

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ



HAVALİMANI PİSTLERİNDE YABANCI MADDE TESPİTİ İÇİN
DERİN ÖĞRENME YAKLAŞIMI

Necip Şahamettin KÜÇÜK

Yüksek Lisans Tezi

HAVACILIK BİLİMİ VE TEKNOLOJİLERİ ANABİLİM DALI

HAZİRAN 2025

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Havacılık Bilimi ve Teknolojileri Anabilim Dalı

Yüksek Lisans Tezi

HAVALİMANI PİSTLERİNDE YABANCI MADDE TESPİTİ İÇİN
DERİN ÖĞRENME YAKLAŞIMI

Tez Yazarı

Necip Şahamettin KÜÇÜK

Danışman

Dr. Öğr. Üyesi Ömer Osman DURSUN

HAZİRAN 2025

ELAZIĞ

T.C.
FIRAT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Havacılık Bilimi ve Teknolojileri Anabilim Dalı

Yüksek Lisans Tezi

Başlığı: Havalimanı Pistlerinde Yabancı Madde Tespiti için Derin Öğrenme Yaklaşımı

Yazarı: Necip Şahamettin KÜÇÜK

İlk Teslim Tarihi: 10.06.2025

Savunma Tarihi: 27.06.2025

TEZ ONAYI

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına göre hazırlanan bu tez aşağıda imzaları bulunan jüri üyeleri tarafından değerlendirilmiş ve akademik dinleyicilere açık yapılan savunma sonucunda OYBİRLİĞİ ile kabul edilmiştir.

Danışman:	Dr. Öğr. Üyesi Ömer Osman DURSUN Fırat Üniversitesi, Sivil Havacılık Yüksekokulu	<i>İmza</i> Onayladım
Başkan:	Doç. Dr. Hakan AYGÜN Fırat Üniversitesi, Sivil Havacılık Yüksekokulu	Onayladım
Üye:	Dr. Öğr. Üyesi Fatih TOPALOĞLU Malatya Turgut Özal Üniversitesi, Mühendislik Fakültesi	Onayladım

Bu tez, Enstitü Yönetim Kurulunun/...../20..... tarihli toplantısında tescillenmiştir.

İmza

Prof. Dr. Burhan ERGEN
Enstitü Müdürü

BEYAN

Fırat Üniversitesi Fen Bilimleri Enstitüsü tez yazım kurallarına uygun olarak hazırladığım “Havalimanı Pistlerinde Yabancı Madde Tespiti için Derin Öğrenme Yaklaşımı ” Başlıklı Yüksek Lisans Tezimin içindeki bütün bilgilerin doğru olduğunu, bilgilerin üretilmesi ve sunulmasında bilimsel etik kurallarına uygun davrandığımı, kullandığım bütün kaynakları atıf yaparak belirttiğimi, maddi ve manevi desteği olan tüm kurum/kuruluş ve kişileri belirttiğimi, burada sunduğum veri ve bilgileri unvan almak amacıyla daha önce hiçbir şekilde kullanmadığımı beyan ederim.

27.06.2025

Necip Şahamettin KÜÇÜK



ÖNSÖZ

Havalimanlarındaki yabancı madde hasarı (YMH), uçak motorlarına ve havalimanı personeline ciddi zararlar verebilen bir tehdit faktörüdür. Havalimanı pistlerindeki yabancı maddeleri doğru bir şekilde tespit etmek, uçakların güvenli bir şekilde kalkış ve iniş yapmasını sağlamak için önemlidir. Bu amaçla, çeşitli geleneksel ve otomatik yöntemler kullanılarak pistler düzenli olarak denetlenmektedir. Ancak, mevcut yöntemlerin yabancı maddeleri etkili bir şekilde tespit etmede bazı zorlukları vardır. Örneğin, radar ve LiDAR gibi sensör sistemleri küçük ölçekli YMH'leri zayıf sinyalleri nedeniyle algılayamamakta veya sabit kamera sistemleri uzaktaki YMH'leri doğru bir şekilde tanımlayamamaktadır. Ayrıca, bu sistemlerin maliyeti oldukça yüksektir. Bu nedenle, havalimanlarındaki yabancı madde tespitinde daha uygun maliyetli ve daha yüksek performanslı bir yöntem ihtiyacı duyulmaktadır.

Bu çalışmada, havalimanlarındaki yabancı madde tespiti için YOLOv4 (You Only Look Once) tabanlı bir algılama algoritması sunulmaktadır. Bu yöntem, CSPDarknet53 adlı bir omurga ağı kullanarak özellik çıkarma ve SPP (Spatial Pyramid Pooling) ve PAN (Path Aggregation Network) gibi modüller kullanarak çoklu ölçekli özellik füzyonu yapmaktadır. Ayrıca, Mish adlı yeni bir aktivasyon fonksiyonu ve CIoU (Complete Intersection over Union) adlı yeni bir kayıp fonksiyonu kullanarak küçük ölçekli YMH'leri tespit etmektedir. Yabancı madde örnek veri setleri oluşturularak önerilen yöntem doğrulanmıştır. Deneyler, YOLOv4 tabanlı algılama algoritmasının yabancı maddeleri etkili bir şekilde tespit ettiğini ve hem doğruluk hem de hız açısından YOLOv3'ten daha üstün performans gösterdiğini göstermiştir.

Bu çalışmanın katkısı, havalimanlarındaki yabancı madde tespiti için yeni bir veri seti oluşturmak ve bu veri seti üzerinde yaygın olarak kullanılan makine öğrenmesi modelleri için nesne algılama performansını değerlendirmektir. Ayrıca, bu çalışma, havalimanlarındaki yabancı madde tespiti için bilgisayar görüşü ve makine öğrenmesi alanında ilerlemeye katkıda bulunmayı amaçlamaktadır.

Seminer çalışmamın gerçekleşmesi sürecinde desteklerinden dolayı danışman hocam Sayın Dr. Öğr. Üyesi Ömer Osman DURSUN'a teşekkür ederim.

Ek olarak, seminer çalışması süresince her zaman bana destek olan ve yanımda olan aileme çok teşekkürlerimi sunarım. Hayatta başarılar dilerim.

Bu tez çalışması, Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK) tarafından TÜBİTAK 1002-A 124E424 protokol numaralı proje ile desteklenmiştir.

Necip Şahamettin KÜÇÜK

ELAZIĞ, 2025

İÇİNDEKİLER

	Sayfa
ÖNSÖZ	iv
İÇİNDEKİLER	v
ÖZET.....	vii
ABSTRACT	viii
ŞEKİLLER LİSTESİ	ix
TABLULAR LİSTESİ	x
KISALTMALAR	xi
1. Giriş	1
1.1. Literatür Taraması	2
2. MATERYAL VE METOT	6
2.1. YOLOv4.....	6
2.1.1. Omurga.....	7
2.1.2. Genişlemiş evrişim.....	8
2.1.3. Boyun.....	10
2.1.4. Kafa.....	11
2.2. YOLOv8.....	12
2.2.1. Omurga.....	13
2.2.2. ELAN (Efficient Layer Aggregation Networks)	14
2.2.3. Boyun.....	14
2.2.4. Kafa.....	15
2.2.5. Yolov8 mimarisi.....	16
2.3. YOLOv11	18
2.3.1. Omurga.....	19
2.3.2. Boyun.....	20
2.3.3. Kafa.....	21
2.4. YOLO Mimarilerinin Hepsinde Orta Bulunan Özellikler	21
2.4.1. Sınırlayıcı Kutular	21
2.4.2. Bağlantı kutusu.....	22
2.4.3. Açgözlü-Maksimum Olmayanı Bastırma	23
2.4.4. Havuzlama Katmanı	23
2.4.5. Uzamsal Dikkat Modülü	23
2.4.6. Aktivasyon Fonksiyonu.....	24
2.4.7. Tanh Aktivasyon Fonksiyonu	24
2.4.8. Softplus Aktivasyon Fonksiyonu	25
2.4.9. Mish Aktivasyon Fonksiyonu	25
2.4.10. Normalizasyon	27
2.4.11. Küme Normalleştirme	27
2.4.12. Çapraz Mini Küme Normalizasyonu.....	27
2.5. Karışıklık matrisi	27
2.5.1. Doğruluk	27
2.5.2. Kesinlik	28
2.5.3. Duyarlılık	28
2.5.4. F-Skor.....	28

2.5.5. Özgüllük.....	29
2.5.6. Eğrinin Altındaki Alanın Alıcı İşletim Karakteristiği	29
2.5.7. Logaritmik Kayıp	29
3. BULGULAR.....	31
3.1. FOD-RUNWAY veri seti	31
3.2. YOLOv5 için detaylı sonuçlar.....	32
3.3. YOLOv8 için detaylı sonuçlar.....	33
3.4. Yolov11 için detaylı sonuçlar.....	34
3.5. Genel sonuç	35
4. TARTIŞMA	37
5. SONUÇLAR.....	39
ÖNERİLER	40
KAYNAKLAR.....	41
ÖZGEÇMİŞ	

ÖZET

Havalimanı Pistlerinde Yabancı Madde Tespiti için Derin Öğrenme Yaklaşımı

Necip Şahamettin KÜÇÜK

Yüksek Lisans Tezi

FIRAT ÜNİVERSİTESİ
Fen Bilimleri Enstitüsü

Havacılık Bilimi ve Teknolojileri Anabilim Dalı

Haziran 2025, Sayfa: xi + 44

Pist üzerindeki yabancı nesnelere, uçuş güvenliğini doğrudan tehdit eden ciddi bir risktir ve bu nedenle hızlı, doğru ve güvenilir şekilde tespit edilmeleri hayati öneme sahiptir. Çalışmanın amacı; havalimanı pistlerinde bulunan yabancı nesnelere yabancı madde hasarı (YMH) tespitinde kullanılacak, derin öğrenmeye dayalı bir nesne algılama sistemi kurmak ve farklı YOLO mimarilerinin (YOLOv5, YOLOv8 ve YOLOv11) performanslarını karşılaştırmalı olarak değerlendirmektir.

Tezin temel hipotezi, modern derin öğrenme temelli nesne algılama algoritmalarının YMH tespiti görevlerinde geleneksel yöntemlere kıyasla daha yüksek doğruluk, daha az yanlış negatif oranı ve farklı çevresel koşullara karşı daha yüksek dayanıklılık sağlayabileceğidir. Bu doğrultuda, hazır olarak sunulan YOLOv5s, YOLOv8s ve yeni nesil YOLOv11-s modelleri kullanılarak kapsamlı deneyler gerçekleştirilmiştir.

Tez kapsamında kullanılan veri seti, havalimanı ortamında toplanmış olan ve 31 farklı yabancı nesne sınıfı ile çeşitli hava ve ışık koşullarını içeren FOD-RUNWAY veri setidir. Değerlendirme sürecinde modeller; doğruluk (precision), duyarlılık (recall), F1 skoru, mAP@0.50 ve mAP@0.50–0.95 gibi çeşitli metrikler kullanılarak kıyaslanmıştır. Elde edilen sonuçlara göre, YOLOv11-s modeli, hafif yapısına rağmen YOLOv5s ve YOLOv8s modellerine kıyasla daha yüksek F1 skoru (0.861) ve daha iyi duyarlılık (0.826) sergilemiştir. Bu durum, özellikle güvenlik açısından kritik olan uygulamalarda yanlış negatiflerin azaltılması açısından önemli bir avantaj sağlamaktadır.

YOLOv8s modeli, genel doğruluk ve mAP performansı bakımından öne çıkarken, YOLOv5s bazı durumlarda daha düşük recall oranlarına sahip olmuştur. Ayrıca, YOLOv11-s modelinin yalnızca 9.45 milyon parametreye sahip olması ve yaklaşık 45 dakikalık bir sürede orta seviye donanımda eğitilebilmesi, sınırlı kaynaklara sahip gömülü sistemlerde veya gerçek zamanlı havaalanı uygulamalarında kullanılabilirliğini artırmaktadır.

Sonuç olarak, bu tezde herhangi bir yeni mimari geliştirilmeden, hazır ve güncel YOLO modellerinin performansları derinlemesine analiz edilmiştir. Elde edilen bulgular, derin öğrenme yöntemlerinin YMH tespitinde geleneksel sistemlere göre önemli avantajlar sunduğunu ve özellikle YOLOv11-s modelinin hafif yapısı ve güçlü genelleme kabiliyeti sayesinde havalimanlarında gerçek zamanlı ve ölçeklenebilir güvenlik çözümleri için etkili bir seçenek olabileceğini ortaya koymaktadır.

Anahtar Kelimeler: Havalimanı, yabancı nesne tespiti, derin öğrenme, YOLOv4, FOD-A

ABSTRACT

Deep Learning Approach for Foreign Object Detection on Airport Runways

Necip Şahamettin KÜÇÜK

Master's Thesis

FIRAT UNIVERSITY

Graduate School of Natural and Applied Sciences

Department of Aviation Science and Technologies

June 2025, Pages: xi + 44

The aim of this thesis is to establish a deep learning-based object detection system for identifying foreign object debris (FOD) on airport runways and to conduct a comparative performance evaluation of different YOLO architectures (YOLOv5, YOLOv8, and YOLOv11). Foreign objects on runways pose a serious threat to flight safety, making their fast, accurate, and reliable detection critically important.

The central hypothesis of the thesis is that modern deep learning-based object detection algorithms offer higher accuracy, lower false negative rates, and greater robustness to varying environmental conditions compared to traditional methods for FOD detection. Accordingly, comprehensive experiments were conducted using publicly available YOLOv5s, YOLOv8s, and the latest YOLOv11-s models.

The dataset used in this study is the FOD-A dataset, which was collected in airport environments and includes 31 different FOD classes under various weather and lighting conditions. The models were evaluated using performance metrics such as precision, recall, F1-score, mAP@0.50, and mAP@0.50–0.95. According to the results, the YOLOv11-s model, despite its lightweight architecture, outperformed YOLOv5s and YOLOv8s in terms of F1-score (0.861) and recall (0.826). This demonstrates a significant advantage in minimizing false negatives, especially in safety-critical applications.

While the YOLOv8s model stood out in terms of overall precision and mAP performance, YOLOv5s exhibited lower recall in certain scenarios. Furthermore, the YOLOv11-s model, with only 9.45 million parameters and a training time of approximately 45 minutes on a mid-range GPU, offers high applicability in resource-constrained embedded systems or real-time airport applications.

In conclusion, this thesis does not introduce a new architecture but provides an in-depth analysis of the performance of existing YOLO models. The findings highlight that deep learning methods offer significant advantages over traditional systems in FOD detection, and the YOLOv11-s model, in particular, emerges as an effective and scalable solution for real-time runway safety due to its lightweight structure and strong generalization capability.

Keywords: Airport, foreign object detection, deep learning, YOLOv4, FOD-A

ŞEKİLLER LİSTESİ

	Sayfa
Şekil 2.1. YOLOv4 genel mimari.....	7
Şekil 2.2. YOLOv4'ün detaylı mimarisi gösterimi	7
Şekil 2.3. Erişim işlemi gösterimi	8
Şekil 2.4. Genişlemiş evrişiminin görsel olarak anlatımı	9
Şekil 2.5. CSPDarknet53 omurgası	9
Şekil 2.6 C2f modülünün genişletilmiş modülü	13
Şekil 2.7 Gradyan akış yapısı	13
Şekil 2.8 Yolov8'in mimarisi	17
Şekil 2.9. MSCOCO veri setinden sınırlayıcı kutu örnekleri gösterimi	22
Şekil 2.10. Bağlantı kutusu veri setindeki ln(genişlik) ve ln(yükseklik) marjinal dağılımları gösterimi	22
Şekil 2.11. Maksimum havuzlama metodu	23
Şekil 2.12. Tanh aktivasyon fonksiyonu	24
Şekil 2.13. Softplus aktivasyon fonksiyonu	25
Şekil 2.14. Mish aktivasyon fonksiyonu	26
Şekil 3.1. Yolov5 modelinin eğitim çıktıları	32
Şekil 3.2 Eğitim sonuçları.	33
Şekil 3.3 Yolov5 ile Yolov8'in karşılaştırılması	34

TABLULAR LİSTESİ

	Sayfa
Tablo 2.1 FPN Yolov8'deki katman değerleri	15
Tablo 3.1. FOD-RUNWAY veri setine elde edilen sonuçlar	31
Tablo 3.2. Yolov5 mimarisinin alt modülleri üzerindeki sonuçları	33
Tablo 3.3. Yolov8 mimarisinin alt modülleri üzerindeki eğitim sonuçları	33
Tablo 3.4. Eğitilen tüm yolo modellerinin s çıktılarının sonuçları.....	35



KISALTMALAR

Kısaltmalar

YMH	: Yabancı madde hasarı
YOLO	: You Only Look Once (Sadece Bir Kere Bak)
SPP	: Uzamsal Piramit Havuzlama
PAN	: Yol Toplama Ağı
CloU	: Tam Kesişim Üzeri Birlik
FOD-A	: Havaalanlarında Yabancı Nesne Enkazı
FLOPs	: Saniyede kayan nokta işlemi
FPS	: Saniyede kare sayısı
mAP	: Ortalama hassasiyetin ortalaması
CSP	: Aşamalar Arası Kısmi
AUC	: Eğrinin Altındaki Alan
Açgözlü-NMS	: En Büyük Olmayanları Bastırma
CSP	: Aşamalar Arası Kısmi
SAM	: Uzamsal Dikkat Modül

1. GİRİŞ

Havayolu taşımacılığı, çağımızın en hızlı ve en güvenli ulaşım türlerinden biridir. Ancak, hava yolculuğunun güvenliği ve verimliliği, çeşitli risk faktörlerinin etkin bir şekilde yönetilmesini gerektirmektedir. Bu risklerden biri de havalimanı pistlerinde bulunan yabancı madde hasarı (YMH)'dir. FAA, YMH'yi, hava sahasında veya havaalanlarında bulunan ve havacılık operasyonlarını etkileyebilecek veya tehlikeye atabilecek herhangi bir yabancı cisim olarak tanımlar. Yabancı nesne tehlikesi, uçakların kalkış ve iniş sırasında kritik hasar almasına neden olabilir. Dolayısıyla YMH tespiti, hava taşımacılığı güvenliği açısından büyük önem taşımaktadır. Bu çalışmada, havalimanlarında derin öğrenme yöntemleri kullanarak YMH tespiti yapabilecek bir sistemin geliştirilmesi amaçlanmaktadır (FAA, 2010; SHGM, 2014).

Önerilen çalışma, geliştirilecek olan sistemin çeşitli ışık ve hava koşullarında etkin bir şekilde çalışabilmesi ve küçük ölçekli YMH'leri çoklu ölçekli öznelik birleştirme ile tespit edebilmesi öngörülmektedir. Derin öğrenme yöntemlerinin yüksek doğruluk ve dayanıklılık sağladığına dair genel çıkarımlara dayanarak çalışma derin öğrenme yöntemlerine odaklı olacaktır. YMH tespitine yönelik literatürdeki bazı yöntemler aşağıda belirtilmiştir.

Gözle tespit yöntemi, çalışma alanlarında, pistlerde ve benzeri yerlerde daha büyük türdeki YMH'leri gözle kontrol etmeyi içermektedir. Bu yöntemin avantajı basit ve ucuz olmasıdır. Ancak, bu yöntemin dezavantajları zaman alıcı olması, insan hatasına açık olması ve küçük ölçekli YMH'leri kaçırma riski taşımasıdır.

Kamera ve kızılötesi teknolojileri, YMH varlığını otomatik olarak taramak için kamera ve kızıl ötesi teknolojilerinden yararlanılmaktadır. Bu yöntemin avantajı daha hızlı ve doğru olmasıdır. Ancak, bu yöntemin dezavantajları ise yüksek maliyetli olması, hava koşullarından etkilenmesi ve küçük ölçekli YMH'leri algılamada zorluklar yaşamasıdır (J. Wang vd., 2021).

