



**MİKROSKOBİK GÖRÜNTÜLER ÜZERİNDE  
DERİN ÖĞRENME KULLANARAK GERÇEK  
ZAMANLI PROTOZOA TESPİTİ**

**2024  
DOKTORA TEZİ  
BİLGİSAYAR MÜHENDİSLİĞİ**

**İdris KAHRAMAN**

**Tez Danışmanı  
Prof. Dr. İsmail Rakıp KARAŞ**

**MİKROSKOBİK GÖRÜNTÜLER ÜZERİNDE DERİN ÖĞRENME  
KULLANARAK GERÇEK ZAMANLI PROTOZOA TESPİTİ**

**İdris KAHRAMAN**

**Tez Danışmanı  
Prof. Dr. İsmail Rakıp KARAŞ**

**T.C.  
Karabük Üniversitesi  
Lisansüstü Eğitim Enstitüsü  
Bilgisayar Mühendisliği Anabilim Dalında  
Doktora Tezi  
Olarak Hazırlanmıştır**

**KARABÜK  
Ocak 2024**

İdris KAHRAMAN tarafından hazırlanan “MİKROSKOBİK GÖRÜNTÜLER ÜZERİNDE DERİN ÖĞRENME KULLANARAK GERÇEK ZAMANLI PROTOZOA TESPİTİ” başlıklı bu tezin Doktora Tezi olarak uygun olduğunu onaylarım.

Prof. Dr. İsmail Rakıp KARAŞ

.....

Tez Danışmanı, Bilgisayar Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından Oy Birliği ile Bilgisayar Mühendisliği Anabilim Dalında Doktora tezi olarak kabul edilmiştir. 15/01/2024

Ünvanı, Adı SOYADI (Kurumu)

İmzası

Başkan : Prof. Dr. İsmail Rakıp KARAŞ (KBÜ)

.....

Üye : Doç. Dr. Muhammed Kâmil TURAN (KBÜ)

.....

Üye : Doç. Dr. Ümit ATİLA (GÜ)

.....

Üye : Dr. Öğr. Üyesi Yasin ORTAKCI (KBÜ)

.....

Üye : Dr. Öğr. Üyesi Muhammet ÇAKMAK (SNÜ)

.....

KBÜ Lisansüstü Eğitim Enstitüsü Yönetim Kurulu, bu tez ile, Doktora derecesini onamıştır.

Doç. Dr. Zeynep ÖZCAN

.....

Lisansüstü Eğitim Enstitüsü Müdürü

*“Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim.”*

İdris KAHRAMAN

## ÖZET

**Doktora Tezi**

### **MİKROSKOBİK GÖRÜNTÜLER ÜZERİNDE DERİN ÖĞRENME KULLANARAK GERÇEK ZAMANLI PROTOZOA TESPİTİ**

**İdris KAHRAMAN**

**Karabük Üniversitesi**

**Lisansüstü Eğitim Enstitüsü**

**Bilgisayar Mühendisliği Anabilim Dalı**

**Tez Danışmanı:**

**Prof. Dr. İsmail Rakıp KARAS**

**Ocak 2024, 58 sayfa**

Tatlı sulardan protozoa tespiti ve sınıflandırılması ve mikroskopik görüntüleme, çevresel izleme, parazitoloji, bilim, biyolojik süreçler ve bilimsel araştırmalarda kritik bileşenlerdir. Suyun bakteriyel ve parazitik kontaminasyonu toplum sağlığında önemli bir rol oynamaktadır. Geleneksel yöntemler genellikle manuel tanımlamaya dayanır, bu da zaman alıcı analizlere ve sınırlı ölçeklenebilirliğe neden olur. Bu çalışmada, olağanüstü hızı ve doğruluğu ile bilinen etkili bir derin öğrenme modeli olan YOLO algoritmasını kullanarak gerçek zamanlı bir protozoa tespit eden bir model önerilmektedir. Çalışmada kullanılan veri seti, tatlı sularda bulunan ve farklı şekil, boyut ve hareketlere sahip olan Bdelloid Rotifera, Sty-lonychia Pustulata, Paramecium, Hypotrich Ciliate, Colpoda, Lepocinclis Acus ve Clathrulina Elegans gibi protozoa türlerinin görüntülerinden oluşmaktadır. Çalışmamızın en önemli özelliklerinden biri, yağmur suyu ve su birikintileri gibi çeşitli su kaynaklarından farklı kültürler oluşturarak bir veri kümesi oluşturmaktır. Ağ mimarimiz, protozoanın

tespitini optimize etmek için dikkatlice uyarlanmıştır ve tek hücreli canlıların hassas bir şekilde tespitini ve sınıflandırılmasını sağlar. Modelin başarısını doğrulamak için, gerçek dünya mikroskopik görüntü veri kümeleri kullanılarak kapsamlı deneyler gerçekleştirilmiştir. Sonuçlar, YOLO tabanlı modelin yüksek tespit doğruluğuna ulaştığını ve hız ve hassasiyet açısından geleneksel yöntemlerden önemli ölçüde daha iyi performans gösterdiğini ortaya koymaktadır. Modelin gerçek zamanlı çalışma yetenekleri, büyük ölçekli veri kümelerinin hızlı bir şekilde analiz edilmesini sağlayarak dinamik çalışmalar ve zamana duyarlı uygulamalar için son derece uygun hale getirmektedir. Değerlendirme parametreleri f1-skor 0.95, ortalama kesinlik 0.92, duyarlılık 0.99 ve mAP 0.983 olarak elde edilmiştir. Önerilen model %98 doğruluk oranına ulaşmıştır. Yüksek verimliliğe ulaşıldıktan sonra, modelin test edilmesini sağlamak için bir masaüstü uygulaması geliştirilmiştir. Önerilen modelin hızı ve doğruluğu, paramesyoloji/parazitoloji çalışmaları için destekleyici bir araçtan su kalitesi değerlendirmelerine kadar çeşitli alanlar için önemli etkilere sahiptir.

**Anahtar Sözcükler :** Derin öğrenme, yapay zekâ, protozoa, YOLO, tıbbi görüntü işleme

**Bilim Kodu** : 92432

## **ABSTRACT**

**Ph. D. Thesis**

# **REAL-TIME PROTOZOA DETECTION USING DEEP LEARNING ON MICROSCOPIC IMAGES**

**İdris KAHRAMAN**

**Karabük University  
Institute of Graduate Programs  
Department of Computer Engineering**

**Thesis Advisor:**

**Prof. Dr. İsmail Rakıp KARAŞ**

**January 2024, 58 pages**

Detection and classification of protozoa from freshwater and microscopic imaging are critical components in environmental monitoring, parasitology, science, biological processes and scientific research. Bacterial and parasitic contamination of water plays an important role in public health. Traditional methods often rely on manual identification, resulting in time-consuming analysis and limited scalability. In this study, we propose a real-time protozoa detection model using the YOLOv4 algorithm, a state-of-the-art deep learning model known for its exceptional speed and accuracy. The dataset used in the study consists of images of protozoa species such as Bdelloid Rotifera, Sty-lonychia Pustulata, Paramecium, Hypotrich Ciliate, Colpoda, Lepocinclis Acus and Clathrulina Elegans, which are found in freshwater and have different shapes, sizes and movements. One of the most important features of our work is to build a dataset by creating different cultures from various water sources such as rainwater and puddles. Our network architecture has been carefully

adapted to optimize the detection of protozoa and enables precise detection and classification of unicellular organisms. To validate the success of the model, extensive experiments were conducted using real-world microscopic image datasets. The results reveal that the YOLOv4-based model achieves exceptional detection accuracy and significantly outperforms traditional methods in terms of speed and precision. The model's real-time operating capabilities enable large-scale datasets to be analyzed quickly, making it highly suitable for dynamic studies and time-sensitive applications. The evaluation parameters were obtained as f1-score 0.95, precision 0.92, average recall 0.99 and mAP 0.983. The proposed model achieved 98% accuracy. After achieving high efficiency, a desktop application was developed to test the model. The speed and accuracy of the proposed model has important implications for various fields, ranging from a supporting tool for paramesiology/parasitology studies to water quality assessments.

**Key Word** : Deep learning, artificial intelligence, protozoa, YOLO, medical image processing

**Science Code** : 92432



## TEŐEKKÜR

Bu tez alıőmasının planlanmasında, araőtırılmasında, yürütülmesinde ve oluşumunda ilgi ve desteęini esirgemeyen, engin bilgi ve tecrübelerinden yararlandıęım, yönlendirme ve bilgilendirmeleriyle alıőmamı bilimsel temeller ışığında őekillendiren sayın hocam Prof. Dr. İsmail Rakıp KARAAŐ'a sonsuz teőekkürlerimi sunarım.

alıőmanın baőlangıcından itibaren bilgi birikimiyle, tezimin izleme aőamasında ve deney sürecinde katkı saęlayan Do. Dr. Muhammed Kâmil TURAN'a ve deęerli bilgilerini paylaőan Do. Dr. Ümit ATILA'ya teőekkür ederim.

Bu tez alıőması Karabük Üniversitesi Bilimsel Araőtırma Projesi (KBÜBAP-17-DR-436) kapsamında desteklenmiőtir. Desteklerinden dolayı Karabük Üniversitesi'ne teőekkür ederim.

Sevgili aileme tüm destekleri ve manevi hiçbir yardımı esirgemedен yanımda oldukları için tüm kalbimle teőekkür ederim.

## İÇİNDEKİLER

	<u>Sayfa</u>
KABUL.....	ii
ÖZET.....	iv
ABSTRACT.....	vi
TEŞEKKÜR.....	viii
İÇİNDEKİLER .....	ix
ŞEKİLLER DİZİNİ.....	xi
ÇİZELGELER DİZİNİ .....	xii
KISALTMALAR DİZİNİ.....	xiii
BÖLÜM 1 .....	1
GİRİŞ .....	1
1.1 TEZİN TANIMI .....	1
1.2. TEZİN AMACI .....	1
1.3. ÖZGÜN DEĞER.....	2
1.4. TEZ ORGANİZASYONU .....	3
BÖLÜM 2 .....	5
LİTERATÜR ÖZETİ.....	5
BÖLÜM 3 .....	15
DERİN ÖĞRENME.....	15
3.1. KONVOLÜSYONEL SİNİR AĞLARI (CNN).....	17
3.2. NESNE TANIMA .....	18
3.3. TEK AŞAMALI TESPİT ALGORİTMALARI.....	18
3.3.1 SSD.....	19
3.3.1.1. SSD Framewok .....	19
3.3.1.2. SSD Model.....	20
3.4. İKİ AŞAMALI TESPİT ALGORİTMALARI.....	22
3.4.1. R-CNN.....	22
3.4.2. Fast R-CNN .....	25
3.4.3. Faster R-CNN .....	26

	<b><u>Sayfa</u></b>
3.4.4. Mask R-CNN .....	27
3.5. ÇALIŞMADA KULLANILAN YOLO ALGORİTMASI.....	29
3.5.1. YOLOv4 .....	30
3.5.1.1. Dedektör.....	30
3.5.2. YOLOv5 .....	32
3.5.3. YOLOv6 .....	34
3.6. SINIFLANDIRMA MODELLERİNDE BAŞARI KRİTERLERİ .....	35
BÖLÜM 4 .....	39
DENEYSEL ÇALIŞMA .....	39
4.1. VERİ SETİNİN OLUŞTURULMASI .....	39
4.1.1. Protozoa Kültürünün Oluşturulması .....	39
4.1.2. Kültürlerden Protozoa Görüntülerinin Elde Edilmesi .....	40
4.1.3. Veri Setinin Oluşturulması .....	40
4.2. VERİ ETİKETLEME .....	41
BÖLÜM 5 .....	43
DENEYSEL SONUÇLAR VE TARTIŞMA.....	43
5.1. YOLOV4 ALGORİTMASININ PERFORMANSININ DEĞERLENDİRİLMESİ.....	43
5.2. YOLO Modellerinin Karşılaştırılması.....	48
BÖLÜM 6 .....	51
SONUÇLAR VE DEĞERLENDİRMELER .....	51
KAYNAKLAR .....	53
ÖZGEÇMİŞ .....	588

## ŞEKİLLER DİZİNİ

### Sayfa

Şekil 3.1 Derin öğrenme mimarisi .....	16
Şekil 3.2 SSD'nin birden fazla nesneyi algılamak için çalıştığı şekli .....	20
Şekil 3.3 SDD mimarisi [36].....	21
Şekil 3.4 Konvolüsyon noktası SSD'de ayarlanmış farklı boyutlarda çapa kutuları .	21
Şekil 3.5 Görüntü Segmentasyonu [39] .....	23
Şekil 3.6 Seçici arama metodu için nesne tanıma eğitim süreci [39].....	23
Şekil 3.7 Seçici arama yöntemi ile elde edilen en yüksek skorlarla nesne tespiti .....	24
Şekil 3.8 R-CNN algoritmasının çalışma şekli .....	24
Şekil 3.9 The Fast R-CNN mimarisi .....	26
Şekil 3.10 Mask R-CNN algoritmasının çalışma şekli [14].....	28
Şekil 3.11 Mask R-CNN algoritmasının sonucu [43] .....	28
Şekil 3.12 YOLO algoritmasının şematik gösterimi.....	31
Şekil 3.13 Birleşim üzerinde kesişim (IOU), nesne algılama performansı ölçümü... 31	
Şekil 3.14 YOLOv5'in varsayılan çıkarım akış şeması [50].....	32
Şekil 3.15 YOLOv5'in varsayılan çıkarım akış şeması [50].....	33
Şekil 3.16 (a) YOLOv6'nın boynu (N ve S gösterilmiştir). .....	34
Şekil 4.1 Bdelloid Rotifera protozoa türünün hareket halindeki farklı şekil formları	39
Şekil 4.2 Veri setindeki protozoa türleri .....	41
Şekil 4.3 Makesense.io ile etiketleme işlemi .....	42
Şekil 5.1 Karışıklık matrisi .....	45
Şekil 5.2 Normalize edilmiş karışıklık matrisi.....	45
Şekil 5.3 Protozoa tür tespit test görüntüsü.....	47
Şekil 5.4 mAP ve kayıp grafiği. ....	48

## ÇİZELGELER DİZİNİ

	<b><u>Sayfa</u></b>
Çizelge 2.1 Görüntüler üzerinde bakterilerin tanımlanması üzerine yapılan çalışmaların özeti .....	10
Çizelge 3.1 Hata matrisi elemanları .....	36
Çizelge 5.1 Farklı girdi boyutlarına göre mAP değerleri.....	43
Çizelge 5.2 512x512 girdi boyutunda eğitim sonuçları .....	44
Çizelge 5.3 YOLOv4 ile sınıf başına eğitim sonuçları .....	46
Çizelge 5.4 YOLOv4, YOLOv5, YOLOv6 modellerinin karşılaştırılması .....	50

## KISALTMALAR DİZİNİ

- CNN : Konvolüsyonel Sinir Ağları  
ML : Makine öğrenmesi  
NN : Nöral ağlar  
DL : Derin öğrenme  
DNN : Derin sinir ağları  
FT : Hassas ayar  
TP : Doğru pozitif  
TN : Doğru negatif  
FP : Yanlış pozitif  
FN : Yanlış negatif  
MLP : Çok katmanlı algılayıcı  
mAP : Genel ortalama kesinlik  
AUC : ROC eğrisi altındaki alan (Area under the ROC Curve)  
SSD : Tek atış algılama (Single shot detector)

## BÖLÜM 1

### GİRİŞ

#### 1.1 TEZİN TANIMI

Protozoa, çevre sağlığının göstergeleri ve besin döngüsüne katkıda bulunanlar da dahil olmak üzere çeşitli ekosistemlerde önemli roller oynayan tek hücreli mikroorganizmalardır. Bununla birlikte, doğru ve verimli bir şekilde tespit edilmeleri ve sınıflandırılmaları mikrobiyoloji ve çevre bilimlerinde zorlu görevler olmaya devam etmektedir. Bu tez, You Only Look Once (YOLO) algoritmasını kullanarak mikroskobik görüntülemeden gerçek zamanlı protozoa tespiti için yeni bir yaklaşım önererek bu zorluğun üstesinden gelmeyi amaçlamaktadır.

You Only Look Once (YOLO) algoritması, görüntülerdeki nesnelere tespit etmedeki hızı ve doğruluğu ile bilinen etkili bir nesne tespit algoritmasıdır. Bu araştırma, bu algoritmayı protozoa tespiti bağlamına uyarlayarak, bu mikroorganizmaların doğrudan mikroskobik görüntülerden hızlı ve kesin bir şekilde tanımlanmasını sağlamayı amaçlamaktadır.

Tez, önerilen metodolojinin farklı protozoa türlerini doğru bir şekilde tanımlamadaki etkinliği, büyük hacimli görüntü verilerini işlemek için ölçeklenebilirliği ve çevresel izleme, tıbbi teşhis ve biyolojik araştırmalardaki potansiyel uygulamaları da dahil olmak üzere çeşitli yönlerini araştıracaktır.

#### 1.2. TEZİN AMACI

Bu tezin temel amacı, You Only Look Once (YOLO) algoritmasını kullanarak mikroskobik görüntülemeden gerçek zamanlı protozoa tespiti için sağlam ve verimli

bir çerçeve geliştirmektir. Araştırma spesifik olarak aşağıdaki hedeflere ulaşmayı amaçlamaktadır:

- Protozoa tespit doğruluğunu artırmak için farklı türlere ait oldukça fazla verilere ulaşarak etkin bir veri seti oluşturmak.
- Protozoa türlerini gerçek zamanlı olarak doğru bir şekilde tanımlamak ve sınıflandırmak için protozoa görüntüleri üzerinde eğitilmiş YOLO tabanlı bir algılama modeli geliştirmek.
- Gerçek dünya mikroskobik görüntülerini kullanarak önerilen çerçevenin performansını tespit doğruluğu, hız ve ölçeklenebilirlik açısından değerlendirilmesi.
- Geliştirilen protozoa tespit sisteminin çevresel izleme, tıbbi teşhis ve biyolojik araştırma alanlarındaki potansiyel uygulamalarını araştırmak.

Bu hedeflere ulaşarak, bu araştırma protozoa tespit teknolojilerinin ilerlemesine katkıda bulunmayı, bilim insanlarına ve uygulayıcılara çeşitli ekosistemlerdeki mikrobiyal toplulukları anlamak ve yönetmek için güçlü bir araç sağlamayı amaçlamaktadır.

### **1.3. ÖZGÜN DEĞER**

Dünya tatlı su kaynaklarının gün geçtikçe yetersiz kalması ve artan nüfus, suyun daha dikkatli ve özenli kullanılmasını gerektirmektedir. Suyun bakteriyel ve parazit kontaminasyonu su kaynaklı enfeksiyonların yayılmasına yol açmaktadır. Bu sulardaki bakterilerin tespiti sağlığımız için kritik öneme sahiptir.

Önerilen araştırma, tatlı sularda bulunan mikroskobik görüntülere uygulanan derin öğrenme tekniklerinin gücünden yararlanan yenilikçi bir gerçek zamanlı tek hücreli protozoaların tespit sisteminin tasarımı ve uygulanmasına odaklanmaktadır. Mikroskobik tek hücreli organizmalar olan protozoa, çeşitli ekolojik sistemlerde önemli bir rol oynamakta ve çevre sağlığının göstergeleri olarak hizmet



edebilmektedir. Protozoa tespitine yönelik geleneksel yöntemlerin genellikle zaman alıcı ve manuel olması, otomatik ve verimli bir çözüme ihtiyaç duyulmasına neden olmaktadır.

