



# Beyin Tümörü Tanıları İçin YOLOv7 Algoritması Tabanlı Karar Destek Sistemi Tasarımı

## Decision Support System Design Based on YOLOv7 Algorithm for Brain Tumor Diagnosis

Serhat YILMAZ<sup>1,\*</sup> <sup>1</sup> Elektronik ve Haberleşme Mühendisliği, Kocaeli Üniversitesi, Kocaeli, Türkiye, **Orcid:** 0000-0001-9765-7225

### Araştırma Makalesi

Gönderilme Tarihi : 16/01/2023

Kabul Tarihi : 08/06/2023

### Anahtar Kelimeler

Derin Öğrenme  
Evrimsel Sinir Ağı  
Makine Öğrenmesi  
Nesne Tespiti  
You Only Look Once (YOLO)

### Research Paper

Received Date : 16/01/2023

Accepted Date : 08/06/2023

### Keywords

Deep Learning  
Convolutional Neural Network (CNN)  
Machine Learning  
Object Tracking  
You Only Look Once (YOLO)

### Özet

Sağlık alanında derin öğrenme algoritmalarının karar destek sistemleriyle bağdaştırılması her geçen gün daha tutarlı teşhisler ve etkin tedavilerin önünü açmaktadır. Sağlık çalışanları, daha fazla veri, belge ve deneyim içeren karar verme modelleri ile etkileşime geçerek hızlı ve doğru kararlar alabilmektedir. Derin öğrenme modellerindeki gelişmelerle birlikte, imge ve video görüntülerinden doğru ve hızlı bir şekilde nesne tespiti için evrimsel sinir ağları gibi mimariler yaygın olarak kullanılmaya başlanmıştır. Evrimsel sinir ağları üzerine inşa edilen nesne tespit modellerinde, eğitim aşamasından geçtikten sonra test aşamasında önce nesnenin tahmini yapılır ve belirlenen nesne çerçeve içine alınarak tespit aşaması tamamlanır. Bu çalışmada, YOLO algoritması kullanılarak beyin tümörlerinin ve konumlarının MR görüntüleri üzerinde tespiti hedeflenmiştir.

YOLOv7 ve YOLOv7-tiny algoritmaları üzerinde 3203 eğitim görüntüsü kullanılarak model eğitimleri tamamlanmıştır. Bunu, çok sayıda klinik deneyimin karar destek sistemine aktarılması olarak yorumlamak yanlış olmayacaktır. Model eğitim çıktılarının başarımı kesinlik, hatırlama ve F1 puanı gibi ölçütler açısından değerlendirilmiştir. Eğitim görüntüleri haricinde test ve doğrulama görüntüleri de oluşturulmuştur. Görüntüler "makesense.ai" kullanılarak etiketlenmiştir ve Google Colab üzerinden eğitimleri gerçekleştirilmiştir. Tespitlerde %97'lere varan doğruluklar elde edilmiştir. YOLO algoritmasının karar destek sistemlerinin doğruluğunu artırması farklı sağlık kurumlarındaki tanılar arasında tutarlılığı sağlayacak, kişisel kaynaklı tespit hatalarını en aza indireyecek ve izlenecek tedavi prosedürlerini olumlu yönde etkileyecektir.

### Abstract

Associating deep learning algorithms with decision support systems in the field of health paves the way for more consistent diagnoses and effective treatments. Healthcare professionals can make fast and accurate decisions by interacting with decision-making models that contain more data, documents and experience. With the developments in deep learning models, architectures such as convolutional neural networks have been widely used for accurate and fast object detection from images and video images. In object detection models built on convolutional neural networks, after the training phase, first the object is estimated in the test phase, and the detection phase is completed by taking the determined object into the frame. In this study, it is aimed to detect brain tumors and their locations on MR images using the YOLO algorithm.

Model trainings were completed using 3203 training images on YOLOv7 and YOLOv7-tiny algorithms. It would not be wrong to interpret this as transferring a large number of clinical experiences to the decision support system. The performance of the model training outcomes was evaluated in terms of criteria such as precision, recall and F1 score. In addition to the training images, test and validation images were also created. Images are tagged using "makesense.ai" and trained via Google Colab. Accuracies up to 97% were obtained in the determinations. Increasing the accuracy of the decision support systems of the YOLO algorithm will ensure consistency between diagnoses in different health institutions, minimize personal-based detection errors and positively affect the treatment procedures to be followed.

## 1. Giriş

Beyin tümörleri, kafatası bölgesinde, beyin veya çevresindeki dokularda hücrelerin kontrolsüz bir şekilde

büyümesi olarak tanımlanır. Bu büyümenin kafatası içinde yol açtığı basınç artışına bağlı olarak şiddetli baş ağrısı, kusma ve bulantı gibi yaygın belirtiler görülebilir. Hasta hikayesi dinlenerek, şikayetler tümör semptomları ile örtüşüyorsa, yaygın bir tanı yöntemi olan bilgisayarlı tomografiye başvurulmalıdır [1]. Günümüzde gelişmekte

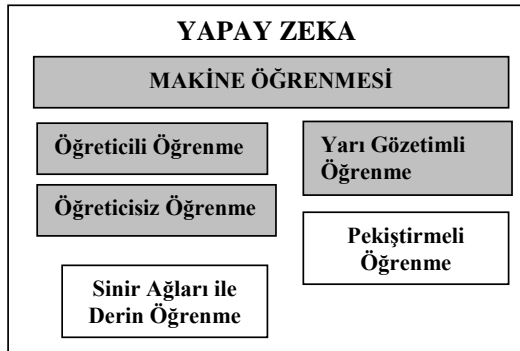
\* Sorumlu Yazar (Corresponding Author): serhaty@kocaeli.edu.tr



olan yapay zeka çalışmaları sağlık sektöründe de başarılı uygulama alanları bulmuştur. Bunlardan biri de hekimlerin işlerini kolaylaştıran, yanlışları en aza indiren ve tüm sağlık birimlerinde standart ve doğru teşhisler koyabilmeyi mümkün kılan karar destek sistemleridir [2]. Derin öğrenme, Yapay Sinir Ağlarının (YSA) gelişmiş bir alt dalıdır ve son yıllarda, Python, C ve C++ dilleri üzerine inşa edilen OpenCV, Tensorflow gibi kütüphanelerle kullanılarak nesne tespiti için bir makine öğrenmesi yöntemlerinden birisidir [3, 4]. YOLO modeli, CNN mimarisi üzerine kurulmuş bir yapıdır ve nesne tespiti konusunda tercih edilir. Bu model, algılanan nesnenin etrafında bir çerçeve çizerek nesneyi tanımlarken, aynı zamanda nesne tahmini yapabilir. YOLO algoritmasının gün geçtikçe daha kullanışlı ve daha etkin sürümleri çıkmaktadır [5]. Bu çalışmada tomografi görüntülerinin işlenmesi, YOLO algoritması kullanarak sınıflandırılması ve sonrasında yeni görüntüler üzerinde tümör tespiti yapmak hedeflenmiştir. Eğitim ve sınav süreçlerinde güncel sürümlerin içerdiği YOLOv7 ve YOLOv7-tiny algoritmaları üzerinde model eğitimleri tamamlanmıştır ve yüksek doğrulukta tümör tespiti yapabilen sonuçlar elde edilmiştir. Model eğitiminde 3203 görüntü kullanılmıştır. Eğitim görüntüleri haricinde test ve doğrulama görüntüleri de oluşturulmuştur. Görüntüler “makesense.ai” [6] kullanılarak etiketlenmiştir ve Google Colab [7] üzerinden eğitimleri gerçekleştirilmiştir.

