



AKSARAY ÜNİVERSİTESİ

SOSYAL BİLİMLER ENSTİTÜSÜ

YÖNETİM BİLİŞİM SİSTEMLERİ ANA BİLİM DALI

**RÖNTGEN GÖRÜNTÜLERİNİN DERİN ÖĞRENME
ALGORİTMALARIYLA SINIFLANDIRILMASI**

YÜKSEK LİSANS TEZİ

DİNÇER AYDİÇ

DANIŞMAN

DR. ÖĞR. ÜYESİ TARIK YILMAZ

AKSARAY-2024



AKSARAY ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
YÖNETİM BİLİŞİM SİSTEMLERİ ANA BİLİM DALI

**RÖNTGEN GÖRÜNTÜLERİNİN DERİN ÖĞRENME
ALGORİTMALARIYLA SINIFLANDIRILMASI**

YÜKSEK LİSANS TEZİ

DİNÇER AYDİÇ

DANIŞMAN : DR. ÖĞR. ÜYESİ TARIK YILMAZ

2. DANIŞMAN: DOÇ. DR. SADULLAH TURHAN

JÜRİ ÜYESİ : PROF. DR. ZELİHA SEÇKİN

JÜRİ ÜYESİ : PROF. DR. HASAN ERDİNÇ KOÇER

AKSARAY-2024

TELİF HAKLARI VE TEZ FOTOKOPİ İZİN FORMU

Bu tezin tüm hakları saklıdır. Kaynak göstermek koşuluyla tezin teslim tarihinden itibaren 6 (altı) ay sonra tezden fotokopi çekilebilir.

YAZARIN

Adı : Dinçer
Soyadı : AYDIÇ
Bölümü : Yönetim Bilişim Sistemleri
İmza :
Teslim Tarihi : 28/06/2024

TEZİN

Türkçe Adı : Röntgen Görüntülerinin Derin Öğrenme Algoritmalarıyla Sınıflandırılması

İngilizce Adı : Classification of X-Ray Images with Deep Learning Algorithms

ETİK İLKELERE UYGUNLUK BEYANI

Tez yazma sürecinde bilimsel ve etik ilkelere uyduğumu, yararlandığım tüm kaynakları kaynak gösterme ilkelerine uygun olarak kaynakçada belirttiğimi ve bu bölümler dışındaki tüm ifadelerin şahsıma ait olduğunu beyan ederim.

Dinçer AYDİÇ

İmza:

T.C.
AKSARAY ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
JÜRİ ONAY SAYFASI

Dinçer AYDİÇ tarafından hazırlanan “Röntgen Görüntülerinin Derin Öğrenme Algoritmalarıyla Sınıflandırılması” başlıklı tez çalışması aşağıdaki jüri tarafından oy birliği / oy çokluğu ile Aksaray Üniversitesi Sosyal Bilimler Enstitüsü Yönetim Bilişim Sistemleri Anabilim Dalı’nda Yüksek Lisans tezi olarak kabul edilmiştir.

Danışman: Dr. Öğr. Üyesi Tarık YILMAZ

(Yönetim Bilişim Sistemleri, Aksaray Üniversitesi)

Üye: Prof. Dr. Zeliha SEÇKİN

(Yönetim Bilişim Sistemleri, Aksaray Üniversitesi)

Üye: Prof. Dr. Hasan Erdinç KOÇER

(Devreler ve Sistemler Anabilim Dalı/Selçuk Üniversitesi)

Tez Savunma Tarihi: 24/06/2024

Sosyal Bilimler Enstitüsü Yönetim Kurulu’nuntarih ve sayılı kararı ile onaylanmıştır.

Sosyal Bilimler Enstitüsü Müdürü

Prof. Dr. Aybala ÇAYIR

İmza

TEŞEKKÜR SAYFASI

Çalışmanın her aşamasında desteğini esirgemeyen ve yol gösteren danışmanım Dr. Öğr. Üyesi Tarık YILMAZ'a, uygulama geliştirme sürecinin en önemli aşamalarından biri olan veri setinin etiketlenmesi aşamasında verdiği değerli katkılarından dolayı Antalya Eğitim Araştırma Hastanesi, Ortopedi ve Travmatoloji Uzmanı hocam Doç. Dr. Sadullah TURHAN'a, ihtiyaç duyduğum her durumda yardımlarını esirgemeyen hocam Zeliha SEÇKİN'e, araştırmalarım sırasında verdiği desteklerden dolayı arkadaşım Muhammed ÖZDEMİR'e sonsuz teşekkür ederim.

Hayatımın her anında olduğu gibi bu süreçte de bana destek olan sevgili eşim Özlem GÜNEŞ AYDİÇ'e, kızım Bilge'ye ve oğlum Barış'a, dualarını üstümden eksik etmeyen annem Meliha AYDİÇ ve kıymetli babam Selahattin AYDİÇ'e, motivasyonuma katkı sağlayan kardeşim Hacer ve eşi Kerem GÜN'e çok teşekkür ederim.

ÖZET

Günümüzde yapay zeka algoritmaları, hızları ve doğrulukları nedeniyle modern sağlık sistemlerinde, personellere yardımcı olmak adına, sıklıkla kullanılmaktadır. Ortopedi servisinin olmadığı ya da sağlık sistemine uzak mesafede bulunan yerlerde yaşayan bireylerin yaşadıkları kemik kırığı problemlerinin hızlı ve doğru bir şekilde teşhis edilmesi önem arz etmektedir. Bu çalışmada, röntgen görüntüleri derin öğrenme algoritmalarıyla işlenerek, kırık olup olmadığı, varsa kırığın yerinin tespit edilmesi amaçlanmaktadır. Çalışmada; You Only Look Once (YOLOv8 ve YOLOv9) algoritmaları kullanılmıştır. Yapılan değerlendirmeler, en iyi sonucun, gerçek zamanlı nesne tespitine imkân tanıyan ve en güncel algoritma olarak bilinen YOLO algoritmasının 9. versiyonu ile alındığını göstermiştir. Geliştirilen yeni yaklaşım ve kullanılan YOLOv9 algoritmasıyla, %97 mAP50 değerine ulaşılmıştır. Derin öğrenme modelinin geliştirilmesi aşamasında, Stanford Üniversitesi Makine Öğrenmesi Grubu tarafından oluşturulan ve halka açık en büyük radyografik görüntü veri kümelerinden biri olan MURA v1.1 veri seti kullanılmıştır. Bu veri seti, çeşitli kırık türlerini içeren geniş bir yelpazeye sahip olup, modelin eğitimi ve test edilmesi için idealdir. Sonuç olarak, geliştirilen yöntem ile profesyonel sağlık hizmetlerine ulaşımın zor ya da imkânsız olduğu yerlerde yaşayan bireylerin yaşadığı kemik kırığı problemlerini hızlı ve doğru bir şekilde teşhis etmek ve aynı zamanda acil servislerdeki yoğunluk kaynaklı teşhis hatalarını en aza indirmek üzere bir uygulama geliştirilmiştir.

Anahtar Kelimeler: Derin öğrenme, Kırık teşhisi, YOLO, CNN, R-CNN, Röntgen görüntüleri

Bilim Kodu: 114614

Sayfa Adedi: 71

Yazar: Dinçer AYDİÇ

Danışman: Dr. Öğr. Üyesi Tarık YILMAZ

ABSTRACT

Nowadays, artificial intelligence algorithms are frequently used in modern healthcare systems to assist staff due to their speed and accuracy. It is important to quickly and accurately diagnose bone fracture problems experienced by individuals living in places where there is no orthopedic services or far away from the health care system. In this study, X-ray images are processed with deep learning algorithms to determine whether there is a fracture or not, and if there is a fracture, to detect the location of the fracture. You Only Look Once (YOLOv8 and YOLOv9) algorithms were used in the study. The evaluations showed that the best results were obtained with the 9th version of the YOLO algorithm, which is known as the most up-to-date algorithm that allows real-time object detection. With the new approach we have developed and with usage of YOLOv9 algorithm, a mAP50 value of 97% was achieved. The MURA v1.1 dataset, which is one of the largest publicly available radiographic image datasets created by the Stanford University Machine Learning Group, was used to develop the deep learning model. This dataset contains a wide range of fracture types and is ideal for training and testing the model. As a result, an application has been developed to quickly and accurately diagnose bone fracture problems experienced by individuals living in areas where access to professional healthcare services is difficult or impossible, and at the same time minimize the diagnostic errors due to overcrowding in emergency services.

Keywords: Deep learning, Fracture diagnosis, YOLO, CNN, R-CNN, X-ray images

Science Code: 114614

Number of Pages: 71

Author: Dinçer AYDİÇ

Supervisor: Dr. Öğr. Üyesi Tarık YILMAZ

İÇİNDEKİLER

TELİF HAKLARI VE TEZ FOTOKOPİ İZİN FORMU	iv
ETİK İLKELERE UYGUNLUK BEYANI	v
TEŞEKKÜR SAYFASI	vii
ÖZET	viii
ABSTRACT	ix
İÇİNDEKİLER	x
TABLolar LİSTESİ	xii
ŞEKİLLER LİSTESİ	xiii
KISALTMALAR	xiv
GİRİŞ	1
BİRİNCİ BÖLÜM	3
LİTERATÜR TARAMASI	3
İKİNCİ BÖLÜM	15
DERİN ÖĞRENME ALGORİTMALARINA GENEL BİR BAKIŞ	15
2.1. Yapay Sinir Ağları (ANN – Artificial Neural Networks)	15
2.1.1. Yapay Sinir Ağlarının Faydaları	16
2.1.2. İnsan Beyni ve Sinir Ağları Arasındaki İlişki	17
2.2. Evrişimli Sinir Ağları (CNN – Convolutional Neural Networks)	17
2.2.1. Üç Katman Türü	18
2.2.2. Bağlantı ve Optimizasyon	19
2.3. R-CNN - Bölge Tabanlı Evrişimli Sinir Ağı	19
2.4. Fast R-CNN	20
2.4.1. Neden Fast R-CNN, R-CNN'den daha hızlıdır?	21
2.5. Faster R-CNN	21
2.5.1. Fast R-CNN ile Faster R-CNN arasındaki farklar	22
2.6. YOLO (1. Versiyon)	23
2.7. Karışıklık matrisi (Confusion Matrix)	26
2.8. Kesinlik (Precision), Hassaslık (Recall) ve F_1 Skoru	26
ÜÇÜNCÜ BÖLÜM	28

MODEL EĞİTİMLERİ.....	28
3.1. Veri Setinin Oluşturulması.....	28
3.1.1. Veri Setinin İncelenmesi ve Hataların Düzeltilmesi	28
3.1.2. Görsellerin Etiketlenmesi ve Doğrulanması.....	28
3.2. Model Oluşturma	29
3.2.1. Modelin Oluşturulması ve Eğitimi	29
3.2.2 Model Eğitim Süreci.....	29
3.2.2 Model Eğitim Süreci 1. Aşama	30
3.2.2 Model Eğitimi 2. Aşama.....	32
3.2.2 Model Eğitimi 3. Aşama.....	35
3.2.3 Model Eğitimi 4. Aşama.....	37
3.2.3 Model Eğitimi 5. Aşama.....	39
3.2.3 Model Eğitimi 6. Aşama.....	41
3.2.3 Sonuç Grafiklerinin Yorumlanması	42
3.2.4 Yeni Yaklaşımla Geliştirilen 3 Modelin Sonuçlarının İncelenmesi.....	43
3.2.5 YOLOv8 ile YOLOv9 Arasındaki Performans Tablolarının Karşılaştırmalı İncelenmesi.....	44
3.2.6 YOLOv8 ile YOLOv9 ile Eğitilen Modellerin Performans Grafiklerinin Karşılaştırmalı İncelemesi	46
DÖRDÜNCÜ BÖLÜM	49
UYGULAMA GELİŞTİRME	49
4.1. Uygulama Akış Diyagramı	49
4.2. Kullanılan Teknolojiler	50
4.3. Uygulama Arayüzü	50
4.3.1 Örnek Görseller	51
SONUÇ.....	52
KAYNAKÇA	53
AKSARAY ÜNİVERSİTESİ TEZ DEĞERLENDİRME FORMU	56

TABLÖLAR LİSTESİ

Tablo 1. Son Teknoloji Ürünü Gerçek Zamanlı Nesne Tespiti Algoritmalarının Karşılaştırılması	25
Tablo 2. Karmaşıklık Matrisi	26
Tablo 3. YOLOv8 Performans Ölçütleri.....	29
Tablo 4. Model Eğitimi 1. Aşama Tahmin Sonuçları	31
Tablo 5. Model Eğitimi 2. Aşama Tahmin Sonuçları	33
Tablo 6. Model Eğitimi 3. Aşama Tahmin Sonuçları	35
Tablo 7. Model Eğitimi 4. Aşama Tahmin Sonuçları	38
Tablo 8. Model Eğitimi 5. Aşama Tahmin Sonuçları	39
Tablo 9. Model Eğitimi 6. Aşama Tahmin Sonuçları	41
Tablo 10. Yeni Yaklaşımla Eğitilen Modellerin Eğitim Performansları	43
Tablo 11. YOLOv9c ile Eğitilen Modellerin Parametre ve Performansları	44
Tablo 12. Model Eğitimi 4. Aşama ile 7. Aşama Performans Karşılaştırması	45
Tablo 13. Model Eğitimi 5. Aşama ile 8. Aşama Performans Karşılaştırması	45
Tablo 14. Model Eğitimi 6. Aşama ile 9. Aşama Performans Karşılaştırması	46

ŞEKİLLER LİSTESİ

Şekil 1. Yapay Sinir Ağı.....	15
Şekil 2. Çok Katmanlı Yapay Sinir Ağı	15
Şekil 3. Sadece Beş Katmandan Oluşan Basit Bir CNN Mimarisi.....	18
Şekil 4. R-CNN: Bölge Tabanlı Evrişimli Sinir Ağı	19
Şekil 5. Fast R- CNN Mimarisi	20
Şekil 6. Faster R-CNN.....	22
Şekil 7. YOLO Algılama Sistemi	23
Şekil 8. MS COCO Veri Kümesindeki Gerçek Zamanlı Nesne Algılayıcılarının Karşılaştırılması.....	24
Şekil 9. Model Eğitimi 1. Aşama (A) Etiket ve (B) Tahmin Örnekleri.....	31
Şekil 10. Model Eğitimi 1. Aşama Performans Grafikleri	32
Şekil 11. Model Eğitimi 2. Aşama (A) Etiket ve (B) Tahmin Örnekleri.....	33
Şekil 12. Model Eğitimi 2. Aşama Performans Grafikleri	34
Şekil 13. Model Eğitimi 3. Aşama (A) Etiket ve (B) Tahmin Örnekleri.....	35
Şekil 14. Model Eğitimi 3. Aşama Performans Grafikleri	36
Şekil 15. Model Eğitimi 4. Aşama (A) Etiket ve (B) Tahmin Örnekleri.....	38
Şekil 16. Model Eğitimi 4. Aşama Performans Grafikleri	39
Şekil 17. Model Eğitimi 5. Aşama (A) Etiket ve (B) Tahmin Örnekleri.....	40
Şekil 18. Model Eğitimi 5. Aşama Performans Grafikleri	40
Şekil 19. Model Eğitimi 6. Aşama (A) Etiket ve (B) Tahmin Örnekleri.....	41
Şekil 20. Model Eğitimi 6. Aşama Performans Grafikleri	42
Şekil 21. Model Eğitimi 7. Aşama Performans Grafikleri	46
Şekil 22. Model Eğitimi 8. Aşama Performans Grafikleri	47
Şekil 23. Model Eğitimi 9. Aşama Performans Grafikleri	48
Şekil 24. Uygulama Akış Diyagramı.....	49
Şekil 25. Uygulama Arayüzü.....	50
Şekil 26. Uygulamanın Yaptığı Tahmin Örnekleri.....	51

KISALTMALAR

ALU:	Arithmetic Logic Unit (Aritmetik Mantık Birimi)
ANN:	Artificial Neural Networks (Yapay Sinir Ağları)
AP:	Average Precision (Ortalama Hassasiyet)
ASIC:	Application-Specific Integrated Circuit (Özel Amaçlı Entegre Devre)
AUC:	Area Under the Curve
AUROC:	Area Under the Receiver Operating Characteristic Curve (Alıcı İşletim Özelliği Eğrisinin Altındaki Alan)
CAD:	Computer Aided Diagnosis (Bilgisayar Destekli Tanılama)
CNN:	Convolutional Neural Network (Evrişimli Sinir Ağları)
ConvNets:	Convolutional Neural Networks (Evrişimli Sinir Ağları)
CPU:	Central Processing Unit (Merkezi İşlem Birimi)
FPGA:	Field-Programmable Gate Array (Alan Programlanabilir Kapı Dizisi)
GELAN:	Generalized Efficient Layer Aggregation Network (Genelleştirilmiş Etkin Katman Toplama Ağı)
mAP:	mean Average Precision (Ortalama Doğruluk Ortalaması)
MS COCO:	Microsoft Common Objects in Context (Microsoft Bağlamsal Ortak Nesneler)
MURA:	MUsculoskeletal Radiographs (kas-iskelet sistemi radyografileri)
NoC:	Networks on Convolutional Feature Maps (Evrişimsel Özellik Haritaları Üzerindeki Ağlar)
PASCAL VOC:	Pattern Analysis, Statistical Modelling and Computational Learning Visual Object Classes (PASCAL Görsel Nesne Sınıfları)
PGI:	Programmable Gradient Information (Programlanabilir Gradyan Bilgisi)
PRISMA:	Preferred Reporting Items for Systematic Reviews and Meta-Analyses (Sistematik İncelemeler ve Meta-Analizler için Tercih Edilen Raporlama Öğeleri)
QDR-SRAM:	Quad Data Rate Synchronous Dynamic Random Access Memory (Yüksek Hızlı Çift Veri Hızı Senkron Dinamik Rastgele Erişim Belleği)
RBF:	Radial Basis Function (Radyal Baz Fonksiyonu)
R-CNN:	Region-based Convolutional Neural Network (Bölge Tabanlı Evrişimli Sinir Ağı)

ResNet:	Residual Neural Network (Artık Sinir Ağı)
RoI:	Region of Interest (İlgi Alanı)
RPN:	Region Proposal Network (Bölge Önerme Ağı)
SOFM:	Self-Organizing Feature Map (Kendi Kendini Düzenleyen Özellik Haritası)
SPPnet:	Spatial Pyramid Pooling Network (Uzamsal Piramit Havuzlama Ağı)
SVM:	Support Vector Machines (Destek Vektör Makineleri)
VGG:	Visual Geometry Group (Görsel Geometri Grubu)
YOLO:	You Only Look Once (Sadece Bir Kere Bakarsın)
YZ:	Yapay Zeka



GİRİŞ

Günümüzde tıbbi görüntüleme teknolojilerindeki hızlı gelişmeler, tanı ve tedavi süreçlerinde önemli dönüşümlere yol açmıştır. Bu teknolojik ilerlemeler, sağlık hizmetlerinin daha verimli ve etkili hale gelmesini sağlamış, hastaların yaşam kalitesini artırmak adına kritik adımlar atılmasına olanak tanımıştır. Özellikle kemik rahatsızlıklarının erken teşhisi ve tedavisi, hastaların uzun vadeli sağlık durumları üzerinde belirleyici bir rol oynamaktadır. Kemik hastalıklarının erken evrelerde doğru ve hızlı bir şekilde tespit edilmesi hem tedavi sürecinin başarısını artırmakta hem de hastaların gereksiz ağrı ve komplikasyonlardan korunmasına yardımcı olmaktadır.

Bu bağlamda, yapay zeka ve derin öğrenme tekniklerinin tıbbi görüntüleme alanında sunduğu potansiyel çözümler, geleneksel teşhis yöntemlerine kıyasla daha hızlı, doğru ve etkili sonuçlar elde etme imkanı sunmaktadır. Yapay zeka algoritmaları, büyük veri setleri üzerinde eğitim alarak tıbbi görüntülerdeki karmaşık desenleri ve anomalileri tanımlama yeteneğine sahiptir. Bu sayede, doktorların teşhis süreçlerinde zaman kazanmalarını ve hata paylarını minimize etmelerini sağlamaktadır.

Bu çalışmada, kemik kırıklarının tespiti ve teşhisi sürecinde yapay zeka algoritmalarının kullanımı derinlemesine incelemektedir. Özellikle, MURA (Musculoskeletal Radiographs) veri seti üzerinde eğitilen YOLO (You Only Look Once) modelinin, kemiklerde meydana gelen problemlerin hızlı ve etkili bir şekilde tespit edilmesini sağlama potansiyeli araştırılmıştır. YOLO modeli, gerçek zamanlı nesne tespiti konusunda üstün performans sergileyen bir derin öğrenme algoritmasıdır ve tıbbi görüntüleme alanında önemli bir yenilik olarak değerlendirilmektedir.

Çalışmada, YOLO modelinin kullanımının yanı sıra, nesne tespiti alanında önemli bir yere sahip olan yapay sinir ağları (ANN), evrişimli sinir ağları (CNN), Bölge Tabanlı Evrişimli Sinir Ağı (R-CNN), Fast R-CNN ve Faster R-CNN gibi yöntemlerin prensipleri ve aralarındaki farklar da ele alınmıştır. Bu algoritmaların her biri, tıbbi görüntülemede farklı avantajlar sunmakta olup, belirli senaryolarda üstün performans göstermektedir. Bu nedenle, bu algoritmaların detaylı bir şekilde incelenmesi ve karşılaştırılması, kemik rahatsızlıklarının tespiti için en uygun yöntemin belirlenmesine katkı sağlayacaktır.

Çalışmanın birinci bölümünde, literatürde bu konuda yapılmış olan çalışmalar, kullanılan algoritmalar ve benzer nitelikteki araştırmalar incelenerek özetlenmiştir. Bu bölümde, daha önceki çalışmaların bulguları ve kullanılan metodolojiler ayrıntılı olarak ele alınmış, mevcut literatürdeki boşluklar ve geliştirilmesi gereken alanlar belirlenmiştir. İkinci bölümde; çalışmada kullanılan yöntem ve teknikler ayrıntılı bir şekilde açıklanmıştır. Bu bölümde; veri setinin hazırlanmasından modelin eğitime kadar tüm süreçler detaylandırılmış, kullanılan algoritmaların uygulama adımları sunulmuştur. Üçüncü bölümde; model geliştirme süreçleri detaylandırılmıştır. Dördüncü bölümde ise geliştirilen modelin kolaylıkla kullanılması için bir web uygulaması geliştirilmiş ve çevrimiçi ortamda kullanıma sunulmuştur. Sonuç bölümünde çalışma süresince elde edilen bulgular ve bunların klinik uygulamalara potansiyel etkileri tartışılmıştır.

Sonuç olarak, bu tez çalışması, yapay zekâ ve derin öğrenme tekniklerinin tıbbi görüntüleme alanında sunduğu yenilikçi çözümleri vurgulamakta ve kemik rahatsızlıklarının tespiti sürecinde bu tekniklerin etkinliğini ortaya koymayı amaçlamaktadır. Elde edilen bulguların, gelecekteki araştırmalar ve klinik uygulamalar için değerli bir kaynak oluşturacağı ve sağlık hizmetlerinin kalitesini artırmaya katkı sağlayacağı öngörülmektedir.

BİRİNCİ BÖLÜM

LİTERATÜR TARAMASI

Kemik kırıklarının tespitinde derin öğrenme algoritmalarının kullanımını ele alan akademik çalışmalar, çeşitli yöntemler ve uygulamalarla önemli bulgular ortaya koymaktadır. Bu çalışmalar, derin öğrenme teknolojilerinin kemik kırıklarının doğru ve hızlı bir şekilde tanımlanmasındaki etkinliğini ve potansiyelini göstermektedir. Aşağıdaki bölümde, bu alanda gerçekleştirilen araştırmaların ana hatları ve elde edilen sonuçlar özetlenmiştir.

Açık kaynaklı MURA veri setini kullanarak ve 26 önceden eğitilmiş derin öğrenme modelini değerlendirdikleri çalışmalarında Uysal vd., (2021); iki farklı kümülatif/topluluk öğrenme (ensemble learning) modeli geliştirmişlerdir. Geliştirdikleri bu modeller EL1 ve EL2 olarak adlandırılmıştır. L1 modelinde, çalışmalarında değerlendirdikleri modellerden en iyi sonuç veren üç modeli (ResNeXt50, DenseNet169 ve DenseNet201) birleştirilmiş tek bir tam bağlantılı katman (fully connected layer) oluşturarak AUC değerini en üst düzeye çıkarmayı hedeflemişlerdir. EL2 modelinde ise üzerinde çalışılan veri setindeki her sınıf için en iyi modeli seçen bir karar sistemi kullanmışlardır. Sonuç olarak EL2 modeliyle en yüksek test doğruluğuna ve en iyi Cohen'in kappa değerine ulaşılmıştır. EL1 modeliyle elde edilen sonuçlar incelendiğinde en yüksek AUC değerine ulaşıldığı tespit edilmiştir. Bu çalışma, omuz kırıklarının tespiti için topluluk öğrenme modellerinin kullanılabileceğini ve öğrenme transferinin modellerin başarısını artırdığını göstererek alana önemli katkılar sağlamıştır.

