

**T.C.
İNÖNÜ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**EVRIŞİMLİ SİNİR AĞLARI KULLANARAK VIDEO VERİLERİ ÜZERİNDE
DRONE TESPİTİ**

YÜKSEK LİSANS TEZİ

Büşra ÖZCAN

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Dr. Öğr. Üyesi Abdullah Erhan AKKAYA

TEMMUZ 2025

**T.C
İNÖNÜ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**EVRIŞİMLİ SİNİR AĞLARI KULLANARAK VIDEO VERİLERİ ÜZERİNDE
DRONE TESPİTİ**

YÜKSEK LİSANS TEZİ

**Büşra ÖZCAN
36233619009**

Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı: Dr. Öğr. Üyesi Abdullah Erhan AKKAYA

TEMMUZ 2025

TEŞEKKÜR VE ÖNSÖZ

Tez çalışmamın her aşamasında bilgi, tecrübe ve desteğini esirgemeyen, önerileriyle beni yönlendiren ve çalışmalarımı titizlikle takip eden değerli danışmanım Dr. Öğr. Üyesi Abdullah Erhan AKKAYA'ya içtenlikle teşekkür ederim.

Hayatımın her alanında olduğu gibi, akademik çalışmalarım boyunca da maddi ve manevi desteklerini her zaman hissettiğim aileme ve arkadaşlarıma sonsuz teşekkürlerimi sunarım.

Tezin uygulama aşamasında değerli veri setini benimle paylaşarak bu çalışmanın gerçekleşmesine katkıda bulunan Wosdetc grubuna ayrıca teşekkür ederim.



ONUR SÖZÜ

Yüksek Lisans tezi olarak sunduğum “Evrişimli Sinir Ağları Kullanarak Video Verileri Üzerinde Drone Tespiti ve Takibi” başlıklı bu çalışmanın bilimsel ahlaka ve araştırma etiğine uygun olarak, dışarıdan yardım almadan tarafımdan yazıldığını ve kullandığım kaynakların tümünün metin içinde ve kaynakçada doğru şekilde verildiğini beyan ederim.

Büşra ÖZCAN



İÇİNDEKİLER

TEŞEKKÜR VE ÖNSÖZ	i
ONUR SÖZÜ.....	ii
İÇİNDEKİLER.....	iii
ÇİZELGELER DİZİNİ.....	iv
ŞEKİLLER DİZİNİ.....	v
SEMBOLLER VE KISALTMALAR	vi
ÖZET	vii
ABSTRACT	viii
1. GİRİŞ.....	1
1.1 Amaç	3
1.2 Literatür Araştırması.....	3
1.3 Literatürün Genel Görünümü.....	14
1.4 Görüntü İşleme Adımları	15
2. DERİN ÖĞRENME	17
2.1 Evrişim Katmanı	18
2.2 Adım Ve Dolgu.....	21
2.3 Aktivasyon Fonksiyonu	23
2.4 Havuzlama Katmanı	23
2.5 Yolov11 Mimarisi.....	24
2.5.1 C3k2 bloğu	26
2.5.2 SPFF bloğu	27
3. YÖNTEM VE METOT.....	29
3.1 Veri Seti	29
3.2 Ön Deneyler.....	32
3.3 Yöntem.....	32
3.4 Eğitim Parametreleri	32
3.5 Değerlendirme Metrikleri	33
3.5.1 Karmaşıklık matrisi (Confusion Matrix)	34
3.5.2 Kesinlik (Precision)	35
3.5.3 Duyarlılık (Recall).....	35
3.5.4 F1-Skoru	36
3.5.5 IoU (Kesişim birleşme oranı)	36
3.5.6 Ortalama doğruluk (mAP@0.5 & mAP@0.5:0.9)	37
4. EĞİTİM SONUÇLARI VE DEĞERLENDİRME	39
4.1 Ön Deney Sonuçları.....	39
4.2 Ana Deney Sonuçları	40
5. SONUÇ VE ÖNERİLER	48
KAYNAKLAR.....	55
ÖZGEÇMİŞ	59

ÇİZELGELER DİZİNİ

Çizelge 4.1: Ön deneylere ait eğitim parametreleri ve sonuçlar	39
Çizelge 4.2: Eğitilen modeller ve elde edilen başarı ölçütleri	40



ŞEKİLLER DİZİNİ

Şekil 1.1	Girdi Görüntüsü	15
Şekil 1.2	Tahmin sonucu elde edilen sınırlayıcı kutu örneği	16
Şekil 2.1	Özellik haritası oluşturmak için çekirdek ve girdi verisi işlemi ilk adım	19
Şekil 2.2	Özellik haritası oluşturmak için çekirdek ve girdi verisi işlemi ikinci adım	19
Şekil 2.3	Özellik haritası oluşturmak için çekirdek ve girdi verisi işlemi üçüncü adım	20
Şekil 2.4	Özellik haritası oluşturmak için çekirdek ve girdi verisi işlemi dördüncü adım	20
Şekil 2.5	Adım sayısı 3 olduğu takdirde elde edilecek özellik haritası boyutu	21
Şekil 2.6	Adım sayısı kaydıkaçı dahil edilen pikseller	22
Şekil 2.7	Görüntüye dolgu eklendiği zaman çekirdek matrisi ve girdi görüntüsü	22
Şekil 2.8	Aktivasyon fonksiyonu temsili kullanım	23
Şekil 2.9	Özellik haritasına Max Pooling işlemi uygulama	24
Şekil 2.10	YOLOv11 Mimari Yapısı	25
Şekil 2.11	C3K2 Blok yapısı “(He, Zhou, Liu, Zhang, et al., 2025)	27
Şekil 3.1	Veri setinde bulunan drone görüntüsü örneği	30
Şekil 3.2	Veri setinde bulunan uçan bir kuş görüntüsü örneği	30
Şekil 3.3	Veri setinde bulunan aynı karede uçan drone ve kuşların olduğu görüntü örneği	31
Şekil 3.4	Veri setinde bulunan arka plan karmaşıklığından ötürü gözlemlenemeyen drone görüntüsü	31
Şekil 3.5	Karmaşıklık Matrisi	34
Şekil 3.6	IoU hesaplama	37
Şekil 4.1	En başarılı modele ait karışıklık matrisi grafiği	42
Şekil 4.2	En başarılı modele ait F1 Curve grafiği	43
Şekil 4.3	En başarılı modele ait eğitim esnasında gözlemlenen kayıp fonksiyonları ve performans metrikleri	44
Şekil 4.4	En başarılı modele ait P-Curve grafiği	45
Şekil 4.5	En başarılı modele ait R-Curve grafiği	46
Şekil 4.6	En başarılı modele ait PR-Curve grafiği	47
Şekil 5.1	En iyi model ağırlıkları ile tespit edilen drone görüntüsü 1	49
Şekil 5.2	En iyi model ağırlıkları ile tespit edilen drone görüntüsü 2	49
Şekil 5.3	En iyi model ağırlıkları ile tespit edilen drone görüntüsü 3	50
Şekil 5.4	En iyi model ağırlıkları ile tespit edilen kuş görüntüsü 1	50
Şekil 5.5	En iyi model ağırlıkları ile tespit edilen kuş görüntüsü 2	51
Şekil 5.6	En iyi model ağırlıkları ile tespit edilen kuş görüntüsü 3	51
Şekil 5.7	YOLOv11 modelinin 640 ve 320 piksel çözünürlüklerde FREEZE parametresine göre Precision, Recall ve F1 skorlarının değişimi	52
Şekil 5.8	YOLOv8 modelinin 640 piksel çözünürlükte FREEZE parametresine göre Precision, Recall ve F1 skorlarının değişimi	52

SEMBOLLER VE KISALTMALAR

YOLO	: You Only Look Once
YOLOv11	: YOLO mimarisinin 11. versiyonu
YOLOv8	: YOLO mimarisinin 8. versiyonu
CNN	: Convolutional Neural Network
IoU	: Intersection over Union
F1	: Harmonik Ortalama
Precision	: Kesinlik
Accuracy	: Doğruluk
Recall	: Duyarlılık
mAP	: Mean Avarage Precision
FPS	: Frame Per Second
Batch Size	: Aynı anda eğitime alınan örnek sayısı
Epoch	: Dönem
Freeze	: Modelde bazı katmanların eğitime kapatılması
Adam	: Adaptive Moment Estimation
SGD	: Stochastic Gradient Descent
SPPF	: Spatial Pyramid Pooling - Fast
C2PSA	: Com Cross Stage Partial with Spatial Attention
TP	: True Positive
FP	: False Positive
TN	: True Negative
FN	: False Negative

ÖZET

Yüksek Lisans Tezi

Evrişimli Sinir Ağları Kullanarak Video Verileri Üzerinde Drone Tespiti

Büşra ÖZCAN

İnönü Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

59 sayfa

2025

Danışman: Dr. Öğr. Üyesi Abdullah Erhan AKKAYA

Bu tez çalışmasında, görüntü işleme ve derin öğrenme tabanlı yaklaşımlar kullanılarak insansız hava araçlarının (drone) gerçek zamanlı tespiti amaçlanmıştır. Çalışmada, YOLO mimarisinin en güncel versiyonu olan YOLOv11 modeli temel alınmış ve farklı hiperparametre kombinasyonları altında drone-kuş ayrımı üzerine eğitimler gerçekleştirilmiştir. Eğitimlerde kullanılan veri seti, hem drone hem de kuş görüntülerinden oluşmakta olup, video formatından elde edilen karelerle manuel olarak etiketlenmiştir. Eğitim süreci boyunca freeze parametresi ve giriş boyutu gibi değişkenler sistematik olarak test edilmiştir. Deneysel sonuçlara göre, YOLOv11 modeli yüksek doğruluk ve duyarlılık değerleri ile karmaşık sahnelerde etkili nesne tespiti gerçekleştirebilmiştir. Elde edilen bulgular, özellikle güvenlik, sınır kontrolü ve hava sahası gözetimi gibi alanlarda kullanılabilecek yapay zekâ destekli gözetim sistemlerinin geliştirilmesine katkı sağlamaktadır. Ayrıca, bu çalışmada YOLOv8 modeli ile karşılaştırmalı analizler sunularak, YOLOv11'in üstünlükleri deneysel olarak ortaya konmuştur.

Anahtar Kelimeler: YOLOv11, drone tespiti, derin öğrenme, görüntü işleme, nesne tanıma

ABSTRACT

Phd. Thesis

DRONE DETECTION ON VIDEO DATA USING CONVOLUTIONAL NEURAL NETWORKS

Büşra ÖZCAN

Inonu University
Graduate School of Nature and Applied Sciences
Department of Computer Engineering

59 page

2025

Supervisor: Assistant Proffesor Abdullah Erhan AKKAYA

In this thesis study, it was aimed to detect unmanned aerial vehicles (drones) in real time by using image processing and deep learning-based approaches. The YOLOv11 model, which is the most recent version of the YOLO architecture, was used as the main model, and drone-bird classification was performed under different hyperparameter settings. The dataset used in training consists of both drone and bird images, and the images were manually labeled by extracting frames from video footage. During the training process, variables such as the freeze parameter and input size were tested systematically. According to the experimental results, the YOLOv11 model achieved effective object detection even in complex scenes with high precision and recall values. The findings of this study can contribute to the development of AI-assisted surveillance systems, especially in areas such as security, border control, and airspace monitoring. In addition, the performance of YOLOv11 was compared with the YOLOv8 model, and the experimental results revealed the advantages of YOLOv11 over the older version.

Keywords: YOLOv11, drone detection, deep learning, image processing, object detection

1. GİRİŞ

Günümüzde hızla gelişen teknoloji, bireylere ve kurumlara çeşitli faydalar sağlarken, aynı zamanda güvenlik açısından yeni tehditleri de beraberinde getirmektedir. Özellikle küçük, taşınabilir ve düşük maliyetli insansız hava araçları (drone'lar), kötü niyetli bireyler tarafından yasa dışı faaliyetlerde kullanılabilir. Bu durum, hem sivil alanlarda özel hayatın ihlali hem de askeri bölgelerde güvenlik zafiyeti yaratmakta ve dolayısıyla etkili bir drone tespit sistemine olan ihtiyacı gündeme getirmektedir. Kullanım kolaylığı, taşınabilirliği ve üretim maliyeti nedeniyle çok faydalı olabilecek bu ufak insansız hava araçları yanlış ellerde büyük suç aletleri olarak kullanılmaktadır.

Drone teknolojileri başlangıçta lojistik, tarım, haritalama ve medya gibi alanlarda çeşitli faydalar sağlamak amacıyla geliştirilmiş ve yaygınlaştırılmıştır. Özellikle kargo taşımacılığı ve hızlı teslimat gibi uygulamalarla kamuoyunda ve sivil halk arasında olumlu bir algı oluşturmuştur. Ancak düşük üretim maliyeti, kompakt yapısı ve kolay erişilebilirliği, bu teknolojilerin kötü niyetli kullanımını da mümkün kılmıştır. Drone'ların yasa dışı veri toplama, sınır ihlali, kaçakçılık ve saldırı gibi amaçlarla kullanıldığı çok sayıda vaka literatürde yer almaktadır. Bu durum, kişisel mahremiyetin ihlali ve kamu güvenliği açısından ciddi tehditler oluşturmuştur. Örneğin İrlanda bu konuda yasal düzenleme getiren öncü ülkelerden birisi olmuştur (Tsiamis et al., 2019). Ancak, artan nüfus, teknolojinin hızlı gelişimi ve ülkeler arası politik gerilimler, yasal önlemlerin yetersiz kalmasına neden olmuştur. Özellikle 2025 yılı itibarıyla devam eden Rusya-Ukrayna ve İsrail-Filistin çatışmalarında, drone'ların düşük maliyetli ve insansız yapıları sebebiyle sıkça kamikaze amaçlı kullanıldığı gözlemlenmektedir. Tüm bu gelişmeler, güvenlik tehditlerine karşı yalnızca fiziksel ya da yasal önlemlerin değil; aynı zamanda gelişmiş, gerçek zamanlı çalışan dijital izleme ve tespit sistemlerinin de kaçınılmaz bir gereklilik haline geldiğini göstermektedir.

Yapay zekâ alanında yaşanan ilerlemeler, insan müdahalesine olan ihtiyacı her geçen gün azaltmakta ve buna bağlı olarak hata oranlarını düşürmeyi amaçlayan çeşitli uygulamaların geliştirilmesine olanak tanımaktadır. Bu gelişmeler doğrultusunda, insansız hava araçlarının (drone'ların) tespiti ve analizi gibi güvenlik temelli konular da

araştırmacıların yoğun ilgi gösterdiği alanlardan biri haline gelmiştir. Drone kullanımının giderek yaygınlaşması, beraberinde yasa dışı faaliyetlere zemin hazırlayabilecek potansiyel riskleri de artırmaktadır.

Bu tez çalışmasında, güvenlik açısından risk taşıyan bölgelerde izinsiz drone varlığının tespiti için, görüntü işleme temelli ve gerçek zamanlı çalışan bir sistem ihtiyacından hareketle çözüm önerisi sunulmaktadır. Bu kapsamda, derin öğrenme mimarilerinin nesne tespiti konusundaki başarısı dikkate alınarak, YOLO (You Only Look Once) temelli yaklaşımlar tercih edilmiştir. YOLOv11 mimarisi, önceki sürümlere kıyasla sunduğu yüksek doğruluk oranı ve işlem hızı sayesinde bu çalışmada uygulanacak modelin temelini oluşturmaktadır.

Uygulama kapsamında yalnızca RGB formatında, sabit kameralar aracılığıyla açık hava koşullarında elde edilmiş video görüntüleri kullanılacaktır. Çalışma süresince termal kamera verisi veya radar görüntüsü gibi ek sensör verilerine yer verilmemiştir. Modelin temel hedefi, drone'ların diğer nesnelerden, özellikle kuşlardan ayırt edilmesi ve pozisyon bilgilerinin başarıyla izlenebilmesidir.

Drone tespiti için derin öğrenme tabanlı nesne tanıma algoritmalarının önemi, karmaşık arka planlara ve değişken ışık koşullarına rağmen yüksek doğrulukla sonuç verebilmelerinde yatmaktadır. Özellikle YOLO mimarileri, bölgesel öneri mekanizmalarına gerek duymadan tek bir ileri besleme adımında nesneleri tespit edebilmesi sayesinde gerçek zamanlı uygulamalara son derece uygundur. Bu özellik, kalabalık şehir merkezleri, etkinlik alanları veya askeri tesisler gibi yüksek güvenlik gerektiren bölgelerde izinsiz drone girişlerinin anlık olarak saptanabilmesine olanak tanır. Ayrıca YOLO'nun esnek mimarisi, farklı boyutlardaki droneleri çeşitli mesafelerde algılayabilme kabiliyetiyle geniş kapsamlı kullanım potansiyeline sahiptir.

Son yıllarda YOLO algoritmasının farklı sürümleri arasında yapılan karşılaştırmalar, modelin mimari geliştirmeleriyle birlikte drone tespiti görevlerinde belirgin performans artışları sağlandığını ortaya koymaktadır. YOLOv1'den YOLOv11'e kadar gelen süreçte katman sayılarının artması, aktivasyon fonksiyonlarının iyileştirilmesi ve veri işleme hızının optimize edilmesi, özellikle küçük ve hızlı hareket eden nesnelerin algılanmasını kolaylaştırmıştır. Bu ilerlemeler, yalnızca görüntü işleme değil, aynı zamanda sınır güvenliği, hava sahası yönetimi ve kritik altyapıların korunması gibi farklı güvenlik uygulamalarında da bu teknolojinin etkin bir şekilde kullanılabileceğini göstermektedir.

1.1 Amaç

Bu çalışmanın temel amacı, güncel derin öğrenme mimarilerinden faydalanarak güvenlik güçleri ve kamu kurumlarının ihtiyaç duyduğu dijital gözetim sistemlerine katkı sağlayacak bir model tasarlamaktır. Literatürde, görüntü üzerinden nesne tespiti yapmaya yönelik çok sayıda çalışma mevcut olup, bu alanda kullanılan birçok önceden eğitilmiş mimari geliştirilmiştir.

Bu kapsamda, nesne tespitinde sıklıkla tercih edilen YOLO (You Only Look Once) mimarisinin en güncel sürümü olan YOLOv11 temel alınarak çeşitli modelleme çalışmaları gerçekleştirilmiştir. Modeller üzerinde belirli parametre değişiklikleri yapılmış; bu değişikliklerin eğitim sürecine ve model başarımına olan etkileri analiz edilmiştir. Aynı parametreler ile eğitilen YOLOv8 modelinden alınan çıktılar ile asıl hedef konumuz olan YOLOv11 modelinin sonuçları karşılaştırılmıştır. Elde edilen çıktılarla, mevcut literatüre hem yöntemsel hem de uygulama düzeyinde katkı sağlanması hedeflenmektedir.

1.2 Literatür Araştırması

Yapılan çalışma kapsamında YOLO modelleri kullanarak nesne tespiti yapılmış olan çalışmaları inceleyeceğiz.

Derin öğrenme modelleri kullanarak drone ve kuş tespiti yapılan ‘A Deep Learning Approach to Classify Drones and Birds (Sethu Selvi et al., 2022) başlıklı çalışmada güvenlik ve gözetleme alanında etkili bir çözüm geliştirmek hedeflenmiştir. Bunun için tek aşamalı nesne tespit algoritmaları olan YOLOv4 ve YOLOv5 modelleri ile eğitim yapılmış ve sonuçlar karşılaştırılmıştır. Çalışma kapsamında kullanılan veri seti 664 drone, 236 kuş görüntüsünden oluşmaktadır ve bu veri seti 90:10, 80:20 ve 70:30 oranlarında eğitim ve test olarak bölünmüştür. YOLOv4 modeli %97.4, YOLOv5 ise %95 ortalama doğruluk (mAP) elde etmiştir. YOLOv4 %98 F1 skoru ve 54 FPS algılama hızı, YOLOv5 %94 F1 skoru ve 77 FPS elde etmiştir. Yapılan deneylerde YOLOv4 modeli doğruluk olarak daha yüksek bir başarı verirken YOLOv5 ise algılama hızı açısından daha yüksek bir performans sergilemiştir. Her iki modelde drone ve kuş ayrımı yapmakta başarılı sonuçlar elde etmiştir.

Xu (Xu, 2023), geleneksel yöntemlerin dışında renk-kenar bilgisi ve Evrişimli Sinir Ağları (CNN) tabanlı bir yöntem önermiştir. Bu yaklaşım, düşük hesaplama gücüne sahip cihazlarda çalışabilecek şekilde optimize edilmiştir. Geliştirilen model, morfolojik işlemler

kullanarak aday hedefleri belirleyerek orijinal görüntüleri yamaları ile evrişimli katmanlara giriş olarak sunmaktır. Bu sayede tespit performansının artırılması amaçlanmıştır. Önerilen yöntemin performansını değerlendirmek için 350 kareden oluşan bir veri seti kullanılmıştır ve yöntem YOLO modelleri gibi geleneksel yöntemler ile karşılaştırılmıştır. Önerilen yöntem %90.5 doğruluk ve %1 yanlış alarm oranı verirken; CNN modeli %87.43 doğruluk ve %4.5 yanlış alarm oranı, SVM modeli %82.57 doğruluk ve %3 yanlış alarm oranı, ANN modeli ise %79.14 doğruluk ve %2 yanlış alarm oranı vermiştir. Önerilen yöntem diğer modellere göre daha yüksek doğruluk ve daha düşük yanlış alarm oranı ile üstünlük sağlamıştır.

Cai ve takımı (Cai et al., 2022), YOLOv4 tabanlı hafif ve doğru İHA tespit yöntemi geliştirmişlerdir. Çalışmada, YOLOv4 modeli optimize edilmiştir ve orijinal CSPDarknet-53 omurgası MobileNet serisi ağlarla değiştirilmiştir. Parametrelerini azaltmak için derinlemesine ayrılabilir evrişim (depthwise seperable convolution) kullanılarak ağ yapısı düzenlenmiştir. Önerilen yöntemin performansını değerlendirmek için model; SSD, Faster R-CNN ve EfficientDet modelleri ile karşılaştırılmıştır. Deneyde kullanılan veri seti, 4 farklı İHA türü ve 20.365 görüntüyü içermektedir. Veri setinin %90'ı eğitim, %10'u test için ayrılmıştır. Veri seti orijinal görüntüler üzerinde mozaik veri arttırmak yöntemi kullanılarak genişletilmiştir. Önerilen yöntem olan güncellenmiş YOLOv4 %96.02 ortalama doğruluk oranı ve 40 FPS, Faster R-CNN %86.61 ortalama doğruluk ve 14 FPS, SSD %79.25 ortalama doğruluk ile 88 FPS ve EfficientDet %40.5 ortalama doğruluk ile 28 FPS sonuçlarını vermiştir. Önerilen yöntem algılama hızı bakımından SSD modelinin gerisinde kalmış olsa da SSD modeline kıyasla doğruluk açısından daha üstün bir performans sergilemiştir.