Ses tabanlı akıllı YMH tespiti, roket tanklarının son montajında YMH tespiti için ses sinyallerinden faydalanan bir yöntemdir. Bu yöntemin avantajı, görüntü tabanlı yöntemlere göre daha az hesaplama kaynağı gerektirmesi ve karanlık veya sisli gibi zorlu ortamlarda çalışabilmesidir. Ancak yöntemin dezavantajı ses sinyallerinin arka plan gürültüsünden etkilenmesi ve farklı YMH türlerini ayırt etmede zorluklar yaşamasıdır (Lin vd., 2023).

Milimetre dalga radar sensörleri tabanlı YMH tespiti, YMH tespiti için milimetre dalga radar sensörleri kullanılmaktadır. Bu yöntemin avantajı, yüksek çözünürlükte görüntüler üretebilmesi ve hava koşullarından etkilenmemesidir. Ancak yöntemin dezavantajı yüksek maliyetli olması ve radar sinyallerinin yansıma veya saçılma gibi etkilerden etkilenmesidir (TÜBİTAK, 2023; J. Zhong vd., 2021).

Derin öğrenme tabanlı nesne algılama yöntemleri, YMH'leri görüntülerden sınıflandırmak için derin sinir ağları kullanan yöntemdir. Yöntemin avantajı çoklu ölçekli özellik füzyonu

yapabilmesi ve küçük ölçekli YMH'leri yüksek doğrulukla tespit edebilmesidir. Ancak, bu yöntemin dezavantajları yeterli miktarda eğitim verisine ihtiyaç duyması, hesaplama kaynağı gerektirmesi ve farklı ortamlara uyarlanmasının zor olmasıdır (Li & Li, 2020; Papadopoulos & Gonzalez, 2021).

Çalışmada ele alınan yöntem, derin öğrenmeye dayalı nesne algılama algoritmaları olan YOLOv5, YOLOv8 ve YOLOv11 mimarileridir. Bu modeller, özellikle gerçek zamanlı nesne tespiti konusunda yüksek başarı oranları ve işlem verimliliği ile öne çıkmaktadır. YOLOv5, PyTorch tabanlı açık kaynaklı bir mimari olup esnek yapısı ve hızlı çalışması ile dikkat çekerken; YOLOv8, gelişmiş katmanlar, optimize edilmiş parametreler ve daha yüksek doğruluk oranları ile nesne algılamada son teknoloji bir yaklaşım sunmaktadır. YOLOv11 ise, daha hafif yapısına rağmen güçlü bir genel performans sergileyen yeni nesil bir modeldir. Bu modellerin tamamı, çeşitli boyutlardaki YMH'lerin (Yabancı Madde Hataları) yüksek doğrulukla tespit edilmesini sağlayacak şekilde değerlendirilmiştir.

Çalışma kapsamında kullanılan FOD-Runway FOD veri seti, havalimanı pistlerinde bulunan yabancı nesnelere tespiti amacıyla geliştirilmiştir. FOD-Runway veri seti, 71 farklı yabancı nesne sınıfını içeren ve çeşitli hava, ışık ve mesafe koşullarında toplanmış 60.000'den fazla etiketli görüntüden oluşmaktadır. FOD-Runway FOD veri seti ise pist ortamlarına özgü nesne türlerini içeren, farklı çekim açıları ve uzaklık seviyelerinde hazırlanmış görüntüleri barındırmaktadır. Bu veri setleri sayesinde derin öğrenme modellerinin farklı çevresel koşullara karşı dayanıklılığı test edilebilmiş, ayrıca küçük ve uzak nesnelere tespiti konusundaki başarıları da detaylı biçimde analiz edilmiştir.

Bu bağlamda yapılan çalışma, YOLO mimarilerinin FOD tespiti görevlerinde nasıl bir performans sergilediğini kapsamlı bir şekilde karşılaştırmalı olarak incelemeyi amaçlamaktadır. Geleneksel yöntemlerin ötesinde, modern derin öğrenme yaklaşımlarının pist güvenliğini artırmada ne ölçüde etkili olabileceği değerlendirilmektedir. Çalışmanın nihai amacı, hava taşımacılığı güvenliğini artırmak amacıyla pist üzerindeki potansiyel tehlikeleri hızlı ve doğru şekilde tespit edebilen, gömülü sistemlere uygun, gerçek zamanlı çalışabilecek nesne algılama çözümleri için uygun model önerileri sunmaktır.

1.1. Literatür Taraması

Qin vd. (2021), havalimanı pistlerinde YMH tespiti için milimetre dalga (MMW) gözetleme radarı kullanarak otomatik hedef tespiti algoritması önermişlerdir. YMH, havaalanında uçaklara veya personellere zarar verebilecek herhangi bir yabancı maddeyi ifade eder. YMH tespiti için mevcut radar sistemlerinin etkili menzili kısadır ve tespit edilebilir nesnelere radar kesit şiddeti en az -20 dBsm'dir. Bu nedenle, yazarlar, 660 m'den fazla uzun menzilli koşullarda küçük hedef tespitini iyileştirmek için bütünlük bir YMH otomatik hedef tespiti algoritması geliştirmişlerdir.

Algoritma, YMH'nin sinyal formunu ve milimetre dalga radarı tarafından alınan yer paraziti modelini teorik olarak kullanmakta ve kurmaktadır. Otomatik çıkarma adımında, pist bölgesi ilgi alanı olarak süreklilik özelliklerine dayanan pist kenarı tespit yöntemi kullanılmaktadır. Parazit haritası sabit yanlış alarm tespit algoritmasını takiben, yazarlar, hayati tespit işlemcisi olarak işlev gören bir zaman alanı algoritması kullanmışlardır. Ayrıca, YMH tespit performansının karakteristik nicel bir şekilde geliştirilen açık bir tanımını sunmuşlardır. Bu kriter, tüm YMH radar sistemleri için mutlak bir referans değeri içermektedir. Yazarlar, iyi tasarlanmış bir YMH frekans modüleli sürekli dalga MMW gözetleme radarı kullanmışlar ve deneysel çalışmaları Çin'in Pekin kentindeki gerçek bir havaalanında yapmışlardır. Sonuçlar, önerilen yöntemin etkinliğini ve uzun menzilli durumlarda YMH hedef tespitinin üstün performansını doğrulamıştır (Qin vd., 2021).

Lin vd. (2023), roket tanklarının son montajında YMH tespiti için ses tabanlı akıllı bir tespit sistemi önermişlerdir. Roket tankları, sondaj, perçinleme ve cıvatalama gibi işlemler içeren bir bağlama süreci ile bir araya getirilen çoklu tanklardan oluşur. Ancak, bu süreç istemeden tankın içinde küçük cıvata, somun, vida, titanyum tel ve diğer metal kalıntıları gibi YMH'ye neden olabilir. YMH, zamanında tespit edilip çıkarılmazsa fırlatma sırasında yakıtla birlikte motora girebilir ve ciddi kazalara yol açabilir. Bu nedenle, yazarlar, manuel tespit dezavantajlarını aşmak ve kazaları önceden önlemek için ses sinyallerine dayalı otomatik akıllı bir YMH tespit sistemi geliştirmişlerdir. Öncelikle, ses sinyalinin yüksek örnekleme hızını azaltmak için log-Mel dönüşümü kullanmışlardır. Ayrıca, şüpheli YMH olaylarını tespit etmek için çok ölçekli zamansal özellik çıkarma zorluklarını aşmak için çok ölçekli evrimsel ve zamansal evrimsel ağ (MS-CTCN) önermişlerdir. Son olarak, YMH olay tespit sonuçlarını iyileştirmek için etiket yumuşatma ve eşik ayırımı gibi önerilen son işlem stratejilerini kullanmışlar ve son olarak YMH varlığını belirlemişlerdir. Önerilen yöntem YMH deneyleri ile doğrulanmıştır. Sonuçlar, yöntemin YMH tespitinde %99,16 doğruluk oranına sahip olduğunu ve temel yöntemle göre kazaları önlemede daha iyi bir potansiyele sahip olduğunu göstermiştir (Lin vd., 2023).

Papadopoulos ve Gonzalez (2021), pistteki YMH tespiti için insansız hava aracı (UAV) ve yapay zekâ (AI) uygulaması önermişlerdir. YMH tespiti için mevcut yöntemler arasında radar, kızılötesi teknolojileri ve pist üzerine monte edilmiş sabit kameralar bulunmaktadır. Ancak, bu yöntemlerin hepsinin küçük YMH öğelerini tespit etmede yetersiz olduğu ve yüksek yanlış pozitif oranına sahip olduğu belirtilmektedir. Bu nedenle, yazarlar, manuel tespit dezavantajlarını aşmak ve kazaları önceden önlemek için ses sinyallerine dayalı otomatik akıllı bir YMH tespit sistemi geliştirmişlerdir. Sistem, bir UAV, bir RGB kamera, derin öğrenme yöntemleri kullanılarak eğitilen bir AI detektörü ve pist üzerinde yerel olarak toplanan görüntülerden oluşmaktadır. Özellikle kağıt, metal, cıvata, plastik ve plastik şişe gibi sınıflara bakılmıştır. Görüntüler toplanırken hem tam güneş ışığı hem de bulutlu hava koşulları dikkate alınmıştır. Detektör, yeniden boyutlandırma, döndürme ve renk artırma gibi çeşitli veri artırma teknikleri ile eğitilmiştir. Sonuçlar, yöntemin YMH

tespitinde %99,16 doğruluk oranına sahip olduğunu ve temel yöntemle göre kazaları önlemede daha iyi bir potansiyele sahip olduğunu göstermiştir (Papadopoulos & Gonzalez, 2021).

Li ve Li (2020), havalimanı pistlerinde YMH tespiti için YOLOv3 (You Only Look Once) tabanlı bir tespit algoritması sunmuşlardır. YMH tespiti için mevcut yöntemler arasında radar, kızılötesi teknolojileri ve pist üzerine monte edilmiş sabit kameralar bulunmaktadır. Ancak, bu yöntemlerin hepsinin küçük YMH öğelerini tespit etmede yetersiz olduğu ve yüksek yanlış pozitif oranına sahip olduğu belirtilmektedir. Bu nedenle, yazarlar, derin artıklı ağ (deep residual network) kullanarak özellik çıkarma ve çok ölçekli özellik füzyonu kullanarak küçük ölçekli YMH'leri tespit etmek için YOLOv3 tabanlı bir algoritma önermişlerdir. YMH için bir örnek veri seti oluşturmuşlar ve önerdikleri yöntemi denemişlerdir. Deney sonuçları, YOLOv3 tabanlı tespit algoritmasının YMH'yi etkili bir şekilde tespit ettiğini ve iyi bir doğruluk ve sağlamlığa sahip olduğunu göstermiştir (Li & Li, 2020).

Munyer vd. (2022), havalimanı pisti görüntülerinde YMH tespiti için kendinden denetimli yerleştirme ve görüş dönüştürücü (Vision Transformer) tabanlı bir yöntem önermişlerdir. YMH tespiti için mevcut yöntemler arasında YOLO ve SSD gibi denetimli nesne tespiti algoritmaları bulunmaktadır. Ancak, bu yöntemler YMH tespiti için uygun değildir, çünkü YMH herhangi bir nesne olabileceği halde bu yöntemler önceden tanımlanmış sınıfları tespit edebilmektedir. Bu nedenle, yazarlar, YMH anotasyon örneklerinin sayısallaştırılmasından kaçınmak için pist görüntülerini tahmin etmeyi öğrenen kendinden denetimli bir YMH yerleştirme yöntemi sunmuşlardır. Yerleştirme yöntemi, yerleştirme performansını artırmak için Görüş Dönüştürücü (ViT) kullanılmaktadır. Deneyle, yöntemin gerçek dünya pist durumlarında keyfi YMH'leri başarıyla tespit ettiğini göstermiştir. Makale ayrıca, yerleştirme sonucuna sınıflandırma yapmak için bir uzantı da sağlamıştır; bu da alt seviye görevler için yararlı olabilir. Yerleştirme eğitimi için bu makale ayrıca sadece temiz pist görüntülerini toplayan basit ve gerçekçi bir veri seti oluşturma çerçevesi de sunmuştur. Bu yöntem için eğitim ve test verileri insansız hava araçları (UAS) kullanılarak yerel bir havaalanında toplanmıştır. Ayrıca, geliştirilen veri seti kamu kullanımına ve daha fazla çalışmaya sunulmuştur.

Noroozi ve Shah (2023), havalimanı ortamında YMH tespiti için optimal bir yöntem geliştirmişlerdir. YMH tespiti için mevcut yöntemler arasında ultrasonik sensör sistemi, radar tespit sistemi, hibrit tespit sistemi ve kamera tespit sistemi gibi geleneksel ve otomatik tespit yöntemleri bulunmaktadır (Fizza vd., 2021). Ancak, bu yöntemlerin hepsinin YMH'nin tüm çeşitlerini doğru ve zamanında tespit etmede yetersiz olduğu belirtilmektedir. Bu nedenle, yazarlar, YMH'yi hızlı bir şekilde ortamdaki uzaklaştırmak için bir nesne tespit çerçevesi geliştirmişlerdir. Önerilen YMH tespit çerçevesi şunlardan oluşmaktadır:

- 1)Pistleri denetlemek ve veri toplamak için insansız hava aracı kullanmak.

2)Veri setlerinde bulunan yabancı nesne tipleri, hava koşulları ve havaalanı yüzey malzemelerinin sınırlı olması sorununu gidermek için veri ön işleme ve artırma teknikleri belirlemek.

3)Gerçek dünya havaalanı ortamında dağıtım için yüksek doğruluk ve daha hızlı çıkarım süresine ulaşmak için bilgisayar görüşü tabanlı bir nesne tespit modeli geliştirmek.

Eğitim verilerini İHA ile bir hava kuvvetleri aralığında oluşturmuşlar ve çerçevelerinde YOLO (You Only Look Once) nesne detektörü ailesi modellerini (tek aşamalı nesne detektörleri) de dahil olmak üzere çeşitli modeller geliştirmişlerdir. Modelleri, UAS tarafından toplanan daha önce görülmemiş verilerde ve YMH görüntülerinin kamuya açık bir veri setinde değerlendirmişlerdir. Deney sonuçları, YOLO modelinin bir versiyonu (YOLOv4) ile aktarım öğrenmesinin diğer modellerden daha hızlı çıkarım süresine sahip olduğunu ve hassasiyet ve geri çağırma metrik değerlerinde diğer modellerden daha iyi performans gösterdiğini göstermiştir (Noroozi & Shah, 2023).

2. MATERYAL VE METOT

Bu bölüm, tezde kullanılan yöntemlerin ayrıntılı bir incelemesini ve kullanılan materyallerin tanımlanmasını içermektedir. Çalışmada, havaalanlarında YMH tespiti için YOLOv4 tabanlı bir derin öğrenme modeli kullanılmıştır. Çalışmanın amacı, YOLOv4 modelinin çeşitli ışık ve hava koşullarında ve küçük ölçekli YMH'leri tespit etme yeteneğini araştırmaktır.

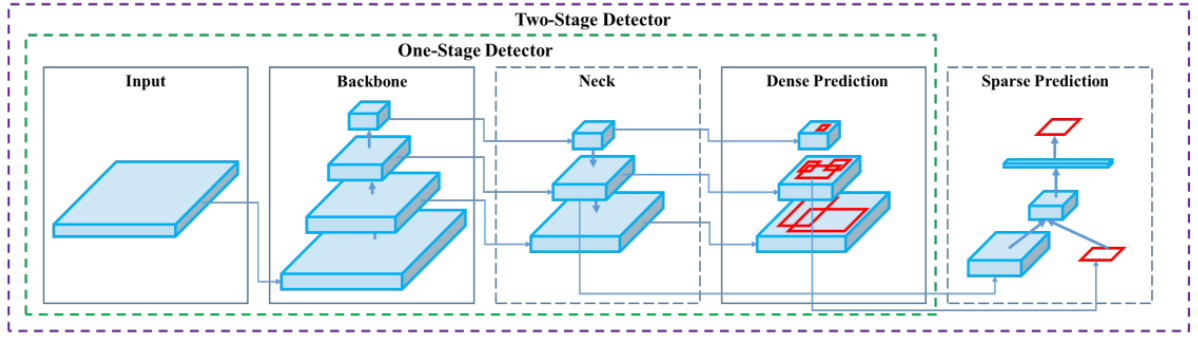
Çalışmada, 31 farklı YMH kategorisi ve 30.000'den fazla etiketli görüntü içeren FOD-A adlı bir veri seti kullanılmıştır (Munyer vd., 2021). Veri seti, farklı ışık ve hava koşullarını kapsar ve bu çeşitlilik, modelin çeşitli koşullarda eğitilmesi ve test edilmesi için ideal bir ortam sağlar.

Çalışma, havaalanlarında YMH tespiti için bir derin öğrenme modeli olan YOLOv4'ün etkinliğini ve bu modelin çeşitli ışık ve hava koşullarında ve küçük ölçekli YMH'leri tespit etme yeteneğini değerlendirmektedir. Bu amaçla, modelin eğitilmesi ve test edilmesi sürecinde geniş bir veri seti olan FOD-A veri seti kullanılmıştır. Modelin performansının değerlendirilmesi için çeşitli metrikler kullanılmıştır. Çalışmanın bulguları, YOLOv4 modelinin YMH tespitinde yüksek doğruluk ve etkinlik sağlayabileceğini göstermiştir. Bu bulgular, hava taşımacılığı güvenliğinin artırılması için potansiyel bir çözüm sunmaktadır.

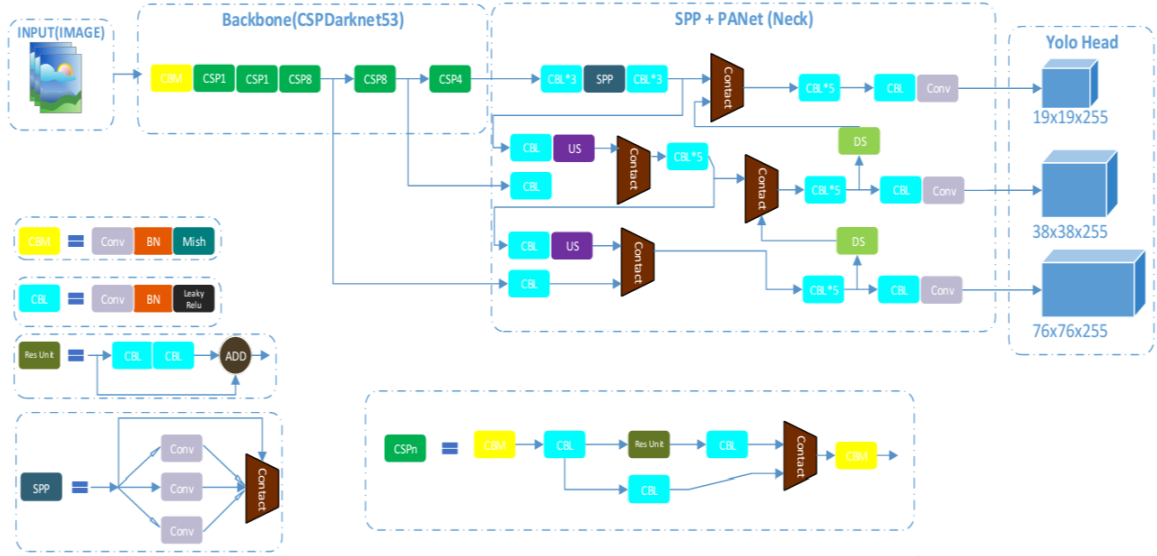
2.1. YOLOv4

YOLOv4, Nisan 2020'de yayınlanan ve COCO veri setinde son teknoloji performansa ulaşan gerçek zamanlı bir nesne algılama modelidir. YOLOv4, nesne algılama görevini iki parçaya bölerek çalışır; sınırlayıcı kutular aracılığıyla nesne konumlandırmasını tanımlamak için regresyon ve sınırlayıcı kutuların içindeki nesnelerin sınıflarını tahmin etmek için sınıflandırma. Çalışmada, YOLOv4 modelinin yapısı, eğitimi ve performansı hakkında ayrıntılı bir inceleme yapılmıştır. Bu amaçla, aşağıdaki başlıklar altında modelin özellikleri ve avantajları ele alınmıştır.

