

**EVRIŐİMSEL SİNİR AĞLARI KULLANARAK SONDAJ  
KAROT SANDIKLARININ İNCELENMESİ VE TENÖR  
TAHMİNİ**

**INVESTIGATION OF DRILLING CORE TRAYS AND  
GRADE ESTIMATION USING CONVOLUTIONAL  
NEURAL NETWORKS**

**HAYDAR ÇINAR**

**DOÇ. DR YASEMİN KAYHAN ATILGAN**

**Tez Danışmanı**

Hacettepe Üniversitesi

Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliğinin

İstatistik Anabilim Dalı için Öngördüğü

YÜKSEK LİSANS TEZİ olarak hazırlanmıştır.



## ÖZET

### **Evrişimsel Sinir Ağları Kullanarak Sondaj Karot Sandıklarının İncelenmesi ve Tenör Tahmini**

**Haydar ÇINAR**

**Tez Danışmanı: Doç. Dr. Yasemin KAYHAN ATILGAN**

**Haziran 2023, 70 sayfa**

Madencilik şirketleri, 2018 yılı için yaklaşık 10 Milyar USD'lık bir kaynağı yeni madenlerin keşfedilmesine ve araştırılmasına ayırmıştır. Sondaj, yeraltı maden yataklarının araştırılmasında uzun yıllardır kullanılan bir yöntemdir. Sondajlardan elde edilen karotlar kullanılarak kayayı oluşturan minerallerin cinsi, değeri ve miktarı hakkında bilgi edinilmektedir. Madencilik endüstrisinde, kaynak tahmini ve maden analizi, operasyonel verimliliği artırmak, maliyetleri düşürmek ve sürdürülebilirliği sağlamak için kritik öneme sahiptir.

Çalışma, Evrişimsel Sinir Ağları kullanarak sondaj karot sandıklarının incelenmesi ve tenör tahmini üzerine odaklanmaktadır. U-Net modeli kullanılarak, demir madeni kaynaklarının değerlendirilmesi için bir görüntü segmentasyonu ve tenör tahmini uygulaması geliştirilmiştir. Veri kümesi, sondaj karot sandıklarından alınan görüntülerden oluşan bir veri tabanından elde edilmiştir. Eğitim sürecinde; U-Net modelleri, karot numunesi ve numune üzerindeki cevherli bölgenin tespiti için eğitilmiştir. Geliştirilen uygulama, karot görüntüleri üzerinden tenör hesaplamalarını gerçekleştirip çıktı olarak sunmaktadır.

Bu çalışmanın, günümüz veri analiz tekniklerinin madencilik sektöründeki potansiyelini vurgulayarak, kaynak tahmini ve veri analizi alanlarında alternatif bir yöntem sunması hedeflenmektedir. Tezin sonuçlarının, öncelikle ülkemiz madencilik endüstrisi için, değerli bir katkı sağlayacağı ve gelecekteki

arařtırmalara, kaynak tahmini tekniklerinin geliřtirilmesine ve operasyonel verimlilięin artırılmasına ilham vereceęi dūřünölmektedir.

**Anahtar Kelimeler:** Evriřimsel Sinir Aęları, Sondaj, Karot, U-Net, Mineralojik Analiz, Maden

## **ABSTRACT**

# **Investigation Of Drilling Core Trays and Grade Estimation Using Convolutional Neural Networks**

**Haydar ÇINAR**

**Supervisor: Yasemin KAYHAN ATILGAN**

**June 2023, 70 pages**

Mining companies have allocated approximately 10 billion USD for the exploration and investigation of new mines in 2018. Drilling has been a long-standing method for the exploration of underground mineral deposits. By using core samples obtained from drilling, information about the type, value, and quantity of minerals composing the rock can be obtained. In the mining industry, resource estimation and mineral analysis are of critical importance for improving operational efficiency, reducing costs, and ensuring sustainability.

This study focuses on the examination of drilling core trays and ore grade estimation using Convolutional Neural Networks. An image segmentation and ore grade estimation application using the U-Net model has been developed for the evaluation of iron ore resources. The data used in this study were obtained from a database consisting of images extracted from drilling core trays. During the training process, U-Net models were trained to detect core samples and ore-bearing regions on the samples. The developed application performs ore grade calculations based on the core images and presents them as output.

The aim of this study is to highlight the potential of modern data analysis techniques in the mining sector by providing an alternative method in the field of resource estimation and data analysis. It is expected that the findings of this thesis will contribute significantly, primarily to our country's mining industry, and

inspire future research in the development of resource estimation techniques and the enhancement of operational efficiency.

**Keywords:** Convolutional Neural Network, Drilling, Core, U-Net, Mineralogical Analysis, Mining

## TEŐEKKÜR

Tez alıőmam sırasında kıymetli bilgi, birikim ve tecrübeleri ile bana yol gösterici olan, bir an olsun bana olan inancını kaybetmeyen pek deęerli danıőman hocam sayın Do. Dr. Yasemin Kayhan Atılgan'a sonsuz teőekkür, saygı ve minnetlerimi bor bilirim.

Yüksek lisans eęitimim boyunca saęladıkları kolaylık ve kullanmam üzere paylaőtıkları veri için Demir Export A.Ő.'ye teőekkürlerimi sunarım.

Son olarak, lisans hayatımdan bugüne kadar destekleriyle beni hiçbir zaman yalnız bırakmayan pek deęerli eőime teőekkürlerimi sunarım.

# İÇİNDEKİLER

ÖZET .....	i
ABSTRACT .....	iii
TEŞEKKÜR .....	v
İÇİNDEKİLER .....	vi
ŞEKİLLER DİZİNİ .....	viii
KISALTMALAR .....	x
1. GİRİŞ .....	11
2. GÖRÜNTÜ İŞLEME .....	15
2.1. Dijital Görüntü .....	15
2.2. Görüntü İşleme Yöntemleri .....	17
2.3. Görüntü İşleme ve Makine Öğrenmesi .....	19
2.3.1. Öğrenme Temelli Nesne Tanıma .....	19
2.3.2. Görüntü Segmentasyonu .....	19
3. SİNİR AĞLARI .....	21
3.1. İnsan Beyni .....	22
3.2. Yapay Sinir Ağları .....	24
3.2.1. Yapay Sinir Ağları Topolojisi .....	25
3.2.2. Aktivasyon Fonksiyonları .....	26
3.3. Çok Katmanlı Algılayıcı (MLP) Model .....	27
3.3.1. Algılayıcı Model .....	27
3.3.2. Çok Katmanlı Algılayıcı Model .....	28
3.4. Öğrenme Stratejileri .....	29
3.4.1. Denetimli Öğrenme .....	30
3.4.2. Denetimsiz Öğrenme .....	30
3.4.3. Yarı Denetimli Öğrenme .....	31
3.4.4. Takviyeli Öğrenme .....	31
3.5. Öğrenme Hızı .....	32
3.6. Evrimsel Sinir Ağları .....	32



3.6.1. Evrişim Katmanı .....	34
3.6.2. Havuzlama Katmanı .....	37
3.6.3. Tam Bağlı Katman .....	37
3.6.4. Evrişimsel Sinir Ağı Mimarileri.....	38
4. DENEYSEL ÇALIŞMALAR .....	44
4.1. Veri Kümesi .....	44
4.1.1. Görüntülerde Karşılaşılan Zorluklar.....	45
4.1.2. Görüntü Özellikleri.....	47
4.2. Veri İşlemede Kullanılan Programlama Dili ve Ekipman .....	47
4.3. Görüntülerin Etiketlenmesi.....	48
4.4. Görüntülerin Okutulması.....	49
4.5. Model Parametreleri ve Eğitim.....	52
4.6. Modellerin Entegrasyonu ve Uygulama .....	54
5. SONUÇLAR VE TARTIŞMA .....	60
6. KAYNAKLAR.....	63
ÖZGEÇMİŞ .....	<b>Error! Bookmark not defined.</b>

## ŞEKİLLER DİZİNİ

Şekil 2.1. Tek renk kanalına sahip görüntü ve görüntü de bir parçanın matris gösterimi.....	16
Şekil 2.2. RGB renk değerleri gösterimi.....	17
Şekil 2.3. Nesne tanıma örneği (Duan, 2022) .....	19
Şekil 2.4. Görüntü segmentasyonu örneği.....	20
Şekil 3.1. Yapay zeka, makine öğrenmesi ve derin öğrenme arasındaki ilişki...21	
Şekil 3.2. Sinir sisteminin akış şeması.....	23
Şekil 3.3. Nöron hücresinin yapısı(Arbib, 1987) .....	24
Şekil 3.4. Nöron modeli.....	25
Şekil 3.5. Sinir ağlarının topolojisi (Hu vd., 2000) .....	26
Şekil 3.6. Yaygın olarak kullanılan aktivasyon fonksiyonları (S. Khan vd., 2018).....	26
Şekil 3.7. Bir algılayıcı sinir ağı modeli (Hu vd., 2000) .....	28
Şekil 3.8. Çok katmanlı algılayıcı bir model (Shanmuganathan&Samarasinghe, 2016) .....	29
Şekil 3.9. Takviyeli Öğrenme.....	32
Şekil 3.10. Üç büyük yayıncıya göre CNN ile ilgili yapılan çalışmalar (Iosifidis & Tefas, 2022) .....	33
Şekil 3.11. Örnek CNN yapısı (Nelson, 2023) .....	34
Şekil 3.12. Evrişim işleminin çalışma mantığı (Chollet, 2017) .....	35
Şekil 3.13. Dolgu özelliğinin kullanılmasına ilişkin basit bir gösterim.....	36
Şekil 3.14. Dolgu parametresi ve adım parametresi olan evrişim operasyonu.....	36
Şekil 3.15. Maksimum ve ortalama havuzlama (Teoh, 2023) .....	37
Şekil 3.16. Tam bağlı katmanlar.....	38