'A Modified YOLOV4 Deep Learning Network for Vision-Based UAV Recognition' başlıklı çalışmada Javan ve arkadaşları (Javan et al., 2022), İHA'ları kuşlardan ayırt etmek üzerine çalışmışlardır. 4 farklı İHA tipinin tanınması için YOLOv4 derin öğrenme ağını uyarlamışlardır. Performansı karşılaştırmak için önerilen YOLOv4 modelini aynı parametreler ile eğitilmiş temel YOLOv4 modeli ile karşılaştırmışlardır. Önerilen model için, temel YOLOv4 modelinde omurga katmanının sonuna 3 yeni evrişimsel ağ eklenmiştir. Baş katmanına, iki adet büyük ölçekli evrişimsel ağ eklenmiştir. Deneyler için 26.000 görüntüden oluşan bir veri seti kullanılmıştır ve bu veri setinin %70'i eğitim %30'u test olarak ayrılmıştır. Deneyler sonucunda temel YOLOv4 %80 kesinlik, %79 duyarlılık, %79 F1 skor ve %79 ortalama doğruluk oranı vermiştir. Önerilen yöntem ise %83 kesinlik,

%83 duyarlılık, %83 F1 skor ve %83 ortalama doğruluk değerlerini vermiştir. Temel YOLOv4 modellerinin üzerine eklenen evrimsel ağların modelin başarısını arttırdığı gözlemlenmiştir.

“An Object Detection Algorithm for Rotary-Wing UAV Based on Awin Transformer” (Fan et al., 2022) başlıklı çalışmada İHA’ların görüntüler üzerinden tespitine yönelik Awin Transformer tabanlı yeni bir nesne tespit algoritması geliştirilmiştir. Geliştirilen yöntemde, halka pencere self-attention mekanizması kullanılmıştır. YOLOv5 mimarisinin omurgasına Channel Attention (CA) eklenmiştir. Omurga kısmındaki PANet yapısı, BiFPN yapısı ile değiştirilmiştir. Nesne kenarlarını farklı yönlerde belirginleştirmek amacıyla eşitli yönlerde kenar bilgisi çıkarımı yapabilen gelişmiş Gabor filtreleri kullanılmıştır. Çalışmada eğitim ve test süreçleri için COCO 2017 veri seti ve yazarlar tarafından oluşturulan, 10.000 döner kanatlı İHA görüntüsü içeren Rotor-Drone veri seti kullanılmış; eğitim süreci RTX 3090 GPU üzerinde, 36 dönem boyunca AdamW optimizasyon yöntemiyle gerçekleştirilmiştir. DeneySEL sonuçlara göre, Awin algoritması COCO veri setinde Faster R-CNN’e göre %15.8, DETR’ye göre %10.3, Deformable DETR’ye göre %6.5 daha yüksek mAP elde ederken, Rotor-Drone veri setinde Faster R-CNN’e göre %11.1 ve RetinaNet’e göre %9.0 oranında iyileşme sağlamıştır.

Yasmine ve arkadaşları (Yasmine et al., 2023), gerçek zamanlı çoklu hava hedefi tespiti ve tanımlaması yapabilen gelişmiş bir YOLOv7 modeli önermeyi amaçlamışlardır. Geliştirilen modelin performans ölçütleri YOLOv5, YOLOv6, YOLOv7, YOLO-tiny ve YOLOX ile karşılaştırılmıştır. Modelde boyun kısmına CSPResNeXt modülü eklenmiştir. C3TR dikkat mekanizması ve ayrıştırılmış kafa yapısı kullanılmıştır. Drone, kuş, uçak, gün ışığı objeleri ve binalardan oluşan 5 sınıflı bir veri seti kullanılmıştır. Mosaic, MixUp ve Cutmix gibi veri arttırma teknikleri uygulanmıştır. Evrimsel algoritma yöntemiyle hiperparametre optimizasyonu gerçekleştirilmiştir. Eğitim esnasında transfer öğrenme, sıfırdan eğitim ve ensemble öğrenme biçimleri kombine edilmiştir. Test sırasında TTA (Test Time Augmentation) uygulanmıştır. 1179 görüntü üzerinde test gerçekleştirilmiştir. Karşılaştırılan modellerden en yüksek oranı %97.9 ortalama doğruluk, %96 kesinlik, %94.6 duyarlılık ile YOLOv7x vermiştir. Geliştirilen model ise %97.9 ortalama doğruluk, %97 kesinlik ve %96.1 duyarlılık ile karşılaştırmalı modellerin tamamından daha üstün performans göstermiştir. Model gerçek zamanlı olarak 92 FPS hızında nesne tespiti gerçekleştirebilmektedir.

İHA'ların doğru ve güvenilir bir şekilde tespit edilmesi için YOLOv5 ve YOLOv7 algoritmalarını karşılaştırmak üzere Dewangan ve ekibi (V. Dewangan et al., 2023) güncel ve alternatif teknolojileri ele almışlardır. Veri seti üzerinde eğitim yapmadan önce RGB, gri tonlama, renk tonu artırma, kenar geliştirme gibi bir dizi ön işlem gerçekleştirilmiştir. 1847 görüntüden oluşan bir veri seti kullanılmıştır. Verilerin %80'i eğitim, %20'si test olarak ayrılmıştır. Eğitim parametreleri; 150 dönem, 16 batch size, 0.01 öğrenme oranı ve SGD optimizasyon algoritmasıdır. Hue (renk tonu) artırma tekniği ile eğitilen YOLOv5 modeli %95 kesinlik, %95.6 duyarlılık ve %96.7 ortalama doğruluk ile en yüksek başarıyı elde eden model olarak öne çıkmıştır. YOLOv7 modeli ise RGB renk skalası ile eğitilen veriler ile; %91.5 kesinlik, %91.1 duyarlılık ve %93.6 ortalama hassasiyet ile en yüksek değerlere ulaşmıştır. Ancak iki model kıyasladığında YOLOv5 modeli daha başarılı sonuçlar elde etmiştir.

'Automated Drone Detection Using YOLOv4' başlıklı çalışma kapsamında Singha ve ekibi (Singha & Aydın, 2021) İHA ve benzeri nesnelerin tespiti için YOLOv4 tabanlı bir model eğitmişlerdir. Kullanılan veri seti 1916 drone ve 479 kuş görüntüsünden oluşmaktadır. Veri seti %90 eğitim, %10 test olarak ayrılmıştır. Model 4000 dönem boyunca eğitilmiş ve her 1000 dönemde model ağırlıkları kayıt edilmiştir. Transfer öğrenme tekniği kullanılmıştır. YOLOv4 modeli %74.36 ortalama doğruluk değeri gözlemlenmiştir. %95 kesinlik, %68 duyarlılık ve %79 F1 skoru elde edilmiştir.

Atik (Atik, 2023) yapmış olduğu çalışmada drone tespiti sağlamak için YOLOv5 algoritmasını geliştirmiştir. 2795 drone görüntüsü içeren bir veri seti kullanılmıştır. Veri seti %70 eğitim, %15 doğrulama ve %15 test olarak ayrılmıştır. YOLOv5 mimarisine BottleneckCSP modülü ve çarpaz konvolüsyon yapısı entegre edilmiştir. Model 200 dönem, 640 image size ve 16 batch size parametreleri ile eğitilmiştir. Önerilen CB-YOLOv5 modeli, YOLOv5s ile karşılaştırılmıştır. CB-YOLOv5 %99.3 ortalama doğruluk, kesinlik %98.9, duyarlılık %97.8 sonuçlarını vermiştir. YOLOv5s %94.5 ortalama doğruluk, kesinlik %92.7 ve %90.8 duyarlılık sonuçlarını vermiştir. Geliştirmiş olduğu model karşılaştırılan modele göre daha üstün performans göstermektedir.

Zhou ve arkadaşlarının yapmış olduğu çalışmada (Zhou et al., 2025) YOLOv11 modeli temel alınarak üzerinde güncellemeler yapılmıştır. Geliştirilen modelin adı LAMS-YOLO'dur ve bu modelde YOLOv11 mimarisinin omurga yapısı, LCBackbone ile değiştirilmiştir. Ek olarak aktivasyon fonksiyonu olarak H-Swish seçilmiştir. Boyun katmanına Adown modülü entegre edilmiştir. Boyun katmanındaki dikkat mekanizması

yerine Mamba mimarisinden esinlenerek geliştirilen MILA (Mamba-Inspired Linear Attention) modülü eklenmiştir. Eğitimler sonucunda YOLOv11 %89.9 ortalama doğruluk, %96.3 kesinlik, %84.1 duyarlılık ve %89.79 F1 skor elde etmiştir. Önerilen model olan LAMS-YOLO ise %93.4 ortalama doğruluk, %96.1 kesinlik, %89.3 duyarlılık ve %92.58 F1 skor elde etmiştir. Önerilen model temel YOLOv11 mimarisine kıyasla daha yüksek doğruluk ve F1 skorları sağlamıştır.

‘Research on object detection and recognition in remote sensing images based on YOLOv11’ başlıklı He ve ekibinin yapmış olduğu çalışmada (He, Zhou, Liu, Cao, et al., 2025) YOLOv11 modeli ile eğitimler gerçekleştirilmiş ve aynı eğitim eski YOLO modelleri ile karşılaştırılmıştır; YOLOv5, YOLOv7, YOLOv8, YOLOv9 ve YOLOv10. Toplam 70.389 görüntüden oluşan 20 hedef kategori barındıran bir veri seti kullanılmıştır. Veri setinin %60’ı eğitim, %30’u doğrulama ve %10’u test için ayrılmıştır. Eğitim parametreleri 1000 dönem, 1280x1280 resim boyutu, AdamW optimizasyon algoritması ve 0.0001 öğrenme oranı olarak belirlenmiştir. Bu eğitim sonucunda %89.2 ortalama doğruluk, %86.6 kesinlik, %85.6 duyarlılık ve %87 F1 skoru elde edilmiştir. Diğer modellerle karşılaştırmak üzere aynı model ve diğer modeller COCO veri seti ile eğitilmiş ve sonuçlar karşılaştırılmıştır. COCO mAP değeri için modellerin başarı ortalamaları sırasıyla; YOLOv11 modeli için %54.7, YOLOv10 için %54.5, YOLOv9 %55.6, YOLOv8 %53.9, YOLOv7 %52.9 ve YOLOv5 %53.2 şeklindedir. En yüksek başarı oranı YOLOv9 modeli tarafından elde edilmiştir.

Mahala ve arkadaşları yapmış oldukları çalışma kapsamında (Mahala et al., 2025) otonom araçlar için gerçek zamanlı nesne tespiti gerçekleştirmişlerdir. Nesne tespiti için YOLOv11 ve YOLOv7 modelleri karşılaştırılmıştır. Çalışmada 7464 görüntüden oluşan KITTI isimli veri setinden faydalanılmıştır. Veriler %70 eğitim, %10 doğrulama ve %20 test olarak ayrıştırılmıştır. Eğitim parametreleri 150 dönem, 640x640 resim boyutu, AdamW optimizasyon algoritması ve cosine LR schedule kullanılmıştır. YOLOv7 modelinden eğitim sonucunda %70.3 ortalama doğruluk, %91.7 kesinlik, %86.9 duyarlılık elde edilmiştir. YOLOv11 modelinden ise %74.3 ortalama doğruluk, %91.2 kesinlik, %88.3 duyarlılık sonuçları elde edilmiştir. Çalışmada yazarlar YOLOv11 modelinin özellikle az örnek içeren sınıflarda üstün performans gösterdiği vurgulanmıştır.

Alkhammash ve arkadaşları (Alkhammash, 2025) bu çalışmanın konusundan farklı bir alanda çalışma gerçekleştirmişlerdir. Orman yangınlarını erken tespit edebilmek için karşılaştırmalı olarak YOLOv9, YOLOv10 ve YOLOv11 modellerini eğitmişlerdir. Eğitilen

bu modeller karşılaştırılmıştır. Veri seti 11.667 görüntüden oluşmaktadır ve veriler %70 eğitim, %20 doğrulama, %10 test olmak üzere bölünmüştür. Yapılan deneyler sonucunda YOLOv11 %85.9 ortalama doğruluk, %84.5 kesinlik, %80.1 duyarlılık elde etmiştir. YOLOv10 modeli %83.7 ortalama doğruluk, %83.6 kesinlik, %76.4 duyarlılık sonuçlarına ulaşmıştır. YOLOv9 modeli ise %82.9 ortalama doğruluk, %81.7 kesinlik, %75.9 duyarlılık sonuçlarını vermiştir. YOLOv11 tüm metriklerde en yüksek başarıyı göstererek en başarılı model olarak öne çıkmıştır.

‘Research on Traffic Target Detection Optimization Based on YOLOv11’ başlıklı çalışmada Tang ve arkadaşları (Tang et al., 2025) YOLOv11 modeli kullanarak bu çalışmanın kapsamından farklı bir alanda çalışma gerçekleştirmişlerdir. Bu model YOLOv5, YOLOv6, YOLOv7-tiny, YOLOv8 modelleri ile karşılaştırılmıştır. Eğitim aşamasında verilere Mozaik, HSV değişiklikleri, yatay çevirme gibi artırma teknikleri uygulanmıştır. CCTSDB veri seti kullanılmıştır. Veri setine ait toplam görsel sayısı ve veri ayırma oranları belirtilmemiştir. Eğitim parametreleri olarak; batch size 64, resim boyutu 640x640, 0.01 öğrenme oranı, 300 dönem, 0.937 momentum, SGD optimizasyon algoritması seçilmiştir. YOLOv11 modeli %91.2 ortalama doğruluk, YOLOv5 %88.2 ortalama doğruluk, YOLOv6 %89.1 ortalama doğruluk, YOLOv7-tiny %90.4 ortalama doğruluk, YOLOv8 %90.6 ortalama doğruluk değeri elde etmiştir. Karşılaştırılan modele göre en yüksek doğruluk oranını YOLOv11 modeli vermiştir.

Bai ve arkadaşları (Bai et al., 2025), küçük nesne tespitine yönelik YOLOv11 tabanlı bir model olan HFC-YOLO11’i önermiştir. Model, derinliği azaltılmış özellik piramidi yapısı ile yüksek çözünürlüklü P2 katmanını koruyarak küçük nesneler için ayrıntılı temsil sunarken, P5 katmanını çıkararak parametre yükünü azaltmıştır. GhostBottleneck ve klasik Bottleneck modülleri ile derinlik-farkındalık modüler yapı tasarlanmıştır. Ayrıca, küçük nesnelerde konumlama hatalarını azaltmak üzere EIoU kayıp fonksiyonu önerilmiştir. Modelin etkinliği, AI-TOD (28.036 görüntü) ve VisDrone2019 (8629 görüntü) veri setleri ile değerlendirilmiş; eğitim/val/test oranları sırasıyla %40/%10/%50 ve %75/%6/%19 olarak belirlenmiştir. YOLOv11s taban modeline kıyasla HFC-YOLO11s modeli, AI-TOD’da %3.4 mAP50 ve %2.0 mAP50-95 artışı sağlarken, %27.4 daha az parametre kullanmıştır. VisDrone testinde ise %2.7 mAP50 ve %1.3 mAP50-95 iyileşmesi elde edilmiştir. Model, küçük nesnelerin tespiti için doğruluk ve verimlilik dengesini geliştirerek başarılı sonuçlar sunmuştur.

Huang ve arkadaşları (Huang et al., 2025), uzaktan algılama görüntülerinde farklı yönelimlere sahip gemilerin tespiti için gerçekleştirmek amacıyla YOLOv11 modeline dayalı hafif ve hızlı bir nesne tespit yöntemi olan LW-YOLO11 algoritmasını geliştirmişlerdir. Modelde, çok ölçekli genişletilmiş dikkat modülü (MSDA), çapraz evreli kısmî yapı (CSPStage) ve GSConv modülü gibi mimari iyileştirmeler entegre edilmiştir. Çalışmada kullanılan HRSC2016 ve MMSHIP veri setleri sırasıyla 1061 ve 5513 gemi görüntüsünden oluşmaktadır; her iki veri seti de %80 eğitim, %10 doğrulama ve %10 test olarak ayrılmıştır. Model, 640x640 piksel boyutunda görüntülerle, SGD optimizasyon algoritması ve 2000 dönem boyunca eğitilmiştir. Yapılan deneylerde LW-YOLO11, HRSC2016 veri setinde %97.6 mAP@0.5 ve %90.1 mAP@0.5:0.95; MMSHIP veri setinde ise %90.1 mAP@0.5 ve %86.7 mAP@0.5:0.95 başarımları göstermiştir. Modelin toplam boyutu yalnızca 7.4 MB olup, gerçek zamanlı tespit için uygun olduğu ve YOLO11n, YOLOv8n gibi güncel yöntemlere kıyasla daha yüksek doğruluk sağladığı belirtilmiştir.

Wang ve arkadaşları (Wang et al., 2025), İHA görüntülerinde küçük nesnelerin daha doğru tespiti amacıyla YOLOv11 modelini geliştirerek S-YOLOv11 adını verdikleri yeni bir algoritma önermiştir. Modelde EMAFPN boyun yapısı, ESDCDH tespit başlığı ve NWD-MPD-IoU kayıp fonksiyonu gibi üç temel iyileştirme gerçekleştirilmiştir. VisDrone2019 veri seti (6471 eğitim, 548 doğrulama, 1580 test), WiderPerson (8000/1000/4382) ve CARPK (989/-/459) veri setleri kullanılmış; model SGD optimizasyon algoritmasıyla 300 dönem eğitilmiştir. S-YOLOv11, VisDrone2019 veri setinde YOLOv11s'e göre mAP@0.5'te %11.8, mAP@0.5:0.95'te %8.1, kesinlikte %9.7 ve duyarlılıkta %9.8 iyileşme sağlamış; ayrıca WiderPerson'da %1.2 ve CARPK'ta %1.0 doğruluk artışı elde edilmiştir. Diğer YOLO ve geleneksel modellerle karşılaştırıldığında, S-YOLOv11 daha yüksek doğruluk ve genel başarı göstererek küçük nesne tespiti açısından üstün performans sergilemiştir.

İHA görüntülerindeki küçük nesnelerin daha doğru ve hızlı tespiti amacıyla Ren ve ekibi (Ren et al., 2025) YOLOv11n modelini geliştirerek EfficiBackbone, C3K2-RVB ve Dynamic Head modüllerini entegre ettikleri yeni bir yapı önermiştir. EfficiBackbone, EfficientViT blokları ve kendiliğinden dikkat mekanizması ile küresel bilgi yakalamayı artırırken, C3K2-RVB modülü RipViT bloklarıyla iyileştirilerek çok ölçekli özelliklerin daha etkin bir şekilde birleştirilmesini sağlamıştır. Ayrıca tespit başlığı, ölçek-farkındalıklı dikkat mekanizmalarını içeren dinamik yapı ile değiştirilmiştir. Modelin başarı oranı VisDrone2019 veri seti kullanılarak değerlendirilmiş, toplam 300 dönem boyunca 640×640

özünürlükte eğitim yapılmıştır. Önerilen model %39.7 mAP@0.5 ve %23.6 mAP@0.5:0.95 elde ederek YOLOv11n'a kıyasla sırasıyla %6.5 ve %4.3 iyileşme göstermiş; YOLOv5s, YOLOv8s, YOLOv10s ve Faster R-CNN gibi modellerden daha yüksek doğruluk sağlamıştır. Ayrıca, hesaplama verimliliği açısından ACAM-YOLO'ya göre 10 kat daha az hesaplama maliyeti ile çalışmıştır. Modelin toplam hesaplama yükü 12.6 GFLOPs olarak belirtilmiş olup, düşük kaynaklı cihazlarda çalışmaya uygun bir yapı sunduğu ifade edilmiştir.

Dewangan ve Srinivas (B. Dewangan & Srinivas, n.d.), İHA görüntülerinde küçük, yoğun ve üst üste binen nesnelerin daha doğru şekilde tespiti amacıyla LIGHT-YOLOv11 adlı geliştirilmiş bir nesne tespit modeli önermiştir. Modelin omurga yapısında Focus bloğu ve C3TR bloğu gibi mimari iyileştirmeler entegre edilmiştir. Focus bloğu, girdilerin mekânsal bilgisini yoğunlaştırırken, C3TR bloğu dikkat mekanizmaları ve çoklu başlık yapısıyla hem yerel hem de küresel anlam ilişkilerini güçlendirerek özellik çıkarımını geliştirmiştir. Model, VisDrone2019 veri seti (6471 eğitim, 548 doğrulama, 3190 test görüntüsü) üzerinde 640x640 çözünürlükte 16 batch size ve 0.01 öğrenme oranı ile SGD algoritması kullanılarak eğitilmiştir. LIGHT-YOLOv11 modeli, temel modele kıyasla %0.453 mAP@50 ve %0.280 mAP@50:95 değerleriyle daha yüksek başarı sağlamıştır. Ayrıca, kamyon ve minibüs gibi zor sınıflarda doğruluk oranlarını da yükseltmiştir. Diğer modellerle karşılaştırıldığında, LIGHT-YOLOv11 %45.3 mAP@50 başarımı ile YOLOv11-M (%44.8), DINO (%37.8) ve Faster R-CNN (%31.3) gibi yöntemlerden üstün performans göstermiş; 19.6 milyon parametre ve 68.7 GFLOPs ile hafif ve verimli bir yapı sunmuştur.