Omurga, modelin özellik çıkarma katmanı olan omurga kısmında CSPDarknet53 adlı yeni bir omurga ağı kullanılmaktadır (Xu vd., 2021). Boyun, modelin özellik füzyonu katmanı olan boyun kısmında SPP (Uzamsal Piramit Havuzlama) ve PAN (Yol Toplama Ağı) gibi modüller kullanılarak çoklu ölçekli özellik füzyonu yapılabilen bir yöntem uygulanmaktadır (K. He vd., 2015; Liu vd., 2019). Kafa, modelin nesne algılama katmanı olan kafa kısmında YOLOv3'te kullanılan yöntemler devam ettirilmektedir. Şekil 2.1'de YOLOv4 modelinin genel mimarisi, Şekil 2.2'de ise YOLOv4 modelinin detaylı mimarisi görülmektedir.



Şekil 2.1. YOLOv4 genel mimari (Bochkovskiy vd., 2020)



Şekil 2.2. YOLOv4'ün detaylı mimarisi gösterimi (Güçlü vd., 2021)

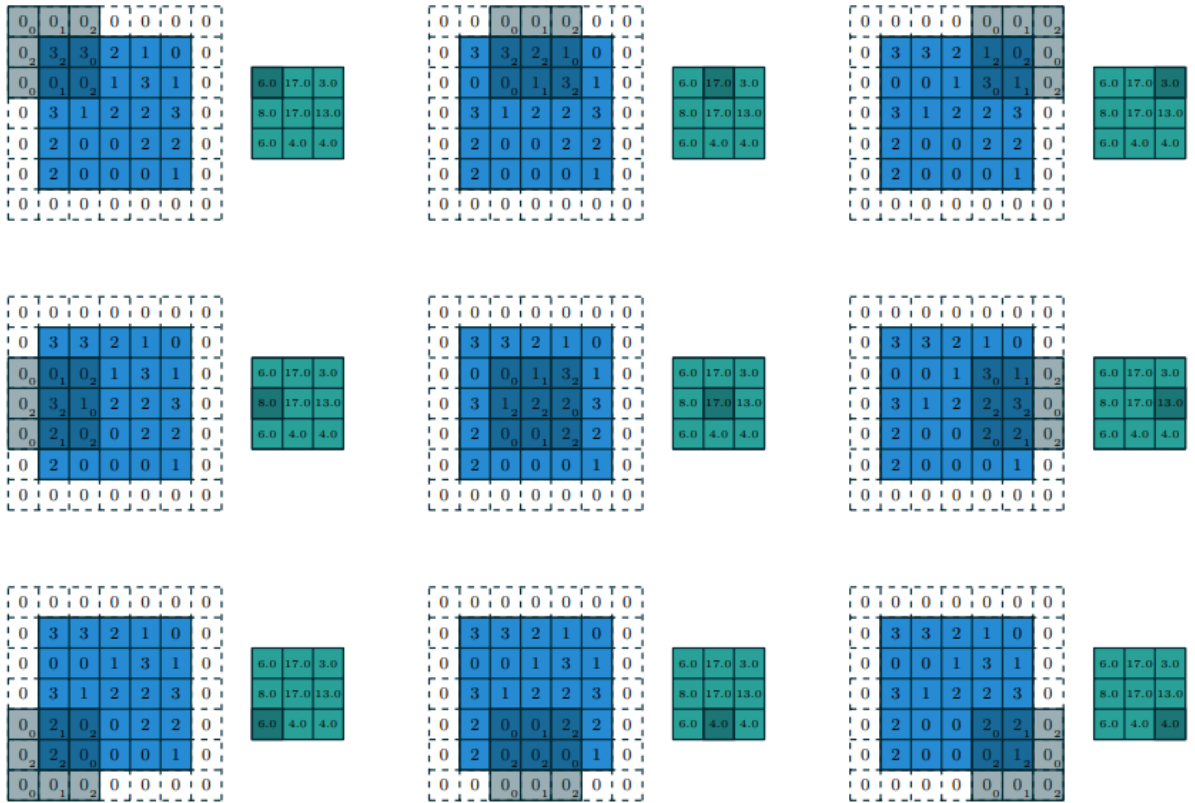
2.1.1. Omurga

Modelin omurga kısmı olan özellik çıkarma katmanında yeni bir omurga ağı olan CSPDarknet53 kullanılır. Girdi olarak verilen görüntülerden işe yarar ve hesaplanabilir bilgiler elde etmek omurga kısmının amacıdır. Bu bilgiler, nesne algılama katmanına aktarılır ve sınırlayıcı kutuların oluşturulmasında ve sınıflandırılmasında kullanılır. Modelin genel başarısını ve verimliliğini etkileyen omurga kısmının performansıdır. Bu nedenle, omurga kısmının hızlı, doğru ve ölçeklenebilir olması önemlidir. CSPDarknet53, Darknet53 adlı omurga ağının geliştirilmiş bir sürümüdür (Xu vd., 2021, Redmon & Farhadi, 2018).

Darknet53, ResNet adlı popüler bir omurga ağının daha hızlı ve daha hafif bir alternatifidir (Li & Li, 2020). ResNet'ten farklı olarak Darknet53'te artık bloklar arasında genişlemiş evrişim (dilated convolution) yerine evrişim kullanılır (Xu vd., 2021, Li & Li, 2020, Dumoulin & Visin,

2016). Genişlemiş evrişim, özellik haritalarının boyutunu korumak için kullanılan ancak hesaplama maliyetini arttıran bir tekniktir (Khalifaoui-Hassani vd., 2021). Evrişim ise özellik haritalarının boyutunu azaltmak için kullanılan ancak hesaplama maliyetini azaltan bir tekniktir. Hesaplama maliyetini düşürmek için evrişim kullanan Darknet53'te, özellik haritalarının boyutunu korumak için artık blokların sayısı artırılır (Xu vd., 2021, Li & Li, 2020, Dumoulin & Visin, 2016).

Evrişim katmanı, evrişimli sinir ağlarının temel bileşenidir ve giriş verisindeki özellikleri çıkarmak için evrişim işlemleri gerçekleştirir. Evrişim katmanı, giriş verisinin boyutlarına göre kaydırılan küçük filtrelerden oluşur. Her filtre, giriş verisi ile çarpılır ve bir öznitelik haritası veya aktivasyon haritası üretir. Evrişim katmanının hiperparametreleri arasında filtre sayısı, filtre boyutu, adım aralığı ve sıfır ekleme vardır. Bu hiperparametreler, evrişim katmanının çıktısının boyutunu ve modelin karmaşıklığını etkiler. Evrişim katmanı, giriş verisindeki düşük ve yüksek seviyeli özellikleri algılamak için kullanılır. Şekil 2.3'de evrişim örneği gösterilmiştir (Dumoulin & Visin, 2016).

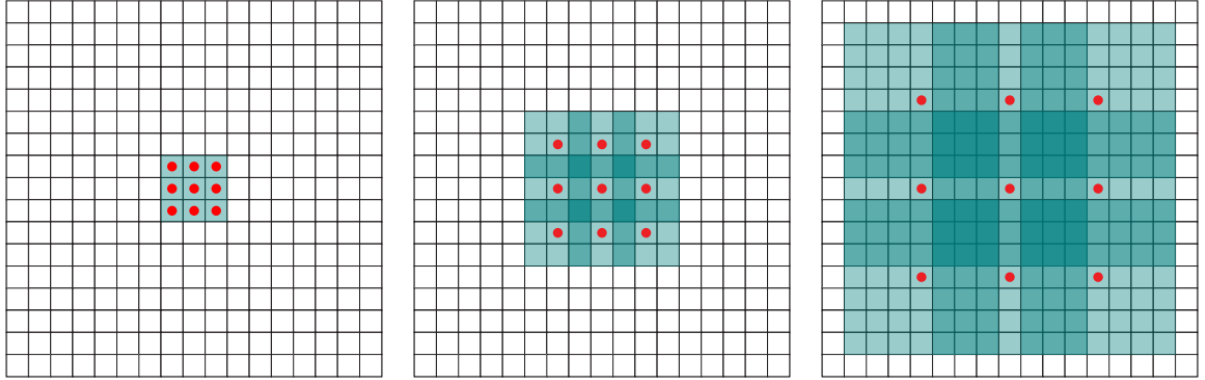


Şekil 2.3. Evrişim işlemi gösterimi (Dumoulin & Visin, 2016)

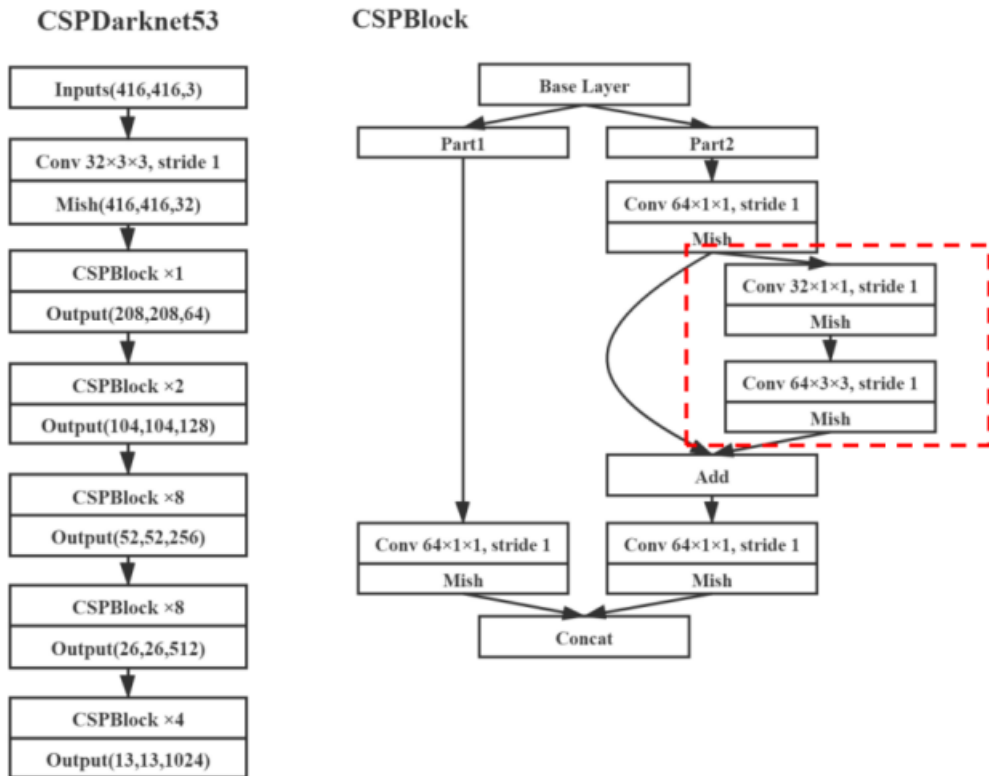
2.1.2. Genişlemiş evrişim

Genişlemiş evrişim, evrişimli sinir ağlarında kullanılan bir evrişim türüdür ve evrişim çekirdeğini (kernel) aralarına boşluklar ekleyerek genişletir. Boşlukların büyüklüğünü belirleyen bir (genişleme oranı) parametresi vardır. Genellikle çekirdek elemanları arasında 1-1 adet boşluk

bulunur. Genişlemiş evrişim, evrişim çekirdeğinin alıcı alanını (receptive field) arttırmak için kullanılır. Alıcı alan, evrişim çekirdeğinin giriş verisinde görebildiği alanı ifade eder. Evrişim çekirdeğinin boyutunu veya parametre sayısını arttırmadan alıcı alanını genişletir. Şekil 2.4’de genişleme evrişim görsel olarak sunulmuştur (Yu & Koltun, 2016). Şekil 2.5’te CSPDarknet53 omurga yapısı görülmektedir.



Şekil 2.4. Genişlemiş evrişiminin görsel olarak anlatımı (Yu & Koltun, 2016)



Şekil 2.5. CSPDarknet53 omurgası (Redmon & Farhadi, 2018)

CSPDarknet53 ise Darknet53'ten farklı olarak residual blokları CSP (Cross Stage Partial) blokları ile değiştirir (Xu vd., 2021). CSP blokları, residual bloklara göre daha az parametre ve daha az FLOPs gerektirir. CSP blokları, girdi özellik haritasını iki gruba ayırarak çalışır. Bir grup artık bloklardan geçerken diğer grup doğrudan geçer. Sonra iki grup tekrar birleştirilir ve sonraki katmana aktarılır. Bu şekilde, CSP blokları hem yüksek seviyeli hem de düşük seviyeli özellikleri koruyabilmektedir.

CSPDarknet53, YOLOv4 modelinin omurga kısmında kullanılmıştır. CSPDarknet53, YOLOv4 modeline hem hız hem de doğruluk avantajı sağlamıştır. CSPDarknet53, COCO veri setinde 43.5 mAP değeri ile YOLOv3'ün omurga kısmında kullanılan Darknet53'ten 1 puan daha yüksek performans göstermişti (Xu vd., 2021, Redmon & Farhadi, 2018, Colleges vd., 2014). Ayrıca CSPDarknet53, 608x608 çözünürlükte 62 kategori ile YOLOv3'ün omurga kısmında kullanılan Darknet53'ten 10 FPS daha hızlı çalışmıştır. Bu sonuçlar, CSPDarknet53'ün YOLOv4 modelinin omurga kısmında uygun bir seçim olduğunu göstermektedir (Bochkovskiy vd., 2020; Redmon & Farhadi, 2018)

2.1.3. Boyun

Yolov4'ün boyun bölümü, giriş verisindeki farklı ölçekli nesnelere tespit etmek için evrişim çekirdeğinin alıcı alanını arttırmayı ve farklı seviyedeki özellikleri birleştirmeyi amaçlar. Yolov4'ün boyun bölümü iki ana modülden oluşur: SPP modülü ve PAN modülü.

SPP modülü, giriş verisinin en küçük ölçekli özellik haritasına uygulanan bir evrişim katmanından sonra devreye girer. SPP modülü, farklı boyutlarda maksimum havuzlama işlemleri gerçekleştirir ve elde edilen özellik haritalarını birleştirir. Böylelikle, SPP modülü, evrişim çekirdeğinin alıcı alanını genişleterek giriş verisindeki çok ölçekli bilgileri muhafaza eder. SPP modülü, 1×1, 5×5, 9×9 ve 13×13 boyutlarında maksimum havuzlama işlemleri uygular (K. He vd., 2015). SPP modülü, YOLOv4'ün neck bölümünün önemli bir parçasını oluşturan Spatial Pyramid Pooling (Uzamsal Piramit Havuzlama) modülünün bir çeşididir. SPP modülü, omurga ağının farklı katmanlarından gelen özellik haritalarını birleştirerek nesne tespiti için daha kapsamlı ve çeşitli bilgiler sağlar. Ayrıca, alıcı alanını genişleterek küçük nesnelere tespitini kolaylaştırır. SPP modülünden gelen birleştirilmiş özellik haritaları, PAN adı verilen bir yol birleştirme ağı ile yüksek çözünürlüklü özellik haritaları ile birleştirilir.

2.1.4. Kafa

PAN modülü, giriş verisinin farklı seviyelerindeki özellik haritalarının birleştirilmesi için yol toplama ağı (PAN) modülü kullanılmıştır. PAN modülü, FPN ve U-Net gibi mimarilerden ilham alınarak tasarlanmıştır (Liu vd., 2019; Weng & Zhu, 2021).

PAN modülü, omurga ağının farklı katmanlarından gelen özellik haritalarını birleştirmek için tasarlanmıştır. PAN modülüne giren düşük çözünürlüklü özellik haritası, mekansal piramit havuzlama (SPP) modülü ile elde edilmiştir. SPP modülü, 1x1, 5x5, 9x9 ve 13x13 boyutlarında maksimum havuzlama çekirdekleri kullanarak düşük çözünürlüklü özellik haritasının maksimum havuzlama çıktılarını birleştirmiştir. Bu işlem, omurga özelliklerinin alıcı alanını artırmış ve küçük nesnelere algılamada ağın doğruluğunu artırmıştır. SPP modülünden gelen birleştirilmiş özellik haritası, PAN modülü aracılığıyla yüksek çözünürlüklü özellik haritası ile birleştirilmiştir. PAN modülü, düşük seviyeli ve yüksek seviyeli özellikleri birleştirmek için yukarı örnekleme ve aşağı örnekleme işlemlerini kullanarak alttan yukarı ve yukarıdan aşağı yollar oluşturmuştur. PAN modülünden çıkan birleştirilmiş özellik haritaları, tahminler için üç adet YOLOv3 algılama başlığına gönderilmiştir. Her algılama başlığı, sınırlayıcı kutuları, nesnellik skorlarını ve sınıflandırma skorlarını hesaplamak için evrişimli katmanlar ve aktivasyon fonksiyonları içermiştir. Algılama başlıkları, 19x19, 38x38 ve 76x76 boyutlarında özellik haritaları üretmiştir (Liu vd., 2019).

Her bir tespit başlığı, çıktıları önceki katmanlardan alır ve nesne tespiti için gerekli olan tahminleri yapar. Bu tahminler, sınırlayıcı kutular (bounding boxes), nesne sınıfları ve nesne varlığı olasılıklarını içerir.

Nesne varlığı olasılıkları, bir sınırlayıcı kutunun belirli bir nesneyi içerip içermediğini tahmin eder. Nesne sınıfları, tespit edilen nesnenin hangi sınıfa ait olduğunu tahmin eder. Sınırlayıcı kutular ise nesnenin görüntü üzerindeki konumunu ve boyutunu belirler. Bu birleşik yaklaşım, YOLOv4'ün hem küçük hem de büyük ölçekli nesnelere tespit etmesini ve sınıflandırmasını sağlar, bu da modelin genel performansını ve doğruluğunu artırır.

Giriş verisinin farklı seviyelerindeki özellik haritalarının birleştirilmesi için yol birleştirme ağı (PAN) modülü kullanılmıştır. PAN modülünde, giriş verisinin en büyük ölçekli özellik haritasına bir evrişim katmanı uygulanmış ve sonraki seviyedeki özellik haritası ile birleştirilmiştir. Bu işlem, en küçük ölçekli özellik haritasına kadar devam ettirilmiştir. Daha sonra, en küçük ölçekli özellik haritasından başlayarak ters yönde aynı işlem tekrarlanmıştır. Böylece, PAN modülü ile hem yukarıdan aşağıya hem de aşağıdan yukarıya bilgi akışı sağlanmış ve farklı seviyelerdeki özellikler zenginleştirilmiştir. PAN modülünde kullanılan evrişim katmanlarının boyutları 3x3 olarak belirlenmiştir.

Yolov4'ün neck bölümü, SPP ve PAN modüllerinin bir arada kullanılmasıyla oluşturulmuştur. Bu bölüm, hem hızlı hem de doğru nesne tespiti yapabilen bir yapı sunmuştur.

2.2. YOLOv8

YOLOv8, Ocak 2023'te Ultralytics tarafından yayınlanan ve gerçek zamanlı nesne algılama alanında yüksek doğruluk ve hız sunan bir versiyonudur. YOLOv8, önceki sürümlerden farklı olarak daha modüler, esnek ve optimize edilmiş bir yapıya sahiptir. Nesne algılama görevini yine iki ana başlıkta yürütmektedir: sınırlayıcı kutular ile nesnelerin konumlarını belirlemek için regresyon ve kutular içerisindeki nesnelerin sınıflarını tahmin etmek için sınıflandırma. Bu çalışmada, YOLOv8 modelinin yapısını, eğitimi ve sunduğu performansı ayrıntılı biçimde inceleyeceğiz (Sohan vd., 2024).