Şekil 3.17. LeNet-5 ağ Mimarisi (S. Khan vd., 2018) .....	39
Şekil 3.18. AlexNet ağ Mimarisi (Jiang vd., 2021) .....	39
Şekil 3.19. VGGNet ağ mimarisi (Alzubaidi vd., 2021) .....	40
Şekil 3.20. GoogleNet ağ mimarisi (Jiang vd., 2021) .....	41
Şekil 3.21. U-Net ağ mimarisi (32x32) (Ronneberger vd., 2015) .....	42
Şekil 4.1. Örnek karot sandığı fotoğrafları.....	44
Şekil 4.2. Eğitim ve test veri kümeleri.....	45
Şekil 4.3. Gölge faktörünün önemli olduğu görüntü örneği.....	45
Şekil 4.4. Birden fazla karot sandığının olduğu görüntü örneği.....	46
Şekil 4.5. Açık ortamda alınan görüntü örneği.....	46
Şekil 4.6. Veri bilimi uygulamalarında kullanılan programlama dilleri (Mooney, 2022) .....	47
Şekil 4.7. Görüntü etiketleme aşamaları.....	49
Şekil 4.8. Görüntü örnek tensör gösterimi.....	50
Şekil 4.9. Karot sandığı içerisindeki numunenin tespiti için etiketlenmiş görüntü örneği.....	51
Şekil 4.10. Numunedeki cevherli bölgenin tespiti için etiketlenmiş görüntü örneği.....	51
Şekil 4.11. Döngü sayısına göre doğruluk oranının değişimi.....	53
Şekil 4.12. Döngü sayısına göre doğruluk oranının değişimi.....	54
Şekil 4.13. Geliştirilen modeller arası ilişki.....	55
Şekil 4.14. Uygulama çıktısı örneği.....	56
Şekil 4.15. Uygulamanın gerçek zamanlı çıktı örnekleri.....	58
Şekil 4.16. Geliştirilen uygulamanın saha çalışmalarına ait bir örnek.....	59

## KISALTMALAR

### Kisaltmalar

CNN	Evrişimsel Sinir Ağları
RGB	Kırmızı Yeşil Mavi
FC	Tam Bağlı Katman
MLP	Çok Katmanlı Algılayıcı

# 1. GİRİŞ

Madencilik şirketleri, gelirlerinin azımsanmayacak bir kısmını (2018 yılı için yaklaşık 10 Milyar USD) yeni madenlerin keşfedilmesine ve araştırılmasına ayırmaktadır (S&P Global Market Intelligence, 2019). Sondaj, yeraltı maden yataklarının araştırılmasında uzun yıllardır kullanılan bir yöntemdir. Numune almak amacıyla sondaj makineleri, karot olarak bilinen silindirik kaya örneklerini yüzeyi keserek çıkarır. Sondaj karotları, potansiyel cevher birikimlerini karakterize etmek için değerli jeolojik bilgiler sağlayan, kaya örnekleri olarak kabul edilir, tüm kısa ve uzun vadeli planlar(yatırım, üretim, satış dahil) bu veriye göre gerçekleştirilir (Gandhi & Sarkar, 2016).

Bir maden yatağı, ancak yataktaki minerallerin doğası ve dağılımı hakkında ayrıntılı bilgi varsa, ekonomik ve verimli bir şekilde kullanılabilir (Gottlieb, 2008). Bu da alınan numunelerin mineralojik analizleri ile mümkündür. Mineralojik analiz, temel olarak, kayayı oluşturan minerallerin cinsini, yapısını analiz ederek, minerallerin doğru sınıflandırılmasını sağlar. Mineralojik analiz ile maden ve maden yataklarının oluşum ve çıkış şekilleri belirlenerek, madenin değerinin belirlenmesine yönelik temel veri elde edilebilir (Ro & Kim, 2021).

Sondaj karotları, sondaj sırasında, sondaj kuyularından çıkarılır ve mineral ölçümleri için gerekli jeolojik ve jeoteknik bilgileri sağlar.

Sondaj karot sandıkları tipik olarak madencilik endüstrisinde sondaj işlemleri sırasında yeraltından çıkarılan silindirik kaya örneklerini veya karotları saklamak için kullanılır. Karot küçük parçalara ayrılarak sandıklarda saklanır. Son zamanlarda karotlar dijital karot kütüphanelerinde fotoğraflanarak arşivlenmektedir. Bunun çeşitli avantajları vardır. Örneğin, kayıp, hasar, depolama maliyeti gibi sorunları azaltarak, bilgiye anında uzaktan erişime olanak sağlar. Karotlar, bölgenin jeolojisi hakkında değerli bilgiler içerir ve maden yataklarının kalitesini ve miktarını tahmin etmek için kullanılabilir. Modern karot tarayıcıları, otomatik olarak yüksek çözünürlüklü karot tepsisi görüntülerini işleyerek, jeolojik yapılar gibi önemli kaya özelliklerini çıkarsa da mevcut karot görüntü analiz yöntemleri daha eski algoritmalar ile çalıştığı için, sonuçların genelleme performansı düşüktür ve istatistiksel açıdan sağlam değildir.

Sondaj karotlarının ilk analizi, alan uzmanları tarafından (jeologlar) göz ile yapılır ve kaydedilir. Yapılan bu işleme “Log” adı verilir. Bu işlem ile alınan numunenin tipi, dokusu gibi karakteristikler belirlenmektedir. Ancak bu çalışma yöntemi kişilerin deneyimlerine bağlı olduğu için, elde edilen sonuçlar yanlı olabilmektedir. Bunun yanı sıra bu işlem fazla zaman alabilmektedir. Sondaj maliyeti ve önemi göz önüne alındığında tek başına yeterli olmadığı sonucuna ulaşılmaktadır (Acosta vd., 2020).

Alan uzmanları tarafından göz ile yapılan işlemin ardından, çeşitli yöntemler ile numune analizleri desteklenir. Kimyasal analizler yardımıyla, mineralojik karakterizasyon testi (Bradshaw, 2014), mineral işleme testi (Bueno vd., 2015) gibi analizler uygulanır. Fakat uygulanan bu analizler hem zaman hem de iş gücü açısından çok maliyetlidir.

Kimyasal analizlerin yanı sıra, numunelerin detaylı görüntüleri üzerinden analiz çalışmaları da yapılmaktadır. Optik mikroskop ile analiz (Hapugoda vd., 2015), X ışını kırınımı kullanılarak yapılan analiz (Halder, 2013), X ışını floresansı kullanarak yapılan analiz (Nikonow & Rammlmair, 2017), taramalı elektron mikroskobu (SEM) kullanarak yapılan analiz (Fandrich vd., 2007) ve plazma kütle spektrometrisi (Layton-Matthews vd., 2014) yöntemi kullanılan tekniklerin örnekleridir. Bu analizler, diğerlerinin (göz ile tarama ve kimyasal analiz) yanı sıra model mineralojisi, mineralojik doku (nicelleştirilmiş mineraloji) ve örneklerin jeokimyası (element konsantrasyonları) hakkında bilgi sağlar (Acosta vd., 2020). Ancak bahsedilen bu destekleyici yöntemler de kullanılan makine ve ekipmanlar bakımından çok maliyetlidir.

Kullanılan yöntemler maliyetli ve analiz süreleri uzun olduğu için, bu yöntemlere alternatif daha hızlı sonuç veren ve daha az maliyetli yaklaşımlar araştırılmaya başlanmıştır. Bu alternatif yöntemlerden birisi, yüksek çözünürlüklü kameralar yardımı ile toplanan görüntülerin işlenmesi ve elde edilen verinin analiz edilmesi esasına dayanır. Mevcut yöntemlerin çoğu log verisini kullanmaya odaklanırken, karot görüntülerinin derin analizi, genellikle yeraltı oluşumları hakkında daha güvenilir bilgiler sağlamaktadır. Sondaj işleminin maliyeti göz önüne alındığında, bunların değerli ve sınırlı bir kaynak olması sebebiyle, titizlikle incelenmesi gerekmektedir. Yeni teknolojilerin gelişmesi, büyük veri ve yapay zekanın evrimi ile birlikte karot görüntülerinin makine öğrenme yöntemleri gibi düşük maliyetli

analiz yöntemleri ile incelenmesi doğru, hızlı ve zahmetsiz çözümler sunacaktır. Görüntü işleme konusu büyük veri, derin öğrenme, makine öğrenmesi ve yapay zeka alanları ile doğrudan ilişkilidir.

Gelişen dijital ortam, çevrimiçi faaliyetler ve hizmetler neticesinde büyük miktarda dijital bilgiyi de içeren büyük veri kavramı, veriye dayalı sistemlerin yönetilmesi ve geliştirilmesinde önemli bir role sahip olması sebebiyle günümüzde tıp, işletme, pazarlama, ekonomi vb. birçok alanda yer almaktadır. Burada, geleneksel veri analiz yöntemlerinden farklı olarak, yüksek hacimli veriyi hızlı işleyerek analiz edebilen makine öğrenme yöntemleri devreye girmektedir.

Görüntü işleme alanında çok tercih edilen bir yöntem, makine öğrenme konusunun altında yer alan derin öğrenme ve dolayısıyla yapay zekadır.

Yapay zeka kavramı Nilsson'ın, *The Quest for Artificial Intelligence* isimli kitabında "Benim için yapay zeka, makineleri akıllı hale getirmeye yönelik faaliyettir. Zeka ise bir varlığın çevresi ile uygun ve öngörülü bir şekilde çalışmasını sağlayan kalitedir" olarak özetlenmiştir. Fogel ise *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence* isimli kitabında yapay zekayı "Bir sistemin çeşitli ortamlarda ve koşullarda hedefler ile buluşmak için davranışları uyarlama yeteneğidir" olarak tanımlamaktadır. Yapay zeka kavramını açıklamak amacıyla çeşitli fikir ve tanımlamalar olsa da, bu kavramın net olarak bir tanımı için henüz literatürde hemfikir olunamamıştır.

Yapay zeka kavramı, Endüstri 4.0 devriminin yıkıcı etkileriyle beraber hemen her sektörde kendisine yer edinmeye başlamıştır. Bu mecburi adaptasyonun da etkisiyle beraber, yapay zeka çalışmaları hızlanmış ve bu alanlara yapılan yatırımlar artırılmıştır. Özellikle Savunma sanayii içerisinde bu kültür geçişi daha hızlı bir şekilde gerçekleşmektedir. Öyle ki Deloitte'un yayınlamış olduğu "The Age of With™- The AI Advantage in Defence and Security" isimli makalesinde savunma sanayiinde; "Tespit etme", "Planlama", "Saha Operasyonları" gibi tüm süreçlerde yapay zeka teknolojilerinin kullanıldığından söz edilmiş ve örnekler verilmiştir. Savunma sanayisinin yanı sıra üretim sektöründe de yapay zekanın etkin kullanımına örnekler verilebilir: Lazer tarama ve görüntüleme cihazlarının yapay zeka ile entegrasyonu ile birlikte üretim bantlarından geçen ürünlerdeki

deformasyonların eş zamanlı olarak tespit ve müdahalesi gerçekleştirilebilmektedir (Javaid vd., 2022).