## 2. Yöntem: Yapay Zeka

Yapay zeka; görme, duyma, öğrenme ve karar verme gibi insana özgü kabul ettiğimiz bir çok etkinliğin yazılımla yapılabilmesidir. Bu amaca erişmek için özellikle makine öğrenimi ve derin öğrenme gibi yöntemler kullanılır (Şekil.1.) [8]. John McCarthy 1956 yılında Yapay Zekâ terimini kullanmıştır ve bu terim, zeki makine ve zeki bilgisayar programları yapma bilimini ve mühendisliğini ifade etmek için kullanılmıştır [9].



Şekil.1. Yapay zeka ve makine öğrenmesi kapsamları

### 2.1 Makine Öğrenmesi

Makine öğrenmesi, yapay zekâ ve bilgisayar bilimleri alanında bir dal olarak, bir makinenin verileri kullanarak

öğrenme yeteneğini geliştirmeyi amaçlar. Bu öğrenme yeteneği sayesinde, bir makine öğrenmesi modeli, veri setinde görülen özellikleri kullanarak bir sınıflandırma görevinde doğruluğu artırabilir veya bir regresyon görevinde tahminleri daha doğru hale getirebilir. Örneğin, bir makine öğrenmesi modeli, veri setinde görülen resimleri kullanarak resimlerin ne olduğunu tahmin etmeye çalışabilir veya veri setinde bulunan müşteri verilerini kullanarak müşterilerin satın alma davranışlarını tahmin etmesi için eğitilebilir [10]. Böylece, makine öğrenmesi sayesinde bir makine karmaşık problemleri çözebilir ve tecrübe ederek o andaki ve gelecekteki durumlarda karar verme yeteneğini kazanır. Makine öğrenmesi; Öğreticili, Öğreticisiz, Yarı Gözetimli, Pekiştirmeli ve Derin Öğrenme olarak beşe ayrılabilir [10]. Makine öğrenmesi, günümüz teknolojisine adapte edilerek sayısal asistanlar, siber güvenlik uygulamaları, medikal görüntü analizi, sürücüsüz otomobiller gibi birçok alanda kullanılır. Teknolojiye ulaşma maliyetinin azalmasıyla birlikte, makine öğrenmesi günlük yaşamımızla daha da yakınlaşacaktır.

### 2.2 Derin Öğrenme

Derin öğrenme, makine öğrenimi yöntemlerinden biridir ve yapay sinir ağları (YSA) tabanlıdır. Bu yöntemler, gerçekçi veri kümelerinden çok sayıda özellik çıkarır ve bu özellikleri kullanarak yüksek doğruluk oranlarına ulaşmayı amaçlar. Derin öğrenme yöntemleri, özellik çıkarma işlemlerini birden fazla katman halinde gerçekleştirir. Bu katmanlar arasında veri taşınır ve her bir katman önceki katmanlardan öğrendiği bilgileri kullanarak daha derin bir anlam çıkarmaya çalışır. Bu sayede, derin öğrenme yöntemleri gerçekçi veri kümelerinde oldukça iyi performans gösterir ve bu nedenle görüntü tanıma, ses tanıma ve metin üzerinde dil işleme gibi alanlarda kullanılır [11].

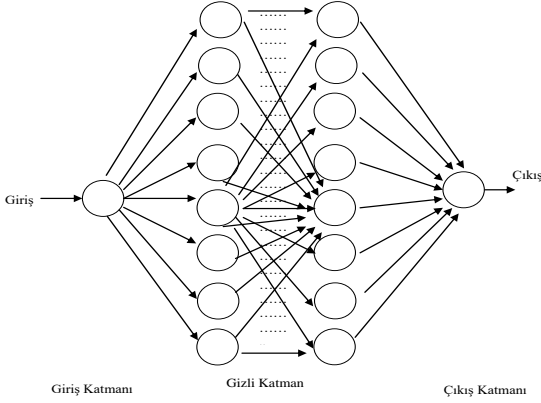
### 2.3. Yapay Sinir Ağları (YSA)

Yapay sinir ağları, gerçek sinir sistemlerinden esinlenilerek geliştirilmiş, yinelemeli olarak parametrelerini güncelleyen (öğrenen), öğrendiği örüntü hakkında tahminlerde bulunabilen, birbiriyle bağlantılı çok sayıda işlem biriminin oluşturduğu bir veri işleme yapısıdır [12]. Yapay sinir ağları, girdi verilerini ağırlıklı katmanları aracılığıyla işler ve çıktı verilerini üretir [13]. Her bir işlem biriminin çıkışı, önceki katmandan gelen ağırlıklı girişlerin ( $u_i$ ) ve eşik değerinin ( $\theta$ ) toplamının ( $S$ ) (Denklem.1) bir aktivasyon fonksiyonundan ( $\psi$ ) geçirilerek bulunur [14]. Burada

$$S = w_1u_1 + w_2u_2 + \dots + w_nu_n - \theta = \left( \sum_{i=1}^n w_iu_i \right) - \theta \quad (1)$$

Bir yapay sinir ağı tasarlarlarken, amaç en uygun çıktı sonucunu verecek  $w$  ve  $\theta$  değerlerini belirlemektir. Bu

değerler, ağın performansını belirleyecek ve en iyi sonuçları elde etmemize yardımcı olacaktır. Bu işlem birimleri giriş, gizli ve çıkış katmalarına yerleştirilerek w ağırlıkları ile birbirine bağlanır (Şekil.2). Veri akışı ileri yöndedir. Parametre güncellemeleri için ise geriye yayılım algoritmaları gibi optimizasyon yöntemlerine başvurulur.



Şekil.2. Çok katmanlı bir YSA yapısı [15]

## 2.4. YSA'da Evrişim Kavramı

Bu işlemin görevi bir filtre aracılığı ile imgenin boyutlarını küçültmek ve her katman için öznelik haritası oluşturmaktır. Bir katmandaki harita, imgenin içindeki belirli bir özelliğe odaklanırken, diğeri kenar bulmayı hedefleyebilir. Şekil.3'te, 3x3'lük filtre, 6x6 boyutundaki imge ile kaydırılarak çarpılmıştır. Böylece 4x4 boyutunda öznelik matrisi elde edilmiştir [16].

