**YOLOV8 MODELİ İLE KEMİK KIRIKLARININ TESPİTİ**

#### Nesibe GÜL

**Doç. Dr. Ali Seydi Keçeli**

**Proje Danışmanı**

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

Bilişim Sistemleri Ana Bilim Dalı

Veri ve Bilgi Mühendisliği Tezsiz Yüksek Lisans için Öngördüğü DÖNEM PROJESİ olarak hazırlanmıştır.

Haziran, 2023

**YOLOV8 MODELİ İLE KEMİK KIRIKLARININ TESPİTİ**

#### Nesibe GÜL

**Doç. Dr. Ali Seydi Keçeli**

**Proje Danışmanı**

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

Bilişim Sistemleri Ana Bilim Dalı

Veri ve Bilgi Mühendisliği Tezsiz Yüksek Lisans için Öngördüğü DÖNEM PROJESİ olarak hazırlanmıştır.

Haziran, 2023

Nesibe GÜL’ ün hazırladığı **“Yolov8 Modeli ile Kemik Kırıklarının Tespiti”** adlı bu çalışma **Doç. Dr. Ali Seydi Keçeli** tarafından **Bilişim Sistemleri Ana Bilim Dalı Veri ve Bilgi Mühendisliği Programı**’nda Dönem Projesi olarak kabul edilmiştir.

Doç. Dr. Ali Seydi Keçeli

Proje Danışmanı

# BİLDİRİM

Hacettepe Üniversitesi Bilişim Enstitüsü, dönem projesi yazım kurallarına uygun olarak hazırladığım bu tez çalışmasında,

* dönem projesi içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
* görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
* başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
* atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
* kullanılan verilerde herhangi bir tahrifat yapmadığımı,

ve bu dönem projesinin herhangi bir bölümünü bu üniversitede veya başka bir üniversitede başka bir tez/dönem projesi çalışması olarak sunmadığımı beyan ederim.

14.06.2023

Nesibe GÜL

# ÖZET

**Yolov8 Modeli ile Kemik Kırıklarının Tespiti**

**Nesibe GÜL**

**Haziran, 2023, 19 Sayfa**

**Tez Özeti:** Bu çalışmada, kemik kırıklarının hızlı tespiti için nesne tespiti modelleri ile ilgili örnek bir uygulama sunulmuştur. YOLO serisinin v8 modeli kemik kırıklarının tespiti için kullanılmıştır. Daha önce yapılan araştırmalar incelenmiş literatür taraması yapılmıştır. Model adam ve sgd optimizasyon yöntemleri ile eğitilmiştir. Çalışmada kullanılan görüntüler roboflow adlı siteden alınmış olup eğitim, doğrulama ve test verisi olarak 3 kısma ayrılmıştır. Ayrıca çıktıların kullanıcı dostu bir şekilde görüntülenebilmesi için streamlit adlı grafik kullanıcı arayüzünde sunulmuştur.

**Anahtar Kelimeler**

Yolov8, Kemik Kırığı, Nesne Tespiti

# İÇİNDEKİLER

KABUL VE ONAY…………………………………………………………………………i

[BİLDİRİM iii](#_Toc137155944)

[ÖZET iv](#_Toc137155945)

[İÇİNDEKİLER v](#_Toc137155946)

[1. GİRİŞ 1](#_Toc137155947)

[2. İLGİLİ ÇALIŞMALAR 3](#_Toc137155948)

[3. YÖNTEM 5](#_Toc137155949)

[3.1 YOLO Nasıl Çalışır: 7](#_Toc137155950)

[3.2 YOLO Modelinin Mimarisi 7](#_Toc137155951)

[3.3 YOLO Modelinin Zayıf Yönleri 8](#_Toc137155952)

[3.4. YOLO Modelinin Diğer Modellerle Karşılaştırılması 9](#_Toc137155953)

[3.5. YOLOv8 Modelindeki Yenilikler 9](#_Toc137155954)

[3.6. Modelin Performans Değerlendirmesi: 10](#_Toc137155955)

[3.7 Veri Kümesi Tanıtımı 11](#_Toc137155956)

[3.8 Optimizasyon Ve Görüntü Boyutunu Büyütme 12](#_Toc137155957)

[4.SONUÇ 13](#_Toc137155958)

[4. TARTIŞMA 17](#_Toc137155959)

[5. KAYNAKÇA 18](#_Toc137155960)

# GİRİŞ

Nesne tespiti görüntü tabanlı yazılım sistemleri için kritik bir bileşen olup birçok sahada kullanılabilir. Nesne tespiti, görüntüdeki bir nesnenin ve yer tespitinin beraber yapılmasıdır [10].

Nesneleri gerçek zamanlı olarak algılamak zorlu bir iştir. Gerçek dünyadaki görüntülerde gürültü, bulanıklık ve titreşim vb. problemler nesne algılamada önemli bir engel teşkil etmektedir. Nesne algılama, bilgisayarlı görü, otomatik araçlar, endüstriyel otomasyon vb. alanlarda önemli bir rol oynamaktadır. [9].