Diğer sektörler ile karşılaştırıldığında madencilik endüstrisinde büyük veri, yapay zeka, derin öğrenme gibi konular hala geliştirilmeye açıktır. Sektöre ilişkin mevcut problemlerin çözümü, genellikle loglar veya sensörlerden toplanan veriler kullanılarak, klasik makine öğrenme yöntemleri ile yapılmaktadır. Oysa modern makine öğrenme yöntemleri, yapay zeka, ve otonom teknolojiler bu sektörde ekonomik açıdan birçok fayda sağlamaktadır (Hyder vd., 2019). Bunlar maliyetlerin düşürülmesi, verimlilik ve verimliliğin artırılması, işçilerin tehlikeli koşullara maruz kalmasının azaltılması vb. biçiminde verilebilir (Hyder vd., 2019).

Bu tez çalışmasında sondaj karotlarının incelenmesi için Evrişimsel Sinir Ağları (CNN, Convolutional Neural Network) kullanılması planlanmıştır. CNN'ler, görüntülerdeki desenleri tanıyabilen, görüntüleri segmente edebilen ve bunları özelliklerine göre sınıflandırabilen bir tür derin öğrenme algoritmasıdır. CNN'lerin karot görüntülerinden ve ilgili sınıf tahminlerinden oluşan büyük bir veri kümesi üzerinde eğitilerek, görsel özellikleri ile cevher özellikleri arasındaki ilişkiyi öğrenmesi, eğitilen model kullanarak da yeni karot numunelerinin tenör tahminlerinin doğru sınıflandırılması hedeflenmektedir. Böylece mevcut analiz yöntemlerine kıyasla zaman ve maliyet açısından etkili bir azaltma yaratılarak, maden arama ve üretim süreçlerinin verimliliğini, kısa ve uzun vadeli planların doğruluğunu artırmak istenmektedir.



## 2. GÖRÜNTÜ İŞLEME

Günümüzde dijital görüntü işleme, bir dijital kamera ile çekilen veya bir tarayıcı tarafından taranan görüntüleri temsil eden dijital sinyallerin işlenmesiyle ilgilenen bir bilgisayar bilimi dalıdır (Cosido vd., 2014).

### 2.1. Dijital Görüntü

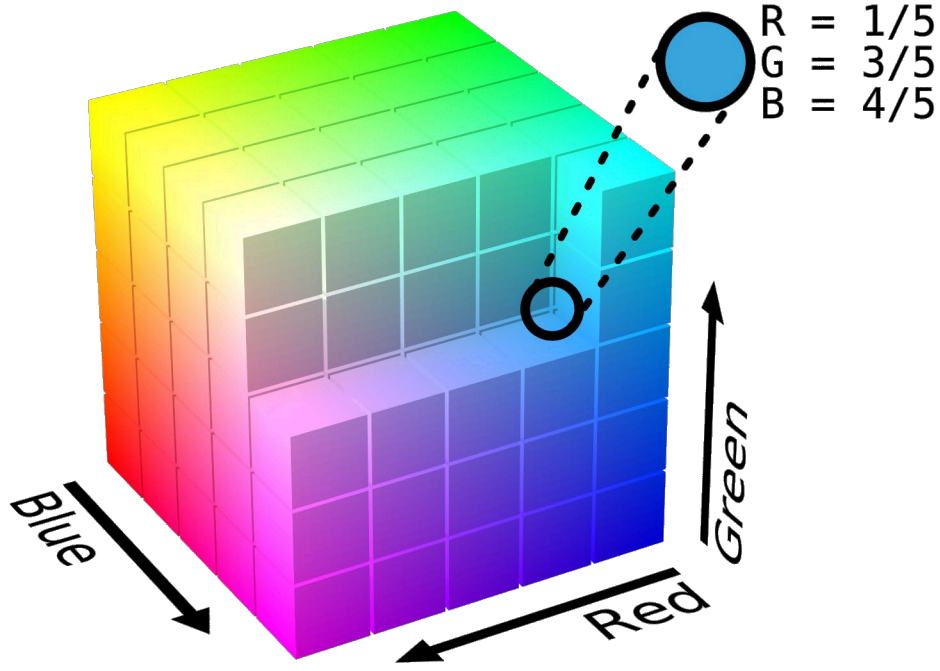
Dijital görüntüler, tıbbi görüntüleme, video izleme, adli soruşturmalar, uzaktan algılama vb. gibi çeşitli uygulamalarda kullanılmaktadır. Görüntülerin, çeşitli görüntü işleme teknikleriyle otomatik olarak analiz edilmesi günümüzde önem kazanmıştır.

Görüntü, bir nesnenin, bir kişinin veya bir sahnenin görsel bir temsilidir. Dijital bir görüntü, sürekli olan iki  $x$  ve  $y$  değişkeninin  $f(x, y)$  fonksiyonu olarak temsil edilir (Tyagi, 2018a). Dijital olarak işlenebilmesi için görüntünün taranması ve bir sayı matrisine dönüştürülmesi gerekmektedir. Matematiksel olarak, bir dijital görüntü, genellikle piksel(pixel) olarak adlandırılan, sonlu sayıda nokta kullanan, iki boyutlu bir görüntünün matris temsilidir. Her piksel, aynı zamanda matris ögesi, sayısal değerlerle temsil edilir: gri tonlamalı görüntüler için, pikselin yoğunluğunu temsil eden tek bir değer yeterlidir (Dastres & Soori, 2021). Şekil 2.1'de tek renk kanallı görüntü örneği verilmiştir.



Şekil 2.1. Tek renk kanalına sahip görüntü ve görüntü de bir parçanın matris gösterimi.

Renkli görüntüler için üç değer (RGB) saklanmaktadır (Tyagi, 2018a). Bu üç değer kırmızı (R, red), yeşil (G, green) ve mavi (B, blue) miktarını temsil etmektedir. Bu renk değerleri genellikle  $[0, 255]$  veya  $[0, 1]$  aralığında değer almaktadır. Renkli görüntü, bu üç renk değerinin birleşimidir. Şekil 2.2'de örnek renk değerleri gösterilmiştir.



Şekil 2.2. RGB renk değerleri gösterimi.

## 2.2. Görüntü İşleme Yöntemleri

Dijital görüntü işleme, uydu görüntü analizinden, mikroskopik boyut kontrolüne kadar çeşitli uygulamalar için kullanılmaktadır (Cosido vd., 2014). Görüntü işleme yöntemlerinde kullanılan modeller, görüntüyü hem girdi olarak almakta hem de çıktı olarak vermektedir. Birçok görüntü işleme yöntemi vardır. Yöntemin seçimi uygulamaya özeldir.

Görüntü işleme yöntemleri aşağıdaki gibi gruplanabilir (da Silva & Mendonca, 2005; Gonzalez & Woods, 2008; Sonka vd., 2007; Tyagi, 2018b);

- Parlaklık ve Kontrast Ayarlama: Bu yöntem, görüntünün parlaklık ve kontrast değerlerini değiştirerek daha iyi görünürlük elde etmek için kullanılmaktadır. Parlaklık ayarlama; görüntünün genel ışık seviyesini değiştirirken, kontrast ayarlama; görüntüdeki farklılık veya renk dağılımını vurgulamaktadır. Bu yöntemler, görsel sunumu iyileştirerek, görüntünün daha anlaşılır olmasını sağlamaktadır.
- Filtreleme: Görüntüye farklı filtreler uygulayarak gürültüyü azaltmak, kenarları belirginleştirmek veya görüntüyü yumuşatmak gibi işlemleri gerçekleştirmektedir. Yaygın kullanılan filtreleme yöntemleri arasında

ortalama filtre, medyan filtre, Gauss filtresi ve kenar tespiti için Laplacian filtresi bulunmaktadır. Bu filtreler, görüntüdeki istenmeyen gürültüyü azaltır ve önemli özellikleri vurgular.

- **Kenar Tespiti:** Görüntüdeki nesnelerin kenarlarını tespit etmek için kullanılan bir yöntemdir. Kenar tespiti algoritmaları, nesnelerin sınırlarını belirlemek için farklı yöntemler kullanmaktadır. Örneğin, Sobel operatörü ve Canny kenar tespiti, görüntüdeki yoğunluk değişikliklerini belirleyerek kenarları tespit etmektedir. Bu yöntemler, nesnelerin sınırlarını belirlemek ve nesne tabanlı analiz yapmak için önemlidir.
- **Segmentasyon:** Görüntüyü farklı bölgelere veya nesnelere ayırmak için kullanılan bir yöntemdir. Segmentasyon yöntemleri, nesnelere ayırt etmek, nesne tabanlı analiz yapmak veya görüntüyü farklı parçalara bölmek için kullanılmaktadır. Klasik segmentasyon yöntemleri iki gruba ayrılabilir. Renk tabanlı segmentasyon, görüntüdeki piksellerin renk değerlerine dayalı olarak bölgelere ayrılmasını sağlamaktadır. Eşikleme, bir pikselin belirli bir eşik değerine göre iki farklı bölgeye ayrılmasını sağlamaktadır.
- **Öznitelik Çıkarımı:** Görüntüdeki önemli özellikleri tanımlama ve çıkarma işlemi için kullanılan bir yöntemdir. Özellik çıkarımı, görüntülerdeki desenler, kenarlar, renk histogramları gibi özellikleri hesaplayarak nesnelere tanımlamak veya sınıflandırmak için kullanılmaktadır. Bu yöntemler, görüntüleri özelliklerine göre analiz etmek ve nesnelere tanıma işlemlerine alt yapı sağlamak için kullanılmaktadır.
- **Nesne Tanıma ve Sınıflandırma:** Nesne tanıma ve sınıflandırma, görüntüdeki nesnelere tanıma ve belirli sınıflara atama yapma işlemleri için kullanılan yöntemlerdir. Makine öğrenmesi ve derin öğrenme algoritmaları kullanılarak eğitilen modeller, görüntüdeki nesnelere tanımak, sınıflandırmak ve segmente etmek için kullanılmaktadır. Örneğin, CNN nesne tanıma ve sınıflandırmada yaygın olarak kullanılan bir derin öğrenme modelidir. Bu yöntemler, görüntülerdeki nesnelere otomatik olarak tanımak ve analiz etmek için kullanılmaktadır.
- **Görüntü Hızlandırma ve Süper Çözünürlük:** Bu yöntemler, düşük kaliteli veya bulanık görüntüleri iyileştirmek veya yüksek çözünürlüklü görüntüler elde etmek için kullanılmaktadır. Görüntü hızlandırma yöntemleri,