Röntgen, özellikle de el röntgen görüntülerinde denetimsiz öğrenme (unsupervised learning) yoluyla anomali tespitini ele alan “Unsupervised anomaly detection for X-ray images” adlı çalışmalarında Davletshina vd., (2020); denetimsiz öğrenme modellerinin performanslarını artırmak için bir ön işleme ardışık düzeni (pipeline) kullanmışlardır. Otokodlayıcılar (autoencoders) ve üretken çekişmeli ağlar (Generative Adversarial Networks - GANs) gibi denetimsiz öğrenme yöntemlerini kullanarak anomali tespiti yaklaşımlarını incelemişler ve ön işleme adımlarının anomali tespitinde algoritmaların performanslarını önemli ölçüde artırdığını bulmuşlardır. Otokodlayıcılar, model tahminlerinin yorumlanmasını kolaylaştıran piksel seviyesinde anomali ısı haritaları üretme avantajı sağlarken, GAN tabanlı yaklaşımların, nispeten daha gürültülü verilerle daha iyi uyum sağlayabileceğini ifade etmişlerdir. Sonuç olarak, çalışmada denetimsiz öğrenme yöntemlerinin, özellikle ön işleme

ile birlikte kullanıldığında, röntgen görüntülerinde anomali tespit etmek için umut vadettiği belirtilmiştir.

Khatik, ve Kadam (2022); kemik kırıklarını tespit etmek için evrişimli sinir ağları (CNN) tabanlı modellerin kullanımına odaklandıkları sistematik derleme çalışmalarını, PRISMA yönergelerine göre yapmışlardır. Çalışmada çeşitli kemik türlerindeki kırıkları tespit etmek için farklı CNN yaklaşımlarını incelemekte ve bu yaklaşımların doğruluğu analiz edilmektedir. Sonuç olarak, CNN'nin kemik kırığı tespitinde oldukça etkili olduğu, özellikle de kalsineus, kol, bilek, uyluk, tibia, kafa, üst gövde, alt gövde, alt ekstremit ve üst ekstremit gibi çeşitli kemik türleri için doğru sonuçlar elde edildiği gösterilmektedir. Ancak, tüm kemik türlerindeki kırıkları tespit etmek için evrensel bir yaklaşımın bulunmadığı ve belirli kemik türleri için özel olarak tasarlanmış modellerin daha yüksek doğruluk oranları sağladığı gözlemlenmiştir. Çalışmada elde edilen başarı oranının farklı CNN yaklaşımlarının entegre edilmesiyle sağlanabileceği, kemik kırığı tespitinde daha yüksek doğruluk oranları elde eden hibrit modellerin geliştirilmesi gerektiği vurgulanmıştır.

MURA standart veri setini kullanarak çeşitli kemik çalışmalarında anomalileri tespit etmek ve eğitmek için görüntü işleme tekniklerini uyguladıkları makalelerinde Kumar vd., (2020); kemik kırıklarını otomatik olarak tespit etmek için bilgisayar destekli bir sistem geliştirmeyi amaçlamışlardır. Sistem kenarları tespit etmek için röntgen görüntülerini bir ön işleme sürecinden geçirerek görüntüleri normal ve anormal olarak sınıflandırarak anomalileri belirlemektedir. Çalışmanın deneysel sonuçları, farklı kemiklerdeki anormallikleri tespit etmedeki sistemin doğruluğunu göstermektedir. Araştırmacılar ayrıca, ResNet ve DenseNet modellerinin performansını analiz ederek sistemi bir kullanıcı arayüzü ile nasıl entegre ettiklerini açıklamaktadırlar. Bu çalışmada kemik kırığı tespiti için otomatik bir sistem oluşturma potansiyelini göstermekte ve sistemin daha fazla veri seti kullanılarak daha da iyileştirilebileceği öne sürülmektedir.

“X-ray bone fracture classification using deep learning: a baseline for designing a reliable approach” adlı çalışmalarında, Tanzi vd., (2020); derin öğrenme algoritmalarını kullanarak röntgen görüntülerindeki kemik kırığı sınıflandırması için güvenilir bir yaklaşım tasarlamak üzere bir temel oluşturmayı amaçlamaktadırlar. Çalışmada, derin öğrenme teknikleri kullanılarak kemik kırıklarının sınıflandırması üzerine yazılmış bir dizi makale incelenmiş ve bulguları değerlendirilmiştir. Çalışmada her bir çalışmanın AUC değeri, test doğruluğu, duyarlılık, özgüllük, veri seti boyutu ve etiketleme güvenilirliği olmak üzere altı metriktenden elde edilen değerler bir radar grafiğiyle özetlenip değerlendirilmiş, bu amacı

gerçekleştirmeye çalışırken dikkate alınması gereken temel noktalar belirlenmiş ve her çalışma temel hatlarıyla karşılaştırılmıştır. Sonuç olarak evrişimli sinir ağları ve diğer derin öğrenme yaklaşımları kullanılarak kemik kırığı sınıflandırmasında insanlarla kıyaslanabilir sonuçlar elde edilmiştir. Doğru bir genelleme benimseyerek, doktorlara yardımcı olmak için doğru tasarlanmış bir bilgisayar destekli tanı (CAD) sisteminin, önemli miktarda zaman kazandıracağı ve yanlış tanı oranını azaltacağı ifade edilmiştir.

Üç boyutlu iskelet görüntülerindeki kırık bölgelerin tespiti ve görselleştirilmesi için YOLO v4 tabanlı bir derin öğrenme modeli geliştirmişlerdir. Model, kaval kemiği ve dirsek kırıklarının tespiti için eğitilmiş ve %71 ve %81 ortalama hassasiyet değerleriyle oldukça başarılı sonuçlar üretmiştir. YOLO v4 tarafından oluşturulan sınırlayıcı kutular (bounding box), 3 boyutlu olacak şekilde yeniden yapılandırılmış kemik görüntüleriyle eşleştirilerek kırık alanlara kırmızı bir maskeyle vurgulanmış ve böylece ortopedi cerrahlarının bu kırıkların kapsamını ve konumunu daha iyi anlamaları sağlanmıştır. Çalışmada YOLO v4'ün sezgisel bir kırık tespiti ve görselleştirme aracı olarak kullanılabileceği ortaya konmuş olsa da kırık parçalarını değil sadece kırık alanlarını tespit ettiği ve sınırlı veriyle eğitilmesi gibi sınırlılıkları bulunduğu ifade edilmiştir (Jeon vd., 2023).

Overview of artificial neural networks. Artificial neural networks: methods and applications” başlıklı makalelerinde Zou vd., (2009); yapay sinir ağlarını (ANN'ler) tanıtmış, farklı mimarilerini, öğrenme algoritmalarını ve uygulamalarını kapsamlı bir şekilde ele alarak incelemişlerdir. Makalede yapay sinir ağlarının nasıl çalıştığını açıklamak için biyolojik nöronlardan yola çıkarak bir analogi kullanılmıştır. Bununla birlikte geri yayımlı (backpropagation) öğrenme algoritması, kendi kendini düzenleyen özellik haritası (SOFM) ve radyal baz fonksiyonu (RBF) ağı gibi popüler öğrenme algoritmaları ve bunların uygulamada nasıl kullanılabileceği kapsamlı bir şekilde tartışılmıştır. Makale, ANN'lerin özellikle zaman serisi tahmini, sistem tanımlaması ve desen sınıflandırma gibi karmaşık görevleri yerine getirebileceğini göstermektedir. Ayrıca ANN tasarımlarının farklı parametrelerinin, örneğin gizli nöron sayısı, başlangıç ağırlıkları ve öğrenme oranı gibi performansı nasıl etkileyebileceği de ele alınmaktadır. Sonuç olarak, makale, ANN'lerin farklı görevleri yerine getirmek için çok yönlü ve güçlü bir araç olduğunu ortaya koymaktadır.

O'Shea ve Nash (2015) tarafından kaleme alınan “An introduction to convolutional neural networks.” adlı makale, resim verilerini işlemek için özel olarak tasarlanmış bir Yapay Sinir Ağı (ANN) türü olan Evrişimli Sinir Ağları (CNN) hakkında bir giriş bilgisi sunmaktadır.

Makalede, CNN mimarisinin üç temel katmanı (evrişimli katmanlar, havuzlama katmanları ve tam bağlantılı katmanlar) açıklanmakta ve her katmanın işlevi, hiper parametreleri ve bağlantıları ayrıntılı olarak incelenmektedir. CNN mimarilerini oluşturmak için yaygın olarak kullanılan birkaç tekniğin de sunulduğu makalede CNN'lerin geleneksel ANN'lerden daha basit bir mimari kullanarak resim verileriyle çalışmak için nasıl optimize edildiğini açıklanmaktadır. Yazarlar, bu algoritmaları daha anlaşılır hale getirmeyi ve başlangıç seviyesindeki kullanıcılar için daha erişilebilir kılmayı amaçlamaktadırlar.

PASCAL VOC veri kümesinde nesne algılama performansının son yıllarda nasıl durgunlaştığını ele aldıkları makalelerinde Girshick vd., (2014); %53'lük bir mAP ederek, VOC 2012'de daha önce elde edilen en iyi ortalama kesinlik (mAP) değerini %30'dan daha fazla artıran basit ve ölçeklenebilir bir algılama algoritması önermektedirler. Yaklaşım iki temel anlayışı birleştirmektedir. Bunlar sırasıyla;

1. Nesnelerin yerlerini tespit etmek ve segmentlemek için alt-üst bölge önerilerine yüksek kapasiteli evrişimli sinir ağlarının (CNN'ler) uygulanmıştır.
2. Etiketli eğitim verilerinin yetersiz olduğu durumlarda, yardımcı bir görev için gözetimli (supervised) ön eğitim sonrasında etki alanına özgü ince ayar yapılmıştır.

Böylece önemli bir performans artışı sağlanmıştır. Bölge önerilerini CNN'lerle birleştirdikleri için yöntemlerine bölge tabanlı evrişimli sinir ağlarının kısaltması olan R-CNN adını vermişlerdir. Araştırmacılar ayrıca, R-CNN'i benzer CNN mimarisine dayalı yakın zamanda önerilen kayan pencereci bir nesne tespiti algoritması olan OverFeat ile karşılaştırmışlar ve R-CNN'nin 200 sınıflı ILSVRC2013 algılama veri kümesinde OverFeat'ten çok daha iyi performans gösterdiğini ortaya koymuşlardır.

Nesne algılama için hızlı bölge tabanlı evrişimli ağ yöntemi olan Fast R-CNN'i inceleyen Girshick, (2015) yaptığı çalışmada Fast R-CNN'in derin evrişimli ağları kullanarak nesne önerilerini verimli bir şekilde sınıflandırmak için önceki çalışmalardan yararlanarak, eğitim ve test hızını artıran ve algılama doğruluğunu yükselten çeşitli yenilikler sunduğunu ifade etmektedir. Sonuç olarak, Fast R-CNN ile VGG16 ağı R-CNN'den 9 kat daha hızlı eğitiliyor, test zamanında 213 kat daha hızlı ve PASCAL VOC 2012'den daha yüksek bir mAP elde ediliyor. SPPnet ile karşılaştırıldığında, Fast R-CNN VGG16'yı 3 kat daha hızlı eğitiliyor, 10 kat daha hızlı test ediyor ve daha doğru sonuçlar üretiyor.

“Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in neural information processing systems” adlı çalışmalarında Ren vd., (2015);

Faster R-CNN'i, gerçek zamanlı nesne algılama için yeni ve etkili bir yöntem olarak sunmaktadırlar. Geleneksel nesne algılama sistemleri, nesne konumlarını tahmin etmek için bölge önerme algoritmalarına dayanır ve bu da yavaştır. Faster R-CNN, algılama ağıyla tam görüntü evrişimsel özelliklerini paylaşarak neredeyse maliyetsiz bölge önerileri sağlayan Bölge Önerme Ağı (RPN) adı verilen yeni bir yöntem kullanır. RPN, her konumda nesne sınırlarını ve nesne olasılık puanlarını eşzamanlı olarak tahmin eden tamamen evrişimli bir ağıdır. Faster R-CNN, evrişimsel özelliklerini paylaşarak RPN'yi ve Fast R-CNN'yi tek bir ağa entegre ederek, algılama ağının bakması gereken bölgeye yönlendirir. Bu sistem, bir GPU üzerinde 5fps'lik bir kare hızına sahipken, PASCAL VOC 2007, 2012 ve MS COCO veri kümeleri üzerinde en son nesne algılama doğruluğunu elde etmektedir. Deneyler, Faster R-CNN'nin Selective Search ve EdgeBoxes gibi geleneksel bölge önerme yöntemlerinden daha yüksek doğruluk elde ettiğini ve büyük ölçekli veri kümeleriyle daha da iyi performans gösterdiğini göstermiştir. Faster R-CNN, gerçek zamanlı nesne algılaması için umut vadediyor ve uygulamalı nesne algılama için etkili bir çözüm haline geliyor.

YOLO algoritmasının ilk versiyonunu ve çalışma prensiplerinin ele alındığı “You only look once: Unified, real-time object detection” başlıklı makalelerinde Redmon vd., (2016); YOLO'nun, resimleri tek bir geçişte işleyerek gerçek zamanlı nesne algılaması gerçekleştiren yeni bir yöntem sunduğunu ifade etmektedirler. Geleneksel nesne algılama sistemleri, önce nesneleri sınıflandırmak için eğitilmiş sınıflandırıcıları kullanırken YOLO, görüntüleri tek bir regresyon problemi olarak ele almaktadır. Tek bir sinir ağı, resimlerden sınırlayıcı kutu koordinatlarını ve ilgili sınıf olasılıklarını aynı anda tahmin etmektedir. Böylece tek bir ağ mimarisi, algılama performansını doğrudan optimize etmeyi mümkün kılarak geleneksel yöntemlere göre önemli ölçüde daha hızlı ve daha etkili hale gelmektedir. YOLO, nesneleri küresel olarak değerlendirmekte, daha geniş bağlamsal bilgilere erişerek daha doğru tahminlerin yapılmasına imkân tanımaktadır. Çalışmada YOLO algoritmasının, çeşitli alanlara da genelleştirilebileceğine dikkat çekilmektedir. Bu, daha genel temsiller öğrenmesinden ve algılama görevine uyum sağlamak üzere özel olarak tasarlanan özellikler yerine, görüntüdeki tüm bilgileri kullanan bir ağ kullanmasından kaynaklanmaktadır. YOLO, saniyede 45 kare işleyebilme ve diğer gerçek zamanlı dedektörlerden iki kat daha yüksek bir ortalama hassasiyet (mAP) elde etme gibi mükemmel bir hız sergilemektedir. Daha küçük bir versiyon olan Fast YOLO ise saniyede 155 kare işleyerek, onu mevcut en hızlı nesne dedektörlerinden biri yapmaktadır. YOLO, küçük nesnelerin doğru yerini belirlemede zorluk yaşasa da arka plan gürültüsünde yanlış pozitifleri tahmin etme

konusunda daha az hata yapmaktadır. YOLO'nun sunduğu hız, doğruluk ve genelleme yeteneği, onu sürücüsüz araçlar, yardımcı cihazlar ve robotik sistemler gibi çeşitli uygulamalarda ideal bir seçim haline getirmektedir.

“A Review of Yolo algorithm developments” başlıklı makalelerinde, Jiang vd., (2022); You Only Look Once (YOLO) algoritmasının ve gelişmiş sürümlerinin bir incelemesini sunmaktadır. YOLO algoritmasının evrimsel gelişimini, hedef tespiti ve özellik seçimi yöntemlerini özetlemekte ve finansal ve diğer alanlarda hedeflenen nense tespiti ve özellik çıkarma için literatür desteği sağlamaktadır. Bu amaçla makale, YOLO, YOLO V2, YOLO V3, YOLO V4 ve YOLO V5 olmak üzere beş temel YOLO versiyonunu ele almakta ve bunların arasındaki temel teknik farklılıkları ve benzerlikleri analiz etmektedir. YOLO algoritmasının temel unsurunun, modelin küçük boyutu ve hızlı hesaplama yeteneği olduğu ifade edilmektedir. YOLO, yapay sinir ağı aracılığıyla doğrudan sınırlayıcı kutuların konumunu ve kategorisini çıkarabilmektedir. YOLO V2'ye, doğruluğu ve hızını artırmak için birkaç iyileştirme eklenmiştir. YOLO V3, çok ölçekli özellik kullanımı ve temel ağ yapısının ayarlanmasıyla YOLO V2'ye göre daha gelişmiş özellikler taşımaktadır. YOLO V4, doğruluğu ve hızını artırmak için çeşitli yöntemlerden yararlanarak, bu alanda en iyi performans gösteren modellerden birini sunmuştur. YOLO V5, kullanımı kolay PyTorch çerçevesine sahip, çok hafif bir model boyutu sunmuş ve doğruluk açısından YOLO V4'e yakın performans sergilemiştir. Ayrıca makale, YOLO algoritmasının evrimi ve versiyonlarının temel özelliklerini ayrıntılı bir şekilde ele alarak araştırmacılar için değerli bir bilgi kaynağı sunmaktadır. Bununla birlikte, YOLO ve diğer nesne algılama literatürüne önemli katkılar sağlamaktadır.

Farabet vd., (2010) makalelerinde, büyük ölçekli evrişimli sinir ağlarını (ConvNets) ve son teknoloji ürünü çok katmanlı yapay görme sistemlerini uygulamak için ölçeklenebilir bir donanım mimarisi sunmaktadırlar. Sistem, mega piksel görüntülerin gerçek zamanlı olarak algılanması, tanınması ve segmentasyonunu gerçekleştirme amacıyla modüler bir görme motoru olarak hizmet vermekte ve tamamen dijital bir sistemdir. Makale, özel donanım uygulamalarında hızlanma gösteren, yazılım, FPGA ve ASIC uygulamaları arasında yapılan performans karşılaştırmasını sunmaktadır. Bu sistem, görüntü verilerinin konumunu kullanarak büyük görüntüleri işlemek için gereken parametre sayısını azaltan özel bir sinir ağı türü olan ConvNets'e dayanmaktadır. Makale, bir ConvNet işlemcisinin FPGA üzerindeki ilk uygulamasının bir Xilinx Virtex-4 SX35 FPGA kartı üzerinde geliştirildiğini ve FPGA'nın özel olarak tasarlanmış bir baskılı devre kartındaki harici QDR-SRAM belleğe

bağlandığını ifade etmektedir. Makale ayrıca, verileri işleme hızını artırmak için birden fazla paralel vektör işleme birimi ekleyerek ve tekil veri akışlarının işleme blokları içinde sorunsuz bir şekilde çalışmasına izin vererek, FPGA'nın mevcut bant genişliğini mümkün olan en iyi şekilde kullanmaması nedeniyle mimarisinde bazı sınırlamalar olduğundan, daha yüksek bir veri çıkış hızı için tasarlanan ikinci nesil mimariyi tanıtmaktadır. Bu yeni mimari, bir Kontrol Ünitesi (CPU), birden fazla paralel Akış Operatörü (ALU) ve bir Bellek Arayüzü Akış Motoru olmak üzere üç ana bileşenden meydana gelmektedir. Makale, sistemin performansını bir dizüstü bilgisayardaki CPU, bir FPGA uygulaması ve gelecekteki ASIC uygulamalarıyla karşılaştırarak değerlendirmektedir. FPGA verileri, 200 MHz'de ve 15 W'da çalışan bir Xilinx Virtex-4 SX35 üzerinde ölçülürken, ASIC verileri, Tezzaron 3D işleminden toplanan simülasyon verileridir. Sonuç olarak, ASIC sisteminin veya modern bir Virtex 6 FPGA'nın, 21x21'e kadar filtre boyutlarında ve 1 megapiksele yaklaşan görüntülerde ConvNet sistemini gerçek zamanlı olarak (> 30 fps) çalıştırabileceğini göstermektedir.

“Object detection networks on convolutional feature maps.” başlıklı çalışmalarında, Ren vd., (2016); nesne algılama sistemlerinde bölgesel sınıflandırıcıların önemini derinlemesine inceleyerek derin evrişimli sınıflandırıcıların, derin evrişimli özellik çıkarıcılar kadar hayati olduğunu vurgulamaktadırlar. Yazarlar, paylaşılan, bölgesel olmayan evrişimsel özellikleri kullanan bölgesel sınıflandırıcı ağları, "Evrişimsel özellik haritaları üzerindeki ağlar" (NoC'ler) adıyla deneysel olarak ele almaktadırlar. Bulgular, derin özellik haritalarına ek olarak, bölgesel derin ve evrişimli bir sınıflandırıcının nesne algılama işleminin doğruluğu için kritik olduğunu göstermektedir. Bununla birlikte, ResNet ve GoogLeNets gibi son teknoloji görüntü sınıflandırma modelleri, böyle bir bölgesel sınıflandırıcı olmadan doğrudan iyi bir algılama performansı sunmamaktadırlar. Deneyler, Faster R-CNN ve ResNet gibi etkili sistemlere rağmen, NoC'lerin tasarımının, ImageNet ve MS COCO 2015 yarışmalarında birincilik kazananlar için temel bir unsur olduğunu kanıtlamaktadır. Araştırmacılar, özellikle, ResNet ile naif bir Faster R-CNN versiyonunun, sık ve evrişimli olmayan bölgesel sınıflandırıcısından dolayı düşük doğruluk (%21,3 AP) sergilediğini göstermiştir. Bunun aksine, derin ve evrişimli bir NoC'un, Faster R-CNN + ResNet'in doğru nesne algılama gerçekleştirmesi için çok önemli olduğu ifade edilmektedir. Çalışma, Pascal VOC 2007 test setinde %71,6 mAP ve MS COCO doğrulama setinde %27,2 AP gibi etkileyici sonuçlar elde etmiştir.

“Real-time flying object detection with YOLOv8.” Başlıklı çalışmalarında araştırmacılar;

transfer öğrenimi ve daha fazla araştırma için kullanılabilen gerçek zamanlı uçan nesne algılama için geliştirilmiş bir modelin yanı sıra uygulanmaya hazır rafine edilmiş bir modeli sunmaktadırlar. Araştırmacılar, modeli 40 farklı uçan nesne sınıfını içeren bir veri kümesi üzerinde eğiterek, modelin soyut özellik temsillerini çıkarmasını sağlamışlardır. Daha sonra bu öğrenilen parametrelerle, "gerçek dünya" ortamlarını (yani daha yüksek örtüşme sıklığı, küçük uzamsal boyutlar, rotasyonlar vb.) daha iyi temsil eden bir veri kümesi üzerinde transfer öğrenmesi gerçekleştirerek rafine edilmiş bir model oluşturmuşlardır. Uçan nesnelerin nesne tespiti, büyük değişken nesne uzamsal boyutları/en boy oranları, hız oranı, örtüşme ve kümelenmiş arka planlar nedeniyle zorlu görev olarak ifade edilmektedir. Sunulan zorluklardan bazılarını ele alırken aynı zamanda performansı en üst düzeye çıkarmak, çıkarım hızı ve mAP arasında en iyi dengeyi bulmak için çalışmanın yapıldığı zaman var olan en iyi tek atışlı dedektör (single-shot detector) olan YOLOv8'i kullanmışlardır. Son geliştirilmiş model, 1.080p videolarda 0,685'lik bir mAP50-95 ve ortalama 50 fps çıkarım hızı elde etmiştir. Son rafine edilmiş model, bu çıkarım hızını korurken 0,835'lik iyileştirilmiş bir mAP50-95 elde etmiştir. Çalışma, YOLOv8 mimarisini ayrıntılı olarak açıklamakta ve uçan nesne algılamada karşılaşılan zorlukları ele almak için modelin performansını değerlendirmektedir. Özellikle küçük nesneleri, karışık arka planları ve farklı uçan nesne türlerini ayırt etme yeteneğine odaklanmaktadır. Modelin performansını vurgulamak için, modelin hem geliştirilmiş hem de rafine edilmiş versiyonları kullanılarak, özellikle küçük, pikseli nesneler, karışık arka planlar ve farklı uçan nesne türleri arasındaki ayrım gibi zorlukları içeren örnek görüntüler sunulmuştur (Reis vd., 2023).