Nesne algılayıcıları oluşturmanın iki yöntemi vardır -tek adımlı yaklaşım ve iki adımlı yaklaşım. Derin öğrenme yöntemleri, ilk olarak bölge öneri ağları oluşturuyor ve ardından bunları sınıflandırdığı bölge önerisi nesne algılama algoritmalarını içeriyordu. Ardından geliştirilen tek adımlı yaklaşım daha hızlı ve daha iyi bir bellek verimliliği göstermektedir. Tek adımlı yaklaşım, görüntülerdeki nesneleri konumlarıyla birlikte tek bir adımda sınıflandırır. İki adımlı yaklaşımda ise ilk adımda, görüntüde nesne olma olasılığı yüksek olan bir dizi bölge oluşturur. Daha sonra da, bu bölgeleri girdi olarak alarak nesnelerin son tespitini ve sınıflandırmasını gerçekleştirir. Bu iki adım sırasıyla Bölge Teklifi Adımı ve Nesne Tespiti Adımı olarak adlandırılır [10].

Tek adımlı yaklaşım, sınıf olasılıklarını ve nesne konumlarını doğrudan tahmin etmek için bu iki adımı birleştirir. SSD ve YOLO modeli tek adımlı yaklaşım temelli, Fast R-CNN, Faster R-CNN modeli iki adımlı yaklaşım temelli modellere örnektir [10].

Medikal alanda kemik kırıklarının hızlı tespiti özellikle acil tıp departmanlarındaki zaman ve personel kısıtları sebebiyle zorlaşmaktadır. Pediatrik bir operasyona karar vermeden önce farklı uzmanlıkların bir araya gelerek kırıkları tespit etmesi gerekmektedir. Bu esnada onlara yardımcı olacak bir teknolojinin bilgisayarlı görü alanında görüntülerin modellenmesinden gelmesi, modellerin doğruluk oranlarının artmasıyla artık daha olası bir hale gelmiştir.

Bu çalışmada tek adımlı nesne tespiti örneği olan YOLO serisinin v8 modeli ile kemik kırıklarının tespitinde örnek bir uygulama yapılmıştır. Bu amaçla model adam ve sgd optimizasyon yöntemleri ile eğitilmiştir. Çalışmada kullanılan görüntüler roboflow[22] adlı siteden alınmış olup eğitim, doğrulama ve test verisi olarak 3 kısma ayrılmıştır. Ayrıca çıktıların, kullanıcı dostu bir şekilde görüntülenebilmesi için streamlit grafik kullanıcı ara yüzünde sunulmuştur.

# İLGİLİ ÇALIŞMALAR

Inoue ve arkadaşları tarafından yapılan bir çalışmada [15] , araştırmacılar pelvik, kaburga ve omurga kırıklarının otomatik lokalizayonunun ve sınıflandırılmasının Faster R-CNN modeli kullanılarak yapılıp yapılamayacağını araştırdılar. Ayrıca bu kırık tespit algoritmasının uzmanlara kırık teşhisinde yardımcı olup olamayacağını da incelediler. 200 hastanın 7664 tane tam vücut bilgisayarlı tomografi aksiyel kesitleri (göğüs, karın, pelvis) kullanıldı. Faster R-CNN modeli kullanılarak teşhis süresinin istatistiksel olarak önemli oranda azaldığı, özellikle yoğun servislerde triaj sisteminde yararlanabileceği belirtilmiştir. Ayrıca modelle beraber kırık tespit etme hassasiyetinin de önemli oranda iyileştirme gösterdiği belirtilmiştir.

Yangling ve arkadaşları tarafından yapılan bir çalışmada [16] Faster R-CNN yardımıyla incelenen kemiğin türü tespit edilmiştir. Bunun için de insan vücudunda bulunan kemikler 20 ayrı çeşide ayrılmış, ve bu bölgeler tespit edildikten sonra kırık olup olmadığı farklı yöntemler kullanılarak tespit edilmeye çalışılmıştır. Öncelikle görüntü işlemeye ait farklı filtreler uygulanmış ardından ResNet yöntemi ile tahminde bulunulmuştur. Bu yöntemin aşırı öğrenme olasılığı ve birçok deneme ve yanılma ile parametrelerinin bulunması sebebiyle, Gabor Evrişimli Nöral Ağ ve ardından klasik evrişimli ağın birleşmesinden oluşan CrackNet yöntemi en son olarak kullanılmıştır. Sonuç olarak Faster R-CNN + CrackNet sistemi ile % 88 doğruluk, % 87.5 duyarlılık, % 89 kesinlik oranları bulunmuştur.

Yee Liang ve arkadaşları tarafından yapılan çalışmada [17] bilek kemiği kırık tespiti ve lokalizasyonu için derin öğrenme yöntemlerinden Faster R-CNN yöntemi kullanılmıştır. Toplamda 7356 radyografi % 90 ve % 10 oranında eğitim ve test verisi olarak kullanılmıştır. Model kırıkların yerini % 90 üzerinde bir doğrulukla tespit etmiş ve sonuç olarak % 95.7 ve % 82.5 oranlarında duyarlılık ve hassasiyet skorları ele edilmiştir.

