

İSTANBUL MEDENİYET ÜNİVERSİTESİ
MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

BİL 485 DERİN ÖĞRENME

ATIK SINIFLANDIRMA PROJE RAPORU

Adı Soyadı : Hasan Özgür Özdemir
Numarası : 23120205090

Ders Sorumlusu
Adı Soyadı : İshak Dölek
Ünvanı/Görevi : Dr. Öğr. Üyesi
Tarih : 28.12.2025

İÇİNDEKİLER

Sayfa

1. Şekil Listesi	3
2. Kısaltmalar	4
3. Abstract	5
4. Giriş	5
5. Literatür Taraması	6
6. Materyal ve Method	7
7. Sonuç	15
8. Çalışmanın Kısıtları	15
9. Referanslar	16

ŞEKİL LİSTESİ

Sayfa

1. Figure 1. Manuel çöp ayırma	5
2. Figure 2. Train veri setinden bir örnek	7
3. Figure 3. Object detection vs. Instance seg.	8
4. Table 1. RF-DETR metrik değerleri.	8
5. Figure 4. RF-DETR mAP grafiği	9
6. Table 2. YOLOv11 object detection metrik değerleri	9
7. Figure 5. YOLOv11 object detection grafiği	9
8. Figure 6. YOLOv8 object detection örnek valid görseli	10
9. Figure 7. YOLOv8 object detection sonuç grafikleri	11
10. Table 3. Roboflow 3.0 instance segmentation metrik değerleri	11
11. Figure 8. Roboflow 3.0 instance segmentation mAP grafiği	12
12. Table 4. YOLOv11 segmentation mAP grafiği	12
13. Figure 9. YOLOv11 segmentation metrik değerleri	12
14. Figure 10. YOLOv8-seg örnek valid görseli	13
15. Figure 11. YOLOv8-seg sonuç grafikleri	14
16. Figure 12. Mobil uygulamadan örnek 1	14
17. Figure 13. Mobil uygulamadan örnek 2	14

KISALTMALAR

NLP	:	Natural Language Processing
TACO	:	Trash Annotations in Context
YOLO	:	You Only Look Once
R-CNN	:	Region-based Convolutional Neural Network
mAP	:	mean Average Precision
WaRP	:	Waste Recycling Plant Dataset
RF-DETR	:	Roboflow Detection Transformer

Abstract

Zaman geçtikçe artan tüketim, halkın yeteri kadar bilinçlendirilememesi sebebiyle geri dönüşüm süreçlerinin yetersiz kalması atık yönetimini ciddi bir çevresel sorun haline getirmiştir. Atık sınıflandırma projesiyle geri dönüşüm oranlarını artırarak doğal kaynakların daha verimli kullanılması amaçlanmıştır. Bu sayede enerji tasarrufu ve ekonomik kazanç sağlanır. Bu çalışmada atık sınıflandırma için bilgisayarlı görü (computer vision) ve derin öğrenme (deep learning) yöntemleri kullanılarak bir model eğitilmiş ve mobil kullanım için Android uygulama üzerinde test edilmiştir. Temel veri seti olarak TACO kullanılmış, tarafımda toplanan ek görüntülerle genişletilmiş ve veri çeşitliliğini artırmak amacıyla Roboflow üzerinde çeşitli augmentasyon teknikleri uygulanmıştır. Model eğitimi sürecinde Google Colab ve Roboflow ortamlarında YOLOv11, YOLOv8, RF-DETR, Roboflow 3.0 Instance Segmentation mimarileri kullanılmıştır. Eğitimler segmentasyon tabanlı yapılsa da mobil uygulamaya geçişte object detection tabanlı model tercih edilmiştir. mAP@0.5 değerleri karşılaştırıldığında en yüksek başarı RF-DETR modelinde %53.75 değeri ile elde edilmiş, diğer modellerde de %31-33 aralığında sonuçlar gözlemlenmiştir. Çalışmanın çıktıları sonucunda doğrudan ürün olarak kullanılabilecek doğruluk seviyesinde olmasa da, gelecek çalışmaların geliştirilmesine katkı sunmak amaçlanmıştır.