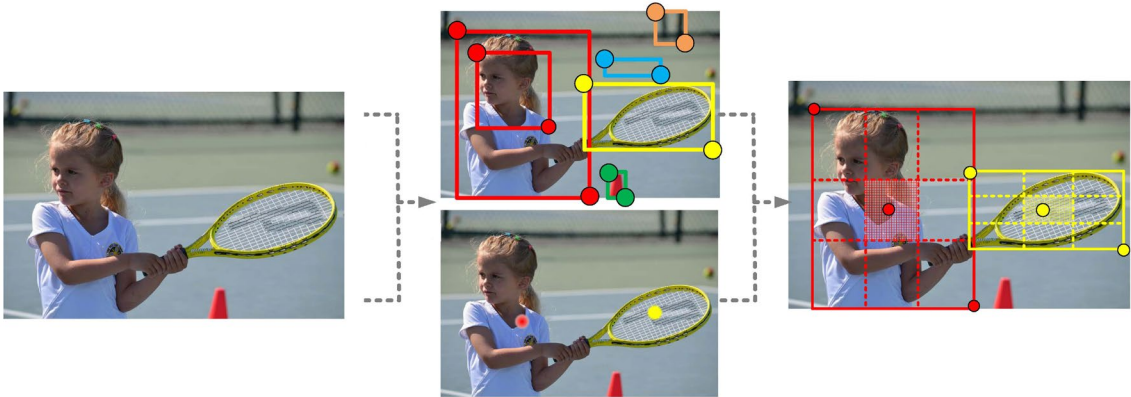
görüntüleri yeniden örnekleme, interpolasyon veya tahminleme yoluyla iyileştirir. Süper çözünürlük yöntemleri, düşük çözünürlüklü görüntülerden daha yüksek çözünürlüklü görüntüler oluşturmayı amaçlar. Bu yöntemler, görüntü kalitesini iyileştirmek ve daha ayrıntılı analizler yapmak için kullanılmaktadır.

### 2.3. Görüntü İşleme ve Makine Öğrenmesi

Günümüzde, görüntü işleme alanında hızla gelişen bir yaklaşım olan makine öğrenmesi, görüntü analizinde büyük bir etkiye sahiptir. Makine öğrenmesi, algoritmalara dayalı olarak veri analizi ve görüntü tanıma yeteneği kazanabilen bir yapay zeka dalıdır. Görüntü işleme için makine öğrenmesi yöntemleri, görüntülerdeki desenleri, nesnelere tanıma ve sınıflandırmayı, segmentasyonu ve diğer birçok işlemi gerçekleştirmek için kullanılmaktadır.

#### 2.3.1 Öğrenme Temelli Nesne Tanıma

Makine öğrenmesi algoritmaları, görüntülerdeki nesnelere tanıma ve sınıflandırmak için kullanılmaktadır. Önceden eğitilmiş bir model kullanılarak veya verilerin üzerine kurulan özel bir model oluşturularak nesne tanıma gerçekleştirilmektedir. Derin öğrenme modelleri, görüntü işleme için güçlü bir araçtır, nesne tanıma ve sınıflandırmada büyük başarılar elde etmektedir. Şekil 2.3'te tenis raketi ve insan tespiti yapılmış bir nesne tanıma örneği verilmiştir.



Şekil 2.3. Nesne tanıma örneği (Duan, 2022).

#### 2.3.2. Görüntü Segmentasyonu

Görüntü segmentasyonu, bir görüntüyü farklı bölgelere veya nesnelere ayırmayı amaçlayan bir işlemdir. Makine öğrenmesi yöntemleri, pikselleri veya görüntü

bölgelerini benzerlik ölçütlerine göre gruplandırmak ve böylece farklı nesne bölgelerini ayırt etmek için kullanılmaktadır. Şekil 2.4'te karot sandığı içerisindeki numunenin ayırt edilmesine ilişkin görüntü segmentasyonu örneği verilmiştir.



Şekil 2.4. Görüntü segmentasyonu örneği.

Görüntü segmentasyonunda yaygın olarak kullanılan algoritmalar derin öğrenme algoritmalarıdır. Derin öğrenme algoritmalarının en iyi örneklerinden biri sinir ağlarıdır. Sinir ağlarının ise görüntü işleme problemleri için ideal olarak kullanılabilen ve çeşitli avantajlara sahip özel bir türü Evrimsel sinir ağlarıdır. Tez çalışmasında, CNN ile görüntü segmentasyonu üzerine çalışmalar yapılacaktır. Bir sonraki bölümde sinir ağları ve CNN detaylı açıklanacaktır.

### 3. SİNİR AĞLARI

Yapay zeka alanında yapılan çalışmalar incelendiğinde temelde “Makine Öğrenmesi”, “Derin Öğrenme” olmak üzere iki alt başlığın geniş yer aldığı gözlemlenmektedir. Şekil 3.1’de yapay zeka, makine öğrenmesi ve derin öğrenme arasındaki ilişki verilmiştir.



Şekil 3.1. Yapay zeka, makine öğrenmesi ve derin öğrenme arasındaki ilişki.

Makine öğrenmesi, hesapların(işlemlerin) performansını artırmak veya doğru tahminler yapmak için, deneyimi kullanan hesaplama yöntemleri olarak tanımlanabilir (Mohri vd., 2012). Burada deneyim, toplanan ve analiz için hazır hale getirilen, öğrencilere sunulan geçmiş bilgileri ifade eder. Bu veri, dijitalleştirilmiş insan etiketli eğitim kümeleri veya çevre ile etkileşim yoluyla elde edilen diğer bilgi türleri şeklinde olabilir. Her durumda, kalitesi ve boyutu, öğrenen tarafından yapılan tahminlerin başarısı için çok önemlidir (Mohri vd., 2012).

Derin öğrenme ise, veri odaklı olarak tutarlı kararlar verebilen büyük sinir ağı modelleri oluşturmaya odaklanan yapay zekanın bir alt dalıdır (Kelleher, 2019). Günümüzde pazar lideri birçok teknoloji devi çeşitli derin öğrenme modelleri kullanmaktadır. Örneğin, Microsoft, Google, Yandex resim arama ve diller arası çeviri gibi işlemler için derin öğrenme modelleri kullanmaktadır. Günümüz akıllı telefonlarında konuşma tanıma, yüz tanıma, ifade tespit etme gibi sistemler için derin öğrenme modelleri sıklıkla kullanılmaktadır. Tesla şirketi gibi dijitalleşmenin öncüsü olan otomotiv şirketlerinde de otonom sürüş gerçekleştiren araçlarda yer tespit etme, rota planlama gibi birçok konuda derin öğrenme modelleri

kullanılmaktadır. Derin öğrenme modelleri dijital oyunlarda da kullanılmakta ve kullanıcının davranışlarına göre oyunu şekillendiren oyun senaryoları geliştirilmektedir.

Yapay sinir ağları, yaygın olarak kısaca “sinir ağları” olarak adlandırılmaktadır. İnsan beyninin çalışma prensibinin, bilgisayarlarda kullanılan hesaplama mantığından tamamen farklı olduğunun anlaşılmasıyla oluşturulmuştur. Nöronlar olarak bilinen beynin yapısal bileşenleri, belirli hesaplamaları (örn., örüntü tanıma, algılama ve motor kontrol) bugün var olan en hızlı dijital bilgisayardan çok daha hızlı bir şekilde gerçekleştirebilmektedir (Haykin, 2009a). Haykin, sinir ağlarının sadece insan beyni değil, diğer tüm canlılarda da ne kadar hızlı karar verebildiğini aşağıdaki cümleleri ile açıklamıştır.

*“Bir yarasanın sonarını düşünün. Sonar, aktif bir ekolokasyon sistemidir. Bir hedefin (örneğin, uçan bir böcek) ne kadar uzakta olduğu hakkında bilgi sağlamanın yanı sıra, yarasa sonarı hedefin hızı, hedefin boyutu, hedefin çeşitli özelliklerinin boyutu hakkında bilgi verir. Tüm bu bilgileri hedef yankıdan çıkarmak için gereken karmaşık sinirsel hesaplamalar, bir erik büyüklüğündeki beyinde meydana gelir. Gerçekten de bir yankı bulucu yarasa, bir radar veya sonar mühendisinin kiskanacağı kolaylık ve başarı oranı ile hedefini takip edebilir ve yakalayabilir.”*

Aleksander ve Morton sinir ağlarını; gözlemsel bilgiyi depolamak ve kullanıma hazır hale getirmek için doğal bir eğilime sahip olan, büyük ölçüde paralel olarak işlenen, basit işlem birimlerinden oluşan bir işlemci olarak tanımlamaktadır. İki yönden beyine benzediği söylenebilir;

Bilgi, ağ tarafından öğrenme süreci sonunda edinilir.

Sinaptik ağlar olarak bilinen internöron bağlantı, edinilen bilgiyi depolamak için kullanılır.

Yapay sinir ağları yönteminin anlaşılabilirliği için, yöntemin örnek aldığı, insan beyninin işleyişinden de bahsedilmesi gerekmektedir.

### **3.1. İnsan Beyni**

Ünlü sinir bilimci Arbib'e göre insan sinir ağı yapısı Şekil 3.2'deki gibi aşamalı olarak gösterilebilir (Arbib, 1987).