Medikal alanda görüntüleme tekniklerinin dar boğazlarda, özellikle doktor ve zaman kısıtlarında büyük önem taşıdığının belirtildiği bir diğer makalede [18] göğüs kemiklerine ait kırıklarının tespitinde YOLO modelinin v3 serisi kullanılmıştır. Görüntülerin 1024\*1024 boyutlarında kullanılması görüntüde belirsizliklere ve teknik kapasitede problemlere yol açtığı belirtilen makalede sonuçların başarılı olamadığı mAP50 skorunun %11.8 de kaldığı görülmüştür.

Rui ve Weiming tarafından yapılan bir çalışmada [19] ameliyat öncesi radyolojist ve cerrahın beraber kemik kırıkları hakkında değerlendirme yapma safhasının kısaltılması amacıyla bilgisayarlı görü modellerinden yararlanmasının önemli olduğu belirtilmiştir. Yolov8 modeliyle eğitilen farklı bölgelere ait kemik kırık tespitinde elde edilen mAP skorunun veri çoğaltma ile % 60.4’ten % 61.2’ye çıktığı tespit edilmiştir. Ayrıca görüntülerin sadece boyutlarının 640’tan 1024’e çıkarmanın performansı % 62.5’ya çıkardığı, veri çoğaltma ile beraber sonucun % 63.1 olduğu belirtilmiştir.

Nattaphon ve arkadaşları tarafından kalça kırık tespitinde yolov4-tiny modelini kullanan makalede [20] % 96.2 oranında duyarlılık skoru, % 94.6 oranında ise kesinlik elde edilmiştir. Aynı deneyde doktorların duyarlılık performansının, tecrübelerine bağlı olarak % 69.2’den % 96.2’ye kadar değiştiği belirtilmiştir.

Fatih ve arkadaşları tarafından yapılan bir başka çalışmada ise [21] başlangıçta ResNet, DenseNet, VGG, InceptionV3, MobileNetV2 kullanılmıştır. Ardından bu modellerin beraber kullanılarak elde edilen iki farklı ensemble model ile % 84.5 ve % 84.7 oranında doğruluk oranları elde edilmiştir.

# YÖNTEM

Başlangıçta birçok nesne tespit modelleri sınıflandırma modellerinin nesne algılama modellerine uyarlanmasından ibaretti. Nesnenin algılanması için görüntü üzerinde farklı bölgelerde model çalıştırılırdı. Deforme parça modelleri gibi sistemlerde kayan pencere adını verdiğimiz bu sistemler oldukça yavaş ve gerçek zamanlı olmaktan uzaktı [5]. Daha sonra R-CNN gibi bölge önerme yöntemine dayanan modeller ile önce potansiyel sınırlayıcı kutular oluşturulur ve sınıflandırıcıları bu bölgelerde çalıştırılırdı. Ardından çoklu tespit edilen nesnelerin tekleştirilmesi, sınırlandırıcı kutuların sınırlarının iyileştirilmesi için işlemler yapılırdı. Tüm bu işlemler yavaş ve optimize edilmesi zordu çünkü her bir parça ayrı ayrı bağımsız olarak eğitilmek zorundaydı [4].

Nesne tespitinde daha sonra geliştirilen diğer bir yaklaşım ise tek fazlı nesne algılama ağı, uçtan uca bir modeldir, yani giriş görüntüsünden ağ çıkışına yalnızca bir ağ çıkarımının gerekli olduğu anlamına gelir. Bu mimariye örnek olarak da YOLO serisi, SSD ve RetinaNet yer almaktadır. [1].

Nesne tespiti modelleri 2012 yılındaki AlexNet algoritması ile hızlı bir gelişme göstermeye başlamıştır [1].

Aşağıdaki şekillerde nesne tespitinde modellerin gelişimi kronolojik olarak belirtilmiştir [Şekil 1, 2].

A picture containing text, line, diagram, font

Description automatically generated

**Şekil1: YOLO modelinin ortaya çıkmasına kadar yıl bazlı nesne tespiti algoritmalarının gelişimi [2]**

A picture containing text, diagram, line, screenshot

Description automatically generated

**Şekil 2: YOLO modelinin ortaya çıkmasından sonra yıl bazlı nesne tespiti algoritmalarının gelişimi**

Günümüzde yaygın olarak kullanılan YOLO modeli oldukça basit, tek bir evrişimli ağ, aynı anda birden çok sınırlayıcı kutuyu ve bu kutular için sınıf olasılıklarını tahmin eden bir yaklaşım içerir. YOLO, görüntüler üzerinde eğitildikten sonra bu modelin performansını doğrudan optimize eder. Bu birleştirilmiş model, geleneksel nesne algılama yöntemlerine göre çeşitli avantajlara sahiptir. İlk olarak, YOLO son derece hızlıdır. Nesne tespitini bir regresyon problemi olarak ele aldığından karmaşık bir mimariye ihtiyaç yoktur [4].

## 3.1 YOLO Nasıl Çalışır:

1. Giriş görüntüsü önceden belirlenmiş bir boyuta yeniden boyutlandırma yapılır. Örneğin, 416x416 piksel boyutunda bir görüntü kullanılabilir.