Önerilen yöntem ile gerçek zamanlı protozoa tespiti kısa sürede ve daha yüksek performansla gerçekleştirilmektedir. Ayrıca geliştirilen bu sistem, paramesiyoloji/parazitoloji çalışmalarına ve Tıp Fakültesi Tıbbi Biyoloji Anabilim Dalı'nda laboratuvar uygulaması olan Tek Hücreli Canlılar dersine kaynak sağlayarak eğitime destek aracı olarak da kullanılabilir. Ayrıca öğrencilerin kendi kendilerini eğitmelerine de yardımcı olmaktadır. Ayrıca, bu tam otomatik sınıflandırıcı, sudaki bakterilerin varlığını, türlerini ve çeşitliliğin yoğunluğunu belirleyebilir. Bu çalışma protozoa teşhis ve türünü tespit etme açısından zaman kaybını en aza indirmesi ve tatlı sularda bakteri tespitinin doğruluğunu artırması açısından literatüre önemli katkı sağlayabilir. Ayrıca çalışmamız için laboratuvar ortamında hazırlanan kültür ve teçhizatla protozoa türlerinin görüntüleri elde edilmiştir.

#### **1.4. TEZ ORGANİZASYONU**

Birinci bölümde “Giriş” kısmına yer verilmiş olup çalışmanın kısa tanımı, tezin amacı, özgün değeri ve tezin organizasyonundan bahsedilmiştir. Burada problem hakkında okuyucu bilgilendirilmiştir ve çözüm önerisi sunulmuştur. İkinci bölümde yer alan “Literatür Özeti” başlığında güncel literatür taraması yapılarak tıbbi görüntülerde bakteri sınıflandırma çalışmalarına yer verilmiştir. Üçüncü bölümde ise “Makine Öğrenmesi” başlığı altında okuyucuya sunulmuştur. Ayrıca, derin öğrenmenin neden protozoa tespit hassasiyeti konusunda uygun bir çözüm olarak kullanılabileceğinden bahsedilmiştir. Bölüm 4’te yer alan “Deneysel Çalışma” kısmında yaptığımız çalışma kullanılan veri setinin oluşturulmasından derin öğrenme modeline kadar kapsamlı bilgilendirme bu bölümde yapılmıştır. Bölüm 5, “Deneysel Sonuçlar ve Tartışma” kısmıdır ve yapılan çalışmalardan sonra elde edilen veriler burada paylaşılıp, yorumlanmıştır. Bölüm 6 ise çalışmamızın “Sonuçlar ve Değerlendirmeler” kısmıdır. Burada tez çalışmamızın genel itibariyle elde ettiği en iyi sonuçlar ve değerlendirmeleri yapılmıştır. Ayrıca, bu çalışmanın literatüre hangi yönlerde katkı

sağladığından ve bu alanda yapılabilecek yeni çalışmalar hakkında da değerlendirmelere yer verilmiştir.

## BÖLÜM 2

### LİTERATÜR ÖZETİ

Dünya tatlı su kaynaklarının gün geçtikçe yetersiz kalması ve artan nüfus, suyun daha dikkatli ve özenli kullanılmasını gerektirmektedir. Tatlı su, insanlar tarafından içme suyu, hayvancılık, tarım arazilerinin sulanması ve su ürünleri yetiştiriciliği gibi faaliyetlerde kullanılmaktadır. Suyun bakteriyel ve parazit kontaminasyonu su kaynaklı enfeksiyonların yayılmasına yol açmaktadır.

Tatlı su kaynaklarından insanlara bulaşan bakteriler bağırsak hastalıkları, anemi, kas ağrısı, rotavirüs ile ilişkili hastalıklar, hepatit, ödem ve ishal gibi sağlık sorunlarına neden olmaktadır [1]. Günlük hayatta kullandığımız içeceklerdeki bakterilerin tespiti sağlığımız için kritik öneme sahiptir.

Günümüzde mikroskopik görüntülerden nesne tespiti ve sınıflandırma işlemleri yoğun bir şekilde yapılmaktadır. Otomatik ve yarı otomatik çalışmalar mevcuttur. Bakterilerin tespiti, tanımlanması ve sınıflandırılması hem çalışma süresi açısından hem de iş gücü açısından önemlidir. Bakterilerin varlığını tespit etmek için çeşitli yöntemler kullanılmaktadır. Protozoaların tespiti ve teşhisine yönelik çalışmalar üç farklı grupta incelenmiştir. İlk olarak laboratuvar teknikleri ile yapılan çalışmalar, ikinci olarak protozoa tespiti için görüntü işleme teknikleri kullanılarak yapılan çalışmalar ve son olarak ise derin öğrenme teknikleri kullanılarak çalışmalar yürütülmektedir.

Derin öğrenme, gerçek zamanlı araç tespiti [2] gibi çeşitli alanların yanı sıra sağlık alanına da önemli katkılarda bulunmuş, tıbbi görüntüleme teşhisi ve görüntü segmentasyonu, ilaç keşfi ve geliştirilmesi, elektronik sağlık kayıtları, genomik ve kişiselleştirilmiş tıp, robotik cerrahi gibi çeşitli alanlarda yenilikçi çözümler ve iyileştirmeler sunmuştur. Tıbbi alanlarında derin öğrenme kullanılarak yapılan

çalıřmalara ařağıdaki bazı örnekler verilebilir: comet assay görüntülerinde DNA hasarları [3], beyaz kan hücrelerinin sınıflandırılması [4], dentinal tübül tıkanıklıklarının sınıflandırılması [5], akciğer kanseri nodül tespiti [6], göğüs enfeksiyonunun sınıflandırılması ve tespiti [7]. Derin öğrenmenin sağık alanındaki uygulamaları gelişmeye devam etmekte ve teşhis, tedavi ve genel hasta bakımında ilerlemeler vaat etmektedir. Bununla birlikte, bu teknolojilerin sağık hizmetleri ortamlarında sorumlu ve etik bir şekilde kullanılmasını sağılamak için veri gizliliğı, yorumlanabilirlik ve düzenleyici hususlar gibi zorlukların ele alınması önemlidir. Bu uygulamalara örnek olarak kan lekesi görüntülerinden sıtma parazitlerini tespit etmek ve sınıflandırmak için yapılan bir çalıřma verilebilir. Bu çalıřmada üç sınıfa ait toplam 1920 görüntü kullanılmıřtır. CNN modeli kullanılarak sırasıyla %95,11 ve %99,59 başarı oranları elde edilmiřtir [8]. Bir başka çalıřmada ise kan hücresi görüntülerinde B-Lenfoblast hücreleri sınıflandırılmıřtır. CNN modeli üzerinde eğıtim verilerek 12.528 görüntü ieren C-NMC veri kümesi kullanılarak model oluřturulmuřtur. %99,4 duyarlılık, %96,7 özgülük, %98,3 AUC ve %98,5 doęruluk elde etmiřlerdir [9]. Bir dięer çalıřmada, COVID-19 enfeksiyonunun řiddetini belirlemek için yapay zeka uygulamaları geliřtirilmiřtir. Klasik makine öğrenimi ve derin öğrenme modelleri klinik, demografik, laboratuvar ve seroloji parametrelerini ieren veriler kullanılarak karřılařtırılmıřtır. Bu çalıřmada, XGBoost algoritması %97,6 ile en yüksek doęruluk sonucunu vermiřtir [10]. B-ALL kanser hücrelerinin sınıflandırılması için verimli bir CNN modeline dayanan bir mobil uygulama geliřtirilmiřtir. Veri hazırlama ve veri artırma için 3242 görüntü yeniden boyutlandırılarak kullanılmıřtır. alıřmada karřılařtırma için CNN tabanlı Efficient-Net, MobileNetV2 ve NASNet Mobile modelleri kullanılmıřtır. MobileNetV2 modeli test verilerinde en yüksek verimi (%100) elde etmiřtir [11].

Mikroskobik görüntüler ışığın rengine ve tonuna göre farklı özelliklere sahip olabilir ve protozoanın hareketliliğı, bazı organizmaların deformasyonu ve sudaki polütanlar da görüntü durumunu etkilemektedir.

Protozoa tespiti için yapılan çalıřmalardan ilki laboratuvar teknikleridir. Bunlardan biri PCR ile fiziksel ve kimyasal spektrum yöntemidir [12–15]. Bu teknikte örnek hazırlama, bölümleme, PCR amplifikasyonu ve tespit adımları uygulanmıřtır.

Örneklerdeki protozoa nesnelere tespiti için bazı farklı dPCR uygulamalarını analiz etmişlerdir. [16,17]'da protozoalar elektrotlar ve elektrokimyasal sensörler kullanılarak tespit edilmiştir. Birkaç elektrokimyasal biyosensör, ortam katmanları ve diğer bazı parçaları içeren bir bakım noktası cihazı geliştirmişlerdir. Karmaşık matrislerle kullanıldığında bile yüksek hassasiyet ve esnekliğe sahip taşınabilir sistemler oluşturmak zordur. [18]'de mikroakışkan empedans sitometri yöntemi uygulanmıştır. Bir diğeri ise moleküler biyolojiye dayalı DNA ve RNA yöntemleridir [19]. Parazitlerin teşhisi için geleneksel yaklaşım mikroskobik incelemeyi içerir. Optimum sonuçlar elde etmek için yoğun çalışma ve tecrübeli uzmanlara ihtiyaç olmasına rağmen önerilen yöntem, özellikle kaynakların sınırlı olduğu ortamlarda protozoa teşhisi için yaygın olarak kullanılabilir. Bununla birlikte, çok sayıda laboratuvar, bağırsak protozoalarının varlığını ve sınıflandırılmasını tutarlı bir şekilde tespit etme konusunda yetkin uzman eksikliğiyle karşı karşıyadır. Bu eksiklik, protozoaları tam olarak tanıma, patojenik olanları patojenik olmayanlardan ayırt etme ve mikroskobik değerlendirmeler sırasında artefaktları ayırt etme yetersizliğine katkıda bulunmakta ve sonuçta bağırsak protozoa tanısının duyarlılığını ve özgüllüğünü tehlikeye atmaktadır.

İkinci olarak, protozoa varlığını tespit etmek için görüntü işleme teknikleri kullanılmıştır. [20]'da önerdikleri yöntem, görüntülerin aktif hale getirilmesi için bir ön işlemden geçirilmesi, bölgelere ayrılması ve özellik çıkarımı için kenar tespiti uygulanması, bölgelerin protozoa veya protozoa olmayan YSA sınıflandırması, bölgelere karar vermek için morfolojik işlemler ve konumu tam olarak belirlenemeyen bölgelere aktif kontur yöntemi uygulanmasından oluşmaktadır. Çalışmada 30 görüntü kullanılmıştır. Önerilen yöntemin zayıf yönü ipliksi yapıyı göz ardı etmesidir. [21]'de parazitler Kenarsız Aktif Kontur ile bölütlenmiştir. Gürültüyü gidermek için morfolojik işlemler ve bir görüntüyü iki alana bölmek için eşikleme yapmışlardır. Veri kümeleri 50 görüntü içerir ve tüm görüntülerin boyutu  $140 \times 140$  pikseldir. Çalışmanın doğruluğu %97,57 ve Yanlış Negatif Oranı %12,04'tür. [22]'de yazarlar mikroskobik görüntülerden sıtma parazitlerini tanımlamak için bir yöntem önermişlerdir. Veri setindeki resimler  $3136 \times 2352$  piksellik 117 mikroskobik alan içermektedir. Başlangıçta, Laplacian filtresi kullanarak bir görüntü gürültüsü giderme işlemi uygulamışlardır. İnce ve kalın kan leke görüntülerini bölütlemek için uyarlanabilir

histogram eşikleme tekniği kullanılmıştır. Ayrıca, bölütlenmiş görüntüleri HSV renk uzayında etiketlemek için 8-bağlantılılık kullanılmıştır. Yazara göre, manuel tespit yönteminde parazit sayısı 576 iken, önerdikleri yöntemle tespit edilen parazit sayısı 627 olmuştur.

Son olarak güncel çalışmalarda Klasik CNN, Retina Net, R-CNN, Fast R-CNN, Faster R-CNN, You Look Only Once (YOLO) ve Single Shot Detector (SSD) gibi algoritmalar dijital sistemlerle görüntülerdeki nesnelere tespitinde öne çıkan derin öğrenme yöntemleri arasında yer almaktadır. [23]'de, salgın patojen vibrio cholera ve plasmodium falciparum'u tespit etmek için klasik bir CNN modeli kullanılmıştır. Eğitim için veri kümesi her iki sınıfa ait 400 görüntü ve test için 80 görüntü içermektedir. CNN'i uygulamak için Tensorflow çatısı yürütülmüştür. CNN modeli 6 evrişimli katmana, ReLU ve  $2 \times 2$  MaxPooling'e sahiptir ve bunu tam bağlı katman takip etmektedir. Son olarak, sistemin sınıflandırma doğruluğu %94 ve doğrulama doğruluğu %97'dir. Başka bir CNN yöntemi Blain ve diğerleri tarafından [24]'te bağırsak protozoa'sını tespit etmek için kullanılmıştır. [25]'te bölge tabanlı konvolüsyonel sinir ağlarına (RCNN) temel olarak segmentasyon odaklı RetinaNet sistemi kullanılmıştır. Yetersiz veri nedeniyle, sekiz sınıf için 69 görüntü ve 117 obje için görüntü iyileştirme yöntemleriyle görüntü çoğaltma yapılmıştır. Ancak bu durum hatalı tespitlere neden olmuştur. Hiyerarşik retina segmentasyonunun ortalama hassasiyeti mAP %93 olarak elde edilmiştir. Yerel minimumlarla karşılaşmayı önlemek için resimler üzerinde rastgele yatay çevirme algoritması uygulanarak veri artırımı yapılmıştır. [26]'te nesne algılama algoritması olarak Tek Atış Algılama (SSD) ve Faster R-CNN yöntemleri kullanılmıştır. Çalışmada  $750 \times 750$  boyutlarında 643 görüntü kullanılmıştır. Faster R-CNN yöntemi, bölge tahminleri oluşturmaktan oluşur ve daha sonra nesne türünü ve sınırlayıcı kutunun konumunu sınıflandırmak için CNN uygulanır. Yöntem, özellikleri belirlemek ve nesnelere kategorize etmek için sınırlayıcı kutu bölgelerine sahiptir. Önerilen ağ, ResNet50 ve ResNet101 üzerine inşa edilmiştir. Performans değerlendirme ölçütü olarak, ResNet101 için mAP %94,48'dir ve çalışma süresi 42,8 ms'dir. Çoklu nesne tespiti için SSD uygulanmıştır. SSD, hızlı tespit ve mobil kullanım nedeniyle seçilmiştir. Önerilen ağ, farklı çözünürlükteki nesnelere hızlı bir şekilde tespit etmek için farklı çözünürlüklere sahip birden fazla özellik

haritasındaki tahminleri birleştirir. SSD (InceptionV2)'nin performans değerlendirme ölçütü olarak mAP %83,97'dir ve tespit süresi 2,24 ms'dir.

Bu yöntemlerin dışında, [27]'de bağırsak parazitlerini tespit etmek ve sınıflandırmak için hibrit bir yöntem uygulanmıştır. Çalışmada, farklı parazit veri tabanı gruplarından alınan görüntülerde 15 farklı parazit türü yer almıştır. Veri kümesi, eğitim için %40, doğrulama için %30 ve test için %30 olarak ayrılmış 12.225 görüntü içermektedir. Çalışma iki bölümden oluşmaktadır. Bunlardan biri manuel özellik çıkarımı (DS1) ve destek vektör sınıflandırmasıdır. İkinci bölümde, görüntü özelliği çıkarma ve sınıflandırma (DS2) için derin sinir ağına dayalı VGG-16 kullanılmıştır. Hibrit model DS1 ve DS2'den oluşmuştur. Verimlilik istenen düzeyde olmadığı tespit edilmiş ve karmaşıklığın basitleştirilmesi gerektiği sonucuna varılmıştır. [28]'de yazarlar leishmaniasis parazit segmentasyonu için tam evrişimli bir ağ modeli olan U-Net'i uygulamışlardır. Veri kümesi leishmania parazitine ait 37 görüntü içermektedir ve bu çalışmada 5 sınıf kullanılmıştır. Performans değerlendirme ölçütü olarak, beş sınıf için ortalama kesinlik değeri %75,32'dir. [25]'de bacillus anthracis bakterisi için  $1536 \times 1536$  çözünürlüğe sahip 954 mikroskopik görüntü içeren veri kümesine CNN tabanlı UNet ve Unet++ modelleri uygulanmıştır. Tüm test görüntüleri için %98'lik bir duyarlılık ve %87'lik bir hassasiyete ulaşmışlardır.

Çizelge 2.1'de protozoa, kolera patojeni, sıtma paraziti ve bağırsak protozoalarının tespitine ilişkin son on yılda yayınlanan bazı makalelerin bir özeti yer almaktadır.

Çizelge 2.1 Görüntüler üzerinde bakterilerin tanımlanması üzerine yapılan çalışmaların özeti

Referans	Organizma Türü	Veri Kümesi	Veri Kaynağı	Sınıf Sayısı	Veri Önileme	Segmentasyon	Performans	Sonuçlar (%)
						/Sınıflandırıcı	Değerlendirm	
[8]	Tatlı su protozoaları	1920 görüntü	Orijinal	4	Segmentasyon, çözünme, yeniden boyutlandırma normalizasyon	Klasik NN, 6-layer CNN	Ortalama kesinlik	NN: 95.11, CNN: 99.59
[20]	Sıtma paraziti	27 görüntü Çözünürlük: 1288 × 966	Orijinal	2 (sıtma paraziti ya da değil)	Gri seviye histogram, eşitleme, medyan filtre	Prewitt Kenar Tespit, ANN	Tespit sayısı	Belirtilmemiş
[21]	Sıtma parazit adayları	50 görüntü Çözünürlük: 1280 × 960	Orijinal	-	Gürültü filtreleme (lowpass filtreleme), kontrast geliştirme (kontrast germe)	Kenarsız Aktif Kontur	Hata matrisi	Doğruluk: 97.57
[23]	Kolera patojeni, sıtma paraziti	400 görüntü	Google kaynaklı görüntüler	2	Uygulanmamış	CNN tabanlı model	Tam görüntü doğruluğu	Doğruluk: 94
[24]	Bağırsak protozoası	372 örneklem	Orijinal	11	Veri artırma (rasgele 250x250 piksel kesitler)	CNN	Toplam duyarlılık	Duyarlılık: 92
[25]	Protozoa	Farklı çözünürlükte 69 görüntü	Orijinal	8	Veri artırma	Hiyerarşik RetinaNet	İkili doğruluk, hassasiyet ve duyarlılık	İkili doğruluk: 98, hassasiyet: 93, duyarlılık: 95



[26]	Sıtma paraziti	643 görüntü Çözünürlük: 750 × 750	Orijinal	2 (sıtma paraziti ya da değil)	Veri artırma (yatay döndürme)	Faster R-CNN, SSD, RetinaNet	Bütün görüntüde sıtma bakterisi tespiti	Faster R-CNN mAP: 94.48, SSD mAP: 83.97, RetinaNet mAP: 77.46
[27]	Bağırsak paraziti	LAR-2: 1852, EGG-9: 12,691, PRO-7: 37,375	LAR-2, EGG-9, PRO-7 veri seti	LAR-2: 2, EGG-9: 9, PRO-7: 7	IFT tabanlı görüntü işleme işlemleri	Hibrit Sistem DS1(p-SVM) + DS2 (Vgg-16)	Ortalama doğruluk	Doğruluk (PRO-7): 91.01
[28]	Leishmania paraziti	45 görüntü Çözünürlük: 1500 × 1300	Orijinal	7	Veri artırımı	U-Net	Ortalama (Dice puanı, hassasiyet, duyarlılık ve f1- skoru)	Dice puanı: 80.01, hassasiyet: 79.15, duyarlılık: 75.53, f1-skoru: 75.85
[29]	Bacillus anthracis bakterisi	956 görüntü Çözünürlük: 512 × 512	Orijinal	1	Özellik normalizasyonu, Veri artırımı	UNET++	Parçada bacillus tespiti	Duyarlılık: 98, doğruluk: 97
[30]	Sıtma paraziti	676 görüntü Çözünürlük: 1388 × 1040	Orijinal	2 (sıtma paraziti ya da değil)	Gürültü filtreleme	CNN1 (40 × 40), CNN2 (100 × 100)	Doğruluk, hassasiyet, özgüllük, kesinlik, f1- skoru	Doğruluk (CNN1): 90.10, doğruluk (CNN2): 88.91
[31]	Sıtma paraziti	1182 görüntü Çözünürlük: 750 × 750	Açık kaynak sıtma paraziti veri seti	2 (sıtma paraziti ya da değil)	Uygulanmamış	Modifiye edilmiş YOLOV4, YOLOV3 modified1,	Bütün görüntüde sıtma bakterisi tespiti	YOLOV4 MOD mAP: 96.32, YOLOV3 MOD1 mAP:

						YOLOV3 modified2	95.46, YOLOV3 MOD mAP: 96.14
[32]	Sıtma paraziti	903 görüntü Çözünürlük: 3264 × 2448	Orijinal	2 (sıtma paraziti ya da değil)	Rastgele dikey ve yatay döndürme yoluyla veri artırımı	Faster R-CNN, SSD MobileNet	Bütün görüntüde sıtma bakterisi tespiti Faster R-CNN mAP: 66.09, SSD MobileNet: 62.92

Mikroskopik parazit teşhisi için derin öğrenmenin kullanılması, tıp uzmanları için verimliliği önemli ölçüde artırma potansiyeline sahiptir; bu da yanlış teşhis, gözden kaçan teşhisler ve uygunsuz ilaç kullanımı vakalarının azalmasını sağlar. Protozoa parazitlerin teşhisi için derin öğrenme uygulamasında kaydedilen ilerlemelere rağmen, bazı sorunlar hala devam etmektedir. Önemli sorunlardan bir tanesi açık kaynaklı etiketli veri setlerinin sınırlı sayıda olmasıdır. Bu sorunun üstesinden gelmek için veri setlerini çoğaltmak için bazı yöntemler kullanılmaktadır [33]. Tıbbi çalışmalarda gerçek görüntülerden üretilen yapay veriler yerine tamamıyla gerçek veriler kullanılması daha gerçekçi ve doğru bir sonuca ulaştırmaktadır. Ancak yeterli görüntünün olmadığı durumlarda farklı görüntü çoğaltma yöntemleri kullanılmaktadır. Bu çalışmada diğer yöntemlerden farklı olarak görüntü çoğaltma yöntemlerine ihtiyaç duyulmadan tamamen gerçek veriler kullanılmıştır. Ayrıca Tıp Fakültesi tıbbi biyoloji laboratuvarında dijital kamera ile entegre mikroskop kullanılarak farklı açı ve zoom miktarlarında video ve görüntüler elde edilmiştir.

