DÜZCE ÜNİVERSİTESİ | LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

TERMAL VE YÜKSEK ÇÖZÜNÜRLÜKLÜ KAMERA GÖRÜNTÜLERİNDEN DERİN ÖĞRENME KULLANILARAK DRONE TESPİTİ

OĞUZHAN YANIK

YÜKSEK LİSANS TEZİ
ELEKTRİK-ELEKTRONİK VE BİLGİSAYAR MÜHENDİSLİĞİ
ANABİLİM DALI

DANIŞMAN
DOÇ. DR. YUSUF ALTUN

DÜZCE, 2023

T.C.
DÜZCE ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ

TERMAL VE YÜKSEK ÇÖZÜNÜRLÜKLÜ KAMERA GÖRÜNTÜLERİNDEN DERİN ÖĞRENME KULLANILARAK DRONE TESPİTİ

Oğuzhan YANIK tarafından hazırlanan tez çalışması aşağıdaki jüri tarafından Düzce Üniversitesi Lisansüstü Eğitim Enstitüsü Elektrik-Elektronik ve Bilgisayar Mühendisliği Anabilim Dalı'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Doç. Dr. Yusuf ALTUN

Düzce Üniversitesi

Jüri Üyeleri

Doç. Dr. Yusuf ALTUN
Düzce Üniversitesi _____

Doç. Dr. Selman KULAÇ
Düzce Üniversitesi _____

Dr. Öğr. Üyesi Veli BAYSAL
Bartın Üniversitesi _____

Tez Savunma Tarihi: 23/06/2023

BEYAN

Bu tez çalışmasının kendi çalışmam olduğunu, tezin planlanmasından yazımına kadar bütün aşamalarda etik dışı davranışımın olmadığını, bu tezdeki bütün bilgileri akademik ve etik kurallar içinde elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara kaynak gösterdiğimi ve bu kaynakları da kaynaklar listesine aldığımı, yine bu tezin çalışılması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını beyan ederim.

23 Haziran 2023

Oğuzhan YANIK

TEŞEKKÜR

Yüksek Lisans öğrenimimde ve bu tezin hazırlanmasında gösterdiği her türlü destek ve yardımdan dolayı çok değerli hocam Doç. Dr. Yusuf ALTUN'a ve eğitim hayatım boyunca yardımlarını ve desteklerini esirgemeyen sevgili aileme ve çalışma arkadaşlarıma sonsuz teşekkürlerimi sunarım.

Ayrıca çalışma süreci boyunca desteklerini esirgemeyen Düzce İl Emniyet Müdürlüğü İHA Büro Amirliğine ve değerli arkadaşım Abdülkadir EROĞLU'na ayrıca teşekkürlerimi sunarım.

Bu tez çalışması, Düzce Üniversitesi BAP- 2020.06.01.1082 numaralı Bilimsel Araştırma Projesiyle desteklenmiştir

23 Haziran 2023 **Oğuzhan YANIK**

İÇİNDEKİLER

Sayfa No

ŞEKİL LİSTESİ vi
ÇİZELGE LİSTESİ vii
KISALTMALAR viii
SİMGELER ix
ÖZET x
ABSTRACT xi
1. GİRİŞ 1
2. TEMEL KAYNAK ARAŞTIRMASI 6
2.1. YAPAY SİNİR AĞLARI 11
2.1.1. İleri Beslemeli Sinir Ağları 12
2.1.2. Yinelemeli Sinir Ağları (RNN) 13
2.1.3. Evrişimli Sinir Ağları (CNN) 13
2.2. DERİN ÖĞRENME 14
3. MATERYAL VE YÖNTEM 16
3.1. FASTER R-CNN 16
3.2. SSD RESNET 19
3.3. YOLO 20
3.3.1. YOLOv4 22
3.3.2. YOLOv5 26
4. VERİ SETİ 29
4.1. MODEL EĞİTİMİ 33
4.2. DEĞERLENDİRME 33
5. DENEYSEL SONUÇLAR 36
6. TARTIŞMA VE SONUÇ 40
7. KAYNAKLAR 41
ÖZGEÇMİŞ 44

ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1 Denetimsiz Öğrenme [22] 6
Şekil 2.2 Denetimsiz Öğrenme Akışı [22] 7
Şekil 2.3 Denetimli Öğrenme[23] 8
Şekil 2.4 Denetimli Öğrenme Akışı [23] 8
Şekil 2.5 Pekiştirmeli Öğrenme [25] 9
Şekil 2.6 Nöron Yapısı [26] 10
Şekil 2.7 Yapay Sinir Ağı Model Gösterimi [26] 11
Şekil 2.8 İleri Beslemeli Sinir Ağı Sembolik Yapısı [27] 12
Şekil 2.9 Evrişimli Sinir Ağı Algoritma Çalışması [29] 14
Şekil 3.1 Faster R-CNN Algoritma Çalışması[31] 17
Şekil 3.2 Faster R-CNN Nesne Tespiti Süreci [32] 17
Şekil 3.3 Faster R-CNN Öznitelik Çıkarımı [31] 18
Şekil 3.4 Faster R-CNN Nesnelerin Belirlenmesi [32] 18
Şekil 3.5 SSD Nesne Tespiti [34] 19
Şekil 3.6 COCO Veri Seti Üzerinde SSD Nesne Tespiti [34] 20
Şekil 3.7 Nesne Alanı Belirleme[35] 21
Şekil 3.8 COCO Veri Setinde YOLO Performansı [35] 22
Şekil 3.9 COCO Veri Setinde YOLOv4 Performansı [36] 23
Şekil 3.10 YOLOv4 Nesne Tespiti İşlemi[37] 23
Şekil 3.11 YOLOv4 Nesne Tespiti İşlem Süreci[37] 24
Şekil 3.12 YOLOv4 CNN Katmanları [37] 25
Şekil 3.13 YOLOv4 Modelinin Çeşitli Görsellerde Çalışma Performansı [37] 25
Şekil 3.14 YOLOv5 Katmanları [39] 26
Şekil 3.15 YOLOv5 Modelleri Arasındaki Tespit Performansındaki Farklılıklar[39] 27
Şekil 4.1 Standart Kamera Verilerindeki Drone Görüntüleri 30
Şekil 4.2 Farklı Arka Planlarda Standart Kamera Verilerindeki Drone Görüntüleri 30
Şekil 4.3 Termal Kamera Verilerindeki Drone Görüntüleri 31
Şekil 4.4 Farklı Arka Planlarda Termal Kamera Verilerindeki Drone Görüntüleri 31
Şekil 4.5 Standart görüntüler ile yapılan eğitimin loss ve mAP değerleri 35
Şekil 4.6 Termal görüntüler ile yapılan eğitimin loss ve mAP değerleri 35
Şekil 5.1 Standart Kamere ile YOLOv4 Nesne Tespiti Görüntüleri Üzerinde Performans 36
Şekil 5.2 Termal Kamere ile YOLOv4 Nesne Tespiti Görüntüleri Üzerinde Performans 37
Şekil 5.3 Karmaşık Arka Planda Standart Kamera ve Termal Kamera Verilerinde Tespit Performansının Karşılaştırılması 38
Şekil 5.4 Termal Arka Planda Standart Kamera ve Termal Kamera Verilerinde Tespit Performansının Karşılaştırılması 39

ÇİZELGE LİSTESİ

Sayfa No

Çizelge 4.1. Model Değerleri..... 34

KISALTMALAR

CNN	Convolutional Neural Network (Konvolüsyonel Sinir Ağı)
COCO	Common Objects in Context (Bağlam İçinde Ortak Nesneler)
CSP	Content Security Policy (İçerik Güvenlik Politikası)
DETR	Detection Transformer (Algılama Dönüştürücüsü)
EfficientDet	Efficient Detection (Verimli Tespit)
EmguCv	Emgu Computer Vision (Emgu Bilgisayar Görüsü)
Faster R-CNN	Faster Region Based Convolutional Neural Network (Daha Hızlı Bölge Tabanlı Evrişimli Sinir Ağı)
FPS	Frames Per Second (Saniyedeki Kare Sayısı)
GPS	Global Positioning System (Küresel Konumlama Sistemi)
İHA	İnsansız Hava Aracı
ImageNet	Image Database Project (İmaj Veritabanı Projesi)
KNN	K-Nearest Neighbors (K-En Yakın Komşular)
MATLAB	MATrix LABoratory
OpenCv	Open Source Computer Vision Library (Açık Kaynak Bilgisayar Görüsü Kütüphanesi)
PAN	Personal Area Network (Kişisel Alan Ağı)
R-CNN	Region-based Convolutional Neural Network (Bölge Tabanlı Evrişimli Sinir Ağı)
RNN	Recurrent Neural Network (Tekrarlayan Sinir Ağı)
ReLU	Rectified Linear Unit (Doğrultulmuş Lineer Birim)
Resnet-101	Residual Network-101 (101 katman Residual Network)
RFCN	Region-based Fully Convolutional Networks (Bölge Tabanlı Tam Evrişimli Ağlar)
RF	Radyo Frekansı
RGB	Red, Green, Blue (Kırmızı, Yeşil, Mavi)
ShuffleNetV12	Shuffle-Based Network (Karıştırma Temelli Ağ)
SPP	Spatial Pyramid Pooling (Uzaysal Piramit Havuzu)
SSD ResNet	Single Shot Multibox Detection with Residual Networks (Residual Ağları ile Tek Çekim Çok Kutu Algılama)
SVM	Support Vector Machine (Destek Vektör Makinesi)
VGG16Net	Visual Geometry Group (Görsel Geometri Grubu)
YOLO	You Only Look Once (Sadece Bir Kez Bak)
YOLOv3	You Only Look Once version 3 (Sadece Bir Kez Bak Versiyon 3)
YOLOv4	You Only Look Once version 4 (Sadece Bir Kez Bak Versiyon 4)
YOLOv5	You Only Look Once version 5 (Sadece Bir Kez Bak Versiyon 5)

SİMGELER

C++	Sınıflar ile C Programlama Dili

ÖZET

TERMAL VE YÜKSEK ÇÖZÜNÜRLÜKLÜ KAMERA GÖRÜNTÜLERİNDEN DERİN ÖĞRENME KULLANILARAK DRONE TESPİTİ

Oğuzhan YANIK

Düzce Üniversitesi
Lisansüstü Eğitim Enstitüsü, Elektrik-Elektronik ve Bilgisayar Mühendisliği Anabilim Dalı
Yüksek Lisans Tezi
Danışman: Doç. Dr. Yusuf ALTUN

Haziran 2023, 43 sayfa

Dronelerin tehdit oluşturan kullanım faaliyetlerinde kayda değer bir artış gerçekleşmektedir. Tehdit oluşturan bu kullanımların yanı sıra oluşan durumlar ise her geçen büyüyen bir sorun haline gelmektedir. Bu durumlar sebebiyle drone faaliyetlerini oluşturan uçuş etkinliğinin tespit edilmesi ve tespit edilen uçuş etkinliğindeki olumsuz durumların belirlenmesi büyüyen sorun için çözüm niteliği taşıyacaktır. Bu tespit ve kestirim çalışmaları için derin öğrenme yöntemleri ile çalışma yapılmıştır. Literatürde yer alan sensör destekli tespit sistemlerinde akustik, termal ve radar çalışmalarından farklı olarak görüntü destekli bir sistem çalışması yapılmıştır. Bu çalışmada literatür çalışmalarında tespit başarısının düştüğü ve nesne tespitinin yapılamadığı durumlar ele alınarak termal ve yüksek çözünürlüklü standart kameradan eş açı ve zamanlama ile yapılan tespit faaliyetinde ortak bir noktada tespit faaliyeti çalışılmıştır. Çalışmadaki tespit faaliyetlerindeki başarı kriterleri için farklı hava koşulları, arka plan yoğunlukları ve karmaşıklıkların da toplanan veri seti ile Faster R-CNN (Residual Ağları ile Tek Çekim Çok Kutu Algılama), YOLOv4, v5 (Sadece Bir Kez Bak) ve SSD ResNet (Residual Ağları ile Tek Çekim Çok Kutu Algılama) derin öğrenme metotlarında karşılaştırmalı çalışmalar yapılmıştır. Uygulama üzerinde yapılan karşılaştırmalar ile görüntü boyut ve özelliklerinde göre en başarılı metodun belirlenmesi ile iyileştirilmiştir. İyileştirilerek elde edilen metot ile tespit nesnelerinin termal yoğunluğunun arka plan yoğunluğuna yakın olması durumunda yüksek çözünürlüklü standart kamera üzerinden desteklenerek, nesne ile benzer RGB (Kırmızı, Yeşil, Mavi) değerli arka planın olması veya karmaşık bir arka planın yer alması durumlarda ise standart kamera termal kamera ile desteklenerek tespit başarısının artırılması sağlanmaktadır. Bu iki sistem desteği ile sağlanan ve başarısı arttırılan tespit sürecinde literatürde de sorun teşkil eden helikopter, uçak ve kuş gibi havada hareketlilik sağlayan nesnelerinde ayırt edilebilmesi için belirlenen metot üzerinde ayrıca eğitimi gerçekleştirilmiştir. Yapılan çalışma sayesinde hareketliliği ve sayısı sürekli artan dronelerin tespiti daha başarılı hale getirilmiştir.

Anahtar sözcükler: Drone Tespiti, Drone Saldırı Tespiti, Kamikaze Drone, İnsansız Hava Aracı, Derin Öğrenme, Yapay Zeka

ABSTRACT

DRONE DETECTION USING DEEP LEARNING FROM THERMAL AND HIGH RESOLUTION CAMERA IMAGES

Oğuzhan YANIK

Duzce University
Graduate School of Education, Department of Electrical-Electronical and Computer Engineer
Master's Thesis
Supervisor: Assoc. Dr. Yusuf ALTUN

June 2023, 43 pages

There is a significant increase in the activity of the threatening use of drones. In addition to these threatening uses, the situations they create are becoming a growing problem. Due to these situations, the determination of the flight activity that creates the drone activities and the determination of the negative situations in the detected flight activity will be a solution for the growing problem. For these detection and estimation studies, deep learning methods were used. Different from acoustic, thermal and radar studies, an image-assisted system study has been carried out in sensor-assisted detection systems in the literature. In this study, by considering the cases where the detection success has decreased in the literature studies and the detection is not done, the detection activity at a common point in the detection activity made with the same angle and timing from the thermal and high-resolution standard camera has been studied. For the success criteria of detection activities in the study, comparative studies were carried out in Faster Region Based Convolutional Neural Network (Faster–RCNN), You Only Look Once (YOLO)v4, v5 and Single Shot Multibox Detection with Residual Networks (SSD ResNet) deep learning methods with the data set collected in different weather conditions, background densities and complexities. It has been improved by the comparisons made on the application and the determination of the most successful method according to the image size and features. With the improved method, in case the thermal density of the detection objects is close to the background density, it is supported by a high-resolution standard camera, and in cases where the object has a similar RGB (Red, Green, Blue) value background or a complex background, the standard camera is supported with a thermal camera to increase the detection success. In the detection process, which is provided with the support of these two systems and whose success is increased, a separate training has been carried out on the determined method for distinguishing objects that provide mobility in the air such as helicopters, airplanes and birds, which also pose a problem in the literature. Thanks to the study, the detection of drones, whose mobility and number is constantly increasing, has been made more successful.

Keywords: Drone Detection, Drone Intrusion Detection, Kamikaze Drone, Unmanned Aerial Vehicle, Deep Learning, Artificial Intelligence

1. GİRİŞ

Günümüzde kullanım alanları ve sayılarında yüksek artış olan İHA (İnsansız Hava Araçları)'nın kullanılan baz modelleri arasında droneler yer almaktadır. Dronelerin kullanımı ise şehirlerimizden taşralara kadar uzanan bir ağ içerisinde yaygın kullanım elde etmektedir. Bu kullanım, internet alışverişindeki paket veya yiyecek transferine kadar evrileceği ileri sürülmektedir. Ancak bu durum şehirlerde yaygınlaşan kullanımının da takip edilebilir ve görüntülenebilir olma ihtiyacını beraberinde getirmektedir. Bu durumun sebebi ise çeşitli sektörlerdeki kolaylıklarının yanı sıra toplum güvenliğini olumsuz etkileyecek tehdit unsurlarına açık oluşturmasıdır. Bu unsurlar, drone araçlarının kullanımı ile terör saldırısı veya terör unsurlarının gözetleme girişimi, yasa dışı taşıma ve faaliyetlerin etkinliği sebebiyle toplum içinde güven ortamının zedelenmesine sebep oluşturacaktır. Bu durumlar gibi açık oluşturan veya oluşturabilecek sebepler nedeniyle drone araçlarının tespit edilmesi önem arz etmektedir [1]. Tespit edilmesi ve konumunun belirlenmesi için literatürde yer alan çeşitli yöntemler bulunmaktadır. Literatürde yer alan başlıca yöntemler şu şekildedir; RF (Radyo Frekansı) temelli radar yaklaşımları, akustik sensör temelli ses tanımlama yaklaşımları, GPS (Küresel Konumlama Sistemi) veya RF temelli sinyalizasyon yaklaşımları ve LİDAR gibi sensör temelli yaklaşımlar ile nesne tespiti sağlanmaya çalışılmaktadır [2]. Bu temel üzerine yaklaşımlar matematiksel veriler üzerinde belirleme sürecini oluşturmaktadır. Ancak bu yöntemlerin beraberinde güncel teknolojileri barındıran derin öğrenme ve makine öğrenmesi başlıklarında da çalışmalar yapılmaktadır [3][4][5], [6][6]. Derin öğrenme ve makine öğrenmesi temelli tespit işlemlerinde başarı oranında artış olsa da farklı nedenlerden dolayı eksikler oluşmaktadır[7].