Lu ve arkadaşları (Lu et al., 2025), İHA görüntülerinde küçük nesnelerin daha başarılı şekilde tespiti için YOLOv11 tabanlı yeni bir model olan MASF-YOLO algoritmasını geliştirmiştir. Modelde, küçük nesnelerin detay kaybını azaltmak için yüksek çözünürlüklü P2 katmanı eklenmiş, çoklu ölçekte bağlamsal bilgi elde etmek için MFAM (Multi-scale Feature Aggregation Module) önerilmiş, arka plan parazitlerini bastırmak için IEMA (Improved Efficient Multi-scale Attention) modülü geliştirilmiş ve çok seviyeli özelliklerin etkili birleşimini sağlamak amacıyla DASI (Dimension-Aware Selective Integration) modülü entegre edilmiştir. Model, VisDrone2019 veri seti üzerinde 640x640 çözünürlükte, SGD optimizasyon algoritması ve cosine LR schedule ile 100 dönem boyunca eğitilmiştir. Deneysel sonuçlara göre MASF-YOLO-s, YOLOv11-s modeline kıyasla %4.6 mAP@0.5 ve %3.5 mAP@0.5:0.95 artış sağlamıştır. Ayrıca, %12.05 parametre boyutuna rağmen YOLOv11-m modelinden daha yüksek doğruluk ve %35.1 mAP@0.5, %28.6

mAP@0.5:0.95 değerleri elde etmiştir. Faster R-CNN, Cascade R-CNN, EfficientDet, YOLOv8-m ve TPH-YOLOv5-s gibi SOTA yöntemlerle karşılaştırıldığında MASF-YOLO-s modeli %49.2 mAP@0.5 ve %32.9 mAP@0.5:0.95 ile en yüksek doğruluğu elde etmiş, küçük nesne tespitinde doğruluk ve verimlilik açısından üstün performans göstermiştir.

Bakirci ve arkadaşları (Bakirci et al., 2024), insansız hava araçları ile yakalanan trafik görüntülerinde çok sınıflı araç tespiti ve sınıflandırması için YOLOv11 modelinin performansını analiz etmişlerdir. Çalışmada YOLOv11 modeli, YOLOv10 ile karşılaştırılmıştır. Veriler, SJRC F11 Pro İHA'sı ile farklı açılardan ve koşullardan toplanmış, 5 sınıf (otomobil, minibüs, otobüs, kamyon, motosiklet) şeklinde sınıflandırılmıştır. Veri seti %72 eğitim, %11 doğrulama ve %17 test olarak bölünmüş; etiketleme işlemleri LabelImg aracı ile yapılmıştır. YOLOv10 modeli %80.3 ortalama doğruluk, %76.2 recall, %76.1 F1 skoru ve 12 ms görüntü başı algılama süresi ile test edilmiştir. YOLOv11 ise %83.2 ortalama doğruluk, %78.1 recall, %79.8 F1 skoru ve 15 ms algılama süresi ile daha yüksek doğruluk ve dengeli tespit performansı sağlamıştır. YOLOv11, tüm araç kategorilerinde daha iyi sonuçlar verirken özellikle otomobil ve otobüs sınıflarında üstünlük göstermiştir.

Xuejian Gong ve arkadaşları tarafından sunulan "AED-YOLO11: A small object detection model based on YOLO11" adlı çalışmada (Gong et al., 2025), küçük nesne tespitine yönelik zorlukların üstesinden gelmek amacıyla YOLO11 tabanlı yeni bir model geliştirilmiştir. AED-YOLO11 modeli, frekans alanında özellik toplama (AFDA), verimli dikkat sıkıştırması (EAC), dinamik örnekleme (DySample), gelişmiş kayıp fonksiyonu (WIoU) ve küçük nesnelere özel optimize edilmiş P2 tespit başlığı olmak üzere beş yeni modül içermektedir. Çalışmada kullanılan veri setleri arasında VisDrone2019 (6471 eğitim, 548 doğrulama, 1610 test), TinyPerson (794 eğitim, 816 test), Impact Moon Craters (8756 eğitim, 1545 doğrulama), DOTAv1.0 (2806 4K görüntü, %50 eğitim, %33 test, %16 doğrulama) yer almaktadır. Tüm eğitimler 640x640 çözünürlükte, 300 dönem, 16 yığın boyutu, öğrenme oranı 0.01 ve SGD optimizasyonu ile gerçekleştirilmiştir. AED-YOLO11 modeli, VisDrone2019 veri setinde YOLO11'e kıyasla %4.2 mAP@0.5, %2.6 mAP@0.5:0.95 artışı; TinyPerson setinde %2.7 mAP@0.5, %1.2 mAP@0.5:0.95 artışı; Impact Moon Craters setinde %1.5 mAP@0.5 ve %2.6 mAP@0.5:0.95 artışı; DOTAv1.0 setinde ise %1.9 mAP@0.5 ve %1.5 mAP@0.5:0.95 artışı sağlamıştır. Ayrıca, her dört veri setinde de parametre sayısında 0.7M azalma ve gerçek zamanlı FPS değerlerinde iyileşme gözlemlenmiştir.

Gao ve arkadaşları tarafından yapılan bu çalışmada (Gao et al., 2025), küçük boyutlu ve uzun mesafeli insansız hava aracı tespiti için geliştirilen hafif ve optimize edilmiş bir YOLOv11 tabanlı model sunulmuştur. Önerilen modelde omurga kısmında HWD (Haar Wavelet Downsampling) modülü kullanılarak parametre sayısı azaltılmış ve özellik kaybı minimuma indirilmiştir. Boyun yapısında daha hafif bir CCFM modülü kullanılmış, büyük ölçekli tespit başlığı çıkarılarak yerine küçük hedefler için yeni bir P2head tespit başlığı eklenmiştir. Modelin başarımı, 10.000 görüntü içeren DUT ANTI-UAV veri seti kullanılarak değerlendirilmiştir. Veri seti %52 eğitim, %22 test, %26 doğrulama şeklinde bölünmüştür. Önerilen model, temel YOLOv11 mimarisine göre %4 kesinlik, %4.5 duyarlılık, %4.1 mAP@0.5 ve %4.9 mAP@0.5:0.95 artışı sağlamış, aynı zamanda parametre sayısını %38.4 oranında azaltmıştır. Ancak modelin FPS değeri yaklaşık %5 düşmüştür. Ayrıca önerilen model, YOLOv3, YOLOv5, YOLOv6, YOLOv8, YOLOv9, YOLOv10 ve RT-DETR modelleri ile karşılaştırılmış; tüm doğruluk temelli metriklerde üstünlük sağlamış ve karmaşık arka planlı senaryolarda daha düşük yanlış tespit oranı ve daha yüksek güven skorlarıyla daha başarılı sonuçlar vermiştir.

Srigrarom ve arkadaşları (Srigrarom et al., 2020) kuşlar ile dronları ayırt etmek için görüntü tabanlı görünüş odaklı yaklaşımlar yerine uçuş yörüngesi karakteristiklerini kullanmıştır. Çalışmada video verilerinden elde edilen konum bilgileri ile dört temel özellik; dönüş açısı, periyodiklik, eğrilik ve hız/ivme hesaplanmış ve Ana Bileşen Analizi (PCA) ile iki temel özelliğe indirgenmiştir. Sınıflandırma doğrusal olmayan çekirdekli Destek Vektör Makineleri (SVM) ile gerçekleştirilmiş ve 24 video sadece kuş, 18 video sadece drone görüntüsünden oluşan toplam 42 video, ve bu videolarda toplam 1192 uçuş yolu barındıran veri seti üzerinde %90 güven seviyesinde %80'in üzerinde doğruluk elde edilmiştir.

Real-Time and Accurate Drone Detection in a Video with a Static Background başlıklı çalışmada (Seidaliyeva et al., 2020), statik arka plana sahip videolarda insansız hava araçlarının gerçek zamanlı ve doğru bir şekilde tespit edilmesi için iki aşamalı bir yöntem geliştirilmiştir. İlk aşamada, hareketli nesnelerin tespiti arka plan çıkarma yöntemi ile gerçekleştirilmiş, ikinci aşamada ise bulunan nesneler üç sınıfa (drone, kuş, arka plan) ayrılarak sınıflandırılmıştır. Sınıflandırıcı olarak düşük hesaplama maliyetli ve hızlı MobileNetV2 tabanlı evrimsel sinir ağı (CNN) kullanılmıştır. Eğitim için Drone-vs-Bird Challenge veri setinden elde edilen 24.075 görüntü (10.155 drone, 1.921 kuş, 9.348 arka plan ve ek olarak 2.651 kuş görüntüsü) kullanılmış ve eğitim %80-20 oranında bölünmüştür.

Sistem NVIDIA GT1030 GPU üzerinde eğitilmiş ve testlerde IoU=0.5 için %70.1 kesinlik, %78.8 duyarlılık ve %74.2 F1 skoruna ulaşmıştır.

Aker ve arkadaşları (Aker & Kalkan, 2017), video görüntülerinde dronların konumunun doğru bir şekilde belirlenmesi için uçtan uca evrimsel sinir ağı tabanlı bir nesne tespit yöntemi önerilmiştir. Çalışmada YOLOv2 mimarisi, dronlar ile kuşların ayırt edilmesini de içerecek şekilde iki sınıflı nesne tespiti için yeniden düzenlenmiştir. Eğitim için herkese açık drone ve kuş görüntülerinin arka plan çıkarma yöntemiyle birleştirilmesiyle oluşturulmuş, farklı sahne varyasyonlarını içeren 676.534 görüntüden oluşan büyük ölçekli yapay bir veri seti hazırlanmıştır. Ağ, ImageNet ağırlıkları ile başlatılıp Drone-vs-Bird Challenge veri seti ve oluşturulan yapay veri setiyle ince ayar (fine-tuning) yöntemiyle eğitilmiştir. Deneylerde Precision ve Recall değerlerinin her ikisinin de 0.9 seviyesine ulaştığı, küçük ve uzak hedeflerde dahi başarılı sonuçlar verdiği gösterilmiştir.

Kassab ve arkadaşlarının yaptığı çalışmada (Kassab et al., 2024), dron tespiti için hem geleneksel makine öğrenmesi yöntemleri hem de derin öğrenme tabanlı tek aşamalı dedektörler karşılaştırılmıştır. Öncelikle SVM ve Rastgele Orman (RF) gibi geleneksel yöntemlerin sınırlamaları incelenmiş, kayar pencere boyutundan kaynaklanan önyargıyı gidermek için değiştirilmiş bir maksimum olmayan bastırma (MNMS) algoritması geliştirilmiştir. Bu yöntem, geleneksel yöntemlerde %25'e varan hassasiyet artışı sağlamıştır. Derin öğrenme tarafında, SSD mimarisi AdderNet filtreleriyle entegre edilerek çarpma işlemleri azaltılmış ve model karmaşıklığı %43,42 oranında düşürülmüştür. Eğitim ve testler Drone-Vs-Bird, Anti-UAV-IR ve Anti-UAV-RGB veri setleri üzerinde gerçekleştirilmiştir.

Mahdavi ve ekibinin çalışmalarında (Mahdavi & Rajabi, 2020), dronların tespit edilmesi için evrimsel sinir ağı, destek vektör makinesi, ve en yakın komşu yöntemleri karşılaştırılmıştır. Çalışmada balıkgözü kamera ile kaydedilmiş videolardan elde edilen görüntüler MATLAB ortamında işlenmiş, dron ve kuş görüntüleri ayrılarak toplam 712 görüntüden oluşan veri seti oluşturulmuştur. Bu veri setinin %80'i eğitim, %20'si test için kullanılmıştır. Önerilen evrimsel sinir ağı mimarisi üç konvolüsyon katmanı, üç havuzlama katmanı ve iki tam bağlı katmandan oluşmuş ve 80 epoch boyunca eğitilmiştir. Sonuçlar, CNN'in %93 doğruluk, %96 kesinlik ve %91 duyarlılık ile %88 doğruluk oranı veren SVM ve %80 doğruluk oranı KNN yöntemlerinden üstün olduğunu göstermiştir.

Coluccia ve ekibinin yaptıkları çalışmada (Coluccia et al., 2021), 2020 Drone vs. Bird Detection Challenge kapsamında geliştirilen üç farklı derin öğrenme tabanlı yaklaşım; Gradient, EagleDrone, Alexis karşılaştırılmıştır. Kullanılan veri seti, farklı drone türleri ve zorlu sahne koşullarını içeren 77 eğitim ve 14 test videosundan oluşmuştur. Gradient ekibi, küçük hedefler için optimize edilmiş özel anchor yapısına sahip Cascade R-CNN mimarisini kullanmış; EagleDrone ekibi, hassasiyet öncelikli Faster R-CNN ve super-resolution teknikleriyle çalışmış; Alexis ekibi ise küçük dronları daha iyi tespit edebilmek için görüntü döşeme (tiling) stratejisi ve YOLOv3-SPP mimarisini uygulamıştır. Ek olarak, Alexis ekibi 26.500 sentetik görüntü üretmiş ve stil aktarımı ile veri çeşitliliğini artırmıştır. Yarışma sonuçlarına göre Gradient %80, Alexis %79,8 ve EagleDrone %66,8 AP değerlerine ulaşmıştır. Çalışma, farklı yaklaşımların tamamlayıcı özellikler sunduğunu ve küçük hedef tespiti ile hareketli kamera sahnelerinin en zorlayıcı durumlar olduğunu göstermektedir.

1.3 Literatürün Genel Görünümü

Yapılan literatür taramasında, YOLO tabanlı nesne tespiti alanında yapılmış güncel çalışmalar incelenmiştir. Özellikle İHA (drone), kuş ve benzeri hava hedeflerinin tespiti üzerine yapılan araştırmalarda YOLOv4, YOLOv5, YOLOv7, YOLOv8, YOLOv9 ve YOLOv10 gibi farklı sürümlerin kullanıldığı gözlemlenmiştir. Yapılan çalışmalar doğruluk, algılama hızı ve model karmaşıklığı açısından karşılaştırılmış; farklı mimari iyileştirmeler, veri artırma teknikleri ve optimizasyon stratejileri kullanılarak model performansları artırılmaya çalışılmıştır.

YOLO modelleri, farklı veri kümeleri ve çevresel koşullar altında yürütülen drone tespiti çalışmalarında başarılı sonuçlar ortaya koymuştur. Özellikle drone ve kuş ayrımı gibi zorlayıcı durumlarda, model mimarilerine yapılan özgün katkıların tespit doğruluğunu önemli ölçüde artırdığı gözlemlenmektedir. Araştırmalarda; hava koşulları, nesneler arası mesafe ve arka plan karmaşası gibi faktörlerin model performansını doğrudan etkilediği belirtilmiştir. Ayrıca, YOLO modellerinin yalnızca drone tespiti değil, güvenlik, trafik izleme ve orman yangını erken uyarı sistemleri gibi farklı uygulama alanlarında da yaygın şekilde tercih edildiği görülmektedir.

Yapılan literatür taraması, drone-kuş ayrımı gibi zorlu senaryolar için geliştirilen yöntemlerin çoğunlukla YOLO gibi derin öğrenme tabanlı tek aşamalı dedektörler veya klasik makine öğrenmesi yaklaşımlarına dayandığını ortaya koymuştur. Çalışmaların önemli bir kısmı sınırlı veri setleri, belirli sahne koşulları ya da tek tip hedef özellikleri ile

sınırlandırılmış olup, küçük ve hızlı hareket eden hedeflerin gerçek zamanlı tespiti açısından kısıtlı başarı sunmuştur. Bununla birlikte, Drone-vs-Bird Grand Challenge gibi karmaşık ve çeşitliliği yüksek veri kümelerinin kullanımı, bu alanda daha genellenebilir ve güvenilir yaklaşımlar geliştirilmesine olanak tanımıştır. Son yıllarda yapılan araştırmalar, model mimarilerinin hafifletilmesi, veri artırma stratejileri ve hiperparametre optimizasyonu gibi konulara yoğunlaşmış, doğruluk-hız dengesinin iyileştirilmesini hedeflemiştir. Bu bağlamda, çalışmamızda güncel bir mimari olan YOLOv11 kullanılarak drone tespiti üzerine odaklanılmış, literatürdeki mevcut boşluklar ve iyileştirme alanları dikkate alınarak performans değerlendirmesi yapılmış ve küçük, hızlı hareket eden hava hedeflerinin daha hassas ve hızlı şekilde tespit edilmesine katkı sağlanmıştır.

1.4 Görüntü İşleme Adımları

Derin öğrenme tabanlı nesne tespit sistemleri, bir görüntü üzerindeki hedef nesneyi tespit edebilmek için ardışık ve sistematik işlemler gerçekleştirir. Bu bölümde detaylı mimari akışa girilmeden süreç genel olarak özetlenmektedir.

Derin öğrenme modelleri, piksel değerlerinden oluşan bir görüntüyü girdi olarak alır. Eğer girdi bir video verisi ise belirli aralıklarla alınan kareler modele giriş verisi olarak sunulur. Bu görüntüler genellikle RGB (renkli) veya gri ton (grayscale) formatındadır. Görüntünün çözünürlüğü ve kanal bilgisi, eğitilecek olan modele göre uyarlanmalıdır.



Şekil 1.1: Girdi Görüntüsü

Görüntüler modele verilmeden önce çeşitli ön işleme adımlarından geçirilmektedir. Bu işlemlerden bazıları yeniden boyutlandırma (bu çalışmada 640 x 640 pikseller kullanılmıştır), normalizasyon (piksel değerlerini 0-1 aralığına indirgeme) ve renk dönüşümleri gibi adımlardır. Bu işlemler, hem modelin daha kararlı öğrenmesini sağlar hem de eğitim sürecini hızlandırır.

Ön işleme sonrası veriler, konvolüsyon katmanlardan geçirilmektedir ve bu katman aracılığı ile verilerden önemli bilgiler çıkarılmaktadır. Bu aşamada görüntü üzerinden Çekirdek (Kernel) adı verilen bir matris gezdirilir. Bu işlem sonrasında model tarafından yorumlanabilecek Özellik Haritası (Feature Map) üretilir.

Elde edilen özellik haritaları, modelin nesne tespiti yapabilmesi için kullanılır. Tespit edilen nesnelere ait sınıf etiketleri ve konum bilgileri belirlenerek çıktı üretilir. Eğer görüntüde birden fazla nesne varsa, her bir nesne için ayrı tahmin yapılır. Bu tahminler, sınırlayıcı kutular (bounding boxes) aracılığıyla görselleştirilir.



Şekil 1.2: Tahmin sonucu elde edilen sınırlayıcı kutu örneği

Model eğitimi sonrasında, her tahmin kutusuna karşılık gelen sınıf etiketi ve doğruluk oranı (confidence score) görsel olarak belirtilir. Bu kutular, modelin sahada nasıl çalıştığını gözlemlemek açısından önemli çıktılar sunar.

2. DERİN ÖĞRENME

Derin öğrenme (deep learning), çok katmanlı yapay sinir ağları kullanarak karmaşık verilerden önceliklendirilmiş özellikler öğrenilmesini sağlayan bir makine öğrenmesi yaklaşımıdır (Shinde & Shah, 2018). Bu yöntemler, konuşma tanıma, görsel nesne tanıma, nesne tespiti ve ilaç keşfi ve genomik gibi birçok diğer alanda son teknolojiyi önemli ölçüde iyileştirmiştir (Lecun et al., 2015). Derin öğrenme, verideki kalıpları öğrenmek için katmanlı yapı üzerinden ileri hesaplamalar yürütür ve geri yayılım algoritmasıyla bu katmanlardaki hesaplamaları yöneten parametreleri adım adım güncelleyerek öğrenir (Lecun et al., 2015). Derin öğrenme mimarileri, geleneksel yöntemlerin ihtiyaç duyduğu el ile özellik çıkarımını ortadan kaldırarak, ham verilerden öğrenme sağlar. Bu yönüyle hem otomasyon hem de doğruluk açısından önemli avantajlar sağlamaktadır.

"What Is Machine Learning, Artificial Neural Networks and Deep Learning?—Examples of Practical Applications in Medicine" başlıklı çalışmada (Kufel et al., 2023), derin öğrenmenin özellikle tıp alanındaki uygulamalarına dair geniş kapsamlı örnekler sunulmuştur. Derin öğrenme mimarilerinin insan uzmanlarla yarışacak düzeyde hassas sonuçlar üretebildiği ve bazı alanlarda insan hatasını azaltarak güvenilirliği artırdığı ifade edilmektedir. Benzer biçimde, güvenlik alanında da yüksek doğruluk gerektiren görüntü analizi uygulamalarında derin öğrenme sistemlerinin benzer başarılarla imza attığı görülmektedir.

Derin öğrenme, yalnızca teknolojik bir yenilik değil; sağlık, güvenlik ve çevre gibi disiplinlerde karar destek sistemleri açısından stratejik bir araç haline gelmiştir. Karmaşık verilerin analizi, sınıflandırılması ve yorumlanması gibi görevlerde, bu mimarilere dayalı çözümler yüksek başarı potansiyeli sunmaktadır. Tez kapsamında incelenen YOLOv11 gibi modern derin öğrenme modelleri, yalnızca tıpta değil, aynı zamanda güvenlik ve gözetleme gibi alanlarda da gerçek zamanlı nesne tespiti gibi kritik görevleri başarıyla üstlenmektedir.

Derin öğrenmenin sunduğu bu yüksek potansiyelin temelinde, onu oluşturan matematiksel yapı taşları yer almaktadır. Bu mimarilerin başarısı büyük ölçüde, katmanlar arasında kurulan işlevsel bütünlüğün sağlanmasına bağlıdır. Bu nedenle, her bir katmanın

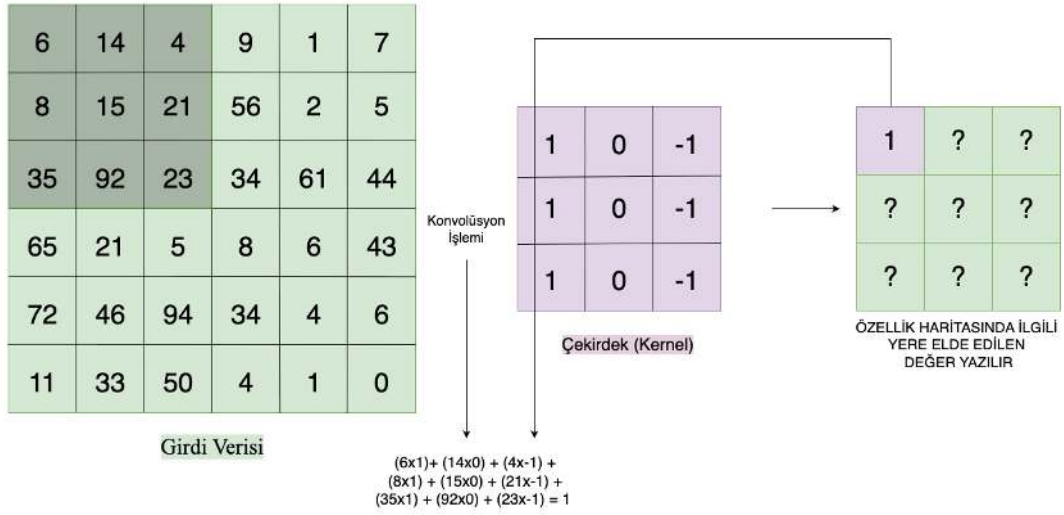
sistem içerisindeki rolünün ayrı ayrı değerlendirilmesi, genel başarıyı anlamlandırmak açısından önemlidir. Özellikle evrişim, aktivasyon ve havuzlama gibi temel yapıların derinlemesine analizi, mimarinin nasıl işlediğini kavramada kritik öneme sahiptir. Bir sonraki bölümde, bu yapısal bileşenlerin işlevleri ve modele olan katkıları detaylı biçimde ele alınacaktır.