Wang vd., (2024), "Learning What You Want to Learn Using Programmable Gradient Information" başlıklı çalışmalarında, derin sinir ağlarında bilgi kaybına neden olan bilgi darboğazı ve tersine çevrilebilir fonksiyonlar olgusunu ele almışlardır. Çalışma, derin ağların çoklu hedeflere ulaşmak için gereken çeşitli değişikliklerle başa çıkmak için Programlanabilir Gradyan Bilgisi (PGI) kavramını önermektedir. PGI, hedef görev için tam giriş bilgisini sağlayarak objektif fonksiyonun hesaplanması için güvenilir gradyan bilgisi elde edilebilmesini sağlamıştır. Bununla birlikte, yeni bir hafif ağ mimarisi olan Geliştirilmiş Etkin Katman Toplama Ağı (GELAN) tasarlanmıştır. GELAN'ın mimarisi, PGI'nin hafif modellerde üstün sonuçlar elde ettiğini doğrulamaktadır. Önerilen GELAN ve PGI, MS COCO veri kümesi tabanlı nesne algılama üzerinde denemeleriyle doğrulanmıştır. Sonuç olarak, çalışma GELAN'ın sadece geleneksel konvolüsyon operatörlerini kullanarak, derinlik bazlı konvolüsyona dayalı olarak geliştirilen en gelişmiş yöntemlerden daha iyi

parametre kullanımı sağladığını göstermiştir. PGI, hafif ve derin modellerde doğruluk açısından önemli gelişmeler sağlayarak, her iki model türünde de uygulanabilirliğini göstermiştir. Sıfırdan eğitilmiş modellerde tam bilgi elde etmek için kullanılan PGI, bu modellerin büyük veri kümeleri kullanılarak önceden eğitilmiş en gelişmiş modellerden daha iyi sonuçlar elde etmesini sağlamıştır. PGI ve GELAN'ı birleştiren YOLOv9, MS COCO veri kümesinde YOLOv8'e kıyasla %49 daha az parametre ve %43 daha az hesaplama ile dikkate değer bir verimlilik sergilerken %0,4 ila %0,6'lık bir AP iyileştirmesi sağlıyor. Bu, YOLOv9'un hesaplama karmaşıklığı ile doğruluk arasında olağanüstü bir denge kurduğunu göstermektedir.

Derin öğrenme tekniklerinin tıbbi görüntüleme alanında kullanımının hızla artması, büyük ve yüksek kaliteli veri kümelerinin geliştirilmesine duyulan ihtiyacın artmasına sebep olmaktadır. “Mura: Large dataset for abnormality detection in musculoskeletal radiographs” başlıklı çalışmalarında Rajpurkar vd., (2017); MURA adlı geniş bir kas-iskelet radyografisi veri kümesi oluşturulmuşlardır. MURA, radyologlar tarafından normal veya anormal olarak elle etiketlenmiş 14.863 çalışmadan elde edilmiş 40.561 görüntü içermektedir. Veri setinin oluşturulmasında modellerin sağlam bir şekilde değerlendirilmesi ve radyolog performansının tahmin edilmesi amacıyla 207 kas-iskelet çalışmasından oluşan bir test kümesi üzerinde altı sertifikalı Stanford radyologundan ek etiketler toplanmıştır. Bu test kümesinde, üç radyolog grubunun çoğunluk oyu altın standart olarak kullanılmıştır. Çalışmada, anormallikleri tespit etmek ve konumlandırmak için 169 katmanlı bir DenseNet temel modeli eğitilmiştir. Model, 0,815 duyarlılık ve 0,887 özgüllük çalışma noktasına sahip 0,929 AUROC değeri elde etmiştir. Model performansı, Cohen'in kappa istatistiği kullanılarak radyologların performansı ile karşılaştırılmıştır. Sonuç olarak, model performansının parmak ve bilek çalışmalarında anormallikleri tespit etmede, en iyi radyolog performansı ile karşılaştırılabilir düzeyde başarılı olduğunu göstermiştir. Ancak, dirsek, önkol, el, üst kol ve omuz çalışmalarında anormallikleri tespit etmede en iyi radyolog performansından daha düşük bir performans sergilediği belirlenmiştir. Bu bulgular, kas-iskelet radyografisinde anormallik tespitinin derin öğrenme modelleri için hala zorlu bir görev olduğunu göstermektedir. MURA veri kümesi, araştırmacılar için ücretsiz olarak kullanıma sunulmuştur.

İskelet ve kas sisteminde meydana gelen anomali tespiti için evrişimli sinir ağlarının kullanılması konulu “Abnormality detection in musculoskeletal radiographs with convolutional neural networks (ensembles) and performance optimization” adlı çalışmada

Banga, ve Waiganjo, (2019); kas ve iskelet sistemi röntgen görüntülerinde anomali tespiti yaparken sağlanan performansı iyileştirebilmek için evrişimli sinir ağları kullanarak bir model geliştirilmesi amaçlanmıştır. Araştırmacılar, Standford Üniversitesi Hastanesi tarafından yapılan 14.863 vaka çalışması ve 40.561 görüntüden oluşan bir kas ve iskelet röntgen veri seti olan MURA veri setini kullanmışlardır. Araştırmada dört farklı CNN mimarisi kullanılarak modeller eğitilmiştir. Bu mimariler Xception, MobileNet, NASNetMobile ve DenseNet201'dir. Daha sonra bu modeller farklı sıralamalarla kullanılarak bir araya getirilmiş ve sonuçları değerlendirilmiştir. "Ensemble200" adını verdikleri kombinasyon en iyi performansı sergileyen model olmuştur. Bu model; DenseNet201, MobileNet ve NASNetMobile modelleri birleştirilerek oluşturulmuştur. Araştırmanın sonuçları, ensemble200 adlı modelin, MURA veri setinde yalnızca bir tane CNN modelinden çok daha iyi performans ortaya koyduğunu göstermiştir. Bu modelin Cohen Kappa puanı 0,66. Tek başına bir MobileNet modelinin Kappa puanı ise 0.41'in biraz üzerinde bir puan elde etmektedir. Bununla birlikte araştırmacılar, modelin daha iyi bir şekilde yorumlanabilmesi için dereceli sınıf aktivasyon haritaları (Grad CAM) kullanmıştır. Bu da modelin ürettiği tahminlerini destekleyen daha iyi görsel açıklamalar sağlamıştır. Bu çalışmanın sonunda elde edilen bulguları, ensembled yani birlikte kullanılan CNN algoritmalarının, kas-iskelet röntgenlerindeki anormalileri daha iyi bir performansla tespit ettiğini ve daha iyi genellemelere ulaşılabileceğini göstermektedir.

“Computer Assisted Solution for bone Anomaly detection in X-Ray images produced no results.” adındaki çalışmada Qamar, Jawaid ve Memon, (2021); kolu oluşturan kemikler üzerindeki anomalileri bilgisayar destekli bir teknikle otomatik olarak tespit etmek için röntgen görüntülerinden faydalanarak bir çözüm üretmeye çalışmışlardır. Geliştirdikleri sistem; röntgen görüntülerinin ön işlemden geçirilmesi, segmentasyon, yoğunluk tabanlı tespit, özellik çıkarma ve sınıflandırma olmak üzere beş ana aşamadan meydana gelmektedir. MURA veri setinin kullanıldığı çalışmada ilk olarak ön işleme aşamasında, veri setindeki kol röntgen görüntüleri üzerinde, gürültü ve kontrast geliştirme işlemleri yapılmıştır. İkinci aşamada, Canny Edge Detection yöntemi kullanılarak, kol kemiklerinin ilgili bölgeleri tespit edilerek segmentasyon işlemi yapılmıştır. Üçüncü aşamada, anomalileri belirlemek için yoğunluk tabanlı bir tespit yöntemi kullanılarak yoğunluk değişiklikleri analiz edilmiştir. Dördüncü aşamada, dairesellik, kemikler arası mesafe ve maksimum yoğunluk gibi özellikleri çıkarmak için bir dizi özellik çıkarma tekniği uygulanmaktadır. Son olarak da elde edilen özellik vektörleri, kol röntgen görüntülerini normal veya anormal

olarak sınıflandırmak için destek vektör makinesi (SVM) sınıflandırma algoritması kullanılarak eğitilmiştir. Sistem, 90 görüntü kullanılarak %93,5 doğruluk oranı ile eğitilmiş ve daha sonra görülmemiş 40 görüntüde %85,7 doğruluk oranı elde edilmiştir. Sonuç olarak, önerilen sistemin kol röntgenlerindeki kemik anomalilerini tespit etmede etkili olduğu tespit edilmiştir.

Gelecekte, farklı özellik vektörleri çıkarılarak ve farklı sınıflandırıcılar kullanılarak sistemin doğruluk oranını iyileştirmenin mümkün olacağı, daha geniş ve daha çeşitli bir röntgen görüntü veri seti kullanılarak sistemin performansının artırılacağı ifade edilmiştir.

“A systematic literature review on deep learning approaches for pneumonia detection using chest X-ray images” başlığıyla yayınladıkları sistematik literatür taraması çalışmasında, Sharma, ve Guleria, (2024), göğüs röntgeni görüntülerini kullanarak pnömoni tespiti için derin öğrenme yaklaşımlarını incelemişlerdir. Çalışma, pnömoninin erken teşhisinin önemini vurgularken insan sağlığı üzerindeki etkilerini de ele almaktadır. Çalışmada derin öğrenmenin pnömoni teşhisinde kullanılmasının, otomatik özellik çıkarımı gibi avantajları vurgulanmaktadır. Taramada, derin öğrenme tabanlı pnömoni tespit modelleriyle ilgili mevcut literatürü incelemek için PRISMA (Sistematik Meta-Analiz için Tercih Edilen Raporlama Öğeleri) yöntemi kullanılmıştır.

Çalışma, Evrişimli Sinir Ağları (CNN) tabanlı modeller, önceden eğitilmiş modeller ve topluluk modelleri olmak üzere üç kategorideki çeşitli modelleri analiz etmektedir. Her kategori için, çeşitli modellerin mimarileri, çalışma prensipleri ve performans göstergeleri ayrıntılı olarak incelenmektedir. Çalışma, her model kategorisi için kullanılan veri kümeleri, artırma yöntemleri ve hiperparametre ayarları gibi önemli faktörleri ele almaktadır. Çalışmada ayrıca, mevcut modellerdeki araştırma boşlukları ve sınırlamalarının ele alınması için olası çözümler tartışılmaktadır. Elde edilen bulgular, topluluk modellerinin; CNN tabanlı modeller ve önceden eğitilmiş modellerle karşılaştırıldığında pnömoni tespiti için daha iyi performans gösterdiğini ortaya koymaktadır. Topluluk modelleri, veri kümelerinin büyüklüğüne ve hiperparametre ayarlarına duyarlı oldukları için, modellerin doğru eğitilmesi için büyük veri kümeleri ve kapsamlı hiperparametre optimizasyonu gerekmektedir. Bulgular incelendiğinde Xception, RetinaNet, GoogleNet ve MobileNet gibi daha fazla önceden eğitilmiş modelin araştırılması gerektiği anlaşılmaktadır.

Verma vd., (2024); çalışmada, omuz ve dirsek arasında bulunan üst kol kemiği olan humerus kemiğinde kırık tespiti için YOLOv4 algoritmasını kullanan otomatik bir sistem geliştirmeyi amaçlamaktadırlar. Geleneksel yöntemler, genellikle tıbbi görüntülerin gözle incelemesine

dayanmaktadır ve bu da potansiyel gecikmelere ve doğruluk sorunlarına yol açabilmektedir. Çalışmada, otomatik kırık tespiti için YOLO algoritmasının uygulanması araştırılmış ve YOLOv4'ün humerus kırıklarını tespit etmek için verimlilik ve doğruluk açısından değerlendirilmesi hedeflenmiştir.

Çalışmada, YOLO algoritmasının eğitilmesi için normal ve kırık humerus kemiği görüntülerini içeren kapsamlı bir veri kümesi oluşturulmuştur. Veri ön işleme teknikleri ve model optimizasyon stratejileri kullanılarak, YOLO algoritmasının radyografik görüntülerde humerus kırıklarını doğru bir şekilde tanımlama yeteneği artırılmıştır. YOLOv4 modeli, F1 skoru, ortalama ortalama kesinlik (mAP), geri çağırma ve kesinlik dahil olmak üzere yaygın ölçütler kullanılarak değerlendirilmiştir.

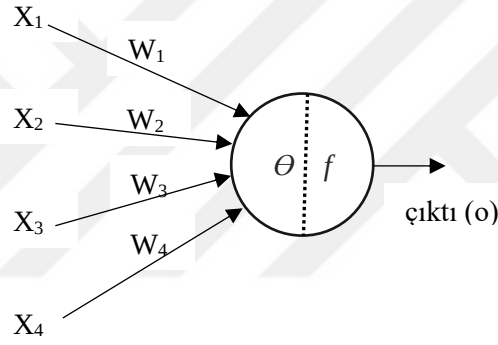
Sonuç olarak, YOLOv4 modelinin humerus kırıklarını tespit etmede yüksek hassasiyet ve doğruluk gösterdiği ortaya konmuştur. Bu, modelin radyoloji ve ortopedi alanlarında tanısal verimliliği artırma ve sağlık hizmetleri sunumunu iyileştirme potansiyeline sahip olduğunu göstermektedir. Ancak, araştırma, YOLOv4 modelinin kırık alanlarını belirlemek için kısıtlı bir veri kümesiyle eğitildiğini, modelin gerçek dünya ortamlarında kırık türlerini ve şiddetini doğru bir şekilde sınıflandırabilme yeteneğinin sınırlı olduğunu göstermiştir.

İKİNCİ BÖLÜM

DERİN ÖĞRENME ALGORİTMALARINA GENEL BİR BAKIŞ

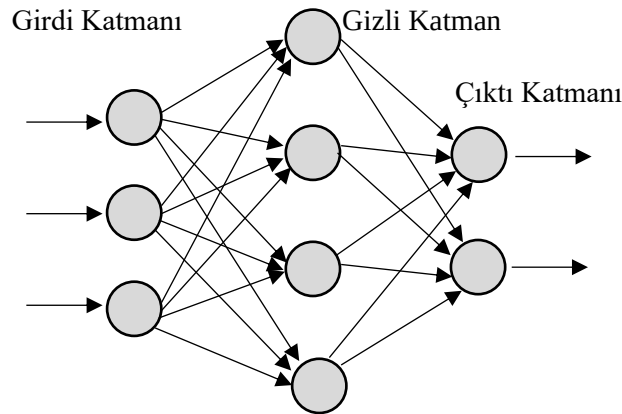
Bu bölümde, yapay sinir ağları, evrişimli sinir ağları, R-CNN ve YOLO algoritmaları ele alınacak ve kullanılan kavramlar açıklanacaktır.

2.1. Yapay Sinir Ağları (ANN – Artificial Neural Networks)



Şekil 1.Yapay Sinir Ağı

Yapay sinir ağları, aynı anda çalışan birbirine bağlı birçok birimden oluşan, biyolojik sinir ağlarının çevrelerindeki dünya ile etkileşime girme şeklini taklit eden, karmaşık sistemlerdir. Sinir ağlarını oluşturan birimler genellikle uyum sağlama yeteneğine sahiptir ve katmanlar halinde organize edilirler.



Şekil 2. Çok Katmanlı Yapay Sinir Ağı

Bu sistemleri tanımlamak için kullanılan tek terim "sinir ağları" terimi değildir. Literatürde karşılaşılabileceğiniz diğer bazı terimler şunlardır:

- Nörobilgisayar
- Bağlantıcı modeller
- Paralel dağıtılmış işleme (O'Shea ve Nash, 2015).

2.1.1. Yapay Sinir Ağlarının Faydaları

Yapay sinir ağlarının doğrusal olmayan verilerin işlenmesine imkân tanımak, güvenilir giriş-çıkış haritaları oluşturmak, hatalara karşı toleranslı olmak, büyük veri kümeleriyle etkin bir şekilde çalışmak gibi faydaları vardır. Bu faydalar şu şekilde sıralanabilir:

Doğrusal olmama: Sinir ağları verilerdeki doğrusal olmayan ilişkileri işleyip modelleyebilmektedir, bu da onları doğrusal modellerin başarısız olabileceği karmaşık problemler için uygun hale getirmektedir.

Giriş-Çıkış Haritalaması: Giriş verilerini öğrenip uygun çıkış verileriyle eşleyerek model tanıma gibi görevlerde etkili hale gelmektedir.

Uyarlanabilirlik: Sinir ağları, değişen girdi modellerine yanıt olarak dâhili parametrelerini uyarlayabilmekte böylece zaman içinde öğrenmelerine ve gelişmelerine olanak tanımaktadır.

Kanıtsal Yanıt: Kanıtsal yanıt sağlamaları, çıktılarının güven düzeyini yansıttığı ve sonuçların güvenilirliğine ilişkin değerli bilgiler sunduğu anlamına gelmektedir.

Bağlamsal Bilgi: Sinir ağları bağlamsal bilgileri dikkate almakta ve ilgili çevre verilerini birleştirerek daha bilinçli kararlar alınmasını sağlamaktadırlar.

Hata Toleransı: Dağıtılmış doğaları nedeniyle hata toleransı sergileyebilmeleri, bileşenlerinde bir dereceye kadar hasar veya hata olsa bile yine de çalışabilecekleri anlamına gelmektedir.

VLSI uygulanabilirliği: Sinir ağları, verimli donanım çözümlerinin geliştirilmesini kolaylaştıran Çok Büyük Ölçekli Entegrasyon (VLSI) teknolojisiyle kullanılabilir.

Analiz ve Tasarımın Tekdüzeligi: Çeşitli sorunların çözümü için tutarlı bir çerçeve sağlayarak analiz ve tasarım için tek tip bir yaklaşım sunmaktadırlar.

Nörobiyolojik Analoji: Sinir ağları insan beyninden ilham alarak biyolojik sinir sistemlerinin işleyişine dair potansiyel bilgiler sunmaktadırlar.

2.1.2. İnsan Beyni ve Sinir Ağları Arasındaki İlişki

Sinir ağlarının geliştirilmesinde, insan beyninin yapısı ve işlevinden önemli ölçüde ilham alınmıştır. Beyin, sinapslarla birbirine bağlanan muazzam bir nöron ağından oluşmaktadır. Bu nöronlar bilgiyi işleyip ileterek düşüncelerimizin, eylemlerimizin ve algılarımızın temelini oluştururlar.

Benzer şekilde yapay sinir ağları da katmanlar halinde organize edilmiş, birbirine bağlı düğümlerden veya yapay nöronlardan oluşmuştur. Bu düğümler, biyolojik nöronların işlevselliğini bir dereceye kadar taklit ederek sinyalleri işler ve iletirler.

Ancak mevcut sinir ağları beynin büyük ölçüde basitleştirilmiş modelleridirler. Nöral hesaplamanın bazı temel ilkelerini yakalıyorlar ancak biyolojik muadillerinin karmaşıklığı ve karmaşıklığından yoksundurlar.

Bu sınırlamalara rağmen, beyin çalışmaları sinir ağı araştırmalarındaki gelişmelere bilgi sağlamaya ve ilham vermeye devam etmektedir. Beynin mimarisi ve mekanizmaları hakkında daha derin bir anlayış kazandıkça daha güçlü ve etkili yapay sinir ağları geliştirebilmemiz beklenmektedir (Haykin, 1999).

2.2. Evrişimli Sinir Ağları (CNN – Convolutional Neural Networks)

Evrişimli Sinir Ağları (CNN'ler), öncelikle görüntü tanıma ve ilgili görevler için tasarlanmış özel bir Yapay Sinir Ağı (ANN) türüdür. Görüntü verilerinden barındırdıkları özellikleri otomatik olarak öğrenme ve çıkarma konusunda uzmanlaştıklarından görüntü sınıflandırma, nesne algılama ve görüntü segmentasyonu gibi görevlerde son derece etkilidirler.

CNN'ler, görüntülerin mekânsal yapısından yararlanarak anlamlı özellikler çıkarmak için özel olarak tasarlanmıştır. Geleneksel yapay sinir ağlarından farklı olarak CNN'ler nöronları, giriş görüntülerinin yapısını yansıtacak şekilde üç boyutta (yükseklik, genişlik ve derinlik) düzenlenirler. Bir CNN katmanındaki nöronlar, önceki katmanın yalnızca alıcı alan olarak bilinen küçük bir bölgesine bağlanırlar. Bu, bağlantı ve parametre sayısını azaltarak ağı daha verimli hale getirirler. CNN'ler, aynı ağırlıkların farklı giriş bölgelerine uygulandığı parametre paylaşımını kullanırlar. Bu varsayım, öğrenilecek parametre sayısını önemli

ölçüde azaltarak, bir alanda faydalı olan özelliklerin diğer alanlarda da faydalı olabileceği fikrinden yararlanmaktadır (Wu, 2017).

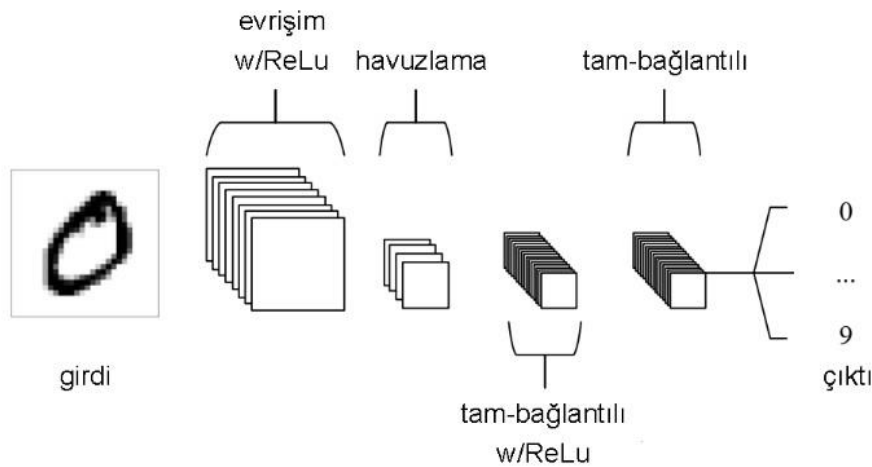
2.2.1. Üç Katman Türü

CNN'ler temel olarak üç tür katmandan oluşmaktadır:

Evrşimsel Katmanlar (Convolutional Layers): Bu katmanlar, öğrenilebilir filtreler (çekirdekler) kullanarak giriş görüntüsünden özellikler çıkarırken filtreler görüntü boyunca evrilip filtre ağırlıkları ile giriş arasındaki skaler çarpım hesaplanarak ve aktivasyon haritaları oluşturulmaktadır. Her filtre belirli özellikleri tespit etmeyi öğrenmekte ve aktivasyon haritaları, bu özelliklerin giriş görüntüsündeki varlığını ve konumunu yakalamaktadır.

Havuzlama Katmanları (Pooling Layers): Bu katmanlar, aktivasyon haritalarının uzamsal boyutlarını alt örnekleyerek parametre sayısını ve hesaplama karmaşıklığını azaltır. Genellikle aktivasyon haritasının küçük bir bölgesindeki en büyük değerin tutulduğu maksimum havuzlama kullanılır.

Tamamen Bağlantılı Katmanlar (Fully Connected Layers): Geleneksel yapay sinir ağlarına benzer şekilde bu katmanlar, bir önceki katmandaki tüm nöronlara bağlı nöronlardan oluşmaktadırlar. Çıkarılan özellikleri yorumlayarak sınıflandırma görevleri için sınıf puanları gibi nihai çıktıyı üretmek için üst düzey akıl yürütme gerçekleştirmektedirler (O'Shea ve Nash, 2015).



Şekil 3.Sadece Beş Katmandan Oluşan Basit Bir CNN Mimarisi

2.2.2. Bağlantı ve Optimizasyon

Evrişimli sinir ağlarında; parametreler paylaşılarak, katmanlar arasındaki bağlantılar ve hiperparametreler optimize edilerek daha düşük bir işlem gücüne ihtiyaç duymakla birlikte daha iyi sonuçlar üretebilmektedirler.

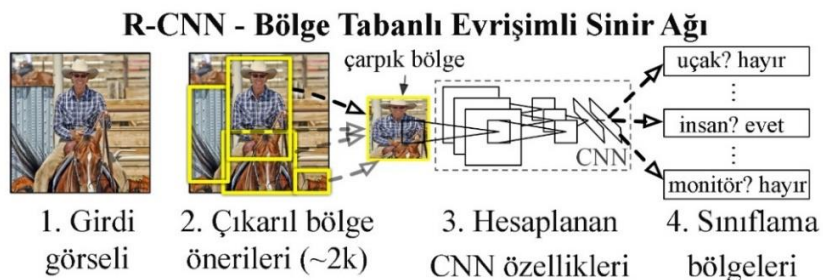
Yerel Bağlantı: Her nöronun önceki katmandaki tüm nöronlara bağlandığı geleneksel yapay sinir ağlarının aksine, CNN'ler yerel bağlantı kullanmaktadırlar. Evrişimsel katmandaki tüm nöronlar girdinin yalnızca alıcı alan adı verilen küçük bir bölgesine bağlanarak parametre sayısını önemli ölçüde azaltılarak ağı görüntüdeki yerel kalıpları öğrenmesine yardımcı olmaktadır.

