişkin kesinlik grafiği



b) Görüntü segmentasyonuna ilişkin duyarlılık grafiği



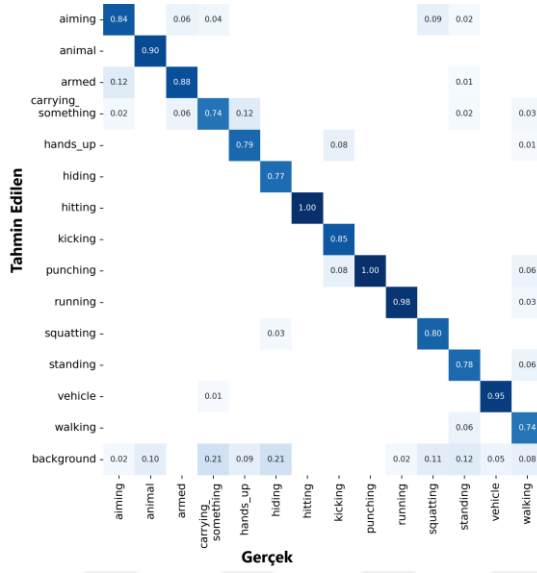
c) Görüntü segmentasyonuna ilişkin mAP@0.5 grafiği



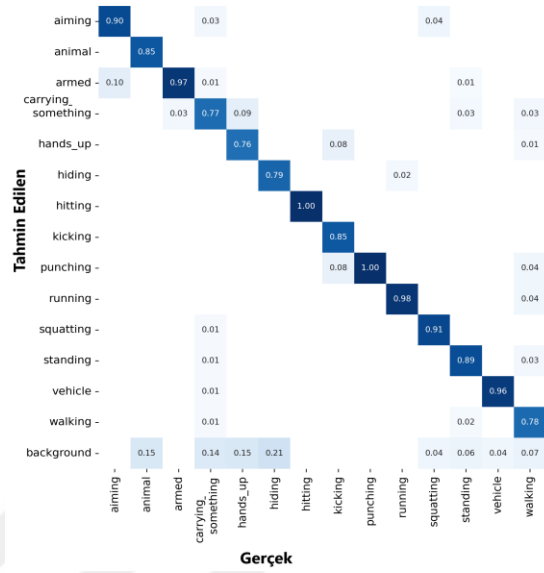
d) Görüntü segmentasyonuna ilişkin doğrulama kaybı grafiği

Şekil 4.6. Çalışmada kullanılan görüntü segmentasyonu modelinin öğrenme döngüsü sürecine ait a) kesinlik, b) duyarlılık, c) mAP@0.5, d) doğrulama kaybı grafikleri

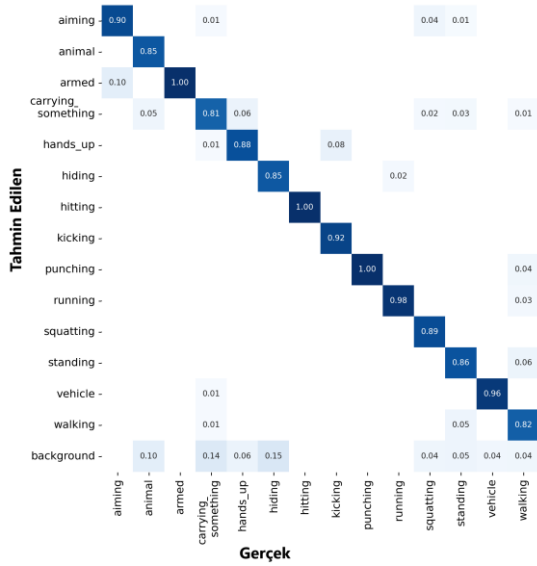
YOLOv8 segmentasyon algoritmasının “*nano*”, “*small*”, “*medium*”, “*large*” ve “*xlarge*” varyantları üzerinden eğitim gerçekleştirildiğinde elde edilen karışıklık matrisi değerleri Şekil 4.5’te gösterilmiştir. Karışıklık matrisleri incelendiğinde tüm varyantlarda elinde bir şey taşıyan kişiyi ifade eden “*carrying_something*” ve yürüyen kişiyi temsil eden “*walking*” sınıflarının en düşük doğru tahmin etme değerine sahip olduğu anlaşılmıştır. Bu etiketin en çok arka planı ifade eden “*background*” sınıfı ile karıştırıldığı gözlemlenmiştir. Geliştirilen yazılıma ait örnek çıktı görselleri Şekil 4.6 ve Şekil 4.7’te görülmektedir.