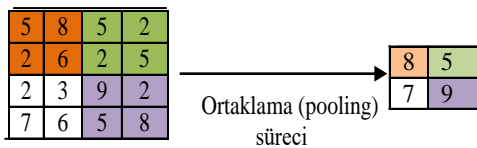
$$\begin{bmatrix} 6 & 3 & 3 & 4 & 5 & 2 \\ 4 & 5 & 4 & 8 & 2 & 5 \\ 2 & 1 & 1 & 4 & 5 & 8 \\ 7 & 7 & 6 & 5 & 4 & 5 \\ 2 & 3 & 2 & 6 & 8 & 2 \\ 7 & 6 & 4 & 3 & 5 & 8 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & -1 \\ 2 & 3 & 2 \\ 7 & 6 & 4 \end{bmatrix} = \begin{bmatrix} 4 & -7 & -4 & 1 \\ 2 & -4 & 0 & -1 \\ 2 & -4 & -8 & 0 \\ 4 & 2 & -5 & -1 \end{bmatrix}$$

6x6 Giriş Matrisi      3x3 Filtre      Öznelik Matrisi

Şekil.3. Evrişim katmanında giriş görüntüsüne filtre uygulanması [17]

## 2.5. YSA'da Pooling (Ortaklama) Kavramı

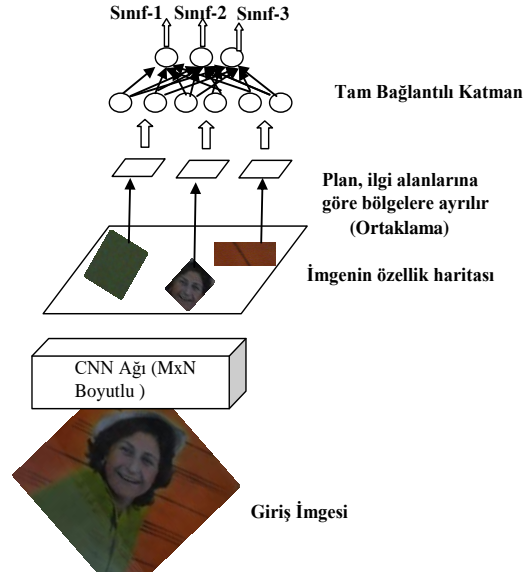
Evrişimsel YSA katmanlarından biridir ve filtre tarafından taranan imgenin boyutunu, imge matrisindeki en büyük (Şekil.4), ortalama veya toplam değeri, o bölgeyi temsil eden tek değer olarak seçerek, en az kayıpla küçültmeyi hedefler [18].



Şekil.4. İmge matrisindeki en büyük değerlerin 2x2'lik filtre ile ortaklanması [18]

## 2.6. Evrişimsel Sinir Ağı (CNN) Modelleri

Yann Lecun, 1998 yılında geri yayılım (back propagation) temelli öğrenme işlemine dayalı evrişimsel sinir ağı modelini ilk kez ortaya çıkarmıştır. AlexNet'in 2012 yılındaki ImageNet verileri üzerinde gerçekleştirilen bir yarışmada, verileri CPU yerine GPU tabanlı olarak eğitmesi evrişimsel sinir ağı alanında çalışanlar için büyük ilham kaynağı olmuştur.



Şekil.5. Fast R-CNN mimarisi [21]

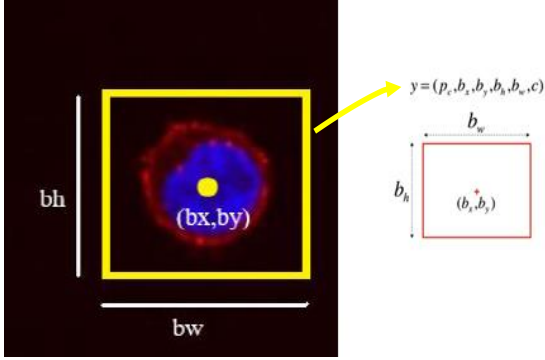
Son dönemlerde, evrişimsel sinir ağı mimarileri arasında bir çok yöntem kabul görmüştür. Seçici arama yöntemi ile görüntüyü nesne olabilecek çok sayıda bölgeye ayırıp, tespit edilen nesnelere sınıflara ayıran Bölgesel CNN (R-CNN) yöntemi bunlardan biridir [19]. Bunu, eğitim ve tespit süresini hızlandıran Fast ve Faster R-CNN yöntemleri (Şekil.5.) izlemiştir [20,21].

İşlemleri hızlandırmak için, imgenin özellik haritası çıkarıldıktan sonra, olası nesne bölgeleri tespit edilir, maskeleme yapılır ve sadece bu bölgelere ortaklama işlemi yapılır. Bu yöntem Mask R-CNN adı verilir [22]. Olasılık olarak bir nesne içeren bölgeler belirlenir. Bu bölgeleri belirlemek için (Region Proposal Network: RPN) kullanılır ve daha sonra bölgesel önerilerde havuzlama işlemi uygulanır.

## 2.7 You Only Look Once (YOLO) Algoritması

Diğer algoritmalar imge üzerindeki nesnelere tespit etmek için görselin tamamını kullanırken, YOLO görseli parçalara böler ve bu alanlardaki nesnelere bulunduğu kutuları çizer. Bu kutulara "sınırlayıcı kutu" (bounding box) denir ve her alan nesnenin orada bulunma ihtimalini hesaplar. Ayrıca, bu ihtimaller için "Güven Skoru" adı verilen bir ölçüt de hesaplanır, bu skor nesnenin sınırlayıcı

kutuda yer alan yüzdesiyle eğitilen nesnelere arasındaki benzerlik oranını gösterir. YOLO, görseli tek seferde sinir ağından geçirir, bu da işlemin daha hızlı gerçekleştirilmesini sağlar. Sınırlayıcı kutular 3x3, 5x5, 7x7 gibi farklı boyutlarda olabilir ve nesnenin olup olmadığını tespit etmek için oluşturulurlar. Daha sonra, nesne kutu içerisindeyse, orta noktasının kutunun içinde olup olmadığına bakılır. YOLO, sınırlayıcı kutular için tek tek değerlendirme vektörü oluşturur ve bu vektörlerin uzunluk, yükseklik ve ait olduğu sınıfı belirler.



Şekil 6. Sınırlayıcı kutu örneği [23]

Şekil 6'daki görüntüde, bir nesnenin tespit edildiği kutu y vektörü ile gösterilmiştir [23]. Bu vektörün parametreleri aşağıdaki gibidir:

pc: Güven skorunu ifade eder ve nesnenin tahmin ettiği nesneyle eşleşme yüzdesini gösterir.

bx: Nesnenin merkez noktasının x koordinatını verir.

by: Nesnenin merkez noktasının y koordinatını verir.