Hiperparametreler: Evrişimli katmanların mimarisi, derinlik (filtre sayısı), adım (filtre hareketinin adım boyutu) ve sıfır doldurma (girdi kenarlığının etrafına sıfırlar eklenmesi) gibi hiperparametreler kullanılarak özelleştirilmektedir. Bu parametreler çıktı özellik haritalarının boyutunu ve boyutluluğunu kontrol etmektedir.

Parametre Paylaşımı: CNN'ler, giriş görüntüsündeki farklı mekânsal konumlarda aynı filtre ağırlıklarının kullanıldığı parametre paylaşımından yararlanmaktadır. Bu varsayım öğrenilecek parametre sayısını azaltmakta ve ağı görüntünün herhangi bir yerinde aynı özellikleri tespit etmesine olanak tanımaktadır (O'Shea ve Nash, 2015).

2.3. R-CNN - Bölge Tabanlı Evrişimli Sinir Ağı

Bölge tabanlı evrişimli sinir ağlarının kısaltması olan R-CNN, evrişimli sinir ağlarının gücünü etkili bir şekilde entegre ederek alanda devrim yaratan bir nesne algılama sistemidir. R-CNN'den önce, nesne tespiti ağırlıklı olarak el yapımı özelliklere ve daha basit modellere dayanmaktaydı ve çoğu zaman doğruluk ve verimlilik konusunda zorluk çekmekteydi. R-CNN, CNN'lerin özellik çıkarma yeteneklerinden yararlanan yeni bir yaklaşım sunarak bu kısıtlamaları ortadan kaldırmaktadır.



Şekil 4. R-CNN: Bölge Tabanlı Evrişimli Sinir Ağı

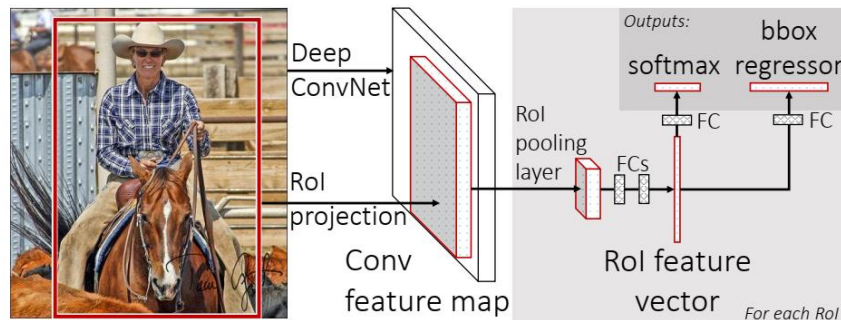
Şekil 4'te akış diyagramı verilen R-CNN, geleneksel kayan pencere tekniklerini kullanmak yerine, bir görüntüdeki potansiyel nesne konumlarını tanımlamak için bölge önerileri oluşturmaktadır. Seçici arama benzeri bir algoritma tarafından oluşturulan bu öneriler, nesneler için potansiyel sınırlayıcı kutular görevi görmektedir. Daha sonra her bölge önerisi sabit bir boyuta dönüştürülerek önceden eğitilmiş bir CNN'den geçirilmektedir. Başlangıçta büyük bir görüntü sınıflandırma veri kümesi (örneğin, ImageNet) üzerinde eğitilen bu CNN'ler, her bölgedeki görsel içeriğin zengin bir temsilini üreten bir özellik çıkarıcı görevi görmektedir.

Çıkarılan özellikler daha sonra sınıflandırma için sınıfa özgü doğrusal SVM'lere beslenerek her bölgenin nesne kategorisini belirlemekte ve algılama sürecini etkili bir şekilde tamamlamaktadırlar. Özellik çıkarımı için bölge önerilerini CNN'lerle birleştiren R-CNN, PASCAL VOC gibi kıyaslama veri kümelerinde önceki en gelişmiş yöntemlere kıyasla nesne algılama doğruluğunda önemli bir sıçrama elde etmiştir.

R-CNN'in başarısı, nesne tespiti için CNN'lerin gücünden yararlanma yeteneğinde yatmaktadır. CNN'ler, ham görüntü verilerinden hiyerarşik özellikleri öğrenmede, geleneksel el işi özelliklerin sıklıkla gözden kaçırıldığı karmaşık ayrıntıları ve karmaşık modelleri yakalamada oldukça başarılı sonuçlar vermektedirler. R-CNN'in bu kadar yüksek düzeyde doğruluk elde etmesini sağlayan şey; zengin, ayırt edici özellikleri doğrudan görüntülerden öğrenme yeteneğidir. R-CNN üzerinde daha sonra yinelemeler ve iyileştirmeler geliştirilmiş olsa da özellik çıkarımı için CNN'lerden yararlanmaya ilişkin temel konsept, modern nesne algılama sistemlerinde temel bir yapı taşı olmaya devam etmektedir (Girshick vd., 2014).

2.4. Fast R-CNN

Şekil 5'teki akış diyagramı Fast R-CNN mimarisini temsil etmektedir.



Şekil 5. Fast R- CNN Mimarisi

Fast R-CNN, adından da anlaşılacağı gibi, R-CNN (Bölge Tabanlı Evrişimli Sinir Ağı) nesne algılama algoritmasının geliştirilmiş bir versiyonudur. Her iki yöntem de bir görüntü içindeki nesne önerilerini sınıflandırmak için derin evrişimli sinir ağlarını kullanırken Fast R-CNN, hızı ve doğruluğu artıran önemli değişiklikler sunmaktadır (Girshick, 2015).

2.4.1. Neden Fast R-CNN, R-CNN'den daha hızlıdır?

Fast R-CNN; özellik paylaşımı, RoI (Region of Interest) Havuzlama katmanının kullanımı, tek aşamalı eğitim ve tüm katmanlara ince ayar yapma şansı tanıdığı için R-CNN'den daha hızlıdır.

Özellik Paylaşımı: Her nesne önerisi için bir ConvNet ileri geçişi gerçekleştiren R-CNN'nin aksine, Fast R-CNN tüm görüntüyü yalnızca bir kez işleyerek hesaplamaları yapmakta ve bir evrişimli özellik haritası oluşturmaktadır.

RoI Havuzlama: Fast R-CNN, verimli bir şekilde her bir öneri için paylaşılan özellik haritasından sabit uzunlukta bir özellik vektörü çıkaran bir RoI havuzlama katmanı kullanmaktadır. Bu, R-CNN'de olduğu gibi bölgelerin eğilmesine gerek olmaksızın, daha hızlı bir işlem yapılmasına imkân tanırken mekânsal bilgiyi de korumaktadır.

Tek Aşamalı Eğitim: Fast R-CNN, çoklu görev kaybı kullanarak sınıflandırma ve sınırlayıcı kutu regresyonunu tek bir aşamada birleştirerek eğitim sürecini kolaylaştırmaktadır. Böylece R-CNN'nin gerektirdiği gibi ayrı SVM ve sınırlayıcı kutu regresör eğitim aşamalarına duymadan daha hızlı eğitim ve doğruluk sağlamaktadır.

Tüm Katmanlara İnce Ayar Yapma: Fast R-CNN, eğitim süreci sırasında evrişimli katmanlar da dahil olmak üzere tüm ağ katmanlarına ince ayar yapılmasına olanak tanıyarak evrişimli katmanların güncellenmesinde sınırlamalara sahip olan SPPnet ve R-CNN'in aksine daha iyi özellik temsillerinin elde edilmesini ve doğruluğun artmasını sağlamaktadır (Girshick vd., 2014).

2.5. Faster R-CNN

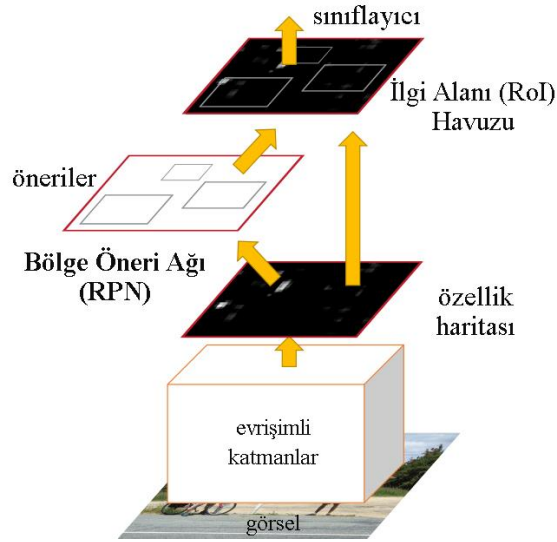
Faster R-CNN, öncülü olan Fast R-CNN gibi yöntemlerde bölge önerisi hesaplamasında yaşanan darboğazı ele alan bir nesne algılama algoritmasıdır. Algılama ağı ile evrişimsel özellikleri paylaşan ve neredeyse maliyetsiz bölge önerileri sağlayan bir Bölge Öneri Ağı (RPN - Regional Proposal Network) sunmaktadır. Bölge öneri ağları, düzenli bir ızgaranın

her konumdaki nesne sınırlarını ve nesne tespit puanlarını eşzamanlı olarak tahmin eden, birleşik ağın dikkatini potansiyel nesne konumlarına etkili bir şekilde yönlendiren tamamen evrişimli bir ağıdır.

Faster R-CNN, bu verimliliği, bölge öneri ağları ile Fast R-CNN dedektörü arasındaki paylaşılan evrişimli katmanları kullanarak elde etmektedir. Bu tasarım, minimum ek maliyetle (örneğin görüntü başına 10 ms) bölge önerilerinin hesaplanmasına olanak tanımaktadır. Faster R-CNN, özellikleri paylaşarak ve öneri sayısını azaltarak, yüksek doğruluğu korurken gerçek zamanlıya yakın nesne algılama hızlarına ulaşarak çeşitli nesne algılama görevleri için pratik ve etkili bir çözüm haline gelmektedir (Ren vd., 2015).

2.5.1. Fast R-CNN ile Faster R-CNN arasındaki farklar

Fast R-CNN ve Faster R-CNN arasındaki temel fark, bölge önerileri oluşturma yaklaşımlarında yatmaktadır. Fast R-CNN, hesaplama açısından pahalı olabilen ve tespit sürecinde bir darboğaz oluşturabilen *seçici arama* gibi harici bölge öneri algoritmalarına dayanmaktadır. Buna karşılık ağı kendisine entegre olan bir Bölge Teklif Ağı (RPN) sunan Faster R-CNN, Fast R-CNN dedektörüyle evrişimli özellikleri paylaşmakta ve minimum ek hesaplama maliyetiyle bölge önerileri oluşturmalarını sağlamaktadır.



Şekil 6. Faster R-CNN

Şekil 6’da gösterildiği gibi bölge öneri ağlarının bu entegrasyonu, nesne algılama sürecini kolaylaştırarak Faster R-CNN’yi, özellikle test sırasında Fast R-CNN’den önemli ölçüde daha hızlı hale getirmektedir. Bununla birlikte bölge öneri ağları, verilerden bölgeler önermeyi öğrenerek farklı veri kümelerine uyum sağlamasına ve seçici arama gibi sabit, elle

hazırlanmış algoritmalarla kıyasla teklif kalitesinin potansiyel olarak iyileştirmesine imkân tanımaktadır (Ren vd., 2016).

2.6. YOLO (1. Versiyon)

YOLO (You Only Look Once), nesne tespiti için yenilikçi bir yaklaşım sunarak, görevi doğrudan görüntü piksellerinden sınırlayıcı kutuları ve sınıf olasılıklarını tahmin eden tek bir regresyon sorunu olarak ele almaktadır. Bölge önerilerine veya kaydırılabilir pencerelere dayanan geleneksel yöntemlerin aksine, YOLO, nesneler ve çevreleri hakkında bağlamsal bilgileri kodlayan tüm görüntüyü global olarak işleyen tek bir sinir ağı kullanmaktadır.

Bu birleşik mimari, algılama performansı için uçtan uca optimizasyon sağlayarak olağanüstü hız ve doğruluk elde etmektedir. YOLO, giriş görüntüsünü bir $S \times S$ ızgaraya bölmekte, her ızgara hücrelerini sınırları içinde merkezi bulunan nesneleri tespit etmek için kullanmaktadır. Her hücrede, güven puanları ve C koşullu sınıf olasılıklarıyla birlikte B sınırlayıcı kutuları öngörülmektedir. Tüm görüntüyü göz önünde bulundurarak, ağ aynı anda sınırlayıcı kutuları ve sınıf olasılıklarını tahmin edip sahnenin daha kapsamlı bir şekilde anlaşılmasını gerektirmekte ve bölgeye dayalı yöntemlere kıyasla arka plan hatalarını azaltmaktadır.



Şekil 7. YOLO Algılama Sistemi

Gerçek Zamanlı Sistemlerle Karşılaştırma: YOLO, hem hız hem de doğruluk açısından diğer gerçek zamanlı nesne tespiti yapan algoritmalarından daha iyi performans göstermektedir. Fast YOLO, saniyede 155 kare (FPS) hızında %52,7'lik etkileyici bir mAP değeri elde ederken, önceki gerçek zamanlı sistemlerin doğruluğunun iki katından fazla bir değere ulaşmaktadır. Standart YOLO modeli ise 45 FPS'de gerçek zamanlı performansı korurken %63,4'lük bir mAP'ye ulaşmıştır.

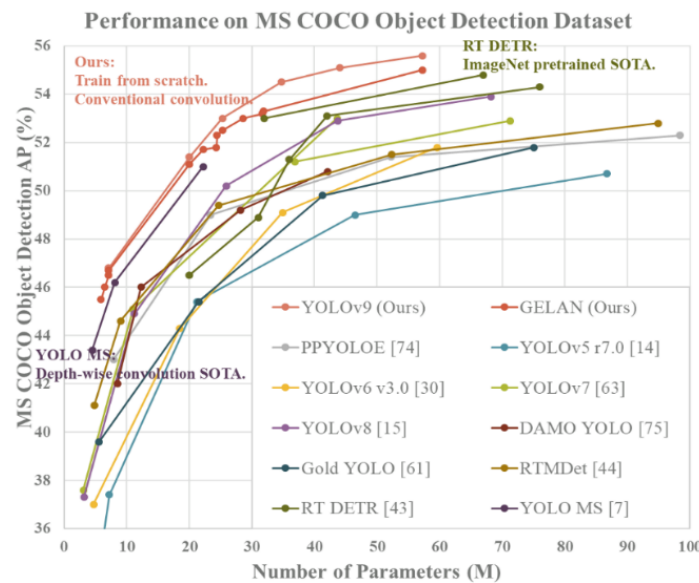
VOC 2007 Hata Analizi: Fast R-CNN ile karşılaştırıldığında YOLO, daha fazla konumlandırma hatası sergilerken önemli ölçüde daha az arka plan hatası göstermiştir. Bu, YOLO'nun hatalı pozitiflere daha az eğilimli olduğunu, nesneleri doğru bir şekilde ancak biraz daha az kesin konumlandırmayla tanımladığını göstermektedir.

YOLO'yu Fast R-CNN ile Birlikte Kullanma: Her iki yöntemin güçlü yanlarından yararlanarak YOLO'yu Fast R-CNN ile birleştirmek önemli bir performans artışı sağlanmaktadır. YOLO'nun arka plan hatalarını azaltma yeteneği Fast R-CNN'yi tamamlayarak elde edilen başarı VOC 2007 test setinde %3,2 mAP artışıyla %75,0'a çıkmaktadır.

Yeni Alanlara Genellenebilirlik: YOLO, Picasso ve People-Art gibi sanat eseri veri kümelerine uygulandığında R-CNN ve DPM'ye kıyasla daha iyi genellenebilirlik göstermektedir. YOLO'nun bağlamsal ilişkilerin yanı sıra nesne boyutunu ve şeklini modelleme yeteneği, sanat eseri ile doğal görüntüler arasındaki stilistik farklılıklara rağmen farklı alanlara etkili bir şekilde uyum sağlamasına olanak tanımaktadır (Redmon vd., 2016).

YOLO v1.0 2016, YOLO v2 2017, YOLOv3 2018, YOLOv4 2020, YOLOv5 2020, YOLOv6 2021, YOLOv7 2022, YOLOv8 2023 ve YOLOv9 2024 yılında yayınlanmıştır. Her yeni versiyonda daha hızlı ve iyi nesne tespiti yapmaktadır.

En güncel versiyon olan YOLOv9, performansı ve verimliliği artırmak için yeni konseptler sunarken öncülü olan modellerin güçlü yanlarını temel alarak gerçek zamanlı nesne algılamada önemli bir ilerlemeyi temsil etmektedir. YOLO serisinin bir üyesi olan YOLOv9, hızlı ve doğru nesne algılama yetenekleri sağlamayı hedefleyerek önceki versiyonların temel ilkelerini korumaktadır. Bununla birlikte, Programlanabilir Gradyan Bilgisi (PGI) ve Genelleştirilmiş Verimli Katman Toplama Ağı (GELAN) gibi iki önemli yeniliğin dahil edilmesiyle kendisini farklı kılmaktadır.



Şekil 8. MS COCO Veri Kümesindeki Gerçek Zamanlı Nesne Algılayıcılarının Karşılaştırılması

Programlanabilir Gradyan Bilgisi, bilgi darboğazı olarak bilinen bir olgu olan derin sinir ağlarındaki ileri besleme süreci sırasında meydana gelen bilgi kaybı sorunuyla mücadele etmektedir. PGI, yardımcı bir geri dönüşümlü dal kullanarak ağın öğrenmesini yönlendiren güvenilir gradyanlar oluşturmakta ve nesne algılamaya yönelik temel özelliklerin daha derin katmanlarda bile korunmasını sağlamaktadır. Bu yaklaşım, geleneksel derin denetim yöntemlerinin sınırlamalarını etkili bir şekilde ele alarak Programlanabilir Gradyan Bilgisini hem hafif hem de büyük ağ mimarilerine uygulanabilir hale getirmektedir (Wang vd., 2024).

Tablo 1. Son Teknoloji Ürünü Gerçek Zamanlı Nesne Tespiti Algoritmalarının Karşılaştırılması

Model	#Param. (M)	FLOPs (G)	$AP_{50:95}^{val} (%)$	$AP_{50}^{val} (%)$	$AP_{75}^{val} (%)$	$AP_S^{val} (%)$	$AP_M^{val} (%)$	$AP_L^{val} (%)$
YOLOv5-N r7.0 [14]	1.9	4.5	28.0	45.7	—	—	—	—
YOLOv5-S r7.0 [14]	7.2	16.5	37.4	56.8	—	—	—	—
YOLOv5-M r7.0 [14]	21.2	49.0	45.4	64.1	—	—	—	—
YOLOv5-L r7.0 [14]	46.5	109.1	49.0	67.3	—	—	—	—
YOLOv5-X r7.0 [14]	86.7	205.7	50.7	68.9	—	—	—	—
YOLOv6-N v3.0 [30]	4.7	11.4	37.0	52.7	—	—	—	—
YOLOv6-S v3.0 [30]	18.5	45.3	44.3	61.2	—	—	—	—
YOLOv6-M v3.0 [30]	34.9	85.8	49.1	66.1	—	—	—	—
YOLOv6-L v3.0 [30]	59.6	150.7	51.8	69.2	—	—	—	—
YOLOv7 [63]	36.9	104.7	51.2	69.7	55.9	31.8	55.5	65.0
YOLOv7-X [63]	71.3	189.9	52.9	71.1	51.4	36.9	57.7	68.6
YOLOv7-N AF [63]	3.1	8.7	37.6	53.3	40.6	18.7	41.7	52.8
YOLOv7-S AF [63]	11.0	28.1	45.1	61.8	48.9	25.7	50.2	61.2
YOLOv7 AF [63]	43.6	130.5	53.0	70.2	57.5	35.8	58.7	68.9
YOLOv8-N [15]	3.2	8.7	37.3	52.6	—	—	—	—
YOLOv8-S [15]	11.2	28.6	44.9	61.8	—	—	—	—
YOLOv8-M [15]	25.9	78.9	50.2	67.2	—	—	—	—
YOLOv8-L [15]	43.7	165.2	52.9	69.8	57.5	35.3	58.3	69.8
YOLOv8-X [15]	68.2	257.8	53.9	71.0	58.7	35.7	59.3	70.7
DAMO YOLO-T [75]	8.5	18.1	42.0	58.0	45.2	23.0	46.1	58.5
DAMO YOLO-S [75]	12.3	37.8	46.0	61.9	49.5	25.9	50.6	62.5
DAMO YOLO-M [75]	28.2	61.8	49.2	65.5	53.0	29.7	53.1	66.1
DAMO YOLO-L [75]	42.1	97.3	50.8	67.5	55.5	33.2	55.7	66.6
Gold YOLO-N [61]	5.6	12.1	39.6	55.7	—	19.7	44.1	57.0
Gold YOLO-S [61]	21.5	46.0	45.4	62.5	—	25.3	50.2	62.6
Gold YOLO-M [61]	41.3	87.5	49.8	67.0	—	32.3	55.3	66.3
Gold YOLO-L [61]	75.1	151.7	51.8	68.9	—	34.1	57.4	68.2
YOLO MS-N [7]	4.5	17.4	43.4	60.4	47.6	23.7	48.3	60.3
YOLO MS-S [7]	8.1	31.2	46.2	63.7	50.5	26.9	50.5	63.0
YOLO MS [7]	22.2	80.2	51.0	68.6	55.7	33.1	56.1	66.5
GELAN-S (Ours)	7.1	26.4	46.7	63.0	50.7	25.9	51.5	64.0
GELAN-M (Ours)	20.0	76.3	51.1	67.9	55.7	33.6	56.4	67.3
GELAN-C (Ours)	25.3	102.1	52.5	69.5	57.3	35.8	57.6	69.4
GELAN-E (Ours)	57.3	189.0	55.0	71.9	60.0	38.0	60.6	70.9
YOLOv9-S (Ours)	7.1	26.4	46.8	63.4	50.7	26.6	56.0	64.5
YOLOv9-M (Ours)	20.0	76.3	51.4	68.1	56.1	33.6	57.0	68.0
YOLOv9-C (Ours)	25.3	102.1	53.0	70.2	57.8	36.2	58.5	69.3
YOLOv9-E (Ours)	57.3	189.0	55.6	72.8	60.6	40.2	61.0	71.4

Kaynak: Wang, Yeh, ve Liao (2024)

2.7. Karışıklık matrisi (Confusion Matrix)

Tablo 2. Karmaşıklık Matrisi

DP	YN
YP	DN

Karmaşıklık matrisi, bir sınıflandırma modelinin performansı hakkında bilgi veren bir araçtır. Temelde bir tahmin işleminde bir sınıfın kaç kere doğru ve yanlış tahmin edildiğini ortaya koyar. Tablo 2’de 2x2’lik bir karmaşıklık matrisi gösterilmektedir. Satırların gerçek sınıfları ve sütunların tahminleri temsil ettiği bu ikili matris, tek sınıflı bir sınıflama modelinin sonuçlarının değerlendirilmesinde kullanılabilir (Caelen, 2017). Tabloda gösterilen değerler şunlardır:

DP (Doğru Pozitif): Sınıflandırma modeli tarafından “pozitif” olarak işaretlenen ve gerçekten de “pozitif” olan sonuçları ifade eder.

YN (Yanlış Negatif): Sınıflandırma modeli tarafından “negatif” olarak işaretlenen ancak gerçekte “pozitif” olan sonuçları ifade eder.

YP (Yanlış Pozitif): Sınıflandırma modeli tarafından “pozitif” olarak işaretlenen ancak gerçekte “negatif” olan sonuçları ifade eder.

DN (Doğru Negatif): Sınıflandırma modeli tarafından “negatif” olarak işaretlenen ve gerçekten de “negatif” olan sonuçları ifade eder (Powers, 2020).

2.8. Kesinlik (Precision), Hassaslık (Recall) ve F₁ Skoru

Kesinlik (precision), bir sınıflama modelinin ürettiği sonuçlardan doğru pozitif tahmin sayısının toplam “pozitif” tahmin sayısına oranını gösterir ve aşağıdaki formül kullanılarak hesaplanır:

$$Kesinlik = \frac{DP}{DP + YP}$$

Hassaslık (recall), doğru pozitif olarak yapılan tahminlerin toplam gerçek pozitif sayısına oranıdır ve aşağıdaki formül kullanılarak hesaplanır:

$$Hassaslık = \frac{DP}{DP + YN}$$

F₁ Skoru: Basit bir şekilde iki model kıyaslanmak istendiğinde, kesinlik ve hassaslık değerlerinin birlikte kullanıldığı aşağıdaki formülle F₁ skoru hesaplanabilir (Géron, 2022):

$$F1 \text{ skoru} = \frac{2}{\frac{1}{Kesinlik} + \frac{1}{Hassaslik}} = \frac{Kesinlik \times Hassaslik}{Kesinlik + Hassaslik} = \frac{DP}{DP \times \frac{YN + YP}{2}}$$

IoU – Kesişimin Birleşime Oranı (Intersection over Union): Bir görsel üzerinde modelin tahmin ettiği sınırlayıcı kutu ile nesnenin gerçekte içinde bulunduğu sınırlayıcı kutunun kesişiminin kapladığı alanın, bu iki kutunun birleşiminin kapladığı alana olan oranını ifade eder ve 1 ile 0 arasında bir değer alır. Bu değer 1'e yaklaştıkça bulunan nesnenin konumu daha doğru tahmin edilmiş demektir.