2. YOLO, derin bir sinir ağı olan bir evrişimli sinir ağı (convolutional neural network - CNN) kullanır. Bu ağ, birçok evrişimli ve tam bağlantılı katmandan oluşur.

3. Giriş görüntüsü, evrişimli katmanlarla işlenerek daha yüksek seviyeli özellik haritaları üretilir. Bu özellik haritaları, görüntüdeki nesne özelliklerini temsil eder.

4. YOLO, özellik haritalarını kullanarak tahminlerde bulunur. Her bir tahmin, görüntüdeki bir nesneyi temsil eder.

5. YOLO, tahminleri, önceden belirlenmiş bir kümeleme (clustering) algoritması kullanarak filtreler. Bu kümeleme işlemi, aynı nesneyi temsil eden tahminleri birleştirir ve en güvenilir tahmini seçer.

6. Nesne tahminlerinin sınıflandırma ve konumlandırma bilgileri aynı anda elde edilir. Her bir tahmin, sınıf etiketi ve nesnenin sınırlayıcı kutusunun koordinatlarını içerir.

7. YOLO, tahminlerin güvenilirliğini belirlemek için sınıflandırma ve konumlandırma hatalarını hesaplar. Bu hatalar, nesne algılama performansını değerlendirmek için kullanılır.

8. Son olarak, YOLO, eşik değerine dayalı bir filtreleme yaparak güvenilirlik skoru altında kalan tahminleri kaldırır. Bu işlem, daha yüksek güvenilirliğe sahip nesne tahminlerini elde etmek için yapılır [4].

## 3.2 YOLO Modelinin Mimarisi

YOLO modelinin küçük boyutu, basit yapısı ve hızlı sonuç üretmesi kullanımını yaygınlaşmıştır. Sınırlayıcı kutunun konumunu ve sınıfını sinir ağı aracılığıyla doğrudan hızlı bir şekilde verebildiğinden videolarda da kullanılabilir. YOLO, güçlü bir genelleme yeteneğine sahiptir çünkü diğer alanlara aktarılmak üzere oldukça genelleştirilmiş özellikleri öğrenebilir. Hedef tespiti problemini bir regresyon problemine dönüştürür, ancak tespit doğruluğunun iyileştirilmesi gerekmektedir. YOLO'nun test sonuçları birbirine çok yakın ve grup halindeki nesneler için yetersizdir. Bu zayıf performansın nedeni, ızgaradaki yalnızca iki kutunun tahmin edilmesi ve yalnızca aynı kategorideki yeni bir nesne sınıfının tahmin edilmesidir. Bu nedenle anormal bir en boy oranı ve zayıf genelleme yeteneği gibi diğer koşullar modelin performansını olumsuz etkilemektedir. Özellikle büyük ve küçük nesnelerin tespitinin güçlendirilmesi gerekmektedir. Orijinal YOLO’nun ilk mimarisi, 24 evrişim katmanından ve ardından iki tam bağlantılı kat-mandan oluşur [8].

## 3.3 YOLO Modelinin Zayıf Yönleri

a) Küçük nesnelerin algılanması: YOLO, küçük nesnelerin algılanması konusunda bazı zorluklarla karşılaşabilir. Küçük nesneler, ızgara hücrelerindeki piksel ölçeğinde daha az temsil edilir ve daha az bilgi içerir. Bu, YOLO'nun küçük nesneleri doğru bir şekilde tespit etmekte zorlanabileceği anlamına gelir [5].

b) Düşük çözünürlüklü nesnelerin algılanması: YOLO, düşük çözünürlüklü görüntülerdeki nesneleri algılama konusunda da bazı zorluklar yaşayabilir. Düşük çözünürlük, nesnelerin ayrıntılarının kaybolmasına neden olabilir ve nesne tespiti performansını olumsuz etkileyebilir [6].

c) Çakışan nesnelerin algılanması: YOLO, çakışan nesnelerin ayrımını yapmakta zorlanabilir. Özellikle nesneler birbirleriyle sıkı bir şekilde örtüşüyorsa, YOLO'nun doğru bir şekilde sınıflandırma ve konumlandırma yapması zorlaşabilir [7].

## 3.4. YOLO Modelinin Diğer Modellerle Karşılaştırılması

**1. Hız ve Gerçek Zamanlı Tespit:**

YOLO, gerçek zamanlı nesne tespiti için hızlı ve tek geçişli bir modeldir. Ayrıca, çoklu sınıf tespiti için etkili bir şekilde kullanılabilir.

**2. Ölçek Değişikliklerine Duyarlılık:**

YOLO, ölçek değişikliklerine karşı daha az duyarlı olabilir ve küçük nesneleri doğru bir şekilde tespit etme konusunda bazı zorluklar yaşayabilir .

**3. Çoklu Sınıf Tespiti:**

YOLO, birçok sınıfı aynı anda tespit edebilme yeteneğine sahiptir. Diğer modellerde genellikle sınıf bazlı bölge önerileri kullanılır ve her bir sınıf için ayrı ayrı işlemler yapılır. YOLO ise tek bir geçişte birden çok sınıfı tespit edebilir. [12].