Drone tespitinde verimli bir tespitin sağlanması için iki aşama yüksek önem taşımaktadır. Bu aşamalar tespit edilen dronenin benzeri diğer nesnelerden (uçak, helikopter, kuş vb.) ayırt edilebilmesi ve kaynak belirtilen ortamın hava koşullarıdır [8]. Bu aşamaların aşılabilmesi için farklı sensör ve kameraların beraber kullanılması önerilmiştir. Ayrıca farklı yöntemlerinde kullanımı sağlanmıştır. Yapılan işlemler ile verimli bir tespit çıkarımı elde edilmeye çalışılmıştır. Çalışma içeriğinde ise Yüksek çözünürlüklü kameradan elde edilen veri seti YOLO v2 ağında eğitilerek tespit için önem arz eden

farklı hava araçları ve canlılardan dronenin ayrımı sağlanmıştır. Çalışma ile yüksek çözünürlüklü kameradan alınan kaynak görüntüsü içeriğinde tespit nesnesinin odağı drone olarak belirlenmiştir [9]. Bir başka çalışmada ise makine öğrenmesi algoritmaları ile sınıflandırma yapılarak drone tespiti ve özellik çıkarımı için ek bir metot ile tespit başarısı arttırılmaya çalışılmıştır. Çalışma içeriğinde tahmin ve tespit çıkarımı sadece çıkarım sağlanan özelliklere bağlanması, veriyi sağlayan sistemin durum değişkenlerine göre değişiklik gösterebilmektedir. Bu nedenle sistem çıktısı için ortam kaynaklı değişkenler kontrol edilmeli ve özellik çıkarımındaki değişkenler gözetilmelidir [10]. Bu aşamalara bir başka yaklaşım ise R-CNN (Bölge Tabanlı Evrişimli Sinir Ağı) ve Faster R-CNN ağlarını MATLAB (MATrix LABoratory) ortamında eğitilmesi ile bir tek sınıf olan drone tespitinde tespit başarısı sağlamıştır. Ancak bu başarı sağlanırken drone dışındaki benzer nesnelerin tespitine yönelik bir eğitim yapılamamaktadır [11]. Literatürde yer alan farklı bir çalışma da ise SSD, Resnet-101 (Residual Network) ve Faster R-CNN ağlarında 65 bin iterasyon ile yapılan eğitimde açık hava koşullarında uzak mesafelerde drone nesnesinin tespiti sağlanmaya çalışılmıştır. Çalışmanın açık hava koşullarında SSD, Resnet-101 ve Faster R-CNN ağları arasında başarılı olan ağı Resnet-101 ile Faster R-CNN olarak belirlemişlerdir [12]. Dronelerin yasa dışı uçuşlarındaki durumları incelemek için farklı çalışmalarda bulunmaktadır. Dronelerin yasa dışı uçuşlarının belirlenmesi üzerine sensör destekli bir yapı kurarak CNN (Konvolüsyonel Sinir Ağı) modellerinden VGG16Net (Görsel Geometri Grubu) ve ShuffleNetV12 (Karıştırma Temelli Ağ) ağlarında eğitimler sağlanarak tehlike analizini yapmak üzere derin öğrenme modeli oluşturulmuştur. Oluşturulan model içerisinde verilerin yakınlık değerine göre alanlar belirlenmiştir. Bu alanlara yaklaşım gösteren 0-2 arası nesnenin yakınlıkları üzerinden bir kestirim sağlanmaya çalışılmıştır. Nesnelerin yakınlıklarına, sayılarına ve yaklaşımlarına bağıl olarak derin öğrenme modelinde tehlike analizi üzerine bir yaklaşım sağlanmıştır. Fakat yapılan çalışmada bu yaklaşım modeli üzerinde yasa dışı uçuş kestiriminin başarısı yüksek değerler taşımamaktadır [13]. Tehlike analizi gibi yaklaşımların nitelikli olarak sağlanması için öncelikli olarak dronenin tespiti ve sınıflandırılması sağlanmalıdır. Bu konuda yapılan çalışmada iki temel yapı kurulmuştur. Bu yapılar, tespit işleminin CNN modeli üzerinde sağlanması ver ardından temel makine öğrenmesi metotları olan SVM (Destek Vektör Makinesi) ve KNN (K-En Yakın Komşular) ile CNN modeli üzerinde bir karşılaştırma sağlamışlardır. Bu yaklaşım üzerinden CNN modeli ile tespit edilen nesnelerin 3 metot ile sınıflandırılması yapılmıştır. Sınıflandırma işleminde CNN, diğer metotlara göre yüksek bir sonuç

vermiştir. CNN modelinde yapılan sınıflandırmanın başarı değerleri ise hassasiyet %91, tahmin %96 ve doğruluk %93 olarak paylaşılmıştır [14]. Dronelerin tespiti için CNN modellerinin yanı sıra güncel modellerinde çalışmaları yapılmıştır. Bu çalışmalar içerisinde YOLO versiyonları olan YOLOv4, v5 ve DETR (Algılama Dönüştürücüsü) ağlarında eğitim sağlanarak tespit için başarı kriterlerine ek olarak eğitime verilen veri setinin eğitim, test ve doğrulama verileri olarak ayrımı konusunda da bilgi verilmiştir. YOLO ağlarında sağladıkları eğitim süresince eğitim, doğrulama ve test sırasıyla %70, %20 ve %10 olarak ayrılarak ağa verilmiştir. DETR ağına ise doğrudan eğitim seti verilmiştir. Ağlar üzerinde yapılan eğitim sonucunda ise YOLO modelinin güncel versiyonu olan v5 üzerinde yüksek oranda doğruluk ile başarı elde edilmiştir [15]. Yine farklı bir çalışmada YOLOv3 ağında çalışmalar sağlanmıştır. Bu çalışmada YOLO temelli tespit çalışmalarında elde edilen başarının drone tespiti üzerindeki başarısı test edilmiştir. Bu test işleminde YOLOv3 üzerinde drone tespiti için oluşabilecek eksiklikler ve sorunların araştırılmasına da yer verilmiştir. Aynı zamanda farklı akademik çalışmaları ve yarışmaları da göz önünde bulundurarak araştırmalarını derinleştirmeye çalışmışlardır [16]. Bu derinlik kazanan çalışma içerisinde termal temelli kamera ile toplanan veriler üzerinden yapılan tespit sırasında tespit nesnesinin kutu içerisine alınma durumlarına bakılmıştır. Bu durumlar YOLOv3 ile eğitim gözetilerek eksik, kaymış ve ek kutu olarak sınıflandırılmıştır [17]. Drone tespiti için yapılan çalımlar içerisinde farklı bir yaklaşım çalışması da yapılmıştır. Bu yaklaşımı sunan çalışmada, arka plan yoğunluğunun tespit başarısını nasıl etkilediği araştırılmıştır. Bunun için, SSD MobilNetV2, EfficientDet-D3 (Verimli Tespit) ve YOLOv4 mimarileri kullanılarak eğitimler yapılmış ve sonuçları karşılaştırılmıştır. Araştırmanın sonuçlarına göre, YOLOv4 mimarisi diğer iki mimariye göre daha başarılı bir performans sergilemiştir. Bu nedenle, başarılı olan model olan YOLOv4 mimarisi ile farklı video örnekleri kullanılarak yapılan denemeler sonucunda, görüntü çözünürlükleri arasında 720p görüntü akışında tespit mesafesi ve performansının oldukça verimli olduğu görülmüştür[18]. Öte yandan, bir başka çalışmada mimarilerin yanı sıra çalışma ortamının ve programlama dillerinin tespit işlemi açısından önemli olduğu vurgulanmıştır. Yapılan bu çalışmada, Matlab, OpenCv (Açık Kaynak Bilgisayar Görüsü Kütüphanesi) ve EmguCv (Emgu Bilgisayar Görüsü) araçları incelenmiş ve ilgili programlama dilleri de göz önünde bulundurulmuştur. Çalışma ortamlarının çıktıları denetlendiğinde ise dört farklı drone modeli ve tespit aşamasında görülebilecek farklı nesneler eğitime sunularak tespit süreleri ile özelliklerince Linux Ubuntu işletim sistemi üzerinde C++ programlama dili kullanılarak açık kaynak kütüphaneler ile tespitin etkin

ve verimli olduğu sonucuna varılmıştır[19]. Drone tespitinde ortam ve mimarilerden yola çıkarak tespit başarısındaki etkileri değiştirmektedir. Ancak bu tespit işleminden yola çıkarak drone tespitinin ardından tespit edilen nesnenin tehdit oluşturması konusunda da çalışmalar yer almaktadır. Bu açıdan ele alınan çalışmada, kullanımında artış olan dronelerin tespit edilen yer ve duruma göre tehdit kestirimi yapılmaya çalışılmıştır. Bu konuda tespit edilen aracın yakınlığı ve hareketi yapay sinir ağları ile denetlenmiştir. Kamera merkezi üzerinden alınan görüntülerde derinlik algısı oluşturularak güvenli bölge, hafifletme bölgesi ve alarm bölgesi olarak 3 bölge oluşturulmuştur. Ele alınan kamera görüntüleri içerisinde drone tespiti için SSD, Faster R-CNN ve RFCN Resnet101 (Bölge Tabanlı Tam Evrişimli Ağlar) modelleri test edilmiştir. Yapılan bu test işlemlerinde RFCN Resnet101 modelinin başarılı olduğu çıkarılmıştır. Çalışma genelinde ise çıkarım, kamera görüntülerinde RFCN Resnet101 modeli ile tespit edilen nesnenin yer aldığı çerçeve içerisindeki bölge ve alan belirlenmesi yönündedir. Çıkarım sonucunda tespit edilen dronenin tehdit kestirimi yapılmıştır[20]. Tespit edilen nesnelerin takibi üzerine olan bir başka çalışmada ise YOLOv2 modeli ile 416 x 416 boyutunda renkli ve termal görüntülerle beraber akustik sensör verisi ele alınmıştır. Bu çalışmada her girdi verisi ayrı ayrı ele alınarak çıktıları oluşturulmuştur. Elde edilen model çıktıları değerlendirilerek nesnenin tespitinde karar kılınmıştır. Ancak çalışmanın test edildiği görsellerdeki düşük gürültü ile termal kamera üzerinde 0,7601, renkli kamera üzerinde ise 0,7849 F1 değeri elde edilmiştir[21].

Nesne tespiti ve derin öğrenme konularında yapılan literatür araştırmalarında drone nesnesinin tespiti için derin öğrenme modelindeki başarı oranının artırılmasına yönelik yaklaşımlar araştırılmıştır. Bu çalışmalar üzerinden başarılı modellerin belirlenmesi ve aralarındaki performans parametrelerindeki değişim incelenmiştir. Performans kriterleri için nesne tespiti modellerinin FPS (Saniyedeki Kare Sayısı) değerleri ön planda tutularak başarılı modellerin seçimi sağlanmıştır. Seçilen modellere verilen girdilerin eksik yönleri ve veri seti modellerindeki eksiklikler araştırma içeriğinde incelenmiştir. Bu eksiklikler göz önüne alınarak uygulanacak yaklaşımda; drone tespiti için karmaşık arka plan bulunan alanlarda veya drone ile renk uyumu bulunan arka planlarda termal kamera ile, arka planın ısıl yoğunluğunun drone nesnesinin ısıl yoğunluğuna yakın olduğu durumlarda standart kamera ile eksiklikler giderilmeye çalışılmıştır. Bu eksikliklerin giderilmesi için drone tespiti süresince iki kamera eş zamanlı olarak çalışmaktadır. Eş zamanlı çalışma sayesinde iki kaynak görüntünün çıktıları belirli ağırlıklar ile

düzenlenerek ilgili durumlarda dronenin tespiti sağlanmıştır. Ayrıca modelin tespiti için kuş, uçak, helikopter gibi diğer nesnelerin tespiti, drone aracının tespitinde sıklıkla karşılaşılan bir problemdir ve bu problem de ele alınmıştır. Bu problemi indirgeyebilmek için drone tespit işlemi sürecinde diğer nesnelerin ayrımı üzerinde veri seti düzenlemeleri yapılarak model eğitimine eklenmiştir. Eklenti sayesinde drone aracının tespiti daha başarılı hale getirilmiştir.

Sonuç olarak, bu araştırma drone tespitinde kullanılan mimarilerin performansını karşılaştırmış ve gerçek zamanlı çalışma performansı için en başarılı mimarinin seçimini sağlamıştır. Ayrıca, kullanılan girdilerin eksik yönleri için bir yaklaşım ele alınmış ve diğer nesnelerin tespiti de ele alınarak drone aracının tespiti daha başarılı hale getirilmiştir. Bu çalışma, drone tespiti alanındaki gelecekteki araştırmalar için bir temel oluşturabilecektir.

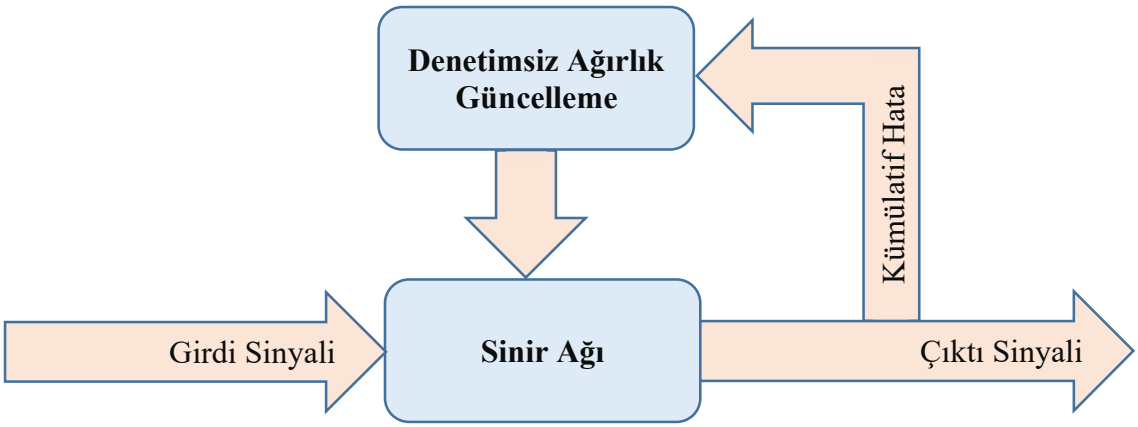
2. TEMEL KAYNAK ARAŞTIRMASI

Yapay zekâ ve makine öğrenmesi günümüzde gelişimini hızlandıran ve yaygınlaşan bir çalışma alanıdır. Yapay zekâ, insanı model alarak makinelerin karar verebilmesi, düşünebilmesi ve öğrenmesi gibi zekaya dayalı işlemleri çözümlemesine olanak tanır. Yapay zekâ teknikleri arasında yer alan makine öğrenmesi ise, makinelerin veriye dayalı öğrenme ve bu öğrenim üzerinden kendini geliştirmesini sağlamaktadır. Öğrenime imkân veren makine öğrenmesi çeşitli şekillerde makine öğrenimini sağlayabilir. Bunlar ise alt başlıklar halinde ele alınmaktadır.

Denetimsiz Öğrenme

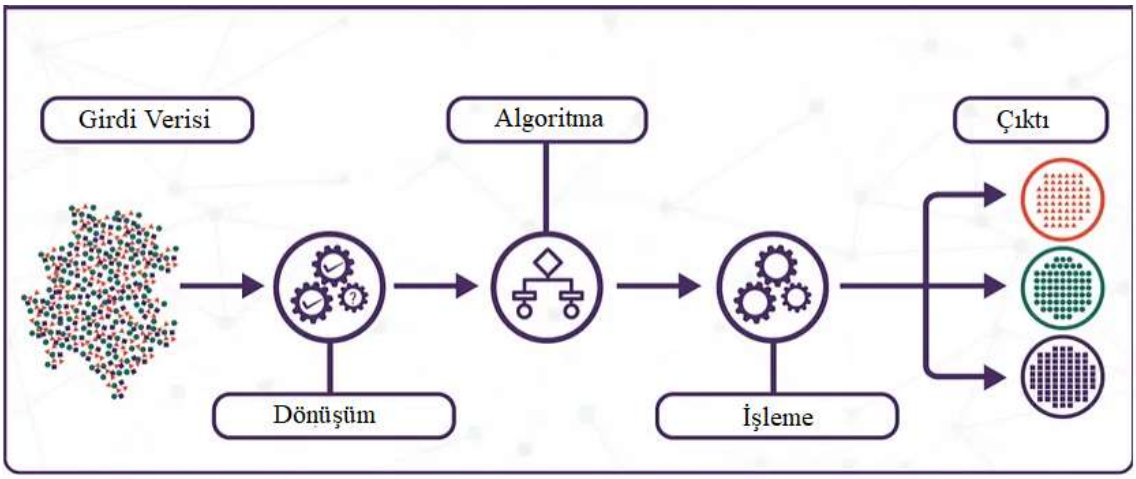
Veri veya veri grupları arasındaki bağlantılı durumu anlamlandırmak için kullanılır. Bu öğrenme türünde ilgili öğrenimi sağlayacak olan makineye veri seti verilir. Ancak verilen veri setinden elde edilen öğrenme sonucunda hangi çıktının elde edileceği belli olmamaktadır. Veri seti sağlanan makine, verilerdeki özellikler üzerinden desen ve yapıya odaklı kendi modelini oluşturmaktadır.

Sıralı veri gruplarında bağlantılı olmayan modeller üzerinde eğitim için bu yaklaşım kullanılarak makine öğrenmesi sağlanabilir. Bu öğrenim yaklaşımında, bağımsız çıktı değerleri oluşturan modellerdeki durumlara karşı tahmin sağlanmaktadır. [22]



Şekil 2.1 Denetimsiz Öğrenme [22]

Şekil 2.1'de denetimsiz öğrenme esnasında giriş sinyalinin model üzerinden çıkışı sağlanmaktadır. Elde edilen çıkış sinyali ise kümülatif bir hata oranıyla beraber modele besleme olarak verilmektedir.