Kesinlik (precision), doğru tahmin edilen kutuların, tahmin edilen tüm sınırlayıcı kutulara oranını ifade ederken hassasiyet (recall), doğru tahmin edilen sınırlayıcı kutuların görsel üzerinde bulunan tüm nesnelerin sayısına olan oranını ifade etmektedir. Tahmin sayısı arttıkça kesinlik azalırken hassasiyet artmaktadır. Bir koordinat düzleminin y eksenine kesinlik (precision), x eksenine de hassasiyet (recall) değerlerini koyar ve bu iki değeri bir grafik olarak ele alırsak, kesişim noktalarında bir eğri oluşacaktır. Her bir hassasiyet değerinin karşılığı olan kesinlik değerlerinin ortalaması alınarak bir ortalama kesinlik değerine ulaşılabilir (AP – Average Precision). Bir veri setindeki tüm görsellerinin ortalama kesinlik değerlerinin ortalaması alınarak ortalama kesinlik (mAP – Mean Average Precision) değeri elde edilmektedir (Brownlee, 2019).

ÜÇÜNCÜ BÖLÜM

MODEL EĞİTİMLERİ

3.1. Veri Setinin Oluşturulması

Veri seti oluşturulurken aşağıdaki adımlar takip edilmiştir.

3.1.1. Veri Setinin İncelenmesi ve Hataların Düzeltilmesi

İlk aşamada, MURA veri seti incelenmiş ve potansiyel hatalar belirlenmiştir. Veri setindeki eksik veya yanlış etiketler, yanlış sınıflandırmalar veya bozuk görüntüler tespit edilerek gerekli düzeltmeler yapılmıştır. Bu aşamada, uzmanlar tarafından yapılan değerlendirme sonucunda belirlenen hatalar, veri setinin doğruluğunu artırmak amacıyla gözden geçirilerek etiketler düzeltilmiştir.

Çalışmanın devamında kullanılmak üzere pozitif görseller yani sadece bir kemik problemi olan görseller seçilmiştir. Orijinal veri setinde 14.863 vakadan elde edilmiş 40.561 adet röntgen görüntüsü bulunmaktadır. Bu görüntüler uzmanlar tarafından pozitif ya da negatif olarak sınıflandırılmıştır. İlk olarak tamamı pozitif etiketli 1.482 adet görselden oluşan bir örneklem seçilmiştir.

3.1.2. Görsellerin Etiketlenmesi ve Doğrulanması

Veri setindeki görüntüler, kemiklerdeki anomalilerin doğru bir şekilde tanımlanması için titizlikle etiketlenmiştir. Etiketleme işlemi, uzmanlar tarafından gerçekleştirilmiş ve her görüntünün içerdiği kemik yapısı ve olası problemler belirlenmiştir. Ardından, etiketlenen veri seti, uzmanlar tarafından yeniden gözden geçirilerek doğrulanmış ve etiketlenmiş verilerin doğruluğu teyit edilmiştir.

Görsellerin etiketlenmesinde açık kaynak kodlu labelImg ve label-studio uygulamaları kullanılmıştır.

3.2. Model Oluřturma

Bu bölümde nesne tespiti ve sınıflandırma modellerinin hangi aşamalar takip edilerek eğitildiğı, eğitim sonucunda elde edilen mAP50 değerleri ve performans grafikleri incelenecektir.

Yapılan çalışmada i5 13. nesil Intel işlemci, 32GB RAM, ve Nvidia 3060Ti donanımlı bir bilgisayar kullanılmıştır. Donanımın değışmesi durumunda parametrelerin değıştirilmesi uygun olacaktır. Ayrıca donanımın değıştirilmesi eğitim ve tahmin sürelerini doğal olarak etkileyecektir.

3.2.1. Modelin Oluřturulması ve Eğitimi

Çalışmada tablo 3'te performans ölçütleri verilen YOLO algoritmasının 8. versiyonu kullanılarak model eğitimleri yapılmıştır. Bu versiyonun nano, small, medium ve large diye adlandırılan ön-öğrenmeli modelleri kullanılmıştır. Tüm deneylerde veri setleri %70 eğitim (train), %20 doğrulama (validation) ve %10 test verisi olarak kullanılmıştır.

Tablo 3. YOLOv8 Performans Ölçütleri

Model	Boyut (Piksel)	mAP Değeri (50-95)	Hız (CPU ONNX) (ms)	Hız (A100 TensorRT) (ms)	Parametreler (M)	FLOPs (B)
YOLOv8n	640	37.3	80.4	0.99	3.2	8.7
YOLOv8s	640	44.9	128.4	1.20	11.2	28.6
YOLOv8m	640	50.2	234.7	1.83	25.9	78.9
YOLOv8l	640	52.9	375.2	2.39	43.7	165.2

Kaynak: https://docs.ultralytics.com/models/yolov8/#__tabbed_1_1

3.2.2 Model Eğitim Süreci

Bu bölümde, model oluřturma aşamasında gerçekleřtirdiğimiz eğitimler ve bu eğitimlerin sonuçları incelenecektir. Ancak, eğitime başlamadan önce, eğitilen modellerin başarı ölçütlerinin ve kayıp verilerinin gösterildiğı grafiklerin anlamlarına ve bu grafiklerin řu şekilde yorumlanabilir:

Eğitim Kayıpları

train/box_loss, val/box_loss: Bu grafikler, sınırlayıcı kutularda meydana gelen kaybı gösterir. Kayıp, modelin nesnelerin konumunu doğru bir şekilde tahmin etme yeteneğini temsil eder ve nesnenin gerçek konumu ile tahmin edilen konumu arasındaki farktan kaynaklanır. Eğitim sırasında bu kaybın azalması eğitimin iyileştiğini göstermektedir.

train/cls_loss, val/cls_loss: Bu grafikler, sınıflandırma kaybını temsil etmektedir. Kayıp, modelin nesnelerin sınıfını doğru bir şekilde tahmin etme yeteneğini gösterir ve gerçek sınıf ile tahmin edilen sınıfın birbirinden farklı olmasından kaynaklanır. Eğitim esnasında kayıpların azalması beklenir.

train/dfl_loss, val/dfl_loss: Bu grafikler, nesnenin güvenilirlik puanını tahmin etmedeki kaybı gösterir. Kayıp, modelin nesnenin varlığına ilişkin güvenilirliğini doğru bir şekilde tahmin etme yeteneğini temsil eder. Eğitim sırasında bu kayıp azalmalıdır (Pingale vd., 2024).

Performans Ölçütleri

metrics/precision(B), metrics/recall(B): Bu grafikler, modelin hassasiyetini ve geri çağırma oranını gösterir. Hassasiyet, doğru pozitif tahminlerin toplam pozitif tahminlere oranını temsil eder. Geri çağırma, doğru pozitif tahminlerin gerçek pozitiflerin toplam sayısına oranını temsil eder. Eğitim sırasında bu ölçütlerin artması eğitimin beklendik yönde ilerlediğini gösterir (Prabhu vd., 2022).

metrics/mAP50(B), metrics/mAP50-95(B): Bu grafikler, ortalama hassasiyet (mAP) değerini gösterir. mAP, modelin farklı sınır kutusu örtüşme eşiklerindeki performansının ortalama bir ölçüsüdür. Eğitim sırasında mAP değerinin artması beklenir (Reis vd., 2023).

3.2.2 Model Eğitim Süreci 1. Aşama

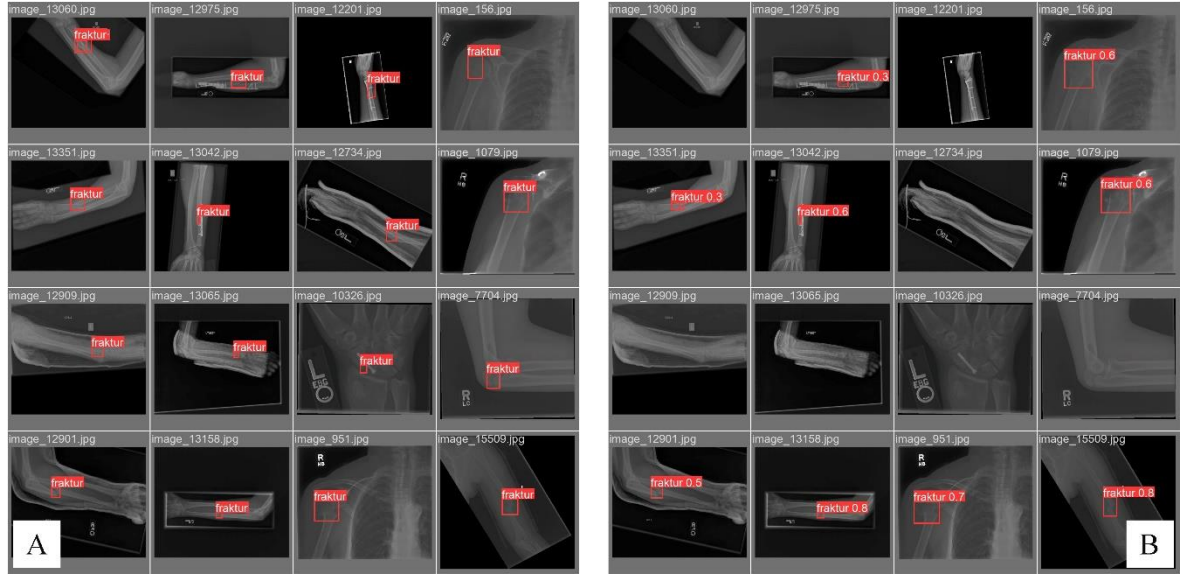
Bu eğitimde parmak, bilek, dirsek, ön kol, üst kol ve omuz kemiklerinden oluşan bir veri seti kullanılmıştır.

Kullanılan parametreler: Tüm eğitimlerde resim boyutu 512 pixel olarak belirlenmiş, epoch (eğitim döngüsü) olarak 300 döngü sayısı girilmiş, batch (toplu işleme sayısı) 8 ve worker (ön işleme işlemlerinin paralelleştirilmesi anlamına gelen işçi sayısı) 4 olarak belirlenmiş ve sonuçları tablo 4'te gösterilmiştir.

Tablo 4. Model Eğitimi 1. Aşama Tahmin Sonuçları

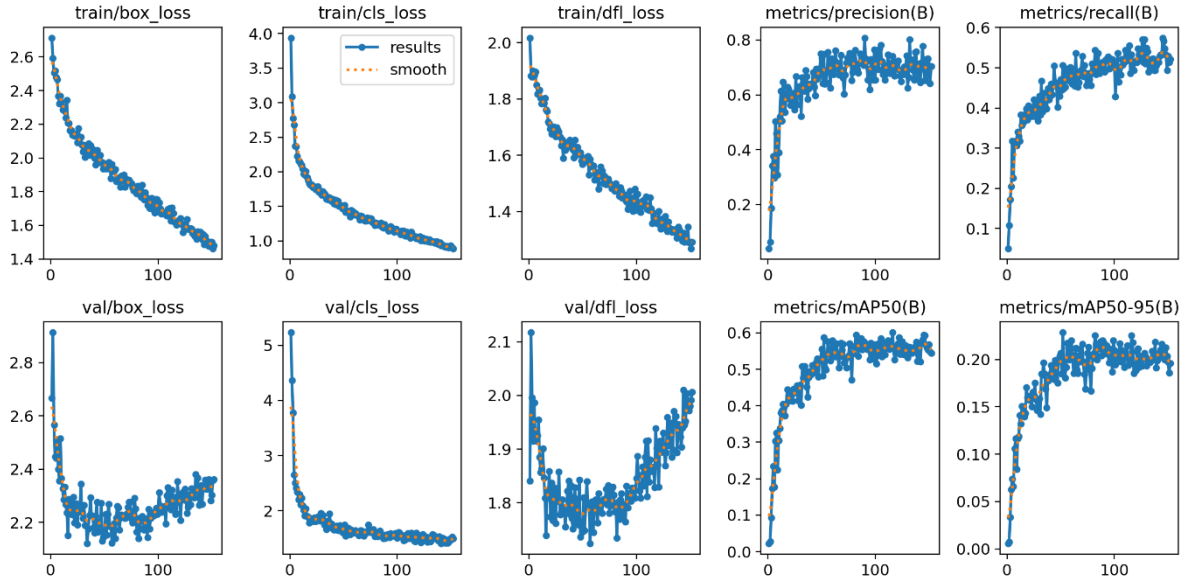
Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
YOLOv8n	168/300	512	8	4	0,75	0,51	0,59	0,23	960	5,71
YOLOv8s	228/300	512	8	4	0,81	0,56	0,66	0,25	1744	7,65
YOLOv8m	162/300	512	8	4	0,78	0,53	0,62	0,24	2235	13,8
YOLOv8l	300/300	512	8	4	0,76	0,53	0,60	0,24	5689	18,96

Şekil 9’da 1. aşamada gerçekleştirilen etiketleme ve tahmin örnekleri gösterilmektedir.

**Şekil 9.** Model Eğitimi 1. Aşama (A) Etiket ve (B) Tahmin Örnekleri

Tablo 4’te görüldüğü üzere, bu denemede eğitilen modellerden en iyi ortalama ortalama kesinlik (mAP) değerine sahip olan model, YOLOv8s ön-eğitilmiş modelle eğitilen modeldir.

Her ne kadar başlangıçta döngü sayısını 300 olarak girilmiş olsa da algoritma daha fazla iyileştirme yapamadığı 50. denemeden sonra döngüyü kesmiştir. Bu durum algoritmanın başarısını etkilemezken eğitim süresinin farklılaşmasına sebep olmuştur. Nano model ile başlatılan eğitim toplam 960 saniye sürmüştür ve her bir döngü için ortalama 5,71 saniyeye ihtiyaç duymuştur. Bununla birlikte en uzun sürede tamamlanan eğitim (geniş) large model ile başlatılan eğitimidir ve 5.689 saniye sürmüştür her bir döngü için de ortalama 19 saniyeye ihtiyaç duymuştur. En iyi mAP50 değerini elde ettiğimiz (küçük) small modelle başlanan eğitimin sonuçlanması toplam 1.744 saniye sürmüştür ve her bir döngüyü ortalama 7,65 saniyede tamamlamıştır. En iyi mAP50 değeri %66, en kötüsü ise %59 olarak tespit edilmiştir. En iyi ve en kötü mAP50 değerine sahip modeller arasında %7’lik bir fark bulunmaktadır.



Şekil 10. Model Eğitimi 1. Aşama Performans Grafikleri

Model eğitimi tamamlandığı zaman, eğitimin geçirdiği süreçler, kayıplar ve performans ölçütleri şekil 10'daki grafiklerle gösterilmiştir. Bu grafikler, genel olarak modelin eğitim sırasında öğrendiğini ve performansının arttığını göstermektedir. Ancak, train/box_loss ve val/box_loss grafikleri göz önünde bulundurulduğunda modelin doğrulama kaybının, eğitim kaybindan daha yüksek bir seviyede seyrediyor olması overfitting ve performans durgunluğu gibi potansiyel sorunlarının olabileceğine işaret etmektedir.

3.2.2 Model Eğitimi 2. Aşama

Röntgen görüntüleri kırık merkezde olacak şekilde kısa kenara oranlayarak kırılarak röntgen görüntüleri yeniden düzenlenerek yeni veri seti oluşturulmuştur.

Kullanılan parametreler:

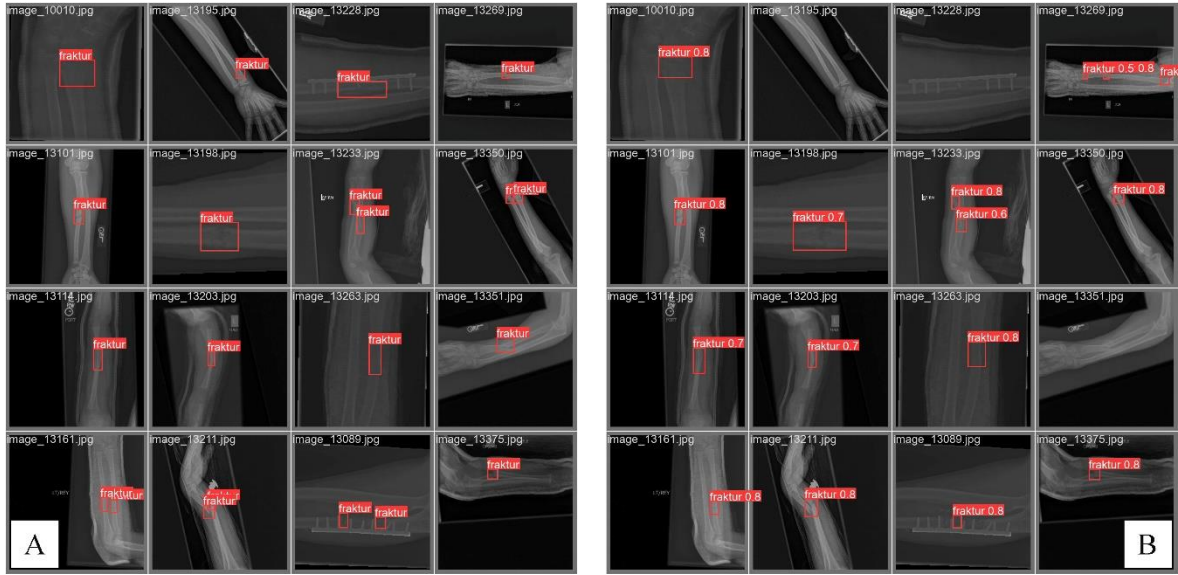
Yeni veri setinde resim boyutu 512 pixel olarak belirlenmiş, epoch olarak 300 döngü sayısı girilmiş, batch 8 ve worker sayısı 4 olarak belirlenmiştir.

Model geliştirme sürecinin bu aşamasında röntgen görüntüleri kırıkları merkeze alarak eni ve boyu, kısa kenara eşit olacak şekilde kare hale getirilmiştir. Ancak, orijinal görüntü üzerinde başka bir işlem yapılmamıştır. Bu aşamada Model Eğitimi 1. Aşamasındaki rutinler tekrar edilmiştir. Elde edilen sonuçlar tablo 4'te verilmiştir.

Tablo 5. Model Eğitimi 2. Aşama Tahmin Sonuçları

Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
YOLOv8n	194/300	512	8	4	0,67	0,57	0,62	0,24	2.441,71	12,59
YOLOv8s	300/300	512	8	4	0,75	0,60	0,67	0,26	3.503,96	11,68
YOLOv8m	300/300	512	8	4	0,73	0,61	0,66	0,24	4.493,98	14,98
YOLOv8l	300/300	512	8	4	0,71	0,60	0,63	0,22	5.818,85	19,40

Şekil 11’de 2. aşamada gerçekleştirilen etiketleme ve tahmin örnekleri gösterilmektedir.

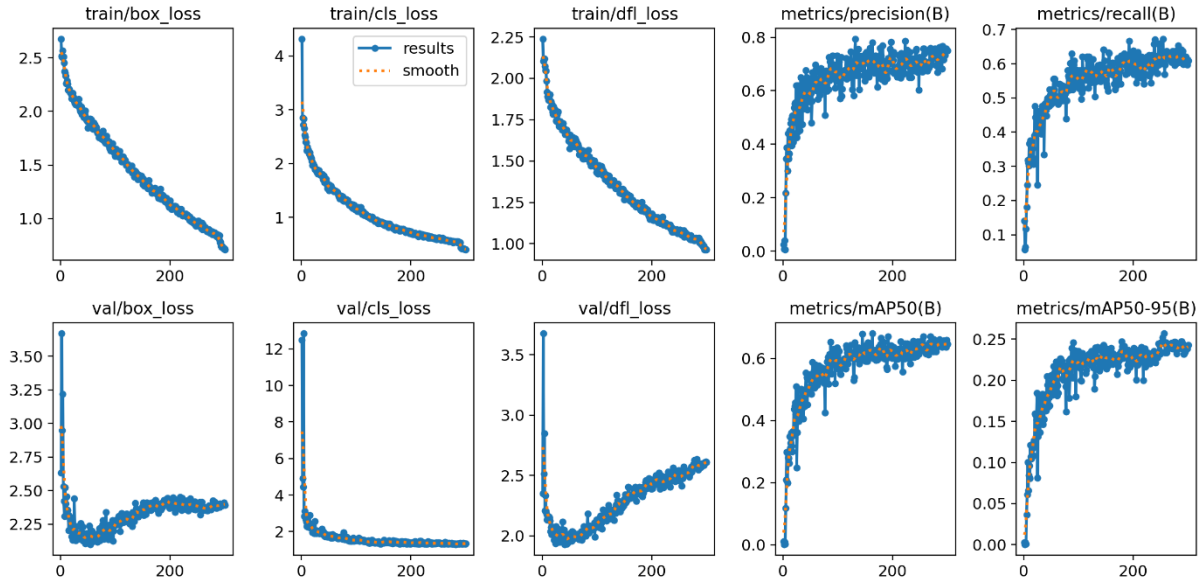
**Şekil 11.** Model Eğitimi 2. Aşama (A) Etiket ve (B) Tahmin Örnekleri

Tablo 5’te ve şekil 11’de model eğitiminin 2. aşamasında kullanılan parametre ve sonuçları yer almaktadır. Bu denemede eğitilen modellerden en iyi mAP değerine sahip olan model tablo 4’te de görüldüğü gibi YOLOv8s ön-eğitilmiş modelle eğitilen modeldir. Nano model dışında kalan tüm modeller başlangıçta tanımladığımız 300 eğitim döngüsünü tamamlamışken, nano modelde 194. döngüde erken durma (early stop) gerçekleşmiştir. Eğitimin tamamlanma süreleri incelendiğinde nano model ile başlatılan eğitim toplam 2.441 saniye sürmüş ve her bir döngü ortalama 12,59 saniyede tamamlanmıştır. Bununla birlikte en uzun sürede tamamlanan eğitim large model ile başlatılan eğitimidir ve 5.818,85 saniye almış ve her bir döngü için de yaklaşık 19,40 saniyeye ihtiyaç duymuştur. En iyi mAP değerini elde ettiğimiz small modelle başlanan eğitimin sonuçlanması toplam 3.503,96 saniye sürüp her bir döngüyü ortalama 11,68 saniyede tamamlamıştır. En iyi mAP50 değeri %67, en kötüsü ise %62 olarak tespit edilmiştir. En iyi ve en kötü mAP50 değerine sahip modeller arasında %5’lik bir fark bulunmaktadır.

İlk iki aşamadaki eğitim sonuçları karşılaştırıldığında, 2. aşamada genel olarak daha iyi bir performans sergilediği görülmektedir. Her iki eğitimde de daha büyük model boyutları (YOLOv8s, YOLOv8m ve YOLOv8l) daha yüksek doğruluk skorları (P, R, mAP50, mAP50-95) sağlamıştır. Bununla birlikte, 2. aşamada YOLOv8n modeli, 1. aşamaya kıyasla daha yüksek doğruluk skorlarına ulaşmıştır. Bu da 2. Aşamadaki eğitimin veri kümesinin veya eğitim parametrelerinin daha yüksek performans elde etmeye katkıda bulunacağı gözlenmektedir.

Eğitim süresi açısından, 2. aşamadaki modeller, 1. aşamadakilere kıyasla çok daha uzun sürede eğitilmiştir. Bunun sebebi 2. aşamada eğitilen modellerin daha fazla döngüde tamamlanmış olmasıdır.

Genel olarak özetleyecek olursak, 2. eğitim aşamasında daha yüksek doğruluk skorları ve daha uzun eğitim süresiyle daha iyi performans gösterdiği sonucuna varılmıştır. Her iki aşamanın eğitim sonuçları, YOLOv8 modellerinin veri kümesi ve eğitim parametrelerine göre farklı performanslar gösterebileceği tespit edilmiştir.



Şekil 12. Model Eğitimi 2. Aşama Performans Grafikleri

Şekil 12’de verilen grafikler genel olarak değerlendirildiğinde, modelin performansında iyileşme olduğu, kayıpların azaldığı ve performans ölçütlerinin arttığı görülmektedir. Bununla birlikte, doğrulama kaybının eğitim kaybından daha yüksek olması ve bazı ölçütlerde durgunlaşma gibi olası sorunlar da bulunmaktadır.