## 3.5. YOLOv8 Modelindeki Yenilikler

YOLOv8, nesne algılama, sınıflandırma ve segmentasyon için geliştirilen en son YOLO modellerinden biridir. Sektöre yön veren YOLOv5 modelini de yazan Ultralytics tarafından geliştirilmiştir. YOLOv8, YOLOv5'e göre çok sayıda mimari ve geliştirici deneyimi değişikliği ve iyileştirmesi içerir [13].

YOLOv8'deki önemli değişikliklerden bazıları şunlardır:

* Darknet53 ağına dayanan ancak performans ve verimliliği artırmak için değiştirilmiş CSPDarknet53 adlı yeni bir omurga ağı eklenmiştir.
* Geleneksel IoU kayıp fonksiyonundan nesne ölçeği varyasyonlarına karşı daha sağlam olan, GIoU adlı yeni bir kayıp fonksiyonu kullanılmıştır.
* Birden çok nesne içeren görüntülerde nesne algılamanın doğruluğunu artırmak için bir uzamsal piramit havuzlama katmanı kullanan YOLOv8-SPP adlı yeni bir eğitim metodolojisi kullanılmıştır.
* YOLOv8'in mobil cihazlar için tasarlanmış hafif bir versiyonu olan YOLOv8-Nano adlı yeni bir model tanıtılmıştır [13].

YOLOv8'in çeşitli nesne algılama kıyaslamalarında YOLOv5'ten daha iyi performans gösterdiği gösterilmiştir. Ayrıca YOLOv5'ten daha hızlı ve daha verimli olması, onu gerçek zamanlı nesne algılama uygulamaları için iyi bir seçim haline getirmektedir [13].

A picture containing text, line, diagram, plot

Description automatically generated

**Şekil 3: YOLOv8 modelinin parametre ve hız bakımından diğer versiyonları ile kıyası [14]**

## 3.6. Modelin Performans Değerlendirmesi:

Mean Average Precision (mAP), nesne tespit modellerinin performansını değerlendirmek için yaygın olarak kullanılan bir metriktir. Sırasıyla mAP şu şekilde hesaplanır [11]:

1. İlk adımda, her nesne sınıfı için bir eşik değeri (threshold) belirlenir. Bu eşik değeri, tahminlerin kabul edilebilirliğini belirler. Örneğin, eşik değeri 0.5 olarak belirlenebilir.

2. Nesne tespit modeli kullanılarak test görüntülerinde nesneler tespit edilir ve tahminler elde edilir.

3. Her bir tahmin, sınıf etiketi, tahminin güvenilirliği (confidence score) ve sınırlayıcı kutusu (bounding box) koordinatlarını içerir.

4. Ardından, her bir tahmin için bir eşleşen gerçek nesne bulunmaya çalışılır. Bu eşleştirmeyi yapmak için iki tahmin arasındaki IoU (Intersection over Union) değeri hesaplanır. IoU, tahminin gerçek nesne ile ne kadar örtüştüğünü ölçer.

5. Eğer bir tahmin, eşik değerini geçen bir IoU'ya sahip gerçek bir nesneyle eşleşirse, bu tahmin doğru kabul edilir. Aksi takdirde, yanlış kabul edilir.

6. Her bir sınıf için doğru ve yanlış tahminlerin sayısı toplanır ve Precision-Recall eğrileri oluşturulur. Precision, doğru tahminlerin toplam tahminlere oranını temsil ederken, Recall, doğru tahminlerin gerçek nesnelere oranını temsil eder.

7. Precision-Recall eğrileri kullanılarak her bir sınıf için AP (Average Precision) hesaplanır. AP, Precision değerlerinin Recall'e göre ortalama birleşik alanını temsil eder.

8. Son olarak, her sınıfın AP değerleri alınır ve bu değerlerin ortalaması alınarak mAP hesaplanır. mAP, tüm sınıfların ortalama hassasiyetini temsil eder ve genel bir performans metriği olarak kullanılır.

Bu adımlar, nesne tespit modellerinin performansını değerlendirmek ve farklı modellerin karşılaştırılmasını sağlamak için kullanılan standart bir yöntemdir [11].

Ayrıca bazı çok kullanılan metriklerle birçok derin öğrenme, nesne tespiti modellerinin performansı ölçülmektedir. Bunların kısaca tanımları şunlardır:

• Kayıp (loss): Modelin tahminlerinin gerçek değerlerden ne kadar uzak olduğunu ölçer. Düşük kayıp değeri daha iyi bir model performansı gösterir.

• Doğruluk (accuracy): Modelin doğru tahmin ettiği veri noktalarının yüzdesidir.

• Kesinlik (precision): Modelin pozitif olarak tahmin ettiği veri noktalarının gerçekten pozitif olma yüzdesidir.

• Duyarlılık (recall): Gerçekten pozitif olan veri noktalarının model tarafın-dan pozitif olarak tahmin edilme yüzdesidir.

## 3.7 Veri Kümesi Tanıtımı

YOLOv8 modelinin eğitilmesi için kullanılan kemik kırıklarına ait veri seti roboflow [22] adlı siteden alınmıştır. Veri seti 1036 adet görüntü içermektedir. Veri seti % 75, % 10 ve % 15 oranlarında eğitim, doğrulama ve test verisi olarak üç kısma ayrılmıştır. Veri setindeki kırık bölgelere ait koordinat bilgileri labels denilen klasörün altında yer almaktadır. Verilerin boyutları 350\*350 olup model eğitimi sırasında 640\*640 boyutuna getirilmiştir. Veriye ait örnek görüntüler aşağıda yer almaktadır [Şekil 4].