Mikroskoplar ve diğer görüntüleme cihazları, literatürdeki yöntemlerde protozoa sınıflandırması için yaygın olarak kullanılmaktadır [14,17,34]. Ancak gerçek zamanlı ve tam otomatik tek hücreli protozoa tespit ve sınıflandırıcıları bulunmamaktadır. Bu çalışmada, protozoa bakterilerini tespit etmek ve türlerini belirlemek için derin öğrenme tabanlı YOLO algoritması ile gerçek zamanlı ve tam otomatik bir tanıma sistemi tasarlanmıştır. Önerilen yöntem ile gerçek zamanlı protozoa tespiti kısa sürede ve daha yüksek performansla gerçekleştirilmektedir. Ayrıca geliştirilen bu sistem, paramesyoloji/parazitoloji çalışmalarına ve Tıp Fakültesi Tıbbi Biyoloji Anabilim Dalı'nda laboratuvar uygulaması olan Tek Hücreli Canlılar dersine kaynak sağlayarak eğitime destek aracı olarak da kullanılabilir. Ayrıca öğrencilerin kendi kendilerini eğitmelerine de yardımcı olmaktadır. Ayrıca, bu tam otomatik sınıflandırıcı, sudaki bakterilerin varlığını, türlerini ve çeşitliliğin yoğunluğunu belirleyebilir. Bu çalışma protozoa teşhis ve türünü tespit etme açısından zaman kaybını en aza indirmesi ve tatlı sularda bakteri tespitinin doğruluğunu artırması açısından literatüre önemli katkı sağlayabilir.

Çalışmalara baktığımızda kullanılan görüntülerin yeterli düzeyde olmadığı dikkat çekmektedir. Bazı çalışmalarda veri artırma yöntemlerinin kullanılması nedeniyle çok

fazla veri kullanılmaktadır. Bu görüntüler gerçek görüntülerden üretilen yapay verilerdir ve orijinal görüntüler değildir. Tıbbi arařtırmaların arka planı güçlü ve doğru bir şekilde yürütülmelidir.

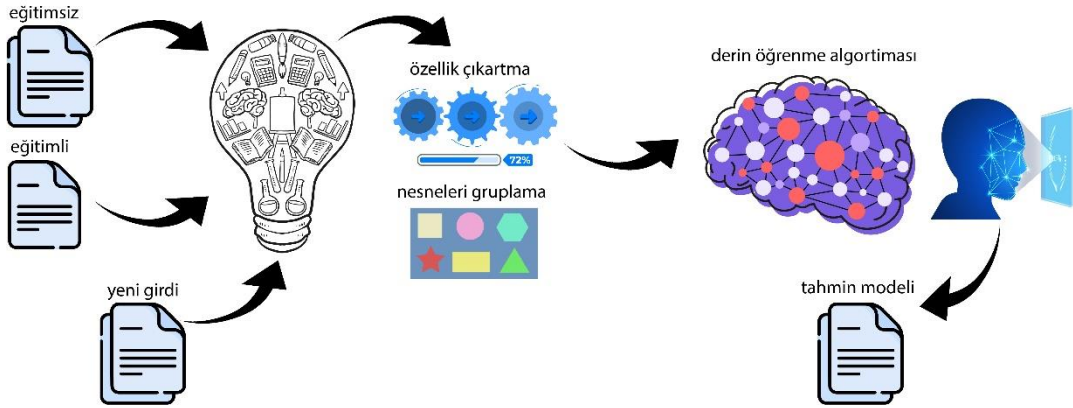
## BÖLÜM 3

### DERİN ÖĞRENME

Yıllar önce, uzmanlar arasında makinelere düşünme yeteneğinin nasıl kazandırılacağını keşfetme vizyonu vardı. Ancak, yavaş hesaplama hızları, sınırlı bellek kapasiteleri, küçük veri kümeleri ve güçlü GPU'ların mevcut olmaması gibi çok sayıda zorluk ilerlemeyi engelledi. Bu engellere rağmen araştırmalar devam etti ve Facebook'ta yüz tanıma, Google'da sanal kişisel asistanlar ve Google Maps gibi navigasyon sistemleri gibi çeşitli alanlarda uygulama alanı bulan modern yapay zekâ ve derin öğrenme teknolojilerinin ortaya çıkmasına yol açtı [35]. Bu ilerlemeler, koşullardaki değişiklikler, algoritmalarındaki gelişmeler, büyük veri kümelerinin kullanılabilirliği ve daha hızlı CPU ve GPU'ların icadı ile desteklendi.

Derin öğrenme, karmaşık algoritmalar ve veri kümeleri ile karakterize edilen modern bir makine öğrenimi dalını temsil eder. Derin öğrenmenin özü, verilerden örüntüleri öğrenmekten sorumlu çoklu katmanlardan oluşan yapay sinir ağlarında (YSA'lar) yatmaktadır. Derin öğrenmedeki "derin" terimi, YSA'larda bulunan çok sayıda katmandan kaynaklanmaktadır. Temel olarak derin öğrenme, depolanan veri kümelerine dayalı olarak insan beyninin karar verme süreçlerini taklit etmeyi amaçlamaktadır. Derin öğrenmenin yapısı Şekil 3.1'de gösterilmektedir.

## DERİN ÖĞRENME



Şekil 3.1 Derin öğrenme mimarisi

YSA'larda, birden fazla yüzeysel ve gizli katman, girdi verilerini işlemek ve çıktı bilgileri üretmek için çalışır. Bu katmanlı yaklaşım, daha küçük problemleri ele alır ve bunları daha büyük problemleri çözmek için bir araya getirir. Böylece veri analizini basitleştirir. Başlangıçta, girdi verileri sistemin hedefleriyle ilgili anlamlı özellikler çıkarmak için bazı dönüşüm işlemlerinden geçer. Örneğin, görüntü işlemede katmanlar bir görüntüyü RGB kanallarına ayırabilir, nesne özelliklerini tanımlayabilir, bunları hedef nesnelere karşılaştırabilir ve benzerlikleri değerlendirebilir. Her katman belirli özellikleri tespit etme konusunda uzmanlaşmıştır. Derin öğrenmenin önemli bir özelliği, birbirini izleyen katmanların anlamlı temsiller öğrenmesine yaptığı vurgudur.

Genel olarak derin öğrenme, geniş veri kümelerinden karmaşık örüntülerin otomatik olarak çıkarılmasını sağlayarak makine öğreniminde devrim yaratır ve böylece bir insan aklının tanımlamasına benzer gelişmiş karar verme süreçlerini kolaylaştırır.

Derin öğrenmede kullanılan modellere Konvolüsyonel Sinir Ağları (CNN), Tekrarlayan Sinir Ağı (RNN), Otomatik Kodlayıcılar, Kısıtlı Boltzmann Makinesi (RBM), Aşırı Öğrenme Makineleri (ELM), Üretken Çekişmeli Ağlar (GAN) örnek olarak verilebilir. Bu çalışmada CNN modeli baz alınarak uygulama geliştirilmiştir.

### 3.1. KONVOLÜSYONEL SİNİR AĞLARI (CNN)

Konvolüsyonel Sinir Ağı (CNN), görüntüler veya zaman serisi verileri gibi yapılandırılmış ızgara verilerini işlemek için özel olarak tasarlanmış bir yapay sinir ağı türüdür. CNN'ler görüntü tanıma, nesne algılama, görüntü segmentasyonu ve daha fazlası dahil olmak üzere çeşitli görevlerde yaygın olarak kullanılmaktadır. CNN'ler, ağın parametrelerini (ağırlıklar ve önyargılar) tahmin edilen ve gerçek çıktılar arasındaki hataya göre ayarladığı geri yayılım adı verilen bir işlemle büyük veri kümeleri kullanılarak eğitilir. CNN'ler bilgisayarla görme alanında devrim yaratmış ve görüntü ile ilgili çeşitli görevlerde, genellikle geleneksel makine öğrenme tekniklerinden daha iyi performans göstererek kayda değer başarılar elde etmiştir.

CNN, verileri işlemek için konvolüsyonel katmanlar, havuzlama katmanları ve tam bağlantılı katmanlardan oluşur. İleri aşama, girilen verileri ağırlıklar ve kayıp fonksiyonu ile işlemekte, ardından tahminlere bağlı olarak çıktıyı tahmin etmektedir. Geri aşama ise kayıp fonksiyonu ve gradyanlardan bilgi alarak katmanları iyileştirmek için güncellemektedir. Bu işlem yeterli tekrar süresi elde edilene kadar döngü halinde devam etmektedir. Ana aşamalar şu şekilde açıklanmaktadır:

**Konvolüsyonel katmanlar:** Bu katmanlar, öğrenilebilir filtreler (çekirdek olarak da bilinir) kullanarak giriş verilerine konvolüsyon işlemleri uygular. Konvolüsyon işlemi, özellikleri çıkarmak için bu filtrelerin giriş verileri üzerinde kaydırılmasını içerir. Evrişimsel katmanlar, girdi verilerindeki örüntülerin uzamsal hiyerarşilerini yakalamada oldukça etkilidir.

**Havuzlama katmanları:** Havuzlama katmanları, alt örnekleme yoluyla girdi verilerinin uzamsal boyutlarını azaltır. Maksimum havuzlama ve ortalama havuzlama, ağın hesaplama karmaşıklığını azaltırken en önemli özellikleri korumak için kullanılan yaygın havuzlama işlemleridir.

**Tam bağlantılı katmanlar:** Bu katmanlar, her nöronun önceki ve sonraki katmanlardaki her nörona bağlı olduğu geleneksel sinir ağı katmanlarıdır. Tam bağlı katmanlar tipik

olarak, evrişimli katmanlar tarafından çıkarılan özelliklere dayalı olarak sınıflandırma veya regresyon görevlerini gerçekleştirmek için ağıın sonuna doğru kullanılır.

### **3.2. NESNE TANIMA**

Nesne algılama, dijital görüntü veya videolardaki nesnelere tanımlamayı ve konumlandırmayı içeren bilgisayarla görmede temel bir görevdir. Otonom araçlar, gözetim sistemleri, tıbbi görüntüleme ve endüstriyel otomasyon dahil olmak üzere çeşitli uygulamalarda kritik bir bileşendir.

Nesne algılama algoritmaları, bir görüntü veya video karesi içindeki ilgi çekici nesnelere tespit edip konumlandırmayı ve aynı zamanda bunları önceden tanımlanmış kategorilere sınıflandırmayı amaçlar. Bir görüntünün tamamındaki baskın nesneyi tanımlayan görüntü sınıflandırmasının aksine, nesne algılama algoritmaları tek tek nesnelere etrafında kesin sınırlayıcı kutular çizer ve algılanan her nesneye etiket atar.

Derin öğrenme ile görüntü üzerinde nesne tanıma konusunda literatürde öne çıkan algoritmalar R-CNN, Fast R-CNN ve Faster R-CNN gibi iki aşamalı algoritmalar ve SSD, YOLO gibi tek aşamalı algoritmalar. Bu algoritmalar, araştırmacılar tarafından çalışmalarında ideal sonuçları elde etmek için farklı yöntemlerde kullanılmıştır. Ayrıca, bu bölümde bu algoritmaların temel özellikleri ve kullanımının mevcut çalışmalarda nasıl öne çıktığı gösterilecektir.

### **3.3. TEK AŞAMALI TESPİT ALGORİTMALARI**

Tek aşamalı dedektörler, sınırlayıcı kutuları ve nesne sınıflandırmasını doğrudan sağlayan tek bir ileri beslemeli tam evrişimli ağı içerir. Single Shot MultiBox Detector (SSD) ve YOLO (You Only Look Once) tek bir birleşik mimari öneren ilk sistemler arasında yer alır. Tek aşamalı bir nesne dedektöründe, doğrudan görüntüden sınıflandırmaya ve sınırlayıcı kutu koordinatlarına gideriz. Görüntüler bir CNN kullanılarak bir özellik çıkarıcıya beslenir ve daha sonra çıkarılan özellikler doğrudan sınıflandırma ve sınırlayıcı kutu koordinatları için regresyon için kullanılır. Tek aşamalı nesne dedektörleri çok hızlıdır ve gerçek zamanlı nesne tespitinde



kullanılabilir, ancak bazen performansları iki aşamalı nesne dedektörlerinden daha düşüktür. Örnekler YOLO ailesi, SSD, RetinaNet, vb.

YOLO dedektörleri, görüntüyü bölgelere ayıran ve her bölge için sınırlayıcı kutuları ve olasılıkları tahmin eden bir başka çapa tabanlı alternatiftir. YOLO ağlarının, RetinaNet kadar doğru olmasa da RetinaNet'ten daha hızlı çıkarım oranlarına ulaştığı kanıtlanmıştır. Ancak YOLOv3, çok ölçekli tahminler ve artık bloklar ve atlama bağlantıları kullanan daha iyi bir omurga ağı olan DarkNet-53'ü dahil ederek tespit doğruluğunu artırmayı başardı [36]. DarkNet-53, önceki DarkNet-19'dan çok daha güçlüdür ve ResNet omurgalarından hala daha verimlidir.

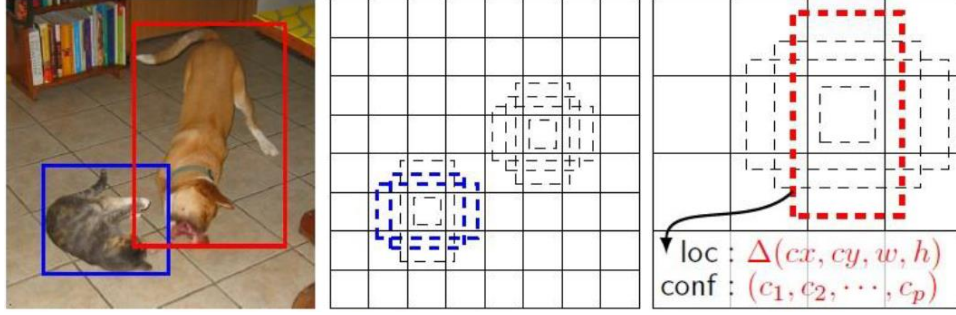
### **3.3.1 SSD**

Bu algoritma Wei Liu, Dragomir Anguelov tarafından 2016 yılında yayınlanmıştır [37]. SSD, farklı boyutlardaki nesnelere kontrol etmek için birçok çözünürlüğe sahip iki kat özellik haritasından tahmini eğitebileceğimiz ve karıştırabileceğimiz kolay bir yöntemdir. Ağ, her kutudaki nesnelere için puanlar üretir; SSD tüm hesaplamayı tek bir ağda birleştirir. SSD'nin doğruluğu pascal VOC üzerinde 300x300 girdi için 59 FPS'de %74,3 mAP'dir ve 512x512 için neredeyse %80 mAP'ye ulaşır. Faster R-CNN'den [38] daha iyi performans gösterir ve doğruluğu artırmak için bir tür veri artırımı kullanır. Faster R-CNN, RPN kullanır ve algoritmayı geliştirmek için paylaşılan evrişimli katmanlara sahipken, SSD, RPN ile Faster R-CNN'nin bir karışımıdır ve daha az karmaşıklığa sahip, eğitilmesi kolay, daha doğru ve daha hızlı bir ağ üretebilir. Dolayısıyla, SSD'nin hıza odaklanan Faster R-CNN'nin başka bir şekli olduğunu söyleyebilir. SSD, nesne sınıflarını ve sınırlayıcı kutular için 4 koordinatı üretmek için küçük Conv filtreleri kullanır ve birden fazla boyutta tespit etmek için birden fazla özellik haritasına Conv filtreleri uygular.

#### **3.3.1.1. SSD Framework**

Herhangi bir nesne tespit algoritması için yapılması gereken ilk şey, bir dizi görüntü içindeki nesnelere etiketlemektir. Bu etiketlere kesin referans kutuları denir. Eğitim, varsayılan kutuyu uygulayacak ve bunu kesin referans kutularıyla karşılaştıracaktır.

Daha sonra bu varsayılan kutulardan hangilerininin 0,5 veya %50'den fazla örtüştüğünü belirleyecek ve bunları pozitif olarak tanımlayacaktır. Geri kalanlar negatif olarak değerlendirilecek ve bundan sonra pozitifler filtrelenerek en doğru olanlar seçilecektir. Eğer pozitif örtüşme varsa (0.6, 0.7, 0.95), sadece 0.95'i seçilmeli ve diğerlerini bir nesne için elimine edilmesi gerekir.



(a) Kesin referanslı görsel (b) 8x8 özellik haritası (c) 4x4 özellik haritası

Şekil 3.2 SSD'nin birden fazla nesneyi algılamak için çalıştığı şekli

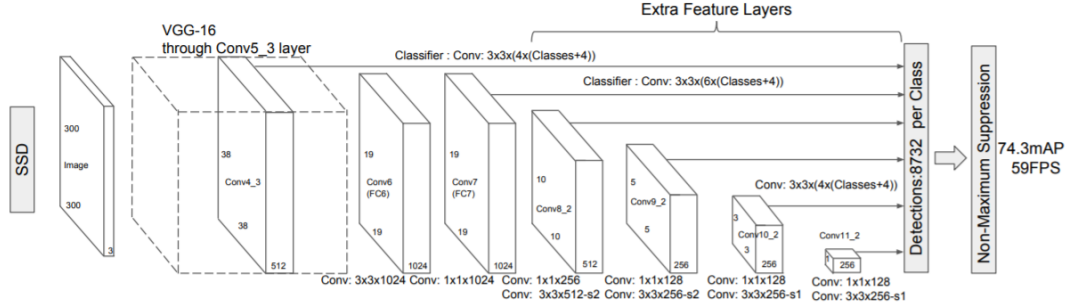
Şekilde, farklı boyutlardaki varsayılan kutulardan oluşan küçük bir set, çeşitli ölçeklerde çoklu özellik haritalarıyla görüntünün farklı konumlarına uygulanmıştır ((b)'de 8x8 ve (c)'de 4x4). Her kutu için, köpek ve kedi nesne sınıfları için hem şekil ofsetlerini hem de güven puanlarını üretirler.