Modelin özellik çıkarma kısmı olan omurga yapısında, YOLOv5'teki CSPNet temelli mimariden ayrılarak, daha sade ve verimli bir mimari benimsenmiştir. YOLOv8, standart bir backbone mimarisi olarak C2f (Concatenate to Fusion) blokları kullanmaktadır. Bu yapı, daha az parametre ile daha yüksek performans elde edilmesine olanak tanır. C2f blokları, derin katmanlardan gelen bilgileri daha etkin şekilde işleyerek bilgi kaybını azaltmakta ve modelin genel öğrenme kapasitesini artırmaktadır.

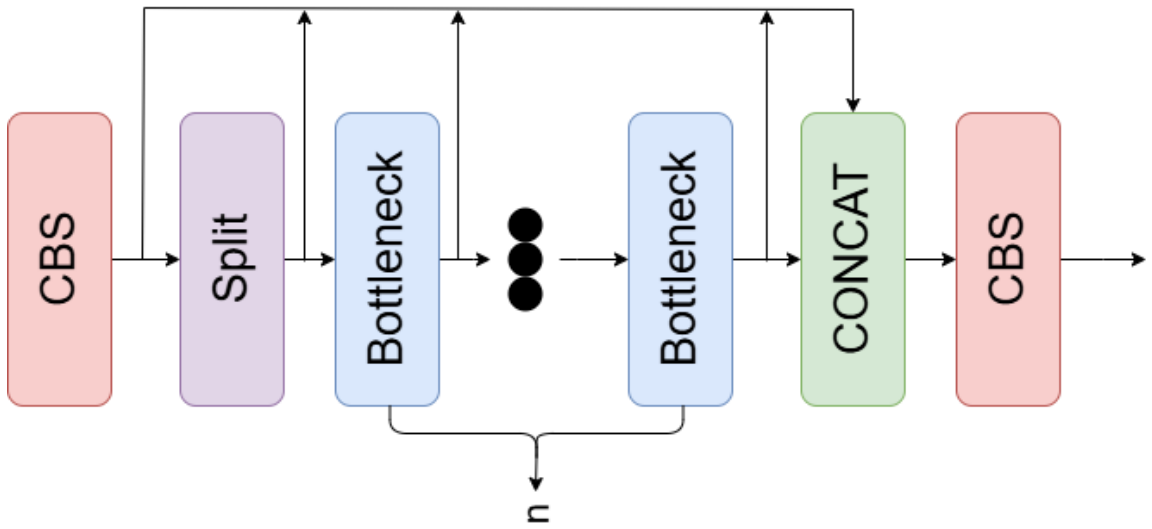
Boyun (neck) kısmında ise önceki sürümlerden aşına olunan PANet (Path Aggregation Network) yapısı tercih edilmiştir. PANet, alt ve üst düzey özelliklerin daha etkili bir şekilde birleştirilmesini sağlayarak farklı boyutlardaki nesnelerin tespiti için daha güçlü bir temsil sunar. YOLOv8'in boyun yapısı, optimize edilmiş yol izleme ve dikkat mekanizmaları ile desteklenerek modelin odaklanması gereken önemli özellikleri daha doğru biçimde birleştirmesini sağlamaktadır(Liu vd., 2019).

Kafa (head) kısmında, YOLOv8 ile birlikte anchor-free (çapasız) bir yaklaşım benimsenmiştir. Bu yöntem, geleneksel anchor tabanlı sistemlere kıyasla daha sade ve hızlıdır. Anchor-free yapı sayesinde, sınırlayıcı kutu tahminleri doğrudan yapılmakta ve bu da hem eğitim süresini azaltmakta hem de farklı nesne ölçeklerine daha iyi uyum sağlanmasını mümkün kılmaktadır. Bu yapı, aynı zamanda karmaşıklığı azaltarak modelin daha verimli çalışmasına katkı sağlar(Yan vd., 2021).

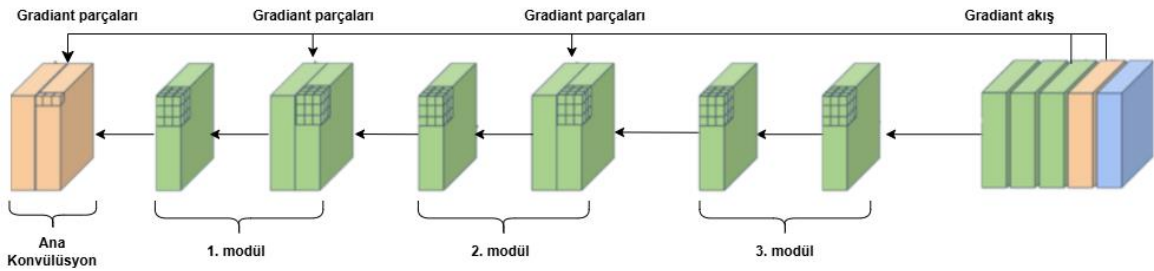
YOLOv8, önceki sürümlerden farklı olarak PyTorch tabanlı bir API ile sunulmuştur. Bu sayede daha modüler bir yapı kazanmış, kullanıcıların modeli farklı görevler için (örneğin sınıflandırma, segmentasyon veya nesne takibi) kolayca özelleştirebilmesi sağlanmıştır. Ayrıca gelişmiş veri artırma teknikleri, otomatik hiperparametre optimizasyonu ve donanım dostu yapısıyla, daha az kaynakla daha yüksek doğruluk oranları elde edilebilmektedir. Bu yönleriyle YOLOv8, hem araştırmacılar hem de endüstriyel uygulamalar için güçlü ve güncel bir çözüm sunmaktadır.

2.2.1. Omurga

YOLOv8'in omurgası, nesne algılama görevinde yüksek doğruluk ve verimlilik elde etmek amacıyla optimize edilmiş bir özellik çıkarım yapısıdır. Bu yapının merkezinde, önceki YOLO sürümlerinde yaygın olarak kullanılan C3 modülünün yerine geçen C2f (Concatenate-to-Fusion) modülü yer almaktadır. C2f modülü, ELAN (Efficient Layer Aggregation Networks) mimarisinden ilham alınarak geliştirilmiş, daha az hesaplama yüküyle daha verimli öğrenme sağlayan bir yapıdır (Xie vd., 2020). Bu modül, giriş özellik haritasını iki paralel gradyan akışı üzerinden işleyerek daha etkili bir bilgi akışı sağlar. Bu sayede, modelin derin katmanlarında gradyan kaybı azaltılmış olurken, hem düşük seviyeli detayların hem de yüksek seviyeli semantik bilgilerin birlikte kullanılması mümkün hale gelir. Şekil 2.6'da C2f modülünün genişletilmiş biçimi verilmiştir. Şekil 2.7'de ise gradyan akış yapısı görülmektedir.



Şekil 2.6 C2f modülünün genişletilmiş modülü



Şekil 2.7 Gradyan akış yapısı

Omurganın diğer önemli bileşenleri arasında standart evrişim katmanları (Conv), aktivasyon fonksiyonu olarak SinLU (Sigmoid Linear Unit) ve batch normalization yer almaktadır (Paul vd., 2022). Bu bileşenler, modelin istikrarlı ve hızlı öğrenmesini desteklerken, C2f blokları sayesinde parametre sayısı azaltılarak daha hafif bir mimari elde edilmiştir. Ayrıca, omurga

yapısına entegre edilen SPPF (Spatial Pyramid Pooling - Fast) modülü, farklı ölçeklerdeki özellik haritalarını birleştirerek çok ölçekli nesne algılama kapasitesini artırmaktadır. SPPF modülü, geleneksel SPP modülüne kıyasla daha hızlı çalışmakta ve modelin genel verimliliğini artırmaktadır (K. He vd., 2015).

YOLOv8'in omurga mimarisi, modern nesne algılama modellerinin gereksinimlerini karşılayacak şekilde sadeleştirilmiş, modülerleştirilmiş ve derinleştirilmiştir. C2f bloklarının getirdiği esnek yapı ve bilgi akışı, derin ağlarda verimli öğrenmeyi mümkün kılarak küçük, orta ve büyük ölçekli nesnelerin daha doğru tespit edilmesini sağlar. Yapılan çeşitli akademik çalışmalarda, bu omurga mimarisinin geleneksel yapılara göre daha az hesaplama kaynağı kullanarak daha yüksek doğruluk sunduğu gösterilmiştir. Ultralytics'in resmi belgeleri, açık kaynak kodları ve literatürdeki güncel çalışmalar (örneğin: "A Review on YOLOv8 and Its Advancements", ResearchGate, 2023; PLOS ONE, 2024) da bu yapının nesne algılama alanındaki önemini ortaya koymaktadır. Sonuç olarak, YOLOv8'in omurgası, gelişmiş yapısıyla hem akademik araştırmalar hem de endüstriyel uygulamalar için güçlü ve modern bir temel sunmaktadır.

2.2.2. ELAN (Efficient Layer Aggregation Networks)

Efficient Layer Aggregation Networks" (ELAN) mimarisi, YOLOv7 modelinde tanıtılmıştır ve YOLOv8'in omurga yapısındaki C2f modülüne ilham kaynağı olmuştur. ELAN, derin sinir ağlarında bilgi akışını optimize etmek ve parametre verimliliğini artırmak amacıyla tasarlanmıştır (C.-Y. Wang vd., 2023).

ELAN mimarisi, giriş özellik haritasını farklı gruplara ayırarak bu grupları paralel yollarla işler ve ardından bu yolların çıktısını birleştirerek zenginleştirilmiş bir özellik temsili elde eder. Bu yaklaşım, modelin hem düşük seviyeli detayları hem de yüksek seviyeli semantik bilgileri etkili bir şekilde öğrenmesini sağlar. Ayrıca, ELAN yapısı, gradyan akışını iyileştirerek derin ağlarda daha etkili öğrenme sağlar (Xie vd., 2020).

YOLOv8'de kullanılan C2f (Concatenate-to-Fusion) modülü, ELAN mimarisinden esinlenerek geliştirilmiştir. C2f, giriş özellik haritasını ikiye ayırarak farklı işlemlerden geçirir ve ardından bu çıktıları birleştirerek zenginleştirilmiş bir özellik haritası elde eder. Bu yapı, hem düşük seviyeli detayları hem de yüksek seviyeli semantik bilgileri etkili bir şekilde birleştirerek küçük nesnelerin tespitinde performansı artırır.

2.2.3. Boyun

YOLOv8'in boyun (neck) bölümü, modelin omurgasından (backbone) gelen çok ölçekli özellik haritalarını birleştirerek zenginleştirilmiş temsil vektörleri üretir. Bu bölüm, nesne algılama görevinde farklı boyutlardaki nesnelerin daha etkili şekilde tanınabilmesini sağlar. YOLOv8'in

boyun yapısı, önceki YOLO sürümlerinde olduğu gibi PANet (Path Aggregation Network) mimarisine dayanmaktadır. PANet, hem yukarıdan aşağıya hem de aşağıdan yukarıya bilgi akışı sağlayarak düşük ve yüksek seviyeli özellikleri birbirine entegre eder. Bu çift yönlü bilgi akışı, küçük nesnelere için düşük seviyeli detayların korunmasını sağlarken, büyük nesnelere için daha semantik zengin özellikler sunar (Liu vd., 2019).

YOLOv8'in boyun kısmı aynı zamanda, optimize edilmiş evrişimli katmanlar ve C2f modülleri ile desteklenmiştir. Bu yapılar, bilgi kaybını en aza indirirken, hesaplama verimliliğini artırarak daha hızlı çıkarım (inference) süreleri sağlar. PANet'in bu versiyonu, klasik FPN (Feature Pyramid Network) ile karşılaştırıldığında, daha fazla bilgi aktarımı ve daha iyi gradyan akışı sunar. Böylece eğitim sırasında modelin daha kararlı öğrenmesi ve çıkarım sırasında daha hassas nesne tespiti yapılması mümkün olur (Liu vd., 2019; Zhao vd., 2019). Tablo 2.1'de FPN'nin YOLOv8'deki katman değerleri görülmektedir.

Tablo 2.1 FPN YOLOv8'deki katman değerleri

Seviye	Girdi Boyutu	Konvolüsyon (Conv)	Çekirdek / Adım (Kernel / Stride)
Düşük Seviye	[1, 64, 80, 80]	Conv(64→64)	k=3, s=1
Orta Seviye	[1, 128, 40, 40]	Conv(128→64)	k=1, s=1
Yüksek Seviye	[1, 256, 20, 20]	Conv(256→64)	k=1, s=1

Boyun kısmındaki bir diğer kritik özellik, çoklu çözünürlükteki özellik haritalarının bir araya getirilmesidir. Bu işlem sayesinde, model yalnızca büyük ve belirgin nesnelere değil, aynı zamanda küçük, karmaşık ya da kısmen görünür nesnelere de başarıyla tespit edebilir. Özellikle PANet'in sunduğu yol toplama (path aggregation) mekanizması sayesinde, omurgadan gelen düşük seviyeli özelliklerle derin katmanlardan gelen yüksek seviyeli semantik bilgilerin birleşimi daha dengeli ve güçlü hale gelir.

YOLOv8'in boyun mimarisi, minimal tasarımıyla parametre sayısını artırmadan doğruluğu ve hız dengesini en üst düzeye çıkarmayı hedeflemiştir. Literatürde yapılan karşılaştırmalı analizlerde, bu boyun yapısının hem COCO hem de özel veri kümelerinde üstün performans gösterdiği, özellikle küçük nesne tespitinde önceki YOLO sürümlerine göre daha başarılı sonuçlar verdiği gösterilmiştir.

2.2.4. Kafa

YOLOv8'in kafa (head) bölümü, modelin çıktısının üretildiği son katmandır ve nesnelere sınıflandırılması ile sınırlayıcı kutuların (bounding box) tahmin edilmesinden sorumludur. Bu

bölüm, omurga (backbone) ve boyun (neck) bölümlerinden gelen çok ölçekli ve zenginleştirilmiş özellik haritalarını kullanarak, her bir nesne adayının konumunu (x, y, w, h), sınıf etiketini ve güven skoru gibi bilgileri doğrudan tahmin eder. YOLOv8'in kafa mimarisi, önceki YOLO sürümlerinin geleneksel kafa yapısına dayansa da, optimize edilmiş bir formda sunulmuştur. Özellikle, çıkış katmanlarının sadeleştirilmesi ve daha verimli düzenlenmesi sayesinde hem hız hem de doğruluk açısından önemli iyileştirmeler sağlanmıştır.

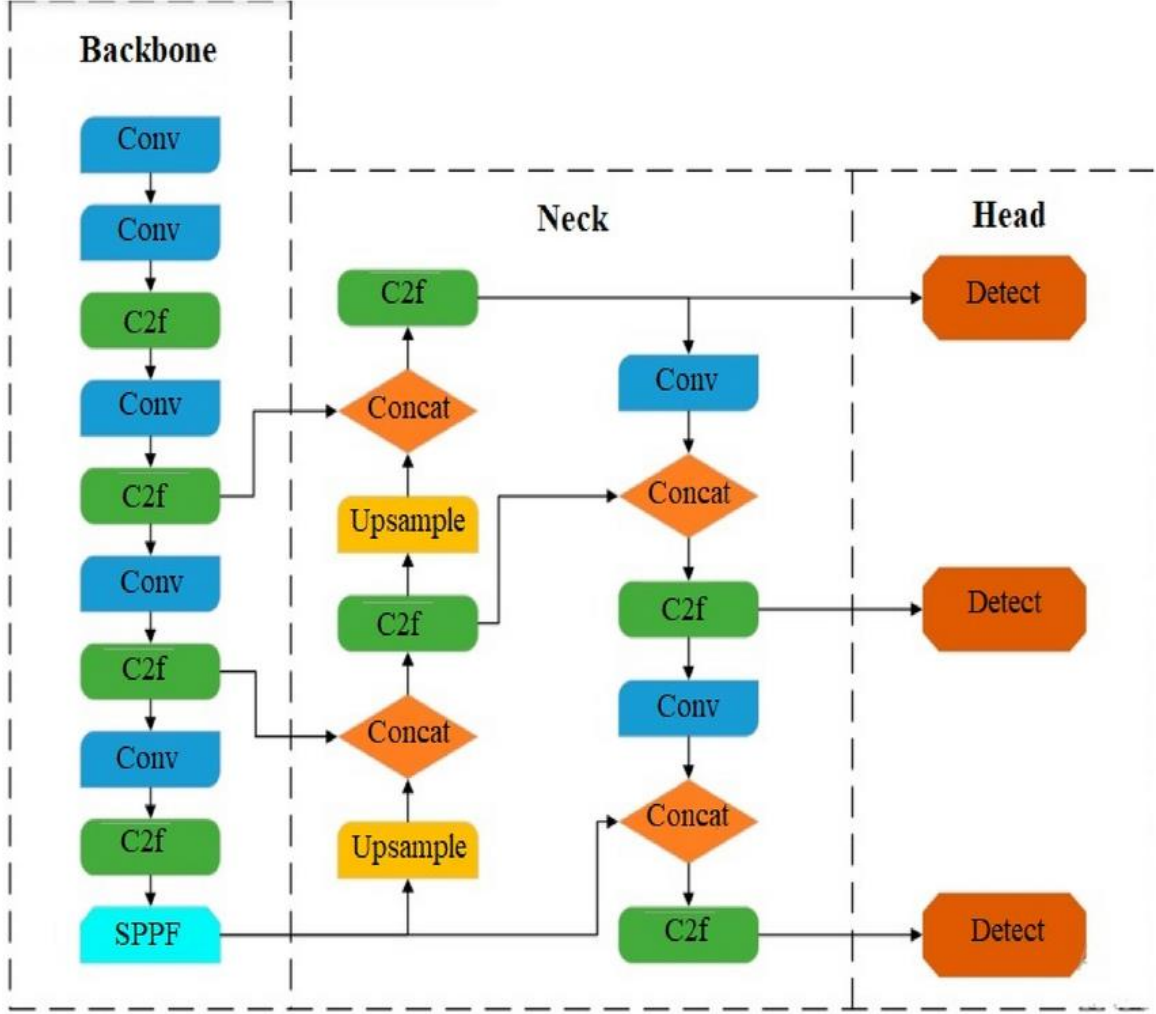
YOLOv8'de kafa kısmı, klasik "dense prediction" yaklaşımını benimser. Yani, her ölçek için belirli bir grid hücresi üzerinde çok sayıda nesne adayları oluşturularak bu adaylar üzerinden tahmin yapılır. YOLOv8, anchor-free bir yapı kullanarak, sabit boyutlu ön tanımlı (anchor) kutular yerine doğrudan nesne merkezine göre regresyon yapar. Bu yapı, anchor tabanlı modellere kıyasla daha az hiperparametre gerektirir ve eğitim sürecini sadeleştirir. Ayrıca, bu yeni anchor-free kafa yapısı, küçük nesnelere daha başarılı sonuçlar verirken, çok sayıda ön tanımlı kutu üretmenin getirdiği hesaplama yükünü de azaltır (Yan vd., 2021).

Modelin çıkışında, her pozisyon için bir sınırlayıcı kutu koordinatları (x, y, genişlik, yükseklik), sınıf olasılıkları ve bir nesne varlık skoru (objectness score) tahmin edilir. Bu tahminler, farklı çözünürlükteki özellik haritaları için ayrı ayrı yapılır ve son aşamada bu çıktılar birleştirilerek non-max suppression (NMS) algoritmasıyla tekrarlayan kutular filtrelenir. YOLOv8, varsayılan olarak "probability-weighted NMS" gibi gelişmiş filtreleme tekniklerini de destekleyerek daha güvenilir sonuçlar sunar.