2.1 Evrişim Katmanı

Yapay sinir ağlarının temel yapı taşlarında, giriş olarak alınan bir vektör, ağırlıklardan oluşan bir matris ile çarpılarak çıktı üretilir. Bu işleme, sabit bir sapma (bias) değeri eklenir ve ardından doğrusal olmayan bir aktivasyon fonksiyonuna tabi tutulur. Bu işlem prensip olarak tüm veri türleri için geçerlidir; ancak bazı veri türlerinin sinir ağına uygun hale getirilebilmesi için düzleştirme (flattening) adı verilen işlemle tek boyutlu hale getirilmesi gerekir.

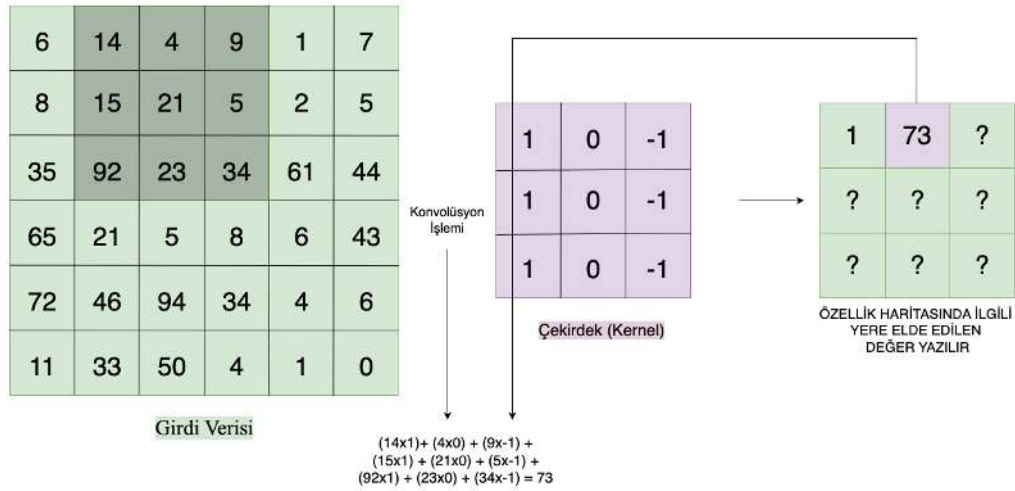
Ancak görüntü, ses veya video gibi veriler yalnızca sayısal değerlerden oluşmaz; aynı zamanda bu veriler içinde belirli bir uzamsal ve yapısal düzen taşır. Bu tür verileri düzleştirmek, örneğin bir pikselin çevresindeki piksellerle olan konumsal ilişkisini ortadan kaldırır. (Dumoulin & Visin, 2016) tarafından da belirtildiği üzere, bu durum verinin içerdiği topolojik bilginin kaybına yol açabilir. İşte bu noktada, konvolüsyon katmanları devreye girer ve verinin yapısal bütünlüğünü koruyarak daha anlamlı özellikler çıkarılmasını sağlar.

Konvolüsyon (evrişim) katmanı, giriş verisi üzerinde küçük boyutlu bir çekirdek matrisinin kaydırılması ile çalışır (Dumoulin & Visin, 2016). Eğer girdi olarak verilecek veri bir görüntü ise çekirdek -1, 0 veya 1'lerden oluşan 2 boyutlu bir matris olacaktır (Mishra et al., 2018). Karşılık gelen çıktı çekirdeğin şeklinde göre şekil alacaktır (Mishra et al., 2018). Çekirdek, belirlenen adım (stride) sayısı kadar kayrılarak tüm görüntü üzerinde gezdirilir. Her konumda çekirdek ile giriş verisi arasında yapılan çarpımların toplamı, o bölgeden çıkarılan özelliği temsil eder. Böylece evrişim katmanları, daha az parametreyle işlem yaparak daha verimli bir öğrenme yapılmasını sağlar ve verinin konumsal bütünlüğü de korunmuş olur. Elde edilen bu sonuca özellik haritası denilir ve bir sonraki katmanda girdi olarak özellik haritası verilir.

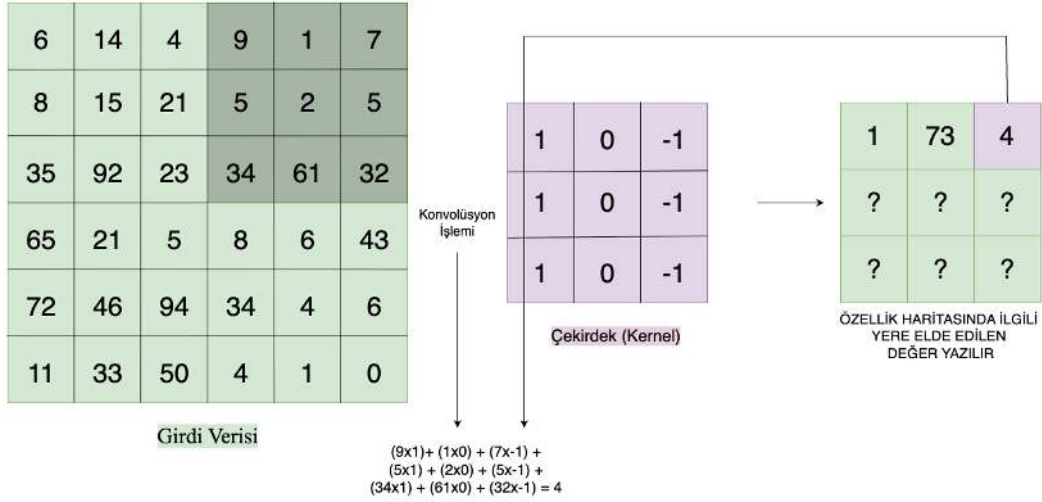


Şekil 2.1: Özellik haritası oluşturmak için çekirdek ve girdi verisi işlemi ilk adım

Konvolüsyon işlemine başlanırken, çekirdek matrisinin ilk konumu giriş verisinin sol üst köşesine yerleştirilir (Şekil 2.1). İki matrisin çarpım sonucu elde edilen toplam değer, özellik haritasındaki ilk hücreye yazılır.

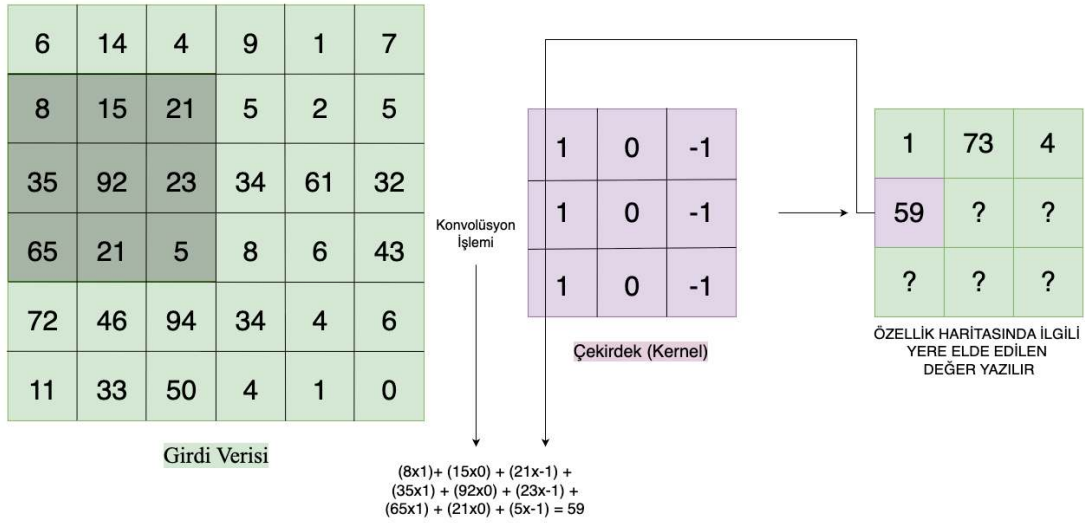


Şekil 2.2 Özellik haritası oluşturmak için çekirdek ve girdi verisi işlemi ikinci adım



Şekil 2.3: Özellik haritası oluşturmak için çekirdek ve girdi verisi işlemi üçüncü adım

Adım sayısı 1 olarak belirlendiğinde, çekirdek yatay olarak bir sütun sağa kaydırılır ve aynı işlem tekrarlanır (Şekil 2.2). Satırın sonuna ulaşıldığında, çekirdek bir satır aşağı indirilir ve yeniden en baştan işlem sürdürülür (Şekil 2.3).

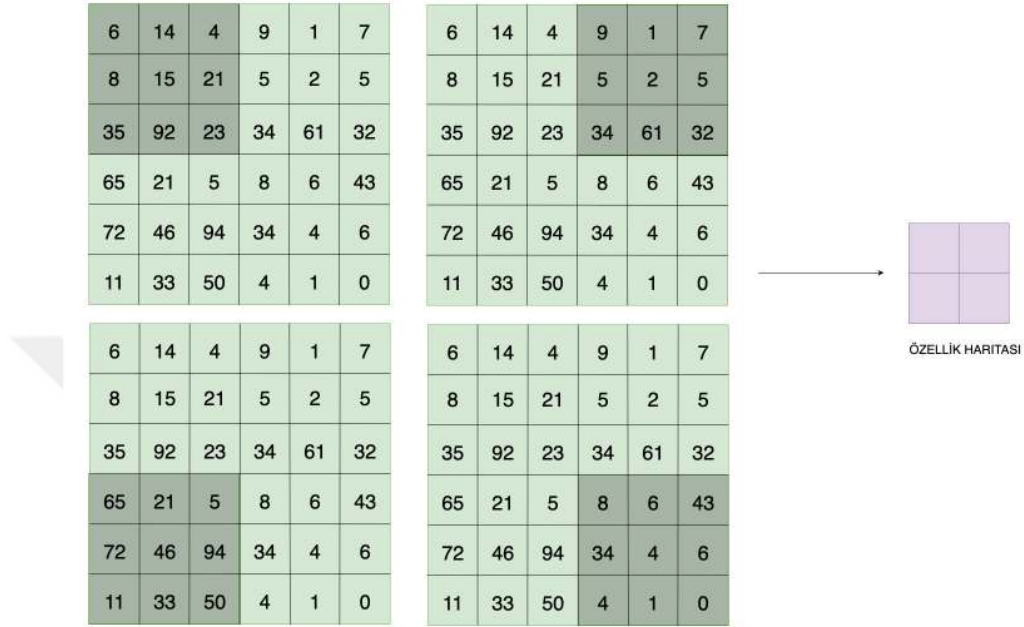


Şekil 2.4: Özellik haritası oluşturmak için çekirdek ve girdi verisi işlemi dördüncü adım

Elde edilen son matris bir sonraki adımda girdi verisi olarak kullanılacaktır. Bu işlem, çekirdeğin tüm görüntü üzerinde gezinmesiyle tamamlanır. Elde edilen son çıktı, özellik haritası olarak adlandırılır ve modelin bir sonraki katmanına giriş olarak verilir. Böylece hem konumsal bilgi korunur hem de daha az parametreyle verimli öğrenme gerçekleştirilmiş olur.

2.2 Adım Ve Dolgu

Konvolüsyon işleminde çekirdek ve girdi matrisi satır satıra çarpılır. Bu işlem bittikten sonra çekirdek adım sayısı kadar kaydırılır ve tekrar aynı işlem yapılır. Ancak her durumda elde ettiğimiz özellik haritasının boyutu, girdi boyutu ile eşit olmaz.



Şekil 2.5: Adım sayısı 3 olduğu taktirde elde edilecek özellik haritası boyutu

Adım değerini 1'den büyük bir değer verilirse, Şekil 2.5'te olduğu gibi 3 verilirse, uygulanan konvolüsyon sonucunda elde edilecek özellik haritasının boyutu küçülecektir. Adım sayısı arttığında özellik haritasının boyutu küçülür ve hesaplama maliyeti azalır. Ancak bu durum bazı ayrıntıların kaybolmasına yol açabilir.

Görüntünün kenarlarına sıfırlardan oluşan değerler eklenerek uygulanan dolgu işlemi, konvolüsyon sonucunda oluşacak çıktı boyutunu kontrol etmek ve özellikle kenar piksellerin de işleme yeterince dahil edilmesini sağlamak amacıyla kullanılır. Kenarlarda bulunan pikseller, görselin ortasındaki piksellere kıyasla çekirdeğin altında daha az sayıda kalır ve bu da eğitim sırasında bilgi kaybına yol açabilir.

6	14	4	9	1	7
8	15	21	5	2	5
35	92	23	34	61	32
65	21	5	8	6	43
72	46	94	34	4	6
11	33	50	4	1	0

6	14	4	9	1	7
8	15	21	5	2	5
35	92	23	34	61	32
65	21	5	8	6	43
72	46	94	34	4	6
11	33	50	4	1	0

6	14	4	9	1	7
8	15	21	5	2	5
35	92	23	34	61	32
65	21	5	8	6	43
72	46	94	34	4	6
11	33	50	4	1	0

Şekil 2.6: Adım sayısı kaydıkça dahil edilen pikseller

Dolgu işlemi uygulanmadığı zaman kenarda kalan pikseller konvolüsyon işlemine diğer pikseller kadar fazla dahil edilmemektedir. Şekil 2.6’da verilen görsellerde görüldüğü gibi, çekirdek matrisi kaydırıldığında en kenarda bulunan 6 numaralı pikselin işleme bir kere dahil edildiği ancak 4,9 gibi piksellerin konvolüsyon işlemine en az 3 kere dahil edildiği görülmektedir.

0	0	0	0	0	0	0	0
0	6	14	4	9	1	7	0
0	8	15	21	5	2	5	0
0	35	92	23	34	61	32	0
0	65	21	5	8	6	43	0
0	72	46	94	34	4	6	0
0	11	33	50	4	1	0	0
0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0
0	6	14	4	9	1	7	0
0	8	15	21	5	2	5	0
0	35	92	23	34	61	32	0
0	65	21	5	8	6	43	0
0	72	46	94	34	4	6	0
0	11	33	50	4	1	0	0
0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0
0	6	14	4	9	1	7	0
0	8	15	21	5	2	5	0
0	35	92	23	34	61	32	0
0	65	21	5	8	6	43	0
0	72	46	94	34	4	6	0
0	11	33	50	4	1	0	0
0	0	0	0	0	0	0	0

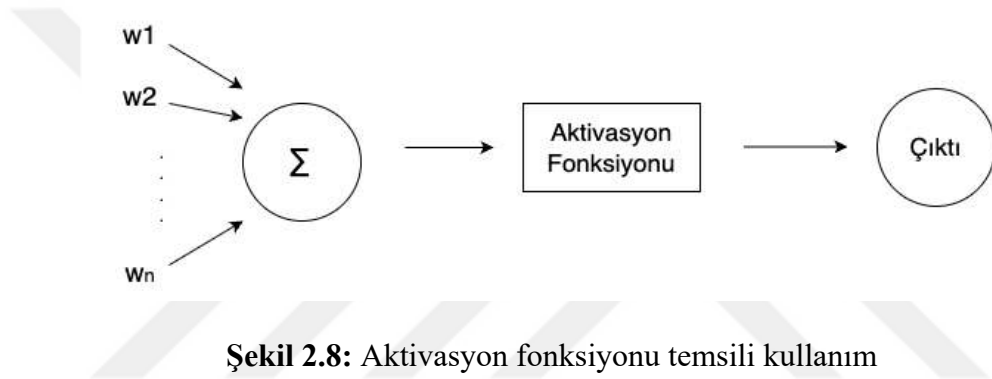
Şekil 2.7: Görüntüye dolgu eklendiği zaman çekirdek matrisi ve girdi görüntüsü

Şekil 2.7’de olduğu gibi görsele 0 değerlerinden oluşan bir dolgu eklenirse kenarlardaki piksel değeri işlem merkezine daha sık denk gelir ve konvolüsyon işlemine daha çok dahil edilirler. Dolgu işlemi yapılmasının nedeni, giriş görüntüsünün boyutlarını koruyarak çıkış görüntüsünün boyutunun azalmasını engellemektir (Yamashita et al., 2018).

$$(Giriş\ boyutu - Kernel\ boyutu + 2 \times Padding) \div Stride + 1 \quad (2.1)$$

2.3 Aktivasyon Fonksiyonu

Aktivasyon fonksiyonları sinir ağlarında nöronun çıktısını üretip üretmemesine karar verme süreci en önemli yapı taşıdır. Aktivasyon fonksiyonları, sinir ağlarında girdilerin ağırlıklı toplamı ve sapma değerini hesaplamak için kullanılan fonksiyonlardır; bu hesaplama, bir nöronun etkinleştirilip etkinleştirilmeyeceğine karar vermek için kullanılır (Nwankpa et al., 2018). Derin öğrenme algoritmalarında, öğrenme sürecinin sağlıklı bir şekilde gerçekleştirilmesi, aşırı öğrenmenin önlenmesi, doğruluk performansının artırılması ve hesaplama maliyetinin düşürülmesi gibi özellikler göz önünde bulundurularak aktivasyon fonksiyonları geliştirilmiştir (KILIÇARSLAN et al., 2021).



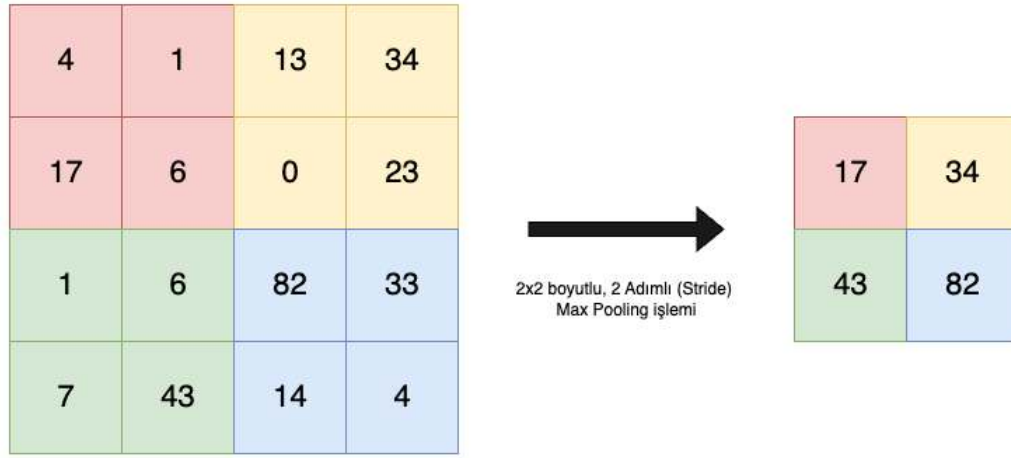
Şekil 2.8: Aktivasyon fonksiyonu temsili kullanım

Konvolüsyon işlemi sonucunda elde edilen özellik haritası, ilk olarak düzleştirme işlemine tabi tutulur. Bu işlemde çok boyutlu matris, tüm elemanları sıralı biçimde tek bir boyuta indirgenerek tek boyutlu bir vektöre dönüştürülür. Ardından bu vektör, sinir ağı içerisindeki öğrenilmiş ağırlıklar ile çarpılır ve üzerine sapma eklenir. Bu işlem sonucunda elde edilen ara sonuç, seçilen aktivasyon fonksiyonu aracılığıyla doğrusal olmayan bir forma dönüştürülür. Böylece model, verideki karmaşık ve doğrusal olmayan ilişkileri öğrenme kapasitesi artırılır.

2.4 Havuzlama Katmanı

Havuzlama katmanı, derin öğrenme sistemlerinde özellik haritalarının boyutlarını azaltmak için kullanılan önemli bir adımdır (Gholamalinezhad & Khosravi, 2020). Özellik haritalarının yüksek boyutlu olması hem bellek tüketimini artırmakta hem de işlem süresini uzatmaktadır. Bu nedenle havuzlama işlemleri, gereksiz bilgilerin elenmesini sağlarken, anlamlı temsilleri koruyarak boyut indirgeme görevini üstlenir. En yaygın kullanılan havuzlama yöntemi ise Max Pooling'dir. Bu yöntem, belirli bir bölgedeki en yüksek değeri

seçerek öne çıkarılacak bilgiyi temsil eder ve bu yolla gereksiz detayların model üzerindeki etkisi azaltılır (Zafar et al., 2022).



Şekil 2.9: Özellik haritasına Max Pooling işlemi uygulama

Yukarıda şekil 2.8’de gösterildiği gibi 4x4 boyutundaki bir özellik haritası üzerinde 2x2 boyutlu ve 2 adımlı bir kayma işlemi ile Max Pooling işlemi uygulandığında 2x2 boyutunda bir çıktı matrisi elde etmiş oluruz. Bu sayede daha az bellek kullanarak, daha az işlem ile daha hızlı bir eğitim gerçekleştiririz.