Keywords: Computer vision, Derin Öğrenme, Segmentasyon, Nesne tespiti, Atık sınıflandırma, TACO dataset, YOLO, mAP, Mobil uygulama

1. Giriş

Dünyada uzun yıllardır devam etmekte olan nüfus artışı, kentleşme ve sanayileşmenin tesiriyle beraber yüksek oranda kaynak kullanılmakta ve bunun neticesinde de yoğun miktarda atık oluşmaktadır [1]. Atıkların doğru yönetimi, sürdürülebilir ve yaşanabilir şehirler inşa etmek için olmazsa olmazdır, ancak birçok gelişmekte olan ülke ve şehir için hala bir zorluk olmaya devam etmektedir [2]. Çevresel atıkların sınıflandırılması ve geri dönüştürülebilir atıkların daha hızlı bir şekilde doğaya kazandırılması, sürdürülebilir bir gelecek için temel oluşturmaktadır. Ayrıca, ham madde üretiminin düşmesi enerji tasarrufu sağlamak ve doğal kaynakların korunmasına yardımcı olmaktadır. Geleneksel yöntemlerle çevresel atıkların toplanması ve ayrıştırılması süreci, manuel işlemlerle zaman alıcı bir süreçtir [3].



Figure 1. Manuel çöp ayırma [4]

Hızla gelişen yapay zeka teknolojilerini bu problemi çözmek için de kullanmak önemli ölçüde fayda getirecektir.

Bilgisayarlı görme (computer vision), makinelerin görüntü ve videlardan anlamlı veriler yorumlamasını, analiz etmesini ve çıkarmasını sağlar. Bu yapay zeka alanı nesneleri, insanları ve desenleri yüksek doğrulukla tanımak için derin öğrenme ve sinir ağlarını kullanır. Başka bir deyişle, insan görüşünü ve görsel verileri yorumlama bilişsel yeteneğini taklit eder [5].

Bu çalışmada bilgisayarlı görü alanında nesne tespiti yaygın şekilde kullanılan ve hız ile doğruluk arasında iyi bir denge sağlayan YOLO ailesi kullanılmıştır [6, 7]. Ayrıyeten bilgisayarla görü modelleri oluşturmak ve dağıtmak için tasarlanmış bir araç paketi olan Roboflow'un sunduğu Roboflow 3.0 ve RF-DETR modelleri kullanılmıştır [8]. Eğitim için Roboflow ve Google Colab ortamları tercih edilmiştir.

2. Literatür Taraması

TACO, atıkların tespiti ve segmentasyonu için hazırlanmış, kitlesel katkılarla büyüyen açık bir görüntü veri kümesidir. **1500 görüntü ve 4784 anotasyon** ile **MASK R-CNN** kullanılarak yapılan çalışmada iki farklı görev üzerine çalışılmıştır: **sınıfsız atık tespiti (TACO-1)** ve **10 sınıflı atık sınıflandırması (TACO-10)**. Deneylerde dört katlı çarpraz doğrulama ile yapılmış ve farklı skor fonksiyonları arasından en iyi sonuç **ratio score** ile elde edilmiş; TACO-1 için **%26.1**, TACO-10 için **%19.4** ortalama doğruluk (AP) değerine ulaşılmıştır. Proença ve Simões tarafından geliştirilen TACO veri seti Mart 2020'de arXiv üzerinde yayımlanmıştır [9].

Shroff, Desai ve Garg (2023) tarafından yapılan “*YOLOv8-based Waste Detection System for Recycling Plants: A Deep Learning Approach*” adlı çalışmada geri dönüşüm tesislerindeki konveyör bantlarından toplanan endüstriyel verilerle oluşturulmuş olan **WaRP** veri seti üzerinde farklı YOLOv8 modelleri test edilmiş; **YOLOv8s**, küçük veri setlerinde düşük işlem süresiyle başarılı sonuçlar elde ederken, **YOLOv8l** modeli ile daha yüksek işlem süresi gerektirmesine rağmen yaklaşık **%59 mAP@0.5** ile en yüksek doğruluk oranı elde edilmiştir [10].

Vieria, Susanto, Purwanto ve Ario (2024), “*Enhancing Waste Classification with YOLOv8 Models for Efficient and Accurate Sorting*” adlı çalışmalarında, Endonezya'daki atık yönetim sorununu ele almışlar ve manuel yöntemlerin yetersizliğini aşmak için derin öğrenme tabanlı yaklaşım önerilmiş. Bu amaçla **3500 etiketli görüntü** ile **YOLOv8 mimarisinin üç varyantı** (nano, small, medium) kullanılarak **altı kategoride** atık sınıflandırılması yapılmış ve YOLOv8 modellerinin yüksek doğruluk sağladığı sonucuna varılmıştır. Özellikle **YOLOv8-nano** modelinin en kısa eğitim süresiyle **%89** doğruluk elde ettiği raporlanmıştır [11].

3. Materyal ve Method

Bu bölümde sırasıyla kullanılan veri seti ve modellerden bahsedilecektir.

3.1 Veri Seti

Çalışmada veri seti Roboflow ortamında toplanmıştır. Temel veri seti olarak TACO ve tarafımda toplanıp etiketlenen verilerle birlikte toplamda 1873 görsel elde edilmiştir. Roboflow’da versiyon oluştururken Auto-Orient ve 640x640 piksele stretch yöntemiyle yeniden boyutlandırma preprocess adımları yapılmıştır. Ayrıca augmentation kapsamında her bir eğitim örneği için 3 farklı çıktı üretilmiş; bu kapsamda flip horizontal, -15° ile 15° arasında rotation, $\%-20$ ile $\%20$ arasında parlaklık değişimi, 1.5 piksele kadar blur ve pikselin $\%0.1$ ’ine kadar gürültü ekleme yöntemleri kullanılmıştır. Bu yöntemler neticesinde 4399 görsel elde edilmiş bunların 3789’u train 414’ü valid ve 196’sı test görsellerinden oluşmaktadır. Görseller *Aluminium foil*, *Can*, *Carton*, *Cup*, *Glass bottle*, *Metal bottle cap*, *Paper*, *Plastic bottle*, *Plastic bottle cap*, *Plastic container*, *Plastic film*, *Plastic lid*, *Pop tab*, *Straw*, *Styrofoam piece*, *Wrapper* ve *Other* olacak şekilde toplam 17 sınıfa ayrılmıştır.



Figure 2. Train veri setinden bir örnek

3.2 Modeller

Bu bölümde kullanılan modeller ve bu modelleri eğitilirken kullanılan iki temel yaklaşımdan: object detection ve instance segmentation bahsedilecektir.

3.2.1 Object detection vs. Instance Segmentation

Hem instance segmentation hem de object detection modelleri, bir nesnenin görüntüdeki konumunu belirler. Ancak, object detection yalnızca nesnenin sınırlayıcı kutusunu tahmin eder [12].

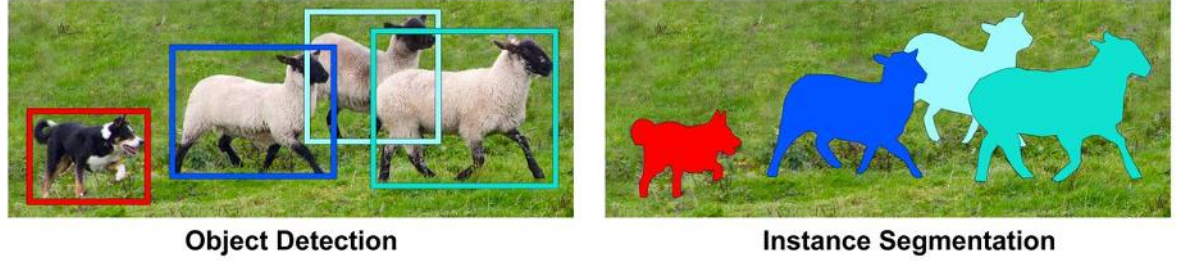


Figure 3. Object detection vs. Instance seg. [13]

Bu çalışmada kıyaslama açısından hem Object detection hem de Instance segmentation yaklaşımları kullanılmıştır. Object detection için; Roboflow ortamında RF-DETR, YOLOv11 mimarileri Google Colab ortamında ise YOLOv8 kullanılmıştır. Instance segmentationda ise; yine Roboflow ortamında Roboflow 3.0 Instance Segmentation ve YOLOv11, Colab'de ise YOLOv8 mimarileri kullanılmıştır.

3.2.2 Object Detection Modelleri

Bu bölümde object detection yaklaşımıyla eğitilen modellerden bahsedilecektir.

3.2.2.1 RF-DETR

RF-DETR, gerçek zamanlı object detection için tasarlanmış transformer tabanlı bir mimaridir [14]. Çalışmada RF-DETR ile toplamda 29 epoch yapılmış 7.epochda %53.75 ile tüm çalışmanın en iyi sonucu elde edilmiştir.

Table 1. RF-DETR metrik değerleri (son epoch)

Metrikler	Sonuçlar
mAP@50	%50.8
Precision	%61.8
Recall	%47.0

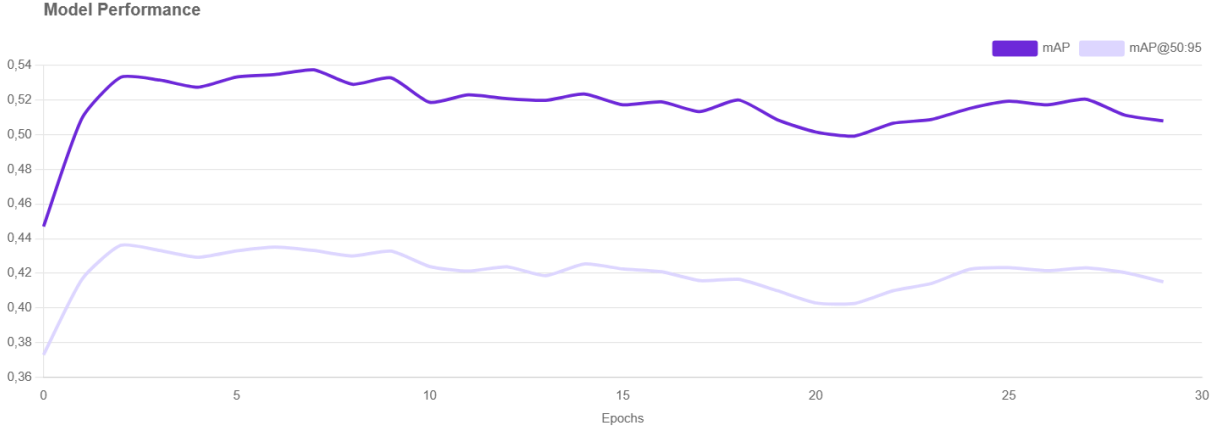


Figure 4. RF-DETR mAP grafiği

3.2.2.2 YOLOv11 Object Detection

Çalışmada Roboflow ortamında YOLOv11 ile toplamda 199 epoch yapılmış, en iyi sonuç 187.epochda %31.46 ile elde edilmiştir.

Table 2. YOLOv11 object detection metrik değerleri (son epoch)

Metrikler	Sonuçlar
mAP@50	%31.4
Precision	%40.9
Recall	%34.1

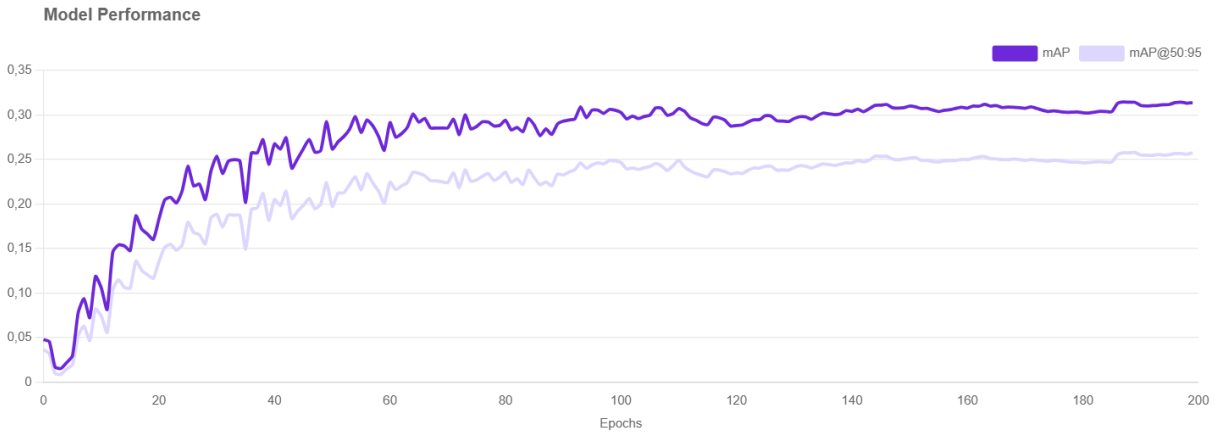


Figure 5. YOLOv11 object detection grafiği

3.2.2.3 YOLOv8 Object Detection

Google Colab ortamında gerçekleştirilen bu eğitimde model olarak YOLOv8s pre-trained ağırlıkları (yolov8s.pt) kullanılmıştır. Eğitim parametreleri şu şekilde belirlenmiştir: eğitim süresi 100 epoch, görüntü boyutu 640x640, batch boyutu 32, optimizör olarak AdamW, başlangıç öğrenme hızı (lr0) 0.003 ve warmup_epochs 3 olarak ayarlanmıştır. Patience 25 olarak ayarlanmış, model 69.epochda en iyi [mAP@0.5](#) değerini %31.7 olarak elde ettiğinden ve 25 epoch boyunca artış sağlanmadığından 94.epochda early stopping ile eğitim durdurulmuştur. Mobil kullanımda bu model kullanılmış detaylar **sonuç** bölümünde verilmiştir.



Figure 6. YOLOv8 object detection örnek valid görseli

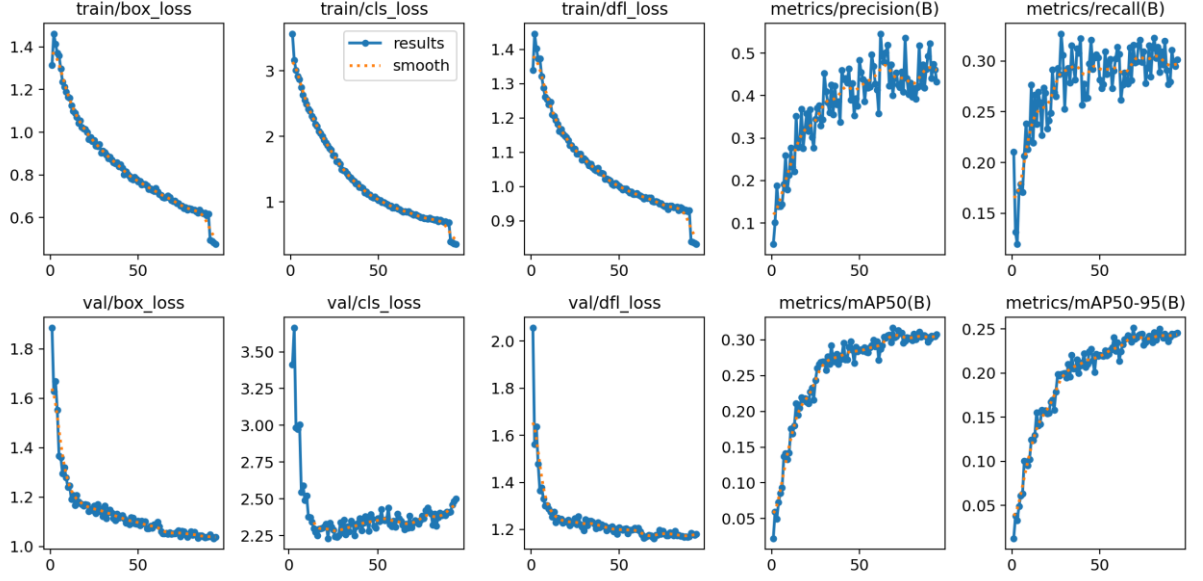


Figure 7. YOLOv8 object detection sonuç grafikleri

3.2.3 Instance Segmentation Modelleri

Bu bölümde instance segmentation yaklaşımıyla eğitilen modellerden bahsedilecektir.

3.2.3.1 Roboflow 3.0 Instance Segmentation (Fast)

Roboflow'un içinde *Roboflow 3.0 Instance Segmentation (Fast)* adında bir varsayılan network var ve bu model Roboflow'un segmentasyon projelerinde kullanılabiliyor. Bu modelde yaptığım eğitimde toplamda 178 epoch içerisinde en iyi sonuç %32.08 ile 139.epochda elde edilmiştir.

Table 3. Roboflow 3.0 instance segmentation metrik değerleri (son epoch)

Metrikler	Sonuçlar
mAP@50	%31.4
Precision	%39.6
Recall	%31.9

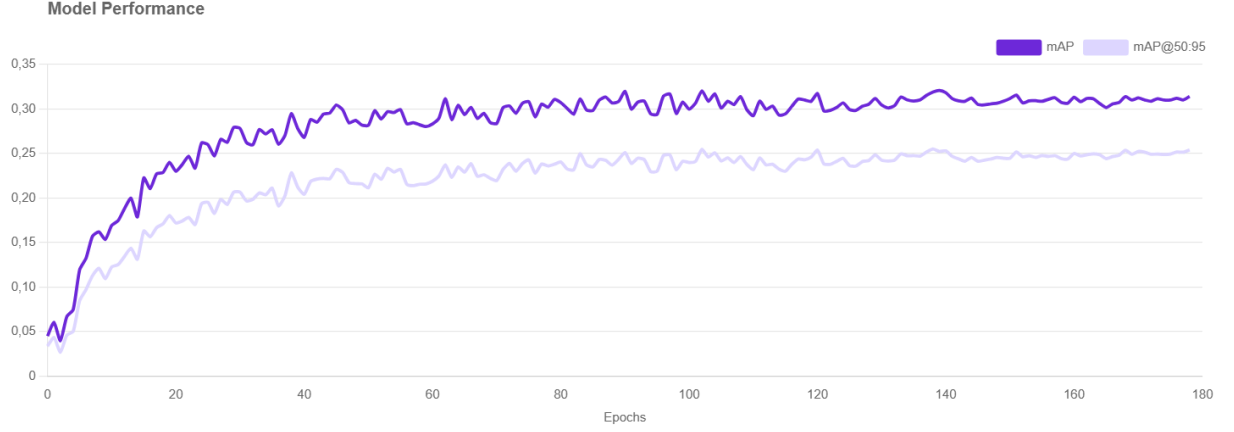


Figure 8. Roboflow 3.0 instance segmentation mAP grafiği

3.2.3.2 YOLOv11 Segmentation

Çalışma Roboflow ortamında toplamda 158 epoch olacak şekilde yapılmış ve en iyi sonuç son epochda %31.96 olarak elde edilmiştir.

Table 4. YOLOv11 segmentation merik değerleri

Metrikler	Sonuçlar
mAP@50	%32.0
Precision	%46.5
Recall	%31.3

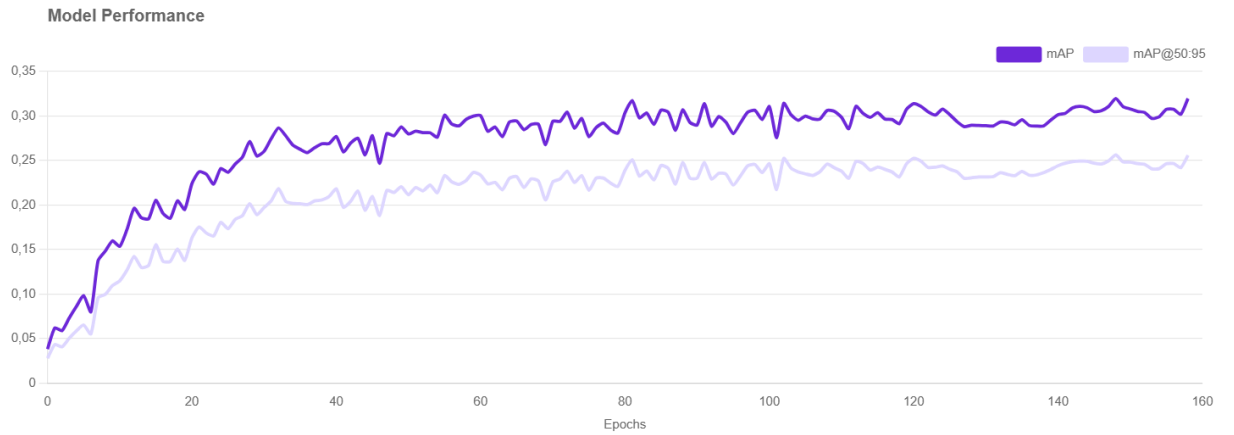


Figure 9. YOLOv11 segmentation mAP grafiği

3.2.3.3 YOLOv8 Segmentation

Google Colab ortamında gerçekleştirilen bu eğitimde model olarak YOLOv8s-seg pre-trained ağırlıkları (yolov8s-seg.pt) kullanılmıştır. Eğitim parametreleri şu şekilde belirlenmiştir: eğitim süresi 100 epoch, görüntü boyutu 640x640, batch boyutu 8, optimizier olarak AdamW, başlangıç öğrenme hızı (lr0) 0.005 ve patience 20 olarak ayarlanmıştır. Model en iyi [mAP@0.5](#) değerini 92.epochda %31.57 olarak almıştır.



Figure 10. YOLOv8-seg örnek valid görseli

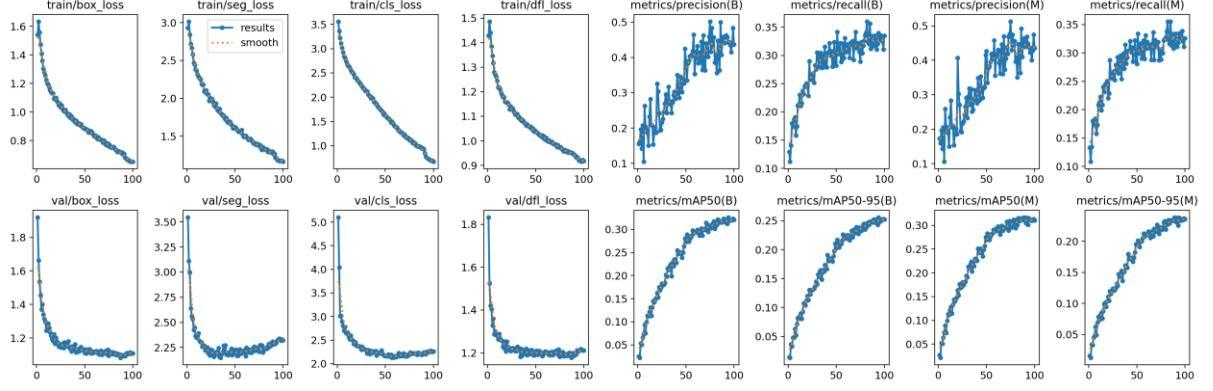


Figure 11. YOLOv8-seg sonuç grafikleri

3.3 Mobil Uygulama ve Gerçek Zamanlı Kullanım

Çalışmanın gerçek zamanlı uygulanabilirliğinin test edilmesi için uygulama kolaylığı açısından mobil cihaz tercih edilmiştir. Model olarak hızlı çıkarım süresi için YOLOv8s gibi hafif bir mimari tercih edilmiştir. Mobil cihazda kullanım için GitHub üzerinde açık kaynaklı *AorohiSingla/Object-Detection-Android-App* projesi [15] temel alınmıştır. Temel alınan projede veri seti çalışmanın veri setiyle değiştirilmiş, sınıf etiket dosyaları assets klasörüne eklenmiş ve model .tflite formatına dönüştürülerek uygulamaya entegre edilmiştir.



Figure 12. Mobil uygulamadan örnek 1



Figure 13. Mobil uygulamadan örnek 2

4. SONUÇ

Bu çalışmada TACO veri seti ve tarafımda toplanıp, etiketlenen görüntüler kullanılarak farklı derin öğrenme mimarileriyle bir atık tespiti çalışması yapılmıştır. Hem object detection hem de segmentation yaklaşımlarıyla YOLOv8, YOLOv11, Roboflow 3.0 Instance Segmentation (Fast) ve RF-DETR modelleri eğitilmiştir. Transformer tabanlı RF-DETR %53.75 ile en iyi [mAP@0.5](#) değerini verirken diğer modeller %31-33 aralığında sonuçlar vermiştir. Gerçek zamanlı kullanım testi için YOLOv8s object detection modeli tercih edilmiştir. Çalışma genel olarak derin öğrenme modellerinin uygulanabilirliğini ortaya koymaktadır ve gelecek çalışmalara ışık tutmak amaçlanmıştır.