YOLOv8'in kafa bölümünün sade ama etkili tasarımı, yalnızca sınıflandırma ve konumlandırma görevlerinde yüksek performans sunmakla kalmaz, aynı zamanda dağıtık sistemlerde ve gömülü cihazlarda (edge devices) kullanımını da kolaylaştırır. Literatürde yapılan çeşitli deneylerde, YOLOv8'in kafa yapısının anchor-free oluşunun, hem eğitimi hızlandırdığı hem de küçük nesnelere tespitinde doğruluk oranını artırdığı gözlemlenmiştir.

2.2.5. Yolov8 mimarisi

YOLOv8, gerçek zamanlı nesne algılama alanında son gelişmeleri temel alarak geliştirilen, anchor-free (çapasız), modüler ve verimli bir mimariye sahip bir modeldir. Ultralytics tarafından 2023 yılında açık kaynak olarak sunulan YOLOv8, hem akademik hem de endüstriyel uygulamalarda yaygın olarak kullanılan YOLO ailesinin en gelişmiş üyelerinden biridir. YOLOv8'in mimarisi; omurga (backbone), boyun (neck) ve kafa (head) olmak üzere üç temel bileşenden oluşur. Bu yapı, modelin hem düşük hesaplama kaynaklarıyla hızlı çalışmasını hem de yüksek doğrulukla nesne tespiti yapmasını mümkün kılar (Sohan vd., 2024; Terven & Cordova-Esparza, 2023; X. Wang vd., 2023; Yan vd., 2021). Şekil 2.8'de Yolov8'in mimarisi görülmektedir.



Şekil 2.8 Yolov8 mimarisi

YOLOv8'in omurgası, görüntüden özellik çıkarımı yapan ilk katmandır ve C2f (Concatenate-to-Fusion) modülünü temel alır. Bu modül, ELAN mimarisinden türetilmiş olup, önceki sürümlerdeki C3 bloklarına göre daha az parametreyle daha iyi bilgi akışı sağlar. Omurganın sonunda yer alan SPPF (Spatial Pyramid Pooling - Fast) modülü ise farklı ölçeklerdeki bilgi temsillerini birleştirerek çok boyutlu nesne algılamayı destekler. Bu sayede model, farklı boyutlardaki nesnelere aynı anda tanıyabilir hale gelir (Han vd., 2023; K. He vd., 2015; Xie vd., 2020).

Boyun kısmı, PANet (Path Aggregation Network) mimarisi ile desteklenmiştir. Bu yapı, omurgadan gelen çok ölçekli özellik haritalarını birleştirerek hem yukarıdan aşağıya hem de aşağıdan yukarıya bilgi akışı sağlar. Bu çift yönlü yapı sayesinde, hem düşük seviyeli detaylar hem de yüksek seviyeli semantik bilgiler modelin çıktısına katkıda bulunur. Bu katman, küçük nesnelerin algılanmasında büyük bir avantaj sağlar ve farklı çözünürlüklerdeki verilerin birleştirilmesini mümkün kılar (Liu vd., 2019).

Kafa bölümü ise modelin çıktısını üretir. YOLOv8, anchor-free bir kafa yapısı kullanarak, geleneksel anchor tabanlı yaklaşımlardaki karmaşık hiperparametre gereksinimini ortadan kaldırmıştır. Bu yapı sayesinde doğrudan nesne merkezlerine göre sınırlayıcı kutuların konumu ve nesne sınıfları tahmin edilir. Anchor-free kafa tasarımı, özellikle küçük nesnelerin tespiti ve modelin farklı veri kümelerine uyarlanabilirliği açısından önemli avantajlar sunar. Çıktılar, non-maximum suppression (NMS) veya probability-weighted NMS gibi filtreleme algoritmalarıyla işlenerek son tahminler elde edilir (Hosang & Planck, y.y.; Yan vd., 2021).

Genel olarak YOLOv8'in mimarisi; yüksek doğruluk, düşük gecikme süresi ve modüler yapı hedeflenerek optimize edilmiştir. Bu mimari sadeleştirilmiş blok yapıları, daha az parametreyle daha yüksek performans sağlamaktadır. Ayrıca, YOLOv8 hem nesne algılama hem de segmentasyon gibi görevler için uyarlanabilir, çok amaçlı bir çatı sunar. Model, mobil cihazlar, gömülü sistemler ve bulut tabanlı servisler gibi farklı donanım yapılarına kolayca entegre edilebilmesiyle de dikkat çekmektedir.

2.3. YOLOv11

YOLOv11, Ultralytics tarafından 2024'ün sonlarında duyurulan ve YOLO (You Only Look Once) ailesinin en yeni nesil üyesidir. Bu model, özellikle önceki sürüm olan YOLOv8'e göre daha verimli, modüler ve genişletilebilir bir yapı sunmayı hedeflemektedir. YOLOv11, yüksek doğruluk ve düşük gecikme süresi sağlayan mimarisiyle hem akademik çalışmalarda hem de endüstriyel uygulamalarda dikkat çekmektedir. Henüz resmi bir akademik makalesi yayımlanmamış olsa da, modelin kaynak kodları ve dokümantasyonu Ultralytics'in GitHub deposu üzerinden erişilebilirdir. YOLOv11, açık kaynak kodlu PyTorch tabanlı bir çerçevede geliştirilmiştir ve model tasarımı, modül bazlı özelleştirilebilirlik üzerine kurulmuştur.

YOLOv11'in mimarisi üç temel bölümden oluşur: omurga (backbone), boyun (neck) ve kafa (head). Omurga kısmında, önceki YOLO sürümlerindeki C2f bloklarına benzer ancak daha derinlemesine özellik çıkarımı yapabilen yeni modüller kullanılmaktadır. Bu modüller, ELAN ve EfficientNet gibi yapılardan ilham alarak hem kanal hem de uzamsal boyutlarda daha etkili bilgi işleme sağlar. Bu sayede düşük çözünürlüklü görüntülerde bile küçük nesnelerin özellikleri daha hassas biçimde çıkarılır. SPPF (Spatial Pyramid Pooling – Fast) modülü bu sürümde optimize edilerek bilgi yoğunluğunu daha az parametreyle artıracak şekilde yeniden yapılandırılmıştır (K. He vd., 2015).

Boyun kısmı ise çoklu ölçekli özellik haritalarının birleştirilmesinde önemli bir rol oynar. YOLOv11, burada gelişmiş bir PANet + BiFPN (Bidirectional Feature Pyramid Network) benzeri hibrit yapı kullanır. Böylece, düşük seviye detay bilgileri ile yüksek seviye semantik bilgilerin daha verimli bir şekilde birleştirilmesi sağlanır (Doherty vd., 2024; Liu vd., 2019). Bu özellik, özellikle

çok yoğun sahnelerde veya farklı boyutlardaki nesnelere içeren görüntülerde modelin başarımını artırmaktadır.

YOLOv11'in kafa bölümü anchor-free yapıyı benimsemeye devam eder. Bu, sabit ön tanımlı anchor kutularına dayanmaksızın doğrudan nesne merkezlerine regresyon yapılmasını sağlar. Böylece modelin eğitim süreci sadeleşir, hiperparametre ayarlamaları azalır ve farklı veri kümelerine adaptasyon kolaylaşır. Bu bölümde ayrıca "Task-Aware Decoupled Head" benzeri bir yapı kullanılarak sınıflandırma ve regresyon görevleri ayrı yollarda işlenir. Bu da her görevin kendi özelliklerine göre optimize edilmesini sağlayarak doğruluk oranını artırır.

YOLOv11, dağıtık sistemler ve kenar cihazlarda çalışmaya uygun olacak şekilde optimize edilmiştir. Model, farklı boyutlarda (YOLOv11n, s, m, l, x) sunularak kullanıcılara donanımsal gereksinimlerine göre uygun sürümü seçme esnekliği sağlar. Ayrıca segmentasyon ve pose estimation gibi görevler için de genişletilebilir bir yapıdadır. Ultralytics tarafından sunulan resmi eğitim ve değerlendirme pipeline'ı, kullanıcıların model eğitimi, ince ayar (fine-tuning) ve çıkarım süreçlerini kolayca yönetmesini mümkün kılar.

2.3.1. Omurga

YOLOv11'in omurga (backbone) yapısı, önceki YOLO sürümlerinin mimari mirasını devralarak, özellikle derin özellik çıkarımı ve hesaplama verimliliği açısından önemli iyileştirmeler sunan yenilikçi bir mimariyle inşa edilmiştir. Bu modelde, temel yapı taşları olarak C3k2 (Cross Stage Partial with 2 kernels), C2-PSA (Parallel Spatial Attention) ve optimize edilmiş SPPF (Spatial Pyramid Pooling – Fast) modülleri kullanılmaktadır. C3k2 modülü, önceki CSP ve ELAN yapılarından türetilmiş olup, ağ boyunca bilgi kaybını minimize eden çapraz aşama kısmi bağlantılar ile derin katmanlardaki temsiliyet gücünü artırırken, modelin parametre verimliliğini korur. C2-PSA ise uzamsal dikkat mekanizmasını paralel bir şekilde işler; bu da, düşük seviyeli konumsal bilgilerle yüksek seviyeli semantik bilgilerin daha dengeli ve anlamlı birleştirilmesine olanak tanır. SPPF modülü ise çok ölçekli bağlam bilgisini hızlı ve parametrik olarak verimli biçimde yakalayarak özellikle küçük nesne tespiti gibi görevlerde modelin başarımını artırır (Nenhancements, 2024).

YOLOv11 omurgasında dikkat çeken bir diğer unsur da GELAN (Generalized ELAN) ve ELAN yapılarına yapılan iyileştirmelerdir. Bu bloklar, kanal bölünmesi ve paralel yapıların etkili kullanımı sayesinde ağ boyunca bilgi akışını yoğunlaştırmakta ve derinlik/karmaşıklık dengesini optimize etmektedir. Bu tasarım tercihleri sayesinde, YOLOv11'in orta seviye modelleri (örneğin YOLOv11-m) önceki sürümlere kıyasla %20-25 civarında daha düşük parametre sayısı ile benzer veya daha yüksek doğruluk (mAP) değerlerine ulaşabilmektedir. Ayrıca omurga yapısı, Swin Transformer, EfficientNet ve ResNet gibi alternatif ağlarla kolayca entegre edilebilecek şekilde

modüler olarak yapılandırılmıştır. Bu durum, YOLOv11'i farklı donanım gereksinimlerine sahip uygulamalar için yüksek derecede özelleştirilebilir kılmaktadır.

Sonuç olarak, YOLOv11'in omurgası, sadece gelişmiş özellik çıkarımı ile değil, aynı zamanda dikkatli parametrik optimizasyon, dikkat mekanizmaları ve modüler tasarımı ile çağdaş derin öğrenme mimarileri arasında öne çıkmaktadır. Henüz resmi akademik bir yayımla belgelenmemiş olsa da, topluluk tarafından yapılan analizler ve açık kaynak kod incelemeleri, bu omurganın hem araştırma hem de endüstriyel düzeyde yüksek potansiyel taşıdığını ortaya koymaktadır.

2.3.2. Boyun

YOLOv11 modelinin boyun (neck) bölümü, çok ölçekli özelliklerin etkin biçimde birleştirilmesi ve ileri düzey bağlamsal temsili artırılması amacıyla yeniden yapılandırılmış, dikkat mekanizmaları ve optimize edilmiş füzyon modülleriyle zenginleştirilmiştir. Önceki YOLO sürümlerinde yer alan PANet (Path Aggregation Network) ve SPP (Spatial Pyramid Pooling) gibi temel bileşenlerin yerini, YOLOv11'de daha esnek, daha hafif ve daha verimli yapılara bırakmıştır. Bu doğrultuda, modelin boyun kısmında C2f (Concatenate-Conv fusion), C2-PSA (Parallel Spatial Attention) ve GELAN bağlantı yapıları gibi modern modüller kullanılmaktadır. Bu modüller, derinlikten gelen semantik bilgileri yüzeysel konumsal bilgilerle birleştirerek, özellikle küçük nesnelerin tespitinde doğruluğu artırmayı amaçlar.

C2-PSA modülü, paralel dikkat yolları ile dikkatli bir özellik füzyonu sağlar. Özellikle kanal ve uzamsal dikkat mekanizmalarının paralel olarak uygulanması sayesinde, ağırlıklı olarak hangi bölgelere odaklanması gerektiği daha yüksek doğrulukla öğrenilir. Bu yapı, PANet'in geleneksel yukarı ve aşağı yönlü bilgi akışı yaklaşımını daha rafine ve bilgi kaybını azaltacak biçimde yeniden yorumlanmaktadır. C2f yapısı ise kanal birleştirme ve 1x1 konvolüsyon katmanlarıyla hafif, modüler bir bilgi transferi sağlar; bu durum, hesaplama yükünü azaltırken özelliklerin bilgi yoğunluğunu korur. Boyun bölümünde kullanılan GELAN bağlantıları ise derin ağ katmanlarının birbirleriyle daha etkili etkileşime girmesini sağlayarak, çok ölçekli bağlamların aynı anda değerlendirilmesini mümkün kılar.

Bu yeni boyun yapısı, sadece doğruluğu değil, aynı zamanda modelin gerçek zamanlı nesne algılama kabiliyetini de doğrudan olumlu yönde etkilemektedir. Yapılan açık kaynak testlerde YOLOv11'in boyun kısmı, COCO gibi zorlu veri setlerinde önceki sürümlere kıyasla daha yüksek mean Average Precision (mAP) skorları ve daha düşük gecikme süreleri ile sonuç vermiştir. Ayrıca, bu boyun mimarisi farklı backbone ve head yapılarıyla yüksek uyumluluk göstererek, mimarinin esnekliğini ve yeniden kullanılabilirliğini artırmaktadır. Sonuç olarak, YOLOv11'in boyun bölümü, ileri düzey dikkat mekanizmaları ve optimize edilmiş çok ölçekli özellik füzyonu sayesinde hem doğruluk hem de hız açısından dengeli ve güçlü bir yapı sunar.

2.3.3. Kafa

YOLOv11 modelinin kafa (head) bölümü, nesne tespitinde sınırlayıcı kutuların koordinatları ve sınıf olasılıklarının tahmin edilmesi amacıyla optimize edilmiştir. Bu katman, çok ölçekli özellik haritalarından (örneğin 80×80 , 40×40 , 20×20) çıkarım yapılmasına olanak tanıyarak küçükten büyüğe farklı nesne boyutlarının eş zamanlı olarak algılanmasını mümkün kılar. Geleneksel anchor-tabanlı yöntemlerin yanı sıra anchor-free yaklaşımlar da desteklenerek yapının esnekliği ve eğitilebilirliği artırılmıştır.

Anchor-free yapılar sayesinde, her piksel konumunda nesne varlığı doğrudan tahmin edilmiş; bu sayede model karmaşıklığı azaltılmış ve eğitim süresi kısaltılmıştır. Kayıp fonksiyonu olarak IoU-tabanlı metrikler (CIoU, DIoU, SIoU) entegre edilerek sınırlayıcı kutuların yalnızca konumu değil, boyut ve hizası da optimize edilmiştir. Çıkış öncesi uygulanan hafif konvolüsyonel bloklar ve dikkat mekanizmaları ile özniteliklerin anlamlı bölgeleri vurgulanmış; bu da sınıflandırma ve lokalizasyon doğruluğunu artırmıştır.

Kafa bölümünün modüler yapısı sayesinde segmentasyon ve duruş tahmini (pose estimation) gibi görevlerin de kolayca entegre edilmesi sağlanmıştır. Böylece model yalnızca doğruluk açısından değil, aynı zamanda görev çeşitliliği ve hesaplama verimliliği bakımından da avantajlı hâle getirilmiştir. Sonuç olarak, YOLOv11'in kafa mimarisi, düşük gecikme süresi ve yüksek doğruluk sunan, modern nesne algılama ihtiyaçlarına yanıt verebilen etkin bir yapı olarak tasarlanmıştır.

2.4. YOLO Mimarilerinin Hepsinde Orta Bulunan Özellikler

2.4.1. Sınırlayıcı Kutular

Nesne algılama modellerinde görüntülerdeki nesnelere dikdörtgen kutularla sınırlanır. Üç farklı ölçekte birden çok bağlantı kutusu, YOLOv4 tarafından nesne algılama için kullanılır. Görüntüdeki nesnelerin şekillerine ve boyutlarına yaklaşık olarak uyacak şekilde önceden tanımlanmış kutular bağlantı kutularıdır. Her bağlantı kutusu için bir sınırlayıcı kutu ve bir sınıf tahmini, YOLOv4 tarafından üretilir. Nesnenin merkez koordinatları (x,y) , genişliği (w) ve yüksekliği (h) şeklinde ifade edilen sınırlayıcı kutular, nesnenin olasılık değerleriyle birlikte hangi sınıfa ait olduğunu belirten sınıf tahminleriyle birlikte verilir.

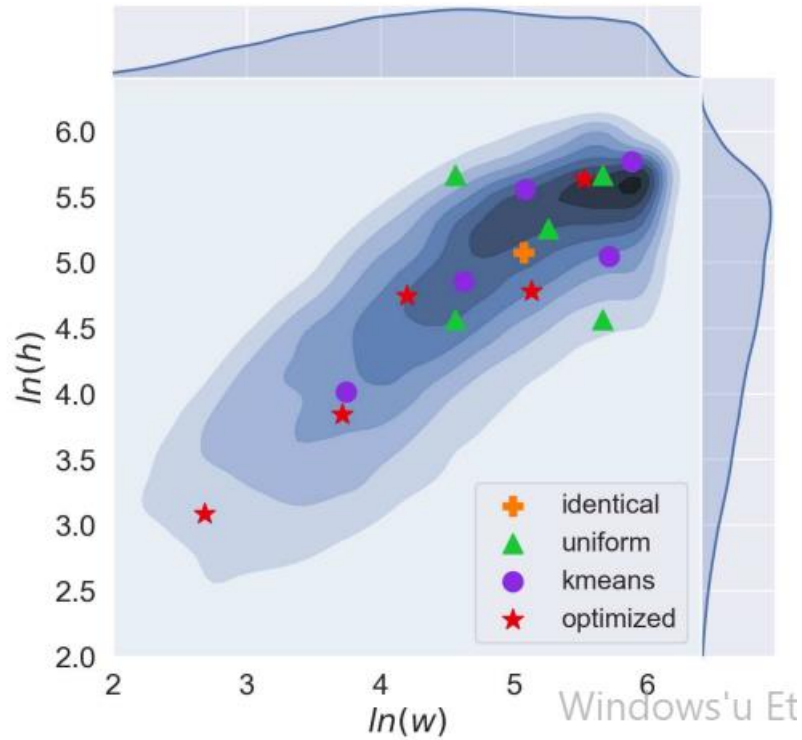
Birden fazla sınırlayıcı kutu, YOLOv4 tarafından üretildiği için, aynı nesneye denk gelen veya güven düzeyi düşük olan kutular Greedy-NMS adlı bir teknikle elenir (Hosang & Planck, y.y.). Güven düzeyi en yüksek olan kutu Greedy-NMS tarafından seçilir ve onunla yüksek oranda kesişen diğer kutular atılır. Böylece, her nesne için tek bir bounding box elde edilir. Şekil 2.9'da sınırlayıcı kutular örnekleri verilmiştir (Y. He vd., 2019).



Şekil 2.9. MSCOCO veri setinden sınırlayıcı kutu örnekleri gösterimi (Colleges vd., 2014)

2.4.2. Bağlantı kutusu

Nesne algılama modellerinde dikkörtgen kutular görüntüler üzerinde yoğun bir şekilde önerilir ve kutuların konumunu ve sınıflandırma güvenini tahmin etmek için ağ tarafından eğitilir. Bağlantı kutuları, görüntüdeki nesnelerin şekillerine ve boyutlarına yaklaşık olarak uyacak şekilde önceden tanımlanır. Birden fazla nesne algılama tekniği tarafından bağlantı kutularını kullanılır. Örneğin, bağlantı kutuları üretmek için YOLOv4 tarafından üç farklı ölçekte birden çok bağlantı kutusu kullanılır. Aynı nesneye denk gelen veya güven düzeyi düşük olan kutular Ağgözlü-NMS adlı bir teknikle bağlantı kutuları tarafından elenir. Nesne algılama performansını ve verimliliğini artırmak için önemli bir rol bağlantı kutuları tarafından sağlanır. Şekil 2.10'da bağlantı kutusu örneği verilmiştir (Y. Zhong vd., 2020).