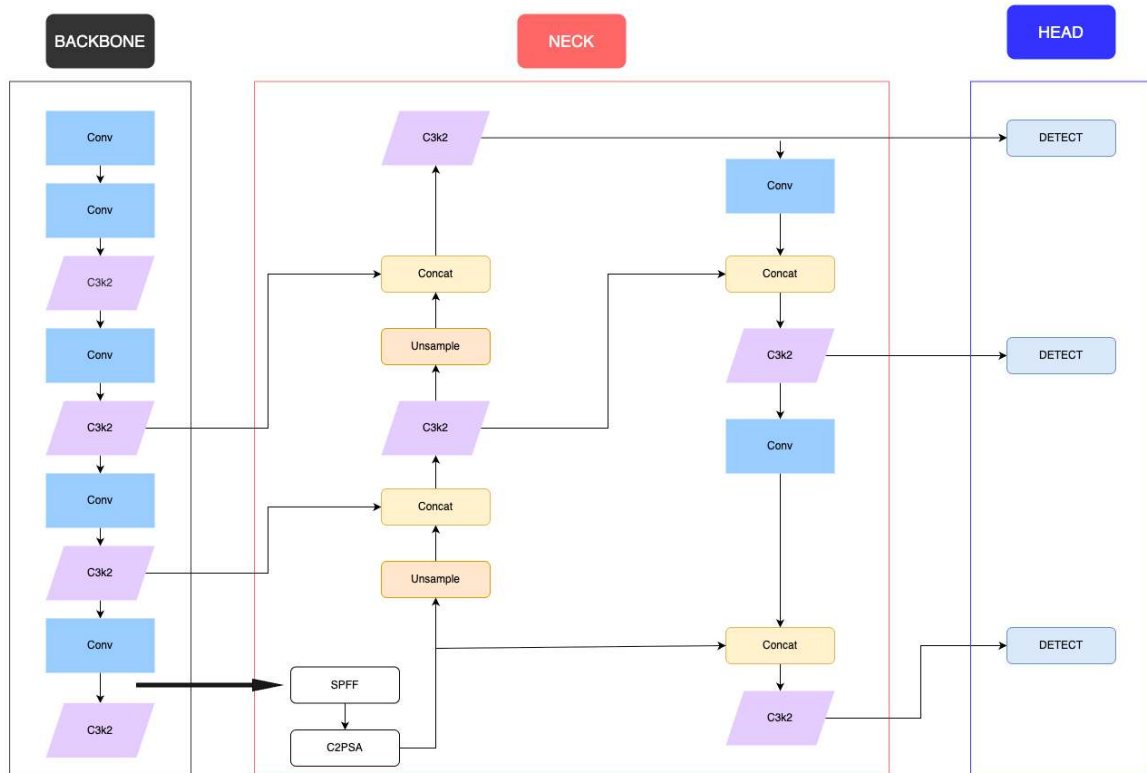
2.5 YOLOv11 Mimarisi

YOLO görüntü işleme uygulamalarında sıklıkla tercih edilen bir algoritmadır (Jiang et al., 2021). YOLO mimarilerinin tercih edilmelerinin sebebi yüksek doğruluk ve hız ile sınırlayıcı kutu ve sınıflandırma özelliklerini bir arada sunmasıdır. YOLO algoritmalarında görüntü, karmaşıklığa bağlı olarak NxN’lik ızgaralara bölünür. Algoritma her bir kutuyu kontrol eder ve herhangi bir nesnenin olduğu ızgarada sınıf tahmini yapar. Her bir ızgara X sayıdaki sınırlayıcı kutu ve bu kutulara karşılık gelen güven skorlarını tahmin eder. Güven puanı nesnenin varlığı veya yokluğu hakkında bilgi vermektedir (Tripathi et al., 2022).

YOLOv11, YOLO serisinin en güncel versiyonlarından birisidir. YOLOv11, YOLOv7 temellerine üzerine kurulmuş bir modeldir. YOLOv7, çok ölçekli özellik çıkarmayı iyileştirmeye ve gerçek zamanlı performansı korumaya odaklanırken, YOLOv11 bu yönleri iyileştiriyor ve küçük nesneleri, karmaşık ortamları daha doğru şekilde analiz edebilmek amacıyla yeni mekanizmalar entegre ediyor. (Mahala et al., 2025). YOLOv11,

geleneksel nesne tespitiin ötesinde daha geniş bir bilgisayarlı görü görev yelpazesini destekleyerek gelişmiş uyarlanabilirliğiyle öne çıkmaktadır (Khanam & Hussain, 2024).

YOLO mimarileri genel olarak üç temel bileşenden oluşur: omurga (backbone), boyun (neck) ve baş (head). Bu bileşenlerin her biri aşağıda özetlenmiştir. Omurga, görüntüden temel ve yüksek düzeyde anlamlı özellikleri çıkaran katmanlardan oluşur. Bu katmanlar, çok sayıda konvolüsyon işlemini içeren bloklar hâlinde yapılandırılmıştır (Rasheed & Zarkoosh, 2024).



Şekil 2.10: YOLOv11 Mimari Yapısı

YOLOv11, YOLOv8'deki çok görevli (multi-tasking) esnekliği korurken, C3k2 bloğu ile verimliliği artırmış; C2PSA modülüyle ise küçük ve üst üste binen nesnelerin daha doğru tespiti için mekânsal dikkat (spatial attention) mekanizması entegre etmiştir (Jegham et al., 2024). Bu yapılar sayesinde modelin farklı ölçeklerdeki nesnelere duyarlılığı artmış, genel tespit başarımı yükselmiştir.

Omurga katmanı, farklı ölçeklerdeki özellikleri birleştirir ve tahmin için Baş katmanına iletir. Bu süreç genellikle farklı seviyelerdeki özellik haritalarının yukarı örnekleme (upsampling) ve birleştirme (concatenation) işlemlerini içerir; böylece model, çok ölçekli bilgileri etkili bir şekilde yakalayabilir. YOLOv11'deki temel iyileştirmeler arasında, öz-

dikkat mekanizmalı çapraz aşamalı kısmi ağ modülü (Cross-Stage Partial with SelfAttention, C2PSA) yer almaktadır. Bu modül, çapraz aşamalı kısmi ağların avantajlarını öz-dikkat mekanizmalarıyla birleştirerek modelin katmanlar arası bilgiyi daha etkili bir şekilde yakalamasını sağlar (Jegham et al., n.d.).

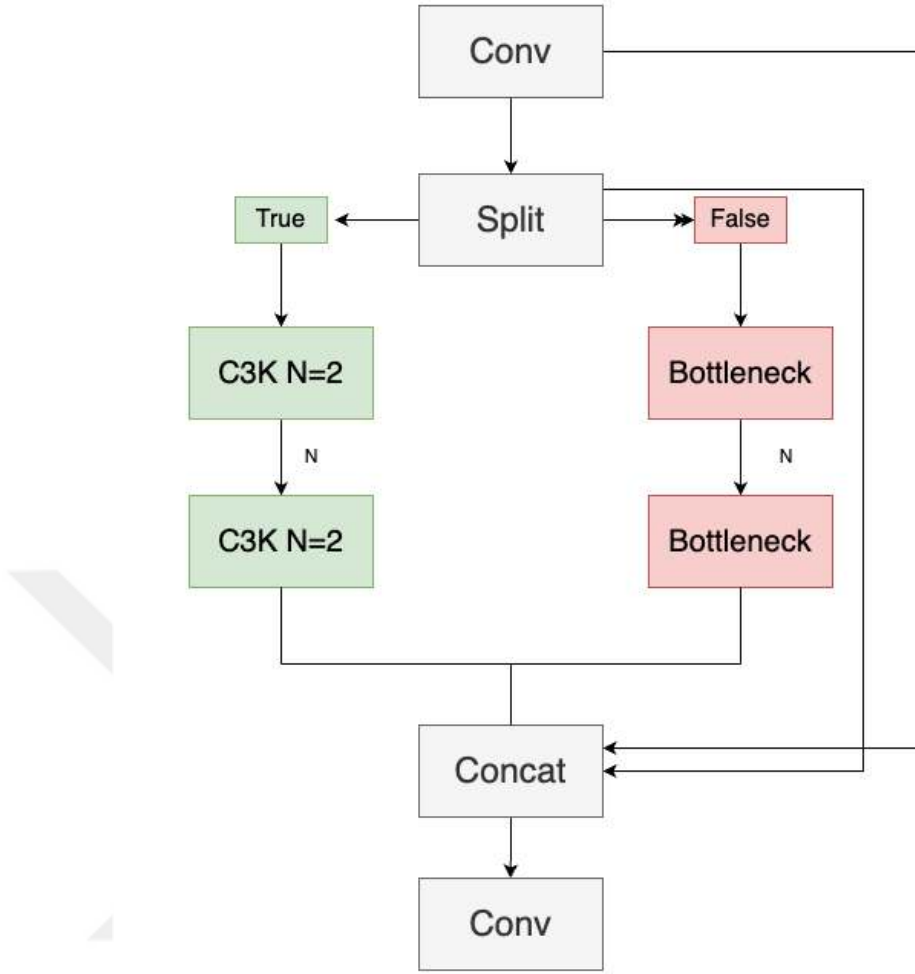
Ayrıca, YOLOv8’de kullanılan C2f bloğu yerine, daha küçük çekirdeklerle çalışan ve CSP mimarisinin özel bir versiyonu olan C3k2 bloğu tercih edilmiştir. Bu yapı sayesinde hem işlem verimliliği hem de hız artırılmış; doğruluk seviyesi korunmuştur. Omurga katmanının son aşamasında yer alan SPPF (Spatial Pyramid Pooling - Fast) ve C2PSA blokları, çoklu max pooling işlemleri ve dikkat mekanizmalarıyla modelin farklı alanlardan bilgi almasını sağlar (Rasheed & Zarkoosh, 2024).

Baş kısmı, nesne tespiti ve sınıflandırma açısından son tahminleri üretmekten sorumludur. Sınırlayıcı kutuları (bounding box) ve sınıf etiketlerini üretmek üzere, boyun kısmından gelen özellik haritalarını işler (Khanam & Hussain, 2024).

2.5.1 C3k2 bloğu

YOLOv11 mimarisinde kullanılan C3K2 bloğu, YOLOv8’deki C2f bloğunun yerini alacak şekilde geliştirilmiştir. C2f bloğu, temel olarak dar parametrelili bir yapı sunarak öz nitelik çıkarımı sağlasa da özellikle küçük nesne tespiti gibi karmaşık detayların ayrıştırılmasında yetersiz kalabilmektedir. Bu bağlamda geliştirilen C3K2 bloğu, özellik çıkarımı sürecini daha verimli hâle getirmek amacıyla CSP (Cross-Stage Partial) mimarisi temel alınarak tasarlanmıştır (Alif, 2024). Bu blok, daha küçük çekirdek boyutlarına sahip evrişimli katmanlar kullanarak karmaşık ayrıntıların daha iyi öğrenilmesini sağlar (Mahala et al., 2025).

C3K2 bloğu, özellik haritasını bölümlere ayırarak modelin temel özellikleri yakalama yeteneğini korurken hesaplama açısından daha verimli olan daha küçük 3x3 evrişimler uygulayarak ağdaki veri akışını iyileştirir (Kasamsumran et al., 2025). C3k2 bloğunun ayarlanabilir çekirdek boyutları ve parametre verimliliği sayesinde doğruluktan ödün vermeden hesaplama yükünü azaltmak mümkün hale gelmektedir (Nahar et al., 2025).



Şekil 2.11: C3K2 Blok yapısı “(He, Zhou, Liu, Zhang, et al., 2025)

Bu modülde, yapılandırmalara izin veren C3k parametresi bulunmaktadır. Şekilde görüldüğü gibi C3k değeri false olduğunda, C3k2 modülü C2f'nin orijinal tasarımını korur ve standart darboğaz modülünü kullanır; c3k değeri true olduğunda, darboğaz modülü daha verimli bir C3k modülü ile değiştirilir (He, Zhou, Liu, Zhang, et al., 2025). Darboğaz modülü, evrişimli sinir ağlarında yaygın olarak kullanılan bir yapıdır ve özellik haritasının boyutunu azaltmak, böylece hesaplama yükünü düşürmek ve azaltılmış özellik haritası üzerinde özellik çıkarımı gerçekleştirmek için tasarlanmıştır.

2.5.2 SPFF bloğu

Eski YOLO versiyonlarında kullanılan SPP (Spatial Pyramid Pooling) modülünün optimize edilmiş halidir. SPFF modülü, tespiti geliştirmek için farklı ölçeklerden gelen özellikleri bir araya getirir. YOLOv11, YOLOv8 versiyonlarında bulunan eski SPFF bloğunu korur, modelin bir görüntünün farklı bölümlerinden katman özelliklerini farklı hızlarda bir araya

getirmesine olanak tanır (Bhagwat et al., 2025). Bu modül, küçük nesne algılama gibi, görüntü parçalarını algılamasını ve ayırt etmesini sağlar. SPFF modülü, çok ölçekli bağlamsal bilgileri toplamak için birden fazla maksimum havuzlama işlemini (değişen çekirdek boyutlarıyla) kullanır.



3. YÖNTEM VE METOT

3.1 Veri Seti

Çalışma kapsamında drone tespiti yapmak için Drone-vs-Bird Grand Challenge (Coluccia et al., 2024) kapsamında paylaşılan drone ve kuş görüntülerinden oluşan videoları barındıran veri seti kullanılmıştır. Bu veri seti ile yapılan ön çalışmalar sonucunda, modelin kuş sınıfına yönelik tahminlerinin istenilen seviyede olmadığı gözlemlenmiştir. Bu nedenle, kuş sınıfına ait örneklerin çeşitliliğini artırmak amacıyla Kaggle platformunda paylaşılan “Birds Flying” adlı veri setinden rastgele seçilen 1500 kuş görüntüsü mevcut veri setine eklenmiştir. Ayrıca, veri setindeki örnek sayısını dengelemek ve farklı açı ile ortamlarda çekilmiş drone görüntülerini modele dahil etmek amacıyla yine Kaggle platformunda paylaşılan “Drone Detection” başlıklı veri setinden rastgele seçilmiş 1000 adet etiketli drone görüntüsü eklenmiştir. Böylece eğitim verisinin hem çeşitliliği hem de sınıflar arası denge seviyesi artırılarak modelin genel performansının iyileştirilmesi hedeflenmiştir.

Alınan veriler önceden etiketli veriler olmadığı için veriler labelImg uygulaması aracılığı ile etiketlenmiştir. YOLOv11 ile eğitim yapmak için görüntü kullanmamız gerektiğinden Drone-vs-Bird Grand Challenge veri setinde bulunan videolardan 1'er saniyelik kare ile görüntüler alınmış, drone veya kuş olarak etiketlenmiştir. Elde edilen veri setinde toplam 6869 görüntü bulunmaktadır. Bu veri seti %70 eğitim, %15 doğrulama ve %15 test olarak ayrıştırılmıştır. Verilerde eğitim öncesi herhangi bir ön işleme adımı uygulanmamıştır. Tüm kodlar Python programlama dili ile yazılmıştır.

Şekil 3.1’de görüldüğü üzere, bazı durumlarda drone ve kuş ayrımını çıplak gözle yapmak dahi oldukça zordur. Özellikle arka planın kalabalık ve karmaşık olduğu sahnelerde, drone nesnesi görüntü üzerinde seçilemez hâle gelebilmektedir. Başka bir sorun olan kuş ve drone objelerinin birbirini ile karışması durumuna örnek bir görsel Şekil 3.2’de sunulmuştur. Görüntü kalitesinin net olmayışı ve mesafeden ötürü kuş ve drone nesnelerinin ayrımı oldukça zor bir problemdir. Ayrıca, kayıt işlemi sırasında yapılan yakınlaştırma (zoom) nedeniyle görüntü kalitesinin düştüğü durumlar da tespit performansını olumsuz etkileyebilmektedir.



Şekil 3.1: Veri setinde bulunan drone görüntüsü örneği



Şekil 3.2: Veri setinde bulunan uçan bir kuş görüntüsü örneği



Şekil 3.3: Veri setinde bulunan aynı karede uçan drone ve kuşların olduğu görüntü örneği



Şekil 3.4: Veri setinde bulunan arka plan karmaşıklığından ötürü gözlemlenemeyen drone görüntüsü

3.2 Ön Deneyler

Yapılan ön çalışmalar kapsamında Drone-vs-Bird Grand Challenge (Coluccia et al., 2024) kapsamında paylaşılan drone ve kuş görüntülerinden oluşan video veri seti kullanılarak ön eğitimler gerçekleştirilmiştir. Bu eğitimler esnasında eğitim oranı parametresi 0.0001 ve 0.001 seçilerek 2 farklı şekilde eğitim gerçekleştirilmiştir. Her bir eğitim oranı değeri için farklı katman dondurma değeri; 0,5,10, 15 ve 20 ile farklı eğitimler gerçekleştirilmiştir. Sonuç olarak toplamda 10 farklı eğitim yapılmıştır.

3.3 Yöntem

Yapılan deneyler YOLOv11 mimarisi kullanılarak gerçekleştirilmiştir. YOLOv11, önceki versiyonlara kıyasla daha verimli katman yapıları (örneğin C3k2) ve gelişmiş dikkat mekanizmaları (örneğin C2PSA) ile donatılmış, literatürde sınırlı sayıda çalışmada ele alınmış güncel bir mimaridir. Bu yüzden çalışma kapsamında amacımız sadece bu mimariyi çalıştırmak değil, belirli parametrelerin bu modeli nasıl etkilediğini görmek ve literatüre katkıda bulunmaktır. Bunun için modelin performansını ve doğruluğunu doğrudan etkileyecek olan resim boyutu (image size) ve katman dondurma (freeze) parametreleri ele alınmıştır. Sonuçlar aynı parametrelerle eğitilmiş YOLO ailesinin eski modellerinden birisi olan YOLOv8 modeline ait sonuçlar ile karşılaştırılmıştır.

Deneyisel çalışmalarda bu parametreler sistematik biçimde değiştirilmiş ve farklı kombinasyonlar altında eğitimler yürütülmüştür. Her bir varyasyon için model, belirli sayıda dönem boyunca eğitilmiş ve ortalama doğruluk, duyarlılık ve kesinlik gibi performans metrikleri karşılaştırmalı olarak değerlendirilmiştir. Böylece, YOLOv11'in hiperparametre duyarlılığı ortaya konmuş ve hangi yapılandırmaların drone tespiti gibi hassas uygulamalarda daha etkili sonuçlar verdiği analiz edilmiştir. Eğitim için resim boyutu 640x640 olarak belirlenmiştir.

3.4 Eğitim Parametreleri

Eğitim sürecinde başlangıç ağırlığı olarak yolov11.pt dosyası kullanılmıştır. Her bir eğitim 200 dönem (epoch) boyunca sürecek şekilde yapılandırılmıştır. Modelin aşırı öğrenmesini (overfitting) ve gereksiz yere uzun süre çalışmasını önlemek amacıyla 'patience' parametresi 20 olarak belirlenmiştir. Bu parametre sayesinde, eğer modelin ortalama doğruluk değerinde 20 dönem boyunca herhangi bir iyileşme gözlemlenmez ya da değerler düşmeye başlarsa, eğitim süreci otomatik olarak durdurulacaktır. Öğrenme oranı 0.0001 olarak seçilmiş, optimizasyon algoritması olarak ise ADAM tercih edilmiştir. Eğitim süresince tüm

görsellerin boyutlarının sabit kalması sağlanmak amacıyla multi_scale parametresi True olarak ayarlanmıştır.

Model performansı, giriş görüntü boyutu ve katman dondurma (freeze) parametresindeki değişimlere göre değerlendirilmiştir. Görüntülerin kalitesi ile eğitim sürecinin maliyeti dikkate alınarak giriş boyutu olarak 640×640 ve 320×320 piksel değerleri seçilmiştir. Katman dondurma işlemi kapsamında, modelin bazı katmanlarının eğitilmesinin devre dışı bırakılmasıyla hem hız hem de doğruluk açısından performans üzerindeki etkiler gözlemlenmiştir. Bu amaçla, her iki görüntü boyutu için freeze parametresi 0, 5, 10, ve 15 olmak üzere dört farklı değerle ayrı ayrı eğitimler gerçekleştirilmiştir. Sonuç olarak toplam 8 farklı model eğitilmiş ve performansları karşılaştırmalı olarak analiz edilmiştir.

3.5 Değerlendirme Metrikleri

Yapılan bu çalışma kapsamında, modelin performansını değerlendirmek amacıyla kullanılan metrikler bu bölümde detaylı olarak açıklanacaktır. Derin öğrenme tabanlı modellerde, eğitilen modelin başarısını objektif şekilde ölçebilmek için doğru ve yanlış tahmin edilen değerlerden hareketle çeşitli denklemler kurulmakta, böylece modelin performansı sayısal verilerle ifade edilebilmektedir. Etiketli veri setleri kullanıldığında, modelin öğrenme sürecinin yanı sıra genelleme yeteneğini de izleyebilmek için veri seti genellikle eğitim (training), doğrulama (validation) ve test olmak üzere üçe ayrılmaktadır. Eğitim sırasında model, sadece eğitim veri setindeki örnekleri kullanarak parametrelerini güncellerken, doğrulama veri seti hiçbir şekilde eğitim için dahil edilmez.

Doğrulama veri seti, eğitim aşamasında her epoch sonunda ya da belirli aralıklarla modelin performansını kontrol etmek için kullanılır. Eğitim süreci devam ederken, model doğrulama veri seti üzerindeki örnekleri kullanarak tahminler üretir ve bu tahminler, veri setinin gerçek etiket değerleriyle karşılaştırılır. Bu aşamada modelin doğruluk, kesinlik, duyarlılık gibi metrik değerleri hesaplanarak izlenir. Böylece modelin sadece eğitim verisinde değil, daha önce görmediği doğrulama verisinde de nasıl davrandığı gözlemlenir; aşırı öğrenme (overfitting) gibi istenmeyen durumlar erken aşamada tespit edilebilir. Doğrulama sırasında elde edilen bu geri bildirimler, hiperparametre ayarlamalarında ve model mimarisinin geliştirilmesinde önemli bir adımdır.

Kullanılan değerlendirme ölçütlerinden hangisinin en anlamlı olduğu ise, geliştirilen modelin özelliklerine, belirlenen görevin doğasına, yanlış sınıflandırmaların ortaya

ıkarabileceėi farklı maliyetlere ve veri setinin dengeli veya dengesiz olup olmadıėına baėlı olarak deėiřkenlik gstermektedir. rneėin, dengesiz veri setlerinde sadece doėruluk oranına odaklanmak yanıltıcı olabilir; bu durumda duyarlılık ve kesinlik gibi ltler ok daha anlamlı bir deėerlendirme sunar. Ayrıca modelin hangi hata trne daha duyarlı olması gerektiėi, uygulamanın hedeflerine gre deėiřebilir; gvenlik odaklı bir sistemde yanlış negatiflerin maliyeti yksekken, bazı ticari uygulamalarda yanlış pozitifler daha byk sorun oluřturabilir. Bu nedenle, kullanılan metriklerin amaca uygun řekilde seilmesi, elde edilen sonuların doėru yorumlanabilmesi iin kritik neme sahiptir.

