



T.C. KIRIKKALE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**DERİN ÖĞRENME YÖNTEMLERİ İLE UYDU GÖRÜNTÜLERİNDEN
GEMİLERİN TESPİTİ VE SINIFLANDIRILMASI**

MEHMET SAMİ TÜRKER
SAVUNMA TEKNOLOJİLERİ ANABİLİM DALI

YÜKSEK LİSANS TEZİ

DANIŞMAN
Dr. Öğr. Üyesi Enes AYAN

KIRIKKALE-2024



T.C. KIRIKKALE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**DERİN ÖĞRENME YÖNTEMLERİ İLE UYDU GÖRÜNTÜLERİNDEN
GEMİLERİN TESPİTİ VE SINIFLANDIRILMASI**

MEHMET SAMİ TÜRKER
SAVUNMA TEKNOLOJİLERİ ANABİLİM DALI

YÜKSEK LİSANS TEZİ

DANIŞMAN
Dr. Öğr. Üyesi Enes AYAN

KIRIKKALE-2024

KABUL ONAY SAYFASI

Mehmet Sami TRKER tarafından hazırlanan “DERİN ÖĞRENME YÖNTEMLERİ İLE UYDU GÖRÜNTÜLERİNDEN GEMİLERİN TESPİTİ VE SINIFLANDIRILMASI” adlı tez çalışması, aşağıdaki jüri tarafından OY BİRLİĞİ ile Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü Savunma Teknolojileri Anabilim Dalında YKSEK LİSANS TEZİ olarak kabul edilmiştir.

Danışman: Dr. Öğr. Üyesi Enes AYAN

İmza.....

Bilgisayar Mühendisliği Anabilim Dalı, Kırıkkale Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

Başkan: Doç. Dr. Serkan Savaş

İmza.....

Bilgisayar Mühendisliği Anabilim Dalı, Kırıkkale Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

Üye: Dr. Öğr. Üyesi Ahmet Haşim YURTTAKAL

İmza.....

Bilgisayar Mühendisliği Anabilim Dalı, Afyon Kocatepe Üniversitesi

Bu tezin, kapsam ve kalite olarak Yüksek Lisans Tezi olduğunu onaylıyorum.

Tez Savunma Tarihi: 12/09/2024

Jüri tarafından kabul edilen bu tezin Yüksek Lisans Tezi olması için gerekli şartları yerine getirdiğini onaylıyorum.

.....

Prof. Dr. Recep ÇALIN

Fen Bilimleri Enstitüsü Müdürü

ETİK BEYANI

Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmada;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmada yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

İmza

Mehmet Sami TÜRKER

20/09/2024

ÖZET

DERİN ÖĞRENME YÖNTEMLERİ İLE UYDU GÖRÜNTÜLERİNDEN GEMİLERİN TESPİTİ VE SINIFLANDIRILMASI

TÜRKER, Mehmet Sami

Kırıkkale Üniversitesi

Fen Bilimleri Enstitüsü

Savunma Teknolojileri Anabilim Dalı, Yüksek Lisans Tezi

Danışman: Dr. Öğr. Üyesi Enes AYAN

Eylül 2024, 61 Sayfa

Uzaktan algılama, yüzeydeki objeleri ve olayları uzaktan gözlemleme ve analiz etme teknolojisidir. Bu yöntemle genellikle uydular, dronlar veya hava araçları gibi platformlar üzerinden çeşitli elektromanyetik dalgalar kullanarak veriler toplanır. Toplanan veriler, optik, kızılötesi veya radar sensörleriyle elde edilen görüntüler şeklindedir ve bu görüntüler çevresel değişikliklerin izlenmesi, tarım verimliliğinin artırılması, şehir planlaması, doğal afetlerin yönetimi, savunma sistemleri ve iklim değişikliği gibi pek çok alanda kullanılabilir. Uzaktan algılama yöntemleri ile elde edilen veri setleri derin öğrenme yöntemleri kullanılarak geniş alanların veri analizi ve bilgi çıkarımı için kullanılmaktadır. Bu sayede geniş alanların hızlı ve etkili bir şekilde değerlendirilmesini sağlanarak, birçok sektörde bilgi ve karar destek sistemlerine katkıda bulunur. Derin öğrenme algoritmaları, uzaktan algılama veri setlerinde nesne tanıma, değişim tespiti ve sınıflandırma gibi görevlerde yüksek doğruluk ve otomasyon sağlamaktadır. Uzaktan algılama yöntemlerinden biri olan optik uydu görüntüleri gemilerin otomatik tespiti ve sınıflandırılması, deniz trafiği yönetimi, çevre kirliliğinin izlenmesi, uyuşturucu kaçakçılığı, göçmen kaçakçılığı, sınır ihlalleri ve meydana gelebilecek diğer suçların takibi açısından kritik bir araştırma alanıdır.

Son yıllarda, derin öğrenme algoritmalarının popüler hale gelmesiyle birlikte, Evrişimli Sinir Ağları (ESA) nesne algılama ve sınıflandırma gibi bilgisayarlı görme problemlerinde başarılı sonuçlar elde etmiştir. Bu çalışma kapsamında bilgisayarlı görü problemlerinde nesne tespit görevlerinde kullanılan You Only Look Once (YOLO) tabanlı on bir farklı mimari (YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, YOLOv7tiny, YOLOv7, YOLOv9c, YOLOv9m, YOLOv9s, YOLOv9t, YOLOv7-CBAM) kullanılmıştır. Söz konusu mimariler transfer öğrenme yöntemi ile optik uydu görüntüleri ile gemileri tespit edebilmeleri için eğitilmiştir. Eğitim sonrası mimarilerin uydu görüntülerinden gemileri tespit edebilme ve sınıflandırma performansları kesinlik, duyarlılık, mAP (ortalama kesinlik) ve F1 skor ölçütleri kullanılarak karşılaştırılmıştır. Elde edilen test sonuçlarına YOLOv5l modelinin 0.984 kesinlik, 0.984 duyarlılık, 0.991 mAP ve 0.984 F1 skor değeri elde ederek diğer modeller arasında en başarılı model olmuştur. Çalışmada sonuç olarak özellikle büyük modellerin tespit performansının daha iyi olduğu gözlemlenmiştir. Çalışmada ayrıca kullanılan dikkat mekanizmasının YOLOv7 modelinde F1 skor ölçütünde %5,4 oranında bir iyileştirme sağladığı tespit edilmiştir. Çalışma sonuçlarına göre YOLO tabanlı mimarilerin uydu görüntülerinden gemi tespiti ve sınıflandırmada umut vadeci sonuçlar elde ettiği bulunmuştur.

Anahtar Kelimeler: Uzaktan algılama, Nesne Tespiti, YOLO, Evrişimli Sinir Ağları, Gemi Tespit ve Sınıflandırma, Derin Öğrenme.

ABSTRACT

DETECTION AND CLASSIFICATION OF SHIPS FROM SATELLITE IMAGES USING DEEP LEARNING METHODS

TÜRKER, Mehmet Sami

Kırıkkale University

Institute of Science and Technology

Department of Defense Technologies, Master Thesis

Consultant: Assoc. Prof. Enes AYAN

September 2024, 61 Pages

Remote sensing is the technology of observing and analyzing objects and events on the surface from a distance. This method entails the collection of data through the utilization of diverse electromagnetic waves, typically from platforms such as satellites, drones or aircraft. The data collected are in the form of images obtained with optical, infrared or radar sensors. These images can be used in a number of areas, including monitoring environmental changes, increasing agricultural productivity, urban planning, management of natural disasters, defense systems and climate change. The data sets obtained through remote sensing methods are employed for the analysis of data and the extraction of information from extensive areas through the utilization of deep learning techniques. In this way, it contributes to the development of information and decision support systems in a number of sectors, providing a rapid and effective means of evaluating large areas. Deep learning algorithms facilitate high accuracy and automation in tasks such as object recognition, change detection and classification in remote sensing data sets. Optical satellite imagery represents a crucial domain of remote sensing research, offering significant potential for automated

detection and classification of maritime vessels, management of maritime traffic, monitoring of environmental pollution, detection of drug and migrant smuggling, and identification of border violations and other criminal activities. In recent years, the proliferation of deep learning algorithms has led to the successful application of Convolutional Neural Networks (CNNs) in a range of computer vision problems, including object detection and classification. In this study, eleven distinct You Only Look Once (YOLO) based architectures were employed for object detection tasks in computer vision problems, namely YOLOv5n, YOLOv5s, YOLOv5m, YOLOv5l, YOLOv7tiny, YOLOv7, YOLOv9c, YOLOv9m, YOLOv9s, and YOLOv9t, as well as YOLOv7-CBAM. The architectures were trained to detect ships from optical satellite images using a transfer learning method. Following the training phase, the performance of the architectures in detecting and classifying ships from satellite images was evaluated using a series of performance metrics, including precision, sensitivity, mAP (mean average precision), and F1 score. The results of the test demonstrated that the YOLOv5l model exhibited the highest degree of success, attaining precision, sensitivity, mAP, and F1 score values of 0.984, 0.984, 0.991, and 0.984, respectively. It was thus observed that the detection performance of the larger models was superior. Furthermore, it was observed that the attention mechanism employed in the study yielded a 5.4% enhancement in the F1 score criterion for the YOLOv7 model. The findings of the study indicate that YOLO-based architectures demonstrate considerable potential for effective ship detection and classification from satellite images.

Keywords: Remote sensing, Object Detection, YOLO, Convolutional Neural Networks, Ship Detection and Classification, Deep Learning.

TEŞEKKÜR

Bu tez çalışmasının hazırlanmasında bana destek olan ve katkı sağlayan birçok kişiye teşekkürlerimi sunmak isterim.

Öncelikle, çalışmamın her aşamasında bana rehberlik eden, bilgi ve deneyimlerini paylaşarak bu çalışmanın gelişimine büyük katkı sağlayan kıymetli danışmanım Sayın Doktor Öğretim Üyesi Enes AYAN'a en içten teşekkürlerimi sunarım.

Beni lisansüstü eğitime teşvik ederek cesaretlendiren ve tez sürecinde yardımlarını esirgemeyen çok kıymetli hayat arkadaşım Buse GÜNAYDIN TÜRKER'e ve dünyaya gelmesine sayılı zaman kalan biricik evladım Oğuz Ata TÜRKER'e tüm kalbi duygularıyla teşekkürü borç bilirim.

Ayrıca, eğitim hayatım boyunca bana her zaman destek olan, varlıklarıyla güç veren sevgili aileme sonsuz teşekkürlerimi iletmek isterim. Annem Cennet TÜRKER, babam Bülent TÜRKER ve kardeşim Ahu Ceren TÜRKER'e, bana verdikleri maddi ve manevi destekten dolayı minnettarım.

Son olarak, çalışmamın her aşamasında bana destek olan değerli amirlerim SG Albay Levent BECERİK ve SG Müh.Yzb. Arda GÜLERER'e; mesai arkadaşlarım SG Müh.Ütğm. Muhammed Gürani AĞIRMAN, SG Müh.Ütğm. Hasan EKER ve SG Müh.Ütğm. Fatih GÜÇLÜ'ye teşekkürü bir borç bilirim.

İÇİNDEKİLER DİZİNİ

Sayfa

ÖZET	iv
ABSTRACT	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER DİZİNİ.....	ix
ÇİZELGELER DİZİNİ.....	xi
ŞEKİLLER DİZİNİ	xii
KISALTMALAR DİZİNİ	xiii
1. GİRİŞ.....	1
1.1 . Tezin Amacı.....	1
1.2. Literatür Taraması.....	3
1.2.1. SAR Ship Detection Dataset (SSDD).....	3
1.2.2. High-Resolution SAR Images Dataset (HRSID).....	4
1.2.3. AIR-SARShip-1.0	4
1.2.4. Airbus Ship Detection Dataset (ASDD)	5
2. EVRİŞİMLİ SİNİR AĞLARI VE NESNE TESPİTİ.....	9
2.1. Yapay Zekâ	9
2.2. Makine Öğrenmesi	9
2.3. Derin Öğrenme	10
2.4. Yapay Sinir ağları	11
2.4.1. Yapay Sinir Ağlarında Ezberleme ve Ezberlemeyi Engelleme Yöntemleri	15
2.5. Evrişimli Sinir Ağları (ESA)	16
2.5.1. ESA Mimarisi	17
2.5.2. ESA’lar ile Nesne Tespiti	23
2.5.3. YOLO	25
3. YOLO İLE GEMİ TESPİTİ.....	37
3.1. Veri Seti	37
3.2. Kullanılan Yöntemler	40
3.2.1. CBAM (Convolutional Block Attention Module)	41
3.3. Değerlendirme Kriterleri	42

3.4. Geliştirme Ortamı ve Eğitim Parametreleri.....	45
4. DENEYLER VE SONUÇLAR.....	46
5. TARTIŞMA	51
6. SONUÇ VE TAVSİYELER	53
KAYNAKLAR.....	55
ÖZGEÇMİŞ.....	61



ÇİZELGELER DİZİNİ

Çizelge 3.1. Veri setinde eğitim ve test grubuna ait sınıf dağılımları	38
Çizelge 3.2. Model eğitim parametreleri.....	45
Çizelge 4.1. Modellerin test verisi üzerindeki performansları.....	46
Çizelge 5.1 Model büyüklükleri.....	44



ŞEKİLLER DİZİNİ

<u>Şekil</u>	<u>Sayfa</u>
Şekil 2.1 Yapay zekâ, makine öğrenmesi ve derin öğrenme ilişkisi	9
Şekil 2.2. Gerçek sinir hücresi ve yapay sinir hücresi	12
Şekil 2.3. Tek sinir hücresi	13
Şekil 2.4. Çok katmanlı yapay sinir ağı örneği	13
Şekil 2.5. Sönümlleme	16
Şekil 2.6. ESA Mimarisi örneği	18
Şekil 2.7. Evrişim işlemi	19
Şekil 2.8. Aktivasyon fonksiyonları	20
Şekil 2.9. Havuzlama işlemi örnekleri	21
Şekil 2.10. Görüntü tanıma, tespit ve yerelleştirme	23
Şekil 2.11. ESA ile kare sınırlama ve sınıflandırma	24
Şekil 2.12. YOLO nesne tespit adımları	26
Şekil 2.13. YOLO algoritmaları tarihsel gelişimi	27
Şekil 2.14. YOLOv5 sınırlayıcı kutu hesaplamaları	28
Şekil 2.15. SiLU aktivasyon fonksiyonu	29
Şekil 2.16. YOLOv5s mimarisi	30
Şekil 2.17. E-ELAN modülleri	31
Şekil 2.18. YOLOv7 ELAN	32
Şekil 2.19. Yolov7 mimarisi	33
Şekil 2.20. Programlanabilir Gradyan mimarisi	34
Şekil 2.21. GELAN modülleri	35
Şekil 2.22 YOLOv9 Mimarisi	36
Şekil 3.1. Veri setinden örnekler	40
Şekil 3.2. Çalışma özeti	40
Şekil 3.3. Convolutional Block Attention mimarisi	41
Şekil 3.4. CBAM entegre edilmiş Yolov7 mimarisi	42
Şekil 3.5. Nesne tespit hesaplama ölçütleri	44
Şekil 4.1. YOLOv5l karışıklık matrisi	48
Şekil 4.2. YOLOv5l Kesinlik-Duyarlılık eğrisi	49
Şekil 4.3. Bazı test görüntüleri ve etiketleri	50
Şekil 4.4. YOLOv5l tahmin sonuçları	50

KISALTMALAR DİZİNİ

ASDD	Airbus Ship Detection Dataset
BCE	Binary Cross Entropy
BN	Batch Normalization
CBAM	Convolutional Block Attention Module
CSP	Cross Stage Partial
DÖ	Derin Öğrenme
ELAN	Efficient Layer Aggregation Network
ESA	Evrişimli Sinir Ağları
GELAN	Genelleştirilmiş Verimli Katman Toplama Ağı
GN	Gerçek Negatif
GP	Gerçek Pozitif
GPU	Grafik İşlem Birimi
HRSID	High-Resolution SAR Images Dataset
IoU	Intersection over Union
İHA	İnsansız Hava Aracı
mAP	Mean Average Precision
MÖ	Makine Öğrenmesi
PGI	Programlanabilir Gradyan Bilgisi
ReLU	Rectified Linear Unit
RPN	Region Proposal Network
R-CNN	Region-based Convolutional Neural Network
SAR	Sentetik Açıklıklı Radar

SPP	Spatial Pyramid Pooling
SSDD	Ship Detection Dataset
VHRShips	Çok Yüksek Çözünürlüklü Gemiler
YN	Yanlış Negatif
YOLO	You Only Look Once
YP	Yanlış Pozitif
YSA	Yapay Sinir Ağları
YZ	Yapay Zeka



1. GİRİŞ

1.1. Tezin Amacı

Bilgi işlem teknolojilerindeki ilerlemeler ile matematiksel hesaplamalar için GPU'ların (Grafik İşleme Birimi- Graphics Processing Unit) kullanılabilirliği, derin öğrenme araştırma alanının daha popüler ve yaygın hale gelmesini sağlayan etmenlerdendir. Derin öğrenme (DÖ), özellikle bilgisayarlı görü problemlerinin çözümünde önemli bir rol oynamaktadır. Bu bağlamda nesne tespiti, otonom sürüş ve bilgisayarla görme gibi uygulamalarda kritik öneme sahiptir.

Savunma teknolojilerinde DÖ, son yıllarda önemli bir gelişme göstermiş ve çeşitli uygulama alanlarında kullanımı hızla artmıştır. Görüntü işleme ve nesne tespiti konularında DÖ algoritmaları, radar ve uydu görüntülerinden gemi, uçak veya diğer askeri hedefleri yüksek doğrulukla tespit edebilmektedir. Deniz güvenliği alanında, DÖ tabanlı sistemler, deniz trafiğini izleme, kaçakçılığı tespit etme ve yasa dışı balıkçılık faaliyetlerini belirleme gibi görevlerde kullanılmaktadır (Liu et al., 2020). Ayrıca, doğal dil işleme teknikleri, istihbarat toplama ve analiz süreçlerinde kullanılmakta, çeşitli dillerdeki metinleri anlamlandırarak kritik bilgileri çıkarabilmektedir. Otonom sistemlerde, DÖ tabanlı algoritmalar, insansız hava araçları (İHA) ve otonom kara araçlarının çevresel algılama ve karar verme kabiliyetlerini artırarak, görevlerini daha etkili ve güvenli bir şekilde yerine getirmelerini sağlamaktadır. Savunma sanayinde siber güvenlik uygulamalarında da DÖ kullanılmakta, olağan dışı durumları tespit etme ve saldırılara karşı önceden önlem alma konusunda önemli avantajlar sunmaktadır. Bu alanlardaki gelişmelerin, savunma teknolojilerinin daha ileri seviyelere taşınmasına ve askerî operasyonların etkinliğinin artırılmasına büyük katkılar sunacağı düşünülmektedir (J. Chen et al., 2022).

Nesne tespit problemi, bilgisayarlı görüde çözülmesi zor problemlerden biridir ve iki temel aşamadan oluşur: nesne yerelleştirmesi ve nesne sınıflandırması. Nesne yerelleştirmesi, bilgisayarın bir görüntüye bakarak nesnelerin konumlarını belirlemesi ve görüntü içerisindeki doğru koordinatları vermesini içerir. Nesne sınıflandırması ise, bilgisayarın tespit ettiği nesneleri farklı kategorilere ayırarak sınıflandırmasını kapsar.

Bu iki bileşen, bir araya gelerek nesne tespiti için temelini oluşturur ve bu sayede bilgisayarlar, karmaşık görüntülerdeki nesneleri hem doğru bir şekilde tanımlayabilir hem de konumlarını belirleyebilirler. Günümüzde, GPU'lar kullanılarak eğitilen DÖ algoritmaları tabanlı nesne tespiti modelleri, otonom araçlardan güvenlik sistemlerine, sağlık sektöründen perakende sektörüne kadar geniş bir yelpazede uygulanmaktadır. Bu gelişmeler, teknolojinin hızlı ilerleyişi ile, daha hassas ve hızlı nesne tespiti yapılabilmesini mümkün kılmaktadır. Hızlı ve güvenilir nesne tespiti güvenlik uygulamaları için büyük önem arz etmektedir. Son yıllarda uydu görüntülerinin yapay zekâ kullanılarak savunma amaçlı kullanıldığı çalışmalar hızla artmaktadır (Petersson et al., 2016).

Uzaktan algılama, yüzeydeki objeleri ve olayları uzaktan gözlemleme ve analiz etme teknolojisidir. Bu yöntem, genellikle uydular, dronlar veya hava araçları gibi platformlar üzerinden çeşitli elektromanyetik dalgalar kullanarak veri toplar. Toplanan veriler, optik, kızılötesi veya radar sensörleriyle elde edilen görüntüler şeklindedir ve bu görüntüler çevresel değişikliklerin izlenmesi, tarım verimliliğinin artırılması, şehir planlaması, doğal afetlerin yönetimi ve iklim değişikliği gibi pek çok alanda kullanılabilir. Uzaktan algılama, geniş alanların hızlı ve etkili bir şekilde değerlendirilmesini sağlayarak, birçok sektörde bilgi ve karar destek sistemlerine katkıda bulunur.

Uydu görüntülerinden gemilerin otomatik tespiti ve sınıflandırılması, deniz trafiği yönetimi, çevre kirliliğinin izlenmesi, uyuşturucu kaçakçılığı, göçmen kaçakçılığı, sınır ihlalleri ve meydana gelebilecek diğer suçların takibi açısından kritik bir araştırma alanıdır. Son yıllarda, DÖ algoritmalarının popüler hale gelmesiyle birlikte, Evrişimli Sinir Ağları (ESA) nesne algılama ve sınıflandırma gibi bilgisayarlı görme problemlerinde başarılı sonuçlar elde etmiştir (Voulodimos et al., 2018).

Bu tez çalışmasında, ESA tabanlı "Sadece Bir Kez Bak" nesne tespit (You Only Look Once, YOLO) üç farklı versiyonunun (YOLOv5, YOLOv7, YOLOv9) uydu görüntülerinde gemi tespiti ve sınıflandırma performansları karşılaştırılmıştır. Araştırma, uydu görüntülerini kullanarak gemileri tespit etmek amacıyla mevcut YOLO tabanlı mimarilerin avantajları ve dezavantajlarını tartışmaktadır. Bu tez çalışması sonucunda uydu görüntülerinden gerçek zamanlı olarak gemilerin tespit

edilerek sınıflandırılması probleminin YOLO tabanlı nesne tespit algoritmaları tarafından çözülebilirliğinin araştırılması amaçlanmıştır.