A close-up of a person's arm

Description automatically generated with low confidence

**Şekil 4: Kemik kırıklarına ait veri setinden görüntü örnekleri**

## 3.8 Optimizasyon Ve Görüntü Boyutunu Büyütme

YOLOv8 modeli uygulanırken modelin küçük boyutta nesnelerin tespitinde zayıf kaldığı bilindiğinden görüntülerin boyutları 640\*640 olarak büyütülmüştür. Aynı zamanda daha yavaş eğitilmesine rağmen başarım oranı daha yüksek kabul edilen YOLOv8l modeli kullanılmıştır.

Model hem Adam hem de SGD (Stokastik Gradyan Descent) optimizasyon yöntemleri ile eğitilmiştir. Adam optimizasyon algoritması, derin öğrenme modellerinde kullanılan bir optimizasyon algoritmasıdır. Stokastik gradyan inişi (SGD) algoritmasına benzer şekilde çalışır. Ancak, SGD’nin aksine, Adam algoritması her iterasyonda öğrenme oranını değiştirir ve bu sayede daha hızlı bir şekilde yakınsar. [Adagrad ve RMSprop gibi diğer optimizasyon algoritmalarına göre daha az hafızaya ihtiyaç duyar ve daha az hassas hiperparametre ayarlaması gerektirir](https://medium.com/geekculture/a-2021-guide-to-improving-cnns-optimizers-adam-vs-sgd-495848ac6008) [23].

# 4.SONUÇ

* Eğitime sokulan eğitim veri setindeki görüntü adedi 781, doğrulama görüntü adedi 99, test görüntü adedi ise 136’dır.
* Başlangıçta YOLOv8l.pt modeli ile tur sayısı 100, görüntü boyutu 640x640, varsayılan optimizasyon yöntemi olan SGD ve öğrenme oranı 0.01 ile eğitim yapılmıştır.
* Ardından optimizasyon yöntemi Adam ile, öğrenme oranı 0.008 ile değiştirilerek aynı tur sayısı ile model tekrar eğitilmiş,
* Daha sonra optimizasyon yöntemi Adam ve öğrenme oranı varsayılan 0.01 ile 150 tur ile model tekrar eğitilmiş
* En son olarak başlangıçta uygulanan SGD ve 0.01 öğrenme oranı ile modelde iyileşme olup olmayacağının anlaşılması için model 150 tur eğitilmiştir.

En son 150 tur ile çalışan model 101 tur sonunda EarlyStopping regülarizasyon yöntemi ile durmuştur. 50 tur boyunca herhangi bir iyileşme yaşanmadığından otomatik olarak 101. tur sonunda eğitimini tamamlamıştır.

En iyi mAP skoru 0.785 ile 4. modelde yani SGD optimizasyon yöntemi ve 0.01 öğrenme oranı ile elde edilmiştir.

Diğer modellerde sırasıyla 0.78, 0.481, 0.34 oranında mAP skorları elde edilmiştir.

En iyi modele ait hata matrisi grafiği aşağıdaki şekilde gösterilmektedir [Şekil 5]. Şekle göre test verisindeki 136 görüntüden 117 adeti kırık içermekte ve bunların 92 tanesi doğru bir şekilde tahmin edilmiştir.

A picture containing screenshot, text, rectangle, diagram

Description automatically generated

**Şekil 5: Yolov8l modelinin SGD optimizasyon yöntemi ile eğitilmesi sonucu elde edilen hata matrisi**

Model değerlendirmesini kayıp skoru üzerinden yaptığımız zaman bu skorun tur sayısı arttıkça yani model eğitimi devam ettiği süre boyunca azaldığı görülmektedir. Kesinlik ve duyarlılık skorlarında da eğitim boyunca iyileşmenin olduğu söylenebilir. Ayrıca bu grafikten de mAP skorunun % 78 oranına kadar iyileşme gösterdiği görülmektedir [Şekil 6].

**A picture containing text, diagram, line, plot

Description automatically generated**

**Şekil 6: Eğitim ve doğrulama veri setlerine ait kayıp, kesinlik, duyarlılık ve mAP skorları**

En iyi modelin doğrulama veri seti üzerinde bazı görüntülerdeki performansına bakmak istersek aşağıdaki görüntülerde [Şekil 7] olduğu gibi kırık tespitlerinin yapılabildiği, bazı resimlerde yapılamadığı görülmektedir.