3.5.1 Karmařıklık matrisi (Confusion Matrix)

Karmařıklık matrisi bir modelin ne kadar iyi tahmin yaptığını gsteren bir tablodur. Bir modelin gerek etiketlerle tahminlerini karřılařtırarak doėru ve hatalı sınıflandırmaların sayısını tablo řeklinde gsteren deėerlendirme aracıdır. Bu tablodaki deėerlere bakılarak diėer deėerlendirme metrikleri hesaplanır.

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FP)
Actual Negative	False Positive (FP)	True Negative (TN)

řekil 3.5: Karmařıklık Matrisi

Eėitim esnasında model, doėrulama veri setindeki her bir rnek iin tahmin gerekleřtirmektedir. Bu tahmin edilen deėerler, veri setinin gerek etiketleri (ground truth)

ile karşılaştırılmakta ve karşılaştırma sonucuna göre ilgili örnek, karmaşıklık matrisinde belirtilen dört etiketten birine atanmaktadır.

Pozitif örnek, hedef sınıfı temsil eden örnektir. Bu çalışmada hedef sınıf drone tespiti olduğu için pozitif örnek, drone sınıfına ait etiketler içeren görüntülerdir. Negatif örnek ise hedef sınıfa ait olmayan, yani drone etiketi taşımayan örneklerdir. True Positive (TP) değeri, pozitif bir örneğin doğru bir şekilde pozitif olarak tahmin edildiği durumları temsil eder. False Positive (FP) değeri, negatif bir örneğin hatalı bir şekilde pozitif tahmin edildiği durumları ifade eder. True Negative (TN) değeri, negatif bir örneğin başarılı bir şekilde negatif olarak sınıflandırıldığı örneklerdir. False Negative (FN) değeri ise pozitif bir örneğin hatalı bir şekilde negatif olarak sınıflandırıldığı durumların sayısını göstermektedir.

3.5.2 Kesinlik (Precision)

Modelin yaptığı pozitif tahminlerin doğruluk oranıdır. Yani modelin doğru olarak tahmin ettiği örneklerin ne kadarının gerçekten doğru olduğunu göstermektedir. Yanlış pozitiflerin en aza indirilmesi gereken durumlarda önemlidir. Bu değer model doğru bir şekilde pozitif tahmin yapıldığında ya da yanlış pozitiflerin sayısı azaldığında artmaktadır.

$$Kesinlik = \frac{TP}{TP + FP} \quad (3.1)$$

Kesinlik değeri, özellikle yanlış pozitif tahminlerin maliyetinin yüksek olduğu çalışmalarda kritik öneme sahiptir. Bu çalışmada, modelin drone içermeyen bir görüntüyü yanlışlıkla drone olarak tanımlaması, gereksiz alarm ve kaynak israfına neden olabileceğinden, yüksek kesinlik elde etmek güvenlik açısından büyük önem taşımaktadır.

3.5.3 Duyarlılık (Recall)

Modelin pozitifleri tahmin etmede ne kadar iyi olduğunu göstermektedir. Pozitif olarak doğru şekilde sınıflandırılan örneklerin sayısının toplam pozitif örnek sayısına oranıdır. Negatif sınıfların nasıl sınıflandırıldığından bağımsız olarak pozitif örneklerin nasıl sınıflandırdığı ile ilgilenmektedir.

$$Duyarlılık = \frac{TP}{TP + FN} \quad (3.2)$$

Duyarlılık değeri, özellikle yanlış negatiflerin maliyetinin yüksek olduğu uygulamalarda önemlidir. Bu çalışmada, bir drone'un tespit edilememesi (yanlış negatif) ciddi güvenlik

açıklarına neden olabileceği için yüksek duyarlılık elde etmek modelin başarısı açısından kritiktir.

3.5.4 F1-Skoru

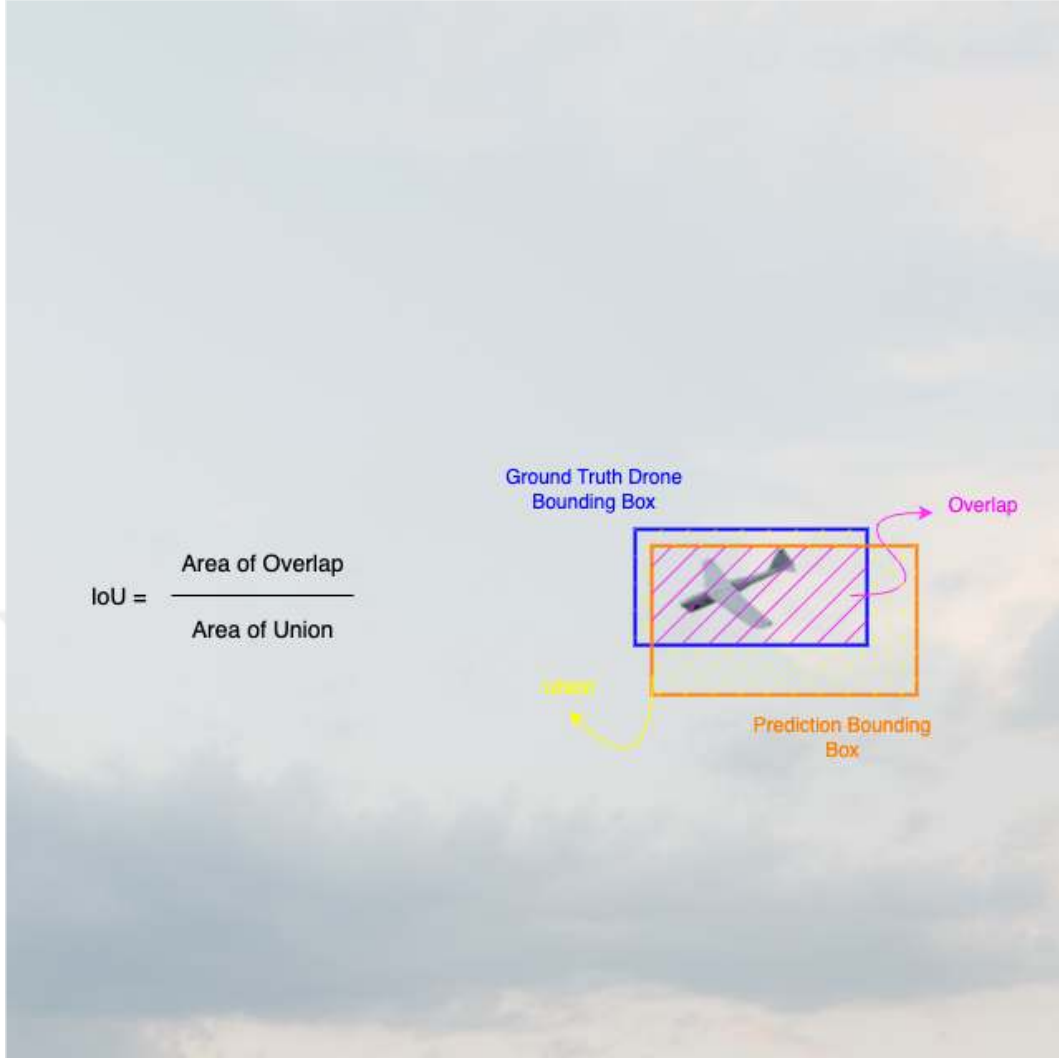
Kesinlik ve duyarlılık değerlerini birlikte dikkate alarak modelin genel performansını özetleyen bir ölçüttür. Kesinlik ve duyarlılığın harmonik ortalaması olarak hesaplanır ve bu iki parametre arasında bir denge sağlar.

$$f1_{score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (3.3)$$

Dengesiz veri kümelerinde model performansı hakkında diğer metriklere göre daha iyi fikir verir. F1 skoru, 0 ile 1 arasında bir değer alır ve 1'e yaklaştıkça modelin pozitif sınıfı doğru ve dengeli şekilde tahmin ettiğini gösterir. Bu çalışmada, hem drone'ları yanlış pozitif tahmin etmemek (kesinlik) hem de mevcut tüm drone'ları doğru tespit edebilmek (duyarlılık) önemli olduğu için F1 skoru, modelin genel başarımını değerlendirmede kritik bir ölçüt olarak değerlendirilmektedir.

3.5.5 IoU (Kesişim birleşme oranı)

Tahmin edilen nesne konumu (Prediction Bounding Bbox) ile gerçek nesne konumunun (Ground Truth Bounding Box) kesişim oranıdır. İki konum ne kadar çakışırsa IoU değeri o kadar yüksek olur. IoU değeri 0 ile 1 arasında değişir; 0 hiç örtüşme olmadığını, 1 tamamen aynı konumda olduklarını gösterir.



Şekil 3.6: IoU hesaplama

Yukarıda görsel olarak verilen şekilde IoU, 2 sınır arasındaki örtüşmeyi ölçer. Bu değere bakarak yapılan tahminin doğruluğu hakkında bilgi edinilir.

3.5.6 Ortalama doğruluk (mAP@0.5 & mAP@0.5:0.9)

Bir modelin verilen görevi ne kadar başarılı şekilde yerine getirdiğini anlamak için kullanılan en yaygın ölçütlerden birisi ortalama doğruluk (mAP) değeridir. Bu ölçüt, modelin doğrulama veya test veri setinde gerçekleştirdiği tahminlerin, gerçek etiketlerle ne kadar örtüştüğünü sayısal olarak ifade eder. Hesaplama sürecinde, model doğrulama/test veri setindeki her nesne için tahminlerde bulunur ve her tahmin sonucunda bir sınırlayıcı kutu (bounding box) ile birlikte bir güven skoru (confidence score) elde edilir. Elde edilen bu tahminler, güven skorlarına göre büyükten küçüğe sıralanır ve modelin en emin olduğu tahminlerden başlayarak doğruluk analizi yapılır.

Sıralanan tahminlerin her biri için, gerçek konumlar (ground truth) ile tahmin edilen kutular arasındaki örtüşme oranı IoU değeri hesaplanır. Belirlenen IoU eşiğine göre (örneğin 0.5) karmaşıklık matrisinde bulunan TP, FP, TN ve FN değerleri güncellenir. Eğer hesaplanan IoU değeri belirlenen eşik olan 0.5'ten büyükse tahmin doğru kabul edilerek TP olarak sınıflandırılır; aksi taktirde FP olarak değerlendirilir. Bu sınıflandırmalar tamamlandıktan sonra her tahmin için kümülatif olarak kesinlik ve duyarlılık değerleri hesaplanır.

Hesaplanan kesinlik ve duyarlılık çiftleri kullanılarak kesinlik-duyarlılık eğrisi oluşturulur. Bu eğrinin altındaki alanın integrali Avarage Precision (AP) değerini verir ve AP, modelin genel başarısını özetler. AP hesaplamak için yaygın olarak iki farklı yöntem kullanılmaktadır. Bunlar PASCAL VOC ve COCO yöntemleridir. PASCAL VOC yönteminde precision değerleri belirli aralıklarda (genellikle 11 noktada) örneklenerek yaklaşık alan hesaplanır. COCO yönteminde ise eğrinin tüm noktalarından hassas integral alınarak daha ayrıntılı bir AP değeri elde edilir.

Eğer birden fazla tahmin edilecek sınıf varsa, her sınıf için AP değeri ayrı ayrı hesaplanır ve bu değerlerin ortalaması alınarak mAP (Mean Avarage Precision) değeri bulunur. Böylece mAP, çok sınıflı problemler için genel performansı tek bir değerle ifade eder.

Öte yandan, modelin yalnızca tek bir IoU eşliğinde değil, farklı hassasiyet seviyelerinde nasıl performans gösterdiğini değerlendirmek için mAP@0.5:0.9 metriği kullanılır. Bu metriğin hesaplanmasında IoU eşikleri 0.5, 0.55, 0.6, ..., 0.9, 0.95 olacak şekilde toplam 10 farklı eşikte AP değerleri hesaplanır; ardından bu 10 AP değerinin ortalaması alınarak mAP@0.5:0.9 elde edilir. Bu değer, modelin hem kaba hem de hassas konum tahminlerinde genel başarısını daha doğru yansıtarak, gerçek dünya uygulamalarında performans değerlendirmesi için kapsamlı bir bakış açısı sunar.

4. EĞİTİM SONUÇLARI VE DEĞERLENDİRME

4.1 Ön Deney Sonuçları

Ön çalışma kapsamında farklı öğrenme oranları ve farklı sayılarda sabitlenen (freeze edilen) katmanlar kullanılarak modelin performansını değerlendirmek amacıyla YOLOv11 modeli üzerinde çeşitli eğitim denemeleri gerçekleştirilmiştir. Tüm deneylerde görüntü boyutu 640 piksel olarak sabit tutulmuş ve modelin drone tespit başarımı; duyarlılık, mAP@0.5 ve mAP@0.5:0.9 metrikleriyle ölçülmüştür. Aşağıdaki tabloda, farklı ayar kombinasyonları sonucunda elde edilen değerlendirme ölçütleri sunulmaktadır. Eğitim esnasında ağırlıklar kaydedilmediğinden diğer metrikler tabloya eklenmemiştir.

Çizelge 4.1: Ön deneylere ait eğitim parametreleri ve sonuçlar

LR	IMG_SIZE	FREEZE	RECALL	mAP@0.5	mAP@0.5:0.9
0.0001	640	0	0.715	0.765	0.332
		5	0.762	0.791	0.342
		10	0.785	0.78	0.332
		15	0.664	0.724	0.305
		20	0.506	0.516	0.214
0.001	640	0	0.545	0.546	0.225
		5	0.672	0.752	0.314
		10	0.705	0.766	0.313
		15	0.71	0.718	0.29
		20	0.552	0.542	0.216

Tablo incelendiğinde düşük öğrenme oranı (0.0001) ile eğitilen modellerin genel olarak daha yüksek duyarlılık ve mAP@0.5 değerlerine ulaştığı görülmektedir. Özellikle FREEZE=5 ve FREEZE=10 seviyelerinde performansın en yüksek seviyeye çıktığı, daha fazla katmanın

sabitlenmesi durumunda ise (örneğin FREEZE=15 ve 20) model başarısının belirgin biçimde düştüğü gözlemlenmiştir. Daha yüksek öğrenme oranı (0.001) kullanıldığında ise başlangıç performansı düşük kalmış, ancak FREEZE değeri artırıldıkça performans nispeten iyileşmiştir; yine de bu ayarlarda elde edilen sonuçlar düşük öğrenme oranlı deneylerin gerisinde kalmıştır. Ayrıca tüm denemelerde mAP@0.5:0.9 değerlerinin mAP@0.5'e göre belirgin şekilde düşük olduğu ve modelin hassas konum tahmininde gelişime açık olduğu anlaşılmaktadır. Bu ön çalışma sonucunda, kullanılan veri setindeki verilerin yetersizliğinden, özellikle kuş görüntülerinin eksikliğinden ötürü doğruluk oranlarının düşük kaldığı gözlemlenmiştir. Bu nedenle çalışma kapsamında kullanılan ana veri seti olan Drone-vs-Bird Grand Challenge kapsamında paylaşılan veri setine ek olarak Kaggle platformunda bulunan verilerden ilave yapılarak ana deneylerimiz gerçekleştirilmiştir.

4.2 Ana Deney Sonuçları

Yapılan ön deneyler sonucunda değerlendirme metriklerinin sonuçlarının düşük kalmasından ötürü verilere ilave yapılarak asıl deneyler gerçekleştirilmiştir. Ayrıca öğrenme oranı düştüğünde yaşanan ani düşüşten dolayı yapılan ana deneylerde öğrenme oranı parametresi 0.0001 değerinde sabit tutulmuştur. Bunun yerine, farklı giriş boyutları ve katman dondurma seviyeleri ile eğitilmesi sonucunda elde edilen doğruluk, kesinlik, duyarlılık ve F1 skoru gibi performans metrikleri karşılaştırmalı olarak analiz edilmiştir. Buna ek olarak YOLOv11 modelinin performansı YOLOv8 modeli ile de karşılaştırılmıştır. Aşağıdaki tabloda, her bir yapılandırmaya ait eğitim sonuçları yer almakta ve modelin genel başarımı üzerinde hangi parametrelerin daha etkili olduğu değerlendirilmektedir.

Çizelge 4.2: Eğitilen modeller ve elde edilen başarı ölçütleri

MODEL	IMG_SIZE	FREEZE	PRECISION	RECALL	F1 SCORE	mAP@0.5	mAP@0.5:0.9
V11	640	0	0.937	0.879	0.907	0.931	0.660
		5	0.930	0.870	0.901	0.929	0.655
		10	0.928	0.865	0.896	0.929	0.652
		15	0.941	0.845	0.891	0.920	0.647
	320	0	0.944	0.877	0.910	0.930	0.660
		5	0.920	0.783	0.846	0.833	0.584

		10	0.926	0.784	0.849	0.842	0.594
		15	0.912	0.755	0.826	0.815	0.573
V8	640	0	0.911	0.903	0.907	0.931	0.649
		5	0.923	0.860	0.890	0.918	0.639
		10	0.925	0.848	0.886	0.912	0.637
		15	0.924	0.836	0.878	0.905	0.634

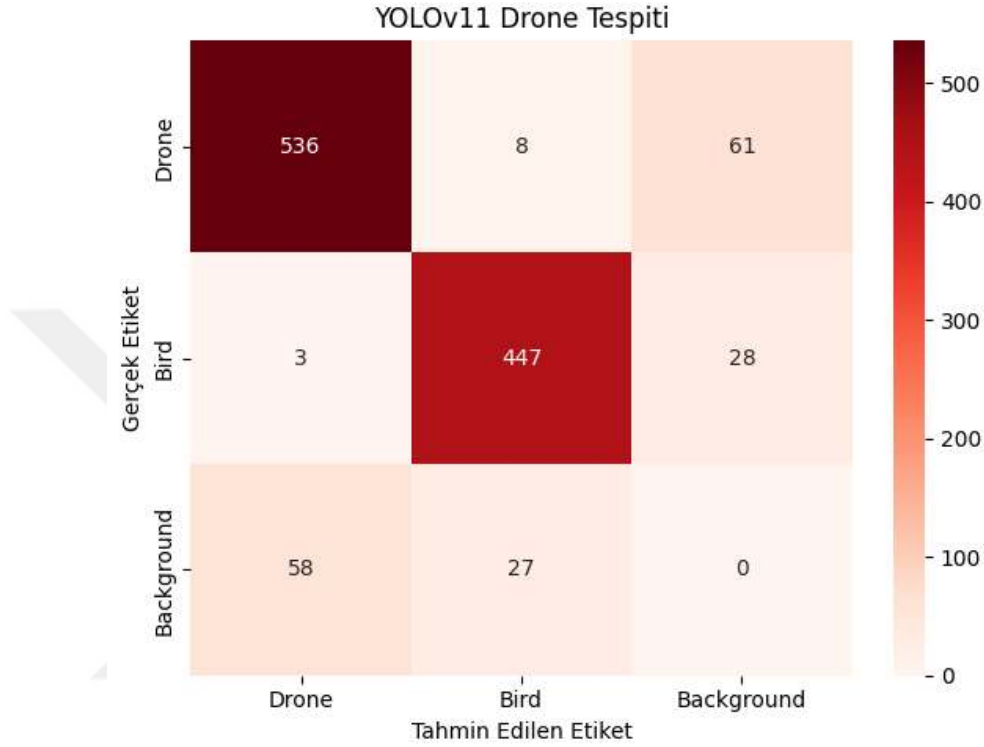
Tablo 4.1’de, farklı görüntü boyutları ve katman dondurma seviyeleriyle eğitilmiş YOLO modellerinin elde ettiği başarı ölçütleri sunulmaktadır. Eğitim sürecinde kullanılan parametre değişikliklerinin model performansı üzerindeki etkileri kesinlik, duyarlılık, F1 skoru ve ortalama doğruluk (mAP) metrikleri üzerinden detaylı bir şekilde değerlendirilmiştir.

Görüntü boyutu 640 piksel olarak belirlendiğinde, YOLOv11 modelinde genel olarak daha yüksek ve istikrarlı başarı değerleri elde edildiği gözlemlenmiştir. Özellikle freeze parametresi 0 olarak ayarlandığında, YOLOv11 modeli %93,7 kesinlik ve %90,7 F1 skoru ile en yüksek başarıyı göstermiştir. Bu durum, modelin tüm katmanlarının eğitime açık olmasının öğrenme kapasitesine olumlu katkı sağladığını ortaya koymaktadır. Öte yandan, YOLOv8 modeli de 640 piksel çözünürlükte güçlü sonuçlar vermiş; ancak FREEZE değeri yükseldikçe performansının düştüğü gözlemlenmektedir.

Görüntü boyutunun 320 piksele düşürülmesiyle birlikte her iki modelde de genel başarı metriklerinde belirgin bir azalma meydana gelmiştir. YOLOv11 modelinde FREEZE=0 konfigürasyonunda bile F1 skorunun 0,910’da kalmasına rağmen, FREEZE arttıkça kademeli bir düşüş yaşanmış ve özellikle FREEZE=15’te F1 skoru 0,826’ya kadar gerilemiştir. Bu durum, düşük çözünürlüklü girişlerin nesne tespit performansını sınırlayabileceğini ve çözünürlüğün drone tespiti gibi detaylı lokalizasyon gerektiren görevlerde kritik bir faktör olduğunu ortaya koymaktadır.