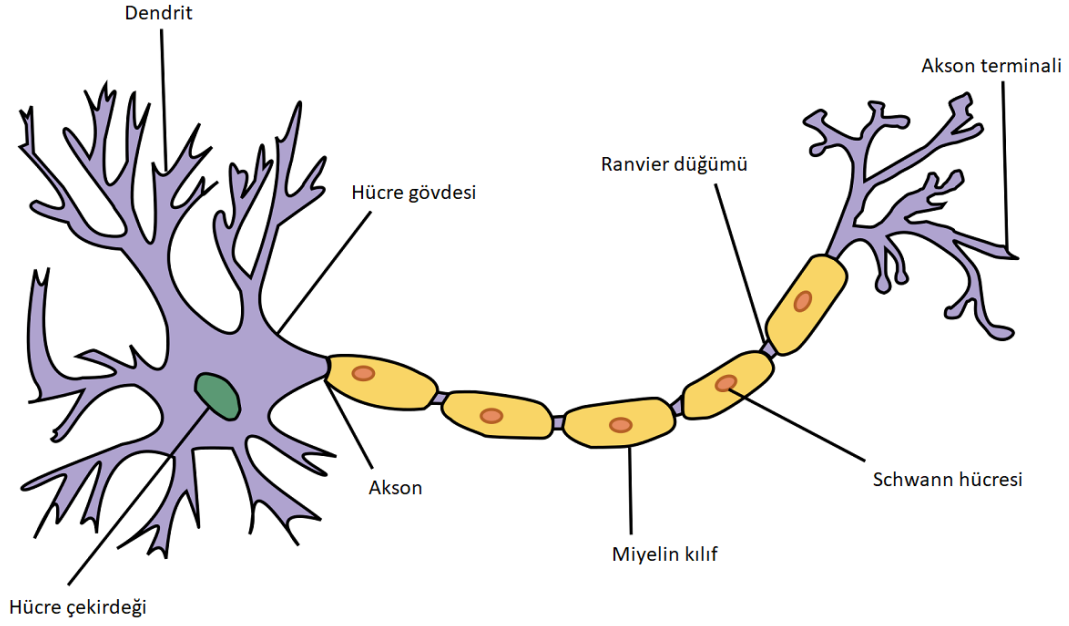


Şekil 3.2. Sinir sisteminin akış şeması.

Sistemin merkezinde, sürekli olarak bilgi alan, onu algılayan ve uygun kararlar veren nöral (sinir) ağı ile temsil edilen beyin bulunur. Sağdan sola doğru işaret eden oklar, sistemdeki geri bildirimleri ifade etmektedir. Reseptörler, insan vücudundan veya dış ortamdan gelen uyarıları, sinir ağına elektriksel sinyaller olarak iletir. Efektörler, sinir ağı tarafından üretilen elektriksel sinyalleri, sistem çıktıları olarak fark edilebilir tepkilere dönüştürür.

Beyni anlama mücadelesi, nöronlar fikrini beyin yapısal bileşenleri olarak ortaya koyan Ramón y Cajál'ın 1911 yılındaki öncü çalışması sayesinde kolaylaştırılmıştır (Haykin, 2009a). Tipik olarak, nöronlarda sinirsel olaylar milisaniye aralığında gerçekleşir. Bununla birlikte, beyin, gerçekten şaşırtıcı sayıda nörona (sinir hücreleri) ve bağlantılara sahiptir. İnsan korteksinde yaklaşık 10 milyar nöron ve 60 trilyon sinaps veya bağlantı olduğu tahmin edilmektedir (Shepherd, 1990).

Şekil 3.3'de nöronların en genel yapısı verilmiştir. Buna göre nöronların temel bileşenleri dendrit, hücre gövdesi, akson ve akson terminalleri olarak sıralanabilir. Dendritler, yakınlarındaki diğer nöronlardan gelen uyarıları kabul eden ve hücre gövdesine ileten dallanmış yapılardır. Hücre gövdesi *soma* olarak da adlandırılmaktadır. Hücre gövdesi mitokondri, golgi, çekirdek, endoplazmik retikulum gibi çeşitli organellere sahiptir. Akson, hücre gövdesinden akson terminaline doğru uyarıların iletilmesine yardımcı olan tüp benzeri yapılardır. Akson terminalleri, uyarıların iletilmesinde terminal görevini gören yapılardır (Arbib, 1987).



Şekil 3.3. Nöron hücresinin yapısı(Arbib, 1987).

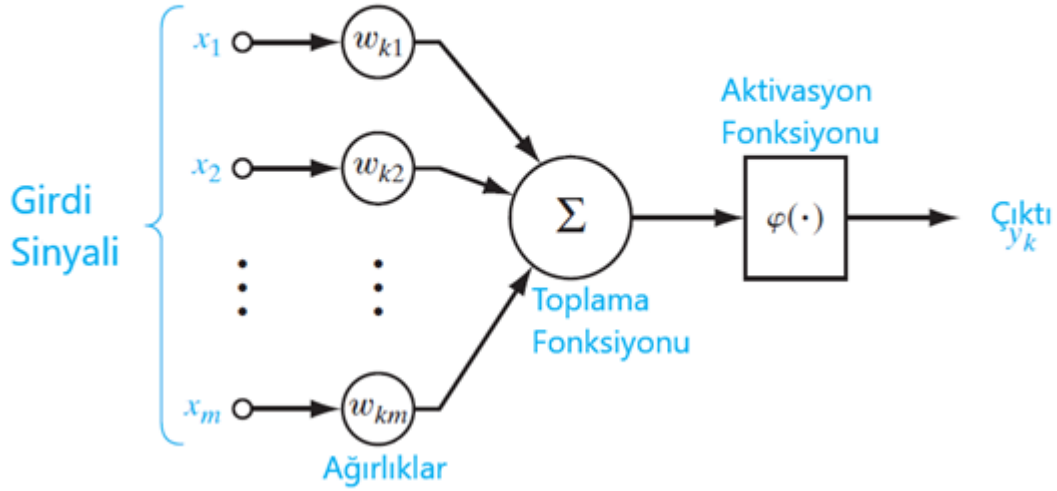
### 3.2. Yapay Sinir Ağları

Nöronlar, bir sinir ağının işleyişi için temel olan bilgileri işleyen birimlerdir. Şekil 3.4'deki diyagram, sinir ağları ailesinin tasarlanması için temel oluşturan bir nöron modelini göstermektedir (Haykin, 2009b). Bir nöral modelin Sinaptik ağırlık, Toplama fonksiyonu ve Aktivasyon fonksiyonu olmak üzere üç temel ögesi vardır.

Sinaptik ağırlık, her biri kendi ağırlığı veya gücü ile karakterize edilen bir dizi sinaps veya bağlantı halkalarıdır. Spesifik olarak,  $k$  nöronuna bağlı  $j$  sinapsının girişindeki bir  $x_j$  sinyali, sinaptik ağırlık  $w_{kj}$  ile çarpılır.  $w_{kj}$  sinaptik ağırlığının alt indislerinin nasıl yazıldığına dikkat etmek önemlidir.  $w_{kj}$ 'deki ilk alt indis, söz konusu nörona atıfta bulunur ve ikinci alt indis ise sinapsın giriş ucuna karşılık gelir. Beyindeki bir sinapsın ağırlığından farklı olarak, yapay bir nöronun sinaptik ağırlığı, pozitif değerlerin yanı sıra negatif değerleri de içeren bir aralıkta olabilir (Haykin, 2009b).

Toplama fonksiyonu nöronun ilgili sinaptik güçleri tarafından ağırlıklandırılmış, giriş sinyallerini toplamak için bir toplayıcıdır. Toplama fonksiyonu doğrusal bir birleştiricidir (Haykin, 2009b).

Aktivasyon fonksiyonu bir nöronun çıkışını sınırlamak için kullanılan bir fonksiyondur. Aktivasyon fonksiyonu, çıkış sinyalinin izin verilen aralığını sonlu bir değere sıkıştırdığı (sınırladığı) için ezme fonksiyonu (squashing function) olarak da adlandırılır. Genellikle bir nöronun çıktı değeri  $[0,1]$  veya  $[-1,1]$  aralığındadır (Haykin, 2009b).



Şekil 3.4. Nöron modeli.

### 3.2.1 Yapay Sinir Ağları Topolojisi

Bir sinir ağında, birden çok nöron, dağıtılmış hesaplamayı kolaylaştırmak için bir ağ oluşturmak üzere birbirine bağlanır. Ara bağlantıların konfigürasyonu, bir grafikte açıklanabilir. Yönlendirilmiş bir grafik, düğümlerden (sinir ağlarındaki nöronlar ve ayrıca girdiler) ve yönlendirilmiş oklardan (sinir ağlarındaki sinaptik bağlantılar) oluşur. Grafiğin topolojisi döngüsel olmayan (acyclic) veya döngüsel (cyclic) olarak kategorize edilebilir. Şekil 3.5 (a)'da gösterildiği gibi döngüsel olmayan topolojiye sahip bir sinir ağı, geri besleme döngüsü içermez. Böyle döngüsel olmayan bir sinir ağı genellikle girdileri ve çıktıları arasında doğrusal olmayan bir modeli tahmin etmek için kullanılır. Şekil 3.5 (b)'de gösterildiği gibi, döngüsel topolojiye sahip bir sinir ağı, geri yönlendirilmiş oklardan oluşan en az bir döngü içerir. Geri besleme döngüsü nedeniyle, tekrarlayan bir ağ, dahili bellek içeren doğrusal olmayan dinamik bir sistem modeline dönüşür. Tekrarlayan sinir

ağları genellikle karmaşık davranışlar sergilemekte ve yapay sinir ağları alanında aktif bir araştırma konusu olmaya devam etmektedir (Hu vd., 2000).

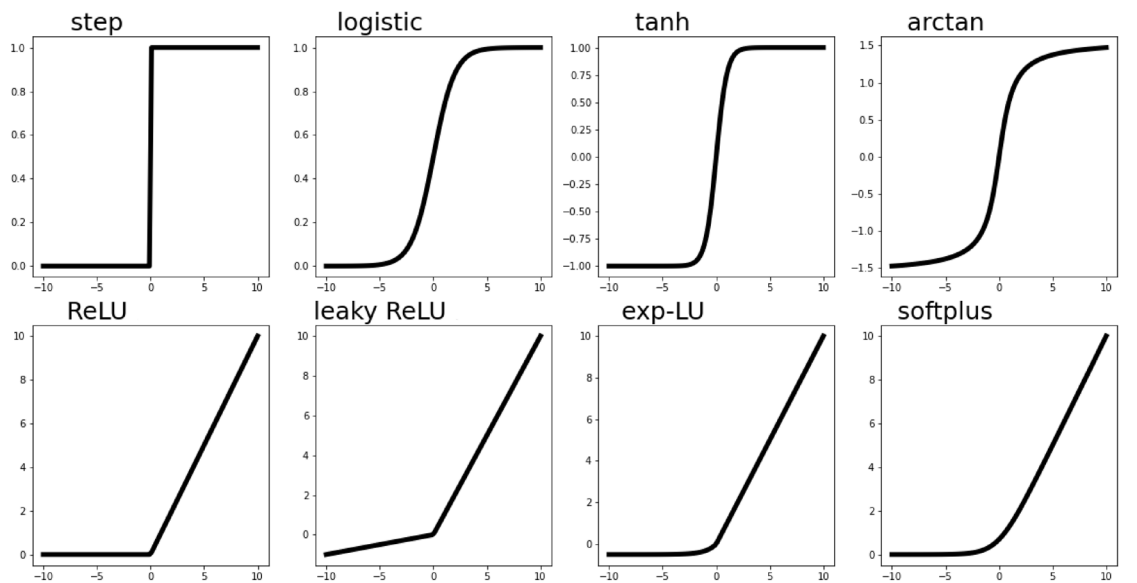


Şekil 3.5. Sinir ağlarının topolojisi (Hu vd., 2000).