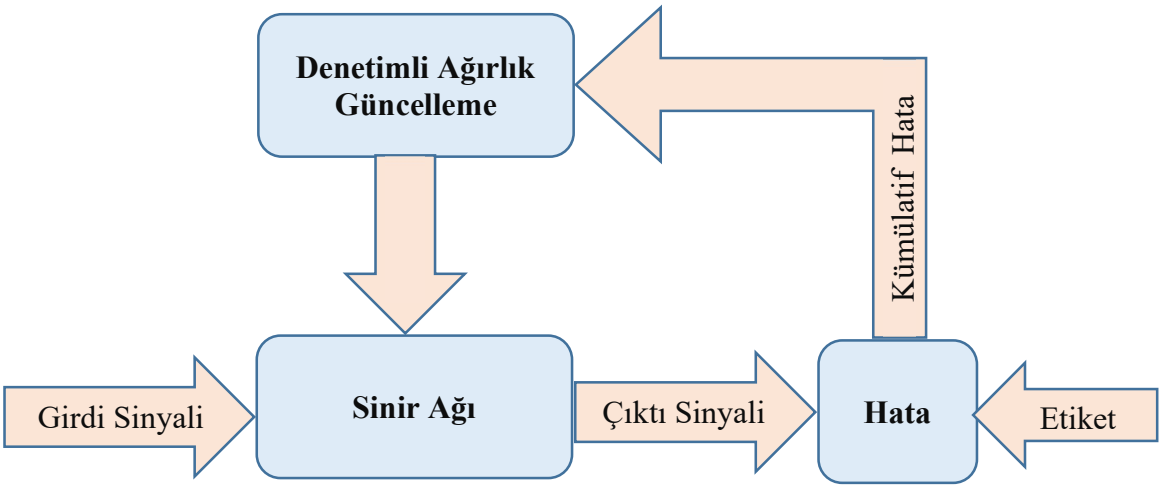
Şekil 2.2 Denetimsiz Öğrenme Akışı [22]

Şekil 2.2'de giriş verisinin çıktısı elde edilirken geri beslemeye dayalı bir sistem gerçekleştirilememektedir. Bu sebeple elde edilen çıktı, gelecek giriş değişkenlerini doğrudan etkilememektedir. Bu sayede model eğitimi gerçekleştirilirken o anki veri işlenmiş olmaktadır.

Denetimli öğrenme

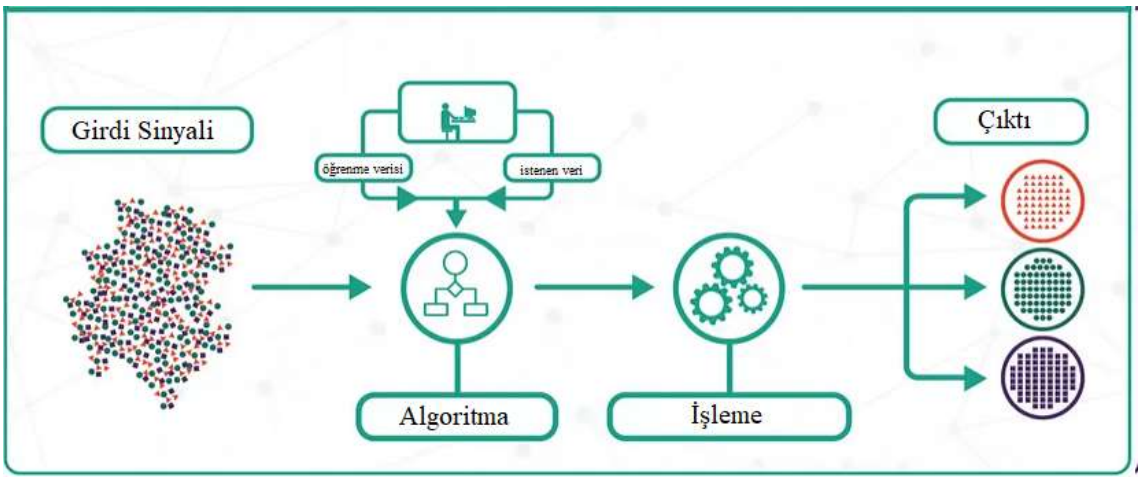
Makine öğrenmesi sürecinde denetimli öğrenme, veri setinin model beslemesi verilirken bir dizi çıktının da doğru olarak verilmesi ile gerçekleşmektedir. Günümüzde yaygın olarak kullanılan nesne tanıma uygulamalarında kullanılan makine öğrenmesi metodu birden fazla değişkene de bağlı olarak başarı değerlerini değiştirebilmektedir. Bu değişkenler modelin eğitimi esnasında verilen çıktıların doğruluğu ve tanımlamalarına bağlı olarak oluşmaktadır.

Bu öğrenme metodu sürecinde makine eğitimi gerçekleşirken model oluşturulur ve elde edilen model üzerinde tahmin gerçekleştirilir. Bu tahmin edilen çıktı ile modele verilen çıktılar karşılaştırılarak model ağırlıkları düzenlenir. Bu sayede denetimli öğrenme modeli sabit özniteliklere dayalı tespit işlemlerinde kullanım sağlanmaktadır[23].



Şekil 2.3 Denetimli Öğrenme[23]

Şekil 2.3'te model eğitimi sürecinde modele girdi sağlayan veriler modelden çıkarılarak elde edilen çıktı, modele verilen çıktılar ile karşılaştırılmaktadır. Bu karşılaştırma sonucunda doğruluğu denetlenir ve modele besleme olarak verilir. Bu sayede modelin gelişimi denetimli olarak gerçekleşir.



Şekil 2.4 Denetimli Öğrenme Akışı [23]

Şekil 2.4'te giriş verisi üzerinden çıktı elde edilirken geri beslemeye dayalı bir sistem gerçekleşmektedir. Bu geri besleme makine eğitimi gerçekleştirilmeden önce verinin bir kısmının çıktısı etiketlenir ve doğrulama verisi elde edilir. Bu doğrulama verisi sayesinde elde edilen çıktı gelecek giriş değişkenlerinin çıktısı için doğrudan model ağırlıklarının düzenlemesinde kullanılır. Bu sayede model eğitimi gerçekleştirilirken kümülatif bir veri işlenmiş olmaktadır.

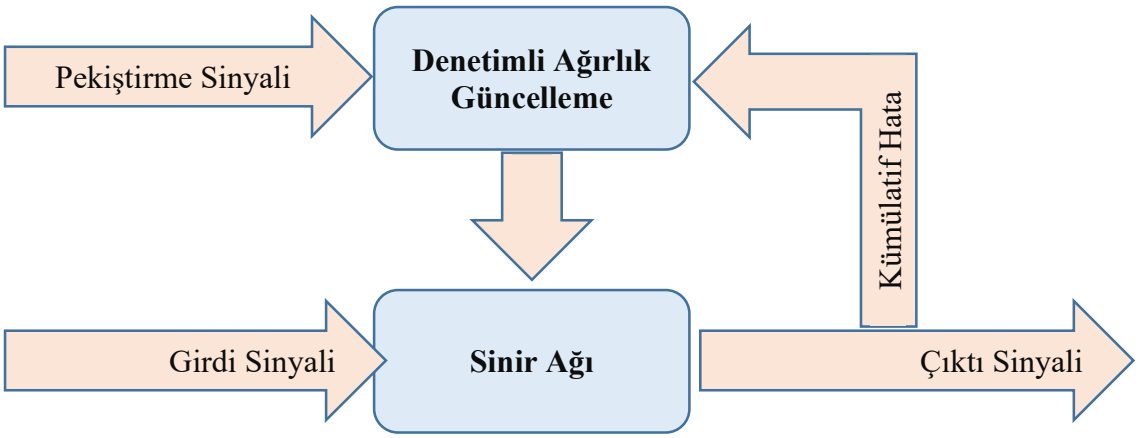
Yarı Gözetimli Öğrenme

Yarı denetimli öğrenme hem denetimli hem de denetimsiz öğrenme tekniklerini esas almaktadır. Bu makine öğrenmesi metodu, bazı girdilerin sisteme verilen çıktılar ile denetimli olarak öğrenmesi ve bazı girdilerin ise denetimsiz olarak öğrenmesiyle sağlanmaktadır. Bu öğrenme metodunda etiketlenen temel veriler ile sistem başlangıcı verilebilmektedir. Ardından detaylandırma veya ilişkilendirme süreçleri için sistemin denetimsiz öğrenmesi ile sağlanmaktadır[24].

Pekiştirmeli Öğrenme

Pekiştirmeli öğrenme metodu, makine öğrenmesi süreçlerinde veriye dayalı öğrenme sürecini oluşturmaktadır. Bu süreçte makineye verilen veri seti üzerinden makinenin çıkarımları ile çıktılar elde etmesi beklenmektedir. Pekiştirmeli öğrenme modelleri, makinenin bir görevi sağlayabilmesi için her çıktı sonucunda tepkiler oluşturmaktadır. Bu aşamada, makine öğrenimi boyunca en iyi tepkiyi alabileceği sonuçlar vermeye yönelmektedir. Bu sayede öğrenim süreci makinenin elde ettiği deneyime bağlı olarak gelişim göstermektedir[25].

Bu model sayesinde günümüzde yaygınlaşan çoğu yapay zeka robotunun kullanımının artmasıyla gelişim göstermektedir. Bu yaklaşım sayesinde sabit girdi ve çıktılardan bağımsız hale gelecek olan model, verilen girdilerin işlenmesi sonucunda daha genel bir çıktı elde etmektedir. Ayrıca daha genel bir kullanıma da olanak sağlayan model genel olarak sistemin kendini beslemesi ile gelişim göstermeye devam edecektir.

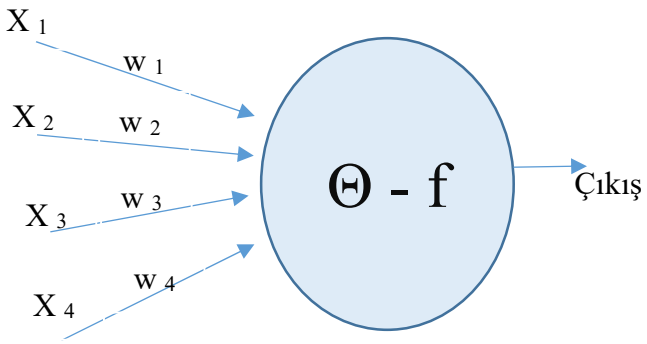


Şekil 2.5 Pekiştirmeli Öğrenme [25]

Şekil 2.5'te sisteme verilen girdinin çıktısı elde edilmektedir. Bu elde edilen çıktı tepkileri ele alarak model ağırlıklarında güncelleme sağlamaktadır. Bu sayede sistemin çıktısı genişletilmiş olmaktadır.

Nöron ve Perceptron

Nöron, girdi sinyallerinin ağırlıklar ile çarpılması sonucunda bir aktivasyon fonksiyonu kullanarak bilgi sinyali üreten temel yapı taşıdır. Yapay sinir ağlarında temel yapı taşını oluşturan nöron, biyolojik nöronlardan esinlenilerek oluşturulmuştur. Genellikle matematiksel işlemler yapmak için kullanılan nöronlar bir çıktı sinyali oluşturur ve oluşturmuş oldukları çıktı sinyalleri sonuç veya bir başka nörona girdi sinyali olarak iletim sağlar. Perceptron ise sınıfların ayırt edilmesi veya çıktı sınıfının matematiksel ağırlıklar ile belirlenmesini sağlayan en temel model olarak ele alınabilir. Bu model yapay sinir ağlarında modelin çıktısını oluşturmak için kullanılır. Bu aşamada, girdi sinyali olarak alınan sinyalin ağırlık matrisi ile çarpılması sonucunda aktivasyon fonksiyonuna iletimi sağlanır. İletimi sağlanan çıktı sinyalinin değeri, oluşturulan eşik değerine göre karşılaştırılarak çıktı sınıfı elde edilir. Yapay zekanın temel kavramları olan nöronlar ve prceptronlar yaygın olarak kullanılan kavramlardır. Bu temel yapı taşlarının bir araya gelmesiyle yapay sinir ağları oluşturulur. Bu oluşumlar ise farklı yapay zeka uygulamalarında yaygın olarak kullanılmaktadır[26].

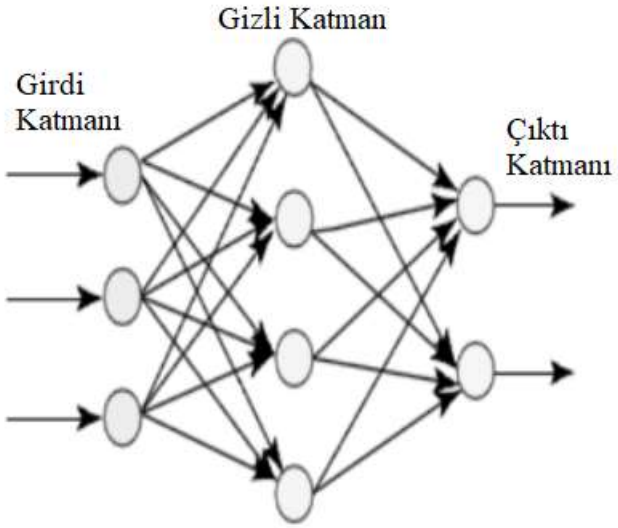


Şekil 2.6 Nöron Yapısı [26]

Şekil 2.6'da bir nöron yapısı verilmiştir. Bu yapıda nörona girdi sağlayan x sinyalleri, w değerli ağırlıklar ile işleme girmektedir. Bu işlem üzerinden bir nöron başka bir nörona girdi veya çıktı sinyali oluşturur. Bu döngü ile nöronlar işlemlerini sağlamaktadır.

2.1. YAPAY SİNİR AĞLARI

Nöronların birbirine bağlanması ile elde edilen ağlar yapay sinir ağları olarak adlandırılır. Yapay sinir ağlarını oluşturan nöronlar aldıkları girdi sinyallerini ağırlıklar ile işleme sokarlar ve çıktı elde ederler. Bu nöron birimlerinin bir araya gelmesi ile karmaşık ve bağlantılı olarak oluşan yapı, yapay sinir ağlarını oluşturur. Oluşturulan yapay sinir ağlarında çıktı sinyalinin elde edilmesi için genellikle bir eğitim süreci gerçekleştirilir. Bu eğitim sürecinde girdi verileri oluşturulan ağ modeline verilir ve her nöron üzerine verilen sinyali ağırlık ile çarparak kendi çıktı değerini elde eder. Elde edilen çıktı değeri bir sonraki nörona aktarılarak işlem döngüsü devam eder. Aktivasyon fonksiyonuna kadar iletilen sinyal bir değeri oluşturmuş olur ve daha önceden oluşturulan çıkış sınıflarının değerleriyle karşılaştırılarak çıktı sinyalinin sınıfı belirlenir.



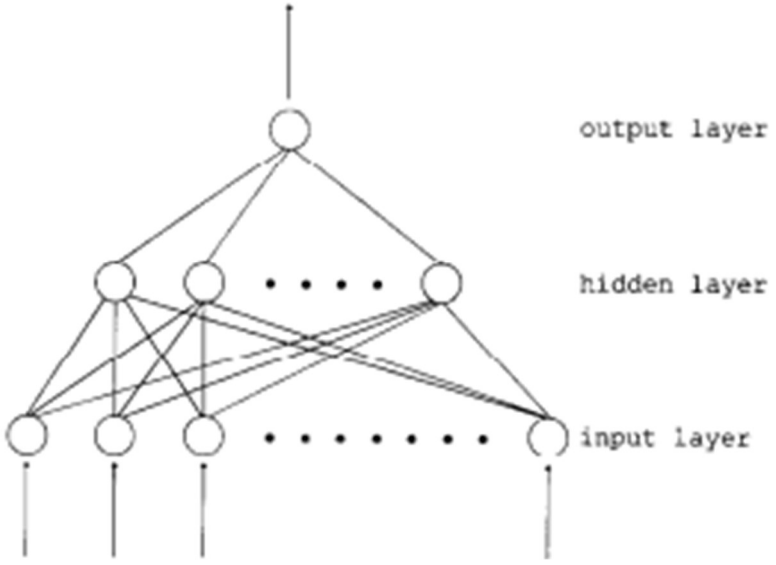
Şekil 2.7 Yapay Sinir Ağı Model Gösterimi [26]

Şekil 2.7'de gösterildiği şekilde yapay sinir ağları genel olarak 3 ana katmandan oluşur. Bu katmanlar girdi katmanı, gizli katman ve çıkış katmanıdır. Girdi katmanından girdi sinyali alınarak gizli katmana aktarılır ve bu katmanda girdi sinyali ağırlıklar ile hesaplamalar yapılarak işlenir. Ardından çıktı katmanında ise bu girdi sinyali için sınıfı belirten bir sonuç elde edilir.

2.1.1. İleri Beslemeli Sinir Ağları

İleri beslemeli sinir ağları lineer veya nonlineer modellerin eğitimi ve çalışması için kullanılan bir yapay sinir ağı modelidir. Sinir ağının yapısal oluşumunda ise bir girdi, bir çıktı ve birden fazla gizli katmandan sağlanmaktadır. Bu yapı özelinde girdi katmanı, modele verilen girdi sinyalini işlem için hazırlar ve girdi üzerindeki öznitelik fonksiyonuna verilerek ileri katmanlara besleme sağlanır. Bu katmanın genel yapısında girdi sinyalinin modele uygun olarak düzenlenmesi sağlanır. Ardından girdi sinyali gizli katmana aktarılır ve girdi sinyali üzerinde öznitelikler çıkarılarak veya oluşturularak girdi sinyalini anlamlandırmaya başlar. Bu aşamada her katman model ağının öğrenmesi için yeni özellikler çıkartarak modelin eğitimine katkı sağlar ve ağ içerisinde her gizli katman çıktısı bir sonraki katmanın girdisini oluşturur. Bu işlemlerin ardından en son katman olan çıktı katmanına sinyal iletilir. Çıktı katmanına gelen sinyal gizli katmanda elde edilen öznitelikler üzerinden bir sınıflandırma veya tahmin ile sonuçlanır. Bu çıktı sınıflandırma veya tahmin değerlerine olan yakınlığa bağlı bir olasılık değeri ve etiket değerini oluşturur[27].

İleri beslemeli sinir ağları, öğrenme algoritmaları kullanılarak eğitilir ve veri setine verilen eğitim verilerini işleyerek yüksek doğrulukta değer üretimi için ağ katmanlarındaki ağırlık ile beraber bias değerlerini günceller. Bu yapı genelinde eğitim süreci güncellemeli olarak değer yenilemesi sağladığı için veri boyutu konusunda da esneklik sağlamaktadır.



Şekil 2.8 İleri Beslemeli Sinir Ağı Sembolik Yapısı [27]

Şekil 2.8'de ileri beslemeli bir yapay sinir ağının yapısı verilmiştir. Bu yapı genel olarak sisteme verilen girdi sinyalinin öznitelikler ile beraber verilmesiyle başlar. Ardından sinyal bir veya birden fazla katmandan oluşabilen gizli katmana aktarılarak işlenir ve çıktı katmanına aktarılır. Çıktı katmanından ise girdi verilen sinyalden bir olasılık ve sınıflandırma sağlanır.