1.2. Literatür Taraması

Gemilerin uydu görüntüleri üzerinden otomatik tespiti ve sınıflandırılması, deniz güvenliğinde giderek artan bir öneme sahip araştırma alanıdır. Gemilerin tespiti ve manevra davranışları hakkında bilgi edinme yeteneği, gemi trafiğinin yönetimi ve liman giriş-çıkışlarının kontrolü açısından büyük önem taşımaktadır (Stofa et al., 2020). Ayrıca güvenlik güçleri, çevre kirliliği, uyuşturucu kaçakçılığı, göçmen kaçakçılığı ve sınır ihlalleri gibi suç faaliyetlerini izlemek ve soruşturmak için gemi tespit teknolojilerinden yararlanmaktadır. Güvenlik ihlallerinin bilgisayarlar tarafından otomatik olarak tespit edilmesi, iş gücü kaybını ve olası insan hatası nedeniyle oluşabilecek yanlış alarmları azaltmaktadır.

DÖ'de, veri setlerinin kalitesi ve boyutunun model performansı üzerinde önemli bir etkiye sahip olduğu yaygın olarak kabul edilmektedir (LeCun et al., 2015). Örnek sayısı fazla ve görüntü kalitesi açısından çeşitli veri setlerinin, modelin genelleme yeteneğini artırarak daha doğru ve güvenilir sonuçlar elde edilmesine olanak tanıdığı düşünülmektedir. Ayrıca, veri setlerinin çeşitliliği ve temsiliyetinin, modelin farklı senaryoları ve varyasyonları öğrenebilmesi açısından kritik bir rol oynadığı vurgulanmaktadır (Krizhevsky et al., 2012). Bu nedenlerle, DÖ algoritmalarının verimli bir şekilde çalışması açısından veri setlerinin dikkatli bir şekilde hazırlanması ve ön işleme tabi tutulması oldukça önemlidir.

Uydu görüntüleri üzerinden nesne tespiti yapılırken özellikle Sentetik Açıklıklı Radar (SAR) ve optik görüntü teknolojileri, bu alanda yaygın olarak kullanılmaktadır. Bu bağlamda, SAR ve optik görüntülerinden gemi tespiti ve sınıflandırma araştırmalarında kullanılan veri setlerinden bazıları; SAR Ship Detection Dataset (SSDD), High-Resolution SAR Images Dataset (HRSID), AIR-SARShip-1.0 ve Airbus Ship Detection Dataset'dir.

1.2.1. SAR Ship Detection Dataset (SSDD)

SSDD, derin öğrenmeye dayalı SAR gemi tespit teknolojisini araştırmak için oluşturulan ilk açık kaynak veri setidir. Literatüre göre, toplam 161 açık kaynak makalenin %46,59'u, DÖ tabanlı gemi tespitini incelemek için SSDD'yi seçmiştir.

Radar teknolojisiyle elde edilen bu veri seti, çeşitli denizcilik faaliyetlerinin yoğun olduğu bölgelerden alınan yüksek çözünürlüklü SAR görüntülerini içerir. Deniz güvenliğinin artırılması, yasa dışı faaliyetlerin tespiti, çevre koruma ve deniz trafiği yönetimi gibi alanlarda kritik öneme sahiptir. 1160 yüksek çözünürlüklü SAR görüntüsünden oluşan bu veri seti, farklı coğrafi bölgelerden ve deniz alanlarından elde edilmiştir. Görüntülerin çözünürlüğü 1m ile 10m arasında değişmekte olup, boyutları genellikle 256x256 veya 512x512 piksel olarak sağlanmaktadır. Veri setinde genellikle gemilerin konumları ve türleri hakkında etiketlenmiş veriler bulunur ve bu etiketler makine öğrenmesi ve yapay zeka algoritmalarının geliştirilmesinde kullanılmaktadır. SSDD denizcilik sektöründe güvenlik ve verimliliğin sağlanmasına yönelik araştırma ve geliştirme çalışmalarında önemli bir rol oynamaktadır (Zhang et al., 2021).

1.2.2.High-Resolution SAR Images Dataset (HRSID)

HRSID veri seti 0,5 m, 1 m ve 3 m çözünürlüğünde toplada 5604 yüksek çözünürlüklü SAR görüntüsünden ve 16951 gemi örneğinden oluşmaktadır. HRSID SAR görüntülerinden gemi tespiti, anlamsal segmentasyon ve örnek segmentasyon görevleri için kullanılan açık kaynak bir veri kümesidir. HRSID veri seti, Microsoft Common Objects in Context veri seti gibi araştırmacıların yaklaşımlarını değerlendirmeleri için bir referans niteliğindedir. HRSID, askeri gözetim, kritik altyapı güvenliği ve hassas denizcilik operasyonları gibi yüksek hassasiyet gerektiren alanlarda kullanılmaktadır (Wei et al., 2020).

1.2.3.AIR-SARShip-1.0

AIR-SARShip-1.0, gemi tespiti ve sınıflandırma amacıyla oluşturulmuş yüksek çözünürlüklü SAR görüntülerini içeren bir veri setidir. Bu veri seti, özellikle SAR görüntüleri kullanarak denizcilik gözetimi, güvenliği ve çeşitli araştırma çalışmalarını desteklemek için tasarlanmıştır (Xian et al., 2019). AIR-SARShip-1.0 farklı çevresel koşullar ve denizcilik senaryolarında çekilmiş SAR görüntülerini kapsar ve çeşitli gemi tiplerini içeren bir veri setidir. Gemi tespiti ve sınıflandırma için doğru etiketlemeler ve etiketler içerir. Bu veri seti, deniz güvenliği, yasadışı faaliyetlerin tespiti, çevresel izleme, deniz trafiği yönetimi ve askeri gözetim gibi birçok alanda kullanılmaktadır. Veri seti Geofen-3 uydusu kullanılarak büyük sahnelerden oluşan 31 SAR görüntüsünden oluşmaktadır.

1.2.4.Airbus Ship Detection Dataset (ASDD)

ASDD optik uydu görüntülerinden oluşan ve optik görüntülerde gemi tespiti ve sınıflandırma çalışmaları için kullanılan bir veri setidir. Veri seti, 192.556 uydu görüntüsü içermekte olup, bu görüntülerde 817.036 gemi anotasyonu bulunmaktadır (Pires et al., 2022). Airbus Ship Detection Dataset, çeşitli gemi türlerini içeren geniş bir yelpazeye sahiptir ve gemilerin optik görüntülerde tespiti için doğru etiketlemeler sunar. Bu veri seti, gemi tespiti algoritmalarının geliştirilmesi ve değerlendirilmesi için önemli bir kaynaktır. Ayrıca, optik görüntülerde gemi tespiti alanında yapılan çalışmalara katkı sağlamaktadır.

Özetle SSDD, HRSID, AIR-SARShip-1.0 ve ASDD, gemi tespiti ve sınıflandırma için önemli yeri olan açık kaynak veri setleridir, ancak farklı gereksinimlere ve teknolojilere hizmet ederler.

SSDD veri seti geniş kapsamı ve çeşitli etiketleme standartları ile genel araştırmalar için uygundur ve geniş bir kullanıcı kitlesine sahiptir. SAR görüntülerinden oluşur. HRSID veri seti yüksek çözünürlüklü ve ayrıntılı SAR görüntüler sunarak daha hassas ve spesifik tespit ihtiyaçlarına cevap verir. Özellikle küçük ve yoğun dağılımlı gemilerin tespiti için uygundur. AIR-SARShip-1.0 veri seti farklı çevresel koşullar ve gemi tiplerini içeren yüksek çözünürlüklü SAR görüntüleriyle kapsamlı bir kaynak sunar. Denizcilik gözetimi ve güvenliği alanlarında yaygın olarak kullanılır. ASDD veri seti optik uydu görüntülerinden oluşur ve optik görüntülerde gemi tespiti ve sınıflandırma çalışmaları için kullanılır. Veri seti geniş bir gemi türü yelpazesine sahip yüksek çözünürlüklü optik görüntülerden oluşturmaktadır.

Dört veri seti de gemi tespiti ve sınıflandırma alanında yapılan çalışmalara önemli katkılar sağlamaktadır. Araştırmacılar, projelerinin gereksinimlerine ve hedeflerine göre bu veri setlerinden birini veya birkaçını kullanarak daha kapsamlı ve güvenilir sonuçlar elde edebilirler. Bu veri setlerinin sunduğu çeşitli etiketleme standartları ve kullanım protokolleri, gemi tespiti ve sınıflandırma alanındaki teknolojik ilerlemeleri desteklemektedir.

SAR görüntüleri, hava koşullarından ve zamandan daha az etkilendikleri için diğer görüntü türlerine göre tercih edilmektedir (Dragosevic & Vachon, 2008; Eldhuset, 1996; Li & Chong, 2008). Ayrıca, hareketli nesnelerin hızını tespit etmek için SAR görüntüleri daha avantajlıdır (Dragosevic & Vachon, 2008). Ancak SAR görüntüleri

genellikle yüksek düzeyde benekler içermektedir. Bu benekler nesne tespitinde ve sınıflandırma da özellikle küçük nesnelerin tespitinde olumsuz sonuçlar ortaya çıkarabilir.

Yüksek çözünürlüklü optik uydu görüntüleri ise SAR'ın aksine özellikle küçük hedeflerin tespiti ve gemilerin tanınması için daha ayrıntılı bilgi sağladığından sıklıkla tercih edilmektedir (Bo et al., 2021; Kızılkaya et al., 2022; Mehran et al., 2023). Uzaktan algılama teknolojisindeki gelişmelerle birlikte yüksek çözünürlüklü görüntüler elde edilmesine rağmen, optik uydu görüntülerinden gemilerin tespiti sis, dalga ve güneş ışığı yansımaları gibi çeşitli koşullar nedeniyle zorlu bir görev olmaya devam etmektedir. Bu nedenle, bu alanda daha gelişmiş çalışmalara ihtiyaç olduğu açıktır (Bo et al., 2021).

Geçmişte nesne tespiti ve sınıflandırma görevleri için klasik makine öğrenmesi yöntemleri kullanılırken, günümüzde bunların yerini DÖ algoritmaları almıştır (Chai et al., 2021). Özellikle, birçok bilgisayarlı görü uygulamasında dikkate değer başarılar elde eden ESA'lar, gemi tespit ve sınıflandırma görevlerini ele almak için umut verici bir yöntem olarak ortaya çıkmıştır (Bo et al., 2021).

Literatürde bu alanda yapılan bazı çalışmalar aşağıdaki gibidir: 2019 yılında Scholler ve arkadaşları (Ophoff et al., 2020), uzun dalga boyu kızılötesi görüntüleri kullanarak denizde nesne tespiti üzerine bir çalışma yürütmüşlerdir. Çalışmada, deniz yüzeyindeki gemi ve mayınların tespitine odaklanan RetinaNet, YOLO ve Faster RCNN (Region-based Convolutional Neural Network) modelleri kullanılmıştır. Ophoff ve arkadaşları (Patel et al., 2022b), araç ve gemi tespiti için özel olarak oluşturulmuş bir veri seti üzerinde dört farklı YOLO modelinin (YOLOv2, YOLOv3, YOLT ve D-YOLO) güçlü ve zayıf yönlerini karşılaştırmışlardır. Bu çalışmanın sonuçları, YOLT modelinin gemi tespitinde YOLOv3 ve D-YOLO'ya göre daha başarılı olduğunu göstermiştir. Atik ve İpbüker (Atik & İpbüker), AIRBUS Ship Detection veri setini kullanarak, ResNet-101 modeli ile Mask-RCNN mimarisini ile gemilerin sınıflandırılmasında %99,17 başarı oranına ulaşmışlardır. Patel ve arkadaşları (Schöller et al., 2019), HRSID veri setini kullanarak gemileri tespit etmek için Grafik Sinir Ağı (GNN) ve YOLOv7 modelini kullanmışlardır. Sonuçlar, YOLOv7 algoritmasının %93,4 başarı oranına ulaştığını göstermiştir. Patel ve arkadaşları (Patel et al., 2022a), yaklaşık 40.000 uydu görüntüsünden oluşan Airbus

veri kümesini kullanarak YOLOv3, YOLOv4 ve YOLOv5 dahil olmak üzere YOLO algoritmasının farklı versiyonlarının performansını karşılaştırmışlardır. Çalışmada, YOLOv5 nesne algılama algoritmasının diğer YOLO versiyonlarına kıyasla üstün performans gösterdiği ortaya çıkmıştır. YOLOv5 %99, YOLOv4 %98, YOLOv3 ise %97 başarı oranına ulaşmıştır. Kızılkaya ve arkadaşları (Kızılkaya et al., 2022), Google Earth görüntüleri ile oluşturulan bir veri kümesini kullanarak gemi türlerinin tespiti ve sınıflandırılması için YOLOv4 algoritmasını eğitmişlerdir. YOLOv4 modeli gemi tanımda %84,31 doğruluk ve %75,34 F1 puanı elde etmiştir. Jiang ve Arkadaşları, SSDD veri setini ve YOLOv4 modelini kullanarak gemi tespitinde %90,37 başarı oranına ulaşmışlar ve 3 kata kadar daha hızlı gemi tespiti yaptıklarını raporlamışlardır (Jiang et al., 2021). Huang ve arkadaşları küçük gemilerin görüntülerden tespit edilmesindeki zorlukları ele almak için YOLOv4-MSW modeli ve ISDS gemi veri setini kullanarak, gemi tipi tanıma araştırmaları için bir çalışma yapmışlardır. Bu çalışma ile ortalama hassasiyette %4,87, duyarlılık oranında ise %10,03 artış gösterilmiştir. Ancak, çevresel koşullar ve farklı gemi tiplerine sahip coğrafi bölgelere uyarlanma gibi zorluklar halen devam ettiği çalışmada belirtilmiştir (Huang et al., 2024). Sun ve arkadaşları HRRSD veri setini kullanarak YOLOv5s modelinde yapılan iyileştirmelerle küçük gemi hedeflerin tespitini geliştiren yeni bir yöntem önermektedir. Geliştirilen modelin F-puanı %95,17' elde ettiği çalışmada raporlanmıştır (Sun et al., 2022). Congan ve arkadaşları SAR Görüntülerinde Döndürülmüş Gemi Tespit Veri Seti (RSDD-SAR) adı verilen, döndürülmüş bir SAR gemisi tespit veri setini kullanarak, S2ANet modeliyle, %90,06 ile en yüksek ortalama hassasiyete ulaşmışlardır (Congan et al., 2022). Lin ve arkadaşları nesne tespitinde sıkıştırma ve uyarma mekanizmasını kullanarak algılama performansını daha da artırmak için Fast R-CNN'ye dayalı yeni bir ağ mimarisi önermişlerdir. Önerilen modelin Sentinel-1 görüntülerinde algılama performansının 0,836'ya ulaştığını göstermişlerdir. Bu çalışma kapsamında F1 skora bakıldığında en son teknolojiye sahip yöntemlerden %9,7 daha başarılı ve %14 daha hızlı olduğu tespit çalışmada belirtilmiştir (Lin et al., 2018).

Literatüre bakıldığında, YOLO algoritmasının nesne tespitinde kullanıldığı ve başarılı sonuçlar verdiği görülmektedir. Ancak bildiğimiz kadarıyla, uydu görüntülerinden gemi tespiti ve sınıflandırması için daha yeni YOLO mimarilerini değerlendiren ve karşılaştıran bir çalışma bulunmamaktadır. Bu tez çalışmasında, YOLO algoritmanın

üç versiyonunun (YOLOv5, YOLOv7 ve YOLOv9) optik uydu görüntülerinden gemileri tespit etme ve sınıflandırmadaki performansı değerlendirilmiştir. Buna ek olarak küçük YOLOv7 modeli geliştirilerek gemi tespit performansının artışı gerçekleştirilmiştir.

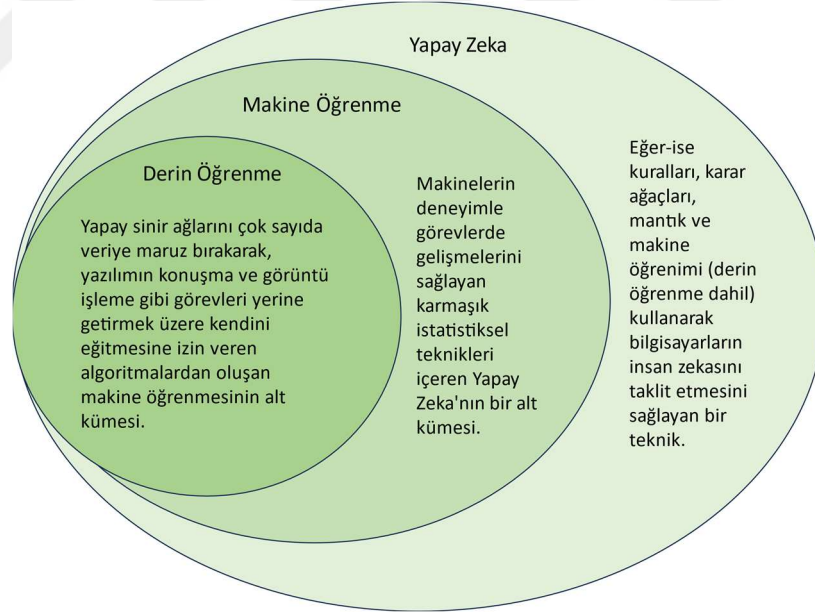


2. EVRİŞİMLİ SİNİR AĞLARI VE NESNE TESPİTİ

2.1. Yapay Zekâ

Yapay zeka (YZ), robotların veya bilgisayar programlarının öğrenme, problem çözme ve karar verme dahil olmak üzere tipik olarak insan zekası gerektiren görevleri üstlenme kapasitesi olarak tanımlanır (Poole & Mackworth, 2010).

YZ başlığı altında farklı araştırma alanları yer almaktadır. Bunlar makine öğrenmesi (MÖ) ve DÖ'dür. MÖ, makinelerin açıkça programlanmadan verilerden öğrenmesini sağlamak için istatistiksel teknikler kullanan bir YZ alt araştırma alanının temsil eder (Ia, 2016). Başka bir deyişle, makinelere kalıpları tanımayı ve verilere dayalı sonuçları tahmin etmeyi öğretme yöntemidir. DÖ verilerden öğrenmek için çok katmanlı yapay sinir ağlarının kullanılmasını baz alan bir MÖ alt araştırma alanıdır. DÖ algoritmaları, büyük miktarda veriyi analiz edebilir ve insanların yeteneklerinin ötesindeki kalıpları ve özellikleri tanımayı otomatik olarak öğrenebilir (Bengio et al., 2021). YZ, MÖ ve DÖ arasındaki ilişki şekil 2.1.'de verilmiştir.



Şekil 2.1 Yapay zekâ, makine öğrenmesi ve derin öğrenme ilişkisi (Ak, 2021)

2.2. Makine Öğrenmesi

Makine öğrenmesi, bilgisayarların veri ve deneyimlerden öğrenerek zamanla performanslarını geliştirmelerine olanak tanıyan bir yapay zeka dalıdır. Bu alanda, çeşitli algoritmalar kullanılarak büyük veri setleri analiz edilir ve bu verilerden belirli

desenler veya ilişkiler öğrenilir. MÖ birçok sektörde geniş bir uygulama yelpazesine sahiptir ve veri odaklı karar verme süreçlerini daha etkili hale getirmeye yardımcı olmaktadır. Denetimli öğrenme, denetimsiz öğrenme ve pekiştirmeli öğrenme gibi farklı MÖ yöntemleri bulunmaktadır. Bu öğrenme yöntemleri farklı gerçek dünya problemlerini çözmek için kullanılmaktadır. Sınıflandırma ve regresyon problemlerinde denetimli öğrenme tercih edilirken, kümeleme, boyut küçültme gibi problemlerde denetimsiz öğrenme kullanılmaktadır. Pekiştirmeli öğrenme ise özellikle robotların bir problemi anlaması ve çözmesi için kullanılan ödül ve ceza sistemine dayanan bir öğrenme türüdür. MÖ yöntemleri yıllar içerisinde birçok problemin çözümünde etkin rol oynamıştır. Ancak MÖ yöntemlerinin bazı dezavantajları bulunmaktadır. MÖ algoritmalarının zayıf yönleri, genellikle veri bağımlılığı, ezberleme, önyargı ve adaletsizlik gibi sorunlardan oluşmaktadır. Bu algoritmalar, kaliteli ve bol miktarda veriye ihtiyaç duyar; eksik veya kirli veri, algoritmanın performansını düşürebilir ve yanıltıcı sonuçlar üretebilir. Ayrıca, modeller genellikle eğitim verisine fazla uyum sağlama eğilimindedir, bu da genelleme yeteneğini azaltabilir. Eğitim verisindeki önyargılar, modelin adaletsiz sonuçlar üretmesine neden olabilir. Özellikle bilgisayarlı görüde klasik MÖ yöntemlerinin performansı görüntüden elde edilen özellikler ile doğru orantılıdır. Farklı ışık altında açılı çekilmiş görüntüler bile modelin genelleme kabiliyetinde düşüşlere sebep olmaktadır (Krizhevsky et al., 2012).

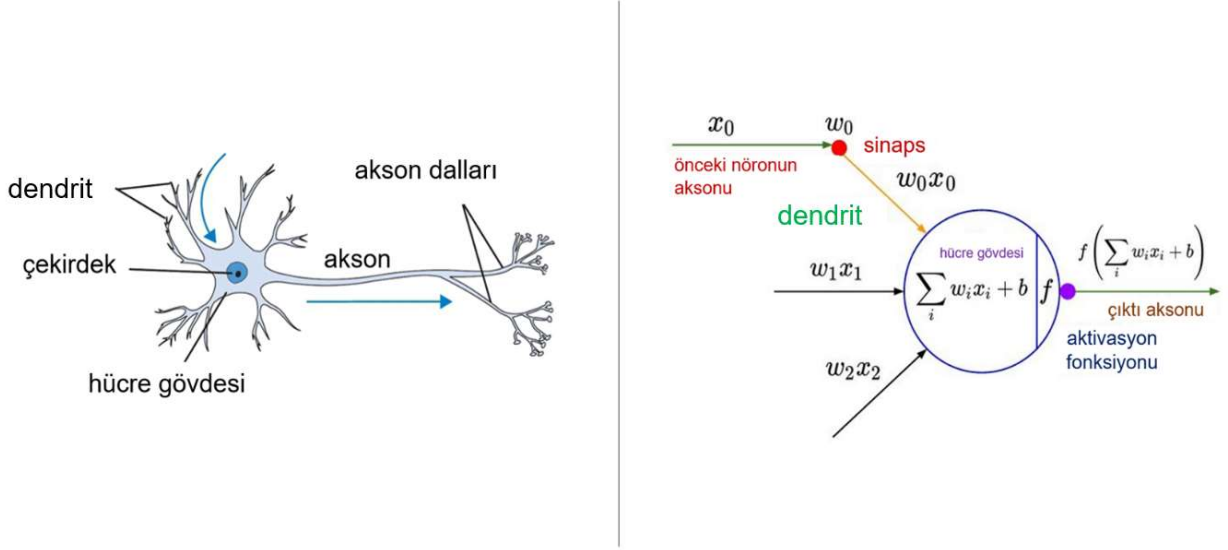
2.3. Derin Öğrenme

Derin öğrenme, MÖ'nün bir alt dalı olup, özellikle yapay sinir ağlarını kullanarak veri analizini gerçekleştiren bir araştırma alanıdır. DÖ, çok katmanlı ağ yapıları (derin sinir ağları) ile karmaşık veri temsili öğrenir ve bu katmanlar aracılığıyla verilerin üst düzey özelliklerini otomatik olarak çıkarabilir. Bu yöntem, görüntü ve ses tanıma, doğal dil işleme ve oyun oynama gibi birçok alanda insanüstü performans göstermektedir. Derin ağlar, büyük veri setleri ve güçlü hesaplama kaynakları gerektirir, ancak büyük veriler ve yüksek işlem gücü sayesinde, detaylı ve yüksek doğrulukta sonuçlar elde edebilirler. Farklı gerçek dünya problemleri için farklı derin ağlar tasarlanmıştır. Örneğin bilgisayarlı görü problemleri için ESA mimarileri kullanılırken, doğal dil işleme problemleri için tekrar eden öz yinelemeli ağlar ve transformatörler kullanılmaktadır. Klasik MÖ yöntemlerinin aksine derin ağlar

problemi çözmek için gerekli olan bilgiyi ham veriden otomatik olarak öğrenmektedir. Bu nedenle genelleme kabiliyetleri MÖ yöntemlerine göre daha yüksektir (Krizhevsky et al., 2012).