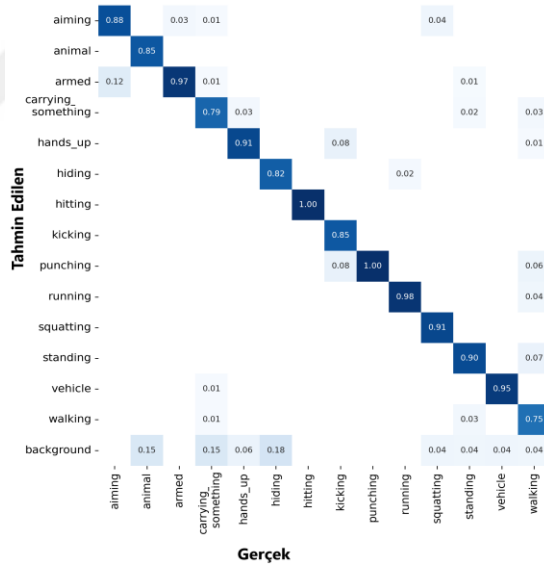
a) YOLOv8n-seg



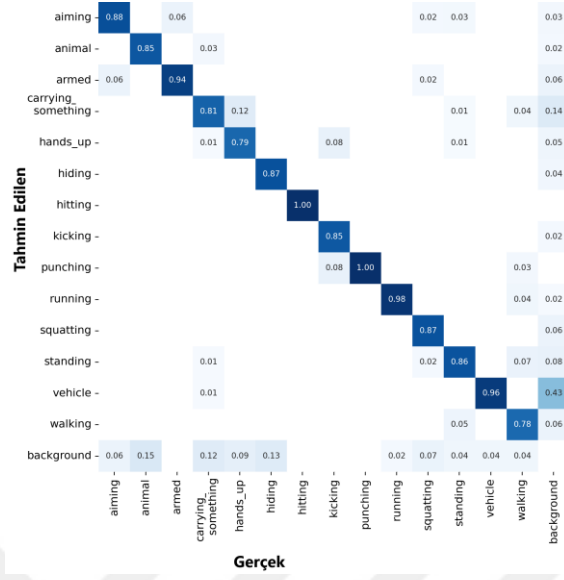
b) YOLOv8s-seg



c) YOLOv8m-seg



d) YOLOv8l-seg

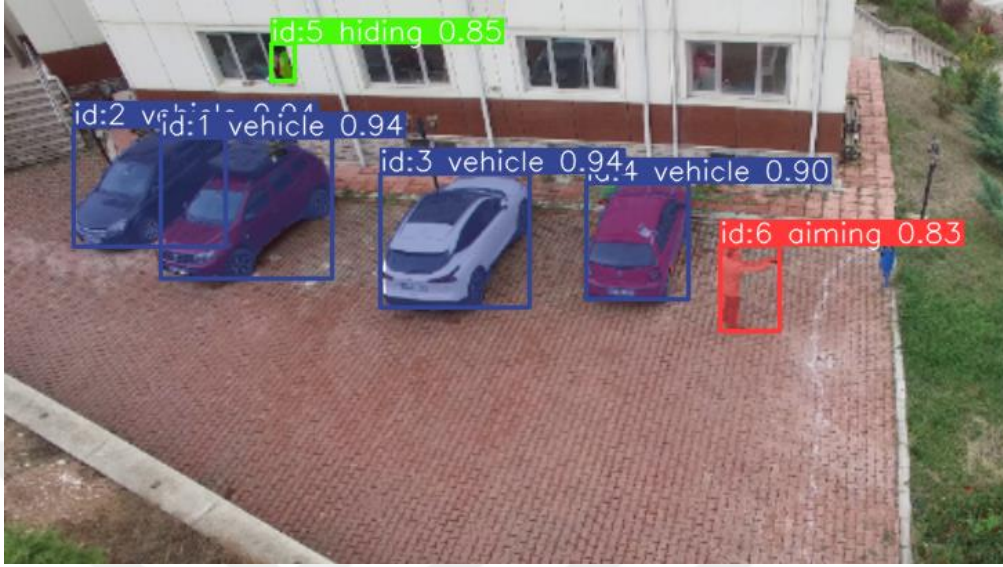


e) YOLOv8x-seg

Şekil 4.7. YOLOv8 görüntü segmentasyonu varyantlarıyla eğitilen modellerin karışıklık matrisleri



Şekil 4.8. Koşan kişiyi ifade eden “*running*” ve ayakta bekleyen kişiyi temsil eden “*standing*” sınıflarına ait örnek çıktı görselleri



Şekil 4.9. Silahla hedef alan kimseyi ifade eden “*aiming*”, gizlenen kimseyi temsil eden “*hiding*” ve araçları ifade eden “*vehicle*” sınıflarına ait örnek çıktı görselleri

4.3. Mesafe Tespitine İlişkin Elde Edilen Bulgular

Geliştirilen yazılımın mesafe ölçüm başarısı; ölçüleri standart olan ve referans alınan mesafe değerlerinin m cinsinden bilindiği futbol sahasında test edilmiştir. Bu kapsamda, 10 m’den 50 m yüksekliğe kadar farklı yüksekliklerde onar adet ölçüm gerçekleştirilmiş ve elde edilen ölçümler gerçek mesafe değerleri ile karşılaştırılmıştır. Futbol sahası referans alınarak gerçekleştirilen ölçüm değerlerinin hata oranı Tablo 4.7’de gösterilmektedir.