A collage of x-ray images of hands

Description automatically generated with low confidence

**Şekil 7: Doğrulama verisi üzerinden modelin yaptığı tahminler**

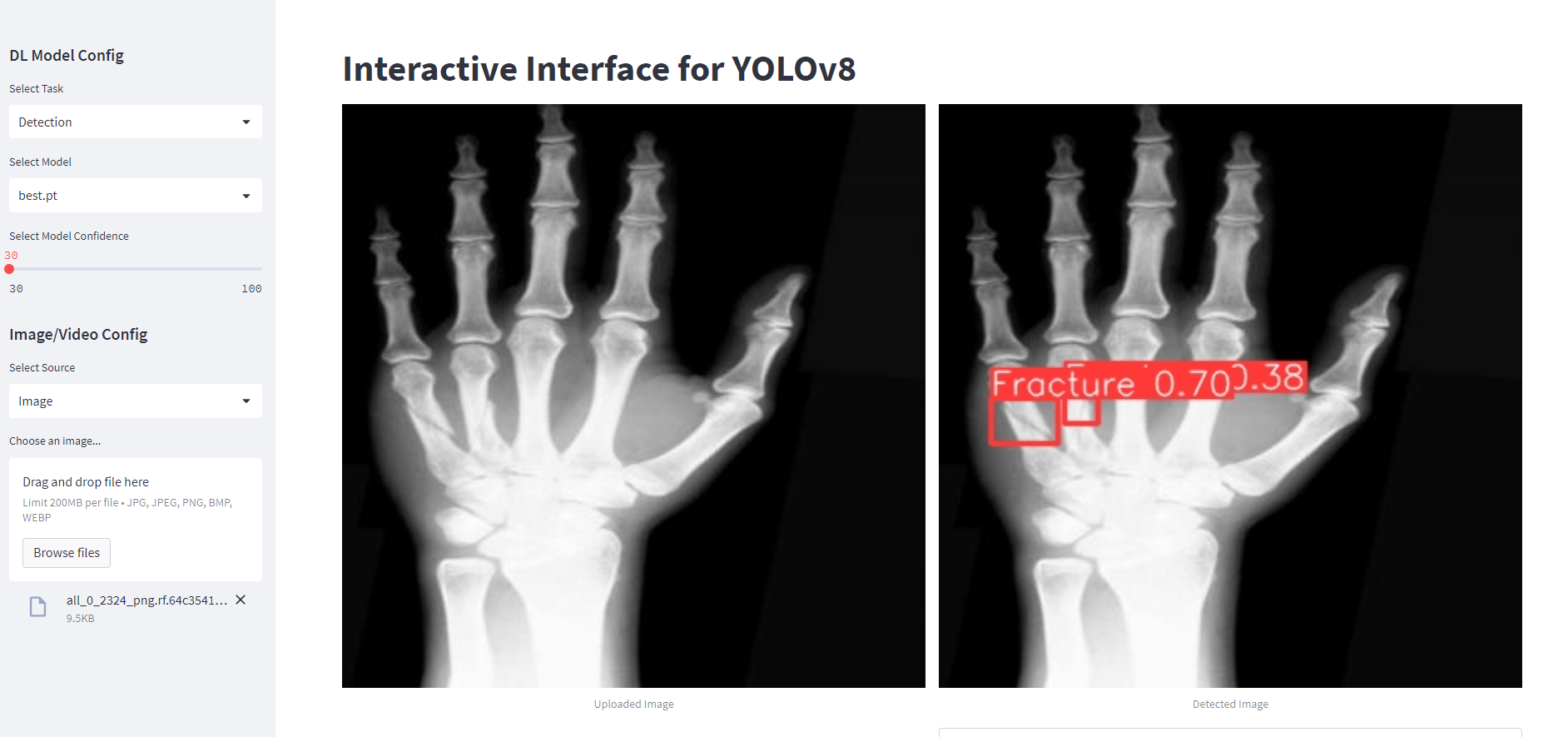
Modelimiz test verisi ile tahmin yapmış olup, test verisindeki bir kaç veri üzerinden yapılan tahminlerin görselleri aşağıda verilmiştir [Şekil 8]. Bazı görüntülerde tahminlerde bulunurken bazılarında kırıklar bulunamamıştır.

A picture containing x-ray film, medical, x-ray, medical imaging

Description automatically generated

**Şekil 8: Test verisi üzerinden modelin yaptığı tahminler**

Çıktılar kullanıcı dostu bir şekilde streamlit arayüzü yardımıyla sunulmuştur. Uygulama streamlit run app.py ile aktif olan ortamda çalıştırılır. Arayüz ile [Şekil 9] resim veya videonun seçilmesi gerekmektedir. Ayrıca eşik değeri seçilerek bu değerin üstündeki tespitler gösterilmektedir.



**Şekil 9: Streamlit ile eğitilmiş model yardımıyla görüntüler üzerinden kırkların tespiti**

# TARTIŞMA

Bu çalışmada kemik kırıklarının tespit edilmesi için nesne tespiti modelleri içerisinde oldukça iyi bir performansa sahip ve popüler olan YOLO modelinin son versiyonu olan YOLOv8’i kullandık. Modelimizi eğitmek için kullandığımız veri setini roboflow adlı siteden YOLO modeline uygun bir metaveri ve dosya düzeni ile aldık. YOLOv8 modelinin farklı hızlarda ve parametrelerle çalışan farklı büyüklükteki modellerinden yavaş çalışmasına rağmen performansı daha iyi olması sebebiyle large modeli ile çalıştık.