3.2.2 Model Eğitimi 3. Aşama

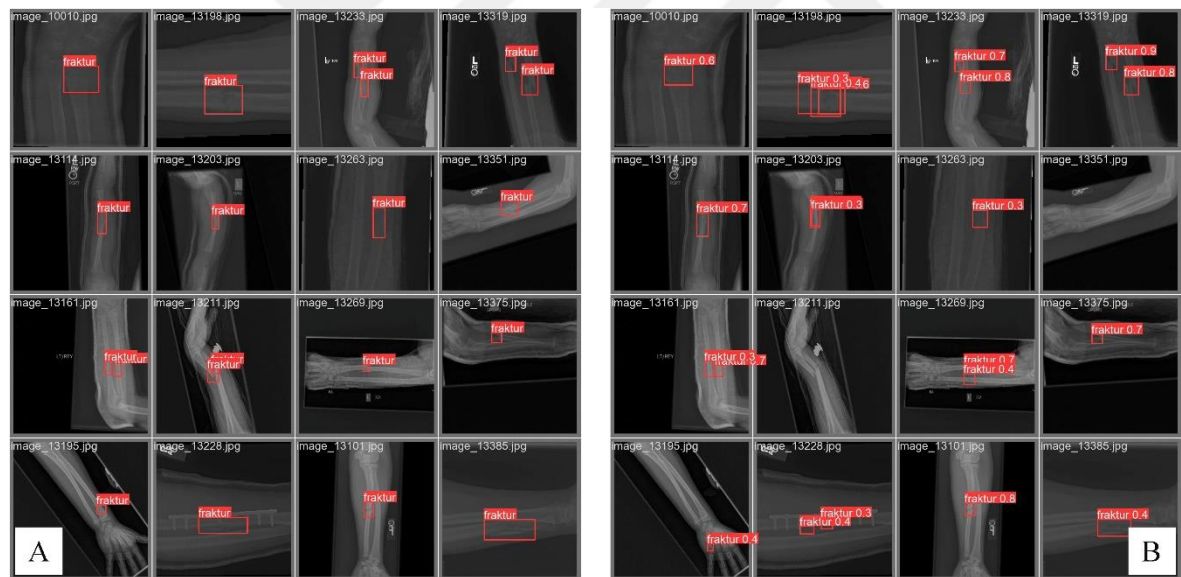
Bu aşamada, 2. aşamada kırparak kare şekline getirilen röntgen görüntülerinin tamamı yeniden boyutlandırılarak 512 x 512 pixel olacak şekilde normalleştirilmiştir.

Kullanılan parametreler: Yeni veri setinde resim boyutu 512 pixel olarak belirlenmiş, epoch olarak 300 döngü sayısı girilmiş, batch 8 ve worker sayısı 4 olarak belirlenmiştir. Modelin eğitim parametreleri ve performans ölçütleri tablo 6’da gösterilmiştir.

Tablo 6. Model Eğitimi 3. Aşama Tahmin Sonuçları

Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
YOLOv8n	253/300	512	8	4	0,74	0,55	0,62	0,22	1.730,32	6,84
YOLOv8s	179/300	512	8	4	0,65	0,56	0,60	0,21	1.430,33	7,99
YOLOv8m	239/300	512	8	4	0,68	0,62	0,65	0,22	3.272,13	13,69
YOLOv8l	300/300	512	8	4	0,76	0,53	0,61	0,22	5.769,74	19,23

3. aşamada gerçekleştirilen etiketleme ve tahmin örnekleri şekil 9’da gösterilmektedir.



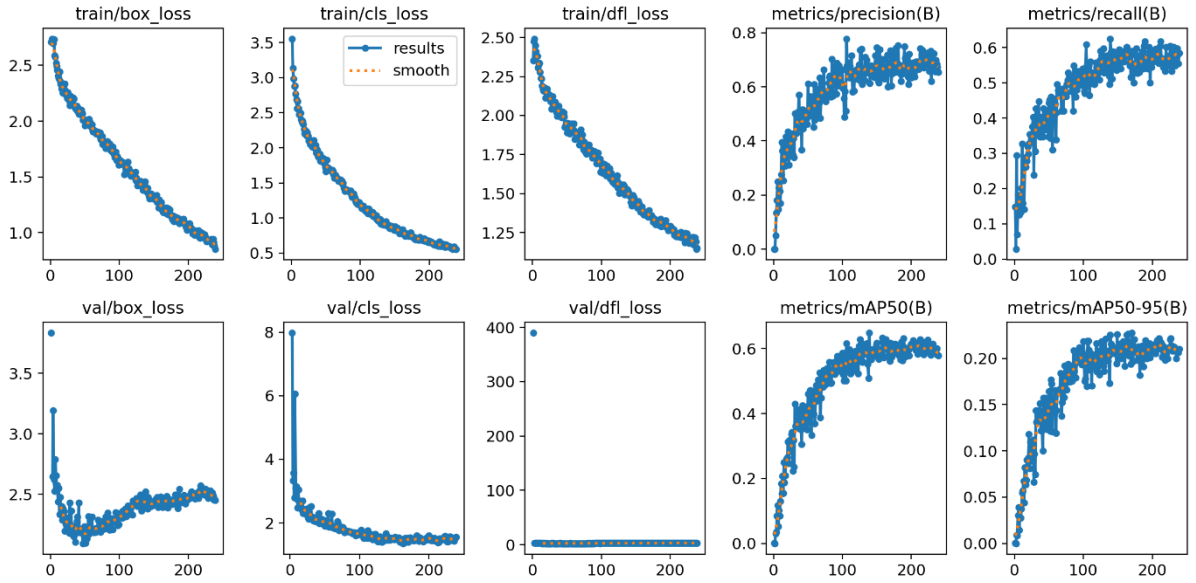
Şekil 13. Model Eğitimi 3. Aşama (A) Etiket ve (B) Tahmin Örnekleri

Şekil 13’te 3. aşamada eğitilen modelin eğitim öncesi etiketleri ve eğitilen modelle yapılan tahmin sonuçları gösterilmektedir. Bu eğitimde YOLOv8m ön-eğitilmiş modelle eğitilen modelin, bu veri setiyle eğitilen modeller arasında en iyi mAP50 değerine sahip olan model olduğu gözlenmektedir. Sadece YOLOv8l adlı ön-eğitilmiş model ile yapılan eğitim başta tanımladığımız 300 döngü sayısına ulaşmıştır. Diğer modellerle yapılan eğitimlerde erken durma gerçekleşmiştir. YOLOv8n modeliyle yapılan eğitim toplam 253 eğitim döngüsü sürmüş ve 1.730 saniyede tamamlanmıştır. Her bir eğitim döngüsü ortalama 6,84 saniyede tamamlanmıştır ve döngü başına en kısa süre gerektiren model olmuştur. İkinci sırada

YOLOv8s modeli gelmektedir. Bu model ile yapılan eğitimde 179 eğitim döngüsü sonrasında erken durma gerçekleşmiştir. Bu işlem toplam 1.430 saniye almıştır. Her bir döngü için gereken zaman ortalama 8 saniyedir. Üçüncü sırada gelen YOLOv8m modeliyle yapılan eğitimde de erken durma yaşanmıştır. Toplam 239 eğitim döngüsü süren eğitim 3272 saniyede tamamlanmıştır. Döngü başına 13,69 saniye zaman geçmiştir. En uzun sürede tamamlanan eğitim YOLOv8l modeliyle yapılan ve 300 eğitim döngüsü süren eğitimidir. Bu eğitim 3.272 saniyede tamamlanarak döngü başına ortalama 13,69 saniye almıştır.

Eğitimler sonunda elde edilen mAP50 değerleri sırasıyla small için %60, large için %61, nano için %62 ve medium için %65 olarak hesaplanmıştır. En iyi ile en kötü model arasında %5'lik bir fark bulunmaktadır.

Sonuç olarak, YOLOv8m modeli, bu veri setinde en yüksek mAP değerini elde etmiştir. Bununla birlikte, YOLOv8n modeli, en hızlı eğitim süresine sahip olmuştur.



Şekil 14. Model Eğitimi 3. Aşama Performans Grafikleri

Şekil 14'teki grafikler genel olarak modelin performansında iyileşme olduğunu göstermektedir. Kayıplar azalıyor ve performans ölçütleri artıyor. Ancak; doğrulama kaybının, eğitim kaybından daha yüksek olması ve bazı ölçütlerde durgunlaşma gibi olası sorunlar da bulunmaktadır. Bu sorunları gidermek için daha fazla eğitim veya farklı bir eğitim stratejisi kullanılabilir.

Model eğitiminin 1, 2 ve 3. aşamasında elde edilen model verileri incelenip sonuçları karşılaştırıldığında, 3. aşamada gerçekleştirilen eğitimin genel olarak en iyi performansı gösterdiği sonucuna varılabilir. Her üç deneyde de daha büyük model boyutları (YOLOv8s,

YOLOv8m ve YOLOv8l) daha yüksek doğruluk skorları (P, R, mAP50, mAP50-95) sağlamıştır. 3. aşamada YOLOv8n modeli, 1. ve 2. aşamada gerçekleştirilen eğitimlere kıyasla daha kısa eğitim süresiyle daha yüksek doğruluk skorları elde etmiştir. Bu, 3. aşamanın veri kümesinin veya eğitim parametrelerinin daha yüksek performans elde etmeye katkıda bulunmuş olabileceğini düşündürmektedir.

Eğitim süresi açısından, 2. aşamadaki modeller en uzun sürede eğitilmiş, ardından 3. aşama ve onun ardından da 1. aşama gelmiştir. Bu farklılık, kullanılan veri kümesinin büyüklüğü, eğitimin gerçekleştiği döngü sayısı ve diğer eğitim parametrelerindeki farklılıklardan kaynaklanıyor olabilir.

Genel olarak, 3. aşamada eğitilen modelin daha yüksek doğruluk skorları ve daha kısa eğitim süresiyle daha iyi performans gösterdiği sonucuna varılabilir.

3.2.3 Model Eğitimi 4. Aşama

MURA veri seti, dirsek, parmak, önkol, el, üst kol kemiği, omuz ve bilek olmak üzere kol üzerindeki 7 farklı bölgede meydana gelen birbirinden farklı anomalilerin röntgen görüntülerini içeren bir veri setidir. Bu veri setinin tamamında bulunan tüm kırıklar üzerinde çalışma yapmak çok fazla zaman ve işlem gücüne ihtiyaç duyulmasına sebep olmaktadır. Ayrıca tablo 4'te gösterilen en yüksek mAP50 değeri olan %67, beklenen başarı seviyesine ulaşmamıştır. Bu yüzden sadece bilekte meydana gelen anomalilerden oluşan daha küçük bir veri seti ve yeni bir yaklaşım ile eğitimlere devam edilmiştir. Bu yaklaşımda nesne (kırık) tespiti yapılmadan kemikleri kırık olan ya da olmayan şeklinde YOLOv8-cls ön-eğitilmiş algoritması kullanılarak eğitilen sınıflama modeli kullanılarak sınıflandırdıktan sonra sadece bilekte ve kırık içeren kemiklerden oluşan veri setiyle bir kırık tespiti modeli geliştirilmiştir.

Bu modelin eğitiminde de YOLOv8 ailesinin ön-eğitilmiş sınıflama modellerinden YOLOv8l-cls adlı modeli kullanılmıştır. %74 doğruluk (accuracy) değerine sahip olan sınıflama modeli, girdi olarak verilen bilek görselinin kırık olup olmadığını incelemektedir. Eğer görsel kırık bir bilek kemiğine aitse pozitif değer döndürmektedir.

Sonraki aşamada sınıflama modelinin kırık olarak belirlediği 435 röntgen görüntüsü model eğitiminin 2. aşamasında elde edilen en başarılı (%67 mAP50 değerine sahip olan ve YOLOv8s ön-eğitilmiş modeliyle eğitilen) model kullanılarak yeniden etiketlenmiştir.

Kullanılan parametreler: Yeni veri setinde resim boyutu 512 pixel olarak belirlenmiş, epoch

olarak 500 döngü sayısı girilmiş, batch 8 ve worker sayısı 4 olarak belirlenmiştir.

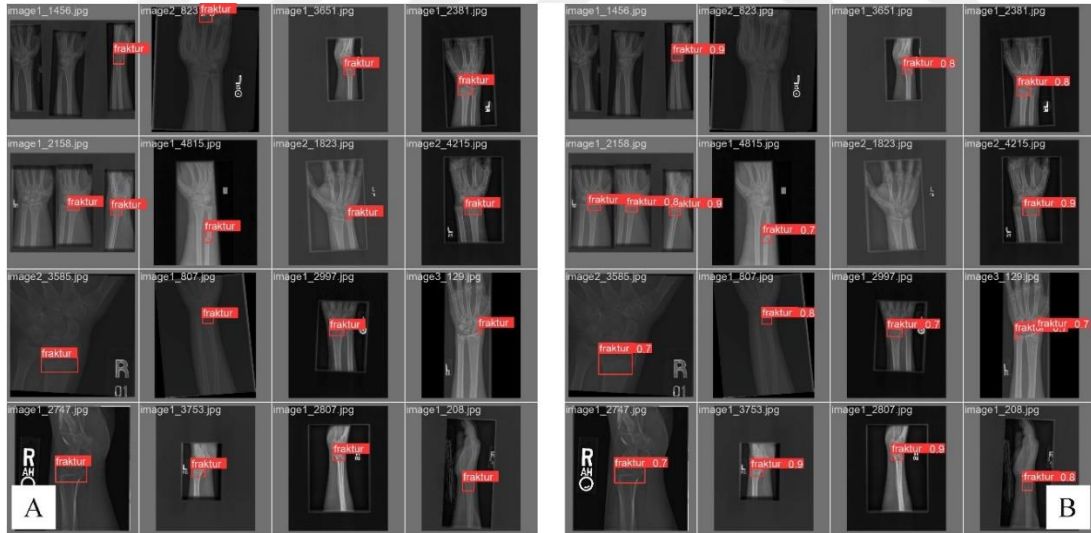
Eğitimin bu aşamasında YOLOv8l ön-eğitimli modeliyle yeni bir kırık tespiti modeli eğitilmiştir. Bu eğitimde kullanılan parametre ve sonuçlar tablo 7’de gösterilmiştir:

Tablo 7. Model Eğitimi 4. Aşama Tahmin Sonuçları

Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
YOLOv8l	288/500	512	8	4	0,88	0,82	0,86	0,53	2.457,00	8,53

Bu eğitim aşamasında, model YOLOv8l ön-eğitimli modelden öğrenme transferi yapılarak eğitilmiştir. Tablo 7’de görüldüğü üzere; eğitimde kullanılan döngü sayısı 500’e çıkarılmış ancak resim boyutu, batch ve worker sayısı önceki eğitimlerle aynı tutulmuştur. Eğitimde 288. döngüde erken durma gerçekleştirmiş ve 2.457 saniyede tamamlamıştır. Her bir eğitim döngüsü ortalama 8,53 saniyede tamamlanmıştır.

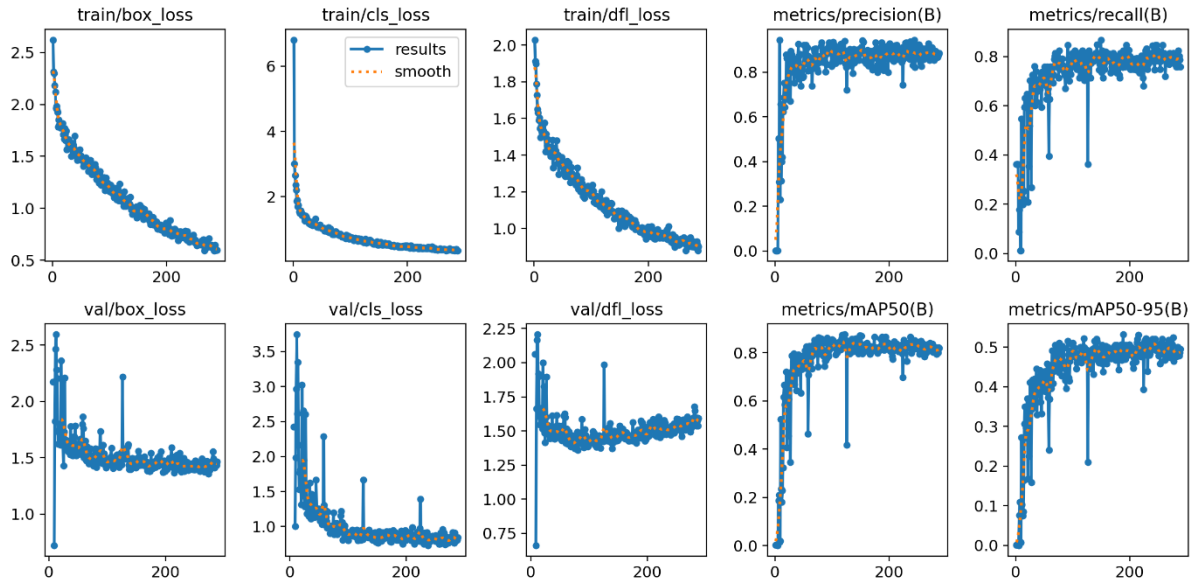
Eğitimler sonunda elde edilen mAP50 değeri %86’dır ve önceki eğitimlere göre %19 daha yüksek bir değere ulaşılmıştır. Bu aşmada gerçekleştirilen eğitime ait etiketleme ve tahmin örnekleri şekil 15’te gösterilmiştir.



Şekil 15. Model Eğitimi 4. Aşama (A) Etiket ve (B) Tahmin Örnekleri

Şekil 15’te model eğitiminin 4. aşamasında (A) el ile etiketlenmiş röntgen görüntüleri ve (B) geliştirilen model tarafından yapılan tahminler verilmiştir.

Şekil 16’daki grafikler modelin performansında iyileşme olduğunu göstermektedir. Eğitimin kayıplarının azaldığı ve performans ölçütlerinin arttığı görülüyor. Ancak, doğrulama kaybının eğitim kaybından daha yüksek olduğu ve bazı ölçütlerde durgunlaşma gözlenmektedir.



Şekil 16. Model Eğitimi 4. Aşama Performans Grafikleri

3.2.3 Model Eğitimi 5. Aşama

Kullanılan parametreler:

Yeni veri setinde resim boyutu 512 pixel olarak belirlenmiş, epoch olarak 500 döngü sayısı girilmiş, batch 8 ve worker sayısı 4 olarak belirlenmiştir.

Bu aşamada 4. aşamada eğitilen ve %80 mAP50 değerine ulaşan model kullanılarak sadece kırık içeren bilek görüntüleri yeniden etiketlenerek veri seti genişletilmiştir. Bu işlem sonucunda, veri setinin boyutu 435'ten 610'a yükselmiştir. Böylece tamamı etiketlenmiş ve bir önceki eğitime kıyasla genişlemiş bir veri seti elde edilmiştir. Bu yeni veri seti kullanılarak, yine YOLOv8l ön-eğitilmiş modeliyle yeni bir model eğitimi gerçekleştirilmiştir.

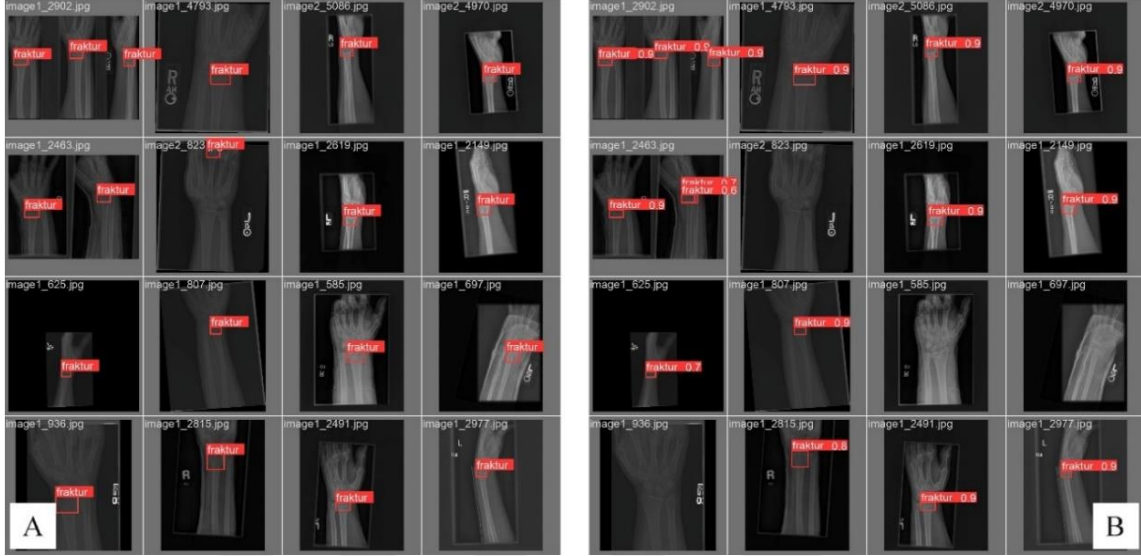
Tablo 8. Model Eğitimi 5. Aşama Tahmin Sonuçları

Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
YOLOv8l	317/500	512	8	4	0,91	0,88	0,93	0,64	3.606,00	11,38

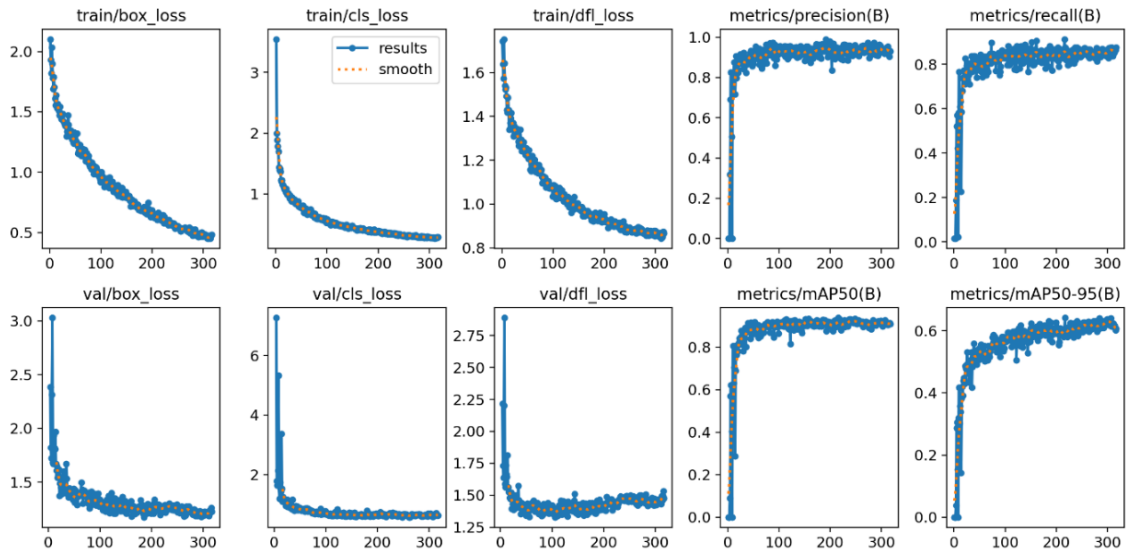
Tablo 8’de model eğitimin 5. aşamasında elde edilen eğitim performansı özetlenmektedir.

Eğitimin başında 500 döngü hedeflenirken 317. döngüde erken durdurulmuştur. Eğitim toplam 3.606 saniyede tamamlanmıştır ve her bir eğitim döngüsü ortalama 11,38 saniye sürmüştür.

Eğitimler sonunda elde edilen mAP50 değeri %93'dür. Bu değer, 4. aşamada elde edilen %86'lık mAP50 değerine kıyasla %7'lik bir artış göstermektedir. Şekil 17'de bu eğitim aşamasında gerçekleştirilen etiketleme ve tahmin örnekleri gösterilmektedir.



Şekil 17. Model Eğitimi 5. Aşama (A) Etiket ve (B) Tahmin Örnekleri



Şekil 18. Model Eğitimi 5. Aşama Performans Grafikleri

Şekil 18'deki grafikler modelin performansında iyileşme olduğu, kayıplar azalırken performans ölçütlerinin artmasına rağmen doğrulama kaybının eğitim kaybından daha yüksek olduğu ve bazı ölçütlerde durgunlaşma gibi olası sorunların bulunduğu görülmektedir. Bu sorunlar daha fazla eğitim veya daha geniş bir veri setinin kullanımıyla giderilebilir.

3.2.3 Model Eğitimi 6. Aşama

Kullanılan parametreler: Yeni veri setinde resim boyutu 512 pixel olarak belirlenmiş, epoch olarak 500 döngü sayısı girilmiş, batch 8 ve worker sayısı 4 olarak belirlenmiştir.