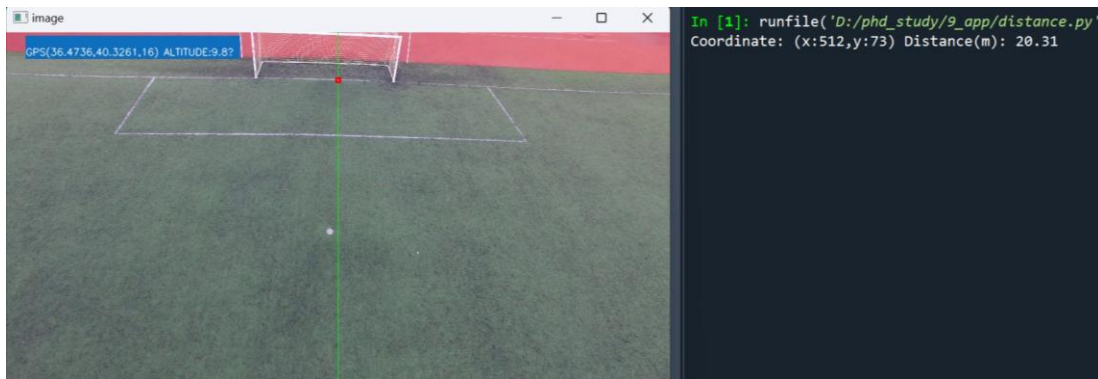
Tablo 4.7. Mesafe tespiti performansı

Yükseklik (m)	Bağıl Hata (Relative Error) (%)	Maksimum Hata (Max Error) (m)	Minimum Hata (Min Error) (m)
10	0.59	1.7	0.04
20	1.45	3.2	0.3
30	1.58	3.19	0.45
40	2.29	4.3	0.2
50	2.43	4.4	0.9
Ortalama	1.66	3.35	0.37

Bağıl hata, gerçek değer ile tahmin edilen değer arasındaki mutlak farkı ifade eden mutlak hatanın (Eşitlik 16) gerçek değere bölünmesiyle elde edilmekte olup yüzde olarak ifade edilmektedir ve birimi yoktur (Eşitlik 17). Tüm yükseklik değerlerinde kıyaslama yapıldığında ölçüm başarısının %1.66 bağıl hata ortalaması ile gerçekleştiği gözlemlenmiştir. Ölçümler arazi zeminine ait eğim açısının 0° olduğu varsayılarak gerçekleştirilmiştir.

$$\text{MUTLAK_HATA}_x = |X_{\text{Gerçek}} - X_{\text{Ölçülen}}| \quad (16)$$

$$\text{BAĞIL_HATA}_x = \frac{\text{MUTLAK_HATA}_x}{X_{\text{Gerçek}}} \quad (17)$$



Şekil 4.10. Futbol sahasında mesafe ölçümü testi

Futbol sahasında 9.8 m'den test işlemi için gerçekleştirilen örnek mesafe ölçümünde; gerçek değeri 20.15 m olan mesafeyi, yazılımın 20.31 m olarak ölçtüğünü gösteren ekran görüntüsü Şekil 4.10'da yer almaktadır. Uygulama içinde kullanılan mesafe ölçümüne ait görsele Şekil 4.11'de yer verilmiştir.



Şekil 4.11. Mesafe tespiti için örnek çıktı

5. SONUÇ VE ÖNERİLER

Bu tez çalışmasında güvenlik olgusunun her geçen gün önem kazandığı günümüze yönelik, insandan bağımsız ve onu tehlikeye atmadan “*drone*” ile istenilen alanı otomatize bir şekilde gözetleyen bir sistem geliştirilmeye çalışılmıştır. Bu kapsamda yapılmak istenilen erken uyarı sistemi için hibrit bir model önerilmiştir. Modelde “*drone*” üzerine monte edilmiş kameradan elde edilen video görüntüleri kullanılmıştır. Görüntüler üzerinden öncelikle hareket tespiti ve nesne tespiti yapılmaya çalışılmıştır. Tespitin gerçekleşmesi durumunda söz konusu nesnenin drona olan uzaklığı ve tehlike arz edip etmediği değerlendirilmiştir. Böylece herhangi bir insana ihtiyaç duyulmadan nöbet tutma görevini üstlenebilecek, hareket tespiti ve bu hareketin insana ait olması durumunda ise söz konusu hareketin analizi ile şüpheli durum ve saldırı tespiti yapabilen bir güvenlik uygulaması geliştirilmeye çalışılmıştır,