Şekil 2.10. Bağlantı kutusu veri setindeki ln(genişlik) ve ln(yükseklik) marjinal dağılımları gösterilmi (Y. Zhong vd., 2020)

2.4.3. Açgözlü-Maksimum Olmayanı Bastırma

Çalışmada, nesne algılama için son işlem olarak kullanılan Açgözlü-Maksimum Olmayan Bastırma algoritması incelenmiştir. Açgözlü-MOB algoritması, aynı sınıfa ait çok sayıda sınırlayıcı kutu arasından en yüksek nesnelik skoruna sahip olanı seçmek için kullanılan bir algoritmadır. Açgözlü-MOB algoritması, sabit bir IoU eşiği ile açgözlü kümeleme yapar ve bu da hatırlama ve kesinlik arasında bir denge kurmayı gerektirir. Açgözlü-MOB algoritmasının avantajları ve dezavantajları literatürde tartışılmıştır. Ayrıca, Açgözlü-MOB algoritmasının performansını iyileştirmek için önerilen bazı yöntemler de ele alınmıştır. Örneğin, NMS işlevini parametrik bir fonksiyon olarak tanımlayarak öğrenilebilir hale getirmek, farklı IoU eşikleri kullanarak adaptif kümeleme yapmak veya kesişim alanını hesaplamak yerine mesafe ölçütleri kullanmak gibi yöntemler literatürde mevcuttur. Bu çalışmada, bu yöntemlerin etkinliği ve uyumluluğu farklı nesne algılama modelleri ve veri kümeleri üzerinde değerlendirilmiştir (Hosang & Planck, y.y.).

2.4.4. Havuzlama Katmanı

Maksimum havuzlama, evrişimli sinir ağlarında kullanılan bir örnekleme küçültme tekniğidir. Maksimum havuzlama, girdi özellik haritasını daha küçük boyutlu bir çıktı özellik haritasına dönüştürmek için belirli bir boyuttaki bir pencereyi kaydırarak çalışır. Her pencere için, pencerenin kapsadığı piksellerin maksimum değeri çıktı özellik haritasına aktarılır. Bu şekilde, maksimum havuzlama, girdi özellik haritasının boyutunu azaltırken, en önemli özellikleri korur. Maksimum havuzlama, ağırlık hesaplama karmaşıklığını azaltmaya, aşırı uyumu önlemeye ve yerel çevresel değişikliklere karşı dayanıklılık sağlamaya yardımcı olur (Akhtar & Ragavendran, 2020). Şekil 2.11'de maksimum havuzlama yöntemi görülmektedir.

58	47	29	17
44	35	32	14
51	42	35	22
40	19	15	36

58	32
51	36

Max Pooling

Şekil 2.11. Maksimum havuzlama metodu (Akhtar & Ragavendran, 2020)

2.4.5. Uzamsal Dikkat Modülü

Nesne algılama modellerinin performansının artırılması için tasarlanmış bir mekansal dikkat mekanizmasıdır. Uzamsal dikkat modülü, girdi özelliği haritasının her konumunda bir dikkat ağırlığı hesaplanmasını ve bunun özelliğe çarpılmasını sağlayan bir konvolüsyonel katmandan

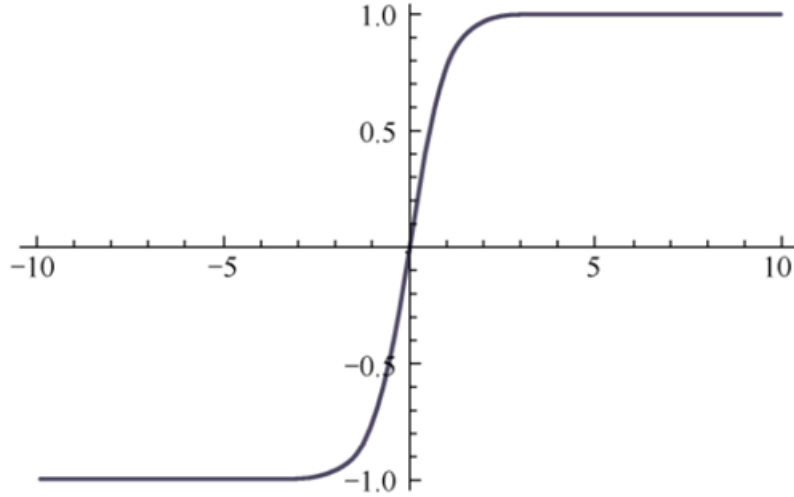
oluşur. Modelin önemli bölgelere dikkatinin yoğunlaştırılması ve arka plan gürültüsünün bastırılmasıdır. SAM, yolov4 gibi modern nesne algılama modellerinde başarıyla uygulanmıştır (Bochkovskiy vd., 2020).

2.4.6. Aktivasyon Fonksiyonu

Aktivasyon fonksiyonları, derin öğrenme modellerinin önemli bir parçasıdır ve genellikle bir yapay sinir hücresinin çıktısını kontrol etmek için kullanılır. Temelde, aktivasyon fonksiyonu bir sinir hücresinin, toplam girdi sinyalinin belirli bir eşiği aşmış aşmadığını ve sonuç olarak bir çıkış sinyali üretip üretmeyeceğini belirler. Bu, sinir hücrelerinin, bir tür kontrol mekanizması aracılığıyla 'uyanık' olup olmadığını belirleyen bir işlevi vardır (Lederer, 2021).

2.4.7. Tanh Aktivasyon Fonksiyonu

Tanh veya hiperbolik tanjant fonksiyonu, aktivasyon fonksiyonlarından biri olup, sigmoid fonksiyonunun genişletilmiş bir versiyonudur ve çıkışını -1 ve 1 arasında bir değere dönüştürür. Bu, ağırlıkların daha hızlı ve etkili bir şekilde ayarlanmasına olanak sağlar. Tanh fonksiyonu sigmoid fonksiyonuna kıyasla daha geniş bir çıktı aralığına sahip olduğundan, gradyanlar genellikle daha büyük olur ve bu da ağırlıkların daha hızlı güncellenmesine yardımcı olur (Dubey vd., 2022). Şekil 2.12'de Tanh aktivasyon fonksiyonu görülmektedir. Denklem 2.1'de Tanh aktivasyon fonksiyonunun matematiksel ifadesi verilmiştir.

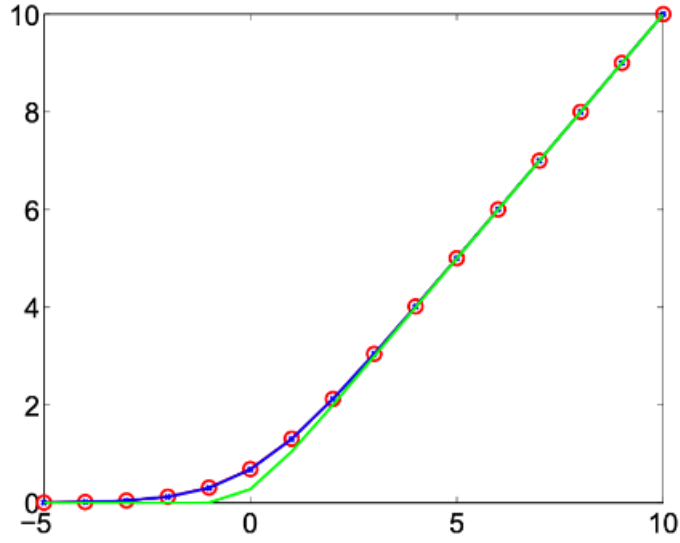


Şekil 2.12. Tanh aktivasyon fonksiyonu (Dubey vd., 2022)

$$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.1)$$

2.4.8. Softplus Aktivasyon Fonksiyonu

Softplus, başka bir önemli aktivasyon fonksiyonudur. Softplus fonksiyonu, ReLU (Düzeltilmiş Doğrusal Birim) fonksiyonunun daha düzgün bir versiyonudur ve çıktı değerlerinin daha geniş bir aralığa yayılmasını sağlar. Softplus, negatif girdileri sıfıra yakın küçük pozitif değerlere dönüştürür, böylece aktivasyon sırasında daha fazla bilgi korunur. Sürekli türevlenebilir olması, belirli uygulamalar için daha uygundur (Howard & Gugger, 2020). Şekil 2.13'de Softplus aktivasyon fonksiyonunun grafiği görülmektedir. Softplus aktivasyon fonksiyonunun matematiksel formülü Denklem 2.2'de ifade edilmiştir.



Şekil 2.13. Softplus aktivasyon fonksiyonu (Howard & Gugger, 2020)

$$f(x) = \log(1 + \exp(x)) \quad (2.2)$$

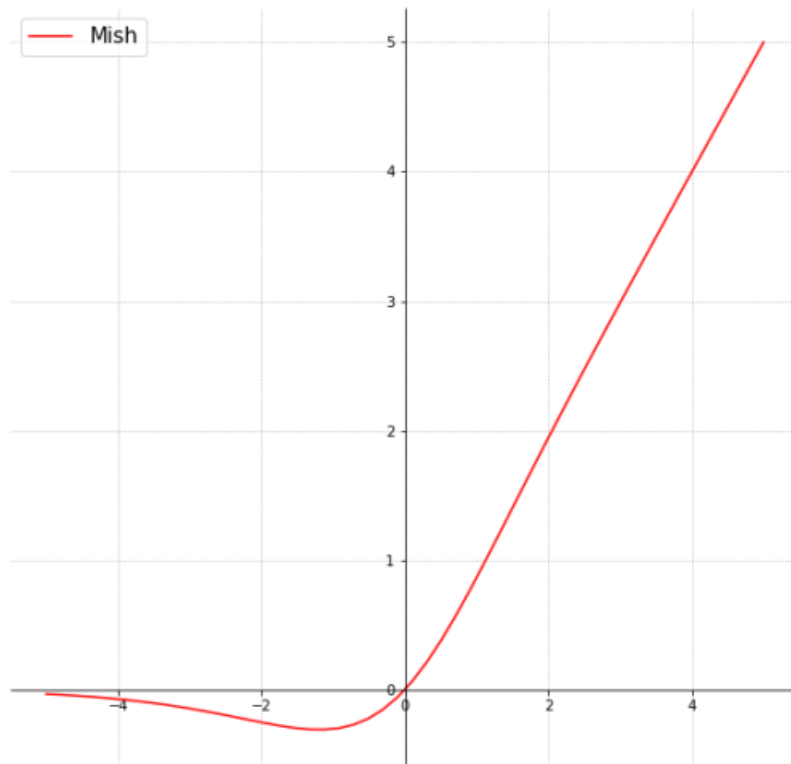
2.4.9. Mish Aktivasyon Fonksiyonu

Mish, kendi kendine düzenleyici bir özelliğe sahip olan ve çeşitli derin öğrenme görevlerinde etkili olan bir aktivasyon fonksiyonudur. Mish, softplus fonksiyonunun bir çıktısını tanh fonksiyonuna verir ve böylece iki fonksiyonun özelliklerini birleştirir. Bu, öğrenme sürecinde gradyanların daha iyi korunmasını ve daha iyi genelleme performansı elde edilmesini sağlar. YOLOv4'te, Mish aktivasyon fonksiyonu kullanılır ve bu, modelin hem hızını hem de doğruluğunu artırır. Mish, diğer aktivasyon fonksiyonlarına kıyasla daha karmaşık bir fonksiyondur. Bu karmaşıklık, gelişmiş gradyan dağılımı ve daha iyi genelleme kabiliyeti gibi avantajlarla karşılanır.

Mish'in etkinliđi ve verimliliđi, özellikle ađırlıklı toplamların (weighted sums) aşırı negatif veya pozitif olduđu durumlarda belirgindir. Mish, bu durumlarda bile aktivasyonu düzgün bir şekilde yöneterek bilgi kaybını minimize eder. Bu özellik, derin sinir ađırlarının eğitim sürecinde çok önemlidir çünkü gradyanların düzgün bir şekilde geri yayılımını (backpropagation) sağlar.

Mish, aynı zamanda kendi kendini düzenleyici bir özelliđe sahiptir. Bu, ađırlıkların ve öğrenme hızının otomatik olarak ayarlanması anlamına gelir. Bu özellik, derin öğrenme modellerinin eğitim sürecini büyük ölçüde basitleştirir ve modelin genel performansını iyileştirir.

YOLOv4 gibi modern derin öğrenme algoritmaları, Mish'in bu özelliklerinden önemli ölçüde yararlanır. Mish'in hem hızı hem de doğruluđu artırma kapasitesi, YOLOv4'ün etkinliđini ve genel performansını büyük ölçüde artırır. Bu nedenle, Mish, YOLOv4 dahil olmak üzere çeşitli modern derin öğrenme uygulamalarında yaygın olarak kullanılan bir aktivasyon fonksiyonudur (Misra, 2019). Şekil 2.14'de Mish aktivasyon fonksiyonu verilmiştir. Denklem 2.3'de Mish aktivasyon fonksiyonunun matematiksel ifadesi yer almaktadır.



Şekil 2.14. Mish aktivasyon fonksiyonu (Misra, 2019)

$$f(x) = x * \tanh(\text{softplus}(x)) \quad (2.3)$$

2.4.10. Normalizasyon

Normalizasyon, bir dizi verinin ölçeği standartlaştırılan bir işlem veya dizi işlemlerdir. Makine öğrenmesinde, normalizasyon genellikle model performansı artırılan ve hızlandırılan bir işlem olarak kullanılır. Normalizasyon ile algoritma, aynı ölçeğe sahip olmayan özellikler arasındaki ilişki daha iyi anlaşılır. Normalizasyon işlemi genellikle her özelliğin ortalama değeri çıkarılır ve standart sapmaya bölünür. YOLOv4 (You Only Look Once version 4), gerçek zamanlı nesne algılama için kullanılan en son ve en güçlü sistemlerden biridir. YOLOv4'te normalizasyon, modelin performansı ve doğruluğu artırılan bir işlem olarak kullanılır.

2.4.11. Küme Normalleştirme

Küme normalleştirme (KN) derin sinir ağlarının eğitimi hızlandıran ve stabilize edilen bir tekniktir. Bir kümedeki tüm örnekler aynı anda normalize edilir. KN ile veri setindeki her bir kümede bulunan verilerin ortalama değeri çıkarılır ve ardından verilerin varyansı ile ölçeklenir. Bu, modelin öğrenmesi hızlandırır ve daha derin ağların eğitimi kolaylaştırır. KN'nin nasıl çalıştığı Denklem 2.4'deki formülde verilmiştir.

$$x' = \gamma x - \mu\sigma + \beta \quad (2.4)$$

Burada x , girdi; x' , normalize edilmiş girdi; μ , batch ortalaması; σ , batch standart sapması; γ , ölçekleme parametresi; β , kaydırma parametresi (Bjorck vd., 2018).

2.4.12. Çapraz Mini Küme Normalizasyonu

YOLOv4'de küçük mini-küme boyutlarında KN'nin azalan etkinliğini çözmek için kullanılan bir tekniktir. CmKN ile birden fazla son iterasyondan örnekler birlikte kullanılır ve ağ ağırlıklarındaki değişiklikler Taylor polinomlarına dayalı bir teknik ile telafi edilir. Böylece istatistikler doğru bir şekilde tahmin edilir ve BN etkili bir şekilde uygulanır. CmKN'nin kullanılması, modelin öğrenmesini hızlandırır ve daha derin ağların eğitimini kolaylaştırır (Yao & Lin, y.y.).

2.5. Karışıklık matrisi

2.5.1. Doğruluk

Sınıflandırma problemlerinde kullanılan en yaygın performans metriklerinden biridir. Doğruluk, doğru sınıflandırılan örneklerin toplam örnek sayısına bölünmesiyle hesaplanır.

Doğruluk, Denklem 2.5'te ifade edilmiştir. Doğruluk, modelin genel doğruluğunu ölçmek için kullanılır (Howard & Gugger, 2020).

$$\frac{TP + TN}{TP + FP + FN + TN} \quad (2.5)$$

2.5.2. Kesinlik

Tahmin edilen pozitif değerlerin gerçek pozitif değerlere oranıdır. Precision, tahmin ettiğimiz pozitif değerlerin kaç tanesinin gerçekten pozitif olduğunu gösterir. Kesinlik, yanlış pozitif tahminlerin maliyetinin yüksek olduğu durumlarda önemli bir metriktir (Grandini vd., 2020).

Kesinlik, Denklem 2.7'de görülen matematiksel formülle ifade edilmektedir.

$$\frac{TP}{TP + FP} \quad (2.6)$$

2.5.3. Duyarlılık

Sınıflandırma problemleri için kullanılan bir performans metriğidir. Duyarlılık, gerçek pozitif değerlerin tahmin edilen pozitif değerlere oranını ölçer (Grandini vd., 2020). Duyarlılık, Denklem 2.7'de verilmiştir.

$$\frac{TP}{TP + FN} \quad (2.7)$$

Duyarlılık, modelin pozitif sınıfı ne kadar iyi yakaladığını gösterir. Duyarlılık değeri 1'e yaklaştıkça, modelin performansı daha iyidir. Duyarlılık, yanlış negatif tahminlerin maliyetinin yüksek olduğu durumlarda önemli bir metriktir (Grandini vd., 2020).

2.5.4. F-Skor

Sınıflandırma problemleri için kullanılan bir performans metriğidir. F-Skor, kesinlik ve duyarlılık değerlerinin harmonik ortalamasıdır. F-Skor kesinlik ve duyarlılık arasında bir denge sağlamak için kullanılır (Grandini vd., 2020; Hand vd., 2021). F-Skor Denklem 2.8'de görülmektedir.

$$\frac{2 * Kesinlik * Duyarluluk}{Kesinlik + duyarlılık} \quad (2.8)$$

2.5.5. Özgüllük

Sınıflandırma problemleri için kullanılan bir performans metriğidir. Özgüllük, gerçek negatif değerlerin tahmin edilen negatif değerlere oranını ölçer. Özgüllük, gerçekte negatif olan değerlerin kaç tanesinin doğru tahmin edildiğini gösterir. Özgüllük, Denklem 2.9'da verilmiştir.

$$\frac{TN}{TN + FP} \quad (2.9)$$

Özgüllük, modelin negatif sınıfı ne kadar iyi ayırt ettiğini gösterir. Özgüllük değeri 1'e yaklaştıkça, modelin performansı daha iyidir. Özgüllük, yanlış pozitif tahminlerin maliyetinin yüksek olduğu durumlarda önemli bir metriktir (Grandini vd., 2020).

2.5.6. Eğrinin Altındaki Alanın Alıcı İşletim Karakteristiği

Sınıflandırma problemleri için kullanılan bir performans metriğidir. AUC-ROC, ROC eğrisinin altında kalan alanın oranını ölçer. ROC eğrisi, modelin farklı eşik değerlerine göre gerçek pozitif oranı (TPR) ve yanlış pozitif oranı (FPR) arasındaki ilişkiyi gösteren bir grafikdir. AUC-ROC, modelin sınıfları ne kadar iyi ayırt ettiğini gösterir. AUC-ROC değeri 1'e yaklaştıkça, modelin performansı daha iyidir. AUC-ROC, sınıflar arasında dengesizlik olduğunda veya yanlış tahminlerin maliyeti farklı olduğunda, doğruluk gibi diğer metriklerden daha iyi bir ölçü olabilir. AUC, ROC eğrisinin altında kalan alanı ifade eder. AUC, modelin sınıflandırma performansını tek bir sayı ile özetlemek için kullanılır. AUC değeri 0 ile 1 arasında değişir. AUC değeri 0.5 ise, modelin sınıflandırma performansı rastgeledir. AUC değeri 1 ise, modelin sınıflandırma performansı mükemmeldir. AUC değeri 0 ise, modelin sınıflandırma performansı tamamen yanlıştır (Yang & Ying, 2023).