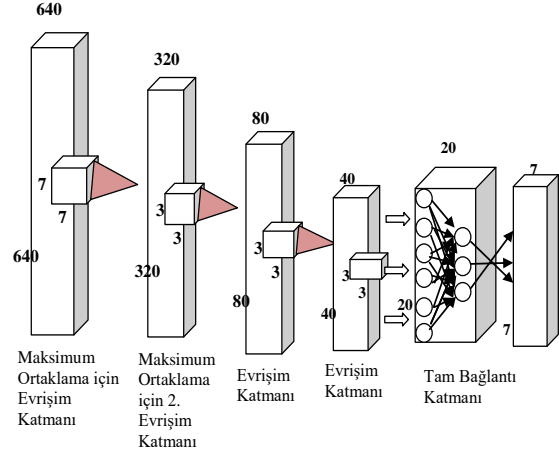
bh: Nesnenin genişliğini gösterir.

bw: Nesnenin yüksekliğini gösterir.

C: Bağlı Sınıf Olasılığı, eğitimde kaç tane sınıf var ise o miktarda tahmin değerini gösterir.

Güven skoru, bağlı sınıf olasılığı ile "kutu güven skoru" olarak adlandırılan tahminin çarpımıyla hesaplanır.

Anlatılan işlemler, bir nesnenin sınırlayıcı kutusunda bulunduğu durumda yapılır. Eğer birden fazla nesnenin bulunduğu bir kutu durumu ile karşılaşırsa, bu durum "çapa kutusu" yöntemiyle çözülür ve her nesne için ayrı kutular oluşturulur. Bu kutular için yukarıda açıklanan değerler hesaplanır ve bu hesaplamada "IoU" (Intersection over Union) kullanılır [23]. Gereksiz kutuların ortaya çıkmasını önlemek için, "Maksimum Olmayanı Bastırma" (Non Max Suppression) algoritması da eklenir. Bu algoritma, ilk hesaplamada düşük güven skoru olan kutuları iptal eder ve sonuç olarak en yüksek güven skoru olan kutuları vermek hedeflenir. YOLO mimarisi, Şekil 7'de sunulmaktadır.



Şekil 7 YOLO mimarisi [24]

Algoritma temel olarak, giriş imgesini bölgelere ayırır. Güven puanı düşük olan çerçeveler, nesne olmadığı varsayılarak elenir. İmgenin boyutu bir filtre ile yüksek seviyeli ortaklama evrişim katmanlarından geçirilerek küçültür.

## 2.8. YOLOv7

YOLOv7, görüntülerdeki nesnelere tespit etme ve sınıflandırma için tasarlanmıştır, önceki YOLO sürümlerinden daha hızlı ve daha doğru sonuçlar veren görüntü tanıma algoritmalarından biridir.

YOLOv7, etkili bir özellik bütünleştirme yöntemine, daha doğru bir nesne algılama başarımına ve artırılmış etiket atama ve model eğitim verimliliğine sahip olması nedeniyle hızlı ve güçlü bir ağ mimarisine sahiptir. Diğer derin öğrenme modellerine göre çok daha az bilgi işlem donanımı gerektirmesi ve önceden eğitilmiş ağırlıklar olmadan bile küçük veri kümeleriyle hızlı bir şekilde eğitilebilmesi, YOLOv7'nin daha etkin bir model olmasını sağlamaktadır [25]

Nesne takip algoritmalarının genel olarak imgeyi bölgelere ayırıp, nesne bulunan olası bölgeleri seçip, her bölgeye ayrı ayrı sınıflandırma işlemi yapmaları işlem yükünü artırmaktadır. YOLOv7, imgeyi tek seferde Evrişimsel YSA'dan geçirip grup normalizasyonu yapar. Bu da 5 FPS'den 160 FPS'ye kadar değişebilen bir çerçeve işleme hızı kazandırır [26].

## 3. MR Görüntülemelerinde Beyin Tümörü'nün Segmentasyonu ve Tespiti

Beyin tümörü, kontrolsüz bir şekilde büyüyen ve beyin dokusuna saldıran kanser hücrelerinin yol açtığı bir hastalık türüdür. Beyin tümörleri iyi huylu ve kötü huylu tümörlerden oluşur. Kötü huylu beyin tümörleri genellikle etrafını saran yağın eşlik ettiği kan pıhtıları şeklindedir [27]. Beyin MR görüntülerinin segmentasyonu, beyindeki anatomik yapıların ölçümü ve görselleştirilmesinde, beyin tümörlerinin incelenmesinde ve bu bilgiler doğrultusunda

yapılacak cerrahi planlamalarda önemli rol oynar. Karşılaştırılabilir araştırmalarla, anizotropik filtreler, beklenti maksimizasyonu (EM) sıvı inceltmeli inversiyon iyileştirme gibi çeşitli yöntem ve tekniklerin birleştirilmesiyle 2 boyutlu (2D) ve 3 boyutlu (3D) beyin MR görüntülerinin iyileştirilmesi tümörün daha iyi segmentasyonu ve saptanması sağlanabilir. Burada başarımlar ölçütleri, segmentasyonun doğruluğu, hassasiyeti ve özgüllüğünün doğru belirlenmesidir [28].

#### 4. YOLOv7 Yöntemi ile Beyin Tümör Tanısı için Karar Destek Sistemi Modelinin Hazırlanması ve Eğitilmesi

Modelin hazırlanması, eğitilmesi ve izlenen adımlar Şekil.8'deki akış şemasında sunulmaktadır. Eğitim sürecinin tamamı üç başlıkta toplanabilir. Bunlar veri seti hazırlama, modelin eğitilmesi ve eğitilen modelin sınanmasıdır.

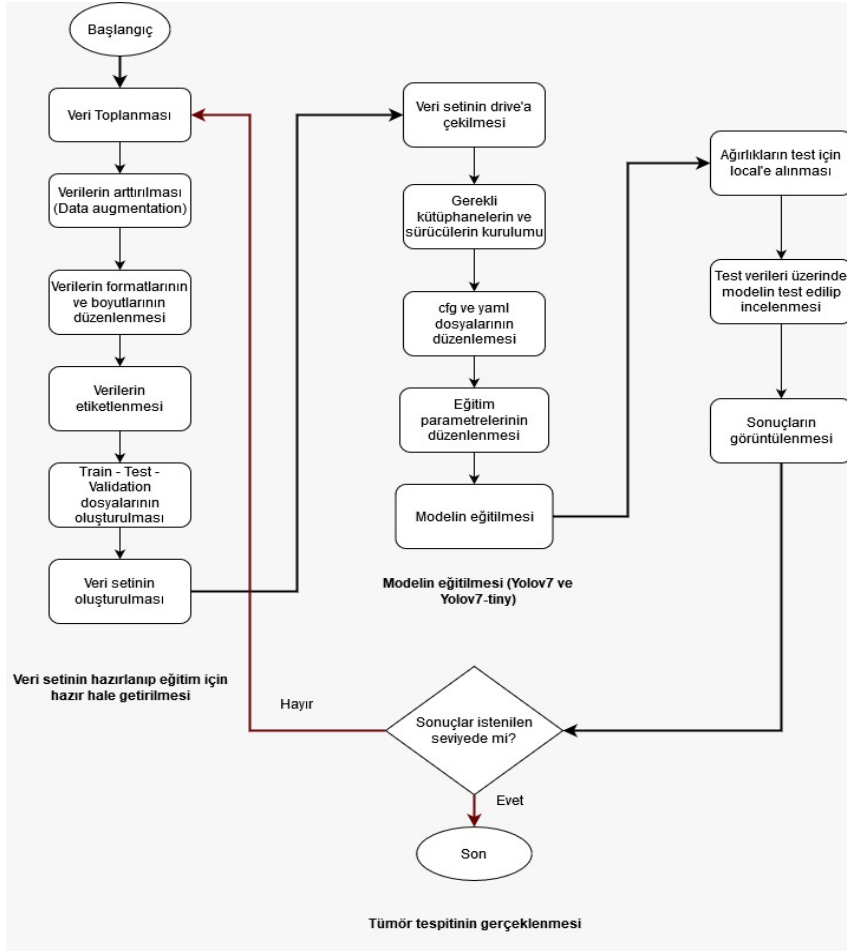
##### 4.1 Veri Seti

The Image Data Resource (IDR) gibi kaynaklar, tıp alanında elde edilmiş yüksek çözünürlüklü görüntü verilerini erişim ve akademik çalışmalarda kullanıma

açmaktadır [29]. Eğitim modelimiz için 3203 tane tümörlü beyin MR görüntüsü toplanmıştır. Eğitim verileri kaggle ve github linklerinde paylaşılan veri setlerinden alınmıştır [30-33]. Ayrıca test için 110, doğrulama için 87 görüntü kullanılmıştır (Şekil.8). Veri setimizdeki görüntülerin çözünürlüğü ne kadar iyiyse doğru orantılı bir şekilde nesne tespitinde başarı oranı da iyi olacaktır. Daha sonra etiketleme işlemi için online etiketleme aracı olan "makesense.ai" [6] kullanılmıştır. Buradan her bir görüntü için etiketin değerlerini yazan .txt formatında dosya elde edilir. Bu değerlerden ilki etiketlenen görüntünün sınıfını belirtir. İkinci değer nesnenin merkez noktasının apsisini verir. Üçüncü değer nesnenin merkez noktasının ordinat değeridir.

Dördüncü değer nesnenin genişliğinin apsisi değerini, son değer ise nesnenin genişliğinin ordinat değerini gösterir. Aşağıda, etiketlenmiş veriye ait değerlerin .txt formatında yazım örneği verilmiştir.

```
tumorBrain2.txt - Notepad
File Edit Format View Help
0 0.825125 0.340674 0.289000 0.246114
```



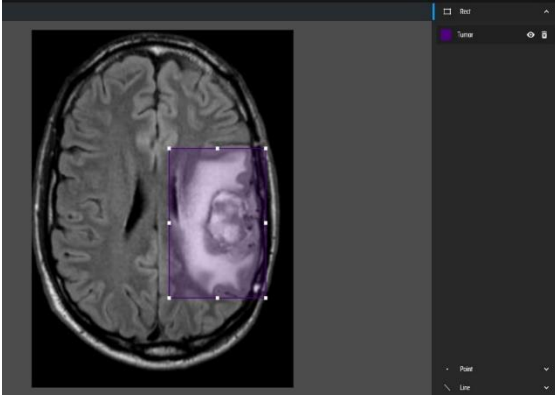
Şekil 8. Eğitime ait akış şeması

MR görüntüsü içindeki sınırları belirli bir kitle, sınırlayıcı çerçeve içine alınmaktadır. Çerçeve merkezinin koordinatları kitle içinde ise burada bir kitle olduğu belirlenmiş olur. Çerçeve merkezinin koordinatları kitleye göre hesaplanır. Ancak yükseklik ve genişliği tüm görüntünün boyutları tarafından belirlenir. Ağın ileri doğru yayılma işlemleri sonucunda YOLO algoritması kitlenin belirli bir sınıfa (burada genel anlamda tümör olarak sınıflandırılmıştır) girme olasılığını puanlandırır (Denklem.2).

$$\text{Puan}_{c,i} = p_c \times S_i \quad (2)$$

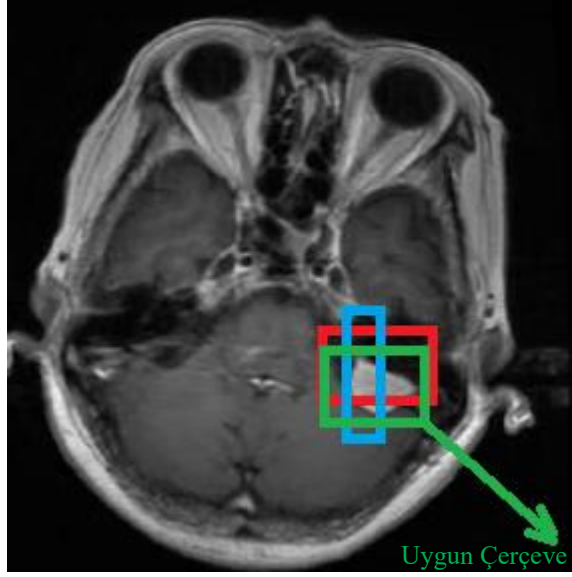
Burada  $p_c$  çerçevenin tümör olasılığı,  $S_i$  ise sınıf numarasıdır. İçlerinden en yüksek olasılığa sahip olan sınıf seçilir ve puan değeri çerçeve üzerine atanır. Tüm çerçeveler için aynı işlem yapılır.

Denklem 2'de sınıf olasılığı hesapladıktan sonra görüntü üzerinde etiketleme yapılır. Algoritma için etiketleme süreci YOLO Modelinin bu nesnelere tanımlaması ve sınıflandırması için görüntüdeki belirli bölgeleri veya nesnelere işaretleme veya açıklama ekleme işlemleridir [23]. Şekil.9'da verilerin etiketlenmiş bir görüntüsü örnek olarak verilmiştir.



Şekil 9. Etiketlenmiş veri görüntüsü

Etiketlemede, en yüksek olasılığa sahip sınıf belirli bir çerçeveye atanır. Görüntüdeki tüm çerçeveler için aynı işlem gerçekleşir (Şekil.10).



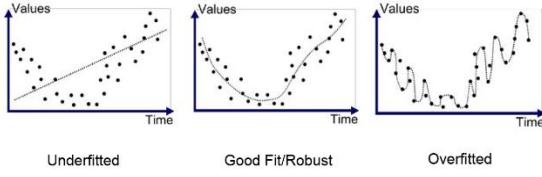
Şekil.10. Oluşan Sınırlayıcı Çerçeveler

Bunu aşmak için maksimum olmayı bastırma yöntemi ile güven skoru en yüksek olan çerçeve ile önceden bir IoU (birlik üzerinden kesişme) oluşturularak birbirlerine çok yakın olan bu sınırlayıcı çerçeveler ortadan kaldırılır (Denklem.3) [23].

$$\text{IoU} = \frac{\text{Örtüşen Alan}}{\text{Birleşik Alan}} \quad (3)$$

En yüksek güven skoruna sahip çerçeve için ve tüm sınırlayıcı çerçeveler için IoU değeri hesaplanır, ardından IoU değerleri eşikten büyük olan sınırlayıcı çerçeveler, aynı nesneyi içerdiği ancak güven skoru düşük olduğu için atılır.

Derin öğrenme eğitim modelleri için veri setinin büyüklüğü ve çeşitliliği önemli bir unsurdur. Gerçekte karşılaşılabilecek durumlar göz önünde bulundurularak veri seti oluşturulmalıdır. Bu durum dikkate alınmaz ise modelin eğitimi aşırı öğrenme (overfitting) durumuna girebilir. Aşırı öğrenme, modelin eğitim verileri üzerinde keskin bir sonuca sahip olup test verilerinde aynı doğrulukta sonuçlar vermemesi durumudur. Kısaca eğitim verileri üzerinden modelin ezber yaptığı istenmeyen bir durumdur. Ek olarak aşırı öğrenmenin aksine, bir model yetersiz öğrenmeye sahipse (underfitting), modelin eğitim verilerine uymadığı, bu nedenle verilerdeki genel eğilimi kaçırdığı anlamına gelir. Yetersiz öğrenme sorunu olan modellerde hem eğitim hem de test veri setinde hata oranı yüksektir. Şekil.11'de bir modelin aşırı öğrenmiş, yetersiz öğrenmiş ve iyi öğrenmiş örnekleri verilmiştir.



**Şekil 11.** Modelin yetersiz öğrenmiş, iyi öğrenmiş ve aşırı öğrenmiş örnek grafikleri [34]

Çözüm olarak izlenebilecek yöntemlerden bir tanesi transfer öğrenme algoritmasıdır. Başka bir metot ise veri artırma (Data Augmentation) yolunun izlenmesidir. Veri artırma, mevcut verilerden yeni veriler üreterek veri miktarını yapay olarak artırma işlemidir. Bu işlem, veri kümesini genişletmek için orijinal verilerin gizli alanında yeni veri noktaları oluşturmak için verilere küçük değişiklikler eklemeyi veya makine öğrenimi modellerini kullanmayı içerir. Artırılmış veri ile yapay veri arasındaki farkı şu şekilde açıklayabiliriz:

- Yapay veriler: Gerçek verilerle doğrudan bir bağlantısı olmadan bilgisayar algoritmaları tarafından rastgele veya bir kurala dayalı olarak üretilen yapay verilerdir.
- Artırılmış veriler: Eğitim setinin çeşitliliğini artırmak için bir tür küçük geometrik dönüşümlerle (çevirme, öteleme, döndürme veya gürültü eklemesi gibi) orijinal görüntülerden türetilmiştir.

Giriş görüntüleri 416x416 boyutunda bir ağa dahil edilmektedir. Kapasitenin üstündeki giriş boyutları eğitim sırasında hatalara yol açıp hiç başlamaması ya da yarım kalmasına yol açabilmektedir.

## 4.2. Modellerin Eğitim Çıktıları

Başarım parametrelerinin oluşturulmasında karışıklık matrisine (confusion matrix) ihtiyaç duyulur. Matrisi oluşturabilmek için bilinmesi gereken dört değer vardır. Bunlar:

- True Positive (TP): Nesnenin gerçekte var olduğu ve tahminin de nesne algıladığı durum (1-1).
- True Negative (TN): Nesnenin gerçekte var olmadığı ve tahminin de nesne algılamadığı durum (0-0).
- False Positive (FP): Nesnenin gerçekte var olmadığı ancak tahminin nesne algıladığı durum (0-1).
- False Negative (FN): Nesnenin gerçekte var olduğu ancak tahminin nesne algılamadığı durumdur (1-0).

Karışıklık matrisi kullanılarak elde edilen sonuçlardan yararlanarak YOLOv7-tiny algoritmasında başarımlar parametreleri hesaplanır. Bunlar sırasıyla Kesinlik, Hatırlama ve F1 puamıdır.

**Kesinlik (Precision):** Görüntüde belirtilen tümörlerin kaçının gerçekte mevcut olduğunu belirleyen ölçüttür. Yine 0 ile 1 arasında değerler alır ve değer ne kadar yüksek

olursa tespitin doğruluğu o kadar kesindir. Denklem(4) yardımıyla hesaplanmaktadır [35].

$$TP/(TP + FP) \quad (4)$$

**Hatırlama (Recall):** Görüntüde gerçekte mevcut olan tümörlerin kaçının tahmin edildiğini ölçen ölçüttür. 0 ile 1 arasında değerler alır ve değer ne kadar yüksek olursa o kadar iyidir. Bu ölçüt aynı zamanda doğruluk oranı olarak da adlandırılır. Denklem.5 yardımıyla hesaplanmaktadır.

$$TP/(TP + FN) \quad (5)$$

**F1 Puanı:** Modelin başarımını daha iyi değerlendirebilmek için, farklı modellerle karşılaştırma yapılır. Ancak hassasiyet ve kesinlik üzerinden yapılacak karşılaştırmalar zor olabilir. Örneğin bir modelin kesinliği yüksek olabilirken diğerinin hassasiyeti yüksek olabilir. Bu durumda karşılaştırma yapmak zor olabilir. Bu nedenle, seçim yaparken tek bir F1 değerinden yararlanılabilir. Çünkü F1 yüksekse, diğer iki değer de yüksek çıkacaktır.

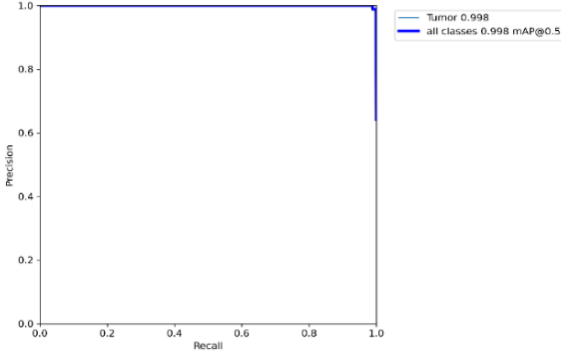
F1 puanı, Hatırlama ve Kesinlik değerlerinin harmonik ortalamasıdır. Harmonik ortalama, sadece ortalamanın alınması yerine, iki değer çarpımının logaritmasının alınması suretiyle hesaplanır. Bu, hassasiyet değeri 1 ve kesinlik değeri 0 olan bir model için F1 puanı 0.5 olması gibi uç durumların değerlendirilmesinde yanlıtıcı etkileri ortadan kaldırır. F1 puanı, Denklem(6) kullanılarak hesaplanır.

$$F1 = \frac{2 * Kesinlik(Precision) * Hatırlama(Recall)}{Kesinlik(Precision) + Hatırlama(Recall)} \quad (6)$$

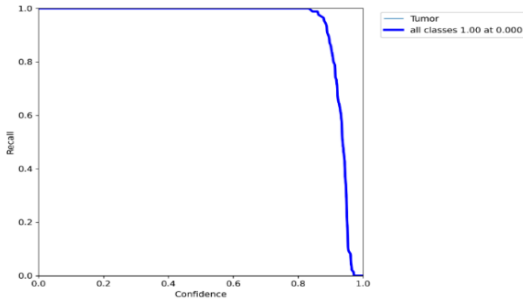
## 5. Sonuçlar

Bu çalışmada beyin MR görüntüleri üzerinden tümör tespiti yapmaya yönelik bir karar destek sistemi modeli ve bunun için gerekli algoritmalar tasarlanmıştır. Model eğitimleri YOLOv7 ve YOLOv7-tiny algoritmaları üzerinde yapılmıştır. Eğitim sürecinde 3203 görüntü kullanılmıştır. Eğitim görüntüleri haricinde test ve doğrulama görüntüleri de oluşturulmuştur. Test için 110, doğrulama için 87 görüntü kullanılmıştır. Görüntüler “makesense.ai” kullanılarak etiketlenmiştir ve Google Colab üzerinden eğitimleri gerçekleştirilmiştir.

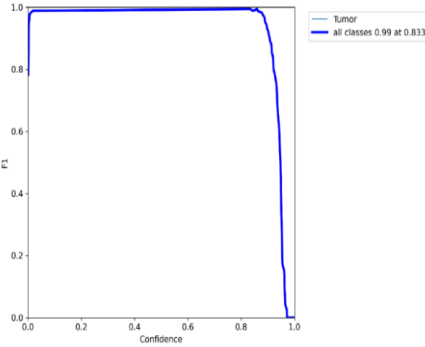
YOLOv7-tiny algoritması ile elde edilen Kesinlik, Hatırlama ve F1 skorları yüksek oranda güvenilir tespitler yapıldığını göstermektedir (Şekil.12).



a) Kesinlik Grafiği



b) Hatırlama Grafiği

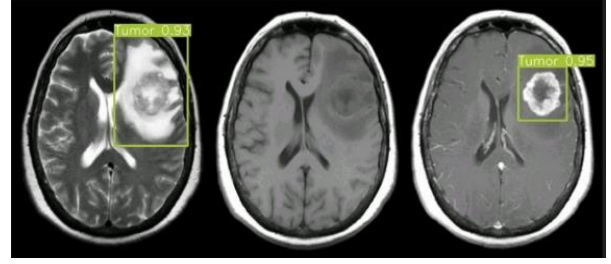


c) F1 Skor Grafiği

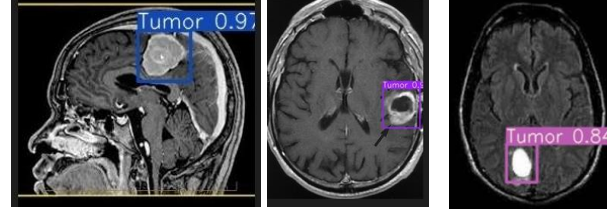
Şekil.12. YOLOv7-tiny için elde edilen Kesinlik, Hatırlama ve F1 grafikleri

Derin öğrenme yöntemi, problemlerin kullanıcıdan bağımsız olarak çok katmanlı bir yapay sinir ağı aracılığıyla çözülmesini sağlayabilmek için yeterli miktarda verinin kullanılmasını gerektirmektedir. Burada 1.0, %100 güvenilirliğe karşılık gelmektedir.

Eğitim sonrasında, önerdiğimiz modele, eğitim setinde kullanılmamış gerçek beyin MR görüntülerinden oluşan 3 test imgesi sorulmuştur. Karar destek modeli, ilk test verisinde tümörü %93 kesinlikle diğer dokulardan ayırmış ve tespit etmiştir (Şekil.13.a).



a) Birinci Test



b) İkinci Test

c) Üçüncü Test

d) Dördüncü Test

Şekil 13. Birinci Test Sonucu

Benzer şekilde farklı kesitlerde ikinci, üçüncü ve dördüncü test verilerinde bulunan farklı tümörleri sırasıyla %97 (Şekil.13.b), %90 (Şekil.13.c) ve %84 (Şekil.13.d) kesinlikle tespit etmiştir.

Yapay zeka yöntemleri her ne kadar 1990'lı yıllarda tıbbi alanında kullanılmış olsa da genelde başarısız olmuştur. Bu, o dönemde henüz yeterince gelişmemiş bilgisayar donanım ve yazılım ve yöntemlerinden kaynaklanmaktadır. Günümüzde, gelişmiş donanımlar ve modeller sayesinde yapay zeka, sağlık alanına da entegre edilmektedir.

Bu, insan sağlığı açısından hataları azaltıp daha iyi sonuçlar elde etme amacıyla karar destek sistemleri olarak kullanılmaktadır. Doğruluğu yüksek karar destek sistemleri insan sağlığını ilgilendiren hataları azaltmakta ve sağlık kuruluşları arasındaki teşhis ve bu teşhislere yönelik olarak karar verilen tedavilerde oluşabilecek farklılıkları en aza indirmektedir.

## Teşekkür

Çalışmada bana destek olan Mehmet Burak ŞEN ve Mehmet Akif ŞEKKELİ'ye katkılarından dolayı teşekkürlerimi sunarım.

## Çıkar Çatışması Beyanı:

Yazarlar tarafından herhangi bir çıkar çatışması belirtilmemiştir.

## Etik Standartlar Beyanı:

Yazarlar bu çalışmada kullanılan materyal ve yöntemlerin etik kurul izni ve yasal-özel izin gerektirmediğini beyan eder.