### 3.2.2. Aktivasyon Fonksiyonları

Aktivasyon fonksiyonları, bir sinir ağı düğümünün veya nöronunun çıkışına uygulanan matematiksel fonksiyonlardır ve bu fonksiyonlar sayesinde nöronun aldığı girdi sinyaline bağlı olarak nöronun etkinleştirilip etkinleştirilmediğini belirlemektedir (Iosifidis & Tefas, 2022). Aktivasyon fonksiyonunun çıkışı belirli bir eşik değerinin üzerindeyse, nöron etkinleştirilir ve sinir ağının bir sonraki katmanına bir sinyal gönderilir. Aktivasyon fonksiyonunun seçimi, bir sinir ağının performansını belirlemede önemli bir faktördür (Iosifidis & Tefas, 2022).

Literatürde birçok aktivasyon fonksiyonu vardır. Şekil 3.6'da yaygın olarak kullanılan bazı aktivasyon fonksiyonlarının grafiği verilmiştir.



Şekil 3.6. Yaygın olarak kullanılan aktivasyon fonksiyonları (S. Khan vd., 2018).

### 3.3. Çok Katmanlı Algılayıcı (MLP) Model

Birçok nöronun katmanlarını istifleme(stacking) ilkesine dayanan sinir ağı mimarisi, çok katmanlı algılayıcı (MLP, Multilayer Perceptron) olarak adlandırılır (Hu vd., 2000).

#### 3.3.1. Algılayıcı Model

MLP modeli, 1950'lerde Rosenblatt tarafından önerilen algılayıcı modelinin bir çeşididir (Rosenblatt, 1958). Algılayıcı modelde, doğrusal ağırlıklara ve sınır değerli (threshold) aktivasyon fonksiyonuna sahip tek bir nöron kullanılır.

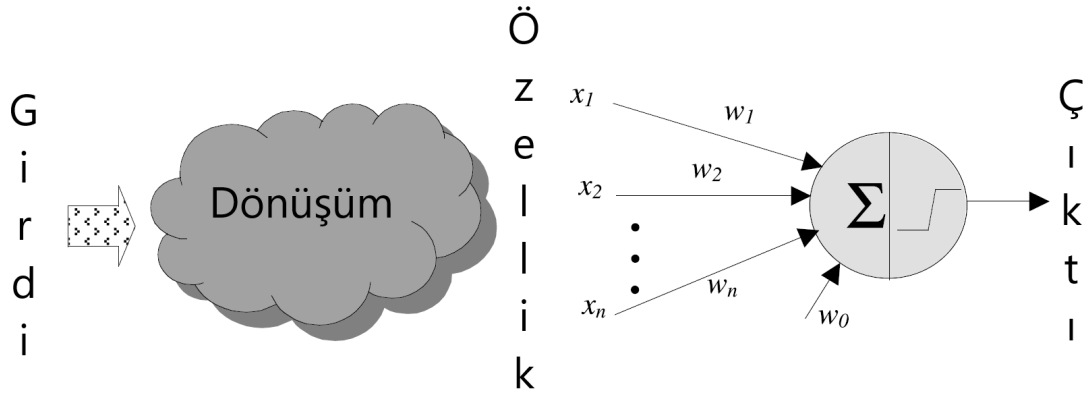
$\underline{x} = (x_1, x_2, \dots, x_n)$  olmak üzere

$$u(\underline{x}) = w_0 + \sum_{i=1}^n w_i x_i$$

$$y(u(\underline{x})) \begin{cases} 1 & u(\underline{x}) \geq 0 \\ 0 & u(\underline{x}) < 0 \end{cases} \quad \text{veya} \quad y(u(\underline{x})) \begin{cases} 1 & u(\underline{x}) \geq 0 \\ -1 & u(\underline{x}) < 0 \end{cases}$$

elde edilir. Aşağıdaki bilgiler, Şekil 3.7 baz alınarak bir nöronu tanımlayan parametrelerdir (Shanmuganathan & Samarasinghe, 2016);

- $\underline{x} = (x_1, x_2, \dots, x_n)$  girdi vektörü,  $x_i$ 'ler ise girdi olarak tanımlanır. Ayrıca  $\underline{x}$  vektörü öznitelik vektörü,  $x_i$ 'ler ise öznitelik olarak da tanımlanmaktadır.
- $w_1, w_2, \dots, w_n$  ağırlıklar olarak adlandırılmaktadır.  $i = 1, 2, \dots, n$  olmak üzere her  $w_i$  ağırlığı bir  $x_i$  girdisi ile eşlenmektedir.  $w_0$  ise yan(bias) olarak tanımlanmaktadır.
- $u$  fonksiyonu, genellikle, toplam fonksiyonu olarak tanımlanmaktadır. Ayrıca  $u$  fonksiyonu girdi fonksiyonu olarak da tanımlanabilmektedir.
- $y$  fonksiyonu aktivasyon fonksiyonu olarak tanımlanmaktadır. Aktivasyon fonksiyonu nöronun aktivasyon seviyesini hesaplar.



Şekil 3.7. Bir algılayıcı sinir ağı modeli (Hu vd., 2000).

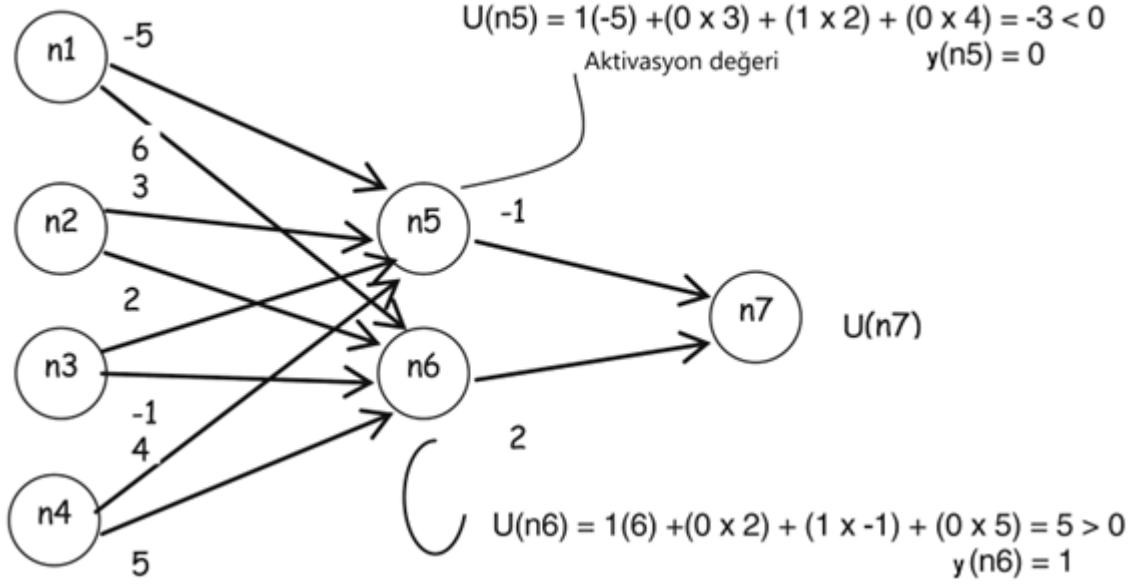
### 3.3.2. Çok Katmanlı Algılayıcı Model

Çok Katmanlı Algılayıcı, sınıflandırma ve regresyon görevleri için makine öğreniminde yaygın olarak kullanılan bir tür yapay sinir ağıdır. Girdi verisi üzerinde doğrusal olmayan dönüşümler gerçekleştiren, birbirine bağlı düğümlerin çoklu katmanlarından oluşur (Bishop, 1995).

Bir MLP eğitimi, tahmin edilen çıktı ile gerçek çıktı arasındaki farkı ölçen bir kayıp fonksiyonunu en aza indirmek için nöronların ağırlıklarını yinelemeli olarak optimize etmeyi içerir (Rumelhart vd., 1986).

MLP'ler, görüntü sınıflandırma, doğal dil işleme ve konuşma tanıma dahil olmak üzere çok çeşitli uygulamalarda başarıyla kullanılmıştır. MLP'lerin güçlü yönlerinden biri, ağın mimarisini ve hiperparametrelerini değiştirerek farklı türdeki girdi verisine ve çıktılara kolayca uyarlanabilmeleridir (Goodfellow vd., 2016).

Bununla birlikte, MLP'lerin küçük veri kümelerine aşırı uyma (overfitting) eğilimi ve yüksek boyutlu veri kümelerini hızlı işleyememe gibi bazı dezavantajları vardır. Bu sınırlamalar, CNN gibi alternatif yapay sinir ağları mimarilerinin geliştirilmesine motivasyon kaynağı olmuştur (LeCun vd., 2015).



Şekil 3.8. Çok katmanlı algılayıcı bir model (Shanmuganathan&Samarasinghe, 2016).

Şekil 3.8'de çok katmanlı algılayıcı bir model örneği verilmiştir. Burada  $\underline{x} = (x_1, x_2, x_3, x_4) = (1, 0, 1, 0)$  vektörü girdileri ifade etmekte,  $w_1, w_2, w_3, w_4, w_5, w_6, w_7, w_8 = -5, 6, 3, 2, 2, -1, 4, 5$  ise ağırlık değerlerini ifade etmektedir.  $U$  fonksiyonu bir toplam fonksiyonudur,  $y$  aktivasyon fonksiyonu ve  $y$  fonksiyonunun çıktısı ise aktivasyon değeridir.