2.5.7. Logaritmik Kayıp

Logaritmik kayıp, Denklem 2.10'da ifade edilmektedir. Burada y gerçek değer, p tahmin edilen olasılıktır. Log-Loss, modelin sınıflandırma performansını tek bir sayı ile özetlemek için kullanılır. Log-Loss değeri 0'a yaklaştıkça, modelin performansı daha iyidir. Log-Loss değeri arttıkça, modelin performansı daha kötüdür. Log-Loss, modelin yanlış tahminlerin olasılıklarını minimize

etmesini sağlar. Log-Loss, verilerin dengesiz olduğunda veya yanlış tahminlerin maliyeti farklı olduğunda, doğruluk gibi diğer metriklerden daha iyi bir ölçü olabilir (Vovk, 2015).

$$-(y * \log(p) + (1 - y) * \log(1 - p)) \quad (2.10)$$



3. BULGULAR

3.1. FOD-RUNWAY veri seti

YOLOv5s, YOLOv8s ve önerilen YOLOv11-s modellerinin karşılaştırmalı analizi; hassasiyet (precision), duyarlılık (recall), F1-skora ve farklı IoU eşiklerinde ortalama Ortalama Doğruluk (mAP) gibi temel performans metrikleri baz alınarak gerçekleştirilmiştir. Deneysel değerlendirmeler sonucunda elde edilen performans değerleri Tablo 3.1’de özetlenmiştir.

Tablo 3.1. FOD-RUNWAY veri setine elde edilen sonuçlar.

Metrikler	YOLOv5s	YOLOv8s	YOLOv11s
Hassasiyet	0.925	0.924	0.899
Duyarlılık	0.699	0.842	0.826
F1-Skor	0.796	0.881	0.861
mAP@50	0.839	0.925	0.895
mAP@50-95	0.625	0.925	0.913

YOLOv11-s modeli, bu çalışmada geliştirilip eğitilmiş olup, tüm değerlendirilen metriklerde dengeli ve sağlam bir performans sergilemiştir. Hassasiyet açısından YOLOv5s (0.925) ve YOLOv8s’e (0.924) kıyasla biraz daha düşük (0.899) olsa da, duyarlılıkta (0.826’ya karşı 0.699) ve F1-skorunda (0.861’e karşı 0.796) YOLOv5s’i kayda değer biçimde geride bırakmıştır. Bu durum, YOLOv11-s’in yanlış negatifleri minimize etmede daha etkili olduğunu göstermektedir ki bu, özellikle pistteki FOD tespiti gibi güvenlik açısından kritik uygulamalarda büyük önem taşımaktadır.

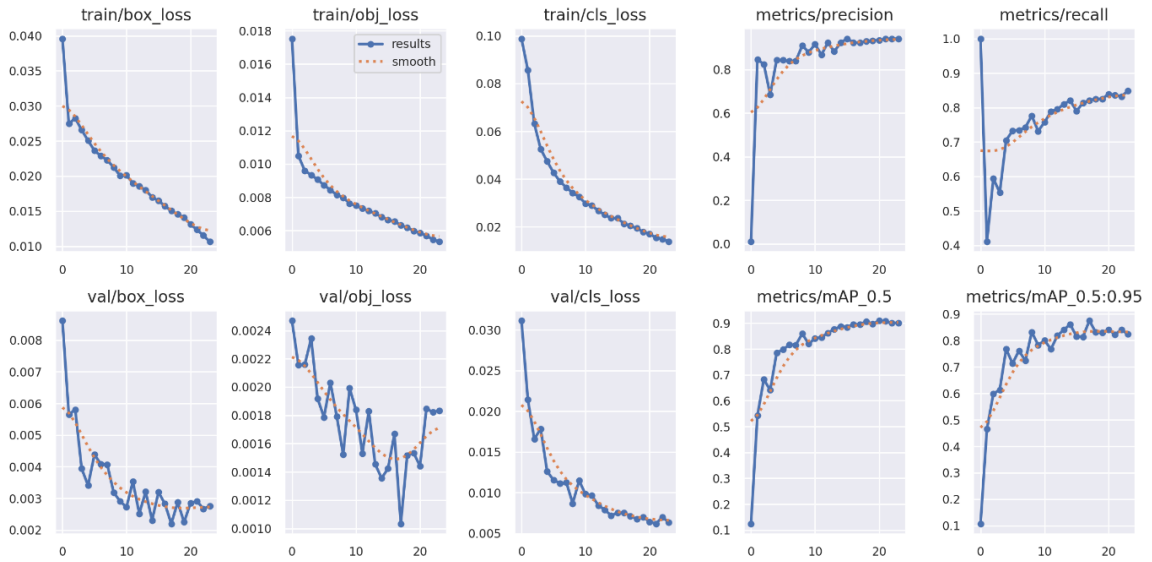
YOLOv8s ile karşılaştırıldığında ise, YOLOv11-s modeli özellikle duyarlılık ve F1-skoru bakımından rekabetçi sonuçlar vermiş, aynı zamanda daha hafif bir mimari yapıya sahiptir (9.45 milyon parametreye karşılık YOLOv8s yaklaşık 11.2 milyon parametre). mAP@0.50 ve mAP@0.50-0.95 değerlerinde YOLOv8s biraz daha yüksek performans göstermesine rağmen aradaki fark minimal düzeydedir; bu da YOLOv11-s’in tespit doğruluğu ile hesaplama maliyeti arasında verimli bir denge sunduğunu ortaya koymaktadır.

Ayrıca, YOLOv11-s modeli orta seviye özelliklere sahip bir L4 GPU’da (4.2 GB bellek) yaklaşık 45 dakikada eğitimi tamamlamış, bu da donanım kaynaklarının kısıtlı olduğu gerçek zamanlı uygulamalar için uygunluğunu vurgulamaktadır. Bu verimlilik ve rekabetçi performans, YOLOv11-s’i özellikle havalimanlarında gömülü FOD izleme sistemleri için güçlü bir aday haline getirmektedir.

Sonuç olarak, bulgular YOLOv11-s'in hem geleneksel nesne algılama sistemlerine hem de daha güncel derin öğrenme modellerine karşı maliyet etkin ve yüksek performanslı bir alternatif sunduğunu doğrulamaktadır. Modelin FOD sınıflarında güçlü genelleme kabiliyeti ve düşük çıkarım gecikmesi, ölçeklenebilir ve gerçek zamanlı pist güvenliği uygulamalarındaki potansiyelini desteklemektedir. İlerleyen çalışmalarda YOLOv11'in daha büyük varyantları (m, l, x) üzerinde yapılacak eğitimlerin, tespit doğruluğunu artırarak YOLOv8 tabanlı modellerle aradaki performans farkını kapatması beklenmektedir.

3.2. YOLOv5 için detaylı sonuçlar

Yolov5'in çıktıları ve sonuçları Şekil 3.1 ve Tablo 3.2'de gösterilmiştir. Şekil 3.1'de YOLOv5 modeli üzerinde gerçekleştirilen eğitim ve doğrulama süreçlerine ilişkin olarak, loss (kayıp) grafiklerini ve precision (kesinlik), recall (duyarlılık) ile mAP gibi performans kriterlerini göstermektedir.



Şekil 3.1. Yolov5 modelinin eğitim çıktıları

Özellikle, %50 IoU (Intersection over Union) eşik değerinde hesaplanan sonuçlar mAP50 ile ifade edilirken; %50 ile %95 arasında değişen farklı IoU eşiklerinde elde edilen ortalama değerler mAP50-95(B) olarak belirtilmektedir. Grafiklerde görüldüğü üzere, modelin kayıp eğrileri yaklaşık 24. epoch civarında dengelenme eğilimi göstermektedir. Bu nedenle, aşırı öğrenmenin (overfitting) önüne geçmek amacıyla eğitim süreci bu noktada durdurulmuştur. Eğitim sonuçları Şekil 3.1'de gösterilmiştir.

Tablo 3.2. Yolov5 mimarisinin alt modülleri üzerindeki sonuçları

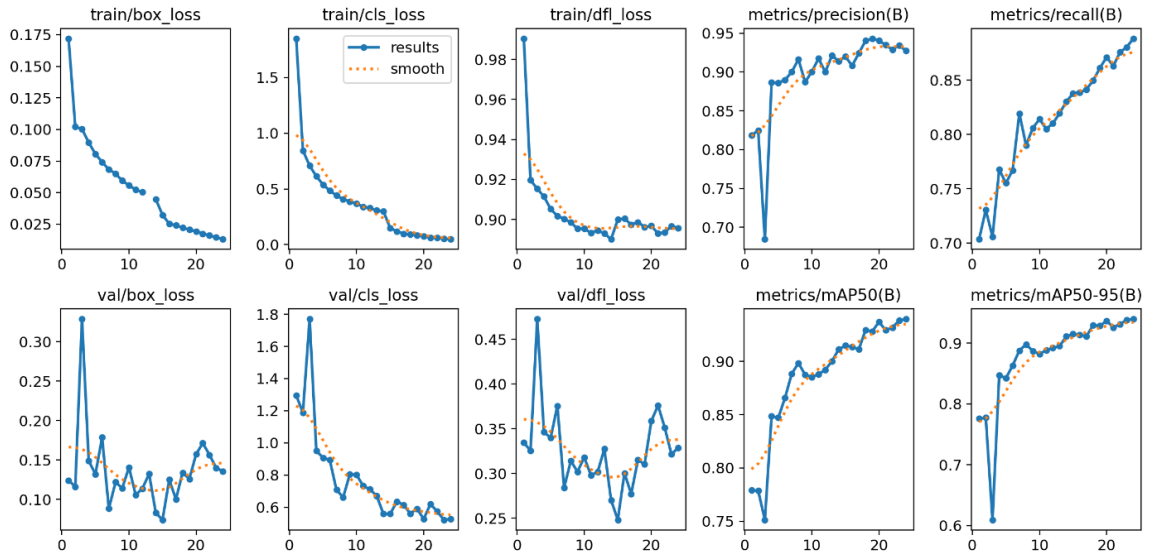
Metrikler	YOLOv5s	YOLOv5m	YOLOv5l	YOLOv5x
Kesinlik Oranı	0.925	0.955	0.915	0.942
Duyarlılık Oranı	0.699	0.841	0.822	0.848
F1-Skor	0.796	0.894	0.866	0.892
mAP50	0.839	0.911	0.883	0.901
mAP50-95	0.625	0.844	0.868	0.824

3.3. YOLOv8 için detaylı sonuçlar

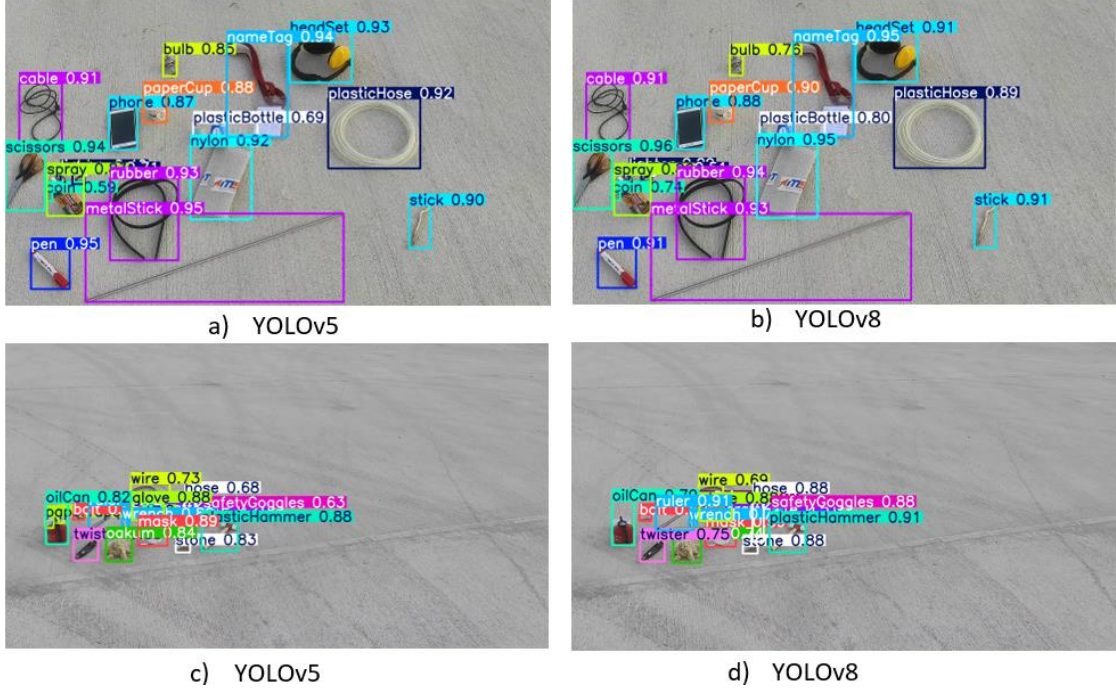
Yolov8 modellerinin türlerine ait kesinlik oranı, duyarlılık oranı, F1-Skoru, mAP50, mAP50-95 değerlendirme kriterleri Tablo 3.3'de ifade edilmiştir. Yolov8'in çıktıları ve sonuçları Şekil 3.2 ve Tablo 3.3'de gösterilmiştir.

Tablo 3.3. Yolov8 mimarisinin alt modülleri üzerindeki eğitim sonuçları

Metrikler	YOLOv8s	YOLOv8m	YOLOv8l	YOLOv8x
Kesinlik Oranı	0.924	0.919	0.934	0.928
Duyarlılık Oranı	0.842	0.856	0.864	0.888
F1-Skor	0.881	0.886	0.897	0.907
mAP50	0.925	0.919	0.924	0.94
mAP50-95	0.925	0.919	0.913	0.939

**Şekil 3.2** Eğitim sonuçları

Tablo 3.3’de, her bir YOLOv8 modeli için beş farklı performans kriterini sunmaktadır. Görüldüğü üzere, genel olarak YOLOv8x daha iyi performans değerleri elde etmektedir. Özellikle, precision (kesinlik) ve recall (duyarlılık) değerlerinden hesaplanan F1-skorda 0,881’den 0,907’ye bir artış gözlemlenirken, mAP50 değeri 0,925’ten 0,94’e yükselmektedir. Ayrıca, modelin mAP50-95 değeri 0,925 ile 0,939 arasında değişmektedir. Şekil 3.2’de Yolov8’in eğitim çıktıları gösterilmiştir. Şekil 3.3’de 2 farklı modelin aynı resim üzerindeki karşılaştırılması gösterilmiştir.



Şekil 3.3 Yolov5 ile Yolov8'in karşılaştırılması

3.4. Yolov11 için detaylı sonuçlar

Geliştirilen ve eğitilen Tablo 3.4’de gösterilen YOLOv11-s modeli, değerlendirilen tüm metrikler açısından dengeli ve sağlam bir performans sergilemiştir. Doğruluk (precision) açısından YOLOv5s (0.925) ve YOLOv8s (0.924) modellerine göre biraz daha düşük bir değere (0.899) ulaşmış olsa da, geri çağırma (recall) (0.826 vs. 0.699) ve F1 skoru (0.861 vs. 0.796) açısından YOLOv5s modelini belirgin şekilde geride bırakmıştır. Bu durum, özellikle pist üzerindeki Yabancı Cisim Tespiti (FOD) gibi güvenlik açısından kritik uygulamalarda YOLOv11-s modelinin yanlış negatifleri en aza indirme konusunda daha etkili olduğunu göstermektedir.

Tablo 3.4. Eğitilen tüm yolo modellerinin s çıktılarının sonuçları

Model	YOLOv5s	YOLOv8s	YOLOv11s
Kesinlik Oranı	0.925	0.924	0.899
Duyarlılık Oranı	0.699	0.842	0.826
F1-Skor	0.796	0.881	0.861
mAP50	0.839	0.925	0.895
mAP50-95	0.625	0.925	0.913

YOLOv8s ile karşılaştırıldığında ise, YOLOv11-s modeli özellikle geri çağırma ve F1 skoru bakımından rekabetçi sonuçlar sunarken, daha hafif bir mimari (9.45M parametreye karşılık YOLOv8s'in yaklaşık 11.2M parametresi) ile çalışmaktadır. Her ne kadar YOLOv8s modeli mAP@0.50 ve mAP@0.50–0.95 ölçümlerinde biraz daha yüksek puanlara ulaşmış olsa da, aradaki fark minimal düzeydedir. Bu da YOLOv11-s'in, algılama doğruluğu ile hesaplama maliyeti arasında verimli bir denge sunduğunu göstermektedir.

Ayrıca, YOLOv11-s modeli yaklaşık 4.2 GB belleğe sahip orta düzey bir L4 GPU üzerinde eğitimi yaklaşık 45 dakikada tamamlamıştır. Bu da, donanım kaynaklarının kısıtlı olduğu gerçek zamanlı dağıtım senaryoları için modelin uygunluğunu vurgulamaktadır. Bu tür bir verimlilik, rekabetçi performansla birleştiğinde, YOLOv11-s'i havaalanlarında gömülü FOD izleme sistemleri için güçlü bir aday hâline getirmektedir.

Sonuç olarak, elde edilen bulgular, YOLOv11-s modelinin hem geleneksel nesne algılama sistemlerine hem de daha yeni derin öğrenme tabanlı modellere kıyasla maliyet açısından verimli ve yüksek performanslı bir alternatif sunduğunu doğrulamaktadır. FOD sınıfları üzerindeki yüksek genelleme yeteneği ve düşük çıkarım gecikmesi sayesinde, modelin ölçeklenebilir ve gerçek zamanlı pist güvenliği uygulamaları için potansiyel taşıdığı ortaya konmuştur. Gelecekte, YOLOv11'in daha büyük varyantlarının (m, l, x) eğitilmesiyle, algılama doğruluğunun daha da artırılması ve YOLOv8 tabanlı modellerle olan kalan performans farkının kapatılması beklenmektedir.

3.5. Genel sonuç

Geliştirilen YOLOv11-s modeli, nesne tespiti görevlerinde yüksek doğruluk ve verimlilik sağlayarak hem geleneksel hem de güncel derin öğrenme tabanlı yaklaşımlara güçlü bir alternatif sunmuştur. YOLOv11-s, YOLOv5s'e kıyasla belirgin şekilde daha yüksek geri çağırma ve F1 skoru elde etmiş, YOLOv8s'e karşı ise daha hafif yapısıyla benzer performans göstermiştir.

Özellikle sınırlı donanım kaynaklarına sahip gerçek zamanlı uygulamalarda kısa eğitim süresi ve düşük parametre sayısı ile öne çıkan model, pist üzerinde yabancı cisim tespiti (FOD) gibi güvenlik açısından kritik sistemlerde kullanılabilirliğini kanıtlamıştır. Elde edilen bulgular, YOLOv11-s'in hem hesaplama maliyetini düşüren hem de algılama doğruluğunu koruyan dengeli bir çözüm sunduğunu göstermektedir.



4. TARTIŞMA

Geliştirilen YOLOv11-s modeli, YOLO mimarisinin evrimsel ilerleyişini temel olarak optimize edilmiş bir yapı sunmuş ve mevcut nesne tespiti modelleriyle karşılaştırıldığında oldukça dengeli bir performans ortaya koymuştur. YOLOv5s, YOLOv8s ve önerilen YOLOv11-s modelleri arasında gerçekleştirilen karşılaştırmalı deneyler; hassasiyet (precision), geri çağırma (recall), F1 skoru ve farklı IoU eşiklerinde hesaplanan ortalama ortalama hassasiyet ($mAP@50$ ve $mAP@50-95$) gibi temel metriklerle dayanmaktadır.