## Kaynaklar

- [1] Oğuz N., İlnem C., Yener F., 2005. Beyin Tümörlerin Neden Olduğu Psikiyatrik Tablolar: İki Olgu Sunumu. Klinik Psikofarmakoloji Bulteni, 15(1), 18-21.
- [2] Tang R., Zhu Z., Yao H., Li Y., Sun X., Hu H., Xie G., Li Y. 2022. Integrating Medical Code Descriptions and Building Text Classification Models for Diagnostic Decision Support, 2022 IEEE 10th International Conference on Healthcare Informatics (ICHI), pp.612-613, Rochester, MN, USA.
- [3] Lee K.B., Shin H.S., 2019. An Application of a Deep Learning Algorithm for Automatic Detection of Unexpected Accidents Under Bad CCTV Monitoring Conditions in Tunnels, 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML), pp.7-11, Istanbul, Turkey
- [4] Şeker A., Diri B., Balık H.H., 2017. Derin öğrenme yöntemleri ve uygulamaları hakkında bir inceleme. Gazi Mühendislik Bilimleri Dergisi, 3(3), 47-64.
- [5] Yu X., Kuan T.W., Zhang Y., Yan T., 2022. YOLO v5 for SDSB Distant Tiny Object Detection, 10th International Conference on Orange Technology (ICOT), Shanghai, China.
- [6] Samkeliso Suku Dube S.S.; Bhuru A., 2022. Snake Identification System Using Convolutional Neural Networks, 1st Zimbabwe Conference of Information and Communication Technologies (ZCICT), 09-10 November, Harare, Zimbabwe.
- [7] Ali I., Khan A., Waleed M., 2020. A Google Colab Based Online Platform for Rapid Estimation of Real Blur in Single-Image Blind Deblurring, 12th International Conference on Electronics, Computers and Artificial Intelligence (ECAI), 25-27 June, Bucharest, Romania.
- [8] Özgür S.B., 2021. Algoritmalar, Yapay Zeka, Makine Öğrenmesi, Derin Öğrenme ve Uygulamaları: Beşeri Fayda Üretiminin Yazılımlar Tarafından Karşılanması. Ekonomi ve Yönetim Araştırmaları Dergisi, 10(1), 1-29.
- [9] Grewal D.S., 2014. A critical conceptual analysis of definitions of artificial intelligence as applicable to computer engineering, IOSR Journal of Computer Engineering, 16(2), 9-13.
- [10] Bonaccorso G., 2017. Machine learning algorithms, Birmingham, UK: Pacht Publishing. pp.10-17.
- [11] Hao X., Zhang G., Ma S., 2016. Deep learning. International Journal of Semantic Computing, 10(3), 417-439.
- [12] Yılmaz S., Dincer H., Eksin I., Kalenderli O., 2007. Heat control in HVDC resistive divider by PID and NN controllers. Energy Conversion and Management, 48(10), 2739-2748.
- [13] Yılmaz S. Dinçer H., 2004. Development of a Neural Training Program which is Used for Improvement in the Measurement of DC HV Resistive Dividers. NEU-CEE 2004, 2nd International Symposium on Electrical, Electronics and Computer Engineering , (Lefkoşa, Cyprus (Kktc)), 132-137.
- [14] Yılmaz S., Kılıcı S. B., 2020. Design of Training and Development Tool for Feedforward Artificial Neural Networks. Kocaeli Üniversitesi Fen Bilimleri Dergisi, 3(1), 133-146.
- [15] Yılmaz S., Ekinci Ö., Şentürk E., Yuce A. M., 2022. Recurrent Neural Networks For Peak Flow Estimation., Environmental Engineering and Management Journal, 21(6), 927-937.
- [16] Kell D.B., Samanta S.; Swainston N., 2020. Deep learning and generative methods in cheminformatics and chemical biology: navigating small molecule space intelligently. Biochemical Journal, 477(23), 4559-4580.
- [17] Özkan İ. N. İ. K., Ülker E., 2017. Derin öğrenme ve görüntü analizinde kullanılan derin öğrenme modelleri. Gaziosmanpaşa Bilimsel Araştırma Dergisi, 6(3), 85-104.
- [18] Gülcü A., Zeki K. U. Ş., 2019. Konvolüsyonel sinir ağlarında hiper-parametre optimizasyonu yöntemlerinin incelenmesi. Gazi University Journal of Science Part C: Design and Technology, 7(2), 503-522.
- [19] Gandhi R., “R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms”, Towards Data Science, <https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e> (Erişim tarihi: 1 Ocak 2023).
- [20] Girshick R., 2015. Fast r-cnn, In: Proceedings of the 2015 IEEE international conference on computer vision., 1440-1448, 7-13 Dec., Santiago, Chile
- [21] Li F., Johnson J., Yeung S., “Detection and Segmentation”, [http://cs231n.stanford.edu/slides/2017/cs231n\\_2017\\_lecture11.pdf](http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf), (Erişim tarihi: 1 Ocak 2023).

- [22] Jiang A., Yan N., Wang F., Huang H., Zhu H., Wei B., 2019. Visible Image Recognition of Power Transformer Equipment Based on Mask R-CNN, 2019 IEEE Sustainable Power and Energy Conference (iSPEC), 21-23 November , Beijing-China, 657-661.
- [23] Altay A., Yılmaz S., 2023. Classification of T Cells Using YOLO Algorithm. İleri Mühendislik Çalışmaları ve Teknolojileri Dergisi, 3(2), 66-81.
- [24] Wu T.H., Wang T.-W., Liu Y.-Q., Ya-Qi Liu, 2021. 3rd World Symposium on Artificial Intelligence (WSAI), 18-20 June, Guangzhou, China, 24-28.
- [25] Wang C.-Y.; Bochkovskiy A.; Liao H.-Y.M., 2022. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv preprint arXiv:2207.02696
- [26] Hu B., Zhu M., Chen L., Huang L., Chen P., He M., 2022. Tree species identification method based on improved YOLOv7, IEEE 8th International Conference on Cloud Computing and Intelligent Systems (CCIS), 26-28 November, Chengdu, China, 622-627.
- [27] Wulandari A., Sigit R., Bachtiar M.M., 2018. Brain Tumor Segmentation to Calculate Percentage Tumor Using MRI, International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC), 29-30 October, Bali, Indonesia, 292-296.
- [28] Jagan A., 2018. A New Approach for Segmentation and Detection of Brain Tumor in 3D Brain MR Imaging, Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 30 September, Coimbatore, India, 1230-1234.
- [29] <https://www.kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-detection>
- [30] <https://www.kaggle.com/datasets/sartajbhuvaji/brain-tumor-classification-mri>
- [31] <https://github.com/MohamedAliHabib/Brain-Tumor-Detection>
- [32] [https://figshare.com/articles/dataset/brain\\_tumor\\_dataset/1512427](https://figshare.com/articles/dataset/brain_tumor_dataset/1512427)
- [33] <https://www.kaggle.com/datasets/masoudnickparvar/brain-tumor-mri-dataset>
- [34] Bhande A., 2023. “What is underfitting and overfitting in machine learning and how to deal with it”, Medium, <https://medium.com/greyatom/what-is-underfitting-and-overfitting-in-machine-learning-and-how-to-deal-with-it-6803a989c76> (Erişim tarihi: 2 Ocak 2023).
- [35] Şen M.B., 2022. YOLO Algoritması Kullanarak Beyin Tümörü Tespiti, KOÜ, Mühendislik Fakültesi, Elektronik ve Haberleşme Bölümü Lisans Bitirme Tezi.