### 3.4. Öğrenme Stratejileri

Öğrenme, doğru tahminler veya sınıflandırmalar yapma yeteneğini geliştirmek için bir modelin, bir veri kümesi üzerinde eğitildiği süreci ifade eder. Model, verideki kalıpları ve ilişkileri öğrenmek için istatistiksel teknikler kullanır ve bu modeller yeni, daha önce karşılaşılmamış veri hakkında tahminler yapmak veya kararlar almak için kullanılır. Jordan ve Mitchell'e (2015) göre öğrenme, "bir modelin bir görevdeki performansını deneyim yoluyla iyileştirme yeteneği" olarak tanımlanmaktadır. Literatürde birçok öğrenme türü vardır. Öğrenme türleri Denetimli Öğrenme (Supervised Learning), Denetimsiz Öğrenme (Unsupervised Learning), Yarı Denetimli Öğrenme (Semi-Supervised Learning) ve Takviyeli Öğrenme (Reinforcement Learning) olarak gruplanabilir (Kollmannsberger vd., 2021).

### 3.4.1. Denetimli Öğrenme

Bu öğrenme türü, en yaygın uygulanan öğrenme türüdür (Chollet, 2017). Bir dizi örnek verildiğinde, girdi verisini bilinen hedeflere eşlemeyi öğrenmekten oluşur. Model, eğitim kümesindeki her girdi için tahmin edilen çıktı ile gerçek çıktı arasındaki hatayı en aza indirerek eğitilir. Denetimli öğrenme genellikle girdi ve çıktı arasında açık bir eşleme olduğunda ve amaç yeni girdilere genellenebilecek bir işlevi öğrenmek olduğunda kullanılır (Murphy, 2013). En sık kullanılan denetimli öğrenme yöntemleri ise şunlardır (Huang & Le, 2021):

- En yakın komşu algoritması
- Karar ağaçları ve rastgele ormanlar
- Destek vektör makineleri
- Derin sinir ağları

### 3.4.2. Denetimsiz Öğrenme

Denetimsiz öğrenme, eğitim verisinin çıktı etiketlerine sahip olmadığı bir tür makine öğrenimidir. Çıktılar yerine model, girdi verisindeki kalıpları, örüntüleri ve yapıyı kendi başına bulmakla görevlidir. Denetimsiz öğrenme, açık etiketler veya hedef çıktılar olmadan veri kümesindeki yapıyı tanımlama sürecidir (Bishop, 2007). Modelin performansı hakkında açık bir geri bildirim olmadığından, denetimsiz öğrenme, genellikle, denetimli öğrenmeden daha zordur (Goodfellow vd., 2016). Denetimsiz öğrenme algoritmasının genel görevleri, kümeleme ve boyut indirgemeyi içerir. Kümeleme, girdi verisini her gruptaki örnekler arasındaki korelasyona dayalı olarak, alt gruplara sınıflandırmak için kullanılır. Boyut indirgeme, veri kümesinin yapısını ve özelliklerini korurken, yüksek boyutlu veriden, daha düşük boyutlu veriye dönüştürme işlemidir. Aşağıdakiler yaygın olarak kullanılan denetimsiz öğrenme yöntemleridir (Huang & Le, 2021):

- Kümeleme sorunu için
  - K-ortalama algoritması
  - Hiyerarşik küme algoritması
- Boyut küçültme sorunu için
  - Temel bileşen analizi algoritması
  - Çekirdek temel bileşen analizi

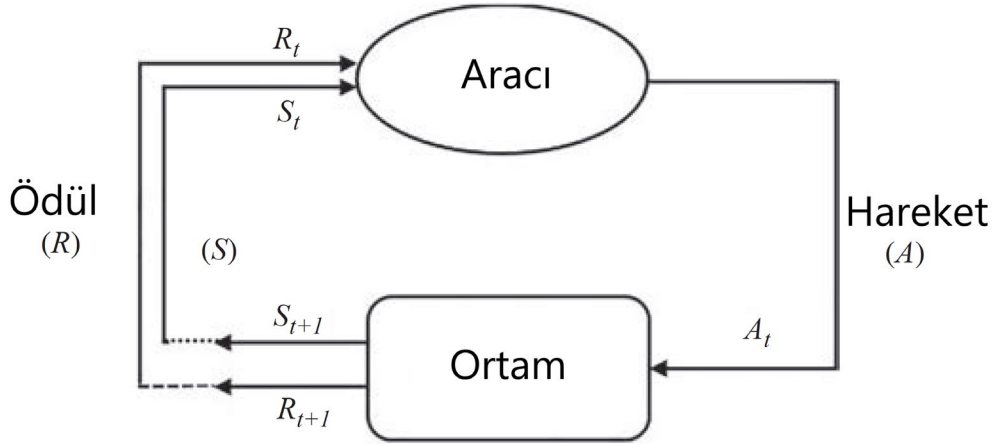


### 3.4.3. Yarı Denetimli Öğrenme

Modelleri eğitmek için, büyük miktarda etiketlenmemiş veri ve az miktarda etiketlenmiş veri içeren bir veri kümesi kullanan bir yaklaşımdır. Tipik olarak, yarı denetimli öğrenme algoritmaları, diğeriyle ilgili bilgileri kullanarak bu iki teknikten birindeki performansı artırmak amacıyla denetimli öğrenme ile denetimsiz öğrenmeyi birleştirir. Bu algoritmalar genellikle, bilgisayar destekli teşhis, dilbilgisi etiketleme gibi, etiketlenmiş verinin az veya elde edilmesinin zor olduğu sınıflandırma problemlerinde uygulanır (Huang & Le, 2021). Yarı denetimli öğrenmedeki zorluklardan biri, modelin etiketlenmemiş veriye fazla uyması durumudur. Bu durum modelin genelleme performansının zayıf olmasına neden olabilmektedir (Zhu, 2005).

### 3.4.4. Takviyeli Öğrenme

Google DeepMind'in bu öğrenme türünü Atari oyunlarını oynamayı öğrenmeye (ve daha sonra en üst düzeyde Go oynamayı öğrenmeye) başarılı bir şekilde uygulamasından sonra son zamanlarda çok fazla dikkat çekmeye başlamıştır (Chollet, 2017). Takviyeli öğrenme, bir temsilcinin beklenen kümülatif ödülü en üst düzeye çıkarmak için etkileşimli bir ortamda öğrenmesine ve harekete geçmesine izin veren bir tekniktir (Huang & Le, 2021). Spesifik olarak,  $t$  anında, ortamdaki durum  $S_t$  ve ödül  $R_t$  aracıya gönderilir, ardından aracı, ortama geri göndermek için bir dizi mevcut eylemden bir  $A_t$  eylemi seçme kararını verir. İşlem, Şekil 3.9'da gösterildiği gibi ortamın yeni bir durumu  $S_{t+1}$  ve ödül  $R_{t+1}$  ile tekrarlanır (Huang & Le, 2021).



Şekil 3.9. Takviyeli Öğrenme.

### 3.5. Öğrenme Hızı

Optimizasyon işlemi sırasında modelde yapılan değişikliklerin büyüklüğünü belirleyen bir hiperparametredir. Öğrenme oranı, yakınsama hızını ve optimizasyon sürecinin kararlılığını dengelemede önemli bir rol oynamaktadır (Goodfellow vd., 2016).

Öğrenme hızı modelin performansı üzerinde önemli bir etkiye sahiptir ve ampirik deneylere dayalı olarak uygun bir öğrenme oranı seçilmesi gerekmektedir (Lecun vd., 1998). Uygulamada, çok yüksek bir öğrenme oranı, modelin optimal çözümü yakınsama esnasında atlamasına neden olurken, çok düşük bir öğrenme oranı ise daha yavaş yakınsamaya veya yerel minimumda takılmaya neden olabilmektedir (Bengio ve Courville, 2013).

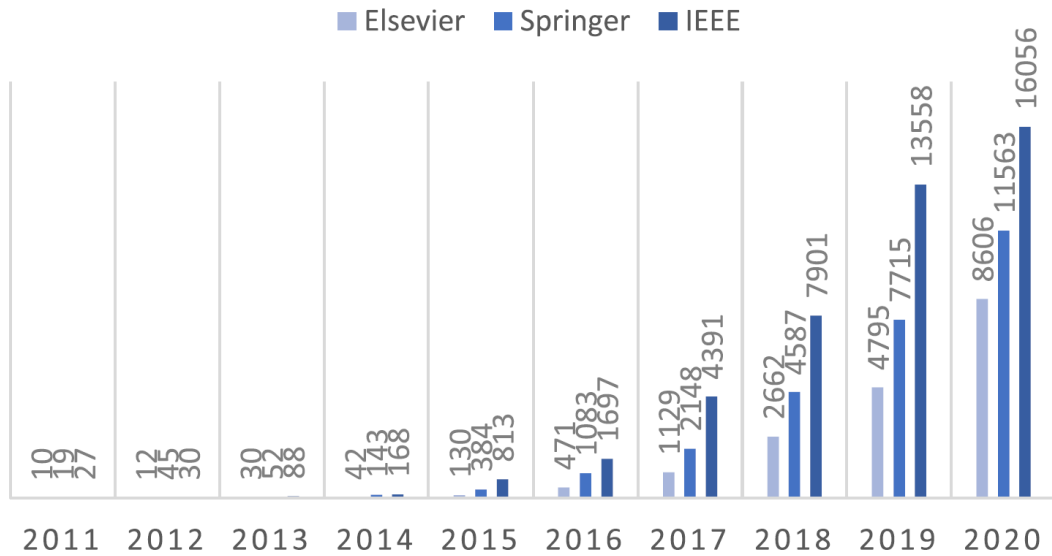
### 3.6. Evrimsel Sinir Ağları

Izgara benzeri bir yapıya sahip veriyi işlemek için geliştirilmiş özel bir sinir ağıdır ve geliştirmeye aşamasında iken, ilk başta sadece görüntü işleme işleri için tasarlanmıştır (LeCun vd., 2015). Bir görüntüdeki bir nesnenin özelliklerini ayırt ederken, nesneyi tanımlayan özelliklerin birbirine yakın bulunduğu varsayılmaktadır. Bir nesnenin özelliklerini tanımlanması için görüntüdeki tam konumu önemli değildir, sadece özelliklerin birbirine göre konumu önemlidir. Sıradan bir sinir ağı gibi, CNN de öğrenilebilir, ağırlıkları optimize edilebilir nöronlardan oluşur. Farklı olan, işlenen verinin düzenlenmesi, evrim ve

havuzlama gibi gizli katmanlarda kullanılan matematiksel işlemdir. Burada girdi katmanı bir vektör değil, bir görüntüdeki piksellerin ızgara benzeri yapısını hesaba katan bir matristir.