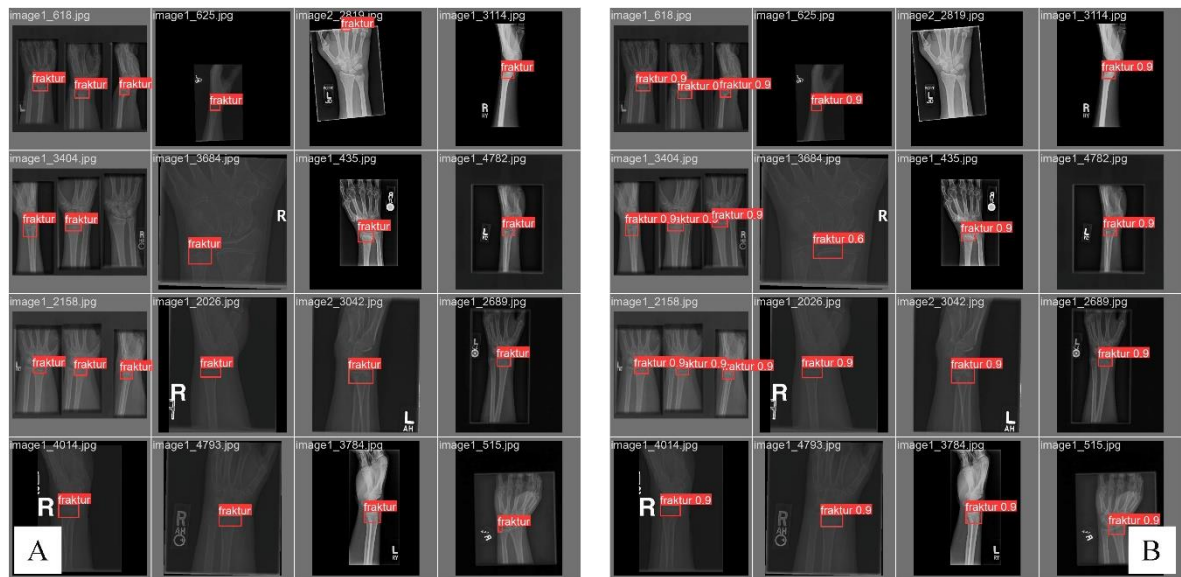
Bu aşamada, 5. aşamasında eğitilen model, kırık içeren tüm bilek röntgen görüntülerinin etiketlenmesi için kullanılmıştır. Bu etiketleme işlemi sonucunda, veri setinin boyutu 610'dan 746'ya çıkarılmıştır. Böylece, tamamı etiketlenmiş ve önceki deneylere kıyasla daha büyük bir veri seti elde edilmiştir. Yeni oluşturulan bu genişletilmiş veri seti ile aynı eğitim parametreleri kullanılarak bir model eğitimi daha gerçekleştirilmiştir. Bu yöntemin tutarlılığı ve tekrarlanabilirliği açısından, eğitim parametrelerinin sabit tutulması önemli bir adım olarak değerlendirilmektedir. Tablo 9'da bu eğitimin sonuçları olarak sunulmuştur.

Tablo 9. Model Eğitimi 6. Aşama Tahmin Sonuçları

Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
YOLOv8l	463/500	512	8	4	0,93	0,88	0,91	0,68	6.717,00	14,51

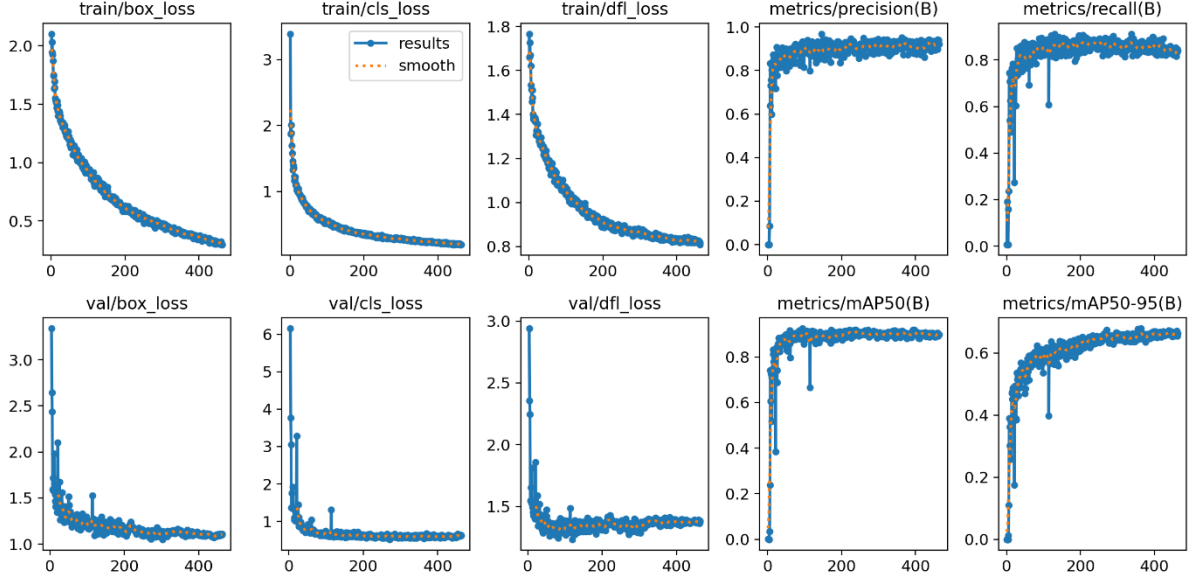
Bu eğitimin 500 döngüde tamamlanması hedeflenmiş ancak 463. Döngüde algoritma tarafından erken durdurulmuştur. Eğitim tamamlanması toplam 6.717 saniye sürmüştür. Her bir eğitim döngüsü ortalama 14,51 saniyede tamamlanmıştır.

Eğitimler sonunda elde edilen mAP50 değeri %91'dir. Bu değer, önceki eğitimde elde edilen %93'lük mAP50 değerine kıyasla %2'lik bir azalma göstermektedir. Eğitim öncekinden daha düşük bir performans gösterdiği için bu yaklaşım sonlandırılmıştır.



Şekil 19. Model Eğitimi 6. Aşama (A) Etiket ve (B) Tahmin Örnekleri

Şekil 20’de model eğitiminin 6. Aşamasına ait performans grafikleri gösterilmektedir. Grafiklerde kayıplar azalırken performans ölçütlerinin arttığı görülmektedir. Bu modelin eğitiminde iyileşme olduğunu göstermektedir. Bununla birlikte grafiklerde doğrulama kaybının eğitim kaybından daha yüksek olduğu da gözlemlenmektedir.



Şekil 20. Model Eğitimi 6. Aşama Performans Grafikleri

3.2.3 Sonuç Grafiklerinin Yorumlanması

İlk 3 aşamada elde edilen sonuç grafiklerinin genel özellikleri Tablo 7’de gösterilmiştir ve şu şekilde yorumlamak mümkündür:

Kayıpların Azalması: Hem doğrulama hem de eğitim veri setlerinde kayıp türlerinin azalması modelin eğitim verilerine uyum sağladığını ve nesne algılama görevini öğrenmeyi başardığını göstermektedir.

Performans Ölçütlerinin Artması: Hassasiyet (precision) ve geri çağırma (recall) değerlerinin hem eğitim hem de doğrulama veri setlerinde artması, modelin doğru bir şekilde nesneleri tanımlama ve doğru pozitifleri belirleme konusunda da başarılı olduğunu göstermektedir.

mAP Değerinin Artması: mAP50 ve mAP50-95 değerleri, eğitim ve doğrulama veri setlerinde genel olarak bir artışın gözlemlendiğini ortaya koymaktadır. Bu, modelin farklı sınırlayıcı kutu örtüşme eşiklerinde daha yüksek performans gösterdiğini göstermektedir.

Doğrulama Kaybının Eğitim Kaybından Daha Yüksek Olması: Özellikle "val/box_loss" ve

"val/cls_loss" grafiklerinde, doğrulama kaybı eğitim kaybından daha yüksek bir seviyede seyretmektedir. Bu, modelin eğitim verilerine aşırı uyum sağladığını ve yeni verilere genelleştirilme konusunda zorlandığını gösterebilir. Bu da overfitting (aşırı uyum) sorununun ortaya çıkmasına sebep olabilir. Overfitting modelin eğitim setindeki gürültüyü veya özellikleri öğrenmesi ve gerçek dünya verilerinde genelleştirilemeyen bir model oluşturmaya sebep olabilir.

Doğrulama Performansının Durgunlaşması: Bazı performans ölçütlerinde, özellikle "metrics/precision(B)" ve "metrics/recall(B)" grafiklerinde, belirli bir noktadan sonra doğrulama veri setindeki performansın artmaması, modelin artık öğrenme sürecini sürdürmediğini ve eğitim verilerine daha fazla uyum sağlayamadığını gösterebilir.

3.2.4 Yeni Yaklaşımla Geliştirilen 3 Modelin Sonuçlarının İncelenmesi

Tablo 10. Yeni Yaklaşımla Eğitilen Modellerin Eğitim Performansları

Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
YOLOv8l	288/500	512	8	4	0,88	0,82	0,86	0,53	2.457,00	8,53
YOLOv8l	317/500	512	8	4	0,91	0,88	0,93	0,64	3.606,00	11,38
YOLOv8l	463/500	512	8	4	0,93	0,88	0,91	0,68	6.717,00	14,51

Tablo 10’da gösterilen 4. 5. ve 6. aşamada gerçekleştirilen model eğitimlerinin sonuçları incelendiğinde, YOLOv8l modelinin farklı eğitim süreçlerine maruz kaldığında performansının nasıl değiştiği gözlemlenebilmektedir. Her üç eğitimde de YOLOv8l ön-eğitilmiş modeli ve aynı eğitim parametreleri (imgsz, batch, workers) kullanılmış ancak eğitimde kullanılan döngü sayısı artırılmıştır. Bu da eğitim süresinin artmasına ve dolayısıyla modelin daha fazla veri öğrenmesine olanak sağlamıştır.

Eğitimler arasındaki ilişki şu şekilde özetlenebilir:

Eğitimde geçen döngü sayısı arttıkça model performansı iyileşmiştir: 4. aşamada 288 döngü ile tamamlanan model eğitimi, 5. aşamada 317 ve 6. aşamada 463 döngüde tamamlanmıştır. Bu artış, P, R, mAP50 ve mAP50-95 değerlerinde gözle görülür bir yükselişe neden olmuştur. Örneğin, P değeri 4. aşamada 0,877 iken 6. Aşamada 0,932’ye yükselmiştir. Bu durum, daha fazla veri üzerinde eğitilmiş modelin, verileri daha iyi temsil edebildiğini ve daha doğru tahminler yapabildiğini göstermektedir.

Eğitim süresi artmıştır: Doğal olarak, daha fazla eğitim döngüsüyle ile eğitim süresi de artmıştır. 4. aşamada 2.457 saniye süren eğitim, 6. aşamada 6.717 saniyeye çıkmıştır. Bu da performans artışının bir bedeli olduğunu göstermektedir.

Eğitim süresine göre performans artışı: Her ne kadar eğitim süresi artmış olsa da performans artışı, eğitim süresindeki artışla uyumlu bir şekilde gerçekleşmiştir. Bu durum, modelin verileri daha etkili bir şekilde öğrendiğini ve daha fazla eğitim döngüsüyle ile performansın daha da iyileşebileceğini gösterir.

Sonuç olarak, 4., 5. ve 6. aşamada gerçekleştirilen eğitimlerin sonuçları, daha fazla veri kullanarak eğitimin, YOLOv8l modelinin performansını iyileştirdiğini göstermektedir. Ancak, bu iyileştirmenin bir bedeli vardır: eğitim süresi artmaktadır. İdeal olarak, performansta istenen seviyeye ulaşırken eğitim süresini dengelemek için uygun bir eğitim döngü sayısı belirlemek önemlidir.

3.2.5 YOLOv8 ile YOLOv9 Arasındaki Performans Tablolarının Karşılaştırmalı İncelenmesi

Bu aşamada çalışma başladıktan sonra yayınlanan YOLOv9 ve 4, 5 ve 6. aşamada eğitilen modellerin eğitiminde kullanılan veri setleri kullanarak üç yeni model eğitilerek sonuçları önceki modellerin sonuçlarıyla karşılaştırılmıştır. Tablo 10'da bu modellerin eğitim parametreleri ve performans ölçütleri gösterilmektedir.

Tablo 11. YOLOv9c ile Eğitilen Modellerin Parametre ve Performansları

Aşama	Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
7	YOLOv9c	500/500	512	8	4	0,89	0,79	0,86	0,56	4107	8,21
8	YOLOv9c	500/500	512	8	4	0,98	0,95	0,97	0,73	6573	13,15
9	YOLOv9c	311/500	512	8	4	0,94	0,84	0,93	0,65	3387	6,77

7, 8 ve 9. aşamada gerçekleştirilen model eğitimlerinde YOLOv9c ön-eğitimli model kullanılmıştır. 7. aşama ile 4. aşamada, 8. aşama ile 5. aşamada ve 9. aşama ile 6. aşamada aynı veri setleri ve parametreler kullanılmıştır. 7. ve 8. aşamada eğitilen modeller hedeflenen eğitim döngüsü olan 500'e ulaşırken 9. aşamada eğitilen model 311. döngüde erken durdurulmuştur. Döngü sayısındaki bu değişiklik modellerin eğitim için ihtiyaç duyduğu süreyi belirgin bir şekilde değiştirmiştir.

7. aşamada eğitilen modelin en az görsele sahip veri seti ile eğitilmiş olmasına rağmen 500 eğitim döngüsünü tamamlaması 4.107 saniye sürmüştür. Her bir döngü için gereken süre ortalama 8,21 saniyedir. Model eğitiminin sonucunda elde edilen mAP50 değerinin, % 86 olduğu gözlemlenmiştir.

Tablo 12’de 4. ve 7. aşamada gerçekleştirilen eğitimin parametre ve performans değerleri gösterilmektedir.

Tablo 12. Model Eğitimi 4. Aşama ile 7. Aşama Performans Karşılaştırması

Aşama	Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
4	YOLOv8l	288/500	512	8	4	0,88	0,82	0,86	0,53	2.457	8,53
7	YOLOv9c	500/500	512	8	4	0,89	0,79	0,86	0,56	4107	8,21

4. ve 7. aşamada eğitilen modellerin ikisinin de mAP50 değerleri aynıdır ve bu veri seti üzerinde her iki model de birbirine yakın bir performans sergilemiştir. YOLOv9c ile eğitilen model daha fazla eğitim döngüsüne ihtiyaç duymuş ve dolayısıyla eğitimin tamamlanması daha fazla zaman almıştır.

8. aşamada eğitilen model 7. aşamadan daha büyük ancak 9. aşamadan daha küçük bir veri setiyle eğitilmiş olmasına rağmen her iki eğitimden de daha uzun sürede tamamlanmıştır. Tamamlanması 6.573 saniye sürmüş ve her bir döngü için ortalama 13,15 saniyeye ihtiyaç duymuştur.

Tablo 13’te 5. aşamada YOLOv8l ön-eğitilmiş modeliyle gerçekleştirilen eğitimde kullanılan aynı veri seti kullanılarak YOLOv9c ön-eğitilmiş modeliyle gerçekleştirilen eğitimlerin parametre ve performans değerleri gösterilmektedir.

Tablo 13. Model Eğitimi 5. Aşama ile 8. Aşama Performans Karşılaştırması

Aşama	Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
5	YOLOv8l	317/500	512	8	4	0,91	0,88	0,93	0,64	3.606	11,38
8	YOLOv9c	500/500	512	8	4	0,98	0,95	0,97	0,73	6573	13,15

Tablo 13’te model eğitiminin sonucunda elde edilen mAP50 değerinin, % 97 olduğu gözlemlenmiştir. Bu değer 5. aşamada YOLOv8l modeliyle gerçekleştirilen eğitimden elde edilen sonuçtan %4 daha fazladır. Bu veri seti üzerinde her iki modelden 5. aşamada eğitilen model toplam süre olarak 8. aşamada eğitilen modelin yarısı kadar zaman alsa da 8. aşamada eğitilen modelin performans değerlerinin çok daha yüksek olduğu görülmektedir. YOLOv9c ön-eğitilmiş modeli ile eğitilen modelin daha fazla zaman almasının sebebi daha fazla eğitim döngüsüne ihtiyaç duymuş olmasıdır.

9. aşamada eğitilen model, en büyük veri setiyle eğitilen model olmasına rağmen en kısa sürede tamamlanan model olmuştur. Bunun sebebi 311. döngüde erken durdurulmuş olmasıdır. Tamamlanması 3.387saniye sürmüştür. Her bir döngü ortalama 6,77 saniyede tamamlanmıştır. Bu model, daha büyük bir veri setiyle eğitilmiş olmasına rağmen 8. aşamada eğitilen modelden daha düşük mAP50 değerine sahiptir. Bu düşüşün sebebi eğitilen modelin veri setine aşırı uyum sağlaması (overfitting) olabilir.

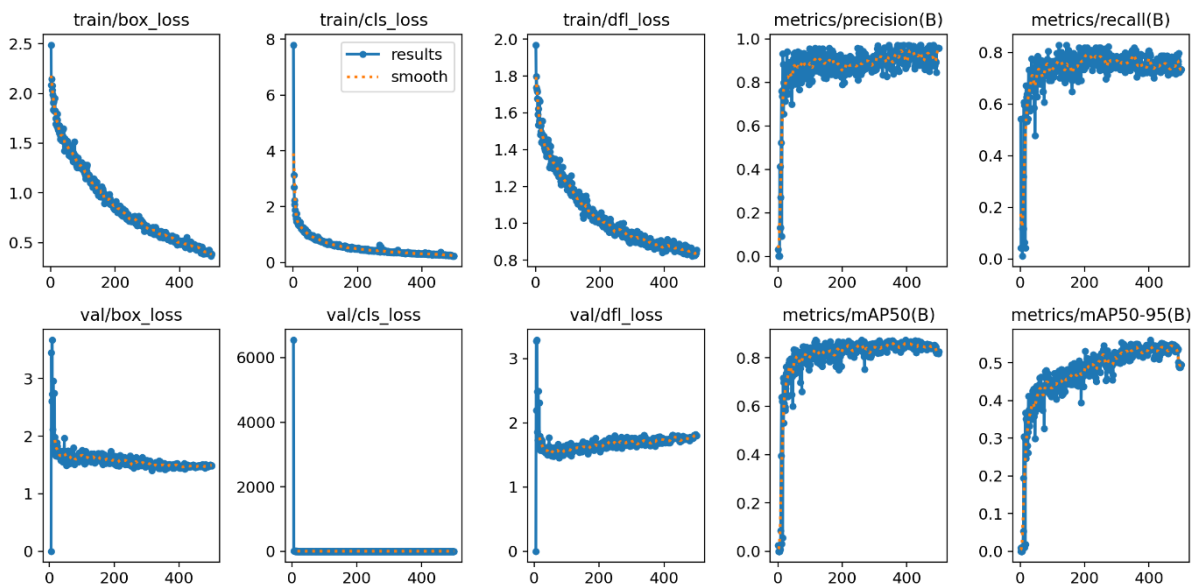
Tablo 14. Model Eğitimi 6. Aşama ile 9. Aşama Performans Karşılaştırması

Aşama	Model	Epoch	imgsz	batch	workers	P	R	mAP50	mAP50-95	Toplam Süre	Epoch Süresi
6	YOLOv8l	463/500	512	8	4	0,93	0,88	0,91	0,68	6.717	14,51
9	YOLOv9c	311/500	512	8	4	0,94	0,84	0,93	0,65	3387	6,77

Tablo 14 incelendiğinde aynı veri setiyle eğitilmiş olmalarına rağmen YOLOv8l ön-eğitilmiş modeliyle gerçekleştirilen eğitimin, YOLOv9c ön-eğitilmiş modeliyle gerçekleştirilenden daha uzun sürdüğü ve %2 daha iyi bir performans sergilediği görülmektedir. Bunun sebebi 9. aşamada gerçekleştirilen eğitimin erken durdurulması olabilir.

3.2.6 YOLOv8 ile YOLOv9 ile Eğitilen Modellerin Performans Grafiklerinin Karşılaştırmalı İncelemesi

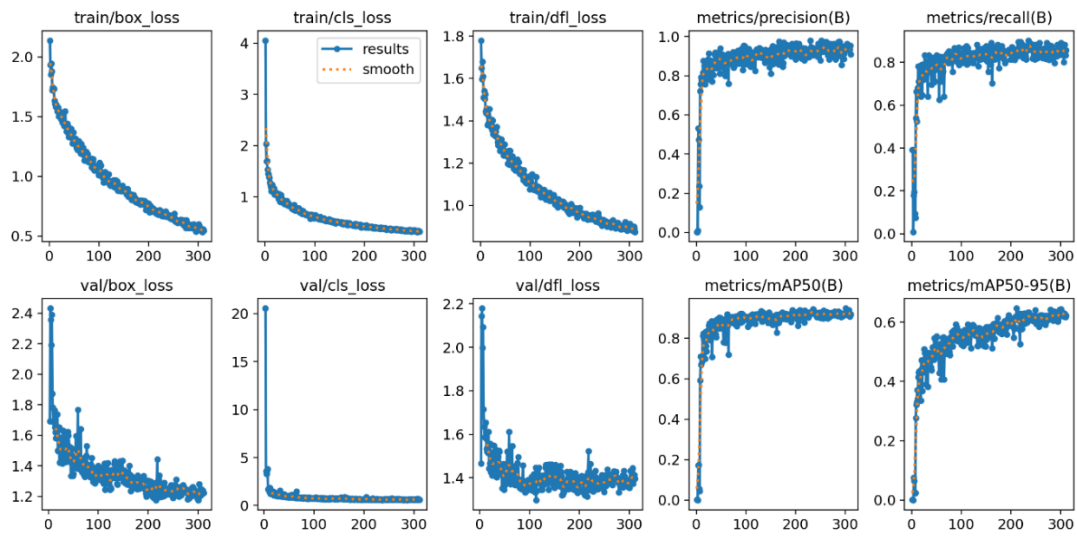
Şekil 21’de 7. aşamada gerçekleştirilen eğitimin performans grafikleri gösterilmektedir.



Şekil 21. Model Eğitimi 7. Aşama Performans Grafikleri

4. ve 7. aşamada gerçekleştirilen eğitimlerin performans grafikleri (şekil 16 ve 21) karşılaştırıldığında, 7. aşamada eğitilen modelin genel olarak daha iyi bir performansa sahip olduğu görülmektedir. Bu aşamada eğitimin, train/box_loss, train/cls_loss, ve train/df_l_loss kayıpları 4. aşamada eğitilen modelin kayıplarının altına inmiştir. Bu da modelin öğrenme performansının daha iyi olduğunu göstermektedir. val/box_loss ve val/cls_loss gibi doğrulama kayıpları incelendiğinde, iki modelin de benzer dalgalanmalar göstermelerine rağmen 7. aşamada bu dalgalanmaların daha az olduğu gözlenmiştir. Performans metrikleri incelendiğinde, kesinlik ve geri çağırma değerlerinin her iki aşamada da %80 civarında olduğu görülmektedir. Ancak, mAP50 değerleri farklılık göstermektedir ve 7. aşamada eğitilen model daha az dalgalanmıştır. Grafiklerin tamamı incelendiğinde 7. aşamada eğitilen modelin daha iyi bir öğrenme sonucunda daha iyi genellemeler yapabileceği sonucuna ulaşılmıştır.

Şekil 22’de 7. aşamada gerçekleştirilen eğitimin performans grafikleri gösterilmektedir.