2.1.2. Yinelemeli Sinir Ağları (RNN)

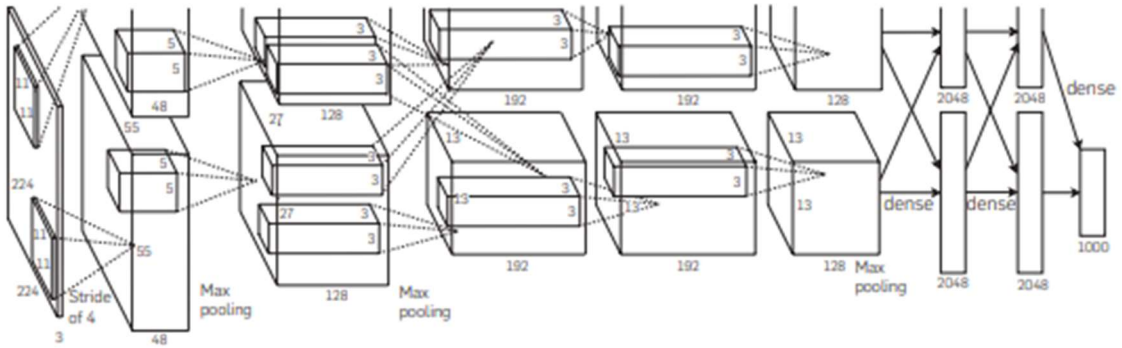
Yinelemeli, sinir ağları genel yapısı itibariyle sıralı veri işleme konularında kullanımı yaygındır. Bunun sebebi ise ağ modelinin yapısında bir önceki çıktı değerinin yeni gelen veri girdisinden elde edilen çıktıda etkin olarak rol oynamasıdır. Ağ yapısı, model çıktılarını geri modele veren bir geri besleme sistemini barındırmaktadır. Genel olarak bu yapı modelden elde edilen önceki çıktılardan bir matris oluşturur. Oluşturulan bu matris yeni çıktıların oluşturulması için kullanılır. Bu kullanım ise modelin girdi ve çıktıları vektörel olarak tanımlamasından kaynaklıdır. Vektörel olarak tanımlanan eski çıktı matrisi, sisteme verilen yeni girdi matrisi ile birleştirilerek yeni bir çıktı matrisi oluşturur. Bu yeni elde edilen vektör matrisi ise yeni gelen sinyalin işlenmesi sırasında önceki durumu temsil eden bir matristir[28].

Yinelemeli sinir ağları genel yapısı sebebiyle geçmiş değer matrisini aktif olarak yenileyen ve sistemi bununla besleyen bir mimariye sahiptir. Bu sebeple kısa hafızalı yapay sinir ağı işlemlerinde daha yaygın kullanıma sahiptir.

2.1.3. Evrişimli Sinir Ağları (CNN)

Evrişimli sinir ağları (CNN), girdi verilerinin filtrelenmesi, özniteliklerinin çıkarılması ve sınıflandırılması için kullanılır. Bu kullanım için ağ yapısı birçok katmandan oluşmaktadır. Bu katmanların yapısı ise belirli filtre boyutlarına ve katman boyutlarına bağlı olarak oluşmaktadır. Her filtre, girdi verilerindeki öznitelik çıkarımına bağlı olarak farklı özellikler için çıkarım yapmaktadır. Bu katmanlarda elde edilen çıktılar öznitelik değerlerine bağlı olarak boyutları küçültülür ve ana katmanlardan geçirilir. Bu katmanların ardından sinyal çıktı katmanına aktarılır ve sınıflandırma sağlanır. Bu katmanlardaki işlemler ve filtrelemeler sayesinde CNN'ler girdilerdeki öznitelikleri model içerisinde çıkararak otomatik öğrenme kabiliyetine sahiptir. Elde edilen öğrenme modeli ile karmaşık verilerden bağımsızlaşarak öznitelik çıkarımını sağlar ve yüksek başarı oranlarına sahip çıktılar çıkarabilirler.

CNN'lerin özellikleri sayesinde nesne tanıma, sınıflandırma, görüntü işleme gibi uygulamalarda sıkça kullanılır. Bu kullanım, örneğin bir görüntü işleme uygulamasında girdi olarak verilen görüntü üzerinden öznitelikler çıkararak verinin tanımlanması için model öğrenim sağlar. Bu öğrenimin ardından sisteme verilen yeni görüntülerde ilgili öznitelikleri çıkarır ve öğrenimden elde ettiği veriler ile karşılaştırarak sınıflandırma veya tahmin çıktısını oluşturur. Bu özellikleri ile çok yönlü bir kullanım sağlayabilirler. Ancak yapılan işlemler için yüksek bir maliyet oluştururlar ve ağ üzerindeki yüksek katmanlar bir karmaşıklık oluşturur[29].



Şekil 2.9 Evrişimli Sinir Ağı Algoritma Çalışması [29]

Şekil 2.9'da yer alan görsel üzerinde CNN ağına verilen bir girdi sinyali üzerinde özniteliklerin olduğu alanlar belirlenir. Ardından bu öznitelikler çok katmanlı filtrelerden geçirilerek öznitelik tanımlamaları sağlanır. Öznitelik çıkarımı ardından öznitelik havuzlarından geçirilen sinyaller çıktı katmanına aktarılır ve özniteliklerden bir çıktı elde edilir. Bu sayede girdi verisindeki öznitelikler doğrudan ağ üzerinde otomatik olarak çıkarılır. Modelin öğrenimi ve ardından tahmin işlemi bu döngü üzerinden belirlenir.

2.2. DERİN ÖĞRENME

Derin öğrenme, yapay sinir ağları ile kurulan bir yapıya sahiptir. Bu öğrenme işlemi sınıflandırma ve tahmin gibi işlemlerin yapılmasına olanak sağlar. Bu işlemler için derin öğrenme, yapay sinir ağlarının belirlenmesinin ardından ağırlıklar ve bias değerlerinin ayarlanması ile yapılır. Bu parametre ayarlamaları sayesinde derin öğrenme işleminin performansı ve öğrenme hızı düzenlenmiş olur. Bu parametrelerin ayarlanma süreci ve yapay sinir ağının seçimi veri setine bağlı olarak ayarlanmaktadır. Derin öğrenme sürecinde öğrenme, veri setinin ağa verilmesi ile başlar. Ağa verilen veri seti genellikle üç başlık altında ayrılarak verilir. Bu başlıklar öğrenme, doğrulama ve test şeklindedir. Üç parçada ağa verilen veri seti, ağ üzerinde farklı amaçlarda kullanılır. Öğrenme verisi

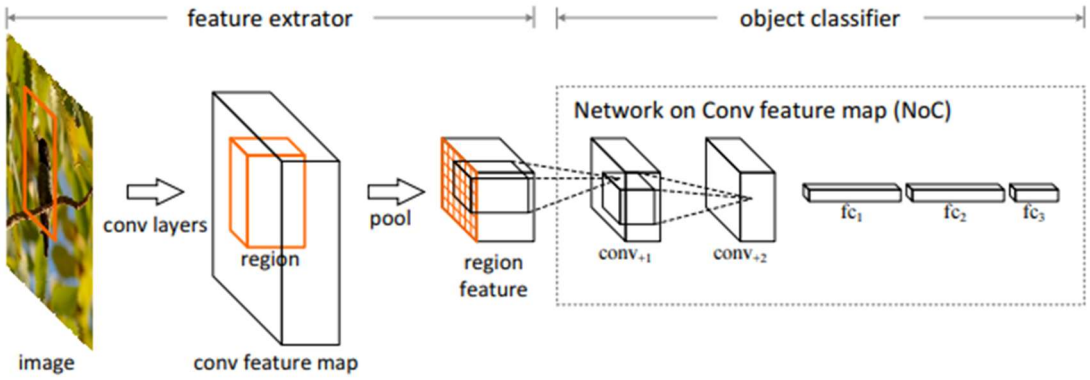
ağın ilgili durumu öğrenmesi için kullanılırken ağın öğrenme sürecinin denetlenmesi ve izlenmesi için doğrulama verisi kullanılır. Derin öğrenme sürecinde verinin ayrılması ve ağın denetimi sağlanarak model eğitimi sürecinde aşırı öğrenme veya az öğrenme durumlarının da denetimi sağlanabilmektedir. Ağ parametrelerine bağlı olarak öğrenme sürecindeki başarının denetlenmesi de test verileri üzerinden sağlanmaktadır. Derin öğrenme işleminde üçe ayrılan veriler farklı derin öğrenme yöntemlerine tabi tutularak ağın öğrenimi sağlanabilir. Bu derin öğrenme yöntemleri genel olarak girdi sağlayan verilerin özniteliklerinin çıkarılması ve öğrenme işlemini gerçekleştirir. Yapay sinir ağlarında çıktı almak için gerekli olan elle öznitelik girişi verme işlemi bu süreçte gerekli değildir. Bu nedenle ağın öznitelik çıkarımı her katmanda sağlanır ve bu belirlenen öznitelikler üst katmanlara aktarılarak detaylandırılır. Bu süreç akışı sayesinde yapay sinir ağı daha fazla öznitelik ile öğrenme sağlar ve ağın başarı sonuçları daha iyi hale gelir. Bu sayede derin öğrenme yöntemi geleneksel yapay sinir ağı öğrenmesine göre daha doğru parametreler ile çalışarak yüksek doğruluk elde etmektedir[30].

3. MATERYAL VE YÖNTEM

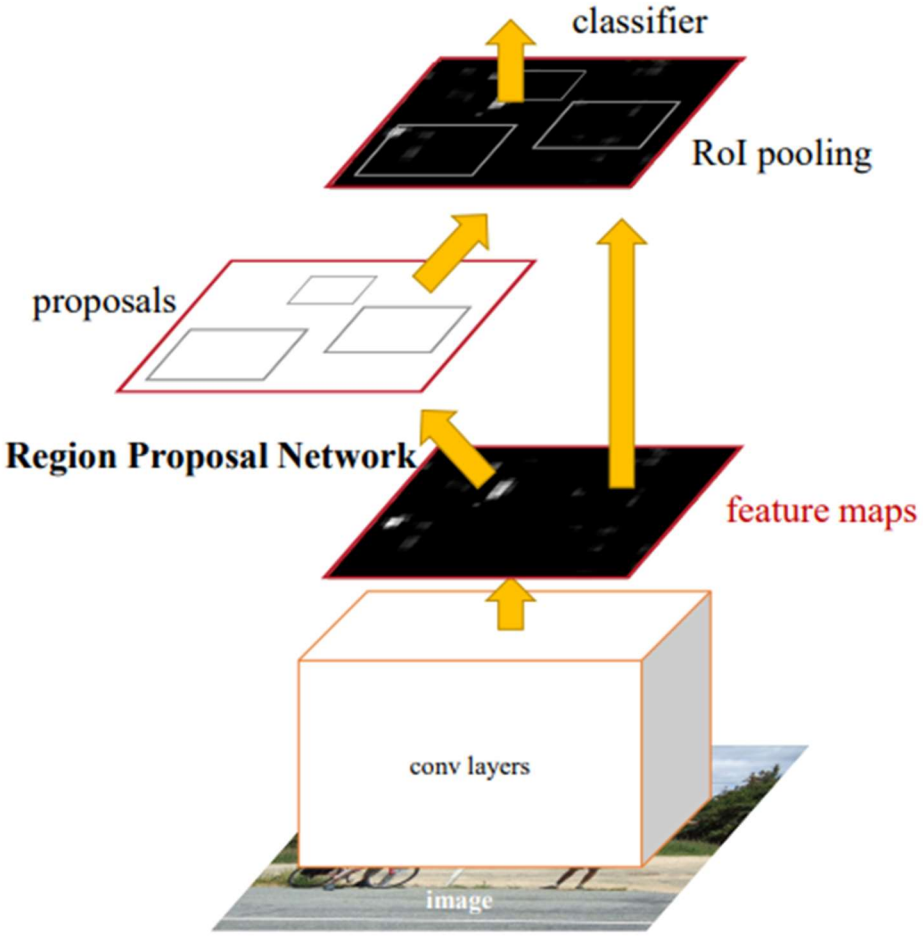
Nesne tespiti gibi görüntü üzerinden tespit sağlanan birçok uygulama bulunmaktadır. Bu uygulamaların genel yapısında da görüntü üzerinden öznitelik çıkarımı ve sınıflandırma işlemleri gerçekleştirilmektedir. Bu işlemler için farklı yapay sinir ağı metotları kullanılarak tespit ve çıkarım performansları iyileştirilmeye çalışılmaktadır. Bu süreç ise ilgili çalışma dizini veya veri setinin yapısına bağlı olarak ağların performansına göre değişmektedir. Özniteliklerin niteliği, tespit nesnesinin kriterleri ve çalışma ortamındaki değişiklikler modele bağlı performans kriterlerinin seçiminde aktif rol oynamaktadır.

3.1. FASTER R-CNN

Faster R-CNN modeli, literatürde nesne tespiti alanında yoğun kullanımı olan bir derin öğrenme modelidir. Bu model sırasıyla R-CNN, Fast R-CNN ve Faster R-CNN olarak gelişim göstermiştir. Modelin gelişimi sürecinde başarı değerleri artırılarak nesne tespitindeki kullanımı yoğunlaşmıştır. Model gelişiminin ardından çalışma süreci olarak öncelikle nesnenin oluşabileceği alanın belirlenmesi ve nesnenin tespit edilmesi aşamalarında çalışma göstermektedir. Bu aşamalarda bölge önerisi görüntünün denetlenmesinde tespit sürecini hızlandıran bir yaklaşım sağlamaktadır. Bu aşamada görüntü üzerinde ilgili tespit nesnesinin olası konumları selective search algoritması ile belirlenir. Selective search algoritması, olası konum koordinatlarını oluşturmak için görüntü piksellerinde ilgili noktaları oluşturan pikselleri gruplayarak bir alan belirler. İlgili alan belirlenene kadar algoritma bütün görüntü üzerinde piksel piksel çalışır ve piksel gruplarını tespit için çalışan algoritmaya yönlendirir. Nesne tespiti aşamasında ise ilgili piksel gruplarından bir öznitelik vektörü oluşturulur. Görüntünün tamamında öznitelik çıkarımının oluşturacağı işlem yükünün yerine ilgili alanları çalışması modelin hızını arttırmaktadır. Faster R-CNN ağında yer alan tespit ağı girdi olarak bölge koordinatları alırken çıktı olarak öznitelik vektörü oluşturur. Çıkarım sağlanan öznitelik vektörü ise model içerisinde yer alan sınıflandırma ağı kullanılarak sınıflandırma sağlar. Bu süreç içerisinde önceden eğitimli sınıflandırma ve tespit ağları yer almaktadır[31], [32].

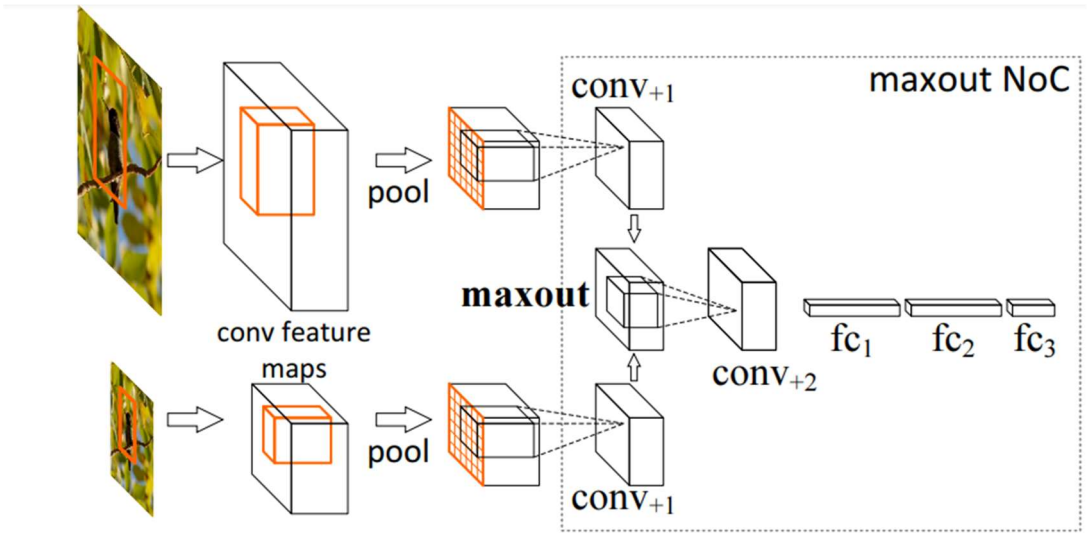


Şekil 3.1 Faster R-CNN Algoritma Çalışması[31]



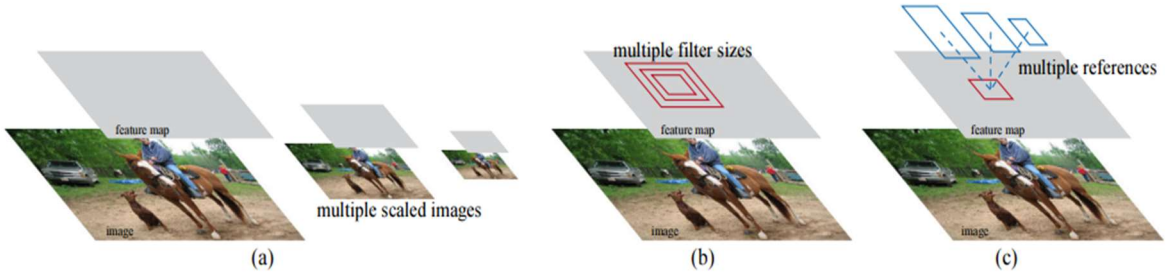
Şekil 3.2 Faster R-CNN Nesne Tespiti Süreci [32]

Şekil 3.1 ve Şekil 3.2'de gösterildiği gibi alan belirleme ve tespit vektörünün oluşturulmasının ardından sınıflandırma ağı üzerinde sınıflandırma işlemi gerçekleştirilir. Bu işlemlerin tamamlanması ile son işlem basamağı olan öznitelik vektörlerinden çıkarılan sınıf etiketlerinin konumlarının belirlenmesi sağlanır. Bu işlem her öznitelik vektörünün oluşturulduğu alan için gerçekleştirilir.