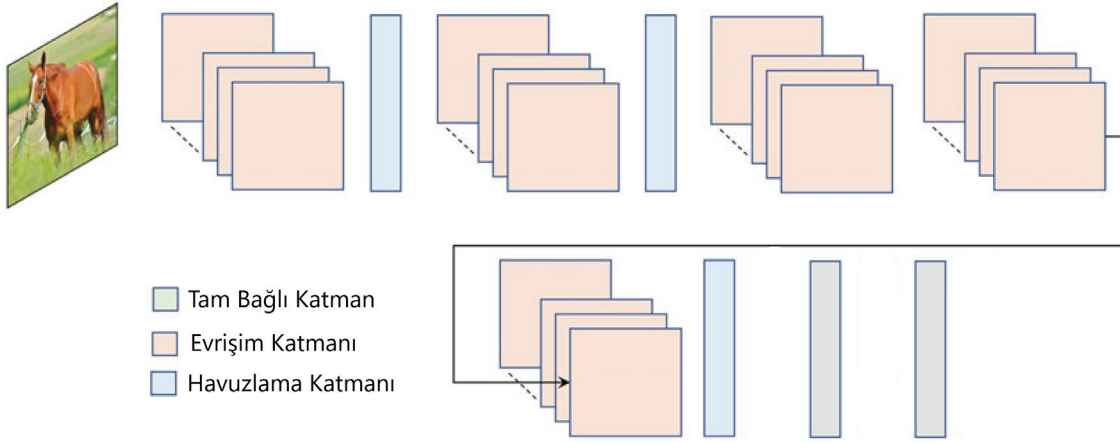
CNN'lerin temel yapısı olan ve şu anda da kullanılan gradyan tabanlı eğitim yaklaşımı 1980'lerde önerilmiş olsa da uzun süre kullanılmamış ve belge okuma gibi birkaç özel durumda kendine yer edinmiştir (Iosifidis & Tefas, 2022). Özellikle, büyük ağlar için gradyan tabanlı eğitimin kaçınılmaz olarak zayıf bir yerel optimuma sıkışıp kalacağı veya optimum noktayı atlayacağına dair yaygın bir görüş bulunmaktaydı (Iosifidis & Tefas, 2022). Bu varsayımın daha sonra hem teorik inceleme hem de CNN'lerin pratik başarısı ile yanlış olduğu ortaya konulmuştur (Iosifidis & Tefas, 2022).

Krizhevsky vd. (2012), 2012 yılında AlexNet isimli bir CNN modeli ile ImageNet Büyük Ölçekli Görsel Tanıma isimli uluslararası yarışmayı kazanmıştır (Russakovsky vd., 2015). Çalışma; 1000 sınıftan oluşan, yaklaşık bir milyon adet görüntü üzerinden, 62 milyon eğitilebilir parametre kullanarak yapılmıştır (Krizhevsky vd., 2012). AlexNet'in başarısı, CNN'e karşı geniş bir ilgi uyandırmış Şekil 3.10'da gösterildiği gibi, üç büyük yayıncı olan Elsevier, Springer ve IEEE'den elde edilen veriye göre hızla artmaya başladığı görülmüştür (Iosifidis & Tefas, 2022).



Şekil 10. Üç büyük yayıncıya göre CNN ile ilgili yapılan çalışmalar (Iosifidis & Tefas, 2022).

Son yıllarda, özellikle derin öğrenme teknikleri arasında, evrimsel sinir ağı birçok alanda kendine yer edinmiştir. Tarımdan radyolojiye, otonom sürüşten uzaktan algılamaya ve böcek bilimine kadar farklı alanlarda yaygın bir şekilde kullanılmaktadır (Iosifidis & Tefas, 2022). Şekil 3.11'de örnek CNN mimarisi verilmiştir. Örnekteki mimari tam bağlı katmanlar, evrişim katmanları ve havuzlama katmanlarından oluşmaktadır.

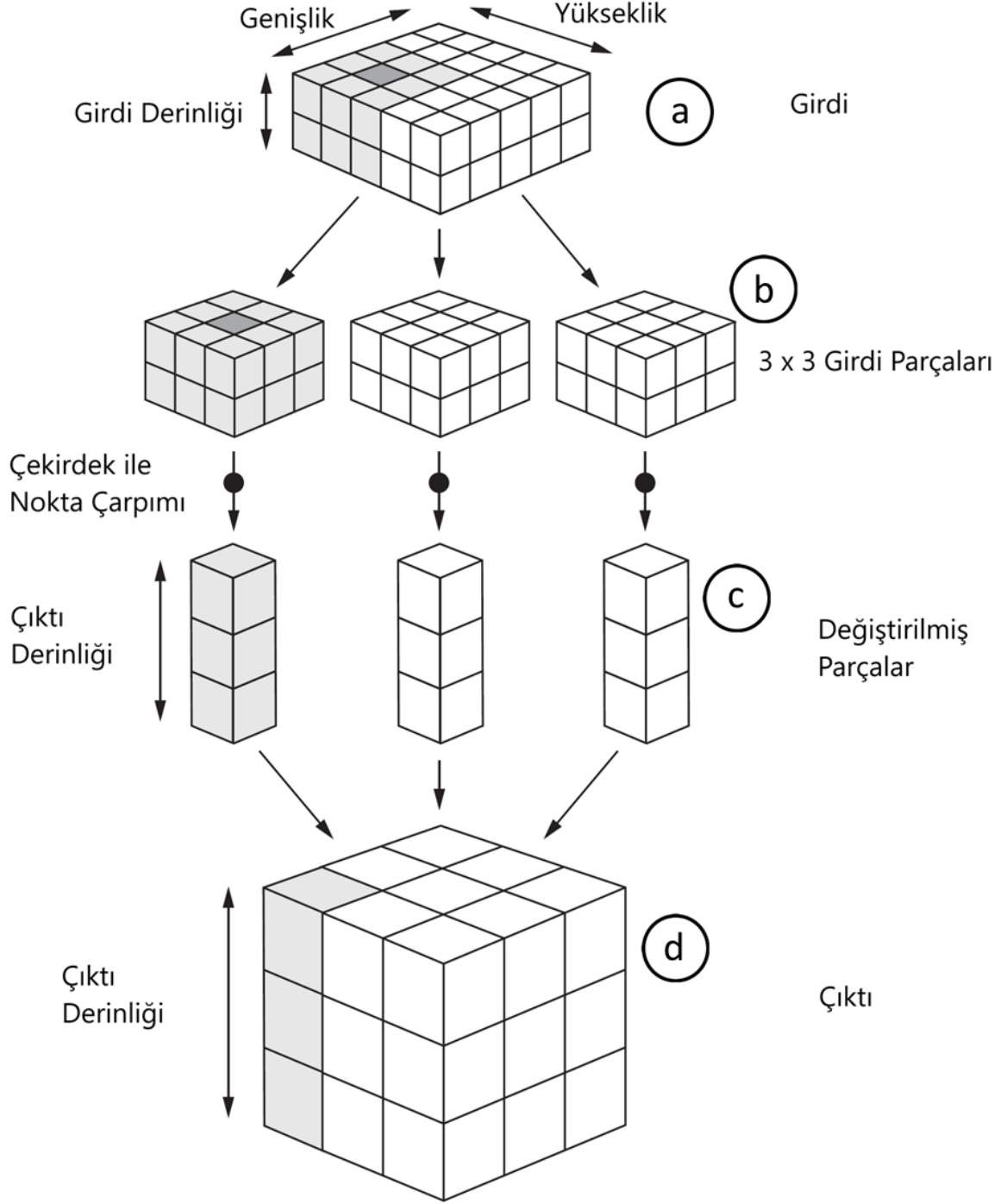


Şekil 11. Örnek CNN yapısı (Nelson, 2023).

### 3.6.1. Evrişim Katmanı

Evrşim işlemi, görüntü, ses ve metin gibi girdi veri kümelerinden özellikler türetmeyi sağlayan, CNN'lerdeki temel işlemdir. Evrişim işlemi kayan bir pencere olarak adlandırılan bir çekirdek içerir. Evrişim işlemi ile ağı, girdi verisinden daha karmaşık özellikler çıkarmayı öğrenebilir ve evrişim işlemi ile nesne algılama, görüntü sınıflandırma ve konuşma tanıma gibi geniş bir yelpazede görevleri yerine getirebilir (Iosifidis & Tefas, 2022). Farklı çekirdeklerle girdi verisi filtrelenerek birden fazla evrişim işlemi uygulanabilir. Şekil 16'de evrişim işleminin mantığı açıklanmıştır;

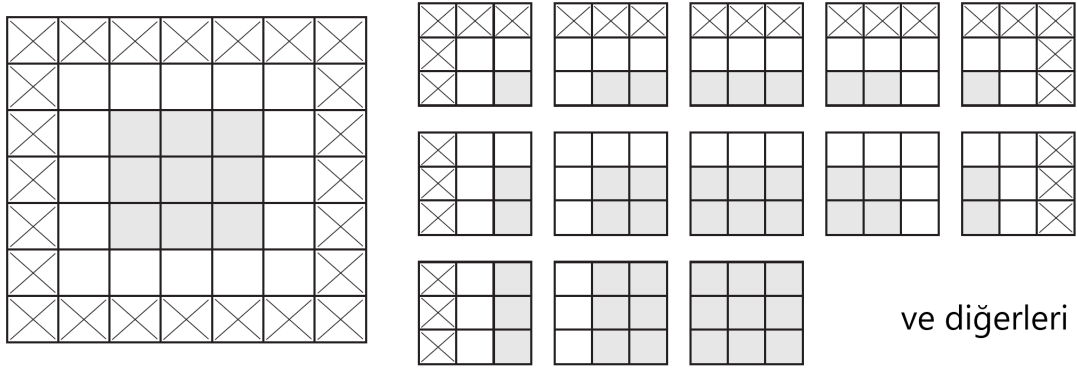
- Şekil 3.12 (a)'de  $5 \times 5 \times 2$  girdi verisi gösterilmiştir.
- Şekil 3.12 (b)'de girdinin,  $3 \times 3 \times 2$ 'lik parçaları verilmiştir.
- Şekil 3.12 (c)'de bir önceki adımdaki girdi parçalarının, çekirdek ile nokta çarpımı işlemi sonucu hesaplanan  $1 \times 1 \times 3$  çıktı parçaları verilmiştir.
- Şekil 3.12 (d)'de  $3 \times 3 \times 3$  çıktı verisi gösterilmiştir.



Şekil 3.12. Evrişim işleminin çalışma mantığı (Chollet, 2017).