2.4. Yapay Sinir ağları

İnsan beyninin temel işlem birimi nöron adı verilen hücrelerdir (şekil 2.2) ve bu hücreler, diğer nöronlarla ortalama 6.000 bağlantı oluşturarak milyarlarca nörondan oluşan karmaşık bir ağı meydana getirir (Schwartz et al., 1991). Nöronlar, girdilerini dendrit adı verilen anten benzeri yapılar aracılığıyla alır, bu girdilerin güçleri, kullanım sıklığına bağlı olarak dinamik olarak değişir, bu da öğrenme ve hafıza süreçlerinde kritik bir rol oynar (Purves, 2019). Gelen sinyaller, nöronun hücre gövdesinde toplanır ve akson boyunca yayılan yeni bir sinyale dönüştürülür, böylece diğer nöronlara iletilir (Bear et al., 2020). Bu biyolojik süreçler, yapay sinir ağlarının (YSA) tasarımında ilham kaynağı olmuştur (şekil 2.2). YSA, insan beyninin bu temel işleyiş biçimlerini taklit ederek veri işleme ve öğrenme yeteneklerini geliştirir. YSA'da, nöronlar benzeri yapılar, çeşitli ağırlıklarla bağlantı kurarak verileri işler ve öğrenme süreci, ağırlıkların güncellenmesiyle gerçekleşir. Bu süreç, derin öğrenme modellerinde, özellikle büyük veri setleri ve yüksek performanslı hesaplama kaynakları kullanılarak karmaşık örüntülerin öğrenilmesi ve tanınmasını sağlar (LeCun et al., 2015). Bu benzerlik, YSA'nın karmaşık veri işleme ve modelleme görevlerinde başarılı bir şekilde kullanılmasını sağlamaktadır.



Şekil 2.2. Gerçek sinir hücresi ve yapay sinir hücresi (Hijazi et al., 2015)

Bir sinir hücresini bilgisayar ortamında taklit edebilmek için yapay sinir ağlarında bazı terimler ve formüller kullanılmaktadır. Bu terimler ve formüller YSA'ların ve dolayısı ile DÖ'nün temellerinin anlaşılması açısından büyük öneme sahiptir.

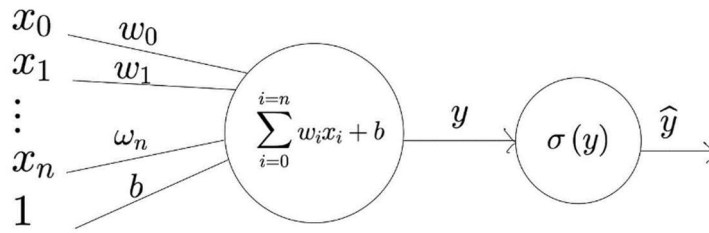
Girdiler: Gerçek sinir hücresinde dentritlerden gelen veriyi temsil etmektedir. Yapay sinir hücresinde ise $x=[x_1,..,x_n]$ adet girdi bir vektör ile temsil edilir. Söz konusu girdi çözülecek probleme ait veriyi temsil etmektedir.

Ağırlıklar: Ağırlıklar, nöronlar arasındaki bağlantıların gücünü belirleyen sinir ağı değişkenleridir. Özetle girdinin çıktıya olan etkisini belirlemektedirler. Her girdinin bir ağırlık değeri bulunmaktadır. Vektörel olarak temsil edilirler ve genelde w sembolü ile gösterilirler. Eğitim sürecinde ağırlıklar, maliyet fonksiyonunu minimize etmek için güncellenirler. Ağırlıklar eğitim başında genellikle rastgele başlatılır ve eğitim süreci sırasında gradyan iniş algoritması gibi yöntemlerle yinelenmeli olarak güncellenir (LeCun et al., 2015).

Önyargı (Bias): Önyargı, giriş ve çıkış değişkenleri arasındaki daha karmaşık ilişkileri öğrenmek için her nörona eklenen bir parametredir. Önyargı, ağırlıklarla birlikte eğitim sürecinde öğrenilir ve güncellenir. Bu parametre, modelin daha esnek ve karmaşık ilişkileri öğrenmesine yardımcı olur (LeCun et al., 2015). Nöronun ateşlenme isteğinin arttırılıp azaltılmasında aktif rol oynar.

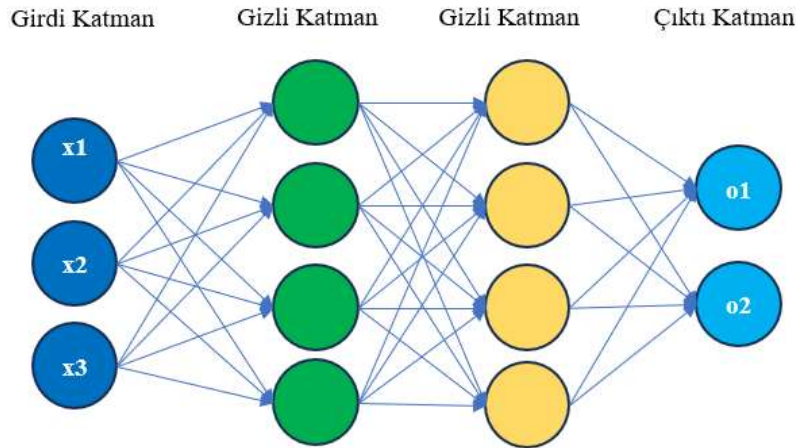
Aktivasyon Fonksiyonu: Yapay sinir ağlarında aktivasyon fonksiyonları, nöronların çıktısını belirleyerek modelin öğrenme ve genelleme yeteneğini şekillendirir. Bu fonksiyonlar, ağı doğrusal olmayan ilişkileri öğrenmesini sağlar ve karmaşık veri yapıları üzerinde etkili bir şekilde işlem yapılmasına yardımcı olur.

Bu bilgiler doğrultusunda yapay bir sinir hücresinde girdiler ile ağırlıklar çarpılıp ön yargı eklendikten sonra aktivasyondan geçen bilgi çıktıyı temsil etmektedir. Üretilen bu çıktı nöronun tahminidir. Yapay bir sinir hücresine ait görsel şekil 2.3'te paylaşılmıştır.



Şekil 2.3. Tek sinir hücresi

Birden fazla yapay sinir hücresinin bir araya gelmesiyle yapay sinir ağları ortaya çıkmıştır. Bu ağlardaki derinlik ve genişlik artırılarak DÖ kavramı ortaya çıkmıştır. Çok katmanlı bir yapay sinir ağına ait bir görsel şekil 2.4'te paylaşılmıştır.



Şekil 2.4. Çok katmanlı yapay sinir ağı örneği

Çok katmanlı YSA'lar geri yayılım denilen bir öğrenme yöntemi kullanarak eğitilirler. Girdilerin ağırlıklar ile çarpılıp, aktivasyon fonksiyonundan geçmesi ile elde edilen tahminlere ileri yayılım denilmektedir. İleri yayılımda her bir girdi örneği için ağ bir

tahmin üretmektedir. Üretilen tahmin ile olması gereken değer arasındaki fark her bir örnek için hesaplanarak ağırlık yapmış olduğu toplam hata elde edilmektedir. Bu hatayı ölçmek için hata fonksiyonu denilen bir fonksiyon kullanılmaktadır. Hata fonksiyonu ya da diğer adıyla maliyet fonksiyonu, modelin tahminlerini eğitim verilerindeki hedef değişkenlerin gerçek değerleriyle karşılaştıran matematiksel bir fonksiyondur. Modeli eğitmenin amacı, bu maliyet fonksiyonunu en aza indirmek için ağırdaki nöronların ağırlıklarını ve önyargılarını ayarlamaktır. Bu fonksiyon, modelin hata miktarını ölçer ve modelin performansını değerlendirmek için kullanılır. Sınıflandırma ve regresyon için farklı hata fonksiyonları kullanılmaktadır. İkili sınıflandırma problemlerinde binary cross entropy, çoklu sınıflandırma problemlerinde cross entropy ve regresyon problemlerinde ise mean square error (MSE) hata fonksiyonları tercih edilmektedir. Hata fonksiyonu kullanılarak elde edilen toplam hata gradyan iniş algoritması kullanılarak ağırlıkların ve ön yargıların güncellenmesinde kullanılmaktadır. Bir ileri bir geri yayılım işlemine bir iterasyon ya da bir epok denilmektedir. YSA'ların eğitimi esnasında belirlenmesi gereken bazı özel hiper parametreler aşağıda belirtilmiştir.

İterasyon (Epoch): Modelin toplam yapağı ileri ve geri yayılım sayısını temsil etmektedir. Bir sinir ağını eğitmek için gereken iterasyon, görevin karmaşıklığı ve veri kümesinin boyutuna bağlı olarak değişir. Genellikle modelin verileri daha iyi öğrenmesi için birden fazla iterasyon gereklidir (Sutskever et al., 2011).

Öğrenme Katsayısı (Learning Rate): Öğrenme katsayısı, gradyan iniş algoritmasının her yinelemede maliyet fonksiyonundaki en aza indirme miktarını belirleyen bir hiperparametredir. Düşük bir öğrenme oranı, yavaş bir yakınsamaya yol açabilirken, yüksek bir öğrenme oranı, modelin minimumu aşmasına neden olabilir. Genellikle 0.1 veya 0.01 gibi küçük değerlerde ayarlanır, ancak bu değer problemin doğasına bağlı olarak değişebilir (Bengio, 2013).

Yığın Boyutu (Batch Size): Yığın boyutu, her gradyan iniş yinelemesinde kaç eğitim örneğinin kullanılacağını belirleyen bir hiperparametredir. Küçük toplu boyutlar, daha hızlı yakınsamaya ve daha az bellek kullanımına yol açabilir, fakat eğitim sürecini daha gürültülü hale getirebilir. Büyük toplu boyutlar ise eğitim sürecindeki gürültüyü azaltabilir ancak daha fazla bellek kullanımı gerektirebilir ve yakınsamaya ulaşma süresi uzayabilir (Ioffe, 2015).

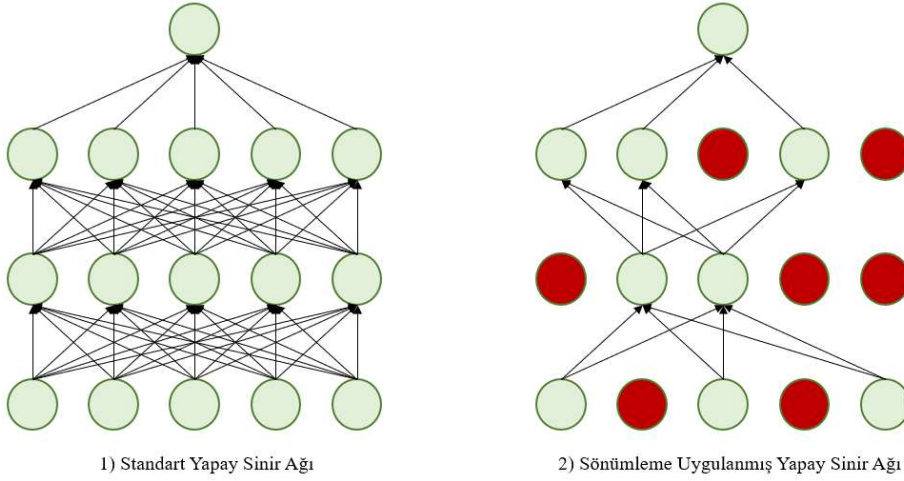
2.4.1. Yapay Sinir Ağlarında Ezberleme ve Ezberlemeyi Engelleme Yöntemleri

Yapay sinir ağlarında ezberleme, modelin eğitim verisine aşırı uyum sağlaması durumudur; bu, modelin eğitim verisindeki ayrıntıları ve gürültüleri öğrenerek yüksek doğruluk gösterirken, yeni ve görülmemiş verilerde düşük performans sergilemesine yol açar. Bu problem genellikle modelin karmaşıklığının arttığı, yani çok fazla katman veya nöron içeren ağlarda ortaya çıkar. Ezberleme (Overfitting), yetersiz veri, aşırı model karmaşıklığı ve düzenleme eksikliği gibi nedenlerle oluşabilir. Bu sorunu önlemek için düzenleme teknikleri (örneğin, L1 veya L2 düzenleme), erken durdurma (early stopping), veri artırma ve çapraz doğrulama gibi yöntemler kullanılır. Bu stratejiler, modelin genelleme yeteneğini artırarak, eğitim verisinin ötesinde başarılı performans göstermesine yardımcı olur.

Veri Artırma (Data Augmentation): Veri artırma, eğitim verilerinin çeşitli modifikasyonlar (dönme, ölçekleme, kırpma vb.) ile çeşitlendirilmesi anlamına gelir. Bu, modelin çeşitli veri varyasyonlarına karşı daha dayanıklı hale gelmesini sağlar ve aşırı uyum riskini azaltır (Wang & Perez, 2017).

Erken Durdurma (Early Stopping): Erken durdurma, modelin eğitim sürecini, doğrulama verileri üzerindeki performansın belirli bir noktada kötüleşmeye başlamasından önce durdurmayı ifade eder. Bu yöntem, modelin aşırı uyum göstermesini engeller ve genelleme performansını artırır (Prechelt, 1998).

Sönümleme (Dropout): Dropout yöntemi, eğitim sırasında rastgele olarak bazı nöronları geçici olarak devre dışı bırakır. Devre dışı bırakılan nöronlar ileri yayılım sırasında hesaba katılmaz ve geri yayılım sırasında güncellenmez, bu da ağın genelleme yeteneğini artırır. Bu, modelin belirli nöronlara bağımlı hale gelmesini önler ve genel özellikleri öğrenmesini teşvik eder. Dropout, sinir ağlarının aşırı uyumunu azaltarak modelin daha iyi genelleştirilmesini sağlar (Hinton et al., 2012).



Şekil 2.5. Sönümleme (Pei et al., 2022)

L1 ve L2 Düzenleştirme: Bu yöntemler, ağırlıklara ceza ekleyerek ağırlıkların büyüklüğünü sınırlamayı hedefler. L1 düzenleştirme, ağırlıkların mutlak değerlerinin toplamını ceza terimi olarak eklerken, L2 düzenleştirme ağırlıkların karelerinin toplamını ceza terimi olarak kullanır. L2 düzenleştirme, ağırlıkların büyük değerlerini ceza uygulayarak sınırlayarak aşırı öğrenmeyi önler (Johansen, 1997).

Model Boyutunu Küçültme: Modelin derinliğini ve genişliği azaltarak güncellenecek parametre sayısını azaltarak ezberlemeyi engellemeyi hedeflemektedir.

2.5. Evrişimli Sinir Ağları (ESA)

Evrişimli Sinir Ağları (Convolutional Neural Networks-CNNs), DÖ paradigmalarının önemli bir bileşenidir ve özellikle görsel veri analizi gibi alanlarda etkili bir şekilde kullanılmaktadır. Bir tür sinir ağı olan ESA'lar, memeli görsel sisteminden esinlenerek geliştirilmiştir. ESA mimarisi, evrişim hesaplamaları ve hiyerarşik özellik çıkarma kapasitesine sahip olmasıyla öne çıkar. Bu sayede özellikle bilgisayarlı görü problemlerinde tercih edilmektedir. ESA mimarilerinin başlıca avantajları arasında, girdi verilerinin uzamsal özelliklerini kullanarak özellik çıkarımı yapabilmesi, ağırlıkların paylaşılmasıyla eğitim süresinin azalması ve modelin genel performansının artması bulunmaktadır. Bu özellikler sayesinde, farklı boyutlardaki ve ölçeklerdeki girdi verilerine uygulanabilirler, bu da çok çeşitli uygulama alanlarına olanak tanır.

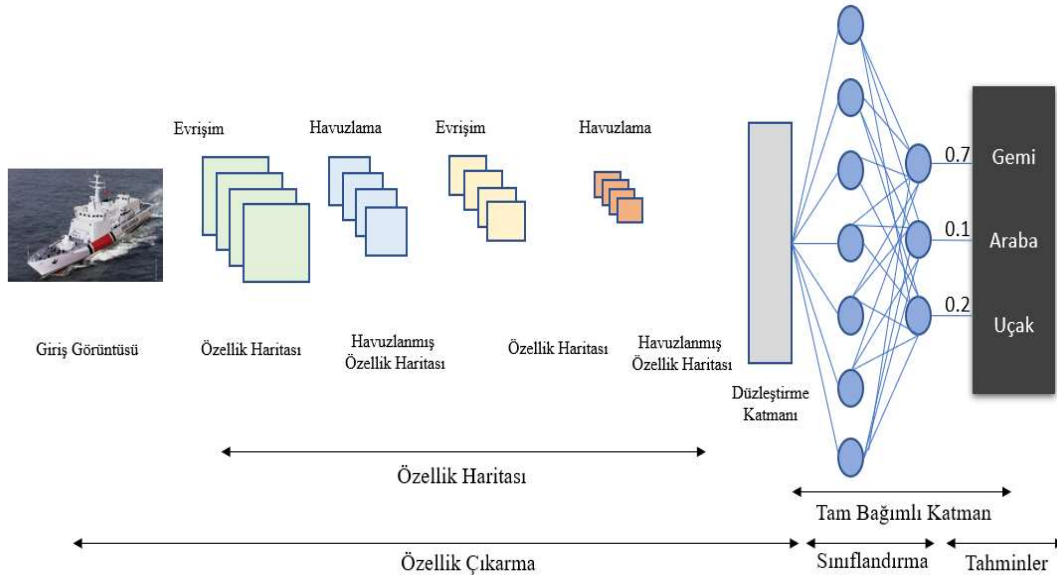
2.5.1. ESA Mimarisi

ESA mimarisi, genellikle giriş katmanı, bir veya birden fazla evrişim katmanı, ardışık havuzlama katmanları, bir veya birden fazla tam bağlantılı gizli katman ve bir çıkış katmanından oluşur. Matematiksel olarak, evrişimli katmanlar giriş verisinde kayan nokta çarpımı veya çapraz korelasyon işlemlerini uygulurlar. Bu işlem, girdi verisinin farklı özelliklerini çıkarmak için bir dizi filtre (kernel) ile yapılan konvolüsyon işlemidir (Krizhevsky et al., 2012). Her evrişimli katmanı takiben genellikle ReLU (Rectified Linear Unit) gibi bir aktivasyon fonksiyonunu kullanılır. ReLU işlevi, negatif değerleri sıfıra eşitlerken pozitif değerleri aynı şekilde bırakır, bu da ağın daha hızlı ve etkili öğrenmesini sağlar.

Evrişim ve aktivasyon işlemlerinden sonra, genellikle havuzlama katmanı gelir. Havuzlama katmanı, evrişimli katmandan gelen özellik haritalarının boyutunu azaltmak için kullanılır. Bu işlem özellik haritalarındaki detayları ortaya çıkarırken ağın yerel hesaplama yükünü azaltır.

Evrişimli katmanlar ve havuzlama katmanları ağ boyunca belirli bir sayıda tekrarlanır. Bu süreç, girdi verisinin hiyerarşik özelliklerini öğrenmeye ve temsil etmeye yöneliktir.

ESA katmanlarının hemen ardından, tam bağımlı (fully connected) katmanlar gelir. Bu katmanlar, kendinden önceki katmanlardan gelen özellikleri kullanarak sınıflandırma veya regresyon gibi nihai bir çıktı görevini gerçekleştirir (Krizhevsky et al., 2012). Klasik bir ESA mimarisine ait görsel şekil 2.6'da verilmiştir.

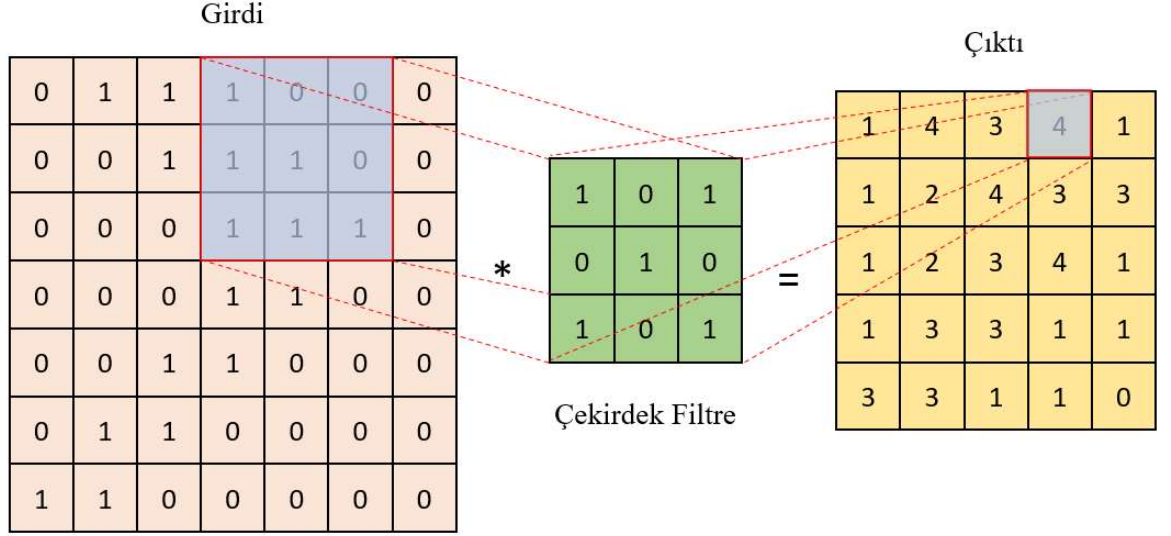


Şekil 2.6. ESA Mimarisi örneği (Yusuf et al.)

2.5.1.1. Evrişim Katmanı

ESA'lar özellikle görüntü işleme alanında kullanılan derin öğrenme algoritmalarının bir türüdür. ESA'nın en kritik bileşeni evrişim katmanıdır. Adını, evrişim işleminin (convolutions) kullanımından alır ve bu işlemler resmin özelliklerini çıkarmak için kullanılır. Evrişim işlemleri, pikseller arasındaki uzamsal ilişkiyi korur ve bu sayede görüntülerin lokal özelliklerini etkili bir şekilde öğrenilmesi sağlamaktadır (Krizhevsky et al., 2012).