Şekil 3.3 Faster R-CNN Öznitelik Çıkarımı [31]

Şekil 3.3'te yer aldığı gibi Faster R-CNN modeli adım basamaklarını uygularken bölge önerisi ve nesne tespiti işlemlerini gerçekleştirerek ağların girdi ve çıktılarını birbirine bağlamıştır. Bu sayede model daha hızlı çalışarak daha doğru tahminler sağlamaktadır. Genel yapısı ve gelişim süreci itibariyle model daha önceden eğitimi sağlanan CNN modellerini kullanarak, özniteliklerin elde edilmesi ve tespitin sağlanması ile sürecini devam ettirmektedir.



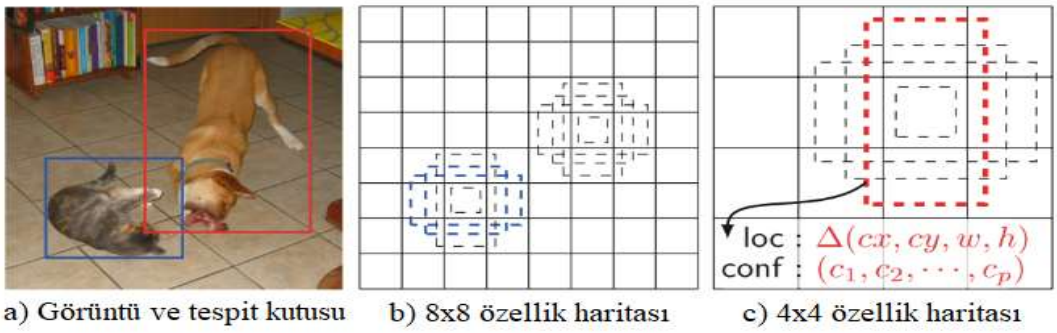
Şekil 3.4 Faster R-CNN Nesnelerin Belirlenmesi [32]

Şekil 3.4'te bu çalışma dizininin adım adım görsel üzerinden işlenmesi verilmiştir. Daha önceden eğitilen modele verilen görüntü girdisi öncelikle a'da görsel üzerinde selective search algoritması ile olası alanları belirler. Ardından b'de bu alan tespit algoritmasına aktarılır. Aktarılan alandan sınıflandırma algoritması ile özellikler çıkarılır ve nesne tespiti sağlanır.

Faster R-CNN Modeli gelişim sürecinde diğer R-CNN modellerine göre test süreci ImageNet (İmaj Veritabanı Projesi) ve COCO (Bağlam İçinde Ortak Nesneler) veri setleri üzerinden sağlanmıştır. Performans kriterlerinin karşılaştırılması da bu veri setleri üzerinden sağlanarak daha performanslı olduğu belirlenmiştir.

3.2. SSD RESNET

Nesne tespiti için geliştirilen SSD yöntemi diğer yöntemlere kıyasla farklı ölçeklerde yer alan öznitelik haritalarını kullanır. Bu yöntem ile daha doğru ve daha hızlı işlem kabiliyetleri sağlamaktadır. SSD modeli nesne tespitinde daha hızlı ve daha doğru sonuçlar üretebilmek için ResNet modelini temel almaktadır. SSD ResNet, bu temel alınan ResNet modeli ile SSD'yi beraber kullanmaktadır. Bu model, ResNet modelini model çıktılarının doğruluğunu ve çalışma hızını artırmak için kullanılmıştır. Bu sayede SSD modeline göre daha yüksek performans değerine sahip olmuştur. Faster R-CNN modelinde olduğu gibi SSD ResNet modeli, ImageNet ve COCO veri setleri ile test edilerek performans parametreleri açısından değerlendirilmiştir. Bu modelin iyileşmesini sağlayan ResNet modeli, ağın her katmanında oluşturulan özniteliklerin bir sonraki ağa aktarılması ile SSD modelini derinleştirmektedir[33], [34]. Bu derinleşen model sayesinde SSD ResNet modeli nesne tespiti işleminde farklı boyutlardaki nesneleri de tespit edebilir hale gelmektedir. Bu durumun oluşması ile model boyutlandırmadan bağımsız olarak nesne tespiti sağlayabilme becerisine de sahip olmuştur. Farklı boyutlardaki nesneleri algılamak için model öznitelik çıkarımı sırasında çeşitli boyutlarda öznitelik haritaları oluşturmaktadır. Oluşturduğu bu öznitelikler üzerinde farklı boyutlarda kaydırma işlemini gerçekleştirir ve her kaydırma konumu için boyutları farklı olan nesne önerileri oluşturmaktadır. Bu işlemin ardından bir sınıflandırma ağı ile bu nesnelerin sınıflandırılmasını ve konumunu sağlayarak tahmin oluşturur.



Şekil 3.5 SSD Nesne Tespiti [34]

Şekil 3.5'te SSD'nin görüntü boyutları üzerinde çalışması daha net gösterilmektedir. Elde edilen özniteliklerin haritalandırılmasının ardından öznitelik haritaları b ve c'deki gibi farklı boyutlandırmaları kaydırılır ve tespit sağlanır. Bu haritalar sınıflandırıcı ile sınıflandırılarak nesnenin konumu ve sınıfı kesinleştirilir ve a'daki gibi net çıktı elde edilir.



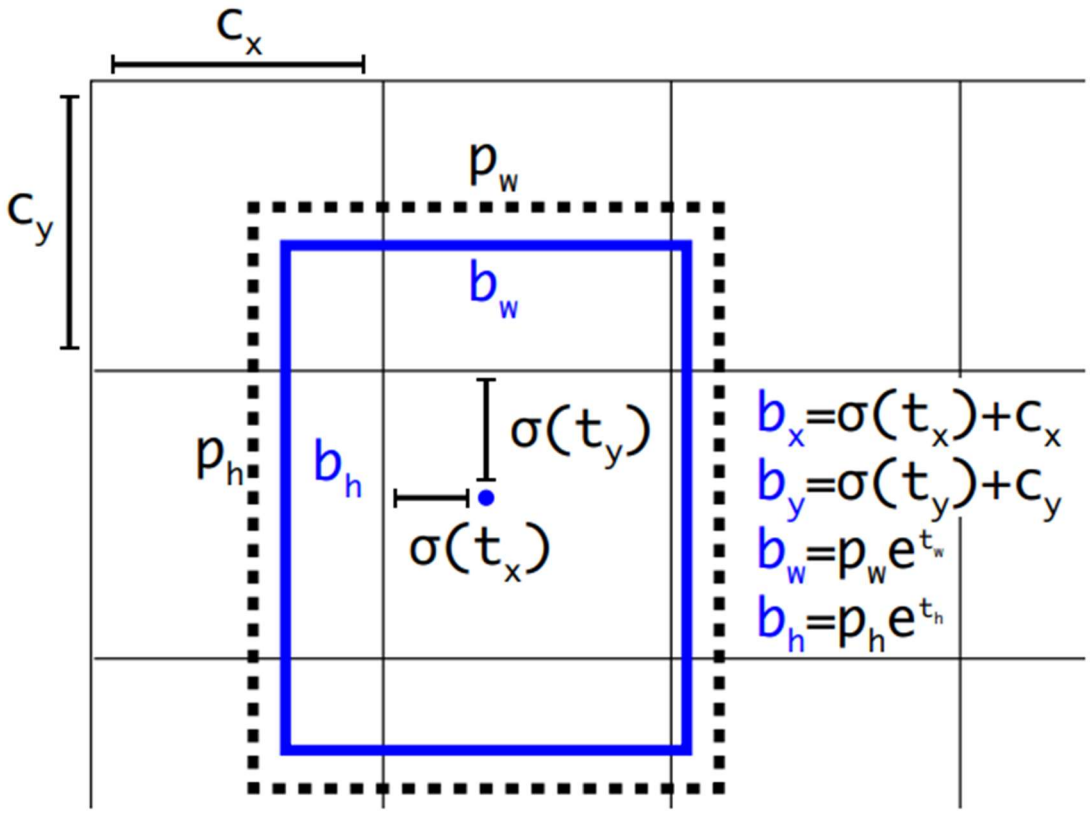
Şekil 3.6 COCO Veri Seti Üzerinde SSD Nesne Tespiti [34]

Şekil 3.6'da SSD modelinin COCO veri seti üzerindeki testleri gösterilmektedir. Bu test sonuçlarına da bağlı olarak modelin nesne boyutu ve karmaşıklığına bağlı olarak performans kriterlerinde başarılı olduğu gösterilmiştir.

3.3. YOLO

Nesne tespitini ve sınıflandırma konusunda oluşan problemler ve açıklıkları çözmek amacıyla YOLO derin öğrenme modeli geliştirilmiştir. Bu konuda etkili bir çözüm sağlayan model nesne tespiti ve sınıflandırma yaparken modele verilen görüntüyü bir bütün olarak işlemektedir. Modelin bu yöntemi üzerine sağladığı en önemli özelliği görüntüyü küçük parçalar halinde parça parça işleme alması yerine tek seferde görüntüyü işleme almasıdır. Genel yapısında CNN modelini temel alan YOLO, modeli birkaç konvolüsyon katmanından oluşan ağın girdisi olarak görüntüyü vermektedir. Ardından görüntüyü işleyerek nesne tespiti ve sınıflandırma işlemini gerçekleştirir. Görüntü

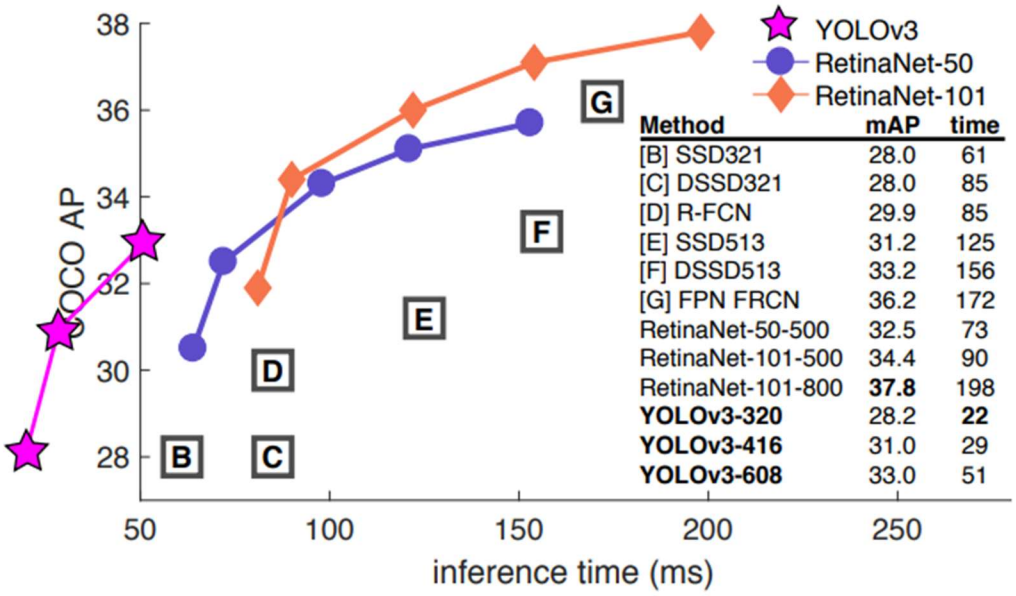
üzerinde yer alan nesne veya nesneler için bölgeleri belirleyerek CNN katmanlarında işlenmesini sağlar. Oluşturulan bölge, nesnenin sınırlarını içerisinde barındırdığı bir dikdörtgen alan olarak belirlenir. YOLO modeli bu belirlenen bölge için olasılıkları hesaplar ve bu bölgedeki nesnenin olasılık değerini işleyerek sınıflandırılmasını sağlar [34], [35].



Şekil 3.7 Nesne Alanı Belirleme[35]

Şekil 3.7'de görüntü üzerinde yer alan nesne için bir sınır belirleme işlemi sağlanmıştır. YOLO modeli bu işlem basamağı olarak tespit işleminin başlangıcında bu işlemi gerçekleştirir. Elde edilen nesne sınırları içerisindeki öznitelikleri CNN katmanında işleme alarak sınıflandırılmasını sağlamaktadır.

YOLO modelinin genel yapısı ve çalışma performansındaki başarısı nesne tespiti ve sınıflandırılması yapmadan önce görüntüyü parçalamak yerine görüntüyü tek seferde işleme alması sayesindedir. Bu işlem basamakları diğer modellerde nesne barındıran bölgelerin parça parça işlenerek ilgili alanların modelde sırasıyla işlemesi ile gerçekleşmektedir. Ancak YOLO modeli, görüntüyü bir bütün halinde alır ve genel görüntüdeki nesne barındıran bölgeleri belirleyerek işler. Bu sayede diğer modellere göre hızlı bir işlem sağlamaktadır.



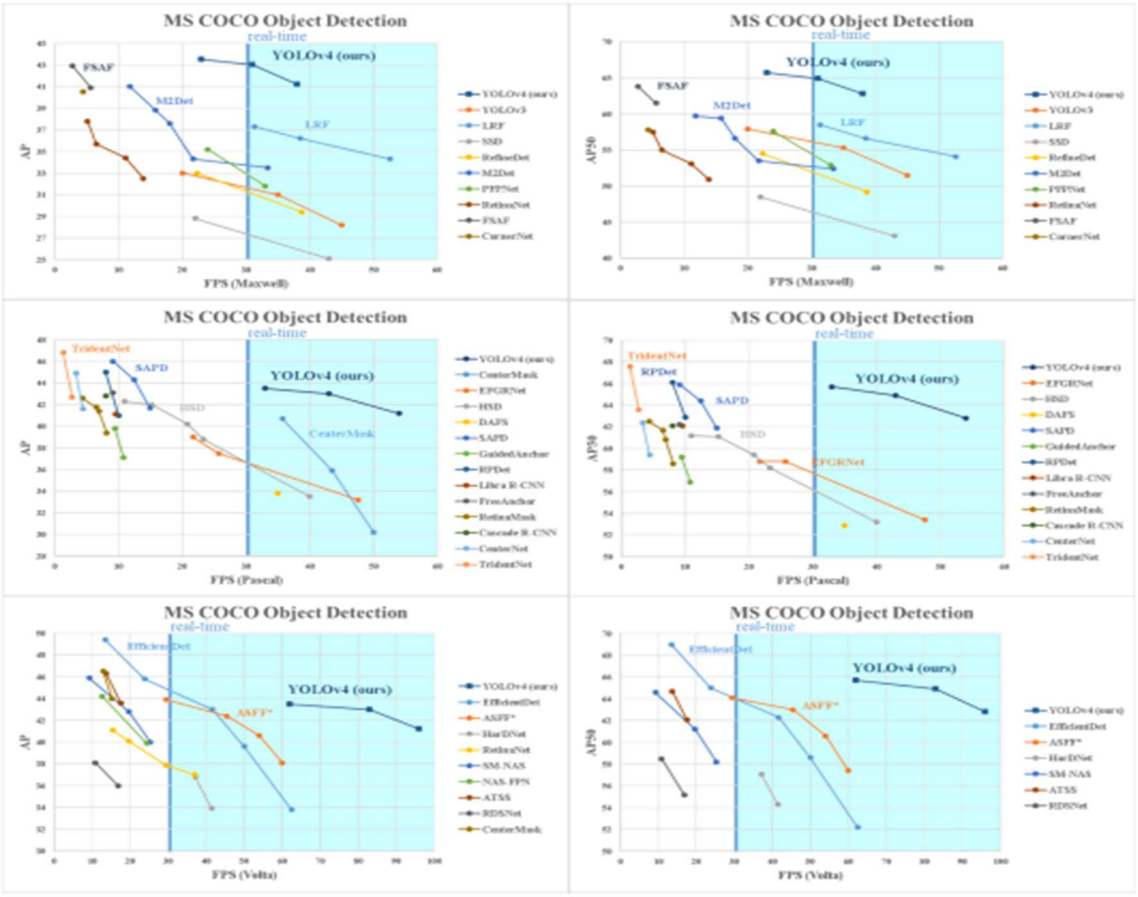
Şekil 3.8 COCO Veri Setinde YOLO Performansı [35]

Şekil 3.8'de işlem kabiliyetleri açısından YOLO modelinin karşılaştırılması sağlanmıştır. SSD ve R-CNN gibi temel modellere karşı COCO veri seti üzerinde işlem süresi ve başarısı gösterilmiştir. Bu açıdan YOLO modeli daha performanslı bir işlem sağlamaktadır.

3.3.1. YOLOv4

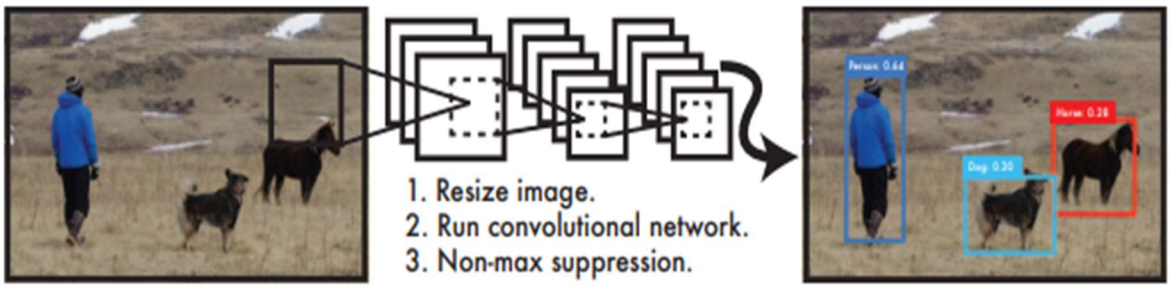
YOLOv4 nesne tespiti için geliştirilmiş bir derin öğrenme modelidir. Bu model YOLO temellerini alan girdi görüntülerini gridlere bölerek tek seferde nesne veya nesnelerin koordinat, boyut ve sınıf özelliklerini çıkarır. Temel YOLO modellerine kıyasla nesne tespiti konusunda hız ve doğruluk parametreleri iyileştirilmiştir. Bu iyileştirme için v4 modelinde farklı teknikler çalışılmıştır. Bu teknikler, CSP Darknet 53 mimarisi, Mish aktivasyon fonksiyonu, SPP (Uzaysal Piramit Havuzu), PAN (Kişisel Alan Ağı) ve farklı katman bağlantıları gibi tekniklerdir. Bu teknikler arasında CSP Darknet 53 (İçerik Güvenlik Politikası) mimarisi, modelin daha az parametre konfigürasyonu ile başarısını arttırmaktadır. Mish aktivasyon fonksiyonu, önceki YOLO modellerinde yer alan ReLU (Doğrultulmuş Lineer Birim) aktivasyon fonksiyonuna göre daha az hata ile daha iyi sonuçlar vermektedir. Farklı boyutlardaki nesnelerden öznitelik çıkarımı konusunda SPP blokları daha iyi sonuçlar çıkarmaktadır. Aynı zamanda PAN blokları ise modelin öğrenme sürecinde ölçekli olarak öğrenebilirlik katmaktadır. Bunların yanı sıra yeni yol bağlantıları, katmanlarda çıkarılan öznitelikleri üst katmanlara aktarırken daha başarılı bir aktarım gerçekleştirmektedir [36]–[38].