### 3.3.1.2. SSD Model

SSD model aşağıdaki kısımlardan oluşur:

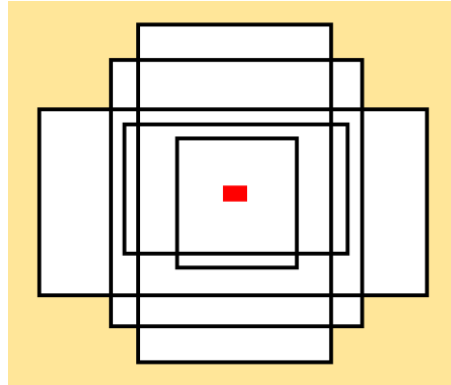
- Tabanlı ağ: sınırlayıcı kutuların içinde bulunan mevcut nesne sınıfı için sabit boyutlu sınırlayıcı kutular ve puanlar çıkararak ileri beslemeli evrişimli ağıdır. Orijinal makaleye göre [37] temel ağ olarak VGG-16 kullanılmaktadır.
- Tespit için çok ölçekli özellik haritaları: Ağın son katmanında konvolüsyonel özellik katmanları içerir ve tespit edilen nesnelere için farklı ölçeklerde tahmin yapılmasını sağlar.
- Tespit için konvolüsyonel tahmin ediciler: Eklenen özellik katmanları, sabit bir tespit  $m \times n$  boyutu ve kanallar üretmek için konvolüsyonel filtreleri kullanır.

Sınıfları, puanları ve her konumdaki sınırlayıcı kutuların 4 koordinatını üreten küçük bir çekirdek  $3 \times 3 \times P$  (kanallar) kullanır.



Şekil 3.3 SSD mimarisi [37]

Varsayılan kutular ve en boy oranı: Her hücre için bir dizi varsayılan kutu vardır. Hücredeki her bir varsayılan kutuda, önerilen nesne için 4 koordinat ve puan tahmin edilir.  $(c + 4) \times k$  her konuma uygulanır,  $c$  sınıfların sayısı ve  $k$  çekirdek içindir. Bu varsayılan kutu, Faster R-CNN'deki [38] çapa kutusuna benzer ve Şekil 3.4'te görebiliriz.



Şekil 3.4 Konvolüsyon noktası SSD'de ayarlanmış farklı boyutlarda çapa kutuları

SSD tüm noktalarda beş varsayılan kutu kullanır. Bu kutular bir dedektör gibi çalışmaktadır ve her biri beş farklı nesneyi tespit etme kapasitesine sahiptir. Bu kutular farklı boyutlardadır, bu nedenle Şekil 3'te gösterildiği gibi farklı boyutlardaki nesnelere algılayabilecektir. Bu kutular farklı boyutlara sahiptir; daha büyük olanlar büyük nesnelere tespit edebilirken, daha küçük olanlar daha küçük nesnelere tespit edebilecektir. Sonunda, bu nesnelere için sınırlayıcı kutunun yanı sıra nesne puanını da tahmin etmeye çalışacaktır. Eğer bulunacak bir nesne yoksa yok sayar.

### 3.4. İKİ AŞAMALI TESPİT ALGORİTMALARI

İki aşamalı nesne tespit algoritmaları, bir görüntüdeki nesnelere tespit etmek için iki temel aşamadan oluşan bir yaklaşımı ifade eder. Bu tür algoritmalar genellikle bölge önerisi (region proposal) ve nesne sınıflandırma aşamalarından oluşur.

**Bölge Önerisi (Region Proposal):** İlk aşamada, görüntü üzerinde potansiyel nesne bölgelerini belirlemek için bir bölge önerisi algoritması kullanılır. Bu aşamada, genellikle seçici arama (selective search), EdgeBoxes veya benzeri yöntemler kullanılarak görüntüde muhtemel nesne konumları tahmin edilir ve bu bölgeler öneri olarak belirlenir. Bu bölgeler, olası nesnelere bulunabileceği bölgeleri kapsar ve daha sonra nesne sınıflandırma aşamasına iletilir.

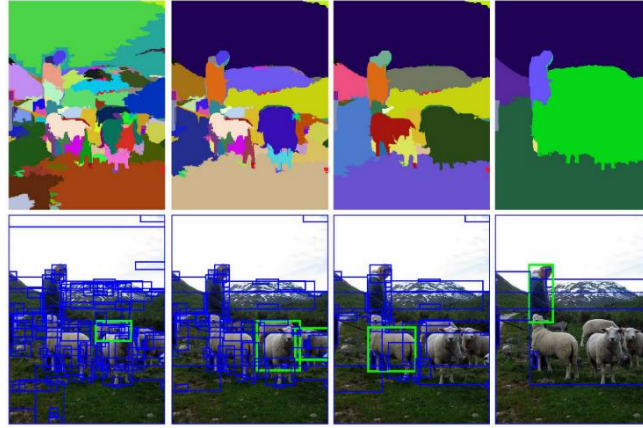
**Nesne Sınıflandırma:** İkinci aşamada, bölge önerileri üzerinde nesne türünü belirlemek için bir sınıflandırıcı kullanılır. Bu sınıflandırıcı genellikle bir derin öğrenme modeli olabilir, örneğin bir konvolüsyonel sinir ağı (CNN). Bölge önerileri, bu sınıflandırıcıya girdi olarak verilir ve her bölge için olası nesne sınıflarının olasılıkları tahmin edilir. Ardından, bu olasılıklar belirli bir eşik değerinin üzerinde olan bölgeler nesne olarak kabul edilir ve ilgili sınıflarla etiketlenir.

İki aşamalı nesne tespit algoritmaları, daha karmaşık ve hassas sonuçlar elde etmek için genellikle daha gelişmiş derin öğrenme teknikleriyle birleştirilir. Özellikle, ikinci aşamada kullanılan sınıflandırıcı, daha iyi performans için derin öğrenme tekniklerinden faydalanabilir. Bu tür algoritmalar, bilgisayarlı görü sistemlerinde ve nesne tespiti uygulamalarında yaygın olarak kullanılmaktadır. Kullanılan başlıca modeller R-CNN, Fast R-CNN, Faster R-CNN, Mask R-CNN'dir.

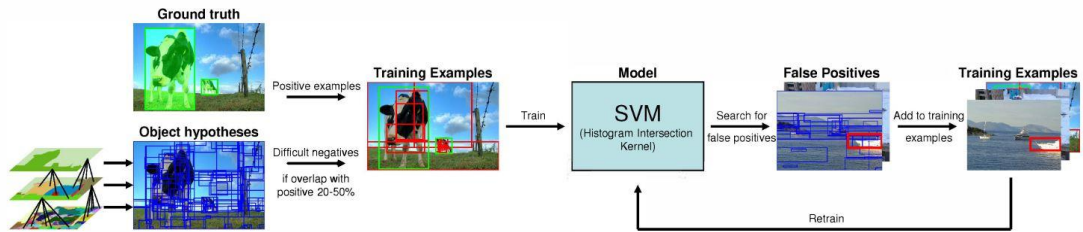
#### 3.4.1. R-CNN

Görüntü üzerinde nesne algılama işlemleri yapan ilk konsept olan R-CNN Ross Girshick tarafından geliştirilmiştir ve 2014 yılında bir makalede yayınlanmıştır [39]. Bölge önerilerini CNN ağı ile birleştirdiği için R-CNN ismi verilmiştir. Üç modelden

oluşur; birincisi bölge önerileri üretir, ikincisi bu bölgeleri girdi olarak kullanır ve her bölge için bir vektör olarak sabit uzunlukta özellikler çıkarmak için büyük bir CNN ağı uygular ve son model Destek Vektör Makinesi'dir (SVM). R-CNN için tespit sistemi üç modelden oluşmaktadır: İlk olarak, bölge önerisi, R-CNN algoritması görüntüde herhangi bir nesneyi tanımlamak için bazı bölgeleri tespit etmeye çalışır. Önerilen bu alanlar nesnenin etrafındaki sınırlayıcı kutulardır. Nesneyi tanımak için bir seçici arama algoritması kullanılır ve sonunda bölge önerisi oluşturulur. Seçici arama algoritması ilgili makaleye [40] göre, nesnelerin konumu belirlendikten sonra her nesneye ayırt edici bir renk verileceği görüntü segmentasyonu adı verilen bir işlem uygulanır ve bu duruma ait örnek aşağıdaki Şekil 3.5'te gösterilmiştir.



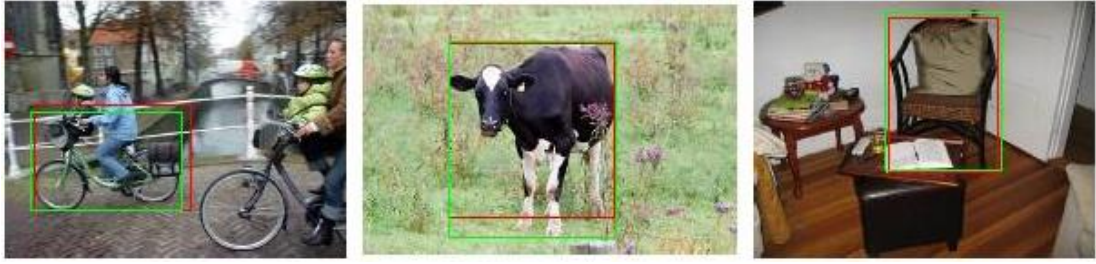
Şekil 3.5 Görüntü Segmentasyonu [40]



Şekil 3.6 Seçici arama metodu için nesne tanıma eğitim süreci [40]

Şekil 3.6 eğitim sürecini temsil etmektedir; Shogun Araç Kutusu [40] kullanılarak histogram kesişim çekirdeğine sahip bir sınıflandırıcı olarak Destek Vektör Makinesi (DVM) kullanılmaktadır. İlk pozitif örnekler tüm sınırlayıcı kutuları ve nesne pencerelerini içeriyordu. İlk negatif örnekler için, seçici arama tarafından üretilen ve %20 ila %50 örtüşme oranına sahip tüm nesne alanları seçilmiştir [40]. Seçici

aramanın testi için: son model, seçici arama tarafından üretilen tüm konumlarda kullanıldı, pencereler sınıflandırıcı puanına göre sıralandı, eğer pencereler yüksek puanlı bir pencere ile %30'dan fazla örtüşüyorsa eş kopyası olması nedeniyle kaldırılır. Şekil 3.7'de seçici arama algoritması kullanılarak inek, sandalye ve bisiklet nesnesi için tespit sonuçları gözükmemektedir. En yüksek puanlı tespitler gösterilmektedir; bisiklet: 0.86, inek: 0.87 ve sandalye: 0.88.



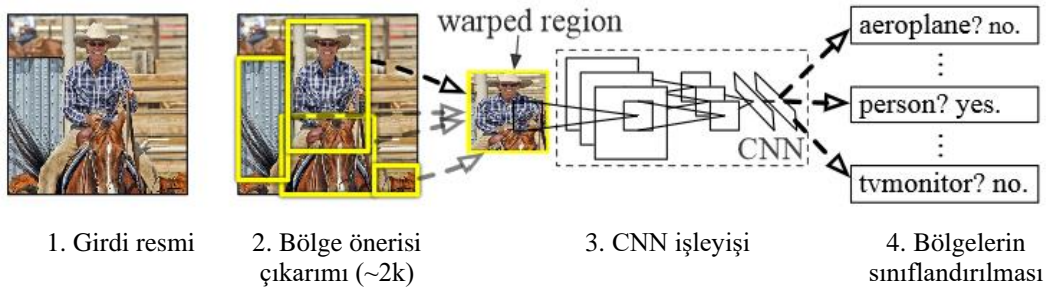
(a) Bisiklet: 0.863

(b) İnek: 0.876

(c) Sandalye: 0.884

Şekil 3.7 Seçici arama yöntemi ile elde edilen en yüksek skorlarla nesne tespiti

Bölge önerisi artık kullanıma hazır olduğu için bir sonraki adım özellik çıkarma olacaktır. Bu bölgeler görüntüleri kesecek ve 5 Conv katmanı ve 2 FC katmanı aracılığıyla (227x227) boyutuna yeniden boyutlandırılır, ardından her bölge için 4096 boyutlu özellik vektörünün çıkarılması için CNN'e aktarılır. CNN mimarisi sabit boyutta (227) girdi gerektirmesinden dolayı yeniden boyutlandırma işlemi gerekmektedir. Son olarak, sınıflandırıcı, doğrusal DVM modeli gibi bir sınıflandırma algoritması kullanarak çıkarılan özellikleri sınıflandırmak için yapılan bir sistemdir.



Şekil 3.8 R-CNN algoritmasının çalışma şekli

Şekil 3.8, R-CNN algoritmasının nasıl çalıştığını göstermektedir. İlk olarak, girdi görüntüleri işleme alınır ve ardından CNN kullanarak her bölge için özellikleri

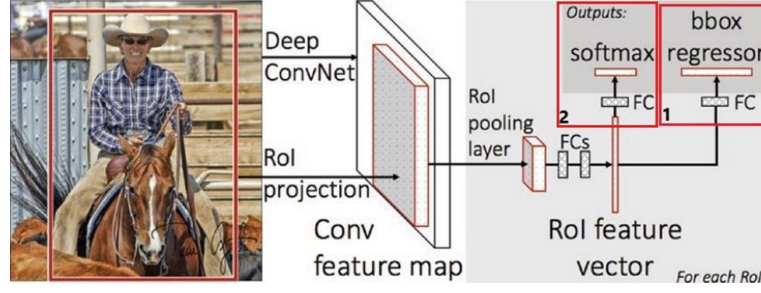
hesaplamak üzere 2000 bölge çıkarır. Daha sonra, bu bölgeleri DVM kullanarak sınıflandırır. R-CNN'nin nesne algılama ile çalışan ilk algoritma olduğu düşünüldüğünde, işlem süresindeki gecikme açısından birkaç sorunu rağmen iyi bir doğruluğa ulaşılmıştı. Aşağıda, R-CNN'nin gecikmesine neden olan, her biri 2000 konvolüsyonel sinir ağı kullanan yaklaşık 2000 bölge önerisindeki her bir görüntü ekstraktından R-CNN için özelliklerin açıklaması yer almaktadır.

Üç farklı modeli eğitmek gerekir, özellikleri oluşturmak için CNN modeli, verileri sınıflandırmak için morphem modeli (sınıflandırıcı) ve tüm bunları bir araya getirmek için üçüncü model (regresyon modeli). CNN modelinden gelen tahmin yavaş olacaktır çünkü her bölge önerisinin tahmin edilmesi gerekmektedir.

### **3.4.2. Fast R-CNN**

Fast R-CNN, yine Ross Grishick tarafından geliştirilen ve 2015 yılında bir makalede yayınlanan R-CNN algoritmasının geliştirilmiş versiyonudur [41]. Bu model önceki versiyondan daha hızlı ve SPP-Net'ten daha hızlı çalışır [42]. Sınıflandırma için bu modele daha fazla doğruluk sağlayan VGG-16 kullanır. Fast R-CNN tüm verileri (görüntü) bir kerede işlerken, R-CNN görüntüyü işlemeden önce birçok bölgeye ayırır. Fast R-CNN'nin adımları aşağıda listelenmiştir:

- Tüm görüntüyü girdi olarak alır ve bölgeyi oluşturmaya başlar. Bir konvolüsyon özellik haritası üretmek için tüm görüntüyü birkaç konvolüsyonel ve maksimum havuzlama katmanıyla işler.
- CNN'in (ROI) ilgi bölgesi olarak bilinen son katmanı bir havuzlama katmanıdır. Bu katman, özellik haritasından sabit bir vektör çıkarır.
- Tam bağlı katmana (FC) beslenen her özellik, biri nesne sınıfları için ürün SoftMax ve ikincisi sınırlayıcı kutu için ürün regresyonu olan iki çıktı katmanına ayrılır.



Şekil 3.9 The Fast R-CNN mimarisi

Şekil 3.9, son katman ROI havuzlama katmanını FC katmanına beslenmiştir ve çıktı, biri SoftMax üreten ve diğeri regresyon üreten iki FC katmanına bölünmüştür.

Fast R-CNN'nin avantajları:

- R-CNN'den (mAP) daha yüksek (mAP) elde edebilir.
- Eğitim tek aşamalıdır ve tüm ağ katmanlarını güncelleyebilir.
- Özellik yakalamayı depolamak için disk depolama gerekli değildir.

### 3.4.3. Faster R-CNN

Faster R-CNN algoritması, 2016 yılında Shaoqing Ren, Kaiming He, Ross Girshick ve Jian Sun tarafından geliştirilen Fast R-CNN'nin geliştirilmiş versiyonudur [38]. Paylaşılan konvolüsyonel Bölge Öneri Ağı (RPN) kullanarak ve Fast R-CNN ile RPN'yi tek bir ağda birleştirerek eğitim hızını ve tespit doğruluğunu geliştirdiler. Algılama sistemi olarak VGG-16 kullanır, ancak bölge önerme modeli Fast R-CNN'deki ile aynı kalır ve bu versiyondaki algılama oranı 5 FPS'ye kadar ulaşabilir.

Model, bölge öneren tam evrişimli ağıdır. Bu model, görüntünün her bölge önerisinde sınırlayıcı kutuları ve nesne puanlarını tahmin etmek için çalışan tam evrişimli bir ağ olan bölge öneri ağını (RPN) içerir. RPN, gereksiz işlemleri en aza indirmek ve dolayısıyla işlem süresini azaltmak için kullanılır. Bölge önerileri oluşturmak için, konvolüsyonel özellik haritası verimi üzerine küçük bir ağ kaydırılır ve son konvolüsyonel katman tarafından üretilen konvolüsyonel özellikler haritasını ve kaydırılan küçük ağı alır. Bu küçük ağ  $N \times N$  girdisini alır ve girilen konvolüsyonel

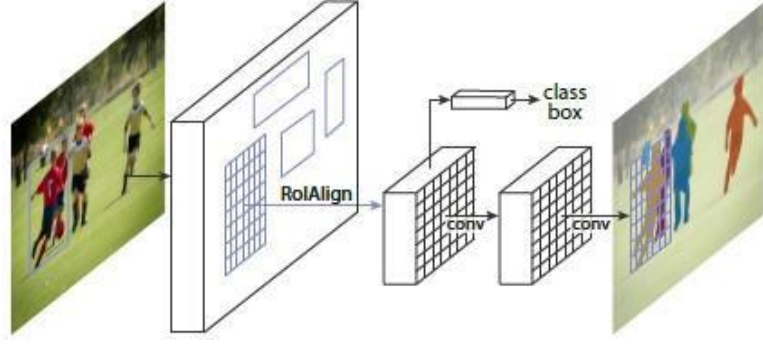


özellik haritasının bir penceresi olarak hareket eder. Tüm pencereler bir alt özellik ile bağlantılıdır. Bu özellik, Fast R-CNN regresyonu ve SoftMax'te aynı iki katman olan iki FC katmanı (regresyon ve sınıflandırma) katmanı için bir girdidir.