Evrişim katmanı, giriş verisi üzerinde bir tür çarpım ve toplam (kernel veya filter) işlemini uygulayarak çalışır. Her bir filtre, giriş verisinde belirli bir özellik veya deseni temsil eder. Evrişim işlemi, bu filtrelerin girdi verisi üzerinde kaydırılarak her konumda yapılmasıyla gerçekleştirilir. Bu işlem sırasında, her filtre için bir çıktı haritası ortaya çıkmaktadır, bu haritalar giriş görüntüsünün farklı özelliklerini temsil eder. Temel bir evrişim işleminin görüntüsü şekil 2.7'de verilmiştir.



Şekil 2.7. Evrişim işlemi

Ağın geri yayılım (backpropagation) sürecinde ise, ağın ileri yayılımı sırasında oluşan hatanın gradyan iniş algoritması kullanılarak geri yayılımı yoluyla filtrelerin ağırlıkları (filtre değerleri) güncellenir. Bu sayede ağ, belirli bir görevi (örneğin, sınıflandırma veya segmentasyon) daha iyi yapacak şekilde kendisini uyarlayarak göreve ait özellikleri otomatik öğrenebilmektedir.

Evrişim katmanının bazı önemli kavramları şunlardır:

Filtre (Kernel): Giriş verisinde belirli desenleri algılayan ve çıkış haritasını oluşturan çeşitli boyutlarda matrislerdir.

Kayma (Stride): Filtrenin girdi görüntüsü üzerindeki kayma miktarıdır.

Doldurma (Padding): Girdi görüntüsünün etrafına eklenen sıfır değerleridir. Görüntünün çıkış boyutunu korumak veya istenilen bir boyuta getirmek için kullanılır.

Bu bilgiler ışığında evrişim uygulanmış bir görüntünün çıktı boyutu formül 1'e göre hesaplanır.

$$n_{\text{çıkı}} = \left\lfloor \frac{n_{\text{girdi}} + 2p - k}{s} \right\rfloor + 1 \quad (1)$$

n_{girdi} : Girdi olarak verilen görüntünün genişlik veya yükseklik değeri

$n_{\text{çıkı}}$: Çıktı görüntüsünün yükseklik ve genişlik değeri

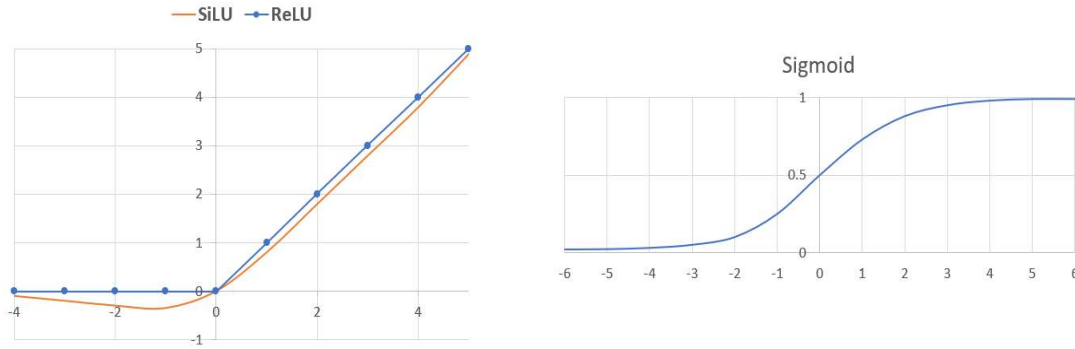
k : Filtrenin yükseklik veya genişlik değeri

p : Doldurma miktarı

s : Filtrenin görüntü üzerinde kayma miktarı

2.5.1.2. Aktivasyon Katmanı

Aktivasyon katmanı, her bir nöronun çıkışını doğrusal olmayan bir şekilde dönüştürerek, ağı karmaşık ilişkileri ve desenleri öğrenmesini sağlar. Bu, modelin veri içindeki karmaşık ve soyut özellikleri yakalamasına yardımcı olur. Evrişim katmanları, giriş verisinde filtrelerle yapılan konvolüsyon işlemlerini uygulayarak özellik haritaları oluşturur. Bu işlem doğrusal bir matematiksel operasyon olarak değerlendirilebilir çünkü giriş verisinde nokta çarpımı işlemi yapılır. Aktivasyon fonksiyonları, evrişim katmanlarının çıktılarını non-linear hale getirerek ağı daha karmaşık ilişkileri öğrenmesine olanak tanır. Bu sayede genelleme yeteneği artar ve ağ, daha çeşitli örüntüleri tanımlamayı öğrenebilir. Literatürde TanH, sigmoid, ELU, ReLU, ELAN, SiLU gibi farklı aktivasyon fonksiyonları tercih edilmiştir (Şekil 2.8.).



Şekil 2.8. Aktivasyon fonksiyonları

Ancak özellikle ReLU (Rectified Linear Unit) aktivasyon fonksiyonu ESA mimarilerinde sıkça kullanılır. ReLU, negatif giriş değerlerini sıfıra eşitlerken pozitif değerleri doğrudan bırakır. Bu özellik, ağı eğitim sürecini hızlandırır ve gradyan kaybolma problemini azaltır ve eğimlerin ileriki katmanlara daha sağlıklı bir şekilde iletilmesini sağlar. Formül 2'de RELU gösterilmiştir

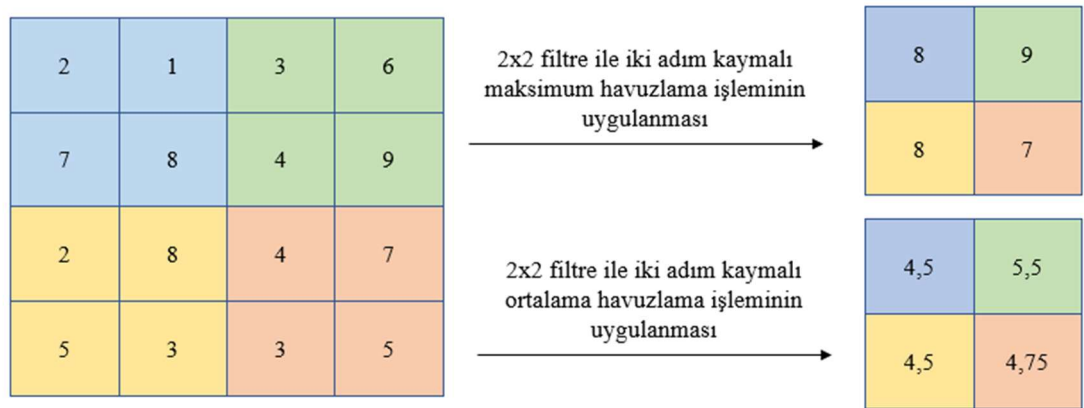
$$f(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \quad (2)$$

2.5.1.3. Havuzlama Katmanı

Girdi görüntüleri evrişim katmanından geçtiğinde, alıcı alan nispeten küçük ve adım uzunluğu nispeten kısa olduğunda, elde edilen özellik haritası hala büyük boyutlarda olacaktır. Bu durumda, boyut küçültme işlemi havuzlama katmanı aracılığıyla gerçekleştirilir ve bu işlem sonucunda çıktının derinliği değişmez; özellik haritası sayısı da aynı kalır ancak girdinin boyunda ve yüksekliğinde hatırı sayılır bir küçülme meydana gelir.

Genellikle maksimum havuzlama ve ortalama havuzlama olmak üzere iki tür havuzlama yöntemi kullanılır: Maksimum havuzlamada (Max pooling) alıcı alanındaki matrisin içinde yer alan maksimum değer alınır. Ortalama havuzlamada (Average pooling) alıcı alanındaki matrisin ortalama değerini alır.

Tarama işlemi, aynı evrişim katmanında olduğu gibi soldan sağa, ardından bir adım aşağı doğru hareket ederek devam eder ve görüntünün sonuna kadar bu şekilde sürdürülür. Burada gözden kaçırılmaması gereken önemli nokta havuzlama filtrelerinin içinde ağırlık değerleri yoktur. Süreç şekil 2.9’da gösterilmektedir:



Şekil 2.9. Havuzlama işlemi örnekleri

Şekil, filtre boyutunun sırasıyla 2x2 ve adımın 2 olduğu maksimum havuzlama işlemini göstermektedir. Görüldüğü gibi 4x4 giriş boyutu 4 parçaya bölünmekte ve her parça kendi en büyük numarası ile sunulmaktadır. Yani çıktı 2x2 özellik haritasıdır.

2.5.1.4. Tam Bağımlı Katmanlar (Fully Connected Layer)

Evrişimli ve havuzlama katmanlarından gelen özellikleri alır ve bu özellikler arasındaki ilişkileri öğrenerek veriyi anlamlı bir şekilde temsil eder. Bu katmandaki her nöron, önceki katmandaki tüm nöronlara bağlıdır ve bu bağlantılar, öğrenme süreci

boyunca optimize edilen ağırlıklar tarafından belirlenir (Krizhevsky et al., 2012). Bu sayede model, karmaşık ilişkileri ve kalıpları öğrenebilir. Tam bağlantı katmanı, derin öğrenme modellerinde, özellik bilgisi kaybını en aza indirmek ve özellikleri daha etkili bir şekilde yeniden düzenlemek amacıyla kullanılır. Bu katman, giriş verilerini yüksek boyutlu bir uzaya yansıtır ve öğrenilen ağırlıklar(özellikler) aracılığıyla verilerin sınıflandırma veya regresyon gibi görevler için optimize edilmesini sağlar. Çıktı katmanı ise modelin nihai hedef sonuca ulaşması için tasarlanmıştır. Bu katman, modelin amacına bağlı olarak farklı işlevler üstlenebilir: Sınıflandırma problemlerinde: Genellikle softmax veya sigmoid aktivasyon fonksiyonları kullanılarak belirli sınıflara olasılık dağılımları üretilir. Regresyon problemlerinde: Lineer aktivasyon fonksiyonları kullanılarak sürekli değerler tahmin edilir.

2.5.1.5. Yığın Normalleştirme Katmanı (Batch Normalization)

Ağın eğitimi sırasında ağırlıkların güncellenmesi, her katmandaki veri dağılımının sürekli değişmesine neden olur; çünkü bir katmanın ağırlıklarının güncellenmesi, sonraki katmanların veri dağılımını etkiler. Örneğin, ağın ikinci katmanını ele alalım: Bu katmanın girişi, birinci katmanın ağırlıkları ve girişi ile hesaplanır. Birinci katmanın ağırlıkları eğitim süreci boyunca sürekli değiştiğinden, ikinci katmanın giriş verilerinin dağılımı da sürekli olarak değişecektir. Bu durum, ağın eğitim sürecinde orta katmanlarda veri dağılımındaki değişiklik olarak tanımlanır ve "İç Değişken Kayması" (Internal Covariate Shift) olarak adlandırılır (Santurkar et al., 2018). Bu sorunu çözmek için Toplu Normalleştirme (Batch Normalization, BN) geliştirilmiştir (Ioffe, 2015).

Aktivasyon katmanı, evrişim katmanı, tamamen bağlı katman ve havuzlama katmanı gibi Toplu Normalleştirme katmanı da ağın bir bileşenidir. Daha önce, giriş ve çıkış katmanları dışında diğer katmanların da eğitim sırasında parametrelerini güncellediğini ve bunun da sonraki katmanlardaki giriş verilerinin dağılımında değişikliklere neden olduğunu belirtmiştir. BN katmanı, üst katman ağının çıktısını ortalama 0 ve varyans 1 olacak şekilde normalleştirir ve bu normalize edilmiş çıktıyı alttaki katmana girdi olarak sağlar. Bu şekilde, BN katmanı "İç Değişken Kayması" sorununu çözmeye yardımcı olur.

2.5.2. ESA'lar ile Nesne Tespiti

Geleneksel nesne tespiti yöntemleri, temel olarak görüntü tanıma tekniklerine dayanmaktadır. Genel bir süreç olarak, bu yöntemlerde ilk adım, resim üzerinde nesnelerin bulunabileceği olası bölge çerçevelerinin seçimidir. Seçilen bu bölge çerçeveleri üzerinden öznitelikler çıkarılır ve ardından bu öznitelikler, genellikle görüntü sınıflandırma yöntemleriyle analiz edilir. Başarılı bir sınıflandırma sonrasında, maksimum olmayan değer bastırma gibi teknikler kullanılarak çıkış sonuçları düzenlenir. Şekil 2.10'da nesne tespitine ilişkin bir görsel paylaşılmıştır. Nesne tespit algoritmaları iki aşamalı ve tek aşamalı olmak üzere ikiye ayrılırlar. Bölge tabanlı ESA'ların temel fikri, önce bir görüntü içinde nesne içerebilecek bir dizi aday bölge veya bölge önerisi oluşturmaktır. Ardından, her bölgeyi bir nesne içerip içermediğini belirlemek için bir ESA kullanarak sınıflandırır. Nesne içerdiği belirlenen bölgeler, daha sonra nesnenin konumunu iyileştirmek ve belirli bir sınıfa atamak amacıyla ek işleme tabi tutulur. Bu yaklaşım "İki Aşamalı Nesne Tespiti" (Two-shot Object Detection) olarak adlandırılır. İki aşamalı nesne tespit algoritmalarında nesne tespit doğruluğu yüksek iken gerçek zamanlı çalışma hızı düşüktür. Bu algoritmalara örnek olarak R-CNN, Fast-R-CNN ve Faster-R-CNN gösterilebilir. İki aşamalı nesne tespit algoritmalarının gerçek zamanlı tespit hızı dezavantajını ortadan kaldırmak için tek aşamalı nesne tespit algoritmaları geliştirilmiştir. Bu algoritmalara YOLO ailesi örnek verilebilir.

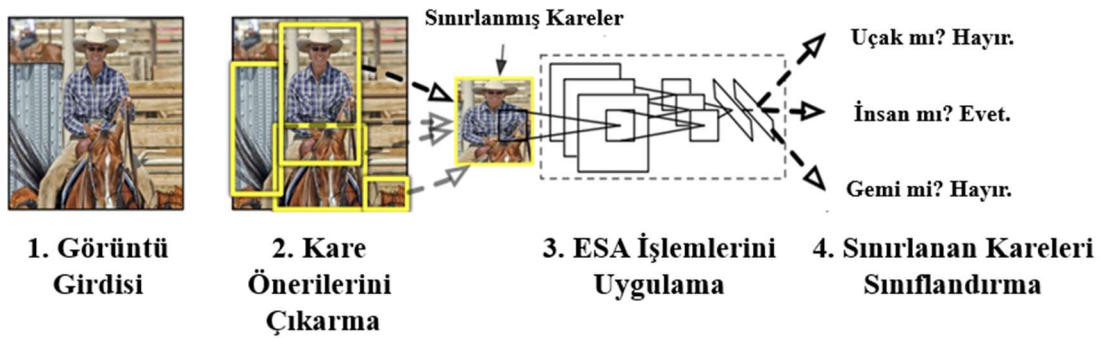


Şekil 2.10. Görüntü tanıma, tespit ve yerelleştirme

2.5.2.1. R-CNN

R-CNN (Region-based Convolutional Neural Networks) mimarisi Girshick ve arkadaşları tarafından 2014 yılında geliştirilmiştir (Girshick et al., 2014). İki aşamalı

bir nesne tespit algoritmasıdır. Algoritmada her bir görüntü için seçici arama (selective search) adı verilen bir yöntemle yaklaşık 2000 aday bölge önerisi üretilir. Bu aday bölgeler daha sonra sabit boyutlara yeniden boyutlandırılarak bir derin ESA modeline girdi olarak verilir. ESA tarafından öznelilik çıkarımı yapılır ve elde edilen özellik vektörleri, her bir bölge için nesnelerin olasılık değerlerini tahmin etmek için çok sınıflı bir destek vektör makinesi sınıflandırıcısına beslenir. Ayrıca, konumlandırma doğruluğunu artırmak için R-CNN, bir sınırlayıcı kutu regresyon modeli kullanarak sınırlayıcı kutu konumlarını düzeltir.



Şekil 2.11. ESA ile kare sınırlama ve sınıflandırma (Yadav et al., 2022)

2.5.2.2. Fast R-CNN

R-CNN (Region-based Convolutional Neural Network) yapısında, tam bağlantılı katmanın giriş görüntü boyutları üzerindeki kısıtlamalar sebebiyle, tüm aday bölgelerin özellik çıkarımı için evrişimli sinir ağına sokulmadan önce deformasyon dönüşümlerinden geçirilmesi gerekmektedir. Ancak bu süreçte, kırpma veya çözünürlük azaltma gibi yöntemler kullanılsa bile, orijinal görüntü bilgisinin tam olarak korunması mümkün olamamaktadır. Bu sorunu çözmek için uzamsal piramit havuzlama (Spatial Pyramid Pooling, SPP) katmanı önerilmiştir, bu katman geleneksel evrişimli sinir ağı yapılarında giriş görüntüsünün boyut kısıtlamalarını etkin bir şekilde aşmaktadır. Görüntü bozulması sorunu bu şekilde aşılmış olsa da, R-CNN'nin hala başka sorunları mevcuttur; örneğin, eğitim adımlarının yavaş olması ve büyük disk alanı gereksinimi gibi. Bu nedenle, R-CNN'nin geliştiricileri Ross ve Girshick, bu sorunlara çözüm olarak Hızlı R-CNN (Fast R-CNN) yöntemini önermiştir (Girshick, 2015). Bu yeni yaklaşım, SPP-Net'in özelliklerini bütünleştirerek hedef tespit hızını önemli ölçüde artırmıştır. Fast R-CNN'in en büyük avantajı, tek bir ileri geçişte hem

sınıflandırma hem de konum tespiti yapabilmesidir, bu da hesaplama süresini önemli ölçüde azaltmaktadır.

2.5.2.3.Faster R-CNN

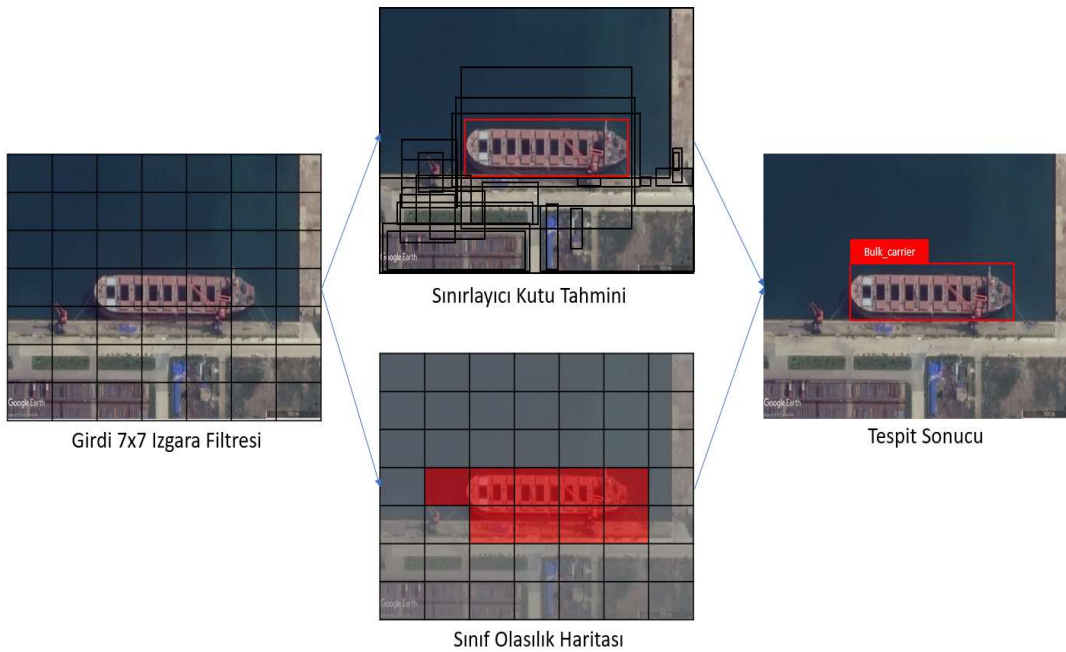
Faster R-CNN, nesne tespiti görevlerinde kullanılan ve Fast R-CNN'in geliştirilmiş bir versiyonu olan bir derin öğrenme modelidir. 2015 yılında Shaoqing Ren ve arkadaşları tarafından tanıtılan Faster R-CNN, nesne tespiti sürecini daha hızlı ve daha verimli hale getirmek için entegre bir Bölge Öneri Ağı (Region Proposal Network, RPN) kullanır (Ren et al., 2016). Bu model, girdi olarak aldığı bir görüntüyü ESA ile işler ve özellik haritası oluşturur. RPN, bu özellik haritası üzerinde çalışarak potansiyel nesne bölgelerini belirler ve bu bölgeleri ROI (Region of Interest) havuzlama katmanına gönderir. ROI havuzlama katmanı, bu bölgeleri sabit boyutlu özellik vektörlerine dönüştürür ve tam bağlantılı katmanlarda işlenir. Son olarak, her bir bölge için sınıf tahmini ve sınır kutusu iyileştirmesi yapılır. Faster R-CNN, RPN ve Fast R-CNN'i tek bir ağda birleştirerek hem bölge önerilerini hem de nesne tespitini aynı anda yapar, bu da modelin doğruluğunu artırırken işlem süresini önemli ölçüde azaltır.

Bölge tabanlı ESA'ler nesne tespiti görevinde tatmin edici bir doğrulukla çözebilse de çeşitli zayıf yönleri vardır. Bu zayıf yönlerden bazıları; verimsizlik, sınırlı genelleme, yüksek hesaplama maliyetidir. Hesaplama maliyetinin özellikle yüksek olması nedeniyle gerçek zamanlı çözüm gerektiren problemlerin çözümünde tercih edilmemektedir.