YOLO modelleri arasında v4 modelinin bu konulardaki gelişimleri, diğer derin öğrenme modellerinin de gelişimine bir yol verebilmektedir. YOLOv4 modelinin bu gelişimi ile nesne tespiti için hız, ölçeklenebilirlik ve yüksek doğruluk elde etmiştir.



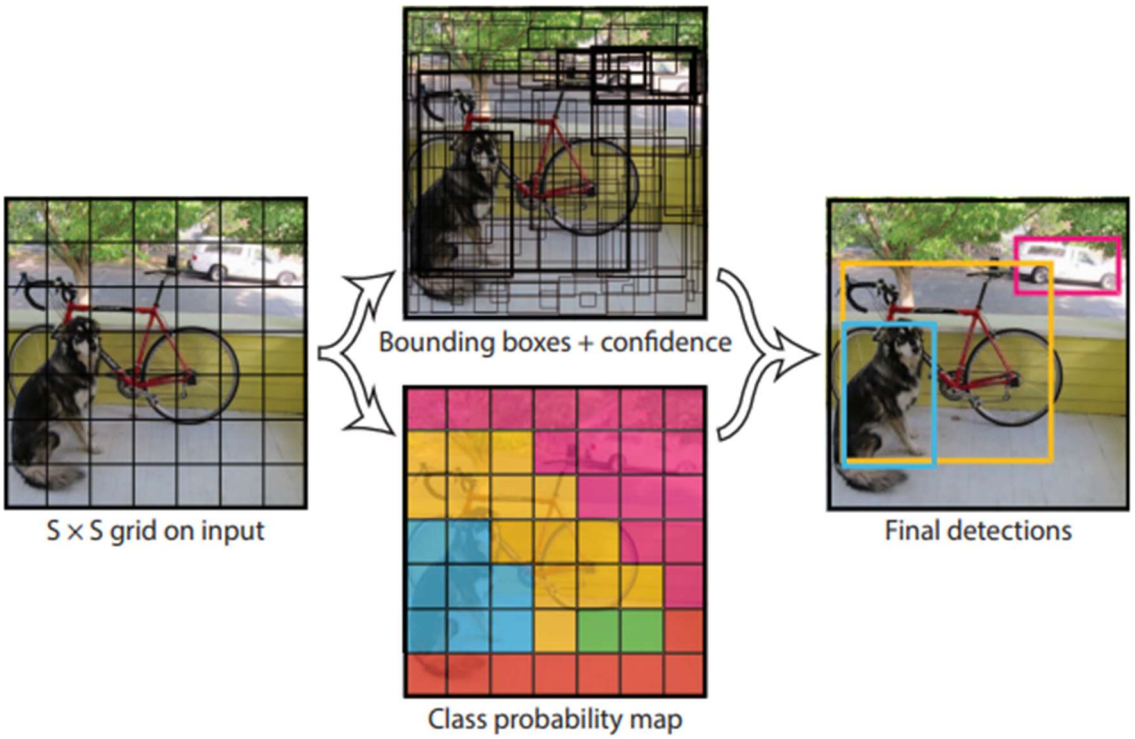
Şekil 3.9 COCO Veri Setinde YOLOv4 Performansı [36]

Şekil 3.9'da yer alan COCO veri seti üzerindeki çıktıları modelin başarı kriterlerini de ortaya çıkarmıştır.



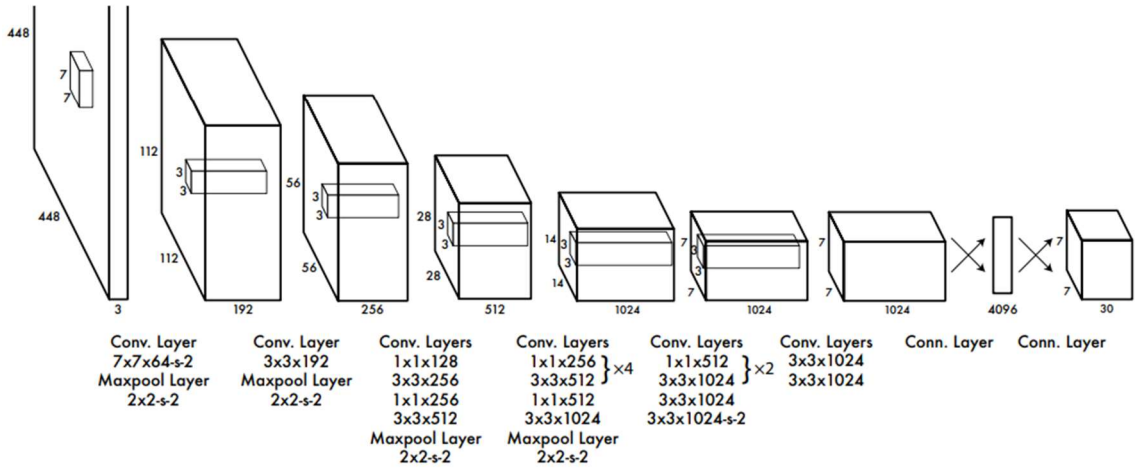
Şekil 3.10 YOLOv4 Nesne Tespiti İşlemi[37]

Şekil 3.10'da bu gelişim sürecinde temel YOLO modellerindeki gibi bir görüntü girdisi olduğu gözlemlenebilir. Bu sayede modelin genel özelliği olan görüntüyü tek seferde işleme kabiliyeti devam ettirilmiştir. Bununla beraber görselde yer alan nesnelerin koordinat, büyüklük ve öznitelik çıkarımı konusundaki gelişimi de yer almaktadır.



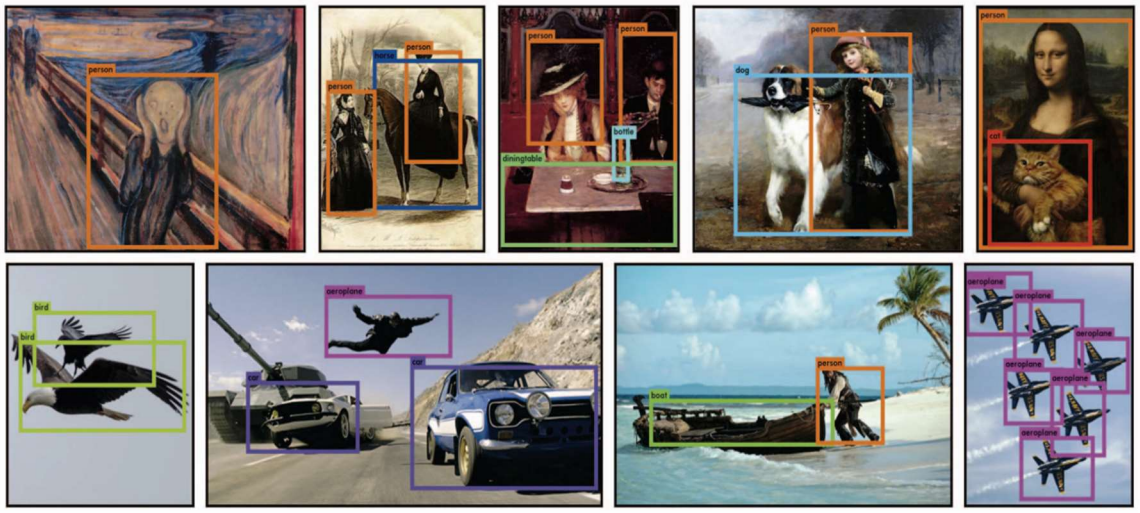
Şekil 3.11 YOLOv4 Nesne Tespiti İşlem Süreci[37]

Şekil 3.11'de ise girdi olarak modele verilen görüntünün gridlere ayrılmasını ve bu gridler üzerinden tespit nesnesinin alan belirleme işleminin görsel olarak aktarımını sağlamaktadır. Bu sayede birden fazla nesnenin yer aldığı görüntülerde model öznitelik çıkarımı için nesnenin yer aldığı koordinatlara odaklanmaktadır. Bu odaklama ile nesnenin daha başarılı ve daha hızlı tespiti sağlanır.



Şekil 3.12 YOLOv4 CNN Katmanları [37]

Şekil 3.12'de model içerisinde görüntünün girdi olarak verilmesinin ardından gerçekleştirdiği süreçlerin mimarisini göstermektedir.

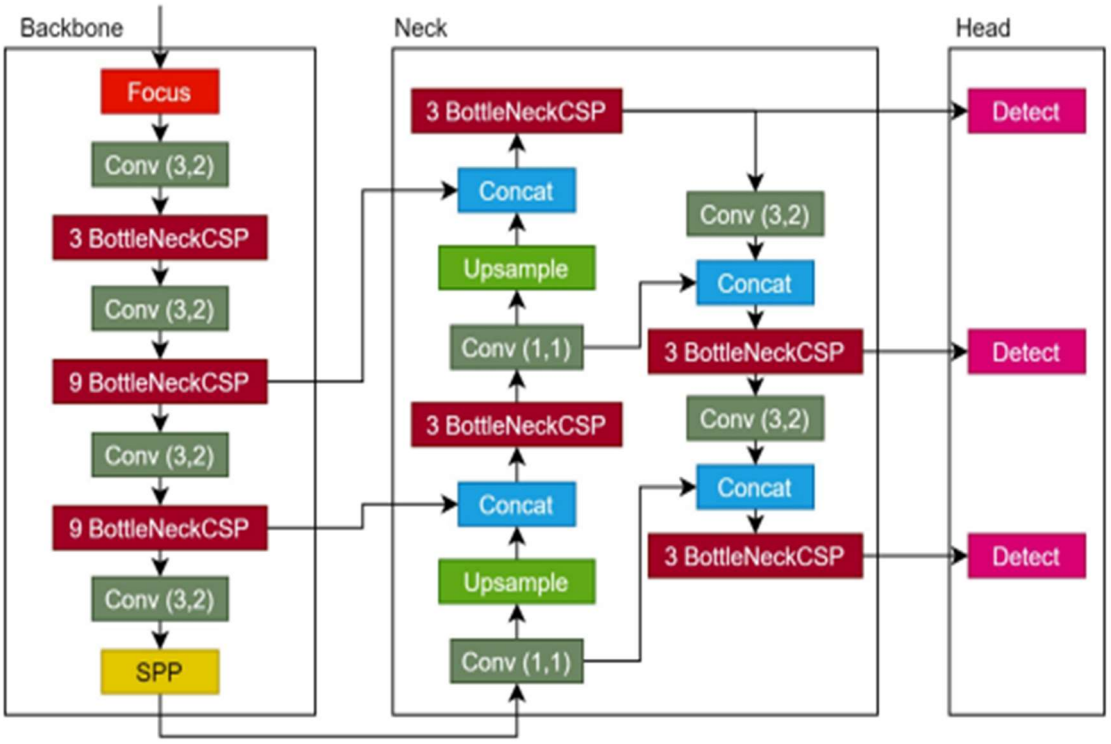


Şekil 3.13 YOLOv4 Modelinin Çeşitli Görsellerde Çalışma Performansı [37]

Şekil 3.13'de yer alan görsel içerisinde YOLOv4 modelinin çalışma performansı test süreçlerinde yer alan görseller ile aktarılmıştır. Bu görsellerde yer alan birden fazla nesne veya iç içe yer alan nesneler ele alındığında modelin karmaşık yapılarda da iyi bir performans gösterdiği çıkarılmaktadır. Aynı zamanda görüntü girdisi verilen modelin başlangıçta yer alan SPP bloklarının çalışma başarısı sayesinde çıktılar daha başarılı bir şekilde elde edilmiştir. YOLOv4 modelinde iyileştirmeler ile modelin işlem kabiliyetleri geniş bir şekilde ele alınarak artırılmıştır.

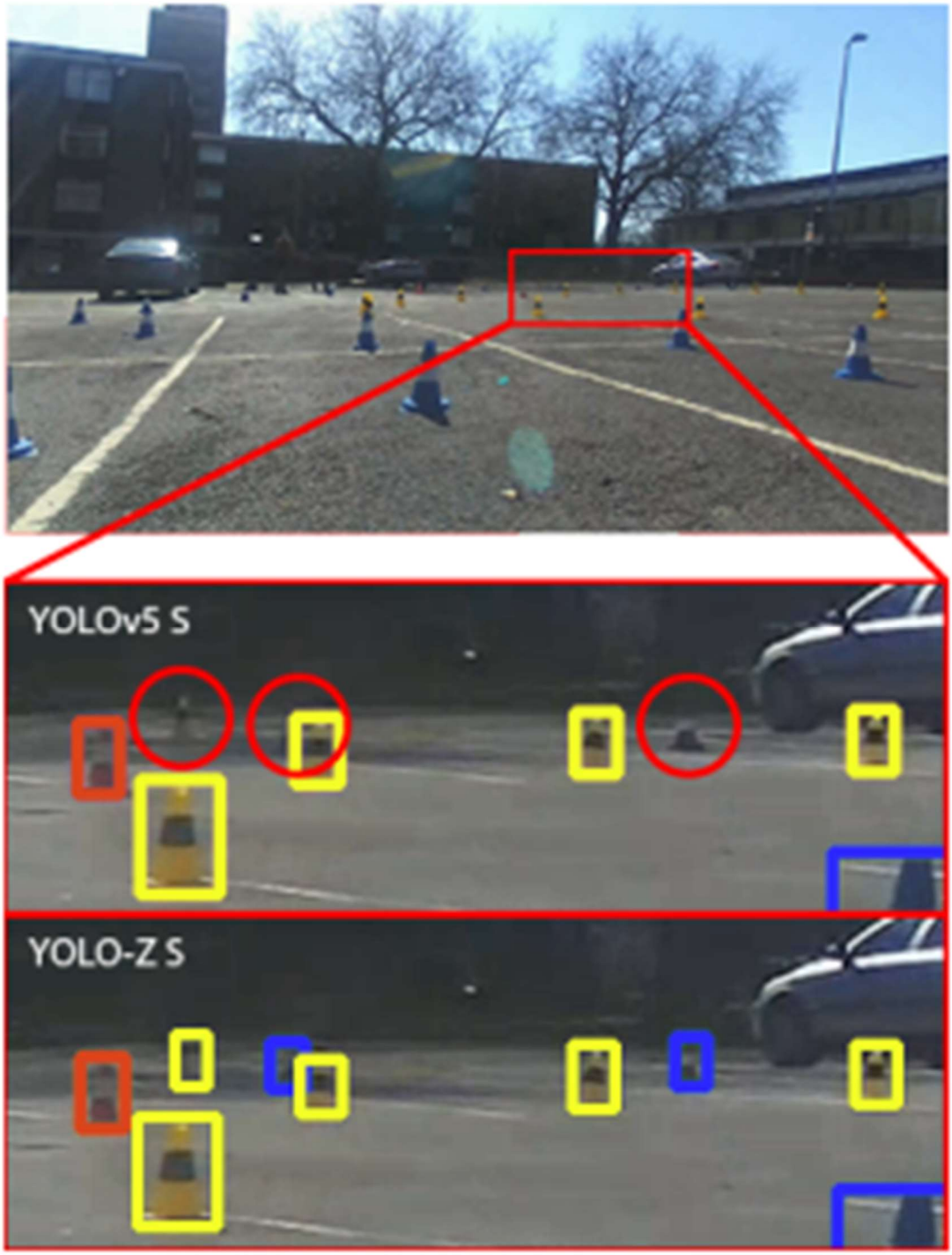
3.3.2. YOLOv5

YOLOv5 modeli önceki YOLO modellerine göre iyileştirmeler yapılarak çıkarılmıştır. Bu iyileştirmelerin başlıca olanı modelde yer alan CNN yapısının yeni bir yapı oluşturularak kullanılmasıdır. Bu değişim sürecinde YOLO modeli iki farklı modelde sunulmuştur. Bu modeller, daha düşük işlem performansı ile çalışan YOLOv5s ve daha geniş bir çalışma gösteren YOLOv5x'tir. İki modelinde temelini oluşturan v5 modeli girdi verilerinin zenginleştirilmesi ve ön işleme teknikleri eklenmesi ile performansı artırarak modelin eğitim sürecinde aşırı öğrenmesini azaltmıştır. Bunların yanı sıra v5 modeli, önceki modellere ek olarak yeni optimizasyon teknikleri ile de zenginleştirilmiştir[39].



Şekil 3.14 YOLOv5 Katmanları [39]

Şekil 3.14'de yer alan YOLOv5'in temel yapısına bakıldığında ana CNN katmanlarında eklemeler ve çıkarmalar yapılarak değişiklikler yapıldığı görülmektedir. Bu değişiklikler sayesinde model önceki versiyonlarından daha performanslı ve kendi kendini iyileştiren bir yapı kazanmıştır. Bu değişimler ile model, gerçek zamanlı çalışmalar için daha uygun bir yapı da kazanmıştır. Model üzerinde yapılan değişiklikler sayesinde v5 versiyonu, eğitim sürecinden nesne tespiti sürecine kadar daha hızlı çalışma sağlayabilir hale gelmiştir.