Buradaki yeni geliştirme, bölge önerilerini tekrar üretmek yerine test zamanında evrimsel katmanları paylaşan RPN'nin tanıtılması olacaktır. Bu, yeni bir bölge üretmek için çıktı bölgesini kullanmasına ve zamandan tasarruf etmesine yardımcı olur. Ayrıca, farklı ölçeklere sahip çapa kutularının kullanılması, algoritmaya önceki versiyonda yeterince iyi olmayan geniş bir aralığa sahip nesnelere tespit etme yeteneği kazandırmaktadır [43]. Bu algoritmada uygulanan geliştirmelerin, nesne algılama doğruluğunu artırmak için uygun maliyetli ve etkili bir çözüm olduğu kanıtlanmıştır [38].

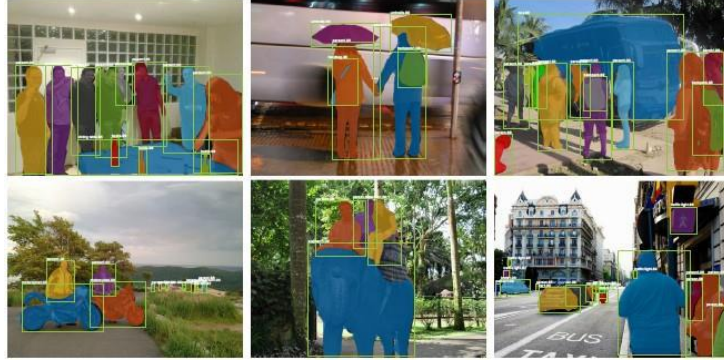
#### **3.4.4. Mask R-CNN**

Mask R-CNN, Faster R-CNN algoritması için bir genişletmedir ve her bir İlgi Bölgesi (RoI) üzerinde tespit edilen nesnenin maskelerini tahmin etmek için iki konvolüsyon katmanı eklenmiştir [44]. Mask R-CNN'nin üç çıktı katmanı, nesne katmanı için bir sınıf adı, sınırlayıcı kutular katmanı için 4 koordinat ve nesne maskesini oluşturmak için ek konvolüsyon katmanları vardır. Mask R-CNN'yi eğitmek ve Faster R-CNN için ekstra katmanlar eklemek kolaydır ve 5 FPS'de çalışabilir. Ancak Mask R-CNN, nesne algılama işlemini çalıştırmak için iyi bir doğruluk değeri gerektirir ve ayrıca algılanan nesnelere için de iyi bir doğruluk elde edilmesi gerekir. Bu algoritma, en doğru algılamayı elde etmek için omurga olarak Res-Net [45] ve FPN Özellik Piramidi Ağı [42] kullanmaktadır. Mask R-CNN, Kaiming He, Georgia Gkioxari, Piotr Doll'ar ve Ross Girshick'ten oluşan (FAIR) FacebookAI Araştırma ekibi tarafından geliştirilmiştir. 24 Ocak 2018 tarihinde kullanıma sunulmuştur.



Şekil 3.10 Mask R-CNN algoritmasının çalışma şekli [14]

Şekil 3.10'da maske modeli, her bir RoI'ye uygulanan ve pikselden piksele bir segmentasyon maskesi tahmin eden küçük bir FCN'dir. Esnek bir mimari tasarım olan Faster R-CNN çerçevesi göz önüne alındığında Mask R-CNN'nin uygulanması ve eğitilmesi çok kolaydır. Maske bölümü küçük bir hesaplama ek yükü getirir. Şekil 3.11'de görebileceğiniz gibi, Mask R-CNN'nin sonucu ve nasıl çalıştığı.



Şekil 3.11 Mask R-CNN algoritmasının sonucu [44]

Omurga, Res-Net [45] 50 veya 101 katmandan oluşmaktadır. Ayrıca özellik çıkarımı için Özellik Piramit Ağı (FPN) [46] adı verilen başka bir omurga kullanır. Özellik çıkarma için omurga olarak Res-Net-FPN kullanmak, Mask R-CNN'ye doğruluk ve hız açısından ek güç sağlar. Ağ kafasına gelince, Faster R-CNN'de [38] sunulan çalışmayı kullanılmıştır ve iki tam konvolüsyonel maske tahmin modeli eklediler.

### 3.5. ÇALIŞMADA KULLANILAN YOLO ALGORİTMASI

YOLO (You Only Look Once), bir görüntüdeki birden fazla nesneyi gerçek zamanlı olarak algılayabilen popüler bir nesne algılama algoritmasıdır. Joseph Redmon, Santosh Divvala, Ross Girshick ve Ali Farhadi tarafından 2016 yılında tanıtılmıştır. YOLO'nun arkasındaki temel fikir, her nesne için sınırlayıcı kutuları ve sınıf olasılıklarını ayrı ayrı tahmin etmeyi içeren geleneksel yaklaşım yerine, nesne algılamayı tek bir regresyon problemi olarak çerçevelemektir. Bu, YOLO'nun gerçek zamanlı işlem hızlarını korurken yüksek algılama doğruluğuna ulaşmasını sağlar.

YOLO, nesne tespiti için geliştirilmiş bir derin öğrenme algoritmasıdır. Diğer geleneksel nesne tespiti yöntemlerinden farklı olarak, YOLO görüntüyü bir kez işleyerek nesne tespiti ve sınıflandırma yapar. Bu, daha hızlı çalışma süresi ve gerçek zamanlı uygulamalarda etkin performans sağlar. YOLO temel olarak omurga, boyun ve baş kısımlarından oluşan bir yapıya sahiptir.

YOLO algoritması RGB formatında bir görüntüyü girdi olarak alır. Görüntüler, genellikle 416x416 piksel boyutlarına yeniden boyutlandırılır ve bu girdi görüntüleri 0 ile 1 arasında normalleştirilir. YOLO, konvolüsyonel sinir ağı (CNN) tabanlı bir derin ağ kullanır. Bu ağ, önceden eğitilmiş ağırlıklar üzerine inşa edilebilir veya sıfırdan eğitilebilir. Evrişim katmanları, görüntüdeki özellikleri çıkarmak için kullanılır. Özellikler, her bir katmanın filtreleri tarafından vurgulanır. CNN tarafından işlenen görüntü, farklı ölçeklerdeki özellik haritalarına dönüştürülür. Her bir özellik haritası, görüntünün farklı özelliklerini temsil eder. Özellik haritaları üzerinde nesne tespiti gerçekleştirilir. Her bir hücre, belirli bir bölgedeki nesnelerin tespit edileceği bir bölgeyi temsil eder. Her hücre, birden fazla sınıflandırıcı kutu (bounding box) ve bu kutuların içerdikleri nesnelerin olasılıklarını tahmin eder. Her bir sınıflandırıcı kutu, bir nesneyi temsil eder ve nesnenin sınıfını ve konumunu belirtir. Nesne tespiti sırasında, aynı nesnenin birden fazla kez tespit edilmesini önlemek için Non-Maximum Suppression (NMS) kullanılır. Bu, örtüşen sınıflandırıcı kutular arasında belirli bir eşik değerinden daha düşük olanları filtreler. Son katmanda YOLO, nesnelerin sınırlayıcı kutularının koordinatlarını, sınıflarını ve güven skorlarını içeren

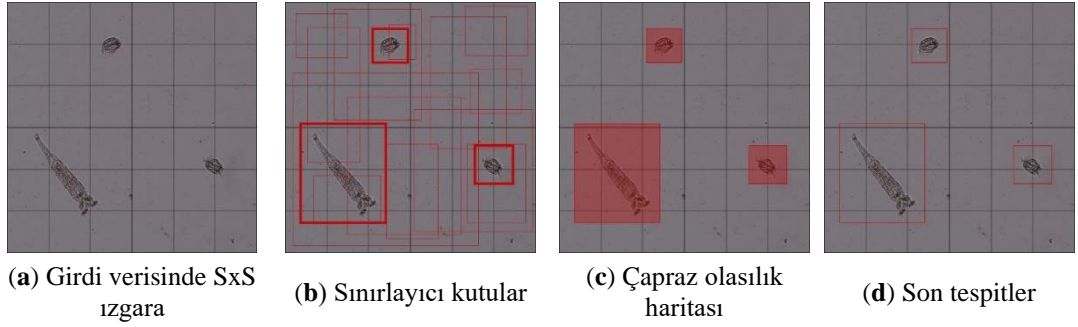
bir çıktı üretir. Bu çıktı, tespit edilen nesnelerin türlerini ve konumlarını belirlemek için kullanılabilir.

### **3.5.1. YOLOv4**

YOLO (You Only Look Once) olarak bilinen derin bir sinir ağı, protozoanın mikroskopik görüntüleri içindeki nesnelere tanımlamak ve sınıflandırmak amacıyla kullanılmıştır. 90,82'lik olumlu bir duyarlılık gösteren [47] bu ağı, dermoskopik görüntülerde bulunan melanom, iyi huylu nevüsler ve seboreik keratozlar gibi cilt lezyonlarını tanımak ve segmentlere ayırmak için de kullanılmıştır. Bu çok yönlülüğü nedeniyle YOLO, mikroskopik görüntülerde bulunabilecek çeşitli boyut, hareketlilik ve türleri kapsayan varlıkları tespit etmek için uygun bir yöntemdir. Protozoa tarafından sergilenen çeşitli hareket yelpazesine ek olarak, farklı türler arasında şekil ve fiziksel özelliklerde bir çeşitlilik vardır. Sistem, giriş görüntülerine dayalı olarak görüntü sınıflandırması gerçekleştirir ve nesne algılama işlemi, nesnenin konumunun onu çevreleyen bir sınırlayıcı kutuya göre belirlenmesini içerir [48].

#### **3.5.1.1. Dedektör**

Tek aşamalı nesne algılama algoritması olan YOLO algılama aşamasında yoğun tahmin gerçekleşir. Son tahmin adımı üç veri grubu içeren bir vektörden oluşur. İlk olarak, sınırlayıcı kutu, merkez, yükseklik ve genişlik içeren koordinatları, tahminin güven puanı ve etiketi ile tahmin edilir. Şekil 1, nesnelerin konumunu belirleyen YOLO'nun gösterimini göstermektedir. Resmi Şekil 3.12(a)'da gösterilen (S S) ızgaralara böler ve ardından Şekil 3.12(b)'de B potansiyel sınırlayıcı kutuları oluşturmak için her bir ızgara hücrelerine bir güven puanı ve sınıf olasılıkları atar.



Şekil 3.12 YOLO algoritmasının şematik gösterimi.

Güven skoru, ızgarada bir nesne bulma olasılığı ile birleşim üzerinde kesişim değerinin çarpımı ile hesaplanır. Nesnenin bulunduğu kutu ile tahmin edilen kutunun kesişimi olan birleşme üzerinde kesişim (IOU) yüzdesi ile çarpılmasıyla hesaplanır ve Şekil 3.13'teki formül kullanılır:

$$\text{Güven Skoru} = \text{Pr}(\text{nesne}) * \text{IoU} \quad (1)$$

$$\text{IOU} = \frac{\text{Kesişim Alanı}}{\text{Birleşim Alanı}}$$

Şekil 3.13 Birleşim üzerinde kesişim (IOU), nesne algılama performansı ölçümü.

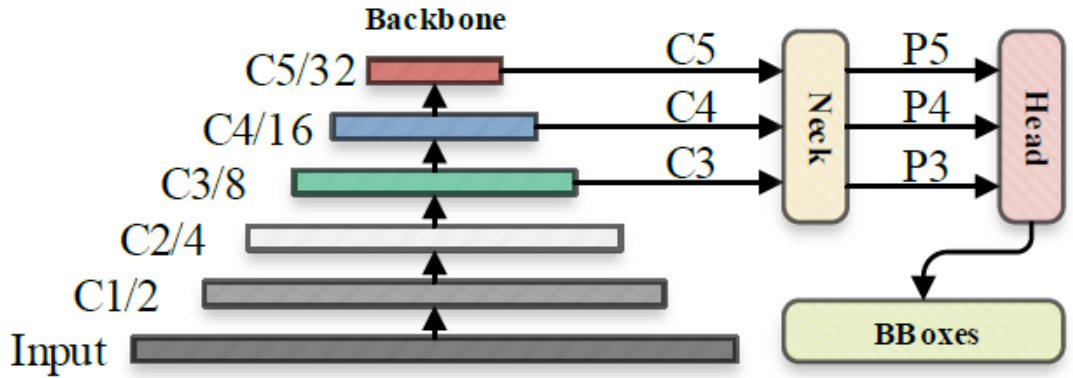
Pr (nesne) olarak gösterilen nesnenin var olma olasılığı 0,7 olduğunda kutunun bir nesne içermesi olasılığı %70'tir. Güven puanının sıfır olması, o hücrede bir nesnenin bulunmadığını gösterir. Güven puanı, belirli bir eşikte Genel Ortalama Kesinlik (mAP) hesaplanmasında kullanılır. Eğer bir ızgara hücresi %60 oranında bir araba içermesi olasılığı öngörüyorsa ( $\text{pr}(\text{Araba}) = \%60$ ), hücrenin gerçekten bir araba içermesi olasılığı %60, içermeme olasılığı ise %40'tır [31]. Görüntü üzerinde gezdirilen sınırlayıcı kutunun toplam beş parametresi vardır: [x, y, w, h, güven puanı]. Burada (x, y) sınırlayıcı kutunun orta noktasının koordinatlarını, (w, h) genişlik ve yüksekliği temsil etmektedir.

v4'ün öne çıkan özelliği, üç farklı ölçekte tespit yapabilmesidir. YOLO tamamen konvolüsyonel bir ağ olarak çalışır ve nihai çıktısı bir özellik haritası üzerinde  $1 \times 1$  çekirdeğin kullanılmasıyla elde edilir. YOLOv4'te tespit süreci, ağ içinde üç farklı konumda stratejik olarak konumlandırılmış çeşitli boyutlardaki özellik haritaları üzerinde  $1 \times 1$  tespit çekirdeklerinin uygulanmasını içerir [49].

Algılama çekirdeğinin konfigürasyonu  $1 \times 1 \times (B \times (5 + C))$  şeklindedir. Bu bağlamda B, özellik haritası içindeki bir hücrenin tahmin edebileceği sınırlayıcı kutuların miktarını temsil etmektedir. "5" değeri, sınırlayıcı kutularla ilgili dört öznitelik ve ek bir nesne güven puanı ile ilgilidir; C ise toplam farklı sınıf sayısını ifade eder. COCO veri kümesi üzerinde eğitilen YOLOv4 modelleri için B 3'e karşılık gelir ve C 80 olarak ayarlanır, bu da  $1 \times 1 \times 255$ 'lik bir çekirdek boyutuyla sonuçlanır [50].

### 3.5.2. YOLOv5

Görüntü bir giriş katmanı (giriş) aracılığıyla işlenmiş ve özellik çıkarımı için omurgaya (backbone) gönderilir. Şekil 3.14'te yer alan omurga, farklı boyutlarda özellik haritaları elde eder ve ardından bu özellikleri özellik füzyon ağı (boyun) aracılığıyla birleştirerek sırasıyla resimdeki küçük, orta ve büyük nesnelere tespit etmek için üç özellik haritası P3, P4 ve P5 (YOLOv5'te boyutlar  $80 \times 80$ ,  $40 \times 40$  ve  $20 \times 20$  boyutlarıyla ifade edilir) oluşturur [51].

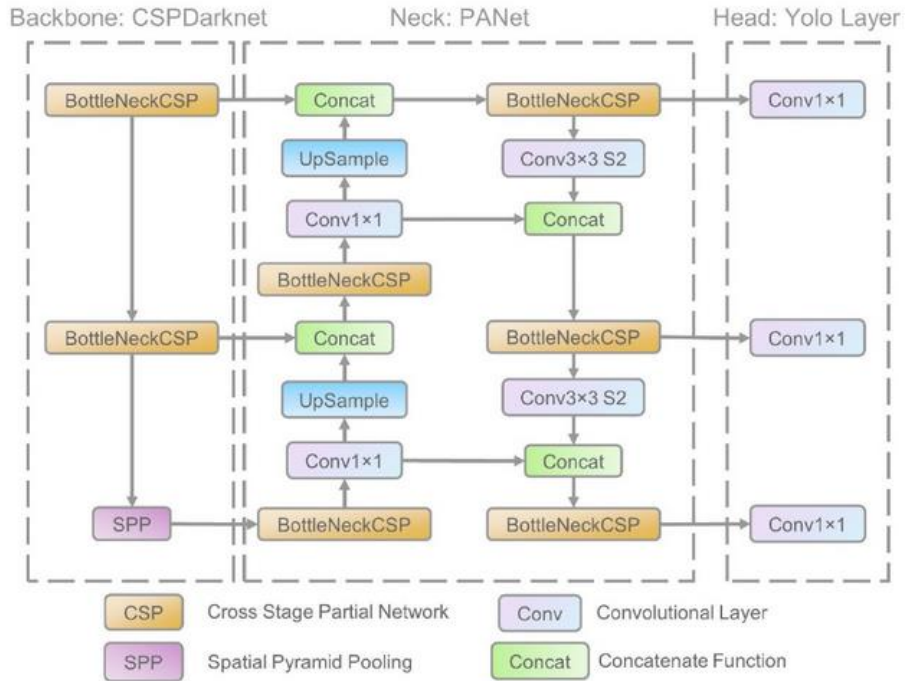


Şekil 3.14 YOLOv5'in varsayılan çıkarım akış şeması [51].

Üç özellik haritası tahmin kafasına (head) gönderildikten sonra, nesne sınıfı, sınıf güven skoru, kutu koordinatları, genişlik ve yükseklik bilgilerini içeren çok boyutlu bir dizi (BBoxes) elde etmek için önceden ayarlanmış ön çapa kullanılarak özellik haritasındaki her piksel için güven hesaplaması ve sınırlayıcı kutu regresyonu gerçekleştirilir.

Dizideki gereksiz bilgileri filtrelemek için ilgili eşik değerlerini (confthreshold, objthreshold) ayarlayarak ve bir NMS işlemi gerçekleştirerek, nihai algılama bilgileri çıktı olarak alınır.

YOLOv5, genellikle bir evrişimli sinir ağı (CNN) omurgası olarak bilinen bir "backbone" kullanır. Bu, genellikle özellik çıkarımı için kullanılan bir derin öğrenme ağıdır. YOLOv5'te, çeşitli boyutlarda ve karmaşıklıklarda backbone modelleri kullanılabilir. Bu, YOLOv5'in "extra small", "small", "medium", "large" ve "x-large" gibi çeşitli versiyonlarını tanımlayan farklı ağırlık yapılarına sahip olmasını sağlar. ResNet, CSPDarknet, EfficientNet gibi popüler CNN mimarileri, YOLOv5'in omurga yapısının bir parçası olarak kullanılabilir. YOLOv5 modelinin omurga yapısı Şekil 3.15'te gösterilmiştir.

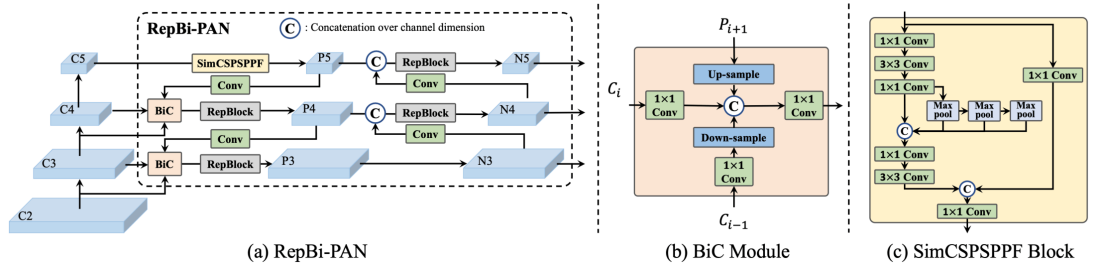


Şekil 3.15 YOLOv5'in varsayılan çıkarım akış şeması [51].