2.5.3.YOLO

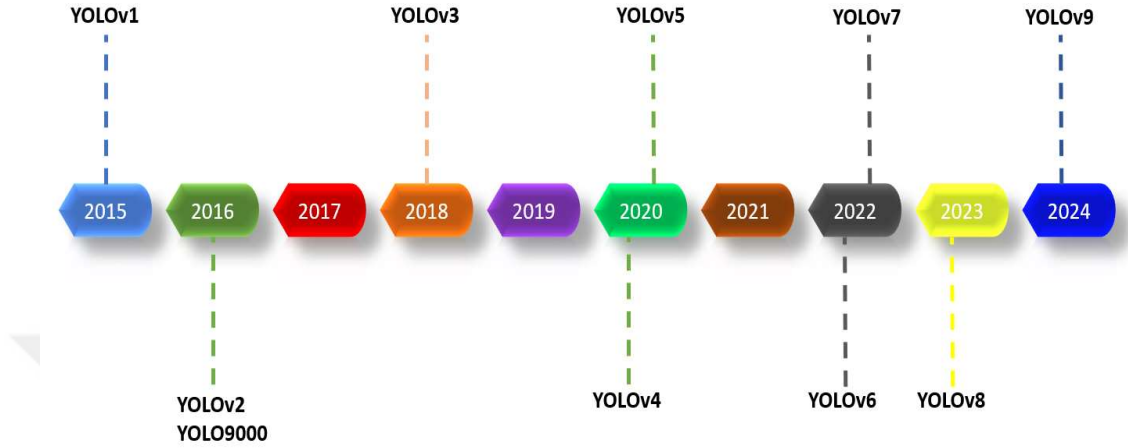
YOLO, nesne tespit alanında önemli bir dönüm noktası olan bir evrişimli sinir ağıdır. Joseph Redmon tarafından 2015 yılında ilk versiyonu sunulan YOLO, özellikle gerçek zamanlı nesne tespiti için etkili bir çözüm sunmaktadır (Redmon et al., 2016). YOLO'nun öne çıkan özellikleri, geleneksel iki aşamalı nesne algılama yöntemlerine alternatif bir yaklaşım getirmesidir. YOLO'nun en temel farkı, nesne algılama problemini bir regresyon problemine dönüştürmesidir. Bu yaklaşım, bir giriş görüntüsü verildiğinde, görüntünün farklı bölgelerindeki nesnelerin sınırlayıcı kutularını ve bu nesnelerin sınıflandırma kategorilerini tek bir aşamada doğrudan tahmin edebilmesidir. Bu, diğer yöntemlerden (örneğin R-CNN ailesi gibi iki aşamalı yaklaşımlar) daha hızlı ve daha verimli bir nesne tespiti sağlar. YOLO nesne tespit aşmalarına ilk olarak girdi görüntüsünü $N \times N$ boyutunda eşit ızgaralara bölerek

başlamaktadır. Şekil 2.12'ye göre bizim örneğimizde N değerimiz 7'dir. Her bir ızgara, nesnenin merkezinin onun kendi içinde olup olmadığını tahmin etmekten sorumludur. Bu amaçla, her bir ızgara hücresi B sayıda sınırlayıcı kutu ve bu kutular için güven skorları tahmin eder. Güven skoru, modelin bir kutunun içinde bir nesne olup olmadığından ne kadar emin olduğunu gösterir. Ayrıca, her ızgara hücresi C sayıda koşullu sınıf olasılığı tahmin eder (potansiyel nesneler için sınıf başına bir tane). Bu olasılıklar, kutunun içinde bir nesne bulunduğu varsayımına dayanır. Sınırlayıcı kutular ve sınıf olasılıkları tahmin edildikten sonra, son işleme adımları uygulanır. Bu adımlardan biri olan Non-Max Suppression (Maksimum Olmayan Bastırma) işlemi, örtüşen sınırlayıcı kutularının sayısını azaltmaya yardımcı olur. Maksimum Olmayan Bastırma, en yüksek güven skoruna sahip kutu ile yüksek örtüşme gösteren diğer sınırlayıcı kutuları ortadan kaldırarak her bir nesne için tek bir sınırlayıcı kutu ortaya çıkmasını sağlar. Nihai çıktıda görüntüdeki her nesne için öngörülen sınırlayıcı kutular ve sınıf etiketleri yer almaktadır. YOLO'nun en önemli avantajlarından biri, görüntünün tamamını tek geçişte işleyerek nesne tespiti, R-CNN ve türevleri gibi iki aşamalı nesne tespit algoritmalarından daha hızlı ve daha verimli hale getirmesidir (Redmon et al., 2016).



Şekil 2.12. YOLO nesne tespit adımları

Yıllar içerisinde Yolov2, Yolov3, Yolov4, Yolov5, Yolov6, Yolov7, Yolov8, Yolov9 olmak üzere farklı versiyonları geliştirilmiştir. Bu çalışma kapsamında Yolov5, Yolov7 ve Yolov9 modelleri değerlendirilmiştir.



Şekil 2.13. YOLO algoritmaları tarihsel gelişimi

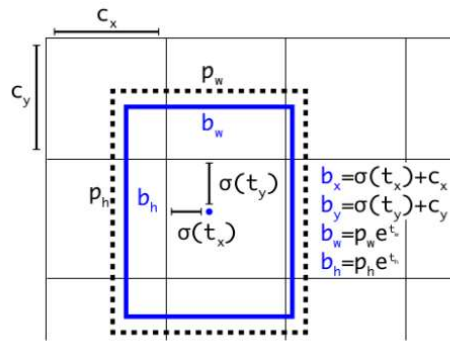
2.5.3.1. YOLOv5

YOLOv5 mimarisi 2020 yılında Ultralytics tarafından Pytorch kütüphanesiyle geliştirilmiştir. YOLOv5 nesne algılama modellerinin beşinci versiyonudur (Jocher et al., 2022). YOLOv5 mimarisi omurga, boyun ve kafa olmak üzere üç ana parçadan oluşmaktadır.

Omurga: Omurga ağın ana gövdesidir. YOLOv5 için omurga, önceki sürümlerde kullanılan Darknet mimarisinin bir modifikasyonu olan yeni CSP-Darknet53 (Cross Stage Partial (CSP)) yapısı kullanılarak tasarlanmıştır. Omurga, görüntüler için zengin özellik temsili çıkarmak amacıyla kullanılan önceden eğitilmiş bir ağıdır. Görüntünün uzamsal çözünürlüğünü azaltmaya ve özellik (kanal) çözünürlüğünü artırmaya yardımcı olmaktadır. Omurgada kullanılan CSPNet mimarisi, DenseNet'in özelliklerin yeniden kullanım avantajını korurken ve gradyan akışını keserek aşırı miktardaki gereksiz gradyan bilgisinin azaltılmasına yardımcı olur. Bu stratejinin uygulanması YOLOv5'e büyük avantajlar sağlamaktadır. Bu yöntem, parametre sayısını azaltarak ve önemli ölçüde hesaplama yükünü (daha az FLOPS) düşürerek tahmin hızını artırır. Bu, gerçek zamanlı nesne algılama modellerinde kritik bir faktördür ve bu sayede modelin daha hızlı çalışmasını sağlar.

Boyun: Mimaride boyun bölgesi omurga ile başı birbirine bağlamaktadır. Boyun özellik piramitlerini çıkarmak için kullanılır. Bu, modelin farklı boyut ve ölçeklerdeki nesnelere iyi bir şekilde odaklanmasına yardımcı olur. YOLOv5 modeli boyun kısmına iki büyük değişiklik getirmiştir. Bu değişiklikler SPPF (Spatial Pyramid Pooling) ve yeni CSP-PAN (Path Aggregation Network) yapılarıdır. PANet, bilgi akışını iyileştirmek ve sınırlayıcı kutu tahminlerinde piksellerin doğru şekilde konumlandırılmasını sağlamak amacıyla YOLOv4'te kullanılan bir özellik piramit ağıdır. YOLOv5'te ise PANet ağı, CSPNet stratejisi ile güncellenmiş ve bu stratejiye göre uyarlanmıştır. SPP bloğu, girdilerden aldığı bilgiyi birleştirir ve sabit uzunlukta bir çıktı döndürür. Böylece, alıcı alanı önemli ölçüde artırma ve en önemli bağlam özelliklerini ayırma avantajına sahiptir, ancak ağın hızını düşürmez. Bu blok, önceki YOLO sürümlerinde (YOLOv3 ve YOLOv4) omurgadan en önemli özellikleri ayırmak için kullanılmıştır. Ancak, YOLOv5 sürümünde ağı hızını artırmak amacıyla, SPP bloğunun bir varyantı olan SPPF kullanılmıştır.

Kafa: Bu kısım son çıktıyı üretmekten sorumludur. YOLOv5 bu amaçla YOLOv3 ve YOLOv4'te kullanılan kafa yapısını kullanmaktadır. Üç konvolüsyon katmanından oluşur ve bu katmanlar sınırlayıcı kutuların (x, y, yükseklik, genişlik), puanların ve nesne sınıflarının konumunu tahmin eder. Eski modellere nazaran bu kısımdaki hesaplama formülleri aşağıdaki gibi güncellenmiştir (şekil 2.14).

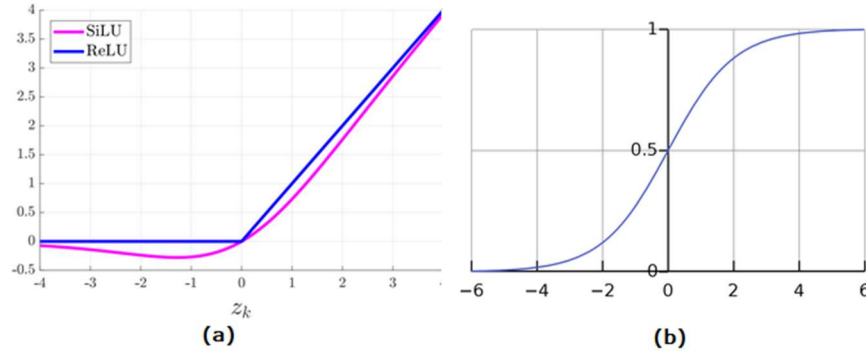


$$\begin{aligned}
 b_x &= \sigma(t_x) + c_x & b_x &= (2 \cdot \sigma(t_x) - 0.5) + c_x \\
 b_y &= \sigma(t_y) + c_y & b_y &= (2 \cdot \sigma(t_y) - 0.5) + c_y \\
 b_w &= p_w \cdot e^{t_w} & b_w &= p_w \cdot (2 \cdot \sigma(t_w))^2 \\
 b_h &= p_h \cdot e^{t_h} & b_h &= p_h \cdot (2 \cdot \sigma(t_h))^2
 \end{aligned}$$

(a) (b)

Şekil 2.14. YOLOv5 sınırlayıcı kutu hesaplamaları (Mitra et al., 2023; Redmon & Farhadi, 2017)

Aktivasyon fonksiyonu: Yolov5'te ReLU yerine SiLU aktivasyon fonksiyonu kullanılmıştır. Kullanılan aktivasyon fonksiyonu şekil 2.15'te paylaşılmıştır.



Şekil 2.15. SiLU aktivasyon fonksiyonu

Hata fonksiyonları: Yolov5'in hata fonksiyonu üç ayrı hata fonksiyonun birleşiminden oluşmaktadır.

Sınıflandırma hatası(BCE Loss): Nesnelerin sınıflandırılmasında ortaya çıkan hatayı ölçer, binary cross entropy fonksiyonu kullanır.

Nesne hatası (BCE Loss): Bir ızgara içerisinde nesnenin var olup olmadığının tespit edilmesinde kullanılan hatadır. Binary cross entropy kullanılır.

Konum hatası (CIoU Loss): Izgara içindeki nesnenin lokalizasyonunda kullanılan hatadır. Intersection Over Union (IOU) yöntemi kullanılarak hesaplanır.

Modelin genel hata fonksiyonu işe formül 3'te verilmiştir.

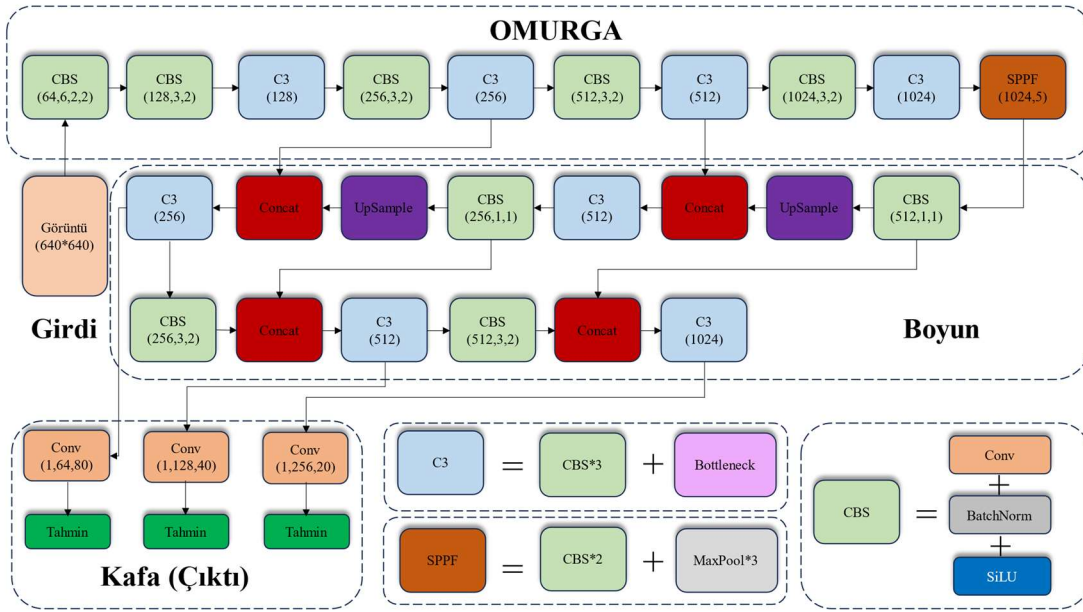
$$Loss = \omega_1 L_{cls} + \omega_2 L_{obj} + \omega_3 L_{loc} \quad (3)$$

Modelin üç farklı katmanda (P3, P4, P5) çıktı vermesinden dolayı nesne hatası kayıpları ağırlıklandırılarak [4, 1, 0.4] hesaplanmaktadır.

$$L_{obj} = 4L_{obj}^{small} + 1L_{obj}^{medium} + 0.4L_{obj}^{large} \quad (4)$$

Yolov5, nano (n), küçük (s), orta (m), büyük (l) ve ekstra büyük (x) olmak üzere dört farklı mimaride sunulmuştur. Küçük modellerin hesaplama karmaşası küçük iken tespit performansları düşmektedir. Büyük modeller ise daha karmaşık ancak tespit performansı küçük modellere göre daha iyidir. Bu tez kapsamında Yolov5'in n, s, m

ve 1 modelleri uydu görüntülerinden gemilerin tespiti için değerlendirilmiştir. Yolov5s'e ait bir görsel Şekil 2.16'da paylaşılmıştır.

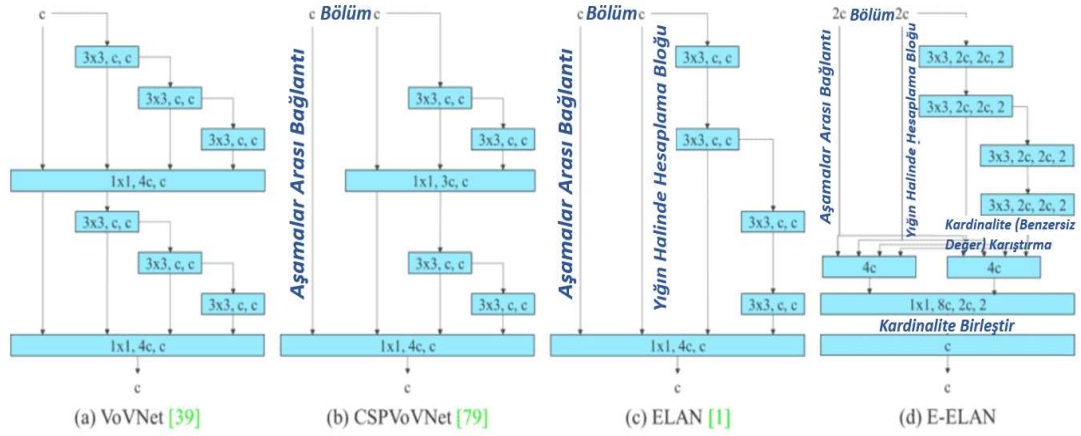


Şekil 2.16. YOLOv5s mimarisi (Liu et al., 2022)

2.5.3.2. Yolov7

YOLO serisinin yedinci versiyonu olan bir nesne tespit mimarisidir. 2022 yılında Chien-Yao Wang ve arkadaşları tarafından tanıtılan bu model, gerçek zamanlı nesne tespitinde geçmiş mimarilere yapılan bazı geliştirmeler ile elde edilmiştir. Söz konusu geliştirmeler aşağıdaki gibidir (Wang et al., 2023).

Extended Efficient Layer Aggregation (E-ELAN): Verimli bir ağ tasarlarlarken, tasarımcılar genellikle hesaplama sayısını ve hesaplama yoğunluğunu optimize etmeyi dikkate alırlar. Bu bağlamda tasarlanan ELAN modülleri, derin ağların etkili bir şekilde öğrenip yakınsamasını sağlamak için en kısa ve en uzun gradyan yolunu kontrol ederek verimli bir ağ tasarlamayı dikkate almaktadır. ELAN'ın modül diyagramı şekil 2.17'de gösterilmiştir.

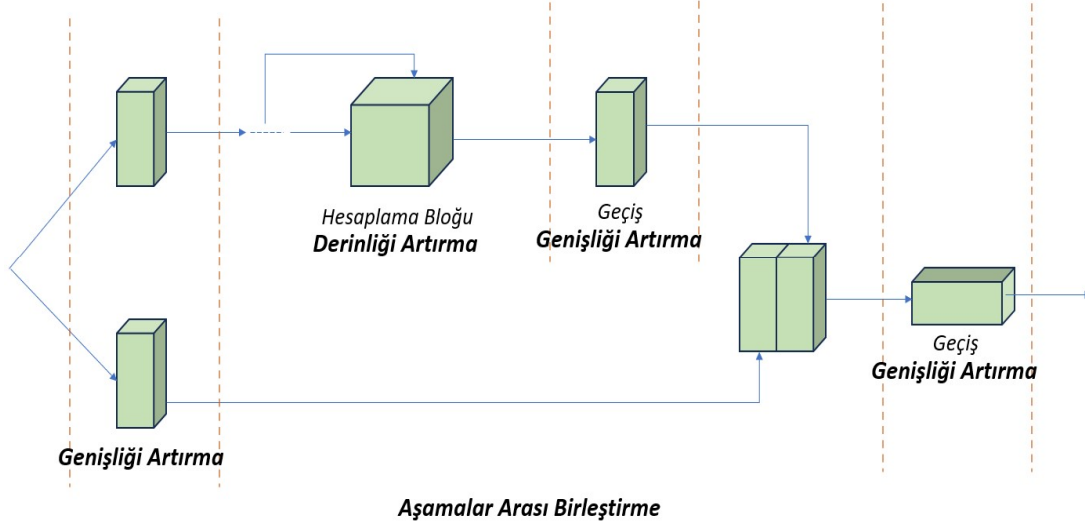


Şekil 2.17. E-ELAN modülleri (Wang et al., 2023)

YOLOv7, ELAN mimarisini modifiye ederek E-ELAN (genişletilmiş verimli katman toplama ağı) adını verdiği yeni bir yapı kullanır. ELAN, model öğrenme yeteneğini gradyan akış yollarını bozmadan geliştirmek için genişletme, karıştırma ve birleştirme kardinalitesi kullanır. Mimari açısından, yalnızca hesaplama bloğundaki mimariyi değiştirir, geçiş katmanındaki mimari ise ELAN ile aynı kalır. E-ELAN, hesaplama bloğunun kanallarını ve kardinalitesini genişletmek için grup evrişimini kullanır. Aynı kanal çarpanı ve grup parametresi, bir hesaplama katmanındaki tüm hesaplama bloklarına uygulanır. Her hesaplama bloğundan elde edilen özellik haritası grup boyutu g'ye göre karıştırılır ve ardından birleştirilir. Son olarak, karıştırılmış grup özellik haritası birleştirme kardinalitesini gerçekleştirmek için eklenir.

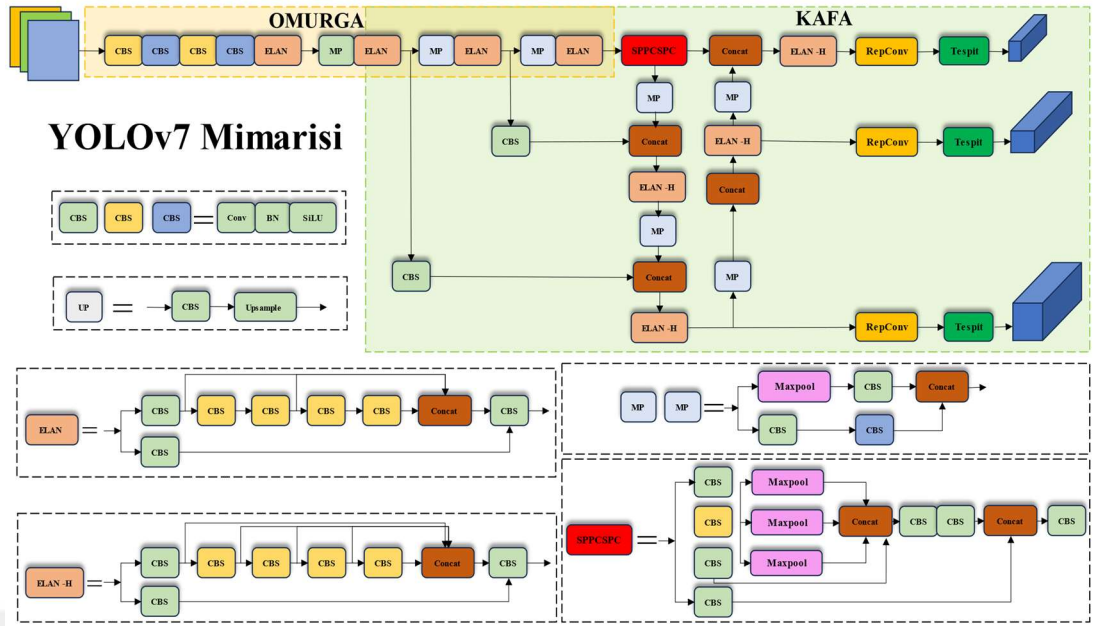
Model Ölçeklendirme: Model ölçekleme kullanılarak modelin derinliğini, görüntü çözünürlüğünü ve genişliğini artırmak mümkündür. Derinliği ölçeklemek, modeldeki katman sayısını artırmak veya azaltmak anlamına gelir. Benzer şekilde, genişlik, model mimarisindeki kanal sayısını ölçeklemeyi ifade eder. Hem derinlik hem de genişlik için model mimarisi dosyalarında tanımlı ölçekleme faktörleri bulunur. YOLOv7, diğer katmanlarla birleştirilmiş bir mimariye sahiptir; bu nedenle, bir hesaplama bloğunun derinlik parametresini ölçeklendirirken, çıkış çekirdeklerindeki değişikliği hesaplamamız gerekir. Ardından, genişliği, hesaplanan çekirdek değişikliği ile aynı miktarda ölçeklendirmeyi dikkate alırız. Bu şekilde, bileşik ölçekleme yaklaşımı, mimarinin sahip olduğu özellikleri korur ve optimal yapıyı sürdürür. Bileşik ölçekleme yöntemi Şekil 2.18'de gösterilmiştir.

YOLOv7 makalesinin yazarları, modelin bileşik ölçekleme yaklaşımıyla daha da optimize edilebileceğini göstermiştir. Burada, genişlik ve derinlik, birleştirme tabanlı modeller için uyumlu bir şekilde ölçeklenir.