YOLOv11 ve YOLOv8 modelleri kıyaslandığında, YOLOv11’in çoğu parametre ayarında daha yüksek kesinlik, duyarlılık ve mAP değerleri sunduğu görülmektedir. YOLOv8’in bazı ayarlarda benzer sonuçlar üretse de FREEZE değeri arttığında performansının daha hızlı düştüğü ve daha değişken davrandığı anlaşılmaktadır. Bu sonuç, YOLOv11’in bu çalışmada

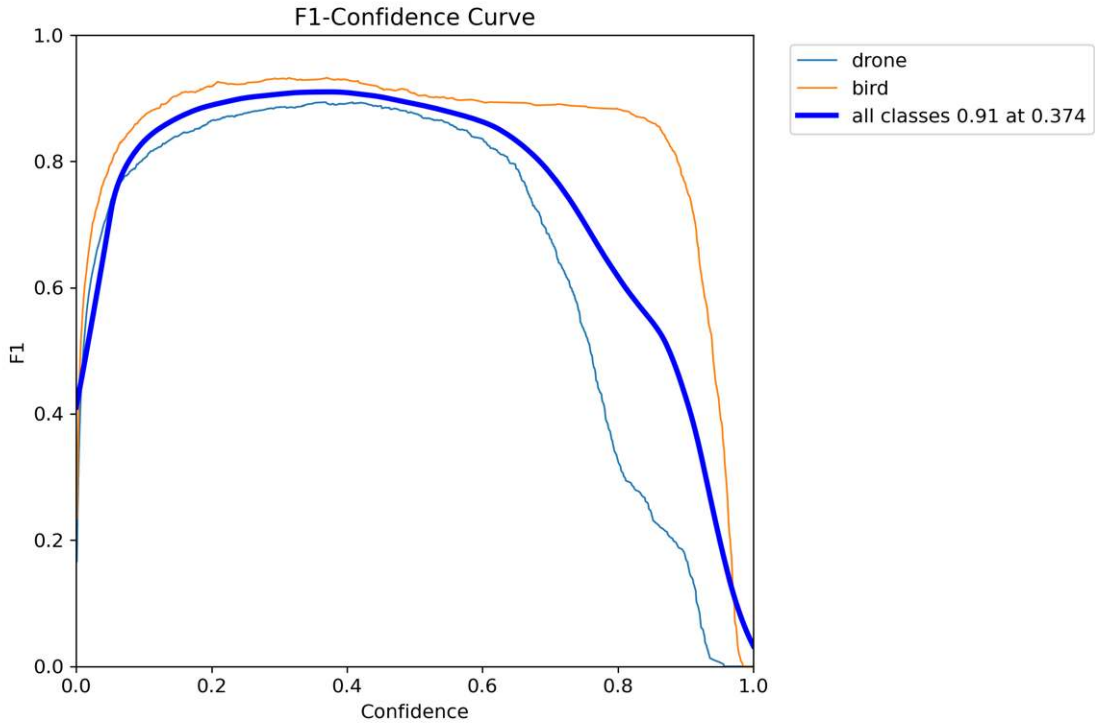
kullanılan drone ve kuş veri setlerinde daha kararlı ve yüksek performanslı bir yapı sunduğunu göstermektedir. Genel olarak, yapılan bu sistematik karşılaştırmalar, hem model seçiminin hem de eğitim parametrelerinin nesne tespiti başarısı üzerinde belirleyici etkileri olduğunu ortaya koyarak, eğitim stratejisinin doğru yapılandırılmasının önemini vurgulamaktadır.



Şekil 4.1: En başarılı modele ait karışıklık matrisi grafiği

Şekil üzerinde gösterilen karışıklık matrisi, YOLOv11 modelinin drone, kuş ve arka plan sınıflarındaki tahmin performansını özetlemektedir. Model, drone ve kuş sınıflarında yüksek doğruluk elde etmiştir. Karışıklık matrisinde arka plan sınıfına ait olan doğru sınıflandırma hücresi, beyaz zemin üzerinde beyaz renkle gösterildiğinden okunabilirliği düşmüş ve bu değer 0 olarak kabul edilmiştir. Özellikle drone sınıfında bazı örneklerin arka plan olarak hatalı etiketlenmesi, modelin bu iki sınıf arasında zaman zaman ayırım yapmakta zorlandığını

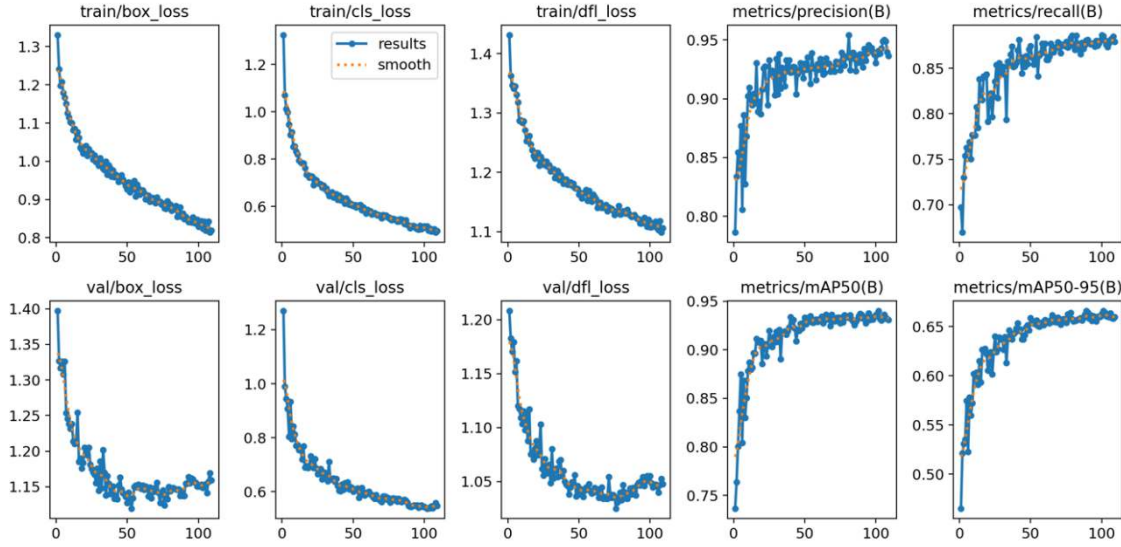
göstermektedir. Buna karşın kuş sınıfında gözlenen düşük hata oranı, modelin bu sınıfa ait nesneleri daha tutarlı ve doğru şekilde tanıyabildiğini ortaya koymaktadır.



Şekil 4.2: En başarılı modele ait F1 Curve grafiği

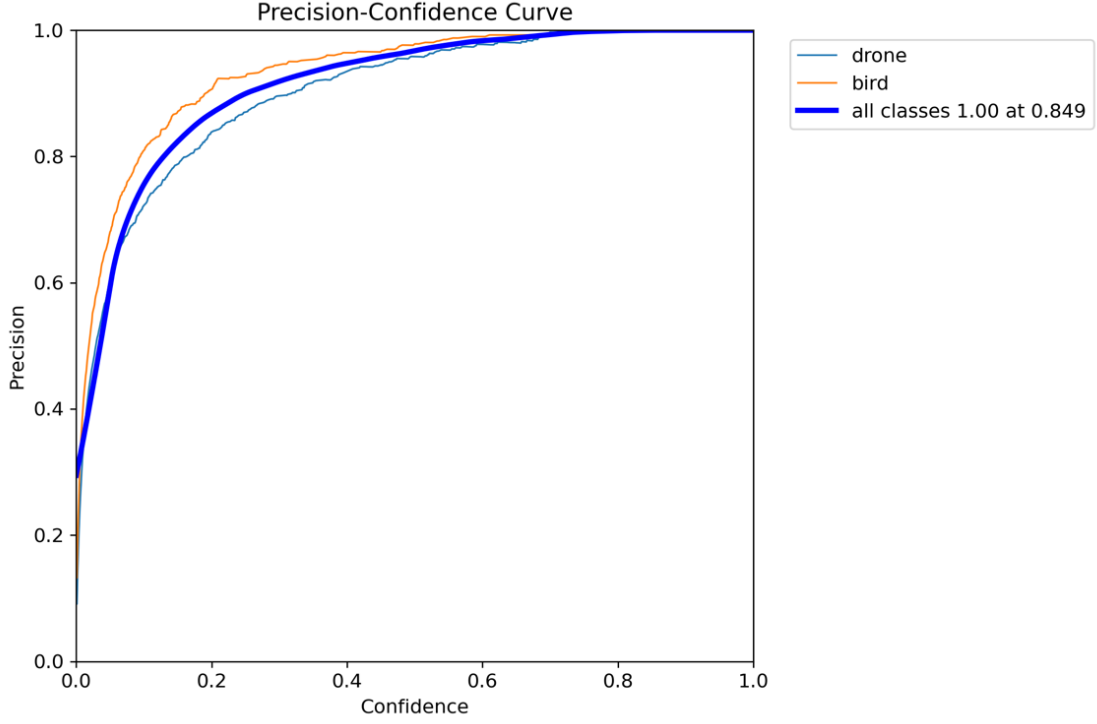
Şekil 4.2’de gösterilmekte olan F1-Confidence eğrisi, modelin farklı güven skorlarına (confidence threshold) göre hesaplanan F1 skorlarını sınıf bazlı ve genel ortalamalarını göstermektedir. Eksenler incelendiğinde, yatay eksenle tahminlerin güven skoru (0.0–1.0 aralığında), dikey eksenle ise karşılık gelen F1 skoru yer almaktadır. Grafikte yer alan mavi ve turuncu eğriler, sırasıyla drone ve bird sınıflarına ait F1 skorlarının güven değerine göre değişimini göstermektedir. Koyu mavi kalın eğri ise tüm sınıflar için hesaplanan ortalama F1 skorunu temsil etmektedir. Eğrinin üzerinde belirtilen “all classes 0.91 at 0.374” ifadesi, tüm sınıflar dikkate alındığında en yüksek F1 skorunun (0.91) %37.4 güven eşliğinde elde edildiğini göstermektedir. Bu, modelin hem yüksek kesinlik hem de yüksek duyarlılık değerlerini sağladığını ortaya koymaktadır. Drone sınıfı için F1 skoru, orta güven değerlerine kadar artış göstermekte ancak %80 üzeri eşiğe gelindiğinde keskin bir düşüş yaşamaktadır. Bu durum, modelin çok yüksek güven skorlarında drone sınıfını daha az tespit ettiğini (yüksek precision, düşük recall) veya bazı örnekleri tamamen kaçırdığını işaret edebilir. Bird sınıfı ise daha istikrarlı bir performans göstermekte ve yüksek confidence değerlerine kadar güçlü F1 skorunu koruyabilmektedir.

Genel olarak, eğrilerden elde edilen bulgular, modelin optimum tespit performansını orta düzey güven eşiklerinde gösterdiğini, ancak çok yüksek eşik değerlerinin tespit oranlarını ciddi biçimde azaltabileceğini göstermektedir. Bu analiz, model çıktılarında kullanılacak ideal güven eşiği değerinin belirlenmesi açısından kritik önem taşımaktadır.



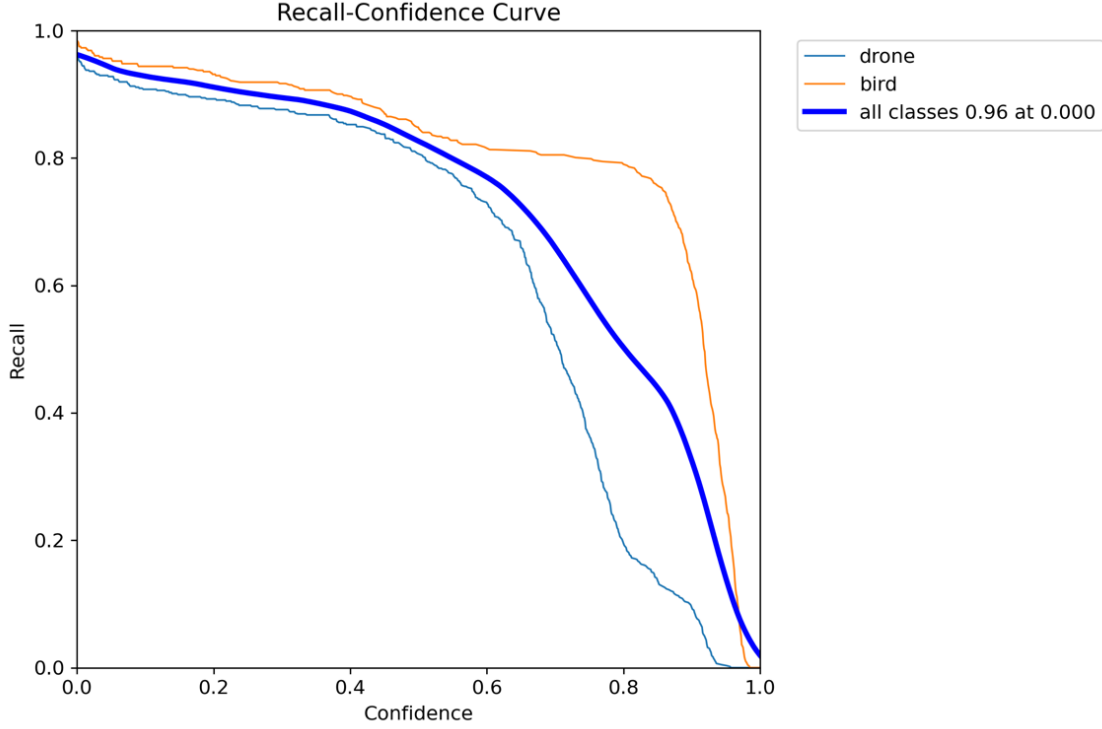
Şekil 4.3: En başarılı modele ait eğitim esnasında gözlemlenen kayıp fonksiyonları ve performans metrikleri

Şekil 4.3'te yer alan grafikler, YOLOv11 modelinin eğitim süreci boyunca izlenen kayıp fonksiyonları ve performans metriklerini göstermektedir. Eğitim sürecinde izlenen train/box_loss, train/cls_loss ve train/dfl_loss grafiklerinde, sırasıyla nesne kutusu, sınıf ve dağılım kayıplarının her dönem sonunda istikrarlı biçimde azaldığı gözlemlenmektedir. Bu düşüş, modelin eğitim verisi üzerinde başarılı bir öğrenme süreci geçirdiğini ortaya koymaktadır. Benzer şekilde, val/box_loss, val/cls_loss ve val/dfl_loss grafiklerinde doğrulama verisi üzerindeki kayıpların da düzenli olarak azaldığı ve modelin aşırı öğrenmeden kaçındığı görülmektedir. metrics/precision(B) ve metrics/recall(B) grafiklerinde, eğitim ilerledikçe modelin doğru pozitif tahminlerinin arttığı, böylece hem kesinlik hem de duyarlılık değerlerinde belirgin bir artış sağlandığı izlenmektedir. Son olarak, metrics/mAP50(B) ve metrics/mAP50-95(B) grafiklerinde ortalama doğruluk oranlarının (mAP) istikrarlı biçimde yükseldiği görülmekte; bu da modelin hem düşük hem yüksek IoU eşiklerinde genel başarımının arttığını göstermektedir. Tüm bu göstergeler, modelin eğitim sürecini dengeli ve verimli bir şekilde tamamladığını ortaya koymaktadır.



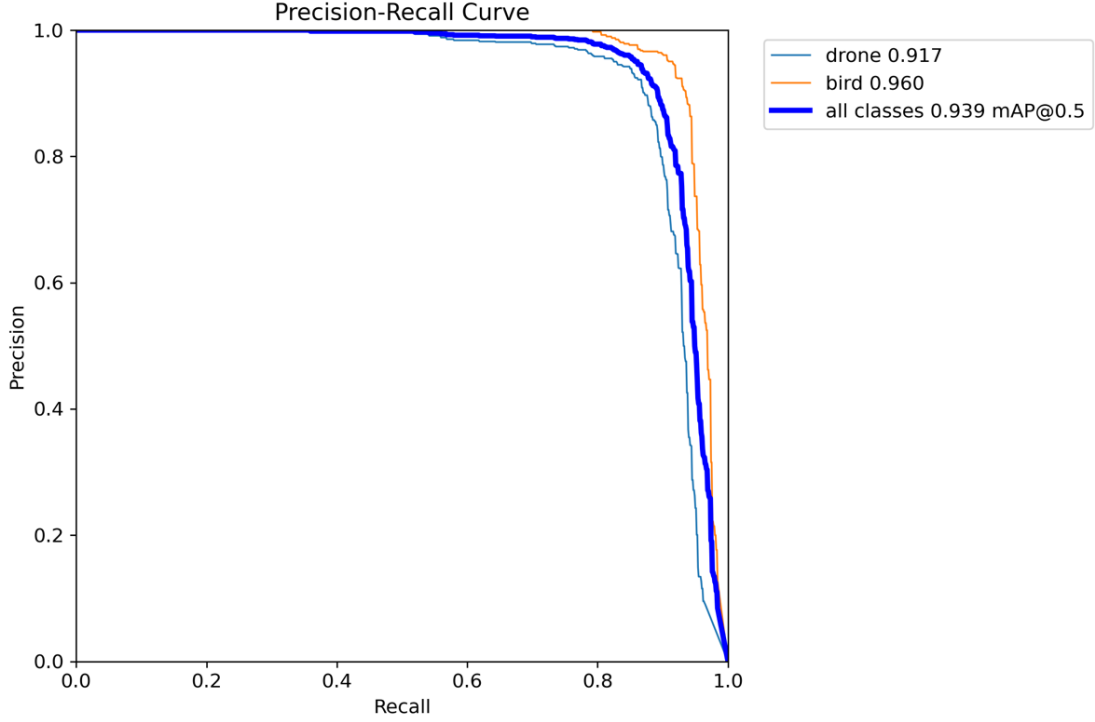
Şekil 4.4: En başarılı modele ait P-Curve grafiği

Şekilde sunulan Precision-Confidence eğrisi, modelin farklı güven eşiklerinde gösterdiği kesinlik performansını sınıf bazında ve genel ortalamada göstermektedir. Eğriye göre, tüm sınıflar için en yüksek precision değeri olan 1.00, yaklaşık 0.849 güven eşğinde elde edilmiştir. Sınıf bazında incelendiğinde, kuş sınıfının daha yüksek kesinlik değerlerine daha düşük güven skorlarında ulaştığı, drone sınıfında ise bu değerin daha kademeli arttığı gözlemlenmektedir. Genel eğilim, modelin güven skoru arttıkça daha az ama daha doğru tahminlerde bulunduğunu, dolayısıyla precision değerinin yükseldiğini göstermektedir.



Şekil 4.5: En başarılı modele ait R-Curve grafiği

Yukarıdaki şekilde sunulan Recall-Confidence eğrisi, modelin farklı güven eşiklerinde elde ettiği değerlerini sınıf bazında ve genel ortalamada göstermektedir. Tüm sınıflar için en yüksek recall değeri olan 0.96, 0.0 güven eşikinde elde edilmiştir. Bu durum, modelin tüm olası nesneleri algılamaya en yatkın olduğu anın düşük güven skorlarında gerçekleştiğini göstermektedir. Grafik genel olarak, güven skoru arttıkça modelin daha az nesne tespit ettiğini, dolayısıyla duyarlılığın azaldığını göstermektedir. Kuş sınıfı, tüm güven aralıklarında drone sınıfına göre daha yüksek duyarlılık değerleri sergilemektedir. Bu da modelin kuş nesnelerini daha kolay ve istikrarlı şekilde tanıyabildiğini göstermektedir.



Şekil 4.6: En başarılı modele ait PR-Curve grafiği

Şekilde gösterilen Precision-Recall (PR) eğrisi, YOLOv11 modelinin tüm sınıflar ve her bir sınıf özelinde kesinlik ile duyarlılık arasındaki ilişkiyi sunmaktadır. Eğride yer alan değerler, kuş sınıfı için 0.960, drone sınıfı için ise 0.917 mAP@0.5 skorlarını göstermektedir. Tüm sınıfların ortalaması olarak hesaplanan mAP@0.5 değeri ise 0.939 olarak elde edilmiştir. Bu yüksek değerler, modelin hem doğru tespit oranlarının hem de kapsamlı tanıma kapasitesinin güçlü olduğunu ortaya koymaktadır. Eğriler genel olarak yüksek precision ve recall değerlerini koruyarak üst köşeye yakın seyretmekte, bu da modelin dengeli bir performans sergilediğini göstermektedir. Özellikle kuş sınıfının eğrisi daha dik ve daha uzun süre üst seviyede ilerlemekte olup, modelin bu sınıfı daha güvenilir biçimde tanıdığı anlaşılmaktadır.

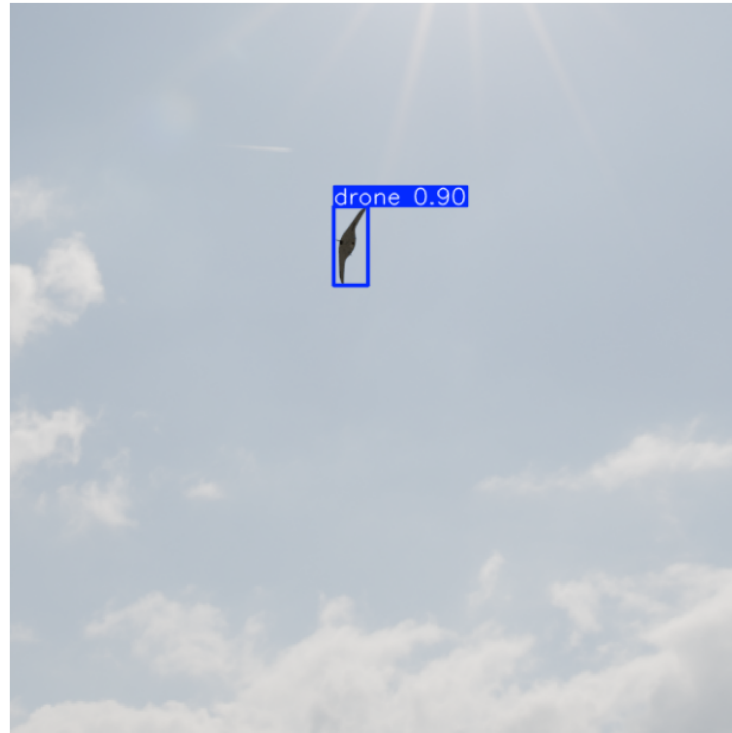
5. SONUÇ VE ÖNERİLER

Bu çalışmada, güncel YOLOv11 mimarisi temel alınarak drone ve kuş nesnelerinin gerçek zamanlı olarak sınıflandırılmasına yönelik bir model tasarlanmış ve bu modelin başarımı farklı parametre kombinasyonları altında detaylı biçimde analiz edilmiştir.