Boyun kısmında ise ilk olarak Mekansal Piramit Havuzlama'nın (SPP) bir çeşidi kullanılmış ve Yol Toplama Ağı (PANet), BottleNeckCSP'yi mimarisine dahil ederek değiştirilmiştir. PANet bir özellik piramidi ağıdır, YOLO'nun önceki sürümünde (YOLOv4) bilgi akışını iyileştirmek ve maske tahmini görevinde piksellerin doğru konumlandırılmasına yardımcı olmak için kullanılmıştır. YOLOv5'te bu ağ, ağın mimari şeklinde gösterildiği gibi CSPNet stratejisi uygulanarak değiştirilmiştir. YOLOv5, YOLOv3 ve YOLOv4 ile aynı baş yapısını kullanır. Sınırlayıcı kutuların konumunu (x, y, yükseklik, genişlik), puanları ve nesne sınıflarını tahmin eden üç konvolüsyon katmanından oluşur.

### 3.5.3. YOLOv6

YOLOv6, hız ve doğruluk arasında önemli bir denge sunarak gerçek zamanlı uygulamalar için tercih edilen bir nesne tespit modelidir. Bu model, mimarisinde ve eğitim yöntemlerinde bir dizi önemli geliştirme sunar. Bu geliştirmeler arasında Çift Yönlü Birleştirme (BiC) modülünün entegrasyonu, çapa destekli eğitim (AAT) stratejisi ve COCO veri setindeki doğruluk için optimize edilmiş bir omurga ve boyun tasarımı bulunmaktadır.



Şekil 3.16 (a) YOLOv6'nın boynu (N ve S gösterilmiştir). M/L için RepBlocks'un CSPStackRep ile değiştirildiğine dikkat edin. (b) Bir BiC modülünün yapısı. (c) Bir SimCSPSPPF bloğu [52]

Çift Yönlü Birleştirme (BiC) Modülü: YOLOv6, dedektörün boyun kısmına BiC modülünü ekleyerek lokalizasyon sinyallerini güçlendirir ve performans kazanımları elde ederken hızda önemsiz bir düşüş yaşatır.



Çapa Destekli Eğitim (AAT) Stratejisi modeli, çapa tabanlı ve çapasız paradigmalardan avantajlarını aynı anda elde etmek için AAT'yi önerir, böylece çıkarım verimliliğinden ödün verilmez. Geliştirilmiş Omurga ve Boyun Tasarımı ile YOLOv6'ya omurga ve boyun kısmını derinleştiren bir aşama ekleyerek yüksek çözünürlüklü girdilerde COCO veri setinde en son teknolojiye sahip performansa ulaşır. Kendi Kendine Damıtma Stratejisi ile YOLOv6'nın daha küçük modellerinin performansını artırması için, eğitim sırasında yardımcı regresyon dalını geliştiren ve çıkarımda belirgin bir hız düşüşünü önlemek için çıkarımdan kaldıran yeni bir kendi kendine damıtma stratejisi uygulanır [52].

YOLOv6 serisi, her biri yüksek performanslı Nesne Algılama için optimize edilmiş bir dizi model sunar. Bu modeller, farklı hesaplama ihtiyaçlarına ve doğruluk gereksinimlerine cevap vererek onları geniş bir uygulama yelpazesi için çok yönlü hale getirir.

### **3.6. SINIFLANDIRMA MODELLERİNDE BAŞARI KRİTERLERİ**

Günümüzde, sınıflandırma işlemleri için kullanılan birçok veri bilimi modelinin doğruluk sonuçları paylaşılmaktadır. Ancak, elde edilen doğruluk sonuçlarının ne kadar gerçeği yansıttığı tam bilinmemektedir. Bir makine öğrenmesi çalışmasında modelin kaç tane durumu doğru olarak tahmin ettiği çok yeterli bir değerlendirme kriteri değildir. Bu yüzden bazı teknikler geliştirilmiştir. Bunlardan birisi ise karışıklık matrisi oluşturmaktır. Karmaşıklık matrisi, bir sınıflandırma algoritmasının performansını değerlendirmek için kullanılan bir tablodur. Her bir sınıf için doğru ve hatalı tahminlerin sayısını göstererek bir sınıflandırma senaryosunda tahmin edilen sonuçlara yoğunlaştırılmış bir genel bakış sağlar. Bu görselleştirme, algoritmanın etkinliğinin değerlendirilmesine ve performansını artırabilecek eğilimlerin belirlenmesine yardımcı olur. Karışıklık matrisi, bir sınıflandırma algoritmasının ne kadar iyi performans gösterdiğini anlamada, eksik alanları vurgulamada ve iyileştirmelere rehberlik etmede değerli olduğunu kanıtlamaktadır. Çizelge 3.1'de yer aldığı gibi Gerçek Pozitif (TP), Yanlış Pozitif (FP), Yanlış Negatif (FN) ve Gerçek Negatif (TN) elemanlarından oluşan matris örneğimizde dört eleman ile temsil

edilmektedir. Bir modelin doğrulama sonucunda bir nesne için bu dört elemandan birine atama yapmaktadır.

Çizelge 3.1 Hata matrisi elemanları

		Tahmin	
		Yok	Var
Gerçek	Yok	TN	FP
	Var	FN	TP

TP, bir sınıfa ait olan obje doğru olarak tahmin edildiği durumdur. Hasta olan bir kişiye hasta tanısı koymak örneği verilebilir. FP, bir sınıfa ait olmayan obje o sınıfa ait olarak tahmin edildiği durumdur. Hasta olmayan bir kişiye hasta tanısı koymak örneği verilebilir. TN, bir sınıfa ait olmayan objeye o sınıfa ait olmadığının tahmin edildiği durumdur. Hasta olmayan bir kişiye hasta olmadığı tanısı koymak örneği verilebilir. FN, bir sınıfa ait olan obje o sınıfa ait olmadığı tahmin edildiği durumdur. Hasta olan bir kişiye hasta olmadığı tanısı koymak örneği verilebilir.

Genel ortalama kesinlik (mAP), nesne algılama modellerini değerlendirmek, performanslarını ve doğruluklarını ölçmek için çok önemli bir metriktir. Ortalama hassasiyet değerlerinin ortalamasını hesaplayan mAP, bir modelin kapasitesinin kapsamlı bir değerlendirmesini sağlar. mAP'nin hesaplanması, her sınıf için ortalama kesinliğin (AP) belirlenmesini ve ardından bunların ortalamasının alınmasını içerir. AP'yi hesaplamak için, veri kümesindeki her örnek için tahmin puanları oluşturulur. Bu puanlar modelin tahmininin güven düzeyini temsil eder. Böylece model için karışıklık matrisi oluşturulur.

mAP değeri hesaplanırken öncelikle modeldeki sınıflara ait tahminler oluşturulur. Daha sonra bu tahmin skorlarına göre sınıf etiketleri elde edilir. Bu tahmin ve etiketler sonucunda karışıklık matrisi oluşturulur. Matristen her sınıfa ait kesinlik ve duyarlılık değerleri elde edilir. Kesinlik ve duyarlılık değerlerine göre her sınıf için kesinlik-duyarlılık eğrisi elde edilir. Bu eğrilerin altında kalan kısmın alanı o sınıfa ait ortalama

kesinlik deęerini oluřturur. Elde edilen ortalama kesinlik deęerlerinin ortalaması hesaplanarak genel ortalama kesinlik (mAP) deęeri bulunur.

Kesinlik, duyarlılık ve f1-skoru, nesne algılama modellerinin performansını deęerlendirmek için kullanılan temel ölçütlerdir. Bu metrikler, modelin görüntülerdeki ilgili nesnelere belirleme becerisi hakkında önemli bilgiler sağlar. Elde edilen karışıklık matrisinden her sınıf için bu ölçütler hesaplanarak elde edilir.

Kesinlik ve duyarlılık eğrisi, nesne algılama modellerinin performansını deęerlendirmede çok önemli bir araçtır. Kesinlik ve geri çağırma deęerlerini modelin güven puanı eřiğine göre çizerek modelin doęruluęu hakkında deęerli bilgiler sağlar. Hassasiyet modelin doęru tahminlerini ölçerken, geri çağırma tüm tahminlerin yapılıp yapılmadığını ölçer.

Duyarlılık (recall), bir makine öğrenimi modelinin veri kümesindeki tüm gerçek pozitif örneklerden pozitif örnekleri (gerçek pozitifler) ne sıklıkla doęru olarak tanımladığını ölçen bir metriktir. Duyarlılığı, gerçek pozitiflerin sayısını pozitif örneklerin sayısına bölerek hesaplanır. İkincisi, gerçek pozitifleri (başarıyla tanımlanan sınıflar) ve yanlış negatif sonuçları (başarıyla tanımlanamayan sınıflar) içerir. Duyarlılık olarak kullanılan parametre ayrıca hassasiyet (sensitivity) olarak da kullanılmaktadır. Duyarlılık deęerinin hesaplanması Denklem 3.1'de gösterilmiştir.

$$Duyarlılık = \frac{TP}{TP + FN} \quad (3.1)$$

Kesinlik (Precision) ise pozitif olarak tahminlediğimiz deęerlerin gerçekten kaç adedinin pozitif olduğunu göstermektedir. Kesinlik, bir makine öğrenimi modelinin pozitif sınıfı ne sıklıkla doęru tahmin ettiğini ölçen bir metriktir. Kesinliği, doęru pozitif tahminlerin (doęru pozitifler) sayısını modelin pozitif olarak tahmin ettiği toplam örnek sayısına (hem doęru hem de yanlış pozitifler) bölerek hesaplanır. Kesinlik deęerinin hesaplanması Denklem 3.2'de gösterilmiştir.

$$Kesinlik = \frac{TP}{TP + FP} \quad (3.2)$$

F<sub>1</sub>-skoru değeri bize Kesinlik (Precision) ve Duyarlılık (Recall) değerlerinin harmonik ortalamasını göstermektedir. Basit bir ortalama yerine harmonik ortalama olmasının sebebi ise uç durumları da göz ardı edilmemesi gerektiğidir. Eğer basit bir ortalama hesaplaması olsaydı Kesinlik değeri 1 ve duyarlılık değeri 0 olan bir modelin f<sub>1</sub>-skor'u 0,5 olarak gelecektir ve bu sonuçları yanıltacaktır. F<sub>1</sub> değerinin hesaplanması Denklem 3.3'te gösterilmiştir.

$$F1 - skoru = 2 * \frac{precision * recall}{precision + recall} \quad (3.3)$$

Model içerisinde sınıf bazlı kesinlik değeri hesaplaması yapılmıştır. Ayrıca tüm sınıfların yani modelin kesinlik, duyarlılık, f<sub>1</sub>-skoru ve genel ortalama kesinlik değerleri hesaplanmıştır. Genel Ortalama Kesinlik değerinin hesaplanması Denklem 3.4'te gösterilmiştir.

$$Genel Ortalama Kesinlik = \frac{1}{\#classes} \sum_{k=1}^{\#classes} Pre(k) \quad (3.4)$$

## BÖLÜM 4

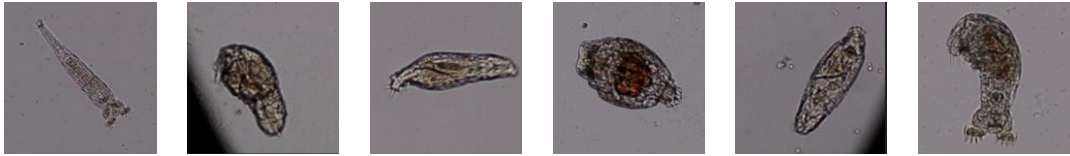
### DENEYSEL ÇALIŞMA

#### 4.1. VERİ SETİNİN OLUŞTURULMASI

Veri seti oluşturma için Nikon D5200 dijital kamera CX3 Olympus Trinoküler Mikroskop'a entegre edilerek kullanılmıştır.

##### 4.1.1. Protozoa Kültürünün Oluşturulması

Protozoa (tek hücreli canlılar), genellikle mikroskobik, bir hücreli ve ökaryotik canlıları içeren bir protista alt âlemidir. Tatlı su formları akarsulardan çok bitki içeren durgun sularda, göller ve gölcüklerde yaşarlar. Protozoaların hücre büyüklükleri genel olarak genellikle 5-250 µm arasında değişmektedir. Hareket şekilleri değişkenlik göstermektedir. İleri-geri doğru, kendi eksenini etrafında, kayma halinde hareket şekilleri vardır. Bazı türler şekil değiştirerek hareket etmeleri Şekil 4.1'de gösterilmiştir.



Şekil 4.1 Bdelloid Rotifera protozoa türünün hareket halindeki farklı şekil formları

Canlılar hayatta kalmak ve üremek için uygun sıcaklık, nem, pH vb. değerlere sahip ortamlara ihtiyaç duyar. Mikroskobik organizmalar göletler, göller, akarsular, nehirler, okyanus diplerindeki haliçler ve hatta beklenmedik bir şekilde birkaç gün boyunca devam eden yağmur birikintileri de dahil olmak üzere çeşitli su ortamlarında yaşarlar. Tek hücreli organizmalar doğada çok yaygın olmalarına rağmen, laboratuvarlarda

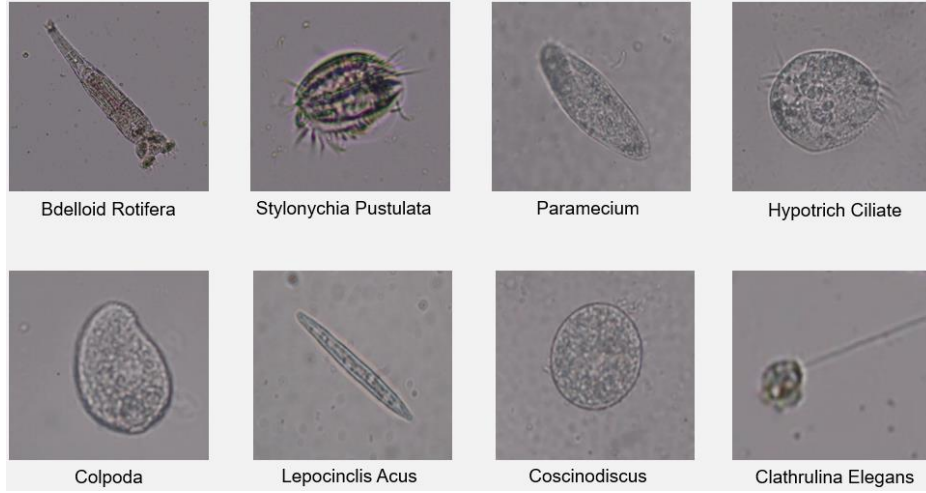
incelenmeden önce kültürlerinin hazırlanması gerekir. Bu amaçla, toplama kavanozlarımız tamamen temiz ve deterjansız olarak hazırlandı. Su örnekleri yağmur birikintilerinden toplandı. Ot, ölü yaprak kalıntısını, muz kabuğu, pirinç gibi farklı maddeler su dolu kavanozlara yerleştirildi. Toplanan maddelerin kavanozlara farklı kombinasyonlarda birleştirilmesiyle altı farklı kavanoz oluşturuldu. Kültür kutuları, bakterilerin ortaya çıkması için ortalama 8-10 gün yarı karanlık bir ortamda bekletilmiştir.

#### **4.1.2. Kültürlerden Protozoa Görüntülerinin Elde Edilmesi**

Oluşturulan kültürlerde bir hafta sonra protozoa türleri belirmeye başlamıştır. Belirli zaman aralıklarında kavanozların dip, orta ve yüzey kısımlarından numune alınarak mikroskop altında incelenmiştir. Oluşturulan kültürlerden belirli zaman aralıklarında numuneler alınarak incelenmiş ve protozoa mevcut olduğunda görüntüler elde edilmiştir. Gözlemlenen protozoaların birçoğunun hareketli olması nedeniyle ve video üzerinde işleme yapmak amacıyla görüntüler video formatında alınmıştır.

#### **4.1.3. Veri Setinin Oluşturulması**

Video kayıtlarından veri seti oluşturulması, bilgisayarlı görüş alanında önemli bir adımdır ve çeşitli uygulamalarda kullanılan veri setlerinin oluşturulmasında temel bir rol oynar. Bu bağlamda mikroskoptan alınan video içeriğinden resimler elde edilmiştir. Video olarak alınan görüntülerde birçok protozoa türünün sürekli hareket halinde olması nedeniyle her saniyede kare setlerinden rastgele üç görüntü elde edilmiştir. Videolardan elde edilen görüntüler ile fotoğraf olarak çekilen görüntüler birleştirilerek veri seti oluşturulmuştur. Veri seti 8 türe ait 1280×720 çözünürlükte 4653 renkli görüntüden oluşmaktadır. Veri setimiz Şekil 4.2'de gösterilen Bdelloid Rotifera, Stylonychia Pustulata, Paramecium, Hypotrich Ciliate, Colpoda, Lepocinclis Acus, Clathulina Elegans tatlı su protozoa türlerini içermektedir.



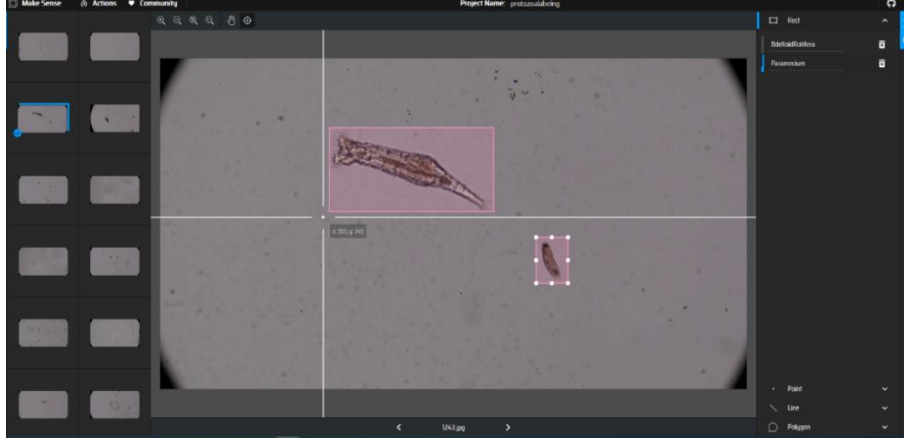
Şekil 4.2 Veri setindeki protozoa türleri

Video framelerinden veri seti oluşturmak, derin öğrenme modelleri gibi karmaşık ve veriye dayalı algoritmaların eğitimi için temel bir adımdır. Doğru ve çeşitli bir veri seti, modelin genelleme yeteneğini artırır ve gerçek dünya senaryolarında daha iyi performans sağlar. Bu nedenle, video verilerinden doğru ve etiketli bir veri seti oluşturmak, birçok bilgisayarlı görüş uygulaması için kritik öneme sahiptir.

#### 4.2. VERİ ETİKETLEME

Görüntüleri etiketlemek için çevrimiçi bir araç olan MakeSense.ai sistemi kullanılmıştır [53]. Sistemden YOLO formatı, VOC XML formatı ve tek bir CSV dosyası olarak etiket verileri oluşturulmaktadır.

MakeSense sistemine 4653 görüntüden oluşan veri kümesi yüklenmiş ve 8 sınıfa ait protozoa türleri etiketlenmiştir. Etiketleme işlemi Şekil 4.3'te gösterilen MakeSense sisteminde gerçekleştirilmiştir. Bazı görüntülerde tek bir nesne bulunurken bazı görüntülerde 25 nesne bulunmaktadır. Tüm görüntüler etiketlendikten sonra etiketli veriler tüm formatlarda elde edilmiştir.