Şekil 2.18. YOLOv7 ELAN (Wang et al., 2023)

YOLOv7, sınırlama kutusu regresyonu ve sınıflandırma görevlerinin doğruluğunu artıran güncellenmiş kayıp fonksiyonlarından yararlanır. Bu iyileştirmeler, nesne tespit görevlerinde genel performansı artırır. Buna ek olarak 640 ve 1280 boyutunda girdi görüntüleri almasında sağlanmıştır. Bu sayede daha küçük nesnelerin tespitinde daha başarılı sonuçlar elde ettiği raporlanmıştır. Orijinal yolov7'e ait bir görsel şekil 2.19'da paylaşılmıştır. Çalışmada Yolov7 ve Yolov7-tiny kullanılmıştır.

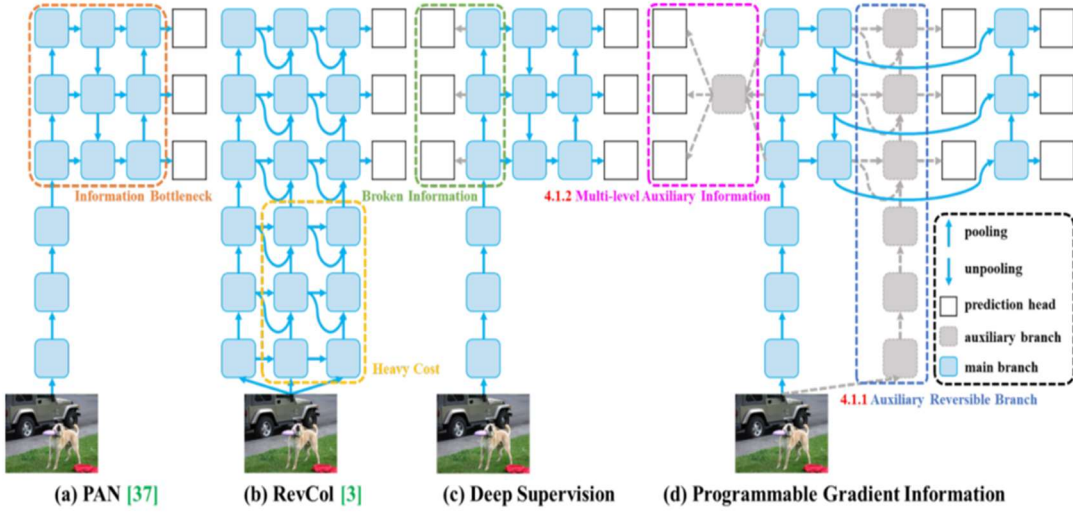


Şekil 2.19. Yolov7 mimarisi (K. Chen et al., 2022)

2.5.3.3. Yolov9

YOLOv9, Chien-Yao Wang ve arkadaşları tarafından 2024 yılında geliştirilen YOLO serisinin en son versiyonlarından biridir (Wang et al., 2024). Önceki modellerden farklı olarak YOLOv9, nesne algılama görevlerinin verimliliğini ve hassasiyetini artırmak için Programlanabilir Gradyan Bilgisi (PGI) ve Genelleştirilmiş Verimli Katman Toplama Ağı (GELAN) gibi çığır açan kavramlar sunmuştur.

Programlanabilir Gradyan Bilgisi (PGI): Gradyan bilgi akışını daha esnek ve etkili bir şekilde yönetmeyi amaçlayan bir yaklaşımdır. Bu yöntem, gradyanların ağ boyunca nasıl iletileceğini ve işleneceğini dinamik olarak belirlemeye olanak tanır. Geleneksel sinir ağlarında, gradyanların aktarımı ve işlenmesi genellikle statik ve sabittir, bu da modelin öğrenme kapasitesini sınırlayabilir. PGI, gradyan bilgilerini programlanabilir ve özelleştirilebilir hale getirerek, her katmanın gradyanlarla nasıl etkileşimde bulunacağını belirleme yeteneği sağlar. Bu, ağın daha derin yapılarını ve karmaşık ilişkileri daha verimli bir şekilde öğrenmesini sağlar, böylece modelin genel performansını ve doğruluğunu artırır. PGI, aynı zamanda eğitim sürecindeki dengesizlikleri azaltabilir ve daha hızlı ve istikrarlı öğrenme sonuçları elde edilmesine yardımcı olabilir. PGI, her biri modelin mimarisi içinde benzersiz ancak birbirine bağlı bir rol üstlenen üç bileşenin entegrasyonu yoluyla gelişir. (Şekil 2.20)



Şekil 2.20. Programlanabilir Gradyan mimarisi (Wang et al.)

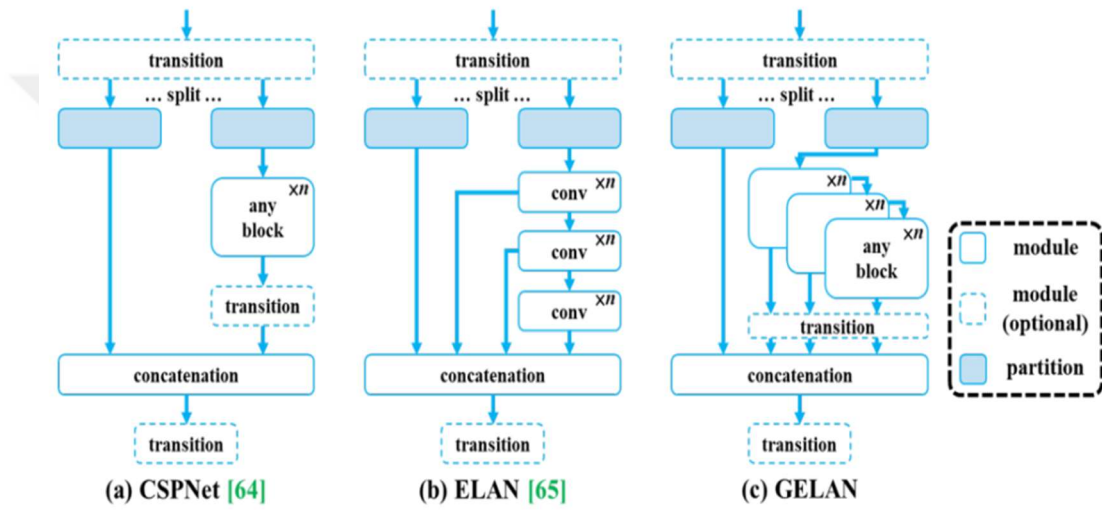
Ana Dal: Çıkarım için optimize edilmiş ana dal, modelin kritik aşamalarda akıcı ve verimli bir şekilde çalışmasını sağlar. Yardımcı bileşenlere ihtiyaç duymadan yüksek performansı sürdüren bu dal, ek hesaplama yükleri getirmeden modelin verimliliğini korur.

Yardımcı Tersine Çevrilebilir Dal: Yardımcı dal, güvenilir gradyanların üretilmesini ve doğru parametre güncellemelerini sağlar. Tersine çevrilebilir mimariden yararlanarak, derin ağ katmanlarındaki doğal bilgi kaybını azaltır ve öğrenme için verinin tam olarak korunmasını ve kullanılmasını sağlar. Bu dalın esnek tasarımı, model derinliği ve karmaşıklığının çıkarım hızını etkilemeden sorunsuz entegrasyon veya kaldırılmasını sağlar.

Çok Seviyeli Yardımcı Bilgi: Bu metodoloji, modelin katmanlarındaki gradyan bilgilerini birleştirmek için özel ağlar kullanır. Derin denetim modellerindeki bilgi kaybı sorununu ele alır ve verinin model tarafından tamamen anlaşılmasını sağlar. Bu teknik, farklı boyutlardaki nesneler için tahmin doğruluğunu artırır.

Generalized Efficient Layer Aggregation Network (GELAN): Genelleştirilmiş Etkili Katman Toplama Ağı (Generalized Efficient Layer Aggregation Network- GELAN), gradyan yol planlaması için CSPNet ve ELAN prensiplerini birleştiren yenilikçi bir mimaridir. GELAN, ELAN'ın katman toplamasını genişleterek herhangi bir hesaplama bloğunu destekler, bu da esneklik sağlar.

GELAN modülü katmanlar arasında bilgiyi daha etkin bir şekilde birleştiren ve yönlendiren bir yapı sunmaktadır. GELAN, katmanlar arasındaki ilişkileri optimize ederek hem hesaplama maliyetlerini hem de bellek kullanımını azaltmayı hedefler. Ayrıca, çeşitli nesne tespitinde performansı artırmak için daha iyi bir genelleştirme yeteneği sağlar. GELAN'ın temel prensibi, katmanlar arasında bilgi akışını daha etkili bir şekilde yönlendirmek ve modelin genel verimliliğini artırmaktır. Bu yaklaşım, özellikle büyük veri setleri ve karmaşık modellerle çalışırken önemli avantajlıdır. Şekil 2.21'de GELAN modele ait bir görsel paylaşılmıştır.



Şekil 2.21. GELAN modülleri (Wang et al., 2024)

YOLOv9, parametre sayısına göre sıralanan dört modelde piyasaya sürüldü: v9-S, v9-M, v9-C ve v9-E. Bu çalışma kapsamında ise YOLOv9'un t,s,m ve mimarileri değerlendirilmiştir.

YOLOv9 Mimarisi

Şekil 2.22 YOLOv9 Mimarisi (Parambil et al., 2024)

3. YOLO İLE GEMİ TESPİTİ

Nesne algılama, bir görüntüde birden fazla nesnenin tespit edilmesi ve konumlarının belirlenmesini sağlayan önemli bir bilgisayarlı görme görevidir. Bu görev, tek bir nesne sınıflandırmasından daha karmaşıktır çünkü birden fazla nesne sınıfının ve bunların yerlerinin belirlenmesini gerektirir. Nesne dedektörleri, bu tür çoklu nesne tespit ve sınıflandırma görevlerini yerine getirmek üzere geliştirilmiş algoritmalar olarak öne çıkmaktadır ve çeşitli endüstriyel uygulamalarda önemli rol oynamaktadır.

YOLO gibi modern nesne algılama yöntemleri, bu karmaşık görevleri etkili bir şekilde gerçekleştirebilen öncü algoritmalar arasında yer alır. YOLO, tek bir ağ yapısında nesne tespitini ve sınıflandırmasını aynı anda gerçekleştiren ve gerçek zamanlı uygulamalara uygun olarak optimize edilmiş bir algoritmadır.

Gemi tespiti için YOLO algoritmaları, daha önceden de kullanılmıştır (Ting et al., 2021). Bu çalışmada yüksek çözünürlüklü optik uydu görüntüleri üzerinde yine YOLO algoritmaları kullanılarak yüksek oranda başarı kaydedilmiştir. YOLO'nun nesne algılama katmanı, gemilerin muhtemel konumlarını belirler ve bu gemilerin hangi sınıfa (örneğin gemi sınıfı) ait olduğunu tahmin eder. Bu süreç genellikle, önceden eğitilmiş bir evrişimli sinir ağı omurga ağı üzerinden geçen görüntü özelliklerini kullanarak gerçekleştirilir.

YOLO algoritmaları, denizcilik güvenliği uygulamalarında gemi trafiği izleme, liman güvenliği, çevresel denetimler ve kaçakçılık gibi alanlarda etkin bir şekilde kullanılabilir. Bu yöntemler, yüksek doğruluk ve gerçek zamanlı işleme yeteneği ile karmaşık ve kritik görevlerde güvenilir sonuçlar sağlayacaktır.

3.1. Veri Seti

Yüzer unsurların optik uydu görüntüleri kullanılarak sınıflandırılmasında önemli bir ilerleme söz konusudur. Çalışmamızda, Google Earth görüntülerinden elde edilen Çok Yüksek Çözünürlüklü Gemiler (VHRShips) veri setini tanıtarak, yüzer unsurların tespit ve sınıflandırma işlemleri yapılmaktadır. VHRShips veri seti, çeşitli gemi türlerini, farklı boyutları, kıyı konumlarını ve değişen veri toplama koşullarını içermektedir, bu da gerçek dünya uygulamalarına daha geniş bir adaptasyon sağlamayı hedeflemektedir (Kızılkaya et al., 2022). Ayrıca, önerilen yöntem ve VHRShips veri

seti, denizcilik alanındaki GeoAI uygulamalarını geliştirmek için büyük ölçekli coğrafi veriler üzerinde derin öğrenmenin uygulanmasına yönelik ileriki çalışmalarda önemli bir referans kaynağı olarak kullanılabilir (Kızılkaya et al., 2022).

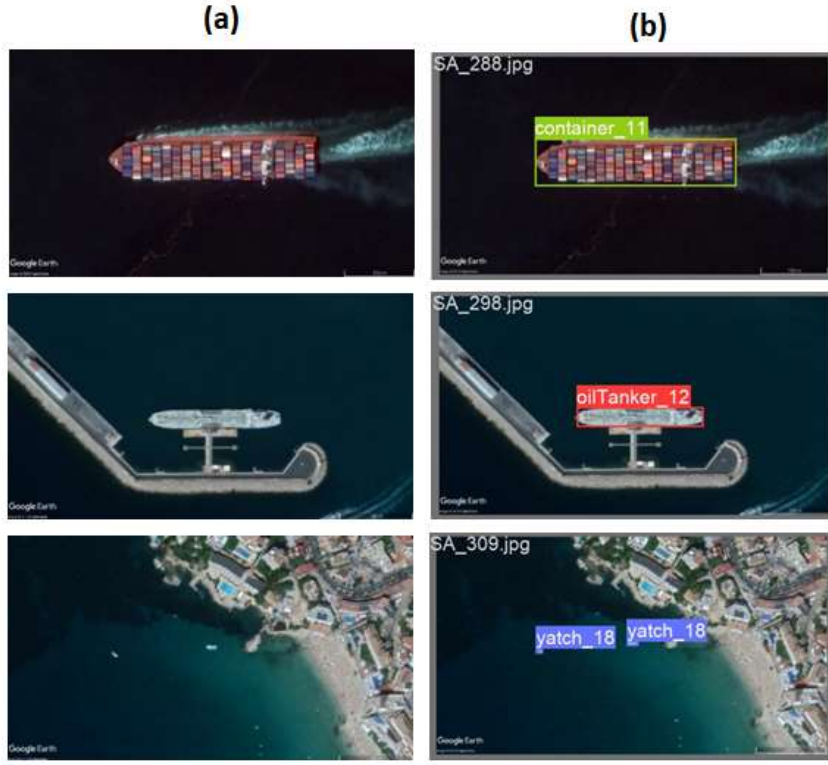
Literatürde gemi tespitine yönelik paylaşılan açık kaynak veri setlerinin Google Earth'ten ve farklı uydulardan kırılan görüntülere dayanmaktadır. Standart bir veri tabanı olmamakla birlikte birçok veri seti açık kaynak olarak paylaşılmamaktadır (Mehran et al., 2023). Bu çalışmada Kızılkaya ve arkadaşları tarafından oluşturulan Very High-Resolution Ships (VHRShips) veri seti kullanılmıştır (Kızılkaya et al., 2022). VHRShips veri seti Google Earth'ten alınmış 720x1280 çözünürlüğe sahip toplamda 6312 görüntüden meydana gelmektedir. Görüntüler içerisinde 1000 tanesinde herhangi bir yüzer unsur bulunmamaktadır. Geriye kalan 5312 tanesi ise içerisinde en az bir veya birden fazla gemi içermektedir. Veri seti toplamda 34 farklı sınıfa sahiptir. Veri seti hali hazırda eğitim ve test olarak ayrılmıştır (Kızılkaya et al., 2022). Eğitim ve test grubuna ait sınıf dağılımları ve toplam adetleri Çizelge 3.1'de verilmiştir.

Çizelge 3.1. Veri setinde eğitim ve test grubuna ait sınıf dağılımları

S.N.	Sınıf Adı	Eğitim Veri Sayısı	Test Veri Sayısı
1	Aircraft	35	6
2	Auxiliary	164	40
3	Barge PonToon	737	138
4	Bulk Carrier	678	128
5	Coast Guard	40	10
6	Coaster	346	55
7	Container	580	111
8	Cruiser	68	14
9	Destroyer	91	17
10	Dredger Revlamation	268	54
11	Dredging	195	42
12	Drill	34	6

13	Ferry	99	17
14	Fishing	37	6
15	Floating Dock	52	12
16	Frigate	71	11
17	General Cargo	655	131
18	Landing	34	9
19	LPG	33	7
20	Offshore	127	25
21	Oil Tanker	592	118
22	Ore Carrier	771	159
23	Other	27	6
24	Passanger	144	27
25	Patrol Force	102	13
26	Roro	102	20
27	Service Craft	151	10
28	Small Boat	944	177
29	Small Passanger	418	72
30	Submarine	50	6
31	Tanker	779	148
32	Tug	854	171
33	Undefined	435	83
34	Yatch	1620	326
Total		5312	1075

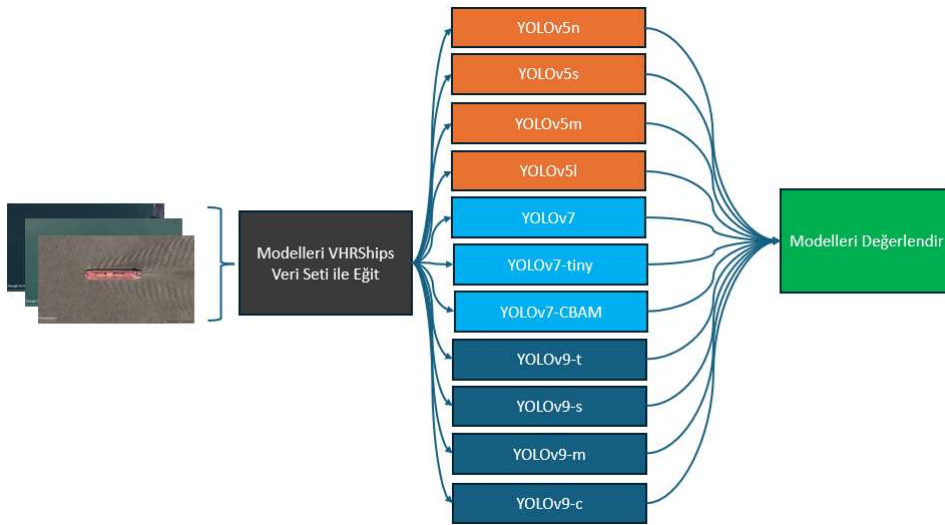
Ayrıca şekil 3.1’de veri setinden bazı örnekler paylaşılmıştır. Görselde (a) orijinal görüntüyü temsil ederken (b) etiketlenmiş görüntüyü göstermektedir.



Şekil 3.1. Veri setinden örnekler

3.2. Kullanılan Yöntemler

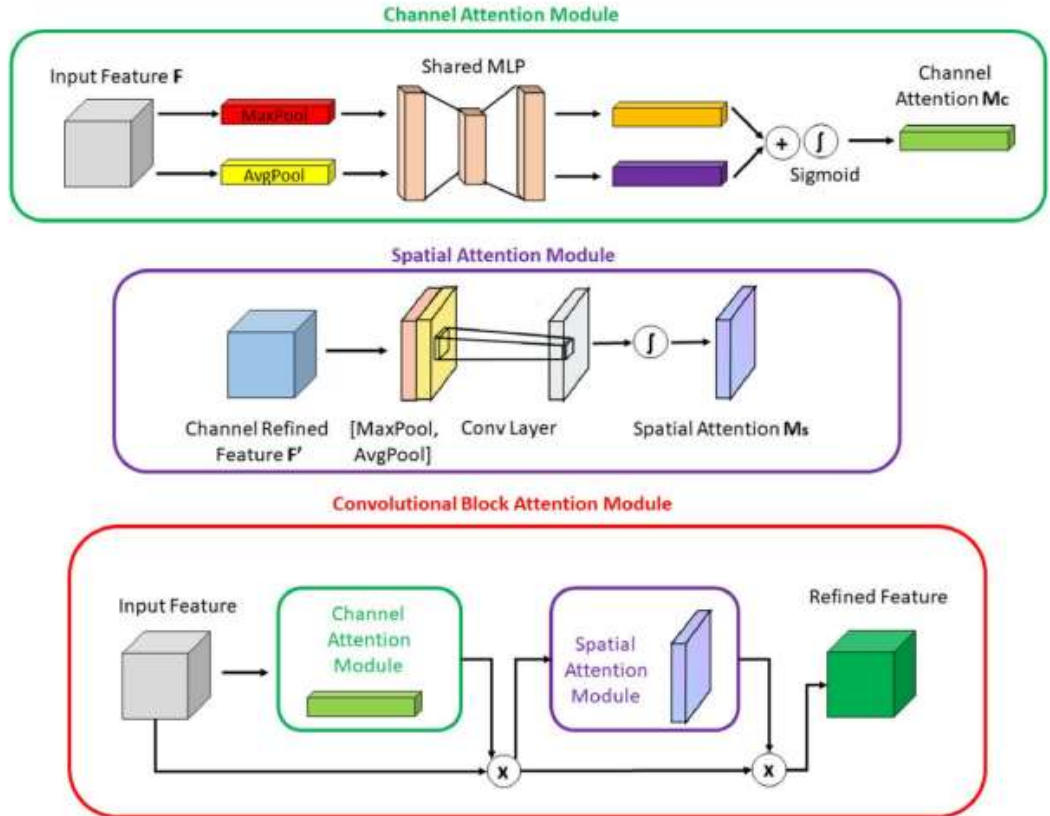
Bu çalışma kapsamında üç farklı YOLO versiyonun optik görüntülerden gemileri tespit ve sınıflandırma performansı karşılaştırılmıştır. Buna ek olarak YOLOv7 modellerinde çeşitli iyileştirmeler yaparak orijinal modeller ile karşılaştırmalar yapılmıştır. Çalışmanın genelini özetleyen bir görsel şekil 3.2’de paylaşılmıştır.