Deneysel bulgular doğrultusunda YOLOv11-s, hassasiyet açısından YOLOv5s (0.925) ve YOLOv8s (0.924) modellerine göre çok az farkla daha düşük bir değer (0.899) üretse de, geri çağırma oranında YOLOv5s'e kıyasla önemli bir gelişim (0.826 vs. 0.699) sağlamıştır. Bu da özellikle güvenlik hassasiyeti yüksek olan pist üzerindeki yabancı cisim tespiti (FOD) gibi uygulamalarda, yanlış negatiflerin azaltılması açısından büyük bir avantajdır. Nitekim, F1 skorundaki artış (YOLOv11-s: 0.861, YOLOv5s: 0.796) modelin genel doğruluk ve tutarlılığını göstermektedir.

YOLOv11-s'in YOLOv8s ile karşılaştırıldığında, özellikle geri çağırma (0.842'ye karşı 0.826) ve F1 skorunda (0.881'e karşı 0.861) rekabetçi bir performans sergilediği gözlemlenmiştir. Ancak asıl öne çıkan unsur, bu performansın 9.45 milyon parametrelilik daha hafif bir mimariyle elde edilmiş olmasıdır; bu değer, yaklaşık 11.2 milyon parametreye sahip YOLOv8s'ten daha düşük bir hesaplama yükünü işaret etmektedir. $mAP@0.50$ ve $mAP@0.50-0.95$ değerlerinde YOLOv8s hafif bir üstünlük sağlasa da (sırasıyla 0.925 ve 0.925'e karşı 0.895 ve 0.913), bu fark minimum düzeydedir ve YOLOv11-s'in doğruluk ile işlem verimliliği arasında etkili bir denge sunduğunu göstermektedir.

Modelin pratik uygunluğunu ortaya koyan bir diğer önemli gösterge ise eğitim süresidir. YOLOv11-s modeli, yalnızca 4.2 GB belleğe sahip orta seviye bir L4 GPU üzerinde yaklaşık 45 dakikalık kısa bir eğitim süresiyle tamamlanmıştır. Bu, sınırlı donanım kaynaklarıyla çalışan gömülü sistemlerde ve uç cihazlarda (edge devices) gerçek zamanlı kullanım için uygun bir profil çizmektedir. Bu bağlamda YOLOv11-s, pist güvenliği gibi gerçek dünyada uygulama alanı bulunan sistemler için düşük gecikme süresi ve yüksek genel performansıyla dikkat çekici bir adaydır. Bununla birlikte, yapılan analizler, YOLOv11-s'in küçük nesnelere ya da uzaktaki nesnelere tespitinde daha etkili olduğunu da ortaya koymuştur. Bu durum, Fig.11'de gözlemlendiği gibi, uzak mesafelerde YOLOv5'e kıyasla daha yüksek doğrulukta tespit sağlayan YOLOv8'in avantajını da paylaşmaktadır. Bu özellik, YOLOv11-s'in de benzer şekilde küçük ve uzak nesnelere daha yüksek tespit başarısı sunabileceğini göstermektedir.

Sonuç olarak, önerilen YOLOv11-s modeli; daha düşük parametre sayısı, hızlı eğitim süresi ve rekabetçi doğruluk performansı ile hem geleneksel YOLO modellerine hem de daha yeni

sürümlere karşı güçlü bir alternatif sunmaktadır. Ayrıca, bu modelin m, l ve x gibi daha büyük varyantlarının eğitilmesiyle doğruluk oranlarının daha da iyileştirilmesi beklenmektedir. Bu durum, YOLOv11-s'in sunduğu yüksek ölçeklenebilirlik ve esneklik ile birlikte, pist güvenliği gibi kritik alanlarda uygulanabilirliğini daha da güçlendirmektedir.



5. SONUÇLAR

Çalışmanın sonuçları, modern derin öğrenme tabanlı nesne algılama algoritmalarının, havalimanı pistlerinde bulunan yabancı madde hasarlarının (YMH) tespiti konusunda etkili ve uygulanabilir çözümler sunduğunu ortaya koymuştur. YOLOv5s, YOLOv8s ve YOLOv11-s modelleri, FOD-A ve Runway FOD veri setleri üzerinde test edilerek karşılaştırmalı bir şekilde değerlendirilmiş; özellikle YOLOv11-s modeli, hafif yapısına rağmen yüksek duyarlılık (recall), F1 skoru ve genelleme kabiliyeti ile öne çıkmıştır. YOLOv8s modeli ise genel doğruluk ve mAP performansları bakımından güçlü sonuçlar vermiştir. Bu bulgular, derin öğrenme modellerinin farklı hava ve ışık koşullarında, çeşitli nesne boyutlarında ve sınırlı donanım kaynaklarında dahi başarılı şekilde çalışabileceğini göstermektedir.

Elde edilen sonuçlar, bu tür modellerin hava taşımacılığı güvenliği için gerçek zamanlı ve ölçeklenebilir çözümler üretme potansiyelini vurgulamaktadır. Özellikle YOLOv11-s modelinin düşük parametre sayısı ve hızlı eğitilebilme kapasitesi, onu gömülü sistemlerde veya düşük güçlü donanımlarda kullanım için uygun hale getirmektedir. Ancak, çalışma yalnızca sınırlı sayıda veri setiyle ve belirli pist ortamlarıyla sınırlı kaldığı için, gelecekte modelin farklı pist tipleri, kamera sistemleri ve coğrafi koşullar altında test edilmesi önem arz etmektedir.

Sonuç olarak, bu çalışma, hazır ve güncel YOLO mimarilerinin YMH tespiti bağlamında sistematik olarak analiz edildiği, karşılaştırmalı bir değerlendirme sunmaktadır. Derin öğrenme temelli bu yaklaşımların, geleneksel yöntemlere kıyasla daha yüksek doğruluk, hız ve esneklik sağladığı görülmüş; özellikle uçuş güvenliği gibi kritik uygulamalarda büyük bir potansiyele sahip olduğu doğrulanmıştır. Gelecekte yapılacak çalışmalar, bu potansiyeli daha da genişleterek, akıllı havaalanı sistemlerinin geliştirilmesine katkı sağlayabilir.

ÖNERİLER

Çalışmada, hazır ve yaygın olarak kullanılan YOLOv5, YOLOv8 ve YOLOv11 modelleri değerlendirilmiş ve bu modellerin çeşitli nesne tespiti görevlerindeki başarıları karşılaştırılmıştır. Model mimarilerinde herhangi bir özgün geliştirme yapılmaksızın, literatürde mevcut olan mimariler doğrudan uygulanarak deneyler gerçekleştirilmiştir. Elde edilen bulgular, her modelin farklı senaryolarda öne çıkan avantaj ve dezavantajlarını ortaya koymuştur. Bu doğrultuda, gelecekteki çalışmalar için şu önerilerde bulunulabilir:

- **Model Özelleştirme ve Hiperparametre Optimizasyonu:** Bu çalışmada modeller varsayılan ayarlarla değerlendirilmiştir. Gelecekte, her bir modelin belirli uygulama senaryolarına göre hiperparametre optimizasyonları (öğrenme oranı, epoch sayısı, veri artırma teknikleri vb.) yapılarak performans daha da artırılabilir.
- **Farklı Değerlendirme Metriklerinin Eklenmesi:** Mevcut çalışmada kullanılan precision, recall, F1-score ve mAP gibi yaygın metriklere ek olarak, inference süresi, model boyutu, enerji tüketimi gibi uygulamaya dönük kriterler de gelecekteki değerlendirmelerde dikkate alınmalıdır.
- **Model Hafifletme ve Gerçek Zamanlı Kullanım:** YOLO modelleri gerçek zamanlı uygulamalarda oldukça popülerdir. Gelecekte, daha düşük kaynak tüketimi için model küçültme (pruning, quantization) teknikleri uygulanarak özellikle gömülü sistemler ve mobil cihazlar için uygun yapılar geliştirilebilir.
- **Yeni YOLO Varyantlarının Dâhil Edilmesi:** YOLOv12 gibi yeni çıkan mimariler de karşılaştırmalı analizlere dahil edilerek, güncel gelişmelerin ışığında daha kapsamlı kıyaslamalar yapılabilir.
- **Transfer Öğrenme ve Domain Adaptasyonu:** Eğer çalışma alanı dar bir veri setine sahipse, transfer öğrenme yöntemleri kullanılarak önceden eğitilmiş modellerin farklı alanlara adapte edilmesi sağlanabilir. Bu durum, etiketlenmiş veri azlığı olan alanlarda özellikle önemlidir.

Sonuç olarak, hazır modellerin karşılaştırılması yoluyla yapılan bu çalışmanın, ileriye dönük özgün model geliştirme ya da uygulama odaklı projeler için temel bir referans niteliğinde olduğu değerlendirilmektedir. Gelecek çalışmalarda hem mimari düzeyde yenilikler hem de farklı uygulama senaryolarına yönelik derinlemesine analizlerle bu çalışmaların genişletilmesi mümkündür.

KAYNAKLAR

- Akhtar, N., & Ragavendran, U. (2020). Interpretation of intelligence in CNN-pooling processes: a methodological survey. *Neural Computing and Applications*, 32(3), 879–898. <https://doi.org/10.1007/s00521-019-04296-5>
- Bjorck, J., Gomes, C., Selman, B., & Weinberger, K. Q. (2018). Understanding batch normalization. *Advances in Neural Information Processing Systems, 2018-Decem(NeurIPS)*, 7694–7705.
- Bochkovskiy, A., Wang, C.-Y., & Liao, H.-Y. M. (2020). *YOLOv4: Optimal Speed and Accuracy of Object Detection*. <http://arxiv.org/abs/2004.10934>
- Colleges, G. T. U. A., Academy, O., Academy, O., Academy, O., Science, A. C., Technology, I., & Science, A. C. (2014). Microsoft COCO 2014. *Eccv*, 740–755.
- Doherty, J., Gardiner, B., Kerr, E., & Siddique, N. (2024). BiFPN-YOLO: One-Stage Object Detection Integrating Bi-Directional Feature Pyramid Networks (Under Review). *Elsevier Pattern Recognition Letters, Pending Re(Pending Review)*.
- Dubey, S. R., Singh, S. K., & Chaudhuri, B. B. (2022). Activation functions in deep learning: A comprehensive survey and benchmark. *Neurocomputing*, 503, 92–108. <https://doi.org/10.1016/j.neucom.2022.06.111>
- Dumoulin, V., & Visin, F. (2016). *A guide to convolution arithmetic for deep learning*. 1–31. <http://arxiv.org/abs/1603.07285>
- FAA. (2010). *Airport Foreign Object Debris (FOD) Management*.
- Grandini, M., Bagli, E., & Visani, G. (2020). *Metrics for Multi-Class Classification: an Overview*. 1–17. <http://arxiv.org/abs/2008.05756>
- Güçlü, E., Aydın, İ., Şahbaz, K., Aakın, E., & Karaköse, M. (2021). Demiryolu Bağlantı Elemanlarında Bulunan Kusurların YOLOv4 ve Bulanık Mantık Kullanarak Tespiti. *Demiryolu Mühendisliği*, 14, 249–262. <https://doi.org/10.47072/demiryolu.939830>
- Han, L., Ma, C., Liu, Y., Jia, J., & Sun, J. (2023). SC-YOLOv8: A Security Check Model for the Inspection of Prohibited Items in X-ray Images. *Electronics (Switzerland)*, 12(20). <https://doi.org/10.3390/electronics12204208>
- Hand, D. J., Christen, P., & Kirielle, N. (2021). F*: an interpretable transformation of the F-measure. *Machine Learning*, 110(3), 451–456. <https://doi.org/10.1007/s10994-021-05964-1>
- He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(9), 1904–1916. <https://doi.org/10.1109/TPAMI.2015.2389824>
- He, Y., Zhu, C., Wang, J., Savvides, M., & Zhang, X. (2019). Bounding box regression with uncertainty for accurate object detection. *Proceedings of the IEEE Computer Society*

- Conference on Computer Vision and Pattern Recognition, 2019-June*, 2883–2892.
<https://doi.org/10.1109/CVPR.2019.00300>
- Hosang, J., & Planck, M. (y.y.). *Learning non-maximum suppression*.
- Howard, J., & Gugger, S. (2020). Fastai: A layered api for deep learning. *Information (Switzerland)*, *11*(2), 1–26. <https://doi.org/10.3390/info11020108>
- Khalfauoi-Hassani, I., Pellegrini, T., & Masquelier, T. (2021). *Dilated convolution with learnable spacings*. *2022*, 1–29. <http://arxiv.org/abs/2112.03740>
- Lederer, J. (2021). *Activation Functions in Artificial Neural Networks: A Systematic Overview*. 1–42. <http://arxiv.org/abs/2101.09957>
- Li, P., & Li, H. (2020). Research on FOD Detection for Airport Runway based on YOLOv3. *Chinese Control Conference, CCC, 2020-July*, 7096–7099. <https://doi.org/10.23919/CCC50068.2020.9188724>
- Lin, T., Zhu, Y., Ren, Z., Huang, K., Zhang, X., Yan, K., & Huang, S. (2023). Sound-Based Intelligent Detection of FOD in the Final Assembly of Rocket Tanks. *Machines*, *11*(2). <https://doi.org/10.3390/machines11020187>
- Liu, S., Qi, L., Qin, H., Shi, J., & Jia, J. (2019). PANet: Path Aggregation Network for Instance Segmentation. (arXiv:1803.01534v3 [cs.CV] UPDATED). *Cvpr*, 8759–8768. <http://arxiv.org/abs/1803.01534>
- Misra, D. (2019). *Mish: A Self Regularized Non-Monotonic Activation Function*. <http://arxiv.org/abs/1908.08681>
- Munyer, T., Huang, P.-C., Huang, C., & Zhong, X. (2021). *FOD-A: A Dataset for Foreign Object Debris in Airports*. <http://arxiv.org/abs/2110.03072>
- Nhancements, E. (2024). *YOLOV11: AN OVERVIEW OF THE KEY ARCHITECTURAL ENHANCEMENTS*. *2024*, 1–9.
- Noroozi, M., & Shah, A. (2023). Towards optimal foreign object debris detection in an airport environment. *Expert Systems with Applications*, *213*(PA), 118829. <https://doi.org/10.1016/j.eswa.2022.118829>
- Papadopoulos, E., & Gonzalez, F. (2021). UAV and AI Application for Runway Foreign Object Debris (FOD) Detection. *IEEE Aerospace Conference Proceedings, 2021-March*. <https://doi.org/10.1109/AERO50100.2021.9438489>
- Paul, A., Bandyopadhyay, R., Yoon, J. H., Geem, Z. W., & Sarkar, R. (2022). SinLU: Sinu-Sigmoidal Linear Unit. *Mathematics*, *10*(3), 1–14. <https://doi.org/10.3390/math10030337>
- Qin, F., Bu, X., Liu, Y., Liang, X., & Xin, J. (2021). Foreign object debris automatic target detection for millimeter-wave surveillance radar. *Sensors*, *21*(11). <https://doi.org/10.3390/s21113853>
- Redmon, J., & Farhadi, A. (2018). *YOLOv3: An Incremental Improvement*. <http://arxiv.org/abs/1804.02767>

- SHGM. (2014). www.shgm.gov.tr
- Sohan, M., Sai Ram, T., & Rami Reddy, C. V. (2024). A Review on YOLOv8 and Its Advancements. *Data Intelligence and Cognitive Informatics Proceedings of ICDICI 2023*, 529–545. https://doi.org/10.1007/978-981-99-7962-2_39
- Terven, J., & Cordova-Esparza, D. (2023). A Comprehensive Review of YOLO: From YOLOv1 and Beyond. 1–34. <http://arxiv.org/abs/2304.00501>
- TUBİTAK. (2023, Haziran 3). <https://bilgem.tubitak.gov.tr/tr/urunler/fodrad-fod-tespit-sistemi>.
- Vovk, V. (2015). The fundamental nature of the log loss function. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9300, 307–318. https://doi.org/10.1007/978-3-319-23534-9_20
- Wang, C.-Y., Bochkovskiy, A., & Liao, H.-Y. M. (2023). YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. 7464–7475. <https://doi.org/10.1109/cvpr52729.2023.00721>
- Wang, J., Yan, S., Zhou, L., Fan, R., Han, X., & Wang, X. (2021). Research on Application of New Technology in Foreign Object Debris Detection Field. *Proceedings of 2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology, ICCASIT 2021*, 460–464. <https://doi.org/10.1109/ICCASIT53235.2021.9633616>
- Wang, X., Gao, H., Jia, Z., & Li, Z. (2023). BL-YOLOv8: An Improved Road Defect Detection Model Based on YOLOv8. *Sensors (Basel, Switzerland)*, 23(20). <https://doi.org/10.3390/s23208361>
- Weng, W., & Zhu, X. (2021). INet: Convolutional Networks for Biomedical Image Segmentation. *IEEE Access*, 9, 16591–16603. <https://doi.org/10.1109/ACCESS.2021.3053408>
- Xie, L., Zhai, J., Wu, B., Wang, Y., Zhang, X., Sun, P., & Yan, S. (2020). ELAN: Towards generic and efficient elastic training for deep learning. *Proceedings - International Conference on Distributed Computing Systems, 2020-Novem*(Figure 1), 78–88. <https://doi.org/10.1109/ICDCS47774.2020.00018>
- Xu, P., Li, Q., Zhang, B., Wu, F., Zhao, K., Du, X., Yang, C., & Zhong, R. (2021). On-board real-time ship detection in hisea-1 sar images based on cfar and lightweight deep learning. *Remote Sensing*, 13(10), 1–18. <https://doi.org/10.3390/rs13101995>
- Yan, Y., Li, J., Qin, J., Bai, S., Liao, S., Liu, L., Zhu, F., & Shao, L. (2021). Anchor-Free Person Search. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 7686–7695. <https://doi.org/10.1109/CVPR46437.2021.00760>
- Yang, T., & Ying, Y. (2023). AUC Maximization in the Era of Big Data and AI: A Survey. *ACM Computing Surveys*, 55(8), 1–37. <https://doi.org/10.1145/3554729>
- Yao, Z., & Lin, S. (y.y.). *Cross-Iteration Batch Normalization*. 12331–12340.
- Yu, F., & Koltun, V. (2016). Multi-scale context aggregation by dilated convolutions. *4th*

International Conference on Learning Representations, ICLR 2016 - Conference Track Proceedings.

Zhao, Y., Han, R., & Rao, Y. (2019). A new feature pyramid network for object detection. *Proceedings - 2019 International Conference on Virtual Reality and Intelligent Systems, ICVRIS 2019*, 428–431. <https://doi.org/10.1109/ICVRIS.2019.00110>

Zhong, J., Gou, X., Shu, Q., Liu, X., & Zeng, Q. (2021). A fod detection approach on millimeter-wave radar sensors based on optimal vmd and svdd. *Sensors (Switzerland)*, *21*(3), 1–19. <https://doi.org/10.3390/s21030997>

Zhong, Y., Wang, J., Peng, J., & Zhang, L. (2020). Anchor box optimization for object detection. *Proceedings - 2020 IEEE Winter Conference on Applications of Computer Vision, WACV 2020*, 1275–1283. <https://doi.org/10.1109/WACV45572.2020.9093498>



ÖZGEÇMİŞ

Necip Şahamettin KÜÇÜK

KİŞİSEL BİLGİLER

[REDACTED] [REDACTED]
[REDACTED] [REDACTED]
[REDACTED] [REDACTED]
[REDACTED] [REDACTED]
[REDACTED] [REDACTED]
[REDACTED] [REDACTED]

ARAŞTIRMACI BİLGİLERİ

Öğrenci Orcid ID : 0009-0009-9063-0076
Danışman Orcid ID : 0000-0001-5605-0419

EĞİTİM BİLGİLERİ

[REDACTED] [REDACTED]
[REDACTED] [REDACTED]
[REDACTED] [REDACTED]
[REDACTED] [REDACTED]
[REDACTED] [REDACTED]

ARAŞTIRMA DENEYİMİ

- ✓ Python
- ✓ YOLO