Şekil 4.3 Makesense.io ile etiketleme işlemi



## BÖLÜM 5

### DENEYSEL SONUÇLAR VE TARTIŞMA

#### 5.1. YOLOV4 ALGORİTMASININ PERFORMANSININ DEĞERLENDİRİLMESİ

Bu çalışmada, Darknet-53 çatısı üzerinde YOLOv4 kullanılarak gerçek zamanlı protozoa tespiti gerçekleştirilmiştir. Çalışmanın değerlendirmesi ortalama doğruluk, ortalama hassasiyet, ortalama duyarlılık, f1-skor ve mAP metrikleri ile yapılmıştır. Eğitimlerde batch sayısı, iterasyon sayısı ve girdi boyutu büyüklüklerine göre genel ortalama kesinlik değerleri incelenmiştir. Eğitim için 416x416, 512x512, 608x608 girdi boyutlarına göre ayrı ayrı eğitilmiştir. Ayrıca batch sayısı da 32 ve 64 olarak ayrı ayrı çalıştırılmıştır.

Çizelge 5.1 Farklı girdi boyutlarına göre mAP değerleri

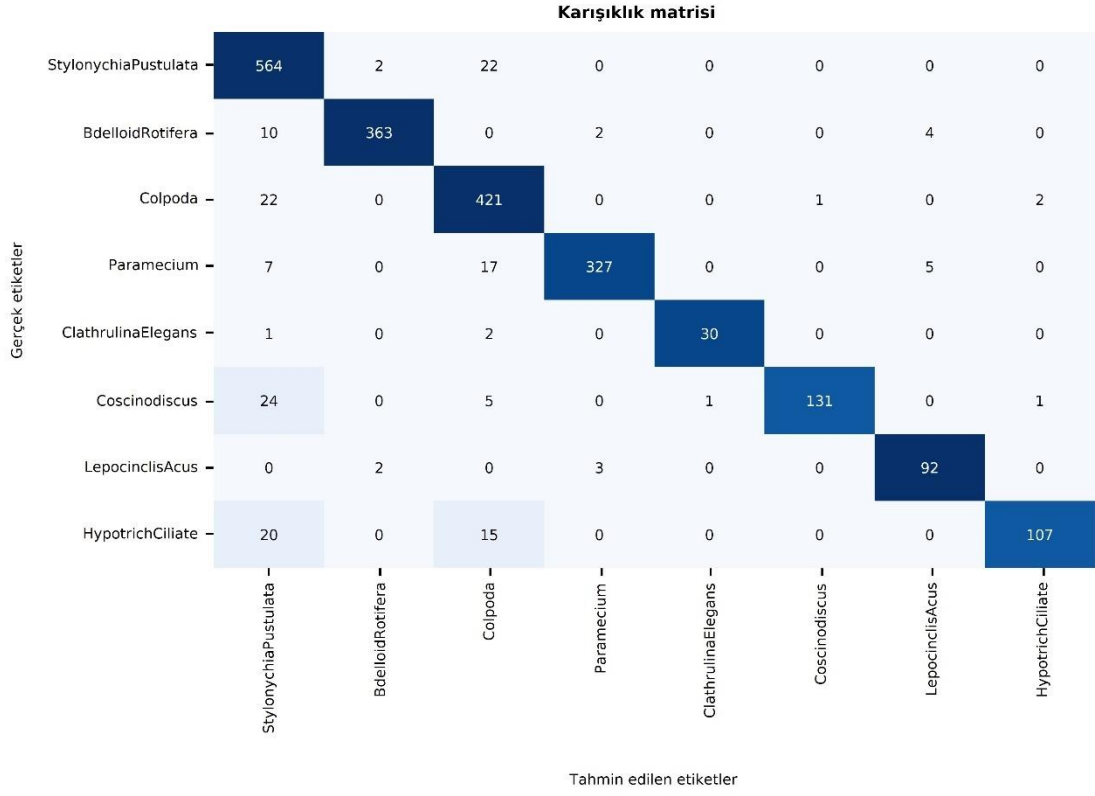
Model	Batch Sayısı	İterasyon				
		1000	2000	3000	4000	5000
YOLOv4 (416x416)	64	74,65	<b>97,98</b>	97,78	97,81	97,74
YOLOv4 (416x416)	32	67,95	97,43	97,5	<b>97,76</b>	97,72
YOLOv4 (512x512)	64	84,48	97,98	98,18	<b>98,3</b>	98,08
YOLOv4 (512x512)	32	59,72	97,53	97,75	<b>98,07</b>	97,94
YOLOv4 (608x608)	64	83,86	<b>97,78</b>	97,7	97,74	97,68
YOLOv4 (608x608)	32	71,98	97,64	<b>97,69</b>	97,1	97,32

Eđitim s¼recinde her 1000 iterasyon sonucunda elde edilen verilere g¼re mAP deęerleri edilerek izelge 5.1 oluřturulmuřtur. izelge incelendięinde en y¼ksek AP deęeri, “512x512” giriř g¼r¼nt¼s¼ ç¼z¼n¼rl¼ę¼nde 4000 iterasyon sonunda 98,3 olarak ¼lç¼lm¼řt¼r. En y¼ksek mAP deęerinin elde edildięi “512x512” aę giriř g¼r¼nt¼ boyutunda her 1000 iterasyon aęırlıęı iin alınan bařarı deęerlendirme ¼lç¼tleri izelge 5.2’de incelenmiřtir.

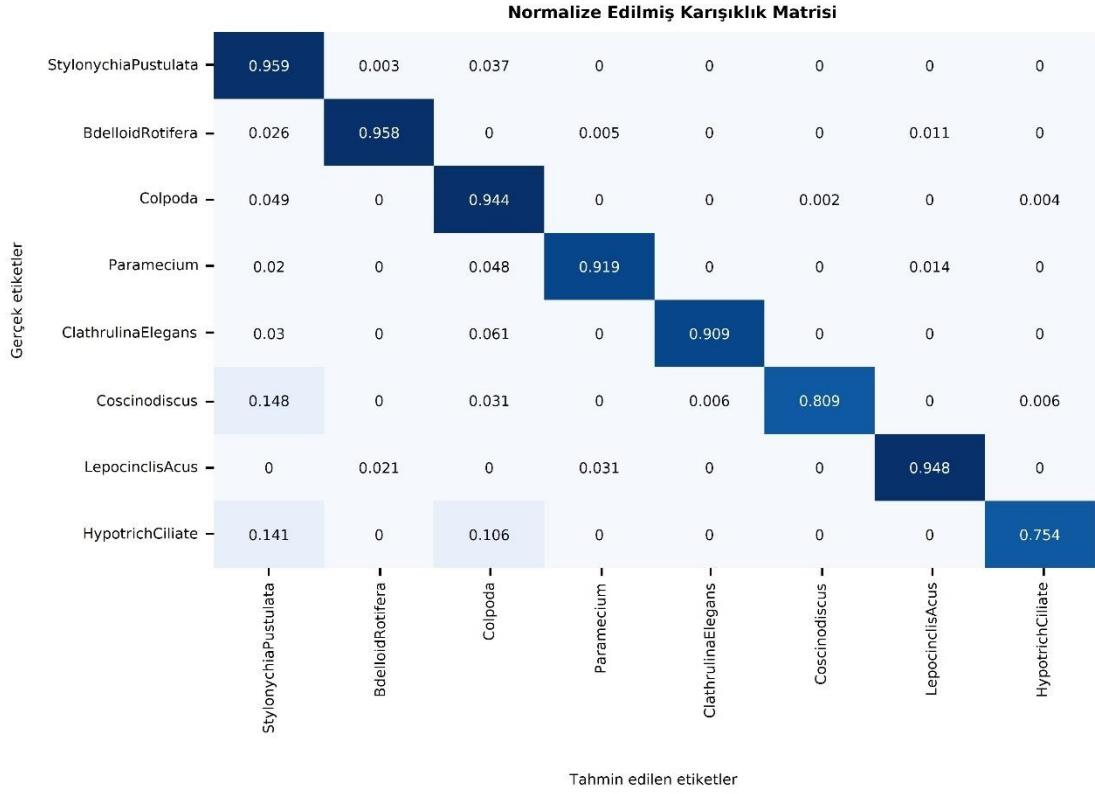
izelge 5.2 512x512 girdi boyutunda eđitim sonuları

Input Size	Batch	İterasyon	Kesinlik	Duyarlılık	F1-skoru	mAP(%)
512x512	32	1000	0,49	0,79	0,6	59,72
512x512	32	2000	0,91	0,98	0,94	97,53
512x512	32	3000	0,91	0,98	0,95	97,75
512x512	32	4000	0,92	0,99	0,95	98,07
512x512	32	5000	0,92	0,98	0,95	98,08
512x512	64	1000	0,7	0,95	0,8	84,48
512x512	64	2000	0,91	0,99	0,95	97,98
512x512	64	3000	0,92	0,99	0,95	98,18
512x512	64	4000	0,92	0,99	0,95	98,3
512x512	64	5000	0,93	0,98	0,95	97,94

Genel Ortalama Kesinlik (mAP), nesne algılama modellerini deęerlendirmek, performanslarını ve doęruluklarını ¼lmek iin ok ¼nemli bir metriktir. Ortalama kesinlik deęerlerinin ortalamasını hesaplayan mAP, bir modelin kapasitesinin kapsamlı bir deęerlendirmesini saęlar. řekil 5.1 ve řekil 5.2’de g¼r¼ld¼ę¼ gibi, aęın performansını duyarlılık, kesinlik ve f1-skor aısından istatistiksel metriklerle analiz etmek iin gerek / tahmin edilen sınıfların kombinasyonlarını g¼rselleřtiren karıřıklık matrisleri ve normalize edilmiř karıřıklık matrisleri oluřturulmuřtur.



Şekil 5.1 Karışıklık matrisi



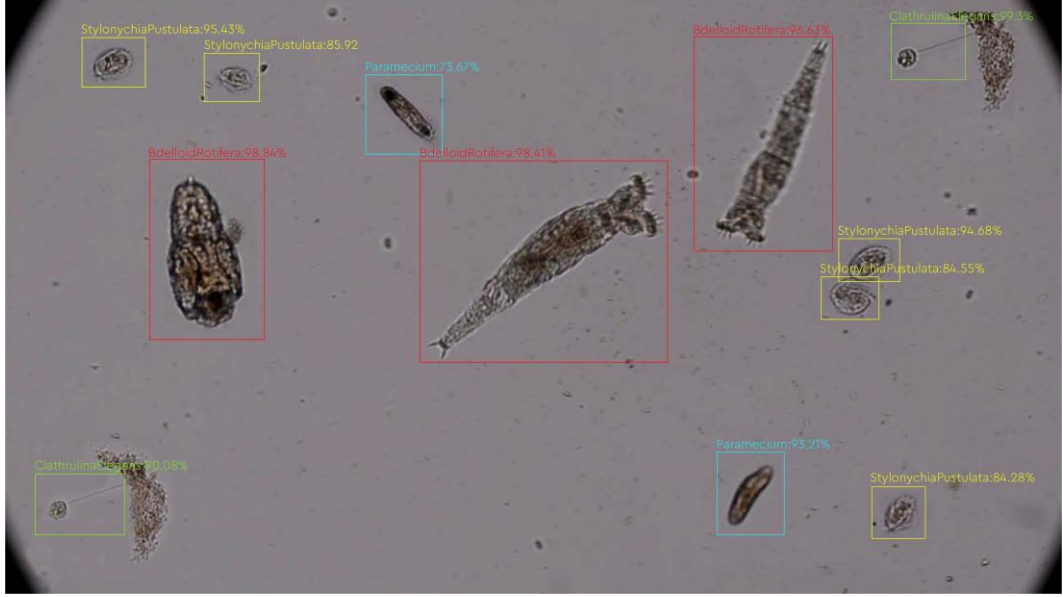
Şekil 5.2 Normalize edilmiş karışıklık matrisi

Sınıf bazlı ortalama hassasiyet deęerleri de elde edilmiřtir. izelge 5.3 sekiz farklı sınıfa ait sınıf ortalama hassasiyet deęerlerini gstermektedir. Sınıf-3'n (Colpoda) ortalama hassasiyet deęeri 426 doęru pozitif ve 60 yanlış pozitif deęer ile %95,56'dır. Colpoda protozoası, dięer bazı protozoa trleri ile řekil benzerlięi nedeniyle YOLOv4 tarafından en dřk mAP deęerine sahiptir. Bu, tm sınıflar arasında en dřk algıdır. Sınıf-6 (Coscinodiscus) tm sınıflar arasında en yksek mAP deęerine ulařmıřtır.

izelge 5.3 YOLOv4 ile sınıf bařına eęitim sonuları

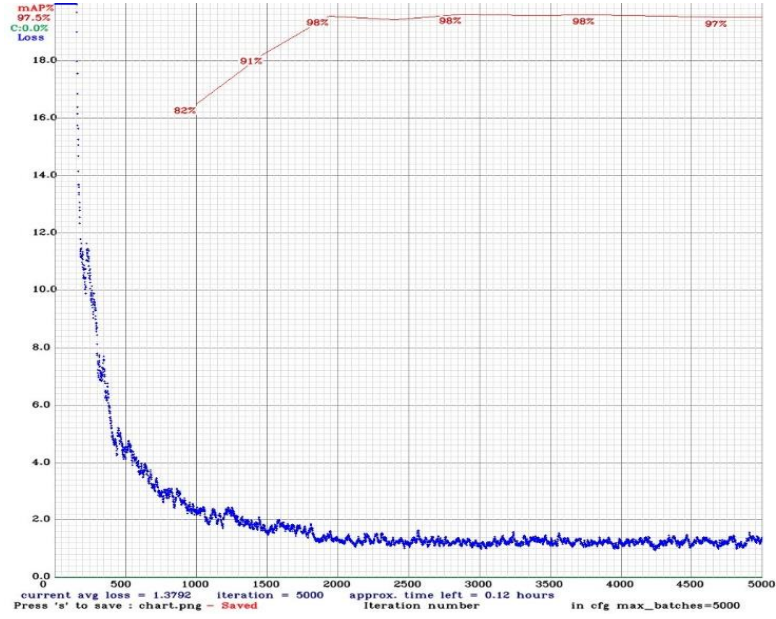
Sınıflar	AP (Ortalama Kesinlik)	TP (Doęru Pozitif)	FP (Yanlış Pozitif)
Sınıf-1 (Stylonychia Pustulata)	98,04	564	84
Sınıf-2 (Bdelloid Rotifera)	98,64	363	4
Sınıf-3 (Colpoda)	95,28	421	61
Sınıf-4 (Paramecium)	99,38	327	5
Sınıf-5 (Clathrulina Elegans)	96,46	30	1
Sınıf-6 (Coscinodiscus)	100	131	1
Sınıf-7 (Lepocinclis Acus)	98,83	92	9
Sınıf-8 (Hypotrich Ciliate)	100	107	3

YOLOv4 tespit edilen nesnelerin sınırlayıcı kutularını oluřturur. řekil 5.3'da protozoa sınıflarının IoU sınırlayıcı kutu deęerleri gsterilmektedir. řekilde grldę gibi kırmızı kutularla iřaretlenmiř yerler Bdelloid Rotifera'yı, mavi kutular Paramecium'u, sarı kutular Stylonychia Pustulata'yı, yeřil kutular ise Clathrulina Elegans'ı gstermektedir. Bu kutular apraz olasılık haritaları arasında en yksek gven puanını gstermektedir. rneęin, řekil 6'da gsterildięi gibi, protozoa srekli hareket halinde olmasına ve srekli řekil deęiřtirmesine raęmen, Bdelloid Rotifera grnt zerinde %96-98 oranında tespit edilmiřtir.



Şekil 5.3 Protozoa tür tespit test görüntüsü

Şekil 5.4, geçerli verilerin hesaplanan mAP değerlerini eğitim iterasyonu eksenindeki kayıp fonksiyonu grafiğiyle birlikte göstermektedir; model daha eğitilmiş hale geldikçe ve YOLO'yu öğrendikçe kayıp değeri düşmektedir. Tipik olarak, mAP ilk birkaç eğitim iterasyonu sırasında yükselir ve belirli bir eğitim miktarından sonra derin öğrenme modelinin aşırı uyum sağlamasının bir sonucu olarak düşer. Şekil 7, geçerli protozoa sınıfları verileri üzerinde eğitim tamamlandıktan sonraki doğruluk ve kayıp oranlarını göstermektedir. YOLOv4, 0,5 IoU eşik değeri ile ortalama %97 doğruluk ve 1,3792 kayıp değeri elde etmiştir. Önerilen YOLOv4 modeli, test sınıfındaki 900 görüntüden 2063 nesnenin 52'sini yanlış sınıflandırmıştır. Bu sonuç, YOLOv4'ün nesne tespit doğruluğunun oldukça yüksek olduğunu göstermektedir.



Şekil 5.4 mAP ve kayıp grafiği.

Önerilen yöntem yüksek bir başarı oranı sağladığı için bu uygulama daha da geliştirilmeye açıktır. Sekiz protozoa türünü içeren veri kümesi, araştırmacılara önemli bir çalışma alanı sağlamak için daha da genişletilebilir. Ayrıca, zenginleştirilmiş veri kümesi tıp fakültelerinde kullanılacak bir tıbbi biyoloji uygulamasına dönüştürülebilir. Düşük ışık, yüksek arka plan gürültüsü veya değişen boyutlar gibi zorlu görüntüleme koşullarında performansını artırmak için YOLOv4 modeline ince ayar yapmak için yöntemler araştırılabilir. Etkileşimli bir kullanıcı arayüzü geliştirmek, kullanıcıların, özellikle de ana uzmanların, modelin tahminlerine gerçek zamanlı olarak açıklama eklemelerine ve düzeltmelerine olanak tanıyacaktır. Protozoa tespit algoritmasını otomatik numune işleme sistemleriyle entegre etmek için otomasyon uzmanlarıyla iş birliği yapılabilir.

## 5.2. YOLO Modellerinin Karşılaştırılması

YOLOv4, YOLOv5 ve YOLOv6'nın eğitim sonuçları karşılaştırıldığında, her sürüm nesne algılamadaki güçlü yönlerini ve ilerlemelerini göstermektedir. YOLOv4, etkili ortalama hassasiyet ve geri çağırma metriklerini korurken, çeşitli veri kümelerinde sağlam performansının göstergesi olan sürekli yüksek mAP değerleri elde edilmektedir. YOLOv4 tarafından atılan temel üzerine inşa edilen YOLOv5, doğru ve kapsamlı nesne algılama görevlerindeki etkinliğini gösteren başarılı ortalama kesinlik

ve duyarlılık değerlerinin yanı sıra genellikle YOLOv4 ile karşılaştırılabilir mAP skorları elde eder. YOLOv6, en son yineleme olarak, mimari iyileştirmelerden ve rafine eğitim stratejilerinden yararlanarak performans ölçümlerini daha da geliştirmeyi amaçlamakta ve daha yüksek mAP puanları için çaba göstermeye devam ederken potansiyel olarak iyileştirilmiş ortalama kesinlik ve duyarlılık değerleri ile sonuçlanmaktadır.

Çizelge 5.4'te gösterildiği üzere YOLOv4 modeli için mAP 0,983, ortalama kesinlik 0,92 ve ortalama duyarlılık olarak 0,99 değerleri elde edilmiştir. YOLOv5s modeli için mAP 0,988, ortalama kesinlik 0,98 ve ortalama duyarlılık olarak 0,982 değerleri elde edilmiştir. YOLOv6s modeli için mAP 0,967, ortalama kesinlik 0,976 ve ortalama duyarlılık olarak 0,978 değerleri elde edilmiştir. Eğitim verisi üzerinde YOLOv4 ile mAP ve ortalama duyarlılık değerleri ile diğer modellere göre daha iyi sonuçlar vermiştir. YOLOv5s modeli ortalama kesinlik ve ortalama duyarlılık değerleri üzerine iyileştirme çalışmaları Çizelge 6'da görülmektedir. Protozoa verileri üzerinde YOLOv5s modeli YOLOv6'ya göre her üç parametre için de daha başarılı sonuçları vermektedir. YOLOv5s, 0,988'lik en yüksek mAP ile öne çıkmakta ve çeşitli veri kümelerinde üstün genel nesne algılama yeteneğini göstermektedir. Ortalama hassasiyet açısından, YOLOv4 ve YOLOv5s benzer değerler gösterirken, YOLOv5s 0,98 ile biraz önde, YOLOv6s ise 0,976 ile biraz geridedir. Bununla birlikte, ortalama duyarlılık dikkate alındığında, YOLOv4 0,99'luk bir değerle diğer modellere göre daha yüksek oranda gerçek pozitifleri yakalama yeteneğini ortaya koymaktadır. YOLOv6s'nın biraz daha düşük bir mAP sergilemesine rağmen, rekabetçi genel ortalama kesinlik ve duyarlılık, kesinlik ve duyarlılık arasında nüanslı bir değiş tokuş olsa da belirli kullanım durumlarında uygulanabilirliğini göstermektedir. Genel olarak, YOLOv4, YOLOv5 ve YOLOv6 arasındaki seçim belirli görev gereksinimlerine bağlı olabilir; YOLOv5'ler yüksek mAP'ye öncelik veren uygulamalar için etkin bir seçenek sunarken, YOLOv4 geri çağırma en üst düzeye çıkarma konusundaki üstünlüğünü göstermektedir.