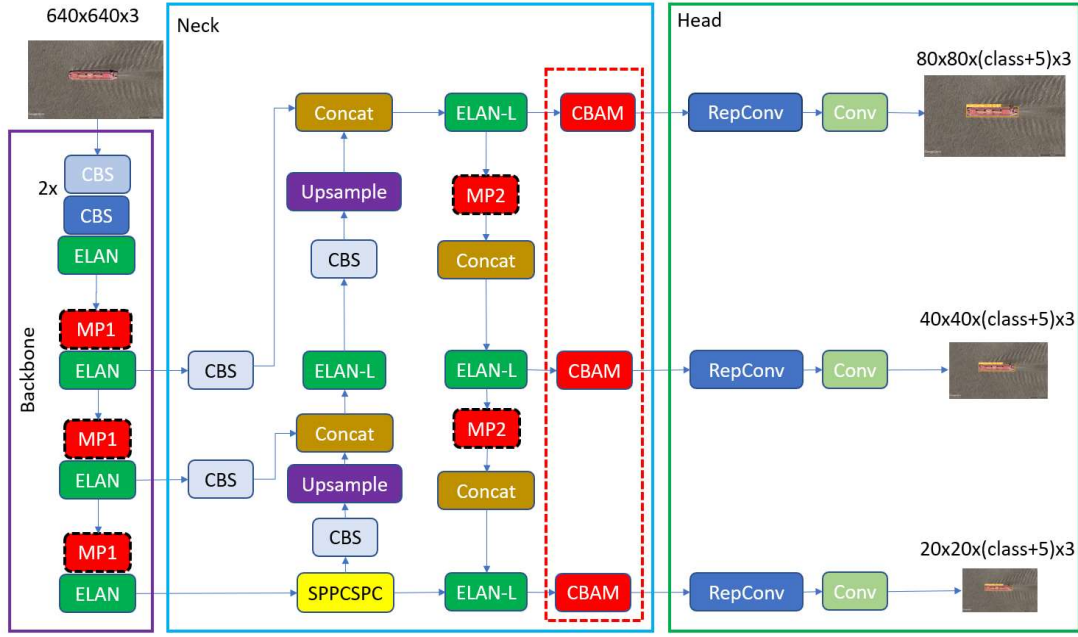
Şekil 3.2. Çalışma özeti

3.2.1. CBAM (Convolutional Block Attention Module)

CBAM (Convolutional Block Attention Module), konvolüsyonel sinir ağlarının performansını artırmak için kullanılan bir dikkat mekanizmasıdır (Woo et al., 2018). Bu mekanizma, özellikle görüntü işleme ve bilgisayarla görme alanlarında etkili sonuçlar elde etmek amacıyla geliştirilmiştir. CBAM, iki ana bileşenden oluşur: kanal dikkat modülü ve mekânsal dikkat modülü. Kanal dikkat modülü, her bir kanalın önemini değerlendirirken, mekânsal dikkat modülü ise her bir piksel konumunun önemini belirler. Bu iki modül, ağın önemli özellikleri vurgulamasına ve gereksiz bilgileri göz ardı etmesine olanak tanır, bu sayede modelin doğruluğu ve genel performansı artırılır. Şekil 3.3'te CBAM modülüne ait bir görsel paylaşılmıştır. CBAM, derin öğrenme modellerinde daha etkili ve verimli öğrenme sağlayarak, özellikle karmaşık veri kümesi ve görevlerde başarılı sonuçlar elde edilmesine yardımcı olur. Bu çalışma kapsamında YOLOv7 mimarisinin boyun çıkışlarına CBAM modülleri eklenmiştir. CBAM eklenen mimari şekil 2.7'de gösterilmektedir.



Şekil 3.3. Convolutional Block Attention mimarisi (Ayhan et al., 2024)



Şekil 3.4. CBAM entegre edilmiş Yolov7 mimarisi

3.3. Değerlendirme Kriterleri

Bu çalışmada, modellerin gemi tespit performansını değerlendirmek için kapsamlı bir değerlendirme metrikleri seti kullanılmıştır. Bu metrikler Precision (P), Recall (R) mean Average Precision (mAP)'dir. Metrikler aşağıdaki formüllere göre hesaplanmıştır.

$$P = \frac{GP}{GP + YP} \quad (5)$$

$$R = \frac{GP}{GP + YN} \quad (6)$$

$$mAP = \int_0^1 P(R) dR \quad (7)$$

$$F1 \text{ Score} = 2x \frac{P \times R}{P + R} \quad (8)$$

Denklemlerde Gerçek Pozitif (GP) doğru tespit edilen gemileri, Yanlış Pozitif (YP) yanlış tespit edilen gemileri, FN ise tespit edilemeyen gemileri temsil etmektedir (şekil 3.5).

Kesişim Üzerinden Birleşim (Intersection over Union - IoU), bilgisayarlı görme ve nesne algılama sistemlerinde model performansını değerlendirmede temel bir metriktir. IoU, modelin görüntülerdeki nesneleri ne kadar hassas bir şekilde yerleştirdiğini ölçer ve genellikle tahmin edilen sınırlayıcı kutu ile gerçek sınırlayıcı kutu arasındaki örtüşme oranını belirler. Bu ölçüm, modelin nesnelerin konumlarını doğru belirlemesini ve nesneleri arka planlardan ayırt etme kapasitesini değerlendirir. Güvenlik sistemlerinde, gözetim görüntülerindeki nesneleri veya kişileri algılamak ve tanımlamak için IoU'dan yararlanılmaktadır (Redmon et al., 2016)

Nesne algılama modellerinin performansını değerlendirmek için kullanılan temel metrikler hassasiyet (precision), geri çağırma (recall) ve F1 puanıdır. Bu metrikler, modelin görüntülerdeki ilgi çekici nesneleri tanımlama becerisini değerlendirir ve modelin doğruluğunu artırma çabasında önemli rol oynar.

Gerçek Pozitif (GP): Modelin nesneleri doğru bir şekilde tanımladığı ve yerleştirdiği durumları ifade eder. Bu, tahmin edilen sınırlayıcı kutu ile gerçek sınırlayıcı kutu arasındaki IoU puanının belirtilen eşik değerine eşit veya daha yüksek olduğu durumları içerir.

Yanlış Pozitif (YP): Modelin gerçek yerde bulunmayan bir nesneyi yanlış bir şekilde tanımladığı durumları ifade eder.

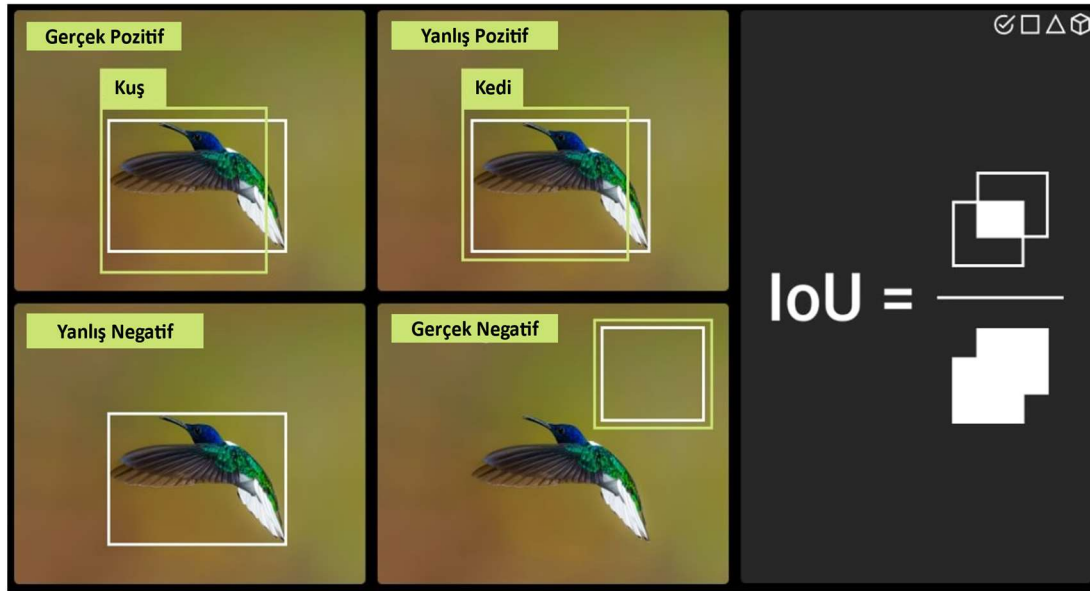
Yanlış Negatif (YN): Modelin gerçek yerde bulunan bir nesneyi tespit edemediği durumları ifade eder.

Gerçek Negatif (GN): Nesne algılamada genellikle kullanılmaz; nesnelerin yokluğunu doğru bir şekilde reddetmeyi ifade eder, ancak nesne algılamada esas olarak nesneleri tespit etmek hedeflenir.

Kesinlik (Precision): Modelin yaptığı pozitif tahminlerin doğruluğunu ölçer. Modelin gerçek nesneleri yanlış pozitiflerden ne kadar iyi ayırt edebildiğini değerlendirir. Yüksek bir kesinlik puanı, modelin yanlış pozitiflerden kaçınmada etkili olduğunu ve güvenilir pozitif tahminler sağladığını gösterir (Szegedy et al., 2015).

Hatırlama (Recall): Hatırlama, modelin tüm ilgili nesneleri yakalama yeteneğini ölçer ve ilgi çekici nesneleri tanımlamadaki eksiksizliğini değerlendirir. Yüksek bir hatırlama puanı, modelin verilerdeki ilgili nesnelerin çoğunu etkili bir şekilde tanımladığını gösterir (Girshick et al., 2014).

F1 Puanı: Kesinlik ve hatırlamanın harmonik ortalamasıdır. F1 puanı, modelin performansının dengeli bir ölçümünü sağlar ve özellikle veri setlerinde pozitif ve negatif sınıflar arasında dengesizlik olduğunda yararlıdır (Cai & Vasconcelos, 2018). F1 puanı, hem yanlış pozitifleri hem de yanlış negatifleri göz önünde bulundurarak modelin genel performansını değerlendirir.



Şekil 3.5. Nesne tespit hesaplama ölçütleri (Kniazieva, 2024)

Ortalama Hassasiyet (AP) ve Genel Ortalama Hassasiyet (mAP), nesne algılama modellerinin değerlendirilmesinde önemli ölçümlerdir. Bu ölçümler, bir modelin nesneleri görüntüler içinde tanımlama ve yerleştirme yeterliliğini ölçmek için kullanılır.

Ortalama Hassasiyet (AP): AP, bir nesne algılama modelinin hassasiyetini çeşitli hatırlama seviyelerinde değerlendirir. Hassasiyet, modelin pozitif tahminlerinin doğruluğunu belirtirken, hatırlama, modelin tüm ilgili nesneleri başarılı bir şekilde tanımlama yeteneğini niceliksel olarak belirler. AP, farklı güven eşiklerinde kesinlik-hatırlama değerlerini hesaplayarak, bu eğrinin altındaki alan (AUC) aracılığıyla

modelin performansını temsil eder. Daha yüksek AUC değerleri, üstün model performansını gösterir (Chen et al., 2014).

Genel Ortalama Hassasiyet (mAP): mAP, AP kavramını genişleterek birden fazla güven eşiğinde ortalama AP'yi hesaplar. Bu yaklaşım, modelin değişen güven seviyelerinde doğru tespitler yapma konusundaki çok yönlülüğünü değerlendirir ve nesne algılama görevlerindeki genel becerisini daha kapsamlı bir şekilde değerlendirir. mAP, modelin farklı tespit senaryolarına uyum sağlama yeteneğini değerlendirir ve gerçek dünya uygulamalarındaki sağlamlığını vurgular (He et al., 2017).

3.4.Geliştirme Ortamı ve Eğitim Parametreleri

Bu çalışmadaki tüm deneyler, Ubuntu 20.04, PyTorch 2.0.0, CUDA 11.8, Python 3.8, 12 GB VRAM'li NVIDIA RTX 1080 Ti GPU ve 2 adet Intel(R) Xeon(R) CPU'dan oluşan bir ortamda gerçekleştirilmiştir. Tüm modeller eşit bir karşılaştırma olması adına önceden tanımlı eğitim parametreleri ile eğitilmiştir. Modeller eğitilirken transfer öğrenme yöntemi kullanılmıştır. Modellerin eğitim parametreleri Çizelge 3.2'de paylaşılmıştır.

Çizelge 3.2. Model eğitim parametreleri

Parametre Adı	Değeri
Epok Sayısı	200
Batch Size	16
IOU	0.50
Confidence	0.20
Optimizer	Stochastic Gradient Descent
Learning Rate	0.01
Momentum	0.93
Weight Decay	0.0005
Input size	640x640x3

4. DENEYLER VE SONUÇLAR

Çalışma kapsamında değerlendirilen modellere ait gemi tespit ve sınıflandırma performansları çizelge 4.2’de paylaşılmıştır. Modeller test edilirken IOU değeri 0.50, confidence (güven) değeri ise 0.25 olarak ayarlanmıştır. Çizelge 4.1’de verilen sonuçlara göre YOLOv5l modelinin 0.984 kesinlik, 0.984 duyarlılık, 0.991 mAP ve 0.984 F1 skor değeri elde ederek diğer modeller arasında en başarılı model olmuştur. En başarılı modelin hemen ardından sırası ile YOLOv5m, YOLOv9c, YOLOv5s, YOLOv9m, YOLOv9s, YOLOv7-CBAM, YOLOv7-tiny, YOLOv5n, YOLOv7, YOLOv9t modelleri gelmektedir.

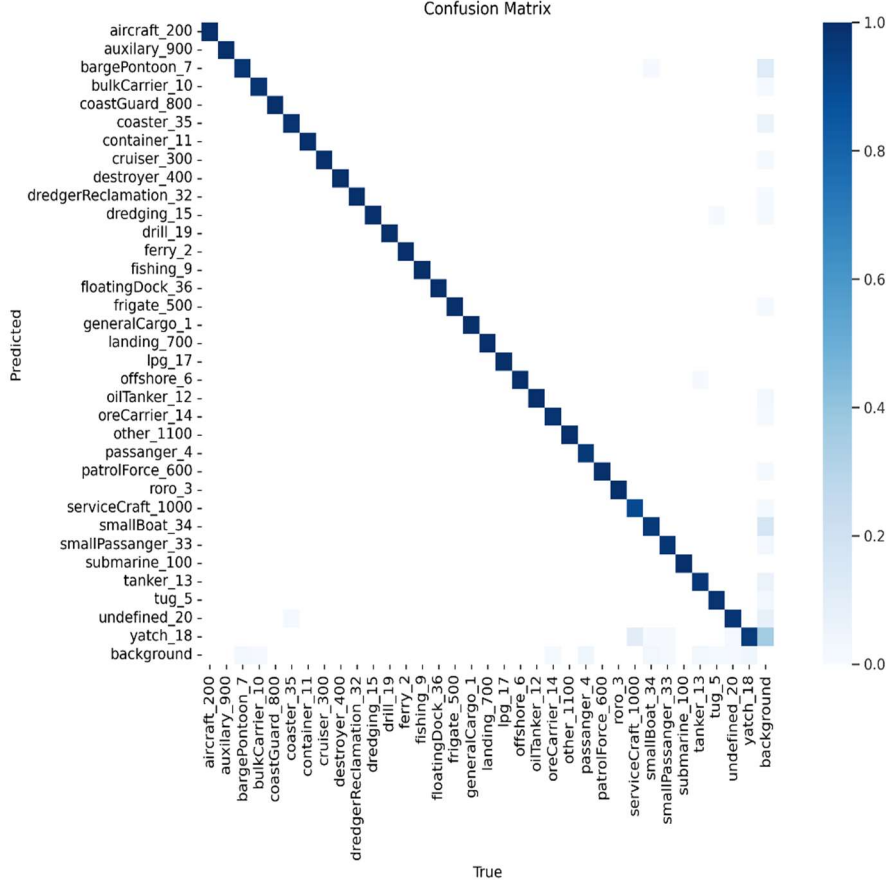
Çizelge 4.1. Modellerin test verisi üzerindeki performansları

Model	Precision (All classes)	Recall (All classes)	mAP50 (All classes)	F1-Score
YOLOv5n	0.888	0.835	0.889	0.860
YOLOv5s	0.966	0.975	0.978	0.970
YOLOv5m	0.983	0.984	0.991	0.983
YOLOv5l	0.984	0.984	0.991	0.984
YOLOv7tiny	0.911	0.832	0.901	0.869
YOLOv7	0.896	0.815	0.872	0.853
YOLOv9c	0.98	0.962	0.983	0.970
YOLOv9m	0.951	0.92	0.96	0.935
YOLOv9s	0.941	0.91	0.948	0.925
YOLOv9t	0.831	0.816	0.875	0.823
YOLOv7-CBAM	0.914	0.902	0.934	0.907

Çizelge 4.2’de YOLOv5l modelinin sınıf bazında elde etmiş olduğu performans değerleri verilmiştir. Sonuçlar incelendiğinde çoğu gemi türünün 0.995 mAP değeri ile tespit edildiği görülmektedir. Sınıflar arasında small boat ve yacht etiketine sahip sınıfların ise sırası ile 0.974 ve 0.956 mAP değeri ile tespit edilmiştir. Söz konusu iki sınıf modelin diğerlerine nispeten tespitite daha çok zorlandığı sınıflardır. Karışıklık matrisi (Confusion matris), gerçek etiketlerle modelin tahmin ettiği etiketler arasındaki ilişkileri gösteren bir tablodur. Modellerin modellerinin sınıflandırma performansını detaylı bir şekilde anlamak için kritik bir araçtır. Doğru tahminlerin yanı sıra yanlış tahminleri de gösterdiğinden, modelin güçlü ve zayıf yönlerini belirlemede yardımcı bir araçtır. Şekil 4.1’de YOLOv5l’e ait karışıklık matrisi verilmiştir. Matris incelendiğinde özellikle yacht sınıfının model tarafından arka plan olarak daha fazla algılandığı görülmektedir.

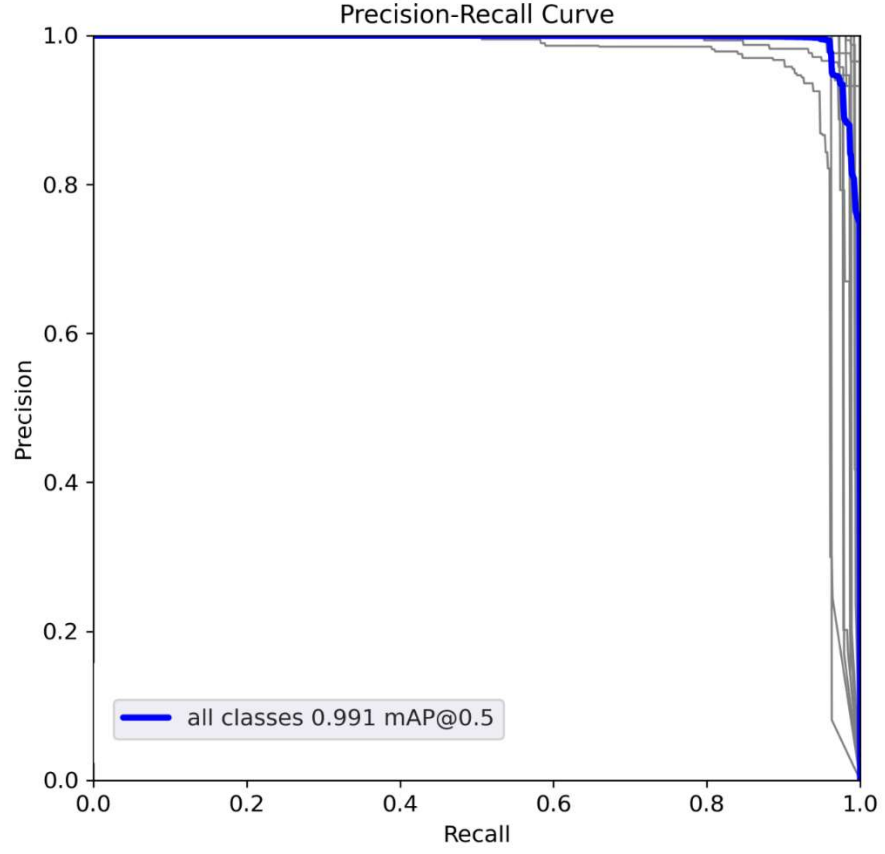
Çizelge 4.2. YOLOv5l modelinin sınıflara göre performans sonuçları

Sınıflar	Instances	P	R	mAP
All	2175	0.984	0.984	0.991
Aircraft	6	0.958	1	0.995
Auxiliary	40	0.988	1	0.995
Barge	138	0.992	0.956	0.977
PonToon				
Bulk Carrier	128	1	0.996	0.995
Coast Guard	10	0.975	1	0.995
Coaster	55	1	0.979	0.994
Container	111	0.997	1	0.995
Cruiser	14	0.977	1	0.995
Destroyer	17	0.987	1	0.995
Dredger	54	0.993	1	0.995
Revlamation				
Dredging	42	0.991	1	0.995
Drill	6	0.95	1	0.995
Ferry	17	0.984	1	0.995
Fishing	6	0.964	1	0.995
Floating Dock	12	0.991	1	0.995
Frigate	11	0.984	1	0.995
General Cargo	131	0.997	1	0.995
Landing	9	0.976	1	0.995
LPG	7	0.962	1	0.995
Offshore	25	0.986	1	0.995
Oil Tanker	118	1	1	0.995
Ore Carrier	159	0.993	0.987	0.995
Other	6	0.969	1	0.995
Passanger	27	0.989	0.963	0.966
Patrol Force	13	0.977	1	0.995
Roro	20	0.986	1	0.995
Service Craft	10	1	0.943	0.995
Small Boat	177	0.982	0.901	0.974
Small	72	1	0.961	0.994
Passanger				
Submarine	6	0.968	1	0.995
Tanker	148	1	0.965	0.991
Tug	171	0.997	0.988	0.987
Undefined	83	0.976	0.976	0.994
Yatch	326	0.969	0.855	0.957



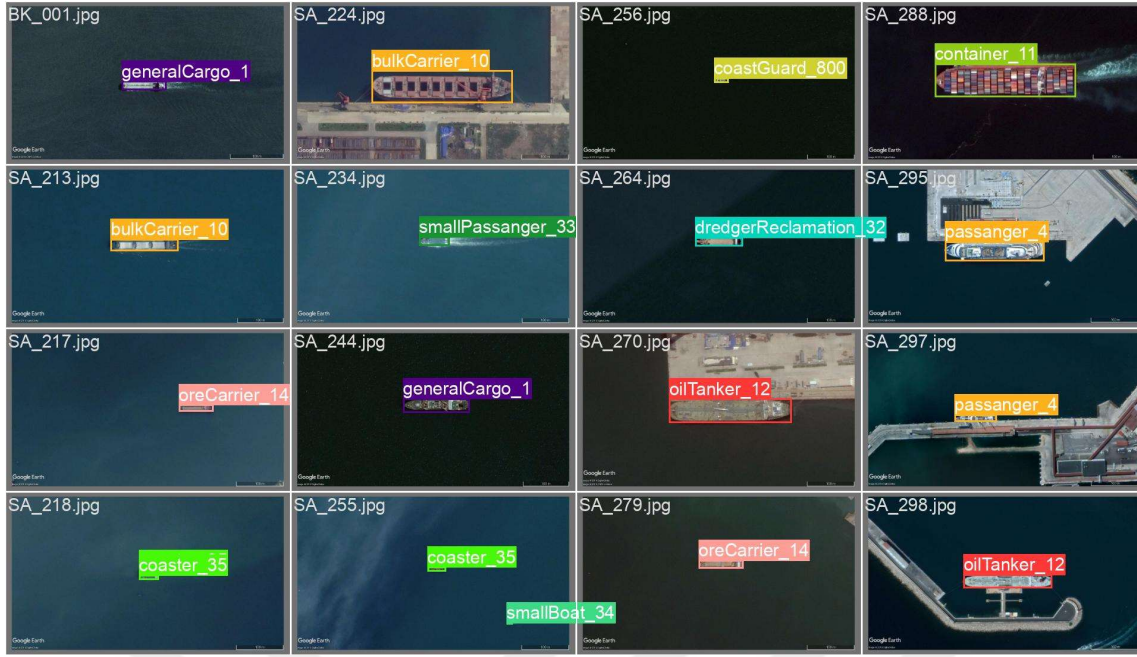
Şekil 4.1. YOLOv5l karışıklık matrisi

P-R (Precision-Recall) eğrisi, sınıflandırma modellerinin performansını değerlendirmek için kullanılan bir grafiksel araçtır. Hassasiyet (precision) ve duyarlılık (recall) arasındaki ilişkiyi anlamada faydalıdır. P-R eğrisi, modelin performansını görselleştirerek, modelin her bir olasılık eşiği için hassasiyet ve duyarlılık değerlerini gösterir. Yüksek hassasiyet ve duyarlılık değerleri, eğrinin sağ üst köşesine yakın olması modelin performansının iyi olduğunu göstermektedir. Şekil 4.2 verilen YOLOv5l modeline ait PR eğrisi modelin 0.5 IoU eşik değerinde 0.991 mAP değerini elde ettiğini göstermektedir.