Şekil 3.15 YOLOv5 Modelleri Arasındaki Tespit Performansındaki Farklılıklar[39]

Şekil 3.15'te yer alan v5 modelinin çıktıları nesne tespiti sürecindeki gelişimi doğrudan çıkarmaktadır. Bu gelişim sayesinde ölçekli olarak yer alan nesneler başarılı bir şekilde tespit edilebilir hale gelmiştir. Ancak bu gelişimde oluşturulan v5s ve v5x modellerinin çıktıları arasında maliyetli bir çalışmaya sahip olan v5x modeli yüksek başarı elde etmiştir. Gerçek zamanlı çalışmaya olanak sağlayan YOLOv5 genel yapısında elde ettiği performanslı çalışmayı da yanında taşımıştır.

YOLOv5 modeli, gelişim ve değişimler ile önceki modellerine kıyasla daha düşük maliyetle hızlı ve doğru nesne tespiti sağlamaktadır. Yapısında kullandığı yeni CNN modeli ile v5s modelinde tespit başarısını artırmıştır. Bu sayede gerçek zamanlı çalışmalarda da kullanıma yüksek bir olanak sağlamıştır. Aynı zamanda v5 modelinin kendi içerisinde veri zenginleştirme ve ön işleme teknikleri ile genel çalışma performansına katkı sağlamıştır. Bütün iyileştirmeler sayesinde YOLOv5 modeli gerçek zamanlı uygulamalarda aktif kullanılabilecek yapıya gelirken performansını da artırmıştır.

4. VERİ SETİ

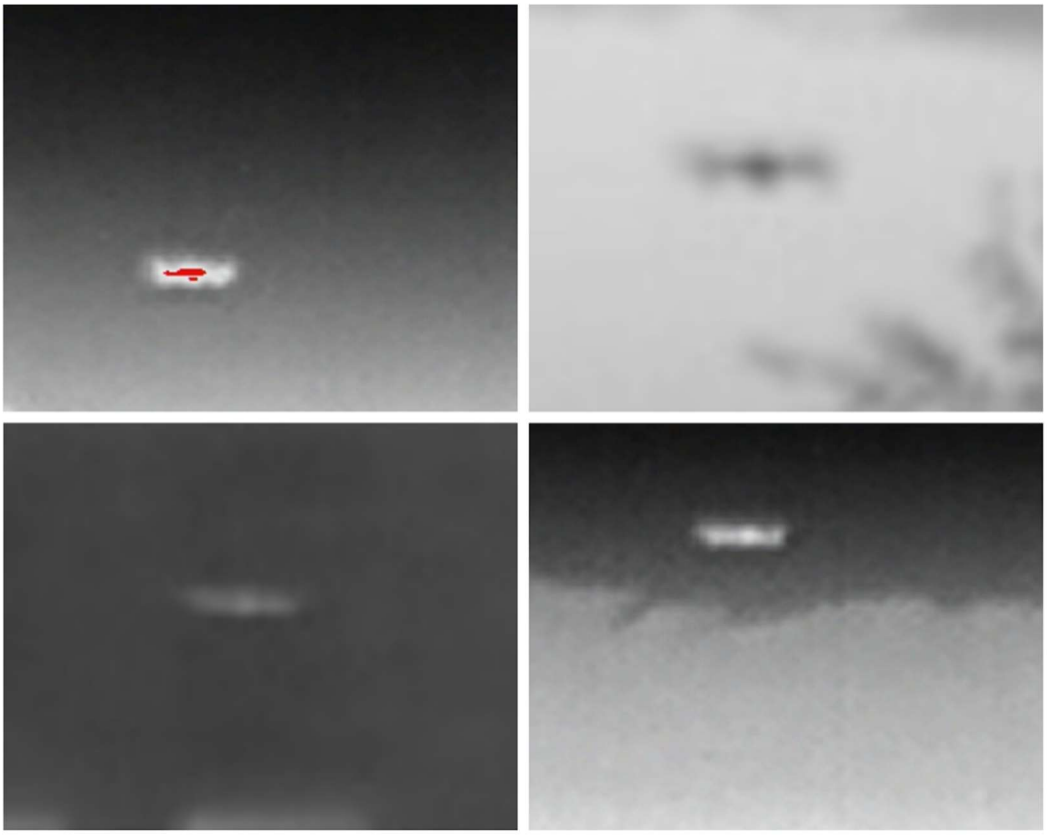
Veri seti, çalışma sürecinde en önemli parametrelerden biridir. Bu süreçte veriler, ortam şartlarına uygun güncel veriler üzerinde çalışmalar sağlanması amacıyla çalışma ekibi tarafından oluşturulmuştur. Veri setinin toplanması aşamasında tek merkez üzerinde odaklanan hem termal hem de standart kamera ile görüntüler toplanmıştır. Bu verilerin toplanması esnasında kamera özelliklerine bağlı olarak termal kamera üzerinde alınan görüntü çıktısı 480x720 iken standart kamerada 1080x1920 boyutlarındadır. Aynı zamanda farklı arka plan karmaşıklıkları içerisinde drone nesnesinin görüntüleri alınmıştır. Çeşitli alanlarda toplanan standart kamera veri seti örnekleri Şekil 4.1 ve Şekil 4.2'de gösterilirken termal kamera veri set örnekleri Şekil 4.3 ve Şekil 4.4 'te gösterilmiştir. Veri seti oluşturulurken araştırma konusuna uygun olarak farklı senaryo ve koşullara uygun veriler dikkat edilerek veri setine eklenmiştir. Bu senaryoların başında arka planda ağaçların yer aldığı, binaların veya karmaşık bulut yapılarının bulunduğu ve inşa aşamasındaki yapıların bulunduğu senaryolara yer verilmiştir. Bu alanlarda termal veya standart kameraların tespit için öznitelik konusundaki eksiklikleri de çalışmanın ilerleyen kısımlarında işlenecektir. Bu çeşitli ortam ve koşullardaki görsellerle desteklenen veri seti ile yapay zekâ modelinin aşırı öğrenme yaşamadan eğitiminin sağlanması amaçlanmaktadır. Aynı şekilde hedeflenen çıktı için veri seti içerisindeki çeşitliliğin arttırılması ve drone benzeri nesnelerde ağın başarısının devamlılığı sebebiyle OpenImage veri setinden "Kuş" ve "Uçak" sınıflarına ait veriler çekilerek veri setine eklenmiştir[40], [41].



Şekil 4.1 Standart Kamera Verilerindeki Drone Görüntüleri



Şekil 4.2 Farklı Arka Planlarda Standart Kamera Verilerindeki Drone Görüntüleri



Şekil 4.3 Termal Kamera Verilerindeki Drone Görüntüleri



Şekil 4.4 Farklı Arka Planlarda Termal Kamera Verilerindeki Drone Görüntüleri

Şekil 4.1 ve Şekil 4.2'de standart kamera görüntülerinden örnekler verilirken, Şekil 4.3 ve Şekil 4.4'te ise termal kamera görüntülerinden toplanan görüntülerden örnekler verilmiştir. Toplanan bu veri setinden çıkarılan görsellerde drone nesneleri yer almaktadır. Etiketleme esnasında bu verilerin olduğu görseller seçilmiş ve net olarak [17] çalışmasında öne çıkarılan kaymış kutu, ek kutu gibi hatalı durumların olmaması için dikkatle etiketlenmiştir. Bu sayede veriler doğru etiketler ile doğru koordinatlar içerisindeki nesnelerden oluşturulmuştur. Bu işlem ile eğitim sürecinde modellerin etiket koordinatlarından yapacağı öznitelik çıkarımları daha doğru bir biçimde sağlanacaktır.

Termal ve standart kameralar ile video formatında verilerin toplanmasının ardından ilk işlem olarak videolardan resimler çıkarılmıştır. Elde edilen resimler incelenerek tespit nesnesi olan droneler gibi nesnelerin görüntü karesinde yer aldığı resimler seçilmiştir. Seçilen görüntüler ile yapay zeka modellerine girdi olarak verilecek görseller hazır hale getirilmiştir. Bu elde edilen görsellerin modele verilebilmesi ve eğitim işleminin gerçekleştirilebilmesi için tespit nesneleri açık kaynak olarak sunulan LabelImg yardımı ile etiketlenmiştir. Görsellerin etiketlenmesi esnasında test edilecek modeller arasında yer alan YOLO temelli etiketleme formatı ile etiketleme yapılmıştır. Aynı etiketleme verileri, github üzerinde Yolo2Pascal-Annotation-Conversion çalışması yardımı ile Faster R-CNN gibi Pascal VOC etiket formatında girdi verisi gereken modeller için ilgili etiket formatına çevrilmiştir. Etiket sürecinde veri setindeki görseller ilgili tespit nesneleri olan drone, kuş, helikopter gibi nesne etiketleri ile etiketlenmiştir. Bu sayede ilgili veri seti farklı derin öğrenme modellerinde eğitilmek üzere hazır hale getirilmiştir.

Veri setinin eğitime hazır hale getirilmesi ile veri seti üç parçaya ayrılmıştır. Bu ayrım literatür araştırmasında,[15] çalışmasından yola çıkarak %70 eğitim, %20 doğrulama ve %10 test verisi olacak şekilde düzenlenmiştir. Bu aşamada verilerin ayrılması, etiket verilerinin farklı formatta geçişlerinin sağlandığı gibi python kodu ile sağlanmıştır. Bu sayede veriler doğru oranlarda doğru şekilde makine tarafından gerçekleştirilmiştir. Bu aşamada verilerin video şeklinde farklı senaryo ve ortamlarda toplandığı için sıralı bir şekilde oluşmuştur. Oluşan durum her durumda çekilen videonun çıktılarının sıralı oluşumundan meydana geldiği anlaşılarak görseller grupsuz olarak rastgele karıştırılarak eğitim, doğrulama ve test görselleri dağınık bir şekilde oluşturulmuştur. Bu sayede doğrulama ve test kısmında kalan senaryo görselleri üzerinde de modellerin eğitimi sağlanabilir hale gelmiştir.

4.1. MODEL EĞİTİMİ

Elde edilen veri seti modellerin eğitimi için %70 eğitim, %20 doğrulama ve %10 test verisi olacak şekilde ayrılmıştır. Eğitim sürecinde modellerin eğitimine sadece eğitim ve doğrulama verileri verilmiştir. Kullanılan her model için iki eğitim yapılmıştır. İlk eğitim normal görüntüler ile ikinci eğitim ise termal görüntüler ile yapılmıştır. Eğitimler tamamlandıktan sonra eğitim sonuçları %10 oranında ayrılan test görselleri ile test edilerek doğruluk ve fps değerleri ön planda olacak şekilde karşılaştırılmıştır. Bu önceliğinin sebebi ise doğruluk değerleri modelin ilgili nesneyi veya nesneleri tespit edecek şekilde yapmış olduğu tahminlerin başarısını göstermesidir. Fps değeri, tespiti sağlayan modelin eğitim sonucundaki çalışma hızını göstermektedir. Bu parametreler sayesinde yapılan eğitim sonuçları değerlendirilebilecektir. Eğitilen modeller bu parametreler üzerinden kıyaslanacaktır.

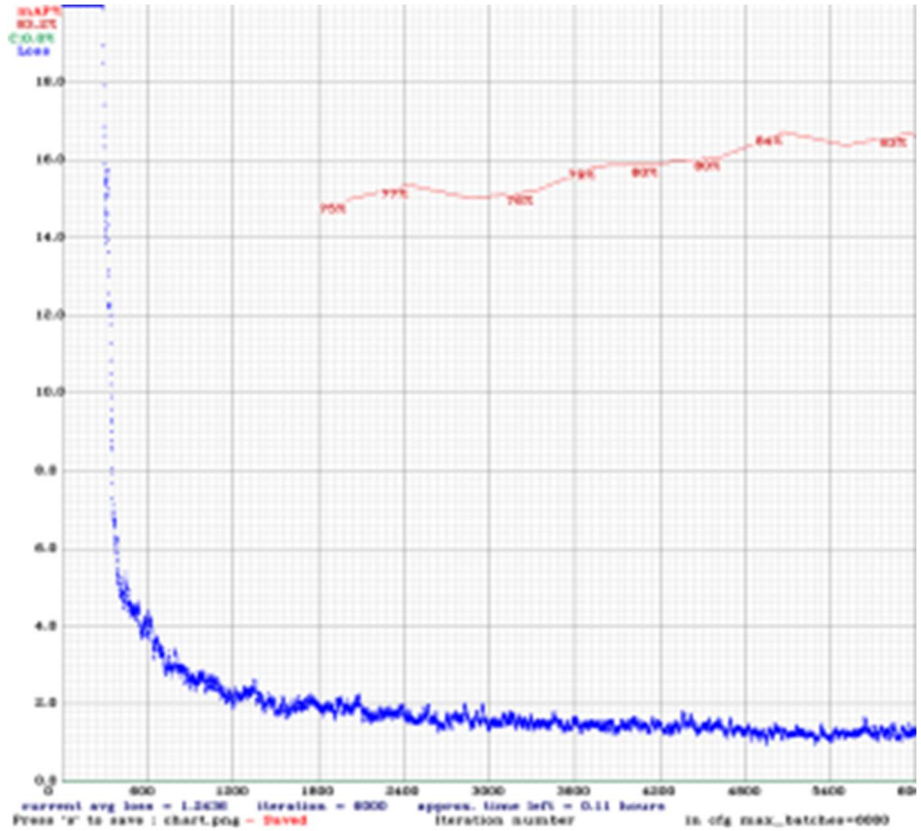
4.2. DEĞERLENDİRME

Çalışma sürecinde Faster R-CNN, SSD ResNet ve YOLO modelleri ile yapılan eğitimler drone nesnesinin tespiti için test edilmiştir. Eğitilen modeller Fps ve doğruluk oranı parametreleri üzerinden değerlendirilmiştir. Yapılan test sürecinde İlgili modellerin doğruluk değerleri ve fps değerleri aşağıdaki çizelgede verilmiştir. Bu sonuçlar ele alındığında performansı yüksek ve fps değeri üzerinden seçim ilgili nesnenin tespit senaryolarına bağlı çalışma modeli seçilmiştir. Seçim aşamasında Faster R-CNN modeli yüksek doğruluk ile tahmin sağladığı görülmektedir. Ancak bu model için tahmin süresi daha yavaş ve çalışma maliyeti daha yüksektir. Ardından Yolov4 modeli ise bu değerler arasında tahmin süresi ve doğruluğu açısından daha verimli bir model olduğu çıkarılmıştır. Bu süreçte YOLOv4 modelinin yapısında ölçeklenebilirlik ve farklı öznitelik katmanları ile daha fazla öznitelik ile nesnenin kestirim yapabilir olduğu da gözlemlenmiştir.

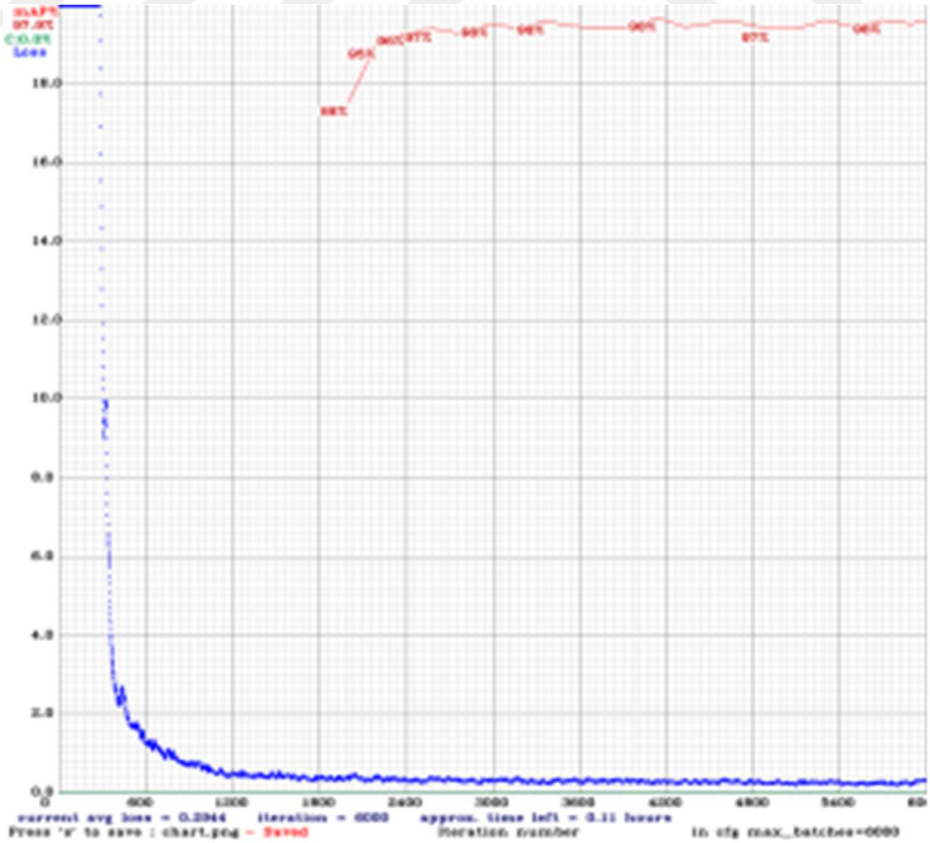
Çizelge 4.1. Model Değerleri

Model	Doğruluk	FPS
Faster R-CNN	0,90	7
SSD ResNet	0,75	15
YOLOv4	0,85	25-30
YOLOv5	0,83	23-28
	0,85	**25-30**

Değerlendirme sürecinde ele alınan doğruluk ve fps değeri bakımından verimli model olarak seçilen YOLOv4' ün termal ve standart görüntüler üzerinden yapılan eğitim sonucundaki mAP ve loss grafikleri çıkarılmıştır. Şekil 4.5'te standart kamera görüntülerinden yapılan eğitim grafiği yer almaktadır. Şekil 4.6'da ise termal kamera görüntülerinden yapılan eğitim sürecindeki model grafiği yer almaktadır. Modelin eğitimi esnasında yapılan iki eğitimine de verilen görüntü sayıları yakın değerlerdedir. Buna rağmen görüntü boyutu ve çeşitlilik gibi durumların oluşturduğu farklılıklar nedeniyle termal görüntüler üzerinde yapılan eğitimde loss değerinde hızlı azalış ve mAP değerinde hızlı bir artış gerçekleşmiştir.