Modeli adam ve sgd gibi farklı optimizasyon yöntemleri ve öğrenme oranları eğittik ve sonuç olarak en iyi performansı sgd yöntemi, 0.01 öğrenme oranı olan eğitim sonucunda mAP skorunu 0.78 olarak elde ettik.

Kullandığımız veri setinde yalnızca 1036 adet görüntü mevcuttu, bu sayıyı arttırmak, modelin eğitimine ve de performansına katkı sağlayabilir. Ayrıca veri setinde farklı bölgelerden görüntülere ait kırıklar yerine belirli alanlarda modelin uzmanlaşması ve böylece performansı arttırılabilir.

# KAYNAKÇA

[1] Viswanatha, V., Chandana, R. K., & Ramachandra, A. C. (2022). Real-Time Object Detection System with YOLO and CNN Models: A Review. Journal of Xi'an University of Architecture & Technology, XIV (7), 144-151. ISSN: 1006-7930.

[2] Diwan, T., Anirudh, G. & Tembhurne, J.V. Object detection using YOLO: challenges, architectural successors, datasets and applications. Multimed Tools Appl 82, 9243–9275 (2023). <https://doi.org/10.1007/s11042-022-13644-y>

[3] Mahshiya V. M., Manju V. M., G.Kiruthiga.(2022). A Survey On Object Detection

Techniques Using Tensorflow, Keras And Yolo. IJARIIE, 8(4), 495-500, ISSN(O)-2395-4396

[4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779-788.

[5] J. Yang, W. Choi, and Y. Lin, "Exploit the Unknown Gradually: One-Stage Small Object Detection," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2019.

[6] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: Fully Convolutional One-Stage Object Detection," in Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019.

[7] Y. Liu, R. Wang, S. Shan, and X. Chen, "Toward Real-Time Multi-Object Tracking," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020.

[8] Jiang, P., Ergu, D., Liu, F., Cai, Y., and Ma, B., "A Review of Yolo Algorithm Development," in Proceedings of the 8th International Conference on Information Technology and Quantitative Management (ITQM 2020 & 2021), Procedia Computer Science, 1066-1073 (2022).

[9] Rekha. B. S., Athiya. Marium, G. N. Srinivasan, and S. A. Shetty, "Literature Survey on Object Detection using YOLO," International Research Journal of Engineering and Technology (IRJET), vol. 07, no. 06, June 2020.

[10] K. Chahal and K. Dey, "A Survey of Modern Object Detection Literature using Deep Learning," August 2018.

[11] Girshick, R. (2015). Fast R-CNN. Proceedings of the IEEE International Conference on Computer Vision (ICCV), 1440-1448.

[12] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. In European conference on computer vision (pp. 21-37). Springer.

[13] [blog.roboflow.com/whats-new-in-yolov8/](https://blog.roboflow.com/whats-new-in-yolov8/) (Haziran, 2023)

[14] <https://learnopencv.com/ultralytics-yolov8/> (Haziran, 2023)

[15] Inoue, T., Maki, S., Furuya, T. et al. Automated fracture screening using an object detection algorithm on whole-body trauma computed tomography. Sci Rep 12, 16549 (2022). <https://doi.org/10.1038/s41598-022-20996-w>

[16] Y. Ma ve Y. Luo, "Bone Fracture Detection Through the Two-Stage System of Crack-Sensitive Convolutional Neural Network," Informatics in Medicine Unlocked 22 (2021)

[17] Y. L. Thian, Y. Li, P. Jagmohan, D. Sia, V. E. Y. Chan, ve R. T. Tan, "Convolutional Neural Networks for Automated Fracture Detection and Localization on Wrist Radiographs, Radiology: Artificial Intelligence, 3(1), (2019), <https://doi.org/10.1148/ryai.2019180001>

[18] H. Lin, J. Xi, H. Yang and N. Zheng, “Yolo v3 Based Rib Fracture Detection Method”, 2020

[19] R. Ju and W. Cai, “Fracture Detection in Pediatric Wrist Trauma X-ray Images Using YOLOv8 Algorithm,” arXiv preprint arXiv:2304.05071, 2023

[20] N. Twinprai, A. Boonrod, A. Boonrod, J. Chindaprasirt, W. Sirithanaphol, P. Chindaprasirt ve P. Twinprai, "Artificial Intelligence (AI) vs. Human in Hip Fracture Detection”, Heliyon 8, 2022doi:https://doi.org/10.1016/j.heliyon.2022.e11266

[21] F. Uysal, F. Hardalaç, O. Peker, T. Tolunay, and N. Tokgöz, “Classification of Shoulder X-ray Images with Deep Learning Ensemble Models,” Applied Sciences, vol. 11, no. 6, p. 2723, Mar. 2021, doi: 10.3390/app11062723.

[22] <https://universe.roboflow.com/fracture-uofxm/bone-fracture-detection-ivsy6> (Haziran, 2023)

[23] M. Zhang, Y. Liu, Y. Liu and J. Zhang, “Adam vs. SGD: Closing the generalization gap on image classification,” 2021 IEEE International Conference on Image Processing (ICIP), 2021, pp. 1-5, doi: 10.1109/ICIP42928.2021.9506363.