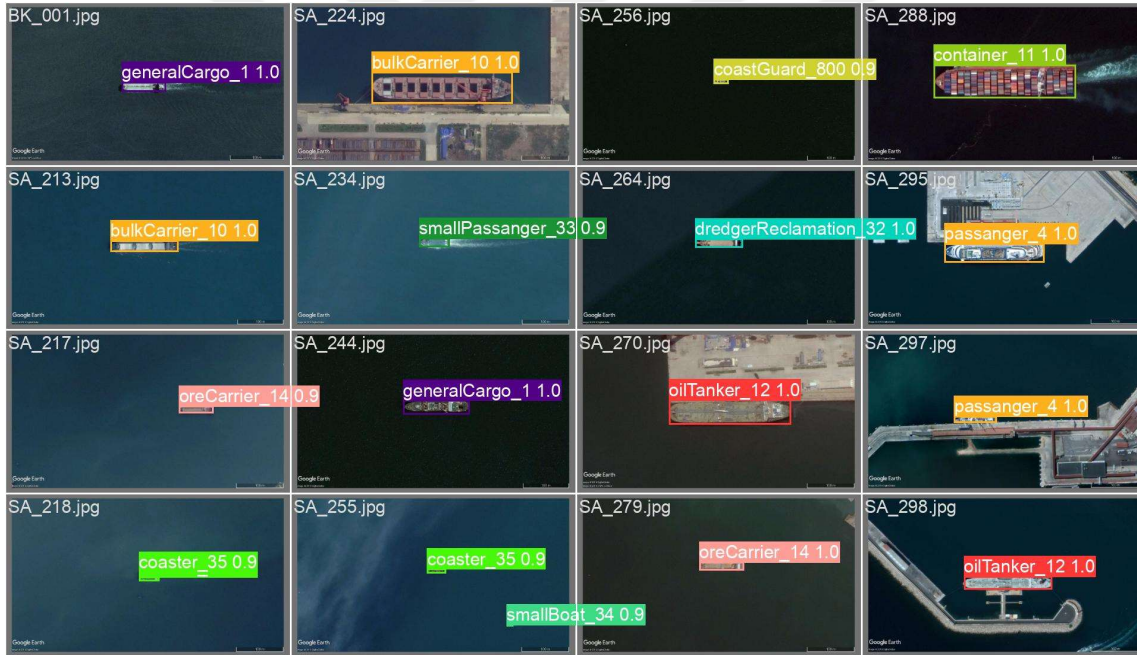


Şekil 4.2. YOLOv5l Kesinlik-Duyarlılık eğrisi

Şekil 4.3'te test verisinden bazı örnekler ve etiketleri paylaşılmıştır. Şekil 4.4'te ise modelin tahminleri gösterilmektedir. Modelin söz konusu test örneklerinde tatmin edici tespit ve sınıflandırma performansı göstermiştir. Özellikle şekil 4.4'teki güven skorlarının 1'e yakın olduğu görülmektedir.



Şekil 4.3. Bazı test görüntüleri ve etiketleri



Şekil 4.4. YOLOv51 tahmin sonuçları

5. TARTIŞMA

Literatürde aynı VHRShips veri setini kullanarak yapılmış bir adet çalışma bulunmaktadır. Bu çalışmada Kızılkaya ve arkadaşları kendi oluşturmuş oldukları VHRShips veri setini YOLOv4 algoritmasının eğitiminde kullanmışlardır. Eğitim sonuçlarına göre YOLOv4 algoritmasının gemileri tanıma performansını %71,34 F1 skor ile ifade etmişlerdir. Bu çalışmada elde ettiğimiz sonuç %98,4 F1 skor değeridir. Bu bağlamda söz konusu veri setinde başarılı sonuçlar elde edildiği görülmektedir. Çalışmada en düşük performans gösteren YOLOv5n modeli bile %86 F1 skoru elde ederek YOLOv4'ten daha başarılı olmuştur. Çalışmada büyük modellerin başarılı olması gömülü sistemlerde kullanılması yönüyle bir dezavantaj olarak görülebilir. Bu sorunun küçük modellere eklenecek dikkat mekanizmaları ile çözülebileceğini öngörmekteyiz. Bu bağlamda YOLOv7 modeline eklediğimiz dikkat mekanizmalarının F1 skor performansında %5,4 lük bir artış gerçekleştirmesi bu tezimizi desteklemektedir.

Çizelge 5.1 modellerin toplam katman sayıları, sahip oldukları parametre miktarı, GFLOPS (Graphics Floating Point Operations Per Second) sayıları ve hafızada kapladıkları toplam alan paylaşılmıştır. Çizelge 4.1'de verilen verilere göre tespit performansı en düşük olan model toplam parametre sayısı en az olan model YOLOv5n olmuştur (Çizelge 5.1). Bunun tam tersini söylemek ise mümkün değildir. Çizelge 4.1'e göre en fazla parametre sahip olan YOLOv9-c modeli başarı sıralamasında üçüncü olarak yer almıştır. Yine de başarılı modelin toplam parametre sayısının sıralamada ikinci olduğu büyüklük ile başarı arasında bir bağlantı olduğunu göstermektedir.

Çizelge 5.1. Model büyüklükleri

Model	Katmanlar	Parametreler	GFLOPS	Boyut
YOLOv5n	157	1,805,167	4.3	3.9 MB
YOLOv5s	157	7,101,823	16.1	14.5 MB
YOLOv5m	212	20,986,287	48.4	42.4 MB
YOLOv5l	267	46,285,983	108.4	93.1 MB
YOLOv7tiny	208	6,096,894	13.4	12.4 MB
YOLOv7	314	36,659,774	103.9	75.1 MB
YOLOv9c	604	50,774,508	237.0	102.9 MB
YOLOv9m	588	32,600,884	131.0	66.4 MB
YOLOv9s	658	9,623,564	39.8	20.4 MB
YOLOv9t	658	2,679,932	12.0	6.2 MB
YOLOv7-CBAM	350	36,704,031	103.8	75.2 MB

Çizelge 4.2. incelendiğinde birçok sınıfın model tarafından yüzde yüz doğrulukla tespit edildiği görülmektedir. Bunun yanında modelin tespit performansı en düşük sınıflar 0.855 duyarlılık değeri ile yacht ve 0.901 ile smallBoat olduğu tespit edilmiştir. Bunun muhtemel sebepleri bu gemi türlerinin boyutlarının çok küçük olmasından kaynaklanmaktadır. Optik uydu görüntülerinde uzaktan algılama problemlerinde küçük cisimlerin tespit performansının düşük olduğu farklı çalışmalarda belirtilmiştir (Nina et al., 2020; Rabbi et al., 2020; Wang et al., 2022). Bu bağlamda modellerin küçük gemileri tespit etme performansının düşük olması doğal karşılanmaktadır. Şekil ‘4.3 ve 4.4’ te modeli test verisi üzerinde yapmış olduğu tespitlerden örneklere de yer verilmiştir.

6. SONUÇ VE TAVSİYELER

Bu tez çalışmasında, Derin Öğrenme (DÖ) yöntemlerinden biri olan YOLO (You Only Look Once) nesne tespit algoritmasının on bir farklı versiyonunun optik uydu görüntülerinden gemileri tespit etme ve sınıflama performansları kapsamlı bir şekilde değerlendirilmiştir. Elde edilen sonuçlara göre, YOLOv5l modeli, diğer YOLO versiyonlarına kıyasla 0.984 hassasiyet, 0.984 duyarlılık, 0.991 mAP ve 0.984 F1 skor değeri elde ederek en yüksek performansı göstermiştir. Bu bulgular, YOLOv5l'in gemi tespiti için yüksek çözünürlüklü optik uydu görüntülerinde etkili bir seçenek olduğunu ortaya koymuştur. Öte yandan, dikkat mekanizması eklenerek geliştirilen YOLOv7-CBAM modelinin, orijinal YOLOv7 modelinden daha başarılı sonuçlar verdiği belirlenmiştir. YOLOv7'ye eklenen CBAM modülleri modelin odaklanma yeteneğini artırmış ve böylece gemi tespitinde daha yüksek doğruluk ve daha iyi sınıflandırma performansı sağlamıştır. CBAM modüllerinin, modelin özellikleri üzerinde yoğunlaşarak bilgiye daha iyi odaklanmasına olanak tanıdığı ve bu sayede genel performansı iyileştirdiği gözlemlenmiştir. Çalışmada ayrıca, parametre sayısı bakımından büyük modellerin genellikle daha başarılı sonuçlar verdiği bulunmuştur. Büyük modeller, daha fazla parametre ve daha karmaşık yapılar içerdiği için genellikle daha yüksek performans sergileyebilir. Bu bağlamda, büyük modellerin sağlamış olduğu başarılar, onların geniş veri setleri ve karmaşık görevlerdeki potansiyel avantajlarını vurgulamaktadır. Ancak kısıtlı donanıma sahip donanımlarda büyük modellerin gerçek zamanlı olarak çalışması pek mümkün değildir. Bu bağlamda, küçük modellere yapılacak iyileştirmeler de önemli bir araştırma alanı olarak ortaya çıkmaktadır. Bu nedenle, gelecekteki çalışmalarda, küçük modellerin performansını artırmak için çeşitli stratejiler ve optimizasyon yöntemleri üzerinde durulması planlanmaktadır. Bu çalışmalar, küçük modellerin büyük modellere kıyasla benzer başarıyı yakalayabilmesini ve daha az hesaplama gücü gerektirmesini sağlamayı amaçlamaktadır. Çalışmada optik uydu görüntülerinin doğası gereği oluşabilecek bulutlanmalar tespit performansını önemli ölçüde düşürmektedir. Bu durum çalışmanın limitasyonlarından biridir. Görüntülerde yer alan küçük gemilerin tespit performansının nispeten düşük olması ise çalışmanın diğer limitasyonudur. Bu limitasyon da modellere eklenecek dikkat mekanizmaları veya daha büyük boyutlarda bir çıktı eklenmesi ile çözülmesi öngörülmektedir.

Sonu olarak, bu tez kapsamında elde edilen bulgular, YOLO tabanlı mimarilerin optik uydu grntlerinden gemilerin tespit ve sınıflanmasında bařarılı olduėunu gstermektedir. Gelecekte, kk modellere yapılacak iyileřtirmeler ve yeni tekniklerin entegrasyonu ile bu bařarıların daha da artırılması hedeflenmektedir.



KAYNAKLAR

- Ak, T. (2021). Yapay Zekâ Teknolojileri, Güvenlik ve Kolluk Kuvvetinin Suç Önleme Faaliyetleri. *SDE Akademi Dergisi*, 1(1), 120-140.
- Atik, Ş. Ö., & İpbüker, C. Uydu Görüntülerinden Örnek Segmentasyonu ile Gemi Tespiti.
- Ayhan, B., Ayan, E., & Bayraktar, Y. (2024). A novel deep learning-based perspective for tooth numbering and caries detection. *Clinical Oral Investigations*, 28(3), 178.
- Bear, M., Connors, B., & Paradiso, M. A. (2020). *Neuroscience: exploring the brain, enhanced edition: exploring the brain*. Jones & Bartlett Learning.
- Bengio, Y. (2013). Deep learning of representations: Looking forward. International conference on statistical language and speech processing,
- Bengio, Y., Lecun, Y., & Hinton, G. (2021). Deep learning for AI. *Communications of the ACM*, 64(7), 58-65.
- Bo, L., Xiaoyang, X., Xingxing, W., & Wenting, T. (2021). Ship detection and classification from optical remote sensing images: A survey. *Chinese Journal of Aeronautics*, 34(3), 145-163.
- Cai, Z., & Vasconcelos, N. (2018). Cascade r-cnn: Delving into high quality object detection. Proceedings of the IEEE conference on computer vision and pattern recognition,
- Chai, J., Zeng, H., Li, A., & Ngai, E. W. (2021). Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Machine Learning with Applications*, 6, 100134.
- Chen, J., Sun, J., & Wang, G. (2022). From unmanned systems to autonomous intelligent systems. *Engineering*, 12, 16-19.
- Chen, K., Yan, G., Zhang, M., Xiao, Z., & Wang, Q. (2022). safety helmet detection based on YOLOv7. Proceedings of the 6th International Conference on Computer Science and Application Engineering,
- Chen, Q., Song, Z., Dong, J., Huang, Z., Hua, Y., & Yan, S. (2014). Contextualizing object detection and classification. *IEEE transactions on pattern analysis and machine intelligence*, 37(1), 13-27.
- Congan, X., Hang, S., Jianwei, L., Yu, L., Libo, Y., Long, G., Wenjun, Y., & Taoyang, W. (2022). RSDD-SAR: Rotated ship detection dataset in SAR images. *雷达学报*, 11(4), 581-599.

- Dragosevic, M. V., & Vachon, P. W. (2008). Estimation of ship radial speed by adaptive processing of RADARSAT-1 fine mode data. *IEEE Geoscience and Remote Sensing Letters*, 5(4), 678-682.
- Eldhuset, K. (1996). An automatic ship and ship wake detection system for spaceborne SAR images in coastal regions. *IEEE transactions on Geoscience and Remote Sensing*, 34(4), 1010-1019.
- Girshick, R. (2015). Fast r-cnn. *arXiv preprint arXiv:1504.08083*.
- Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. Proceedings of the IEEE conference on computer vision and pattern recognition,
- He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. Proceedings of the IEEE international conference on computer vision,
- Hijazi, S., Kumar, R., & Rowen, C. (2015). Using convolutional neural networks for image recognition. *Cadence Design Systems Inc.: San Jose, CA, USA*, 9(1).
- Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., & Sainath, T. N. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, 29(6), 82-97.
- Huang, Q., Sun, H., Wang, Y., Yuan, Y., Guo, X., & Gao, Q. (2024). Ship detection based on YOLO algorithm for visible images. *IET Image Processing*, 18(2), 481-492.
- Ia, G. (2016). Deep learning/Ian Goodfellow, Yoshua Bengio and Aaron Courville. In: Cambridge, Massachusetts: The MIT Press.
- Ioffe, S. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.
- Jiang, J., Fu, X., Qin, R., Wang, X., & Ma, Z. (2021). High-speed lightweight ship detection algorithm based on YOLO-v4 for three-channels RGB SAR image. *Remote Sensing*, 13(10), 1909.
- Jocher, G., Chaurasia, A., Stoken, A., Borovec, J., Kwon, Y., Michael, K., Fang, J., Yifu, Z., Wong, C., & Montes, D. (2022). ultralytics/yolov5: v7. 0-yolov5 sota realtime instance segmentation. *Zenodo*.
- Johansen, T. A. (1997). On Tikhonov regularization, bias and variance in nonlinear system identification. *Automatica*, 33(3), 441-446.
- Kızılkaya, S., Alganci, U., & Sertel, E. (2022). VHRShips: An extensive benchmark dataset for scalable deep learning-based ship detection applications. *ISPRS International Journal of Geo-Information*, 11(8), 445.

- Kniazieva, Y. (2024). *Object Detection: Key Metrics for Computer Vision Model Performance*. <https://labeledyourdata.com/articles/object-detection-metrics>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
- Li, X., & Chong, J. (2008). Processing of Envisat alternating polarization data for vessel detection. *IEEE Geoscience and Remote Sensing Letters*, 5(2), 271-275.
- Lin, Z., Ji, K., Leng, X., & Kuang, G. (2018). Squeeze and excitation rank faster R-CNN for ship detection in SAR images. *IEEE Geoscience and Remote Sensing Letters*, 16(5), 751-755.
- Liu, H., Sun, F., Gu, J., & Deng, L. (2022). Sf-yolov5: A lightweight small object detection algorithm based on improved feature fusion mode. *Sensors*, 22(15), 5817.
- Liu, Q., Xiang, X., Wang, Y., Luo, Z., & Fang, F. (2020). Aircraft detection in remote sensing image based on corner clustering and deep learning. *Engineering Applications of Artificial Intelligence*, 87, 103333.
- Mehran, A., Tehsin, S., & Hamza, M. (2023). An effective deep learning model for ship detection from satellite images. *Spatial information research*, 31(1), 61-72.
- Mitra, A., Mohanty, S. P., & Kougianos, E. (2023). aGRodet 2.0: An Automated Real-Time Approach for Multiclass Plant Disease Detection. *SN Computer Science*, 4(5), 657.
- Ophoff, T., Puttemans, S., Kalogirou, V., Robin, J.-P., & Goedeme, T. (2020). Vehicle and vessel detection on satellite imagery: A comparative study on single-shot detectors. *Remote Sensing*, 12(7), 1217.
- Parambil, M. M. A., Ali, L., Swavaf, M., Bouktif, S., Gochoo, M., Aljassmi, H., & Alnajjar, F. (2024). Navigating the YOLO landscape: A comparative study of Object detection models for Emotion Recognition. *Ieee Access*.
- Patel, K., Bhatt, C., & Mazzeo, P. L. (2022a). Deep learning-based automatic detection of ships: An experimental study using satellite images. *Journal of imaging*, 8(7), 182.
- Patel, K., Bhatt, C., & Mazzeo, P. L. (2022b). Improved ship detection algorithm from satellite images using YOLOv7 and graph neural network. *Algorithms*, 15(12), 473.
- Pei, S., Song, H., & Lu, Y. (2022). Small sample hyperspectral image classification method based on dual-channel spectral enhancement network. *Electronics*, 11(16), 2540.

- Petersson, H., Gustafsson, D., & Bergstrom, D. (2016). Hyperspectral image analysis using deep learning—A review. 2016 sixth international conference on image processing theory, tools and applications (IPTA),
- Pires, C., Damas, B., & Bernardino, A. (2022). An efficient cascaded model for ship segmentation in aerial images. *Ieee Access*, 10, 31942-31954.
- Poole, D. L., & Mackworth, A. K. (2010). *Artificial Intelligence: foundations of computational agents*. Cambridge University Press.
- Prechelt, L. (1998). Automatic early stopping using cross validation: quantifying the criteria. *Neural networks*, 11(4), 761-767.
- Purves, D. (2019). *Brains as engines of association: an operating principle for nervous systems*. Oxford University Press.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. Proceedings of the IEEE conference on computer vision and pattern recognition,
- Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. Proceedings of the IEEE conference on computer vision and pattern recognition,
- Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6), 1137-1149.
- Santurkar, S., Tsipras, D., Ilyas, A., & Madry, A. (2018). How does batch normalization help optimization? *Advances in neural information processing systems*, 31.
- Schöller, F. E., Plenge-Feidenhans, M. K., Stets, J. D., & Blanke, M. (2019). Assessing deep-learning methods for object detection at sea from LWIR images. *IFAC-PapersOnLine*, 52(21), 64-71.
- Schwartz, J. H., Kandel, E. R., Jessell, T. M., Siegelbaum, S. A., & Hudspeth, A. (1991). *Principles of neural science*. Elsevier New York.
- Stofa, M. M., Zulkifley, M. A., & Zaki, S. Z. M. (2020). A deep learning approach to ship detection using satellite imagery. IOP Conference Series: Earth and Environmental Science,
- Sun, X., Zhang, Y., Wang, H., & Du, Y. (2022). Research on ship detection of optical remote sensing image based on Yolo V5. Journal of Physics: Conference Series,
- Sutskever, I., Martens, J., & Hinton, G. E. (2011). Generating text with recurrent neural networks. Proceedings of the 28th international conference on machine learning (ICML-11),

- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE conference on computer vision and pattern recognition*,
- Ting, L., Baijun, Z., Yongsheng, Z., & Shun, Y. (2021). Ship detection algorithm based on improved YOLO V5. 2021 6th International Conference on Automation, Control and Robotics Engineering (CACRE),
- Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018(1), 7068349.
- 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. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*,
- Wang, C.-Y., Yeh, I.-H., & Liao, H.-Y. M. (2024). YOLOv9: Learning what you want to learn using programmable gradient information. *arXiv preprint arXiv:2402.13616*.
- Wang, C., Yeh, I., & Liao, H. YOLOv9: Learning what you want to learn using programmable gradient information. *arXiv 2024. arXiv preprint arXiv:2402.13616*.
- Wang, J., & Perez, L. (2017). The effectiveness of data augmentation in image classification using deep learning. *Convolutional Neural Networks Vis. Recognit*, 11(2017), 1-8.
- Wei, S., Zeng, X., Qu, Q., Wang, M., Su, H., & Shi, J. (2020). HRSID: A high-resolution SAR images dataset for ship detection and instance segmentation. *Ieee Access*, 8, 120234-120254.
- Woo, S., Park, J., Lee, J.-Y., & Kweon, I. S. (2018). Cbam: Convolutional block attention module. *Proceedings of the European conference on computer vision (ECCV)*,
- Xian, S., Zhirui, W., Yuanrui, S., Wenhui, D., Yue, Z., & Kun, F. (2019). AIR-SARShip-1.0: High-resolution SAR ship detection dataset. *雷达学报*, 8(6), 852-863.
- Yadav, S., Gulia, P., Gill, N. S., & Chatterjee, J. M. (2022). [Retracted] A Real-Time Crowd Monitoring and Management System for Social Distance Classification and Healthcare Using Deep Learning. *Journal of Healthcare Engineering*, 2022(1), 2130172.
- Yusuf, M. O., Hanzla, M., Al Mudawi, N., Sadiq, T., Alabdullah, B., Rahman, H., & Algarni, A. Target Detection and Classification via EfficientDet and CNN over Unmanned Aerial Vehicles. *Frontiers in Neurorobotics*, 18, 1448538.

Zhang, T., Zhang, X., Li, J., Xu, X., Wang, B., Zhan, X., Xu, Y., Ke, X., Zeng, T., & Su, H. (2021). SAR ship detection dataset (SSDD): Official release and comprehensive data analysis. *Remote Sensing*, 13(18), 3690.



ÖZGEÇMİŞ

Adı Soyadı	Mehmet Sami TÜRKER
Doğum Tarihi	
Yabancı Dil	İngilizce
Eğitim Durumu	
Lisans	Kırıkkale Üniversitesi – Bilgisayar Mühendisliği
Yüksek Lisans	Kırıkkale Üniversitesi – Savunma Teknolojileri
Çalıştığı Kurum ve Yıllar	Sahil Güvenlik Komutanlığı 2020/...

Yayınları	International Conference on Computational Mathematics and Engineering Sciences (CMES-2024). Performance Comparison of Deep Learning Algorithms For Ship Detection And Classification From Satellite Imagery
-----------	--