Şekil 4.5 Standart görüntüler ile yapılan eğitimin loss ve mAP değerleri



Şekil 4.6 Termal görüntüler ile yapılan eğitimin loss ve mAP değerleri

5. DENEYSEL SONUÇLAR

Standart ve termal kamera ile toplanan görüntüler, literatürde nesne tespiti alanında yaygın kullanımı olan Faster R-CNN, SSD ResNet, YOLOv4 ve YOLOv5 modelleri üzerinde eğitilmiştir. Bu modeller üzerinde yapılan eğitim sonuçlarındaki fps ve doğruluk değerleri incelenerek karşılaştırılmıştır. Modellerin eğitim sürecinde verilen veri setinde yer alan iki tür görüntü kaynağı içinde YOLOv4 modelinin başarılı olduğu belirlenmiştir. Veri seti ile eğitimi tamamlanan ve başarılı olan YOLOv4 modelinin test aşamasındaki çıktıları Şekil 5.1 ve Şekil 5.2'de verilmiştir. Verilen bu görseller üzerinde genel yapı olarak arka plan yoğunluğunun yüksek olmadığı ve termal açıdan yakın nesnelerin daha az olduğu görseller üzerindeki test sonuçları da verilmiştir.



Şekil 5.1 Standart Kamere ile YOLOv4 Nesne Tespiti Görüntüleri Üzerinde Performans



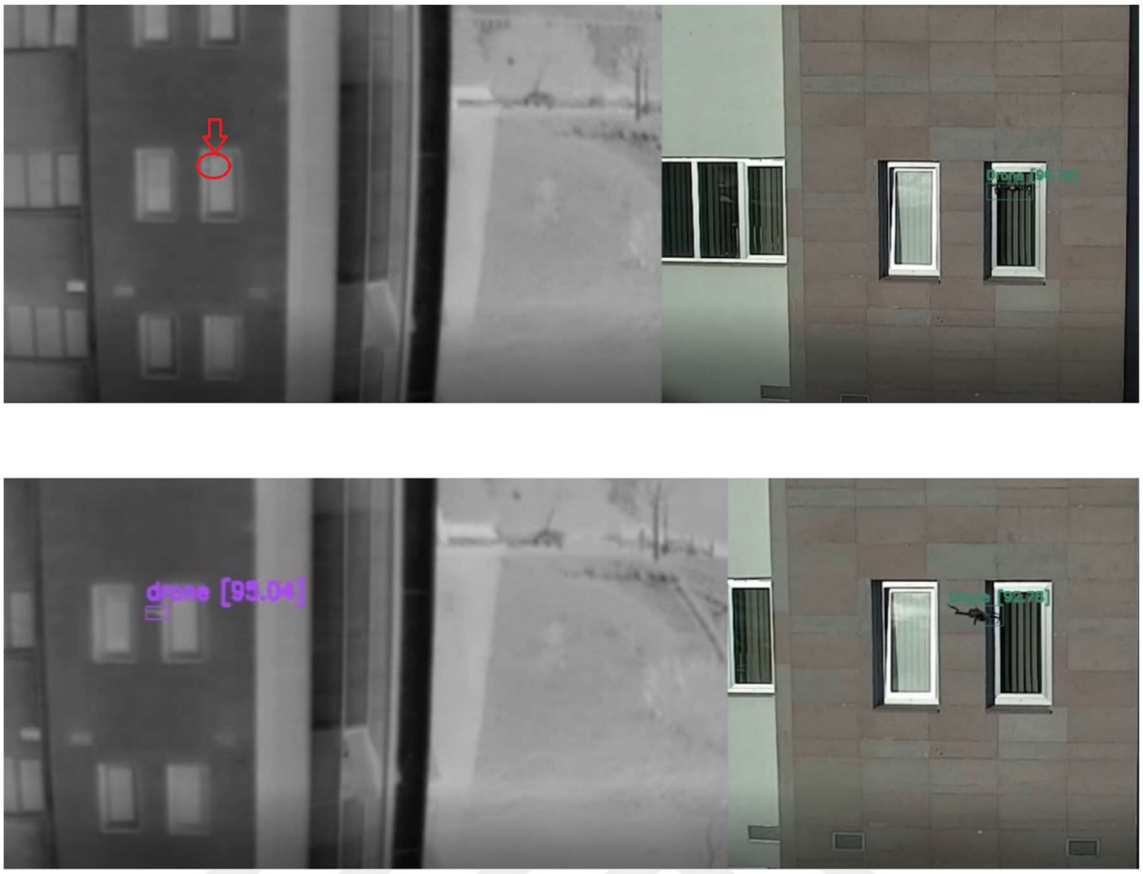
Şekil 5.2 Termal Kamere ile YOLOv4 Nesne Tespiti Görüntüleri Üzerinde Performans

Modele sağlanan veri setinden ayrılan test görselleri üzerinde modelin başarısının belirlenmiştir. Bu süreçte çeşitli görsellerin yer aldığı veri seti sayesinde model dronenin tespitini sağlıklı bir şekilde sağlamıştır. Ayrıca çalışma konusu olarak ele alınan eksikliklerde bu test sürecinde incelenmiştir. Standart kamera görüntüsündeki arka plan renginin drone ile yakınlık göstermesi veya karmaşık olması durumunda termal kamera ile desteklenerek tespitin başarılı biçimde sağlandığı gösterilmiştir. Aynı şekilde arka planın ısıl yoğunluğunun drone nesnesinin ısıl yoğunluğuna yakın olduğu durumlarda da standart kamera ile model desteklenerek dronenin tespitinin başarılı bir şekilde gösterilmiştir. Bu eksikliklerin giderildiği durumlar termal ve standart kamera kaynaklarından eş zamanlı alınan görseller üzerinde incelenerek gösterilmiştir. Bunun için standart kamera girdisinde drone tespitinin zorlaştığı senaryolarda, termal görüntü ile drone tespiti olarak sağlanmış ve modelin termal görüntüler üzerinde başarılı bir şekilde tespit gerçekleştirdiği Şekil 5.3'te gösterilmiştir. Şekil 5.3'te arka planın ağaç ile karmaşıklaştırıldığı durumdaki çıktıları eş zamanlı olarak paylaşılmıştır. Bu şekilde

ağacın renginin drone rengine yakın olması ve karmaşası sebebiyle normal görüntüde tespit sağlanamazken, termal kamera görüntüsü üzerinde başarılı tespitler yapıldığı gözlemlenmiştir. Öte yandan arka plan ısıl yoğunluğunun yakın olduğu durumlarda termal kamerada üzerinden tespitte oluşan zorlukların standart kamera görüntüsü üzerinden tespitin başarılı olarak sağlandığı Şekil 5.4'te gösterilmiştir. Şekil 5.4'te genel itibariyle termal yoğunluğun yakın olduğu durumda termal çıkının başarısının azaldığını ve bu durumun standart kamera çıktısı ile tamamlanarak başarılı bir şekilde tespitinin sağlandığı gözlemlenmiştir.



Şekil 5.3 Karmaşık Arka Planda Standart Kamera ve Termal Kamera Verilerinde Tespit Performansının Karşılaştırılması



Şekil 5.4 Termal Arka Planda Standart Kamera ve Termal Kamera Verilerinde Tespit Performansının Karşılaştırılması

6. TARTIŞMA VE SONUÇ

Yapılan bu çalışma ile literatürde bulunan drone tespit çalışmalarında yapılan tespit işlemi sürecinde yer alan bazı eksiklerin iyileştirilmesine katkıda bulunulmuştur. Bu katkılar, standart veya termal kamera temelli tespit işlemlerinde tespit performansının düşük olduğu veya tespit sağlayamadığı durumları belirleyerek eş zamanlı iki ağın kullanımı ile güçlendirilmesidir. Bu yazıda aktarılan yöntem ile drone tespit doğruluğu yükseltilebilmekte, tek bir görüntü kaynağının başarısız olduğu durumlarda başarılı tespit sağlanabilmektedir. Gelecek çalışmalarda ise küçük nesnelerin tespiti konusunda daha düşük güç gereksinimi duyan ve daha yüksek performans veren modeller kullanılarak çalışma geliştirilebilir. Derin öğrenme sürecinde aktif olarak başarıyı etkileyen veri seti faktörü de ele alındığında daha başarılı ve çeşitlendirilmiş veri seti ile modeller desteklenerek çalışma geliştirilebilir. Aynı zamanda bu yaklaşımın gerçek zamanlı iki farklı görüntü kaynağından alınan veriler üzerinde daha yüksek performansta çalışabileceği de öngörülmektedir.

7. KAYNAKLAR

[1] M. R. Brust, G. Danoy, P. Bouvry, D. Gashi, H. Pathak, and M. P. Goncalves, "Defending Against Intrusion of Malicious UAVs with Networked UAV Defense Swarms," in *Proceedings - 2017 IEEE 42nd Conference on Local Computer Networks Workshops, LCN Workshops 2017*, Institute of Electrical and Electronics Engineers Inc., Nov. 2017, pp. 103–111. doi: 10.1109/LCN.Workshops.2017.71.

[2] B. Taha and A. Shoufan, "Machine Learning-Based Drone Detection and Classification: State-of-the-Art in Research," *IEEE Access*, vol. 7, pp. 138669–138682, 2019, doi: 10.1109/ACCESS.2019.2942944.

[3] S. Salman, J. Mir, M. T. Farooq, A. N. Malik, and R. Haleemdeen, "Machine Learning Inspired Efficient Audio Drone Detection using Acoustic Features," in *Proceedings of 18th International Bhurban Conference on Applied Sciences and Technologies, IBCAST 2021*, Institute of Electrical and Electronics Engineers Inc., Jan. 2021, pp. 335–339. doi: 10.1109/IBCAST51254.2021.9393232.

[4] S. Al-Emadi, A. Al-Ali, A. Mohammad, and A. Al-Ali, "Audio Based Drone Detection and Identification using Deep Learning," in *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, IEEE, Jun. 2019, pp. 459–464. doi: 10.1109/IWCMC.2019.8766732.

[5] M. S. Allahham, T. Khattab, and A. Mohamed, "Deep Learning for RF-Based Drone Detection and Identification: A Multi-Channel 1-D Convolutional Neural Networks Approach," in *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*, IEEE, Feb. 2020, pp. 112–117. doi: 10.1109/ICIoT48696.2020.9089657.

[6] M. U. Sheikh, F. Ghavimi, K. Ruttik, and R. Jantti, "Drone Detection and Classification Using Cellular Network: A Machine Learning Approach," in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, IEEE, Sep. 2019, pp. 1–6. doi: 10.1109/VTCFall.2019.8891229.

[7] F. Gökçe, G. Üçoluk, E. Şahin, and S. Kalkan, "Vision-Based Detection and Distance Estimation of Micro Unmanned Aerial Vehicles," *Sensors*, vol. 15, no. 9, pp. 23805–23846, Sep. 2015, doi: 10.3390/s150923805.

[8] E. Unlu, E. Zenou, N. Riviere, and P. E. Dupouy, "Deep learning-based strategies for the detection and tracking of drones using several cameras," *IPSJ Transactions on Computer Vision and Applications*, vol. 11, no. 1, Dec. 2019, doi: 10.1186/s41074-019-0059-x.

[9] C. Aker and S. Kalkan, "Using Deep Networks for Drone Detection," Jun. 2017, [Online]. Available: http://arxiv.org/abs/1706.05726

[10] K. Abbasi *et al.*, "A Vision-Based Amateur Drone Detection Algorithm for Public Safety Applications," in *2019 UK/China Emerging Technologies, UCET 2019*, Institute of Electrical and Electronics Engineers Inc., Aug. 2019. doi: 10.1109/UCET.2019.8881879.

[11] A. J. Garcia, J. Min Lee, and D. S. Kim, "Anti-Drone System: A Visual-based Drone Detection using Neural Networks," in *2020 International Conference on Information and Communication Technology Convergence (ICTC)*, IEEE, Oct. 2020, pp. 559–561. doi: 10.1109/ICTC49870.2020.9289397.

[12] M. Nalamati, A. Kapoor, M. Saqib, N. Sharma, and M. Blumenstein, "Drone Detection in Long-range Surveillance Videos," 2019.

[13] H. Chen, Z. Wang, and L. Zhang, “Collaborative spectrum sensing for illegal drone detection: A deep learning-based image classification perspective,” *China Communications*, vol. 17, no. 2, pp. 81–92, Feb. 2020, doi: 10.23919/JCC.2020.02.007.

[14] F. Mahdavi and R. Rajabi, “Drone Detection Using Convolutional Neural Networks,” in *2020 6th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*, IEEE, Dec. 2020, pp. 1–5. doi: 10.1109/ICSPIS51611.2020.9349620.

[15] M. S. Kabir, I. K. Ndukwe, and Engr. Z. S. Awan, “Deep Learning Inspired Vision based Frameworks for Drone Detection,” in *2021 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, IEEE, Jun. 2021, pp. 1–5. doi: 10.1109/ICECCE52056.2021.9514124.

[16] A. Koksal, K. G. Ince, and A. A. Alatan, “Effect of Annotation Errors on Drone Detection with YOLOv3,” 2020.

[17] A. Coluccia *et al.*, “Drone-vs-Bird Detection Challenge at IEEE AVSS2021,” in *2021 17th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, IEEE, Nov. 2021, pp. 1–8. doi: 10.1109/AVSS52988.2021.9663844.

[18] K.-P. Phung, T.-H. Lu, T.-T. Nguyen, N.-L. Le, H.-H. Nguyen, and V.-P. Hoang, “Multi-model Deep Learning Drone Detection and Tracking in Complex Background Conditions,” in *2021 International Conference on Advanced Technologies for Communications (ATC)*, IEEE, Oct. 2021, pp. 189–194. doi: 10.1109/ATC52653.2021.9598317.

[19] R. Hamatapa and C. Vongchumyen, “Image Processing for Drones Detection,” in *2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST)*, IEEE, Jul. 2019, pp. 1–4. doi: 10.1109/ICEAST.2019.8802578.

[20] X. Zhang and K. Kusrini, “Autonomous long-range drone detection system for critical infrastructure safety,” *Multimed Tools Appl*, vol. 80, no. 15, pp. 23723–23743, Jun. 2021, doi: 10.1007/s11042-020-10231-x.

[21] F. Svanstrom, C. Englund, and F. Alonso-Fernandez, “Real-Time Drone Detection and Tracking With Visible, Thermal and Acoustic Sensors,” in *2020 25th International Conference on Pattern Recognition (ICPR)*, IEEE, Jan. 2021, pp. 7265–7272. doi: 10.1109/ICPR48806.2021.9413241.

[22] H. B. Barlow, “Unsupervised Learning,” *Neural Comput*, vol. 1, no. 3, pp. 295–311, Sep. 1989, doi: 10.1162/neco.1989.1.3.295.

[23] K.-Y. Tang, C.-Y. Chang, and G.-J. Hwang, “Trends in artificial intelligence-supported e-learning: a systematic review and co-citation network analysis (1998–2019),” *Interactive Learning Environments*, vol. 31, no. 4, pp. 2134–2152, May 2023, doi: 10.1080/10494820.2021.1875001.

[24] Hady, Mohamed Farouk Abdel, and Friedhelm Schwenker. "Semi-supervised learning." Handbook on Neural Information Processing (2013): 215-239.

[25] L. P. Kaelbling, M. L. Littman, and A. W. Moore, “Reinforcement Learning: A Survey,” *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, May 1996, doi: 10.1613/jair.301.

[26] K. Kumar and G. S. M. Thakur, “Advanced Applications of Neural Networks and Artificial Intelligence: A Review,” *International Journal of Information Technology and Computer Science*, vol. 4, no. 6, pp. 57–68, Jun. 2012, doi: 10.5815/ijitcs.2012.06.08.

[27] D. Svozil, V. Kvasnicka, and J. Pospichal, “Introduction to multi-layer feed-forward neural networks,” *Chemometrics and Intelligent Laboratory Systems*, vol. 39, no. 1, pp. 43–62, Nov. 1997, doi: 10.1016/S0169-7439(97)00061-0.

[28] A. Graves, A. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, IEEE, May 2013, pp. 6645–6649. doi: 10.1109/ICASSP.2013.6638947.

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: 10.1145/3065386.

[30] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: 10.1038/nature14539.

[31] S. Ren, K. He, R. Girshick, X. Zhang, and J. Sun, “Object Detection Networks on Convolutional Feature Maps,” *IEEE Trans Pattern Anal Mach Intell*, vol. 39, no. 7, pp. 1476–1481, Jul. 2017, doi: 10.1109/TPAMI.2016.2601099.

[32] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.” [Online]. Available: https://github.com/

[33] Liu, Wei, et al. "Ssd: Single shot multibox detector." Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer International Publishing, 2016.

[34] W. Liu *et al.*, “SSD: Single Shot MultiBox Detector,” 2016, pp. 21–37. doi: 10.1007/978-3-319-46448-0_2.

[35] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” Apr. 2018, [Online]. Available: http://arxiv.org/abs/1804.02767

[36] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection,” Apr. 2020, [Online]. Available: http://arxiv.org/abs/2004.10934

[37] D. Misra, “Mish: A Self Regularized Non-Monotonic Activation Function,” Aug. 2019, [Online]. Available: http://arxiv.org/abs/1908.08681

[38] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” 2016. [Online]. Available: http://pjreddie.com/yolo/

[39] A. Benjumea, I. Teeti, F. Cuzzolin, and A. Bradley, “YOLO-Z: Improving small object detection in YOLOv5 for autonomous vehicles,” Dec. 2021, [Online]. Available: http://arxiv.org/abs/2112.11798

[40] D. P. Papadopoulos, J. R. R. Uijlings, F. Keller, and V. Ferrari, “Extreme clicking for efficient object annotation,” Aug. 2017, [Online]. Available: http://arxiv.org/abs/1708.02750

[41] A. Kuznetsova *et al.*, “The Open Images Dataset V4,” *Int J Comput Vis*, vol. 128, no. 7, pp. 1956–1981, Jul. 2020, doi: 10.1007/s11263-020-01316-z.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Oğuzhan YANIK

Yabancı Dili : İngilizce

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Y. Lisans	Elektrik Elektronik ve Bigisayar Müh.	Düzce Üniversitesi	2023
Lisans	Elektrik Elektronik Müh.	Düzce Üniversitesi	2020

YAYINLAR

- YANIK, O., & ALTUN, Y. STANDART VE TERMAL KAMERA GÖRÜNTÜLERİNDEN DERİN ÖĞRENME KULLANILARAK DRONE TESPİTİ.