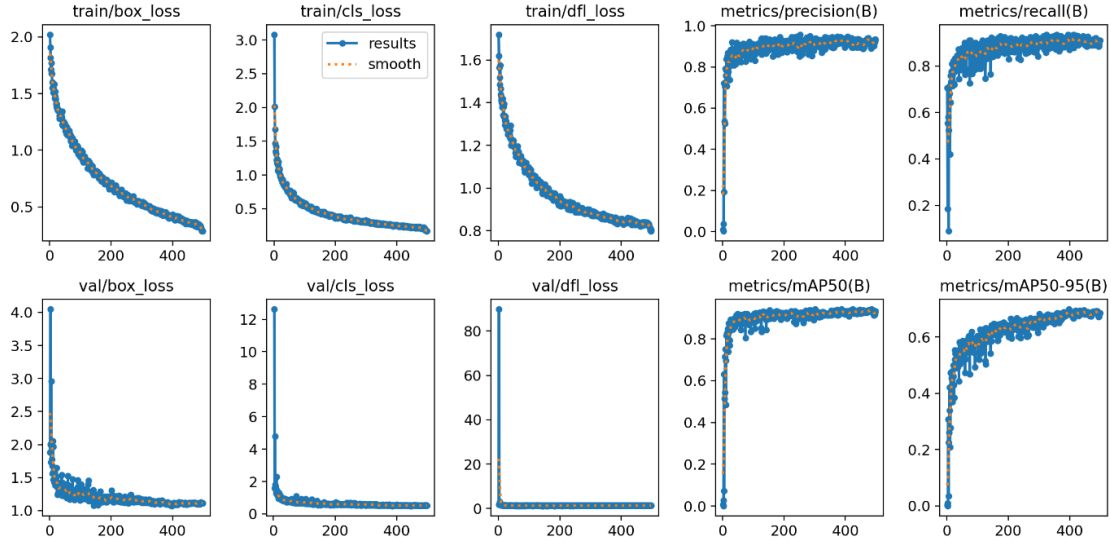


Şekil 22. Model Eğitimi 8. Aşama Performans Grafikleri

5. ve 8. aşamada gerçekleştirilen eğitimlerin performans grafikleri (şekil 18 ve 22) karşılaştırıldığında, 5. aşamada eğitilen modelin eğitim verilerine göre daha iyi performans sergilediğini ancak doğrulama kayıplarının nispeten daha dalgalı olduğunu söylemek mümkündür. Bu modelin aşırı öğrenme problemi yaşıyor olabileceğine işaret edebilir. Ancak 8. aşamada eğitilen modelin başarı grafikleri bu modelin daha fazla eğitim döngüsüyle daha düşük eğitim ve doğrulama kayıplarına sahip olduğunu, dolayısıyla aşırı öğrenme yaşama olasılığının daha düşük olduğunu, kesinlik (precision) ve hassaslık (recall) değerlerinin daha yüksek olduğunu göstermektedir. Genel olarak 8. aşamada eğitilen

modelin, 5. aşamada eğitilen modelden daha iyi sonuçlar üretebildiği gözlemlenmektedir.

Şekil 23'te 6. ve 7. aşamada gerçekleştirilen model eğitimlerinin performans grafikleri gösterilmektedir.



Şekil 23. Model Eğitimi 9. Aşama Performans Grafikleri

6. ve 9. eğitim aşamasında gerçekleştirilen eğitimlerin performans grafikleri (şekil 20 ve 23) incelendiğinde şu sonuçlara ulaşılmıştır:

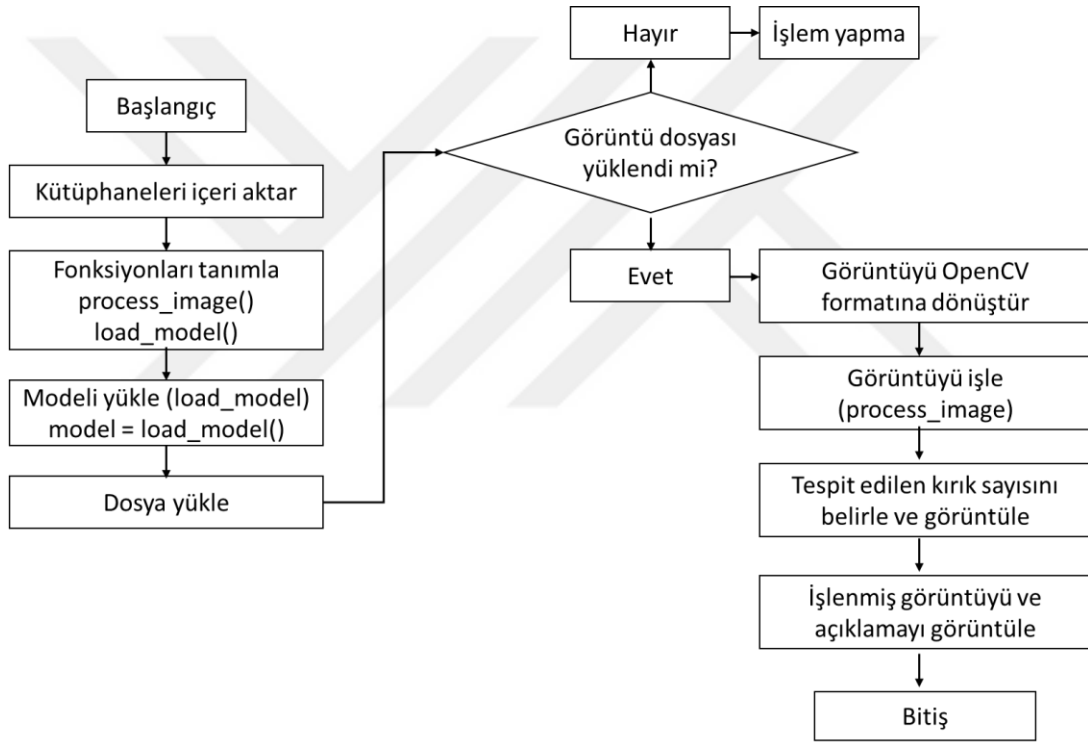
9. aşamada eğitilen modelin, 6. aşamada eğitilen modelden daha iyi performans gösterdiği gözlemlenmiştir. 9. aşamada eğitilen model hem eğitim veri seti üzerinde hem de doğrulama veri setinde, sınıflandırmada ve sınırlayıcı kutu tespit etmede daha düşük kayıplara sahiptir. mAP50 ve mAP50-95 metrikleri de 9. aşamada eğitilen modelin performansının 6. aşamada eğitilen modelden daha başarılı olduğu görülmektedir. Bu da 9. aşamadaki modelin daha iyi genellemeler yapabildiğini göstermektedir. Bu sonuçlar doğrultusunda 9. aşamada eğitilen modelin daha iyi sonuçlar ürettiği ve hedef görev için daha uygun bir model olduğu sonucuna ulaşılmaktadır.

DÖRDÜNCÜ BÖLÜM

UYGULAMA GELİŞTİRME

4.1. Uygulama Akış Diyagramı

Şekil 24’te kemik kırıklarının tespit etmek için 9. aşamada eğitilen ve en yüksek başarıya sahip olan model kullanılarak geliştirilen uygulamanın akış diyagramı gösterilmektedir.



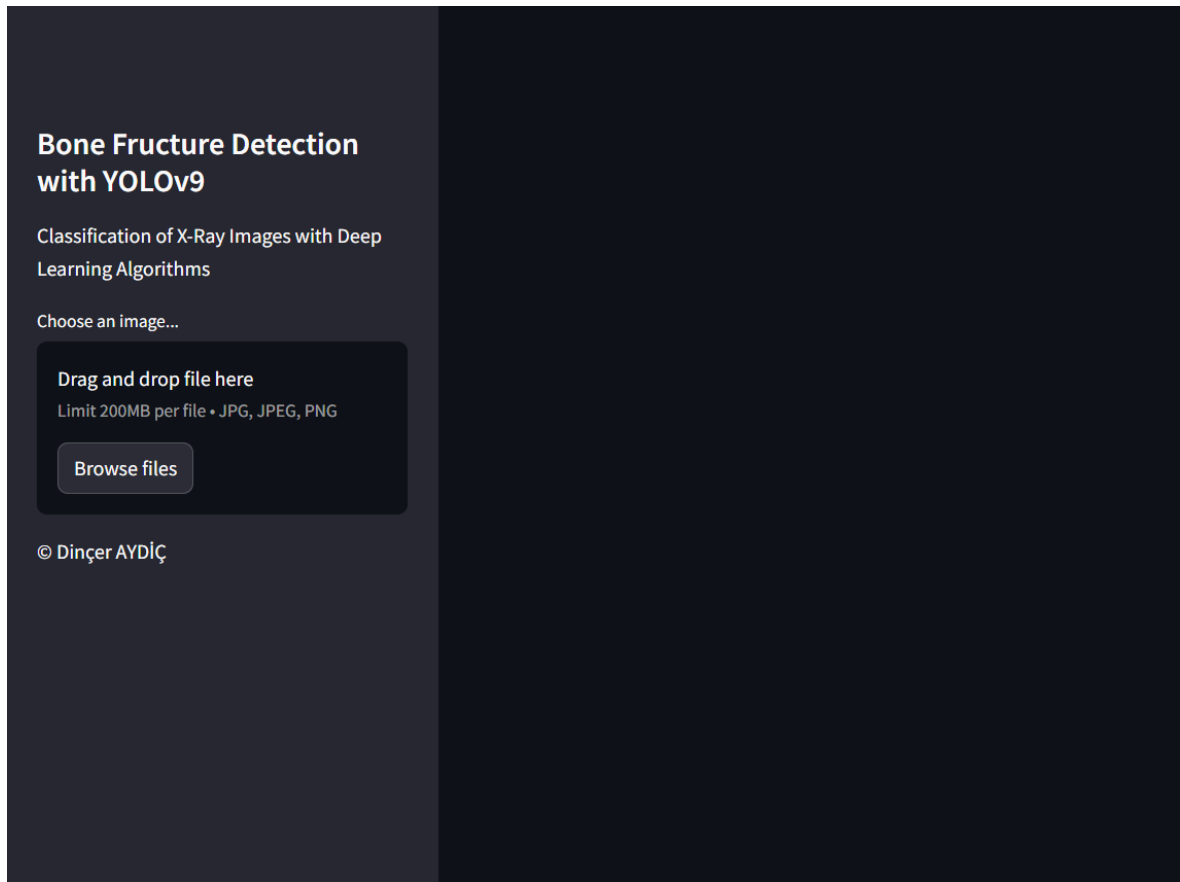
Şekil 24. Uygulama Akış Diyagramı

Şekil 24’te akış diyagramını gösterilen program başlatıldığı zaman öncelikle gerekli (streamlit, opencv, numpy) kütüphaneler içeri aktarılmaktadır. Daha sonra görsellerin işleneceği process_image ve modelin yükleneceği load_model fonksiyonları oluşturulup load_model fonksiyonu çağırılarak model yüklenmektedir. Sonraki aşamada “Browse Files” butonu tıklanarak kemik kırığı içerdiği düşünülen bir görsel yüklenmektedir. Yüklenen görsel OpenCV formatına dönüştürüldükten sonra process_image fonksiyonu çağırılıp işlenerek ve barındırdığı kırık sayısı ile kırıkların konumları tespit edilmektedir. Son olarak tespit edilen kırıklar ve konumları görsel üzerine işaretlenerek kullanıcıya gösterilmektedir.

4.2. Kullanılan Teknolojiler

Python programlama dili kullanılarak oluşturulan uygulamada, streamlit, opencv, numpy gibi kütüphaneler kullanılmıştır. Uygulama şu anda streamlit cloud üzerinde ücretsiz bir şekilde kullanıma sunulmuştur ve **kemik-kirigi.streamlit.app** adresinden ulaşılabilir. Uygulama arayüzü üniversal kullanıma sahip olması açısından İngilizce olarak tasarlanmıştır.

4.3. Uygulama Arayüzü

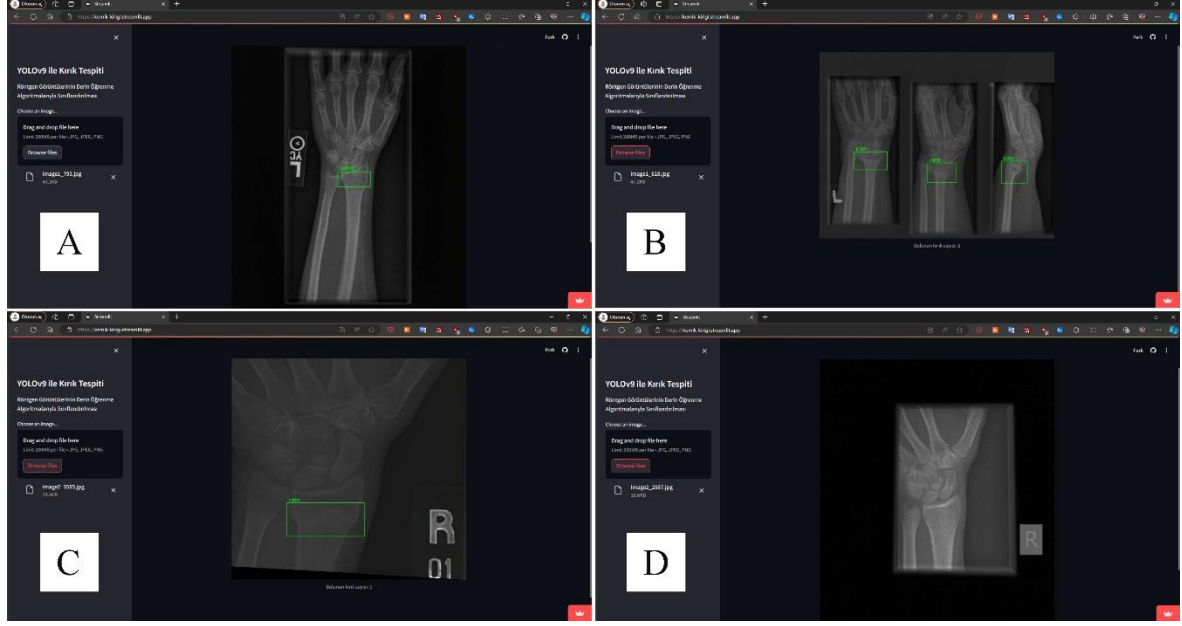


Şekil 25. Uygulama Arayüzü

Oldukça sade bir arayüze sahip olan uygulamayı kullanmak için yapılması gereken sol bölümdeki “Browse Files” butonuna tıklayıp bir bilek röntgen görüntüsü seçmektir. Görüntü seçildikten sonra uygulama otomatik olarak görüntüyü işleyecek ve tahminlerini gösterecektir.

4.3.1 Örnek Görseller

Şekil 26’te görüldüğü üzere, A, B ve C ile gösterilen ekran görüntülerinde uygulama verilen görüntülerde kırıklar tespit etmiş ve bunları yeşil bir kutu içinde ve üst tarafına “KIRIK” etiketi ekleyerek göstermiştir. D ile gösterilen ekran görüntüsünde ise verilen röntgen filminde kırığı tespit etmede başarısız olmuştur. Veri seti büyütülerek ve daha fazla eğitim döngüsüyle sonuçların iyileştirilmesi mümkündür.



Şekil 26. Uygulamanın Yaptığı Tahmin Örnekleri

SONUÇ

Bu çalışmada, sağlık sisteminin yeterli olmadığı yerlerde ya da hasta yoğunluğunun fazla olduğu acil servislerde kullanılması amacıyla kemik kırıklarının röntgen görüntüleri kullanılarak tespit edilmesi ve kırık varsa hangi kemiğin neresinde olduğunu gösterecek bir uygulama geliştirilmiştir. Literatür taraması bölümünde, bu konuda yapılmış önceki çalışmalar incelenmiştir. Yapay zeka araçlarının incelendiği 2. bölümde CNN, R-CNN, Faster R-CNN ve YOLO gibi öne çıkan derin öğrenme algoritmaları incelenmiştir. Eğitim modelinin geliştirildiği 3. bölümde model geliştirme süreci aşamalar halinde açıklanmış, tablo ve grafiklerle her aşamada elde edilen modellerle ilgili açıklamalar yapılmış, sonuçlar yorumlanmıştır.

Model eğitim sürecinin ilk aşamasında geliştirilen 4 modelden, YOLOv8s ön-eğitilmiş modeli ile geliştirilen modelden en iyi mAP50 değeri olan %66 değeri elde edilmiştir. Eğitilen modeller kullanılarak sınıflama ve nesne tespiti algoritmaları ile kendi kendine öğrenen recursive bir yaklaşımla yeni veri setleri oluşturulmuş ve bu veri setleriyle yeni modeller eğitilmiştir. Geliştirilen bu yaklaşım sayesinde YOLOv8l ve YOLOv9c ön-eğitilmiş modeliyle gerçekleştirilen eğitimde bu değer %97 seviyesine ulaşmıştır.

Bu çalışmada elde edilen başarının yeni akademik çalışmalarla, yeni geliştirilen algoritmalar kullanılarak ve daha fazla röntgen filmi etiketlenerek daha büyük veri setleri yardımıyla daha da iyi seviyelere getirilebileceği düşünülmektedir. Bu çalışmada, bu konu üzerinde çalışan akademisyenlerin doğru adımda doğru parametrelerle daha iyi ve hızlı işlem yapabilmelerine yardımcı olmak amacıyla tüm evreler adım adım gösterilmeye çalışılmıştır.

Çalışmanın sonunda elde ettiğimiz istatistiki veriler, yapay zeka araçlarının sağlık sisteminde, özellikle röntgen görüntülerinin işlenerek kırık tespiti yapma gibi görevlerde başarılı sonuçlar üretebildiğini göstermiştir. Geliştirdiğimiz uygulamada başarısı en yüksek olan model kullanılmış ve uygulama <https://kemik-kirigi.streamlit.app> adresinde kullanıma sunulmuştur. Gelecekte programın da kullanılmasıyla birlikte elde edilecek yeni veriler de sisteme dahil edilirse hem örneklem sayısı hem doğruluk oranının daha da artacağı değerlendirilmektedir.

KAYNAKÇA

- Banga, D., & Waiganjo, P. (2019). Abnormality Detection in Musculoskeletal Radiographs with Convolutional Neural Networks(Ensembles) and Performance Optimization. arXiv preprint. <https://doi.org/10.48550/arXiv.1908.02170>
- Brownlee, J. (2019). Deep learning for computer vision: image classification, object detection, and face recognition in python. Machine Learning Mastery.
- Caelen, O. (2017). A Bayesian interpretation of the confusion matrix. *Annals of Mathematics and Artificial Intelligence*, s. 429-450. <https://doi.org/10.1007/s10472-017-9564-8>
- Davletshina, F., Melnychuk, V., Tran, V., Singla, H., Berrendorf, M., Faerman, E., . . . Schubert, M. (2020). Unsupervised anomaly detection for X-ray images. arXiv preprint. <https://doi.org/10.48550/arXiv.2001.10883>
- Farabet, C., Martini, B., Akselrod, P., Talay, S., LeCun, Y., & Culurciello, E. (2010). Hardware accelerated convolutional neural networks for synthetic vision systems. In *Proceedings of 2010 IEEE international symposium on circuits and systems* (s. 257-260). IEEE.
- Géron, A. (2022). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. O'Reilly Media, Inc.
- Girshick, R. (2015). Fast r-cnn. *Proceedings of the IEEE international conference on computer vision*, (s. 1440-1448).
- 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*, (s. 580-587).
- Haykin, S. (1999). *Neural Networks: A Comprehensive Foundation* (2 b.). Prentice Hall.
- Jeon, Y. D., Kang, M. J., Kuh, S. U., Cha, H. Y., Kim, M. S., You, J. Y., . . . Yoon, D. K. (2023). Deep Learning Model Based on You Only Look Once Algorithm for Detection and Visualization of Fracture Areas in Three-Dimensional Skeletal Images. *Diagnostics*, 14(1), 11. <https://doi.org/10.3390/diagnostics14010011>

- Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2022). A Review of Yolo Algorithm Developments. *Procedia computer science*, 199, 1066-1073. <https://doi.org/10.1016/j.procs.2022.01.135>
- Khatik, I., & Kadam, S. (2022). A Systematic Review of Bone Fracture Detection Models using Convolutional Neural Network Approach. *Journal of Pharmaceutical Negative Results*, 153-158. <https://doi.org/10.47750/pnr.2022.13.S09.019>
- O'Shea, K., & Nash, R. (2015). An introduction to convolutional neural networks. *arXiv preprint*. <https://doi.org/10.48550/arXiv.1511.08458>
- Pingale, S., Hundalekar, A., Naidu, V., & Shirsath, V. (2024). Intelligent Monitoring for Anomaly Recognition using CNN and YOLOv9. *Research Square*. <https://doi.org/10.21203/rs.3.rs-4367910/v1>
- Powers, D. M. (2020). Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint*. <https://doi.org/10.48550/arXiv.2010.16061>
- Prabhu, B., Lakshmi, R., Ankitha, R., Prateeksha, M., & Priya, N. (2022). RescueNet: YOLO-based object detection model for detection and counting of flood survivors. *Modeling Earth Systems and Environment*, 8(4), 4509-4516. <https://doi.org/10.1007/s40808-022-01414-6>
- Qamar, B., Jawaaid, M., & Memon, M. (2021). Computer Assisted Solution for bone Anomaly detection in X-Ray images. In *Proc. 2nd Int. Conf. Comput. Sci. Technol*, (s. 21-26).
- Rajpurkar, P., Irvin, J., Bagul, A., Ding, D., Duan, T., Mehta, H., . . . Ng, A. (2017). Mura: Large dataset for abnormality detection in musculoskeletal radiographs. *arXiv preprint*. <https://doi.org/10.48550/arXiv.1712.06957>
- Reis, D., Kupec, J., Hong, J., & Daoudi, A. (2023). Real-Time Flying Object Detection with YOLOv8. *arXiv preprint*. <https://doi.org/10.48550/arXiv.2305.09972>
- 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. <https://doi.org/10.1109/TPAMI.2016.2577031>

- Sharma, S., & Guleria, K. (2024). A systematic literature review on deep learning approaches for pneumonia detection using chest X-ray images. *Multimedia Tools and Applications*, 83, 24101-24151. <https://doi.org/10.1007/s11042-023-16419-1>
- Tanzi, L., VezzettiE., Moreno, R., & MoosS. (2020). X-ray bone fracture classification using deep learning: a baseline for designing a reliable approach. *Applied Sciences*, 10(4), 1507. <https://doi.org/10.3390/app10041507>
- Uysal, F., Hardalaç, F., Peker, O., Tolunay, T., & Tokgöz, N. (2021). Classification of shoulder x-ray images with deep learning ensemble models. *Applied Sciences*, 11(6), 2723. <https://doi.org/10.3390/app11062723>
- Verma, A., Kumar, V., & Yadav, R. (2024). Humerus Bone Fracture Detection Utilizing YOLOv4 Algorithm: A Deep Learning Approach. In 2024 2nd International Conference on Disruptive Technologies (ICDT) (s. 1191-1196). IEEE.
- Wang, C. Y., Yeh, I. H., & Liao, H. (2024). YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv preprint*. <https://doi.org/10.48550/arXiv.2402.13616>
- Wu, J. (2017). Introduction to convolutional neural networks. National Key Lab for Novel Software Technology. Nanjing University. China.
- Zou, J., Han, Y., & So, S. S. (2009). Overview of artificial neural networks. *Artificial neural networks: methods and applications* (s. 14-22). içinde Springer. <https://doi.org/10.1007/978-1-60327-101-1>

AKSARAY ÜNİVERSİTESİ TEZ DEĞERLENDİRME FORMU		
Öğrencinin Adı Soyadı:		EVET
Kapak		
1	Tez Başlığı tutanaktaki başlıkla aynı mı?	
2	Kapaktaki ay ve yıl savunmaya girilen tarihle tutarlı mı?	
3	Kapak format kılavuzdaki kapak formatına uygun mu?	
4	Kapakta yazılan tüm yazılar doğru olarak verilmiş mi?	
İçindekiler		
5	Sayfa numaraları tam verilmiş mi?	
6	Şekil, Çizelge vb. listeleri verilmiş mi? Sıralaması doğru mu?	
7	Özet, Abstract, Giriş, Sonuçlar vb. bölümler var mı?	
8	Yazım hataları kontrolü yapıldı mı?	
Giriş		
9	Hazırlanan tezin önemini anlatıyor mu?	
10	İkinci ve Üçüncü dereceden başlık içermemeli kuralına uyuldu mu?	
Özet/Abstract		
11	Kılavuza uygun mu?	
12	Ay ve yıl savunmaya girilen tarihle tutarlı mı?	
13	Özet; tek sayfa, tek aralık, tek paragraf kuralına uygun olarak yazıldı mı?	
14	Bilim kodu, sayfa adedi, anahtar kelimeler ve tez danışmanı yazıldı mı?	
Kaynakça		
15	Kaynakların tamamına metin içinde atıf yapıldı mı?	
16	Kaynak formatı kılavuzdaki kaynak formatına uygun olarak hazırlanmış mı?	
17	Atıf formatı kılavuzdaki atıf formatına uygun mu?	
Atıf Yöntemi APA 7 ☐ CMS ☐ İSNAD ☐		
Genel Değerlendirme		
18	Etik Beyan açıklaması okundu, uyuldu ve imzalandı mı?	
19	Kabul/Onay sayfası kılavuzdaki formata uygun olarak düzenlenmiş mi?	
20	Kabul /Onay sayfasında belirtilen oy birliği/oy çokluğu seçeneklerinden uygun olanı savunmayla tutarlı olacak şekilde belirlenmiş mi?	
21	Sayfa kenar boşlukları ve sayfa numaraları kılavuzdaki formatına uygun mu?	
22	Paragraf boşlukları ve metin satır aralığı kılavuza uygun olacak şekilde düzenlenmiş mi?	
23	Başlık yazımları kılavuzdaki başlık formatlarına uygun mu?	
24	Yazı tipi ve boyutu kılavuzdaki yazı tipi ve boyutu formatına uygun mu?	
25	Şekil, Çizelge vb. açıklama ve numaralandırmaları kılavuzdaki formata uygun olarak yazılmış mı?	

Dr. Öğr. Üyesi Tarık YILMAZ

İmza