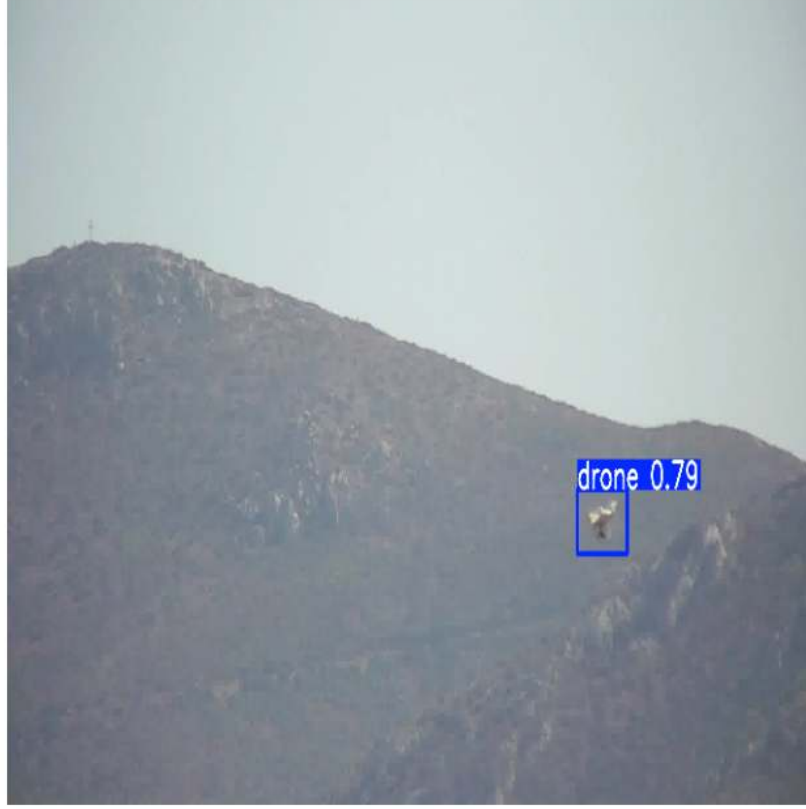




Şekil 5.1: En iyi model ağırlıkları ile tespit edilen drone görüntüsü 1



Şekil 5.2: En iyi model ağırlıkları ile tespit edilen drone görüntüsü 2



Şekil 5.3: En iyi model ağırlıkları ile tespit edilen drone görüntüsü 3



Şekil 5.4: En iyi model ağırlıkları ile tespit edilen kuş görüntüsü 1



Şekil 5.5: En iyi model ağırlıkları ile tespit edilen kuş görüntüsü 2

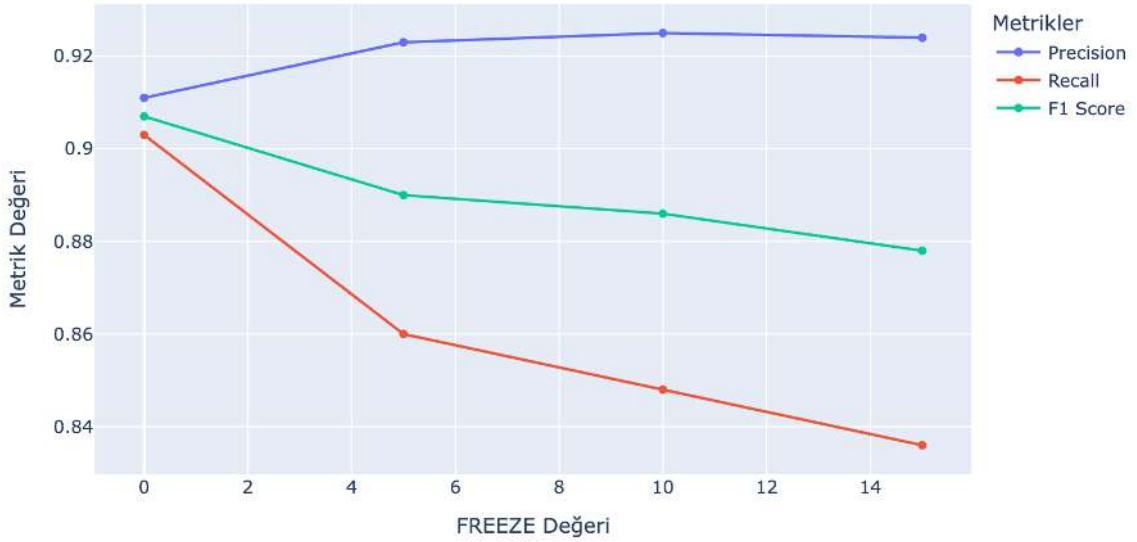


Şekil 5.6: En iyi model ağırlıkları ile tespit edilen kuş görüntüsü 3

Gerçekleştirilen deneylerde, özellikle giriş görüntü boyutunun yüksek tutulmasının (640x640) ve freeze parametresinin sıfıra yakın ayarlanmasının, modelin kesinlik, duyarlılık ve F1 skoru üzerinde pozitif etki yarattığı gözlemlenmiştir.



Şekil 5.7: YOLOv11 modelinin 640 ve 320 piksel çözünürlüklerde FREEZE parametresine göre Precision, Recall ve F1 skorlarının değişimi



Şekil 5.8: YOLOv8 modelinin 640 piksel çözünürlükte FREEZE parametresine göre Precision, Recall ve F1 skorlarının değişimi

Grafikler incelendiğinde, YOLOv8 modelinde FREEZE değeri arttıkça özellikle duyarlılık metriğinde belirgin bir düşüş yaşandığı, kesinlik ve F1 skorunun ise FREEZE=5'ten sonra

hafif bir gerileme trendine girdiği gözlemlenmektedir. Bu durum, YOLOv8'in katman dondurma seviyesine karşı daha hassas olduğunu ve fazla katman dondurmanın modeli olumsuz etkileyerek hatalı tespitleri artırabileceğini göstermektedir. YOLOv11 için hazırlanan grafikte ise hem 640 piksel hem de 320 piksel çözünürlükteki performans değerleri sunulmuştur. 640 pikselde kesinlik ve F1 skorları genel olarak yüksek ve stabil seyrederken, duyarlılık FREEZE arttıkça kademeli olarak düşüş göstermektedir. 320 piksel çözünürlükte ise tüm metriklerde FREEZE değeri arttıkça daha keskin bir performans kaybı gözlenmektedir; özellikle duyarlılık değerinde ciddi düşüşler dikkat çekmektedir. Bu sonuçlar, YOLOv11 modelinin daha yüksek çözünürlükte ve düşük FREEZE ayarlarında daha tutarlı bir performans sergilediğini, çözünürlük düştüğünde ve katmanlar fazlaca dondurulduğunda ise performansın belirgin biçimde azaldığını ortaya koymaktadır. Genel olarak, her iki modelde de FREEZE parametresi arttıkça duyarlılığın en çok etkilenen metrik olduğu, kesinlik ve F1 skorunda ise görece daha az ancak anlamlı azalmalar yaşandığı anlaşılmaktadır. YOLOv8 modeli ile yapılan karşılaştırmalarda, YOLOv11 modelinin genel olarak daha başarılı sonuçlar ürettiği, özellikle karmaşık sahnelerde küçük nesneleri tespit etme kabiliyetinin daha gelişmiş olduğu ortaya konmuştur.

Eğitim süreci boyunca kaydedilen metrik grafiklerinde aşırı öğrenme bulgusuna rastlanmamış, doğrulama ve eğitim hataları arasında dengesizlik oluşmamıştır. Ayrıca Precision-Recall eğrileri, modelin yüksek güven eşiği altında bile istikrarlı performans sergileyebildiğini göstermiştir. Bu çalışma, hem güvenlik alanında kullanılabilecek bir drone tespit sistemi için uygulanabilir bir temel sunmakta hem de YOLOv11 mimarisine dair hiperparametre duyarlılığını ortaya koyarak literatüre katkı sağlamaktadır. Gelecek çalışmalar kapsamında farklı dikkat mekanizmaları veya veri artırma teknikleriyle sistem performansının daha da ileri taşınabileceği öngörülmektedir.

Drone-vs-Bird Grand Challenge veri setini kullanan önceki çalışmalar genellikle klasik makine öğrenmesi (SVM, Rastgele Orman) ya da geleneksel derin öğrenme mimarileri (YOLOv2, YOLOv3-SPP, Faster R-CNN, SSD gibi) ile sınırlı kalmış ve çoğunlukla düşük çözünürlüklü, küçük hedef tespiti gibi zorlu senaryolarda başarı düşüşü yaşamıştır. Bu tezde ise, güncellenmiş YOLOv11 mimarisi kullanılarak hiperparametre optimizasyonu, farklı çözünürlük analizleri ve model dondurma (freeze) stratejileri ile kapsamlı bir deneysel süreç yürütülmüş ve önceki çalışmalara kıyasla daha dengeli bir doğruluk-hız performansı elde edilmiştir. Özellikle küçük ve hızlı hareket eden drone'ların, karmaşık arka planlı sahnelerde dahi yüksek kesinlik ve duyarlılık ile tespit edilebildiği

görülmüştür. Bu sonuçlar, literatürdeki yaklaşımların çoğunun sınırlı olarak ele aldığı gerçek zamanlılık ve farklı çözünürlük senaryolarında da tatmin edici bir performans elde edilebildiğini göstermektedir. Ayrıca, Grand Challenge veri seti üzerinde güncel bir mimari olan YOLOv11 ile yürütülen bu çalışma, mevcut literatürdeki yöntemlere kıyasla dengeli bir doğruluk-hız başarımı ortaya koyarak önemli bir alternatif yaklaşım sunmaktadır.



KAYNAKLAR

- Aker, C., & Kalkan, S. (2017). *Using Deep Networks for Drone Detection*. IEEE.
- Alif, M. A. R. (2024). *YOLOv11 for Vehicle Detection: Advancements, Performance, and Applications in Intelligent Transportation Systems*. <http://arxiv.org/abs/2410.22898>
- Alkhamash, E. H. (2025). A Comparative Analysis of YOLOv9, YOLOv10, YOLOv11 for Smoke and Fire Detection. *Fire*, 8(1). <https://doi.org/10.3390/fire8010026>
- Atik, I. (2023). CB-YOLOv5: Enhancing drone detection with BottleneckCSP and cross convolution for improved performance. *Journal of Radiation Research and Applied Sciences*, 16(4), 100705. <https://doi.org/10.1016/j.jrras.2023.100705>
- Bai, J., Zhu, W., Nie, Z., Yang, X., Xu, Q., & Li, D. (2025). HFC-YOLO11: A Lightweight Model for the Accurate Recognition of Tiny Remote Sensing Targets. *Computers*, 14(5). <https://doi.org/10.3390/computers14050195>
- Bakirci, M., Dmytrovych, P., Bayraktar, I., & Anatoliyovych, O. (2024). Multi-Class Vehicle Detection and Classification with YOLO11 on UAV-Captured Aerial Imagery. *2024 IEEE 7th International Conference on Actual Problems of Unmanned Aerial Vehicles Development, APUAVD 2024 - Proceedings*, 191–196. <https://doi.org/10.1109/APUAVD64488.2024.10765862>
- Bhagwat, A., Dutta, S., Saha, D., & Reddy, M. J. B. (2025). An online 11 kv distribution system insulator defect detection approach with modified YOLOv11 and mobileNetV3. *Scientific Reports*, 15(1). <https://doi.org/10.1038/s41598-025-99756-5>
- Cai, H., Xie, Y., Xu, J., & Xiong, Z. (2022). A Lightweight and Accurate UAV Detection Method Based on YOLOv4. *Sensors*, 22(18). <https://doi.org/10.3390/s22186874>
- Coluccia, A., Fascista, A., Schumann, A., Sommer, L., Dimou, A., Zarpalas, D., Méndez, M., de la Iglesia, D., González, I., Mercier, J. P., Gagné, G., Mitra, A., & Rajashekar, S. (2021). Drone vs. Bird detection: Deep learning algorithms and results from a grand challenge. *Sensors*, 21(8). <https://doi.org/10.3390/s21082824>
- Coluccia, A., Fascista, A., Sommer, L., Schumann, A., Dimou, A., & Zarpalas, D. (2024). The Drone-vs-Bird Detection Grand Challenge at ICASSP 2023: A Review of Methods and Results. *IEEE Open Journal of Signal Processing*, 5, 766–779. <https://doi.org/10.1109/OJSP.2024.3379073>
- Dewangan, B., & Srinivas, M. (n.d.). *LIGHT-YOLOv11: An Efficient Small Object Detection Model for UAV Images*. <https://doi.org/10.1109/CSNT.2025.92>
- Dewangan, V., Saxena, A., Thakur, R., & Tripathi, S. (2023). Application of Image Processing Techniques for UAV Detection Using Deep Learning and Distance-Wise Analysis. *Drones*, 7(3). <https://doi.org/10.3390/drones7030174>
- Dumoulin, V., & Visin, F. (2016). *A guide to convolution arithmetic for deep learning*. <http://arxiv.org/abs/1603.07285>
- Fan, Y., Li, O., & Liu, G. (2022). An Object Detection Algorithm for Rotary-Wing UAV Based on AWin Transformer. *IEEE Access*, 10, 13139–13150. <https://doi.org/10.1109/ACCESS.2022.3147264>
- Gao, Y., Xin, Y., Yang, H., & Wang, Y. (2025). A Lightweight Anti-Unmanned Aerial Vehicle Detection Method Based on Improved YOLOv11. *Drones*, 9(1). <https://doi.org/10.3390/drones9010011>
- Gholamalinezhad, H., & Khosravi, H. (2020). *Pooling Methods in Deep Neural Networks, a Review*.
- Gong, X., Yu, J., Zhang, H., & Dong, X. (2025). AED-YOLO11: A small object detection model based on YOLO11. *Digital Signal Processing: A Review Journal*, 166. <https://doi.org/10.1016/j.dsp.2025.105411>

- He, L., Zhou, Y., Liu, L., Cao, W., & Ma, J. (2025). Research on object detection and recognition in remote sensing images based on YOLOv11. *Scientific Reports*, 15(1), 14032. <https://doi.org/10.1038/s41598-025-96314-x>
- He, L., Zhou, Y., Liu, L., Zhang, Y., & Ma, J. (2025). Application of the YOLOv11-seg algorithm for AI-based landslide detection and recognition. *Scientific Reports*, 15(1). <https://doi.org/10.1038/s41598-025-95959-y>
- Huang, J., Wang, K., Hou, Y., & Wang, J. (2025). LW-YOLO11: A Lightweight Arbitrary-Oriented Ship Detection Method Based on Improved YOLO11. *Sensors*, 25(1). <https://doi.org/10.3390/s25010065>
- Javan, F. D., Samadzadegan, F., Gholamshahi, M., & Mahini, F. A. (2022). A New, Tentative, Modified YOLOv4 Deep Learning Network for Vision-Based UAV Recognition. *Drones*, 6(7). <https://doi.org/10.3390/drones6070160>
- Jegham, N., Koh, C. Y., Abdelatti, M., & Hendawi, A. (n.d.). *YOLO Evolution: A Comprehensive Benchmark and Architectural Review of YOLOv12, YOLO11, and Their Previous Versions*.
- Jegham, N., Koh, C. Y., Abdelatti, M., & Hendawi, A. (2024). *YOLO Evolution: A Comprehensive Benchmark and Architectural Review of YOLOv12, YOLO11, and Their Previous Versions*. <http://arxiv.org/abs/2411.00201>
- Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2021). A Review of Yolo Algorithm Developments. *Procedia Computer Science*, 199, 1066–1073. <https://doi.org/10.1016/j.procs.2022.01.135>
- Kasamsumran, N., Ittichaiwong, P., Chinudomporn, C., Veerakanjana, K., Karopongse, E., & Pora, W. (2025). AI-Assisted Web Application for Leukocyte Abnormality Counting with YOLOv11 and Smartphone Microscopy. *IEEE Access*. <https://doi.org/10.1109/ACCESS.2025.3569767>
- Kassab, M., Zitar, R. A., Barbaresco, F., & Seghrouchni, A. E. F. (2024). Drone Detection With Improved Precision in Traditional Machine Learning and Less Complexity in Single-Shot Detectors. *IEEE Transactions on Aerospace and Electronic Systems*, 60(4), 3847–3859. <https://doi.org/10.1109/TAES.2024.3368991>
- Khanam, R., & Hussain, M. (2024). *YOLOV11: AN OVERVIEW OF THE KEY ARCHITECTURAL ENHANCEMENTS*.
- KILIÇARSLAN, S., ADEM, K., & ÇELİK, M. (2021). An overview of the activation functions used in deep learning algorithms. *Journal of New Results in Science*, 10(3), 75–88. <https://doi.org/10.54187/jnrs.1011739>
- Kufel, J., Bargiel-Łączek, K., Kocot, S., Koźlik, M., Bartnikowska, W., Janik, M., Czogalik, Ł., Dudek, P., Magiera, M., Lis, A., Paszkiewicz, I., Nawrat, Z., Cebula, M., & Gruszczyńska, K. (2023). What Is Machine Learning, Artificial Neural Networks and Deep Learning?—Examples of Practical Applications in Medicine. In *Diagnostics* (Vol. 13, Issue 15). Multidisciplinary Digital Publishing Institute (MDPI). <https://doi.org/10.3390/diagnostics13152582>
- Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. In *Nature* (Vol. 521, Issue 7553, pp. 436–444). Nature Publishing Group. <https://doi.org/10.1038/nature14539>
- Lu, L., He, D., Liu, C., & Deng, Z. (2025). *MASF-YOLO: An Improved YOLOv11 Network for Small Object Detection on Drone View*.
- Mahala, M., Pattanaik, M., & Pandey, G. (2025). Advancing Real-Time Object Detection for Autonomous Vehicles with YOLOv11. *2025 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI)*, 1–6. <https://doi.org/10.1109/IATMSI64286.2025.10984727>

- Mahdavi, F., & Rajabi, R. (2020, December 23).** Drone Detection Using Convolutional Neural Networks. *6th Iranian Conference on Signal Processing and Intelligent Systems, ICSPIS 2020*. <https://doi.org/10.1109/ICSPIS51611.2020.9349620>
- Mishra, D., Buyya, R., & Mohapatra, P. (2018).** *Smart Innovation, Systems and Technologies 153 Intelligent and Cloud Computing*. <http://www.springer.com/series/8767>
- Nahar, A., Islam, M., Shuvo, M. H., Hasan, M., Parvin, S., & Nur, K. (2025).** *A Deep Learning-Based Framework for Accurate Detection and Classification of On-Road Vehicles Using Improved YOLOv11*. <https://doi.org/10.1109/ECCE64574.2025.11013964>
- Nwankpa, C., Ijomah, W., Gachagan, A., & Marshall, S. (2018).** *Activation Functions: Comparison of trends in Practice and Research for Deep Learning*. <http://arxiv.org/abs/1811.03378>
- Rasheed, A. F., & Zarkoosh, M. (2024).** *YOLOv11 Optimization for Efficient Resource Utilization*. <http://arxiv.org/abs/2412.14790>
- Ren, G., Wu, J., & Wang, W. (2025).** Research on UAV Target Detection Based on Improved YOLOv11. *Journal of Computer and Communications*, 13(03), 74–85. <https://doi.org/10.4236/jcc.2025.133006>
- Seidaliyeva, U., Akhmetov, D., Ilipbayeva, L., & Matson, E. T. (2020).** Real-time and accurate drone detection in a video with a static background. *Sensors (Switzerland)*, 20(14), 1–18. <https://doi.org/10.3390/s20143856>
- Sethu Selvi, S., Pavithraa, S., Dharini, R., & Chaitra, E. (2022).** A Deep Learning Approach to Classify Drones and Birds. *MysuruCon 2022 - 2022 IEEE 2nd Mysore Sub Section International Conference*. <https://doi.org/10.1109/MysuruCon55714.2022.9972589>
- Shinde, P. P., & Shah, S. (2018).** *A Review of Machine Learning and Deep Learning Applications*. IEEE.
- Singha, S., & Aydin, B. (2021).** Automated drone detection using YOLOv4. *Drones*, 5(3). <https://doi.org/10.3390/drones5030095>
- Srirgarom, S., Chew, K. H., Lee, D. M. Da, & Ratsamee, P. (2020).** *Drone versus Bird Flights: Classification by Trajectories Characterization*. IEEE.
- Tang, D., Mao, J., & Wang, Y. (2025).** Research on Traffic Target Detection Optimization Based on YOLOv11. *2025 8th International Conference on Advanced Algorithms and Control Engineering (ICAACE)*, 1056–1061. <https://doi.org/10.1109/ICAACE65325.2025.11020430>
- Tripathi, A., Gupta, M. K., Srivastava, C., Dixit, P., & Pandey, S. K. (2022).** Object Detection using YOLO: A Survey. *Proceedings of 5th International Conference on Contemporary Computing and Informatics, IC3I 2022*, 747–752. <https://doi.org/10.1109/IC3I56241.2022.10073281>
- Tsiamis, N., Efthymiou, L., & Tsagarakis, K. P. (2019).** A comparative analysis of the legislation evolution for drone use in oecd countries. *Drones*, 3(4), 1–15. <https://doi.org/10.3390/drones3040075>
- Wang, C., Song, X., Wang, J., & Yan, X. (2025).** An improved YOLOv11 algorithm for small object detection in UAV images. *Signal, Image and Video Processing*, 19(6). <https://doi.org/10.1007/s11760-025-04157-w>
- Xu, Y. (2023).** A Drone Detection Algorithm Based on Color-Edge Joint Information and Convolutional Neural Network. *2023 4th International Conference on Big Data and Artificial Intelligence and Software Engineering, ICBASE 2023*, 389–394. <https://doi.org/10.1109/ICBASE59196.2023.10303263>

- Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018).** Convolutional neural networks: an overview and application in radiology. In *Insights into Imaging* (Vol. 9, Issue 4, pp. 611–629). Springer Verlag. <https://doi.org/10.1007/s13244-018-0639-9>
- Yasmine, G., Maha, G., & Hicham, M. (2023).** Anti-drone systems: An attention based improved YOLOv7 model for a real-time detection and identification of multi-airborne target. *Intelligent Systems with Applications*, 20. <https://doi.org/10.1016/j.iswa.2023.200296>
- Zafar, A., Aamir, M., Mohd Nawi, N., Arshad, A., Riaz, S., Alruban, A., Dutta, A. K., & Almotairi, S. (2022).** A Comparison of Pooling Methods for Convolutional Neural Networks. In *Applied Sciences (Switzerland)* (Vol. 12, Issue 17). MDPI. <https://doi.org/10.3390/app12178643>
- Zhou, S., Yang, L., Liu, H., Zhou, C., Liu, J., Zhao, S., & Wang, K. (2025).** A Lightweight Drone Detection Method Integrated into a Linear Attention Mechanism Based on Improved YOLOv11. *Remote Sensing*, 17(4). <https://doi.org/10.3390/rs17040705>



ÖZGEÇMİŞ

Ad-Soyad : Büşra ÖZCAN

ÖĞRENİM DURUMU:

- **Lisans** : 2021, Cumhuriyet Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği
- **Yüksek Lisans** : İnönü Üniversitesi, Bilgisayar Mühendisliği Ana bilim Dalı

MESLEKİ DENEYİM:

- 2024 - . İnönü Üniversitesi
- 2022 – 2023 Hakan Bilgi Teknolojileri