5. ÇALIŞMANIN KISITLARI

Bu tarz atık sınıflandırma gibi çalışmalarda önemli kısıtlılıklar vardır. En büyük kısıt görsellerin çok çeşitli ortamlarda (çayır, çimen, toprak vb.) gibi ortamlarda çekilmiş olmasının atıkların anlaşılmasını zorlaştırmasıdır. Bir diğer önemli kısıt ise bardak gibi benzer formlara sahip nesnelerde -örneğin plastik, karton, strafor veya cam- malzeme farklılıklarının görsel açıdan ayırt edilmesinin güç olmasıdır, görüntü sayısını artırarak bu kısıtların önüne geçilebilir. Bu çalışmada 1873 görüntü yeterli gelmemiş ve doğruluk oranı YOLO ailesinde %31-32 dolaylarında kalmıştır. Farklı açılardan çekilmiş görüntülerle veri setini destekleyerek, farklı preprocess adımları ya da farklı modeller kullanılarak alınan doğruluk oranı artırılabilir.

REFERANSLAR

- [1] Varol, S., Hark, C., & Kızılloluk, S. (2022). *Geri dönüştürülebilir atıkların sınıflandırılması*. ResearchGate. https://www.researchgate.net/profile/Cengiz-Hark/publication/366153733_GERI_DONUSTURULEBILIR_ATIKLARIN_SINIFLANDIRILMASI/links/63933bd2e42faa7e75ace8ad/GERI-DOeNUeSTUeRUeLEBILIR-ATIKLARIN-SINIFLANDIRILMASI.pdf
- [2] World Bank. (2022). *Solid waste management*. <https://www.worldbank.org/en/topic/urbandevelopment/brief/solid-waste-management>
- [3] Salur, M. U., Elmas, N., Koçak, A. N., & Kaymaz, M. (2024). Derin öğrenme ile çevresel atıkların sınıflandırılmasına dayalı akıllı çöp konteyneri tasarımı ve prototipinin geliştirilmesi. *Adıyaman Üniversitesi Mühendislik Bilimleri Dergisi*. <https://doi.org/10.54365/adyumbd.1557588>
- [4] Struck, D. (2020). *The resiliency of recycling*. <https://dougstruck.com/journalism/the-resiliency-of-recycling/>
- [5] Microsoft Azure. (n.d.). *What is computer vision?*. <https://azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-computer-vision>
- [6] Kotthapalli, M., Ravipati, D., & Bhatia, R. (2025). *YOLOv1 to YOLOv11: A comprehensive survey of real-time object detection innovations and challenges*. arXiv preprint arXiv:2508.02067. <https://doi.org/10.48550/arXiv.2508.02067>
- [7] Terven, J., Córdova-Esparza, D. M., & Romero-González, J. A. (2023). A comprehensive review of YOLO architectures in computer vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Machine Learning and Knowledge Extraction*, 5(4), 1680–1716. <https://doi.org/10.3390/make5040083>
- [8] Ultralytics. (n.d.). *Roboflow integrations*. <https://docs.ultralytics.com/tr/integrations/roboflow/>
- [9] Proença, P. F., & Simões, P. (2020). *TACO: Trash Annotations in Context for Litter Detection*. arXiv. <https://arxiv.org/pdf/2003.06975>
- [10] Shroff, M., Desai, A., & Garg, D. (2023). *YOLOv8-based waste detection system for recycling plants: A deep learning approach*. IEEE. <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=10331643>
- [11] Viera, D., Susanto, R., Purwanto, E. S., & Ario, M. K. (2024). *Enhancing waste classification with YOLOv8 models for efficient and accurate sorting*. *Procedia Computer Science*. <https://www.sciencedirect.com/science/article/pii/S1877050924031247>
- [12] Roboflow. (2025). *Instance segmentation with Roboflow*. <https://blog.roboflow.com/instance-segmentation/>
- [13] MathWiki. (2020). *User:X93ma*. <https://wiki.math.uwaterloo.ca/statwiki/index.php?title=User%3AX93ma&ref=blog.roboflow.com>

- [14] Roboflow. (2025). *RF-DETR: Roboflow Detection Transformer*.
<https://blog.roboflow.com/rf-detr/>
- [15] Singla, A. (2024). *Object detection Android app*. GitHub.
<https://github.com/AaroHiSingla/Object-Detection-Android-App>