Saldırı veya şüpheli olarak değerlendirebilmek için öncelikle “*drone*” üzerinde yer alan kameradan hareketli nesne tespiti yapılmaktadır. Optik akış algoritması kullanılarak ardışık kareler arasında benzer özellikler tespit edilmiş ve bu özellik noktaları referans alınarak görüntüdeki arka planın ayrıştırılması için kareler birbirlerine çarpıtılmıştır. Elde edilen arka planı dengelenmiş ardışık karelerin normalizasyonu sağlanarak farkı alınmış ve hareketli nesne ön plana çıkarılmıştır. Ancak hareketli kameradan hareket tespit edilirken; yaprak, çimen gibi arka plana ait hareketli nesnelerin ayrıştırılması zorlu bir görevdir. Bu zorluğu aşmak için bir derin öğrenme algoritması olan YOLOv8 algoritması kullanılmıştır. YOLOv8 ile karelerin farkından oluşan görüntülerin segmentasyonu sağlanmaya çalışılmıştır. Böylece hareketli nesneye ait görüntünün ilgili kısımları piksel bazında tespit edilebilmiştir. Sistemin gözetleme görevi için hareket tespiti ilk adımdır. Bu işlemlere paralel olarak aynı anda nesne tespiti ve “*drone*” ile arasındaki mesafenin ölçümü de yapılmaya çalışılmıştır. Belirlenen nesnenin insana ait olması durumunda insan hareketlerinin tanımlanması gerçekleştirilmiştir. Bu çerçevede daha çok biyomedikal alanında kullanılan U-Net’in benzer bir mimarisi ile oluşturulmuş YOLOv8 segmentasyon

algoritmasının varyantları ile nesne tespit modelleri ayrı ayrı kullanılmıştır. Her bir modele ait beş ayrı varyantın performans sonuçları incelenmiş ve kıyaslanmıştır. Geliştirilen yazılım ile; nesne tespitinde %96 mAP@0.5, görüntü segmentasyonu ile nesne tespitinde %89 mAP@0.5 ve hareket tespitinde ise %89 mAP@0.5 değerleri elde edilmiştir. Mesafe tespiti ise %1.66 bağıl hata ortalaması ile gerçekleştirilmiştir. Geliştirilen yazılımın kullanıldığı bilgisayar üzerinde ~14 fps değerine ulaşılmıştır.

Video görüntülerinden çıkarımı yapılan nesnelerin “*drone*” ile olan mesafesi ve insana ait aksiyonların skorlanması sağlanarak olası saldırı tespiti yapılmaya çalışılmıştır.

Tez çalışması kapsamında elde edilen sonuçlar ışığında, nesnenin doğru tespiti anlamında YOLOv8 “*large*” varyantı ile geliştirilmiş yazılımın diğer varyantlar ile eğitilmiş modeller arasında en başarılı varyant olarak öne çıktığı görülmüştür (bkz. Tablo 4.4). Görüntü segmentasyonu yöntemiyle nesne tespitinde ise YOLOv8 “*large*” ve “*extra large*” varyantları ile geliştirilmiş yazılımın diğer varyantlar ile eğitilmiş modeller arasında en başarılı olanlar olduğu görülmüştür (bkz. Tablo 4.5). Hareket tespitinde ise YOLOv8 “*medium*” varyantı ile geliştirilmiş yazılımın diğer varyantlar ile eğitilmiş modeller arasında en başarılı sonucu ürettiği gözlemlenmiştir (bkz. Tablo 4.2). YOLO algoritmasına ait varyantlar arasından, yürütülen çalışmanın amacına göre seçim yapılmasının ihtiyatlı bir yaklaşım olacağı değerlendirilmiştir.

Görüntü segmentasyonu ile yapılan çalışmada daha hassas tespitlerin yapılabilirdi ancak nesne tespitinde model eğitimi için kullanılan veriden daha çok görsele ihtiyaç duyulduğu anlaşılmıştır.

Çalışma neticesinde elde edilen diğer sonuçlar şu şekildedir;

- Örnek segmentasyonu; etiketlemesi daha zahmetli bir görevdir ve doğru örnek segmentasyonu modellerini eğitmek daha zordur. Bu sebepten ötürü

segmentasyon modelleri, genellikle daha büyük ve daha yavaştır. Ancak daha hassas tespitler yapılmasına olanak sağlamaktadır.

- Örnek segmentasyon modelleri, nesne algılama modelleriyle aynı doğruluğu elde etmek için daha büyük veri kümelerine ihtiyaç duyabilmektedir,
- “Drone” üzerindeki kameranın zemin ile arasındaki açının 60° fazla olması insan hareketlerini tespit ve analiz etmeyi imkansız hale getirmektedir,
- “Drone”nun yerden yüksekliği ve nesnenin kameraya olan uzaklığı arttıkça ilgili nesneye ait özellikler daha az piksel ile temsil edileceğinden nesne tespiti ve analizi güçleşmektedir,
- Derin öğrenme uygulamalarında veri setinin büyüklüğü ve çeşitliliği öğrenme için en önemli faktör olmasından ötürü daha geniş bir veri seti ile çalışıldığında öğrenme de o oranda iyi olacaktır. Bununla birlikte veri seti büyüklüğü öğrenme için harcanan zamanı ve öğrenme sonunda elde edilen modelin büyüklüğünü de o oranda artırmaktadır. Çok sık eğitilmeyecek ve depolama alanı sorun etmeyecek projelerde öğrenim başarısına karşı zaman ve depolama alanındaki bu kayıplar göz ardı edilebilmektedir. Bununla birlikte sık sık eğitilmesi gereken veya mobil ortamlar gibi depolama alanı problem oluşturacak ortamlarda çalışacak projeler geliştiriliyorsa başarının yanı sıra veri setinin büyüklüğü de değerlendirmeye alınmalıdır.

Sonraki araştırmalar için;

- Yapılacak çalışmalarda çözünürlüğü daha yüksek kamera ile çalışılarak, daha yüksek irtifada daha detaylı ve hassas tespitler yapılabileceği,
- İnsan hareketlerinden çıkarım yapılacak saldırı tespitinde, yüksek çözünürlükteki veri ile geleceğe yönelik saldırı tahminlemesi yapılabileceği,
- Kamera hareketlerinin programlanabildiği bir kamera kullanılarak hedefin kamera ile takip edilmesi ve görüntünün nesneye yaklaşması sağlanabileceği,

- Sisli, yağmurlu vs. gibi farklı hava ve arazi koşullarında elde edilen görüntü örnekleri kullanılarak eğitilecek model ile daha geniş sayıda nesne ve durum tespiti yapılabileceği,
- Gece görüşü ve termal kamera görüntüleri ile zorlu koşullarda daha hassas tespitlerde bulunabileceği,
- Tespiti yapılan nesnenin “*drone*” olana uzaklığının yanı sıra konum bilgisinin de hesaplanabileceği,
- Donanım özellikleri daha yüksek bir bilgisayar kullanıldığında yüksek fps değerleri elde edilebileceği

öngörülmektedir.

6. KAYNAKLAR

- Ağdaş, M. T. ve Gülseçen, S., 2022. Güvenlik Kameralarında Otomatik Silah ve Bıçak Tespit Sistemi:Karşılaştırmalı YOLO Modelleri. Avrupa Bilim ve Teknoloji Dergisi Sayı 41, S. 16-22, Kasım 2022. doi: 10.31590/ejosat.1163675
- Aitfares, W., Kobbane, A. ve Kriouile, A., 2016. September. Suspicious behavior detection of people by monitoring camera. In 2016 5th International Conference on Multimedia Computing and Systems (ICMCS) (pp. 113-117). IEEE.
- Akarun, L., 2020. Üç boyutlu görme ve derinlik kameraları. [https://sarkac.org/2020/09/uc-boyutlu-gorme-ve-derinlik-kameralari/-](https://sarkac.org/2020/09/uc-boyutlu-gorme-ve-derinlik-kameralari/) (25.10.2021).
- Anonim, 2009. UCF Aerial Action Data Set. University of Central Florida. http://csrc.ucf.edu/data/UCF_Aerial_Action.php- (19.09.2021).
- Anonim, 2015. How to Use Background Subtraction Methods.<http://docs.opencv.org/>-(03.11.2022).
- Anonim, 2020. How to create a panorama image using OpenCV with Python. Datahacker. [https://datahacker.rs/005-how-to-create-a-panorama-image-using-opencv-with-python/-](https://datahacker.rs/005-how-to-create-a-panorama-image-using-opencv-with-python/) (18.12.2021).
- Arreola, L., Gudino, G. ve Flores, G., 2019. Object recognition and tracking using Haar-like Features Cascade Classifiers: Application to a quad-rotor UAV. arXiv preprint arXiv:1903.03947.
- Austin, R., 2011. Unmanned aircraft systems: UAVS design, development and deployment. John Wiley & Sons.

- Bordoloi, N., Talukdar, A.K. ve Sarma, K.K., 2020. December. Suspicious Activity Detection from Videos using YOLOv3. In 2020 IEEE 17th India Council International Conference (INDICON) (pp. 1-5). IEEE.
- Boykov, Y., Veksler, O. ve Zabih, R., 2001. Fast approximate energy minimization via graph cuts. IEEE Transactions on pattern analysis and machine intelligence, 23(11), 1222-1239.
- Brownlee, J., 2019. A Gentle Introduction to Object Recognition With Deep Learning. <https://machinelearningmastery.com/object-recognition-with-deep-learning/>- (20.10.2021).
- Chapel, M. N. ve Bouwmans, T., 2020. Moving objects detection with a moving camera: A comprehensive review. Computer science review, 38, 100310.
- Chistyakov, A., Lobacheva, E., Shevelev, A. ve Romanenko, A., 2018. Monotonic models for real-time dynamic malware detection. arXiv preprint arXiv:1804.03643.
- Chriki, A., Touati, H., Snoussi, H. ve Kamoun, F., 2020. Uav-based surveillance system: an anomaly detection approach. In 2020 IEEE Symposium on computers and communications (ISCC) (pp. 1-6). IEEE.
- Das, D. ve Saharia, S., 2014. Implementation And Performance Evaluation Of Background Subtraction Algorithms. International Journal on Computational Sciences & Applications (IJCSA) Vol.4, No.2, April 2014.
- Delibasoglu, I., 2021. UAV images dataset for moving object detection from moving cameras. arXiv preprint arXiv:2103.11460.

- Dhanachandra, N., Manglem, K. ve Chanu, Y. J., 2015. Image segmentation using K-means clustering algorithm and subtractive clustering algorithm. *Procedia Computer Science*, 54, 764-771.
- Dobrea, D. M. ve Dobrea, M. C., 2020. An autonomous UAV system for video monitoring of the quarantine zones. *Roman J Inf Sci Technol*, 23, S53.
- Eigen, D., Puhrsch, C. ve Fergus, R., 2014. Depth map prediction from a single image using a multi-scale deep network. *arXiv preprint arXiv:1406.2283*.
- Elgendy, M., 2020. *Deep Learning for Vision Systems*. Manning Publications Co., 284, Amerika Birleşik Devletleri.
- Elharrouss, O., Almaadeed, N. ve Al-Maadeed, S., 2021. A review of video surveillance systems. *Journal of Visual Communication and Image Representation*, 77, 103116.
- Elqursh, A. ve Elgammal, A., 2012. Online moving camera background subtraction. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12* (pp. 228-241). Springer Berlin Heidelberg.
- Fang, M. T., Chen, Z. J., Przystupa, K., Li, T., Majka, M. ve Kochan, O., 2021. Examination of abnormal behavior detection based on improved YOLOv3. *Electronics*, 10(2), 197.
- Farabet, C., Couprie, C., Najman, L. ve LeCun, Y., 2012. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(8), pp.1915-1929.

- Fischer, P., Dosovitskiy, A., Ilg, E., Häusser, P., Hazırbaş, C., Golkov, V., Van der Smagt, P., Cremers, D. ve Brox, T., 2015. FlowNet: Learning optical flow with convolutional networks. arXiv preprint arXiv:1504.06852.
- Ganin, Y. ve Lempitsky, V., 2014. November. N^4 -Fields: Neural Network Nearest Neighbor Fields for Image Transforms. In Asian Conference on Computer Vision (pp. 536-551). Springer, Cham.
- Ghode, N. ve Bhagat, P. H., 2019. Motion Detection Using Continuous Frame Difference and Contour Based Tracking. Proceedings of the Third International Conference on Trends in Electronics and Informatics (ICOEI 2019) IEEE Xplore Part Number: CFP19J32-ART; ISBN: 978-1-5386-9439-8.
- Girshick, R., 2015. Fast r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 1440-1448).
- Girshick, R., Donahue, J., Darrell, T. ve Malik, J., 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 580-587).
- Girshick, R., Donahue, J., Darrell, T. ve Malik, J., 2015. Region-based convolutional networks for accurate object detection and segmentation. IEEE transactions on pattern analysis and machine intelligence, 38(1), 142-158.
- Gupta, L., Jain, R. ve Vaszkun, G., 2015. Survey of important issues in UAV communication networks. IEEE communications surveys & tutorials, 18(2), 1123-1152.

- Hanbay, K. ve Üzen, H., 2017. Nesne tespit ve takip metotları: Kapsamlı bir derleme. Türk Doğa ve Fen Dergisi, 6(2), 40-49.
- Hantao, L., 2005. Distance Determination from Pairs of Images from Low CostCameras. August, 1–95.
- Hariharan, B., Arbeláez, P., Girshick, R. ve Malik, J., 2015. Hypercolumns for object segmentation and fine-grained localization. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 447-456).
- Hayat, S., Yanmaz, E. ve Muzaffar, R., 2016. Survey on unmanned aerial vehicle networks for civil applications: A communications viewpoint. IEEE Communications Surveys & Tutorials, 18(4), 2624-2661.
- He, K., Gkioxari, G., Dollár, P. ve Girshick, R., 2017. Mask r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 2961-2969).
- Jain, H., Vikram, A., Kashyap, A. ve Jain, A., 2020. Weapon detection using artificial intelligence and deep learning for security applications. In 2020 International conference on electronics and sustainable communication systems (ICESC) (pp. 193-198). IEEE.
- Jocher, G., Chaurasia, A., Qiu, J. ve Akyon, F. C., 2023. YOLOv8 Overview. Ultralytics. <https://docs.ultralytics.com/models/yolov8/>-(12.12.2023).
- Kahveci, M. ve Nazlı, C. A. N., 2017. İnsansız Hava Araçları: Tarihçesi, Tanımı, Dünyada ve Türkiye'deki Yasal Durumu. Selçuk Üniversitesi Mühendislik, Bilim ve Teknoloji Dergisi, 5(4), 511-535.

- Kass, M., Witkin, A. ve Terzopoulos, D., 1988. Snakes: Active contour models. International journal of computer vision, 1(4), 321-331.
- Keçeli, A. S. ve Kaya, A., 2018. Optik Akış Görüntüsü ve Bi-Lstm ile Şiddet İçeren Hareketlerin Sınıflandırılması. Avrupa Bilim ve Teknoloji Dergisi, (14), 204-208. DOI: 10.31590/ejosat.460257.
- Kızrak, M., 2020. Makale İncelemesi: Hareket Tanıma için Gösterim Akış Yaklaşımı. [https://ayyucekizrak.medium.com/makale-inceleme-hareket-tanima-için-gösterim-akş-84e2d20f2079-\(22.03.2020\).](https://ayyucekizrak.medium.com/makale-inceleme-hareket-tanima-için-gösterim-akş-84e2d20f2079-(22.03.2020).)
- Krizhevsky, A., Sutskever, I. ve Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 25, pp.1097-1105.
- Kuklin M., 2021a. Optical Flow in OpenCV (C++/Python). <https://learnopencv.com/optical-flow-in-opencv/>-(10.10.2021).
- Kuklin M., 2021b. RAFT: Optical Flow estimation using Deep Learning. <https://learnopencv.com/optical-flow-using-deep-learning-raft/#flownet-> (10.07.2021).
- Krizhevsky, S., Jyoti, N., Pyati, S. ve Dewan, J., 2018. August. Abnormal crowd behavior detection using image processing. In 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA) (pp. 1-5). IEEE.
- LeCun, Y., Bengio, Y. ve Hinton, G., 2015. Deep learning. nature, 521(7553), 436-444.

- Lee, J. S., Baek, J. Y., Jung, D., Shim, J. S., Lim, H. S. ve Jo, Y. H., 2019. Estimate of coastal water depth based on aerial photographs using a low-altitude remote sensing system. *Ocean Science Journal*, 54, 349-362.
- Lin, C., 2019. Introduction to Motion Estimation with Optical Flow. <https://nanonets.com/blog/optical-flow/>- (11.08.2021).
- Lin, T. Y., Goyal, P., Girshick, R., He, K. Ve Dollár, P., 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 2980-2988).
- Liu, C., Szirányi, T., 2021. Real-Time Human Detection and Gesture Recognition for On-Board UAV Rescue. *Sensors* 2021, 21, 2180. <https://doi.org/10.3390/s21062180>
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y. ve Berg, A.C., 2016, October. Ssd: Single shot multibox detector. In *European conference on computer vision* (pp. 21-37). Springer, Cham.
- Liu, J., Gu, Y. ve Kamijo, S., 2015. Customer behavior recognition in retail store from surveillance camera. In *2015 IEEE international symposium on multimedia (ISM)* (pp. 154-159). IEEE.
- Maslow, A. H., 1943. A theory of human motivation. *Psychological Review*, 50(4), 370–396. <https://doi.org/10.1037/h0054346>
- Mehmood, A., 2021. LightAnomalyNet: A Lightweight Framework for Efficient Abnormal Behavior Detection. *Sensors*, 21(24), 8501.

- Mekdad, Y., Aris, A., Babun, L., El Fergougui, A., Conti, M., Lazzeretti, R. ve Uluagac, A. S., 2023. A survey on security and privacy issues of UAVs. *Computer Networks*, 224, 109626.
- Minaee, S. ve Wang, Y., 2019. An ADMM approach to masked signal decomposition using subspace representation. *IEEE Transactions on Image Processing*, 28(7), 3192-3204.
- Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N. ve Terzopoulos, D., 2021. Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(7), 3523-3542.
- Motlagh, N. H., Taleb, T. ve Arouk, O., 2016. Low-altitude unmanned aerial vehicles-based internet of things services: Comprehensive survey and future perspectives. *IEEE Internet of Things Journal*, 3(6), 899-922.
- Najman, L. ve Schmitt, M., 1994. Watershed of a continuous function. *Signal Processing*, 38(1), 99-112.
- Nock, R. ve Nielsen, F., 2004. Statistical region merging. *IEEE Transactions on pattern analysis and machine intelligence*, 26(11), 1452-1458.
- Oh, S., Hoogs, A., Perera, A., Cuntoor, N., Chen, C. C., Lee, J. T. ve Desai, M., 2011. A large-scale benchmark dataset for event recognition in surveillance video. In *CVPR 2011* (pp. 3153-3160). IEEE.
- Otsu, N., 1979. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1), 62-66.

- Perera, A. G., Al-Naji, A., Law, Y. W. ve Chahl, J., 2018. Human detection and motion analysis from a quadrotor UAV. In IOP conference series: materials science and engineering (Vol. 405, No. 1, p. 012003). IOP Publishing.
- Redmon, J., Divvala, S., Girshick, R. ve Farhadi, A., 2016. You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
- Remanan, S., 2019. Beginner's Guide to Object Detection Algorithms. <https://medium.com/analytics-vidhya/beginners-guide-to-object-detection-algorithms-6620fb31c375-> (15.10.2021).
- Ren, S., He, K., Girshick, R. ve Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. arXiv 2015. arXiv preprint arXiv:1506.01497.
- Phyo, C. N., Zin, T. T. ve Tin, P., 2019. Deep learning for recognizing human activities using motions of skeletal joints. IEEE Transactions on Consumer Electronics, 65(2), 243-252.
- Sambolek, S. ve Ivašić-Kos, M., 2023. Determining the Geolocation of a Person Detected in an Image Taken with a Drone. Computer Vision and Image Understanding.
- Sincan, O. M., Keles, H. Y. ve Tosun, S., 2018. Moving object detection and classification in surveillance systems using moving cameras. Communications Faculty of Sciences University of Ankara Series A2-A3 Physical Sciences and Engineering, 60(2), 63-82.

- Soares, P.G., Silva, A. ve Pereira, L.F., 2019. An assault detection system based on human Pose Tracking for video surveillance. *Anais Estendidos da Conference on Graphics, Patterns and Images (SIBGRAPI)*.
- Stojnić, V., Risojević, V., Muštra, M., Jovanović, V., Filipi, J., Kezić, N. ve Babić, Z., 2021. A method for detection of small moving objects in UAV videos. *Remote Sensing*, 13(4), 653.
- Srivastava, A., Badal, T., Saxena, P., Vidyarthi, A. ve Singh, R., 2022. UAV surveillance for violence detection and individual identification. *Automated Software Engineering*, 29(1), 28.
- Tan, D., 2020. Depth Estimation: Basics and Intuition. <https://towardsdatascience.com/depth-estimation-1-basics-and-intuition-86f2c9538cd1-> (21.05.2021).
- Tan, M., Pang, R. ve Le, Q. V., 2020. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10781-10790).
- Teed, Z. ve Deng, J., 2020. August. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision* (pp. 402-419). Springer, Cham.
- Terven, J., Córdova-Esparza, D.-M., Romero-González, J.-A., 2023. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Mach. Learn. Knowl. Extr.* 2023, 5, 1680-1716. <https://doi.org/10.3390/make5040083>.

- TDK, 2020. TDK (Türk Dil Kurumu) güncel Türkçe sözlük içinde. 9 Eylül 2020. <https://sozluk.gov.tr/>.
- Tsiktsiris, D., Dimitriou, N., Lalas, A., Dasygenis, M., Votis, K. ve Tzovaras, D., 2020. Real-time abnormal event detection for enhanced security in autonomous shuttles mobility infrastructures. *Sensors*, 20(17), 4943.
- Verma, S., Soni, A., Mishra, V., Gupta, V., Krishnamurthi, R. ve Kumar, A., 2022. Smart automated system for classification of emergency heavy vehicles and traffic light controlling. In *Autonomous and Connected Heavy Vehicle Technology* (pp. 245-262). Academic Press.
- Yang, G., Zhang, J., Zhu, L., Huan, W., Lin, G. ve Ma, Z., 2019. UAV-Based Motion Target Detection and Tracking Method in Dynamic Scenes. In *2019 IEEE Fourth International Conference on Data Science in Cyberspace (DSC)* (pp. 604-609). IEEE.
- Yanık, H., Turan, B., Yüksekokulu, M. M., Ü, T. G., Fakülte, M. ve Mühendisliği, B., 2021. Mesafe Ölçümünde Image -Metre : Görüntü İşleme ile Mesafe Tespit Yöntemlerinde Eksik Ne ? *4 th International Conference on Data Science and Applications (ICONDATA'21)*, June 4-6, 2021, TURKEY, 2, 480–486.
- Zhao, C., Sun, Q., Zhang, C., Tang, Y. ve Qian, F., 2020. Monocular depth estimation based on deep learning: An overview. *Science China Technological Sciences*, pp.1-16.
- Zhu, J., Wang, Z., Wang, S. ve Chen, S., 2020. Moving object detection based on background compensation and deep learning. *Symmetry*, 12(12), 1965.

Zhu, Y. ve Elgammal, A., 2017. A multilayer-based framework for online background subtraction with freely moving cameras. In Proceedings of the IEEE International Conference on Computer Vision (pp. 5132-5141).



ÖZGEÇMİŞ

Adı Soyadı

Kişisel Bilgiler

İletişim Bilgileri

Öğrenim Bilgileri

İş Deneyimi