Çizelge 5.4 YOLOv4, YOLOv5, YOLOv6 modellerinin karşılaştırılması

<b>Models</b>	<b>mAP</b>	<b>Average Precision</b>	<b>Average Recall</b>
YOLOv4	0,983	0,92	0,99
YOLOv5s	0.988	0,98	0.982
YOLOv6s	0.967	0,976	0,978



## BÖLÜM 6

### SONUÇLAR VE DEĞERLENDİRMELER

Tatlı sularda bulunan protozoa türleri birçok nedenle önem taşımaktadır. Çevresel izleme açısından çevre sağlığının göstergeleri olarak çeşitli ekosistemlerde önemli roller oynamaktadır. Toprak, su ve diğer habitatlardaki protozoa popülasyonlarının izlenmesi, ekosistem dinamikleri, besin döngüsü ve genel çevre kalitesi hakkında bilgi sağlayabilir. Bazı protozoa türleri su kaynaklı hastalıklara neden olarak insanlarda ve hayvanlarda gastrointestinal hastalıklara yol açabilir. Bu protozoaların su kaynaklarında tespit edilmesi, güvenli içme suyunun sağlanması ve su kaynaklı hastalık salgınlarının önlenmesi için çok önemlidir. Genel olarak, protozoa tespiti halk sağlığının korunması, çevresel bütünlüğün korunması, tıbbi teşhislerin iyileştirilmesi, gıda güvenliğinin sağlanması ve bilimsel araştırma ve eğitimin ilerletilmesinde kritik bir rol oynamaktadır. Gerçek zamanlı protozoa tespit sistemlerinin geliştirilmesi, çeşitli bağlamlarda protozoal tehditleri etkili bir şekilde izleme ve bunlara yanıt verme becerimizi artırabilir.

Derin öğrenme ile bahsedilen bu problemlerin üstesinden gelmek ve protozoa tür ve sayılarının değerlendirme işlemlerini otomatikleştirmek için oldukça etkili bir araçtır. Ayrıca, otomatik ve gerçek zamanlı yapılan bu analiz zaman ve tespit başarısı açısından verimli olmasının yanında teknik uzman olmadan çalıştırılabilir bir model oluşturulabilir. Modelde yüksek miktarda veri kullanarak yapılan öğrenme işleminden sonra protozoalar sınıflandırılarak tespit işlemi yapılabilir ve gürültülü resimlerdeki protozoalar da yüksek doğrulukla tespit edilebilir.

Bu çalışmanın amacı, protozoaların otomatik olarak gerçek zamanlı tanınması, bölütlenmesi ve sınıflandırılması için farklı türlere uygulama potansiyeli olan bir teknik oluşturmaktır. Uygulama sonuçlarına göre, kullanılan görüntülerde aynı türe ait protozoanlar farklı boyutlarda ve hareket halinde şekil değiştirmelerine rağmen

başarılı sonuçlar elde edilmiştir. Çalışmadaki veriler laboratuvarında kendi oluşturduğumuz kültürler kullanılarak elde edilmiştir. Bu çalışmada Bdelloid Rotifera, Stylonychia Pustulata, Paramecium, Hypotrich Ciliate, Colpoda, Lepocinlis Acus, Clathrulina Elegans türlerine ait toplam 4653 görüntü kullanılmıştır. Görüntüler %80 eğitim ve %20 test ve seti olarak ayrılmıştır. Bazı görüntülerde 1 adet protozoa olmasının yanında bazı görüntülerde 30-40 civarı protozoa da yer almaktadır. Tüm görüntülerde toplam 11252 adet protozoa yer almaktadır. YOLOv4 algoritması kullanılarak oluşturulan modelde f1-skor 0.95, ortalama kesinlik 0.92, duyarlılık 0.99 ve mAP 0.983 olarak elde edilmiştir. Önerilen model %98 doğruluk oranına ulaşmıştır.

Protozoa türlerinin derin öğrenme ile gerçek zamanlı olarak tespit edilmesi, YOLO algoritmasının ilk kez gerçek zamanlı olarak uygulanması ve veri setinin tarafımızdan oluşturulması bu çalışmaya özgünlük katmaktadır. Geliştirilen sistem parazitoloji ve parazitoloji çalışmalarında kullanılabilir. Su birikintilerinden tek hücreli organizmaların varlığı ve çeşit sayısı gibi sonuçlar elde edilebilir. Sistem ayrıca bölgedeki araştırmacılara tek hücreli organizmaların yoğunluğu hakkında bilgi ve yazılım desteği sağlaması açısından da önem kazanıyor. Son olarak Karabük Üniversitesi Tıp Fakültesi dönem 1'de tıbbi biyoloji uygulaması olan tek hücreli canlılar dersine kaynak olması ve öğrencilerin kendilerini yetiştirmeleri açısından bir eğitim destek aracı olarak da kullanılabilir. Son olarak bu çalışma için özel olarak oluşturulan etiketli veri seti daha sonra paylaşılacak ve akademik çalışmalara katkı sağlayacaktır. Ayrıca veri seti genişletilerek daha fazla ve farklı protozoa türlerinin de eklenmesiyle model büyütülecektir.

Karabük ilinde bulunan farklı ilçelerde yer alan göl ve göletlerden su numuneleri toplanarak protozoa çeşitliliği gözlemlenebilir. Bunun yanında zengin bir veri seti de oluşturulmuş olacaktır.

## KAYNAKLAR

1. Pandey, P. K., Kass, P. H., Soupir, M. L., Biswas, S., and Singh, V. P., "Contamination of water resources by pathogenic bacteria", *AMB Express*, 4 (1): 1–16 (2014).
2. Al Bayati, M. A. Z. and Çakmak, M., "Real-Time Vehicle Detection for Surveillance of River Dredging Areas Using Convolutional Neural Networks", *International Journal Of Image, Graphics And Signal Processing*, 15 (5): 17–28 (2023).
3. Atila, Ü., Baydilli, Y. Y., Sehirli, E., and Turan, M. K., "Classification of DNA damages on segmented comet assay images using convolutional neural network", *Computer Methods And Programs In Biomedicine*, 186: (2020).
4. Baydilli, Y. Y. and Atila, Ü., "Classification of white blood cells using capsule networks", *Computerized Medical Imaging And Graphics*, 80: (2020).
5. Duru, A., Karaş, İ. R., Karayürek, F., and Gülses, A., "A Deep Learning Approach for Classification of Dentinal Tubule Occlusions", *Applied Artificial Intelligence*, 36 (1): (2022).
6. Ahmed, A. H., Alwan, H. B., and Çakmak, M., "Convolutional Neural Network-Based Lung Cancer Nodule Detection Based on Computer Tomography", *Lecture Notes in Networks and Systems*, Springer, 89–102 (2023).
7. Salih, M. M. M. and Cakmak, M., "Neural Network Approach for Classification and Detection of Chest Infection", *2022 2nd International Conference On Computing And Machine Intelligence, ICMI 2022 - Proceedings*, (2022).
8. Zohre Fasihfar, Hamidreza Rokhsati, Hamidreza Sadeghsalehi, Mustafa Ghaderzadeh, M. G., "AI-driven malaria diagnosis: developing a robust model for accurate detection and classification of malaria parasites", *Iranian Journal Of Blood And Cancer*, 9 (3): 56- (2023).
9. Ghaderzadeh, M., Hosseini, A., Asadi, F., Abolghasemi, H., Bashash, D., and Roshanpoor, A., "Automated Detection Model in Classification of B-Lymphoblast Cells from Normal B-Lymphoid Precursors in Blood Smear Microscopic Images Based on the Majority Voting Technique", *Scientific Programming*, 2022: (2022).
10. Ghaderzadeh, M., Asadi, F., Ramezan Ghorbani, N., Almasi, S., and Taami, T., "Toward artificial intelligence (AI) applications in the determination of

- COVID-19 infection severity: considering AI as a disease control strategy in future pandemics", *Iranian Journal Of Blood And Cancer*, 15 (3): 93–111 (2023).
11. Hosseini, A., Eshraghi, M. A., Taami, T., Sadeghsalehi, H., Hoseinzadeh, Z., Ghaderzadeh, M., and Rafiee, M., "A mobile application based on efficient lightweight CNN model for classification of B-ALL cancer from non-cancerous cells: A design and implementation study", *Informatics In Medicine Unlocked*, 39 (January): 101244 (2023).
  12. Skotarczak, B., "Methods for parasitic protozoans detection in the environmental samples", *Parasite*, 16 (3): 183–190 (2009).
  13. Maas, L., Dorigo-Zetsma, J. W., de Groot, C. J., Bouter, S., Plötz, F. B., and van Ewijk, B. E., "Detection of intestinal protozoa in paediatric patients with gastrointestinal symptoms by multiplex real-time PCR", *Clinical Microbiology And Infection*, 20 (6): 545–550 (2014).
  14. Le Calvez, T., Trouilhé, M. C., Humeau, P., Moletta-Denat, M., Frère, J., and Héchar, Y., "Detection of free-living amoebae by using multiplex quantitative PCR", *Molecular And Cellular Probes*, 26 (3): 116–120 (2012).
  15. Baltrušis, P. and Höglund, J., "Digital PCR: modern solution to parasite diagnostics and population trait genetics", *Parasites And Vectors*, 16 (1): 1–9 (2023).
  16. Houssin, T., Bridle, H., and Senez, V., "Electrochemical detection", *Waterborne Pathogens*, 147–187 (2021).
  17. da Silva, A. D., Paschoalino, W. J., Neto, R. C., and Kubota, L. T., "Electrochemical point-of-care devices for monitoring waterborne pathogens: Protozoa, bacteria, and viruses – An overview", *Case Studies In Chemical And Environmental Engineering*, 5 (December 2021): 100182 (2022).
  18. McGrath, J. S., Honrado, C., Spencer, D., Horton, B., Bridle, H. L., and Morgan, H., "Analysis of Parasitic Protozoa at the Single-cell Level using Microfluidic Impedance Cytometry", *Scientific Reports*, 7 (1): 1–11 (2017).
  19. Bharadwaj, P., Tripathi, D., Pandey, S., Tapadar, S., Bhattacharjee, A., Das, D., Palwan, E., Rani, M., and Kumar, A., "Molecular Biology Techniques for the Detection of Contaminants in Wastewater", *Wastewater Treatment: Cutting-Edge Molecular Tools, Techniques And Applied Aspects*, 217–235 (2021).
  20. Boztoprak, H. and Özbay, Y., "Detection of protozoa in wastewater using ANN and active contour in image processing", *Istanbul University - Journal Of Electrical And Electronics Engineering*, 13 (2): 1661–1666 (2013).
  21. Abidin, S. R., Salamah, U., and Nugroho, A. S., "Segmentation of malaria parasite candidates from thick blood smear microphotographs image using active contour without edge", (2016).

22. Dave, I. R. and Upla, K. P., "Computer aided diagnosis of Malaria disease for thin and thick blood smear microscopic images", *2017 4th International Conference On Signal Processing And Integrated Networks, SPIN 2017*, 561–565 (2017).
23. Traore, B. B., Kamsu-Foguem, B., and Tangara, F., "Deep convolution neural network for image recognition", *Ecological Informatics*, 48: 257–268 (2018).
24. Mathison, B. A., Kohan, J. L., Walker, J. F., Smith, R. B., Ardon, O., Ardon, O., Couturier, M. R., and Couturier, M. R., "Detection of intestinal protozoa in trichrome-stained stool specimens by use of a deep convolutional neural network", *Journal Of Clinical Microbiology*, 58 (6): 1–13 (2020).
25. Pho, K., Mohammed Amin, M. K., and Yoshitaka, A., "Segmentation-driven Hierarchical RetinaNet for Detecting Protozoa in Micrograph", *International Journal Of Semantic Computing*, 13 (3): 393–413 (2019).
26. Nakasi, R., Mwebaze, E., Zawedde, A., Tusubira, J., Akera, B., and Maiga, G., "A new approach for microscopic diagnosis of malaria parasites in thick blood smears using pre-trained deep learning models", *SN Applied Sciences*, 2 (7): 1–7 (2020).
27. Osaku, D., Cuba, C. F., Suzuki, C. T. N., Gomes, J. F., and Falcão, A. X., "Automated diagnosis of intestinal parasites: A new hybrid approach and its benefits", *Computers In Biology And Medicine*, 123: 103917 (2020).
28. Górriz, M., Aparicio, A., Raventós, B., Vilaplana, V., Sayrol, E., and López-Codina, D., "Leishmaniasis parasite segmentation and classification using deep learning", *Lecture Notes In Computer Science (Including Subseries Lecture Notes In Artificial Intelligence And Lecture Notes In Bioinformatics)*, 10945 LNCS: 53–62 (2018).
29. Hoorali, F., Khosravi, H., and Moradi, B., "Automatic Bacillus anthracis bacteria detection and segmentation in microscopic images using UNet++", *Journal Of Microbiological Methods*, 177 (September): 106056 (2020).
30. de Souza Oliveira, A., Guimarães Fernandes Costa, M., das Graças Vale Barbosa, M., and Ferreira Fernandes Costa Filho, C., "A new approach for malaria diagnosis in thick blood smear images", *Biomedical Signal Processing And Control*, 78 (January): (2022).
31. Abdurahman, F., Fante, K. A., and Aliy, M., "Malaria parasite detection in thick blood smear microscopic images using modified YOLOV3 and YOLOV4 models", *BMC Bioinformatics*, 22 (1): 1–17 (2021).
32. Nakasi, R., Mwebaze, E., and Zawedde, A., "Mobile-aware deep learning algorithms for malaria parasites and white blood cells localization in thick blood smears", *Algorithms*, 14 (1): 1–16 (2021).

33. Zhang, C., Jiang, H., Jiang, H., Xi, H., Chen, B., Liu, Y., Juhas, M., Li, J., and Zhang, Y., "Deep learning for microscopic examination of protozoan parasites (Review)", *Computational And Structural Biotechnology Journal*, 20: 1036–1043 (2022).
34. Althomali, R. H., Abdu Musad Saleh, E., Gupta, J., Mohammed Baqir Al-Dhalimy, A., Hjazzi, A., Hussien, B. M., AL-Erjan, A. M., Jalil, A. T., Romero-Parra, R. M., and Barboza-Arenas, L. A., "State-of-the-art of portable (bio)sensors based on smartphone, lateral flow and microfluidics systems in protozoan parasites monitoring: A review", *Microchemical Journal*, 191 (April): 108804 (2023).
35. Rayan SULAIMAN KHALAF, "Guns Detection Using YOLO Algorithm (TEZ)", (May): (2021).
36. Redmon, J. and Farhadi, A., "YOLOv3: An incremental improvement", *ArXiv*, (2018).
37. Ning, C., Zhou, H., Song, Y., and Tang, J., "Inception Single Shot MultiBox Detector for object detection", *2017 IEEE International Conference On Multimedia And Expo Workshops, ICMEW 2017*, (July): 549–554 (2017).
38. Ren, S., He, K., Girshick, R., and Sun, J., "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 39 (6): 1137–1149 (2017).
39. Girshick, R., Donahue, J., Darrell, T., and Malik, J., "Rich feature hierarchies for accurate object detection and semantic segmentation", *Proceedings Of The IEEE Computer Society Conference On Computer Vision And Pattern Recognition*, 580–587 (2014).
40. Van De Sande, K. E. A., Uijlings, J. R. R., Gevers, T., and Smeulders, A. W. M., "Segmentation as selective search for object recognition", *Proceedings Of The IEEE International Conference On Computer Vision*, 1879–1886 (2011).
41. Girshick, R., "Fast R-CNN", *Proceedings Of The IEEE International Conference On Computer Vision*, 2015 Inter: 1440–1448 (2015).
42. He, K., Zhang, X., Ren, S., and Sun, J., "Spatial pyramid pooling in deep convolutional networks for visual recognition", *Lecture Notes In Computer Science (Including Subseries Lecture Notes In Artificial Intelligence And Lecture Notes In Bioinformatics)*, 8691 LNCS (PART 3): 346–361 (2014).
43. Girshick, R., "Fast R-CNN", *Proceedings Of The IEEE International Conference On Computer Vision*, 2015 Inter: 1440–1448 (2015).

44. He, K., Gkioxari, G., Dollár, P., and Girshick, R., "Mask R-CNN", *IEEE Transactions On Pattern Analysis And Machine Intelligence*, 42 (2): 386–397 (2020).
45. He, K., Zhang, X., Ren, S., and Sun, J., "Deep residual learning for image recognition", *Proceedings Of The IEEE Computer Society Conference On Computer Vision And Pattern Recognition*, 2016-Decem: 770–778 (2016).
46. Lin, T.-Y., Dollar, P., Girshick, R., He, K., Hariharan, B., and Belongie, S., "Feature Pyramid Networks for Object Detection", *2017 IEEE Conference On Computer Vision And Pattern Recognition (CVPR)*, 936–944 (2017).
47. Ünver, H. M. and Ayan, E., "Skin lesion segmentation in dermoscopic images with combination of yolo and grabcut algorithm", *Diagnostics*, 9 (3): (2019).
48. Redmon, J., Divvala, S., Girshick, R., and Farhadi, A., "You only look once: Unified, real-time object detection", *Proceedings Of The IEEE Computer Society Conference On Computer Vision And Pattern Recognition*, 2016-Decem: 779–788 (2016).
49. Bochkovskiy, A., Wang, C. Y., and Liao, H. Y. M., "YOLOv4: Optimal Speed and Accuracy of Object Detection", *ArXiv*, (2020).
50. Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L., "Microsoft COCO: Common objects in context", *Lecture Notes In Computer Science (Including Subseries Lecture Notes In Artificial Intelligence And Lecture Notes In Bioinformatics)*, 8693 LNCS (PART 5): 740–755 (2014).
51. Internet: Tsang, S.-H., "Brief Review: YOLOv5 for Object Detection | by Sik-Ho Tsang | Medium", <https://sh-tsang.medium.com/brief-review-yolov5-for-object-detection-84cc6c6a0e3a> (2024).
52. Li, C., Li, L., Geng, Y., Jiang, H., Cheng, M., Zhang, B., Ke, Z., Xu, X., and Chu, X., "YOLOv6 v3.0: A Full-Scale Reloading", 1–7 (2023).
53. Internet: Piotr Skalski, "Make Sense", <https://github.com/SkalskiP/make-sense/> .

## ÖZGEÇMİŞ

İdris KAHRAMAN; 2007 yılında Kara Harp Akademisi Sistem Mühendisliđi bölümünden mezun oldu. 2014 yılında ise Karabük Üniversitesi Bilgisayar Mühendisliđi Yüksek Lisans programından mezun oldu. Aynı yıl Karabük Üniversitesi Bilgisayar Mühendisliđi bölümünde doktora eğitime başladı. 2010 yılında Karabük Üniversitesi – Bilgisayar Mühendisliđi bölümünde göreve başladı ve halen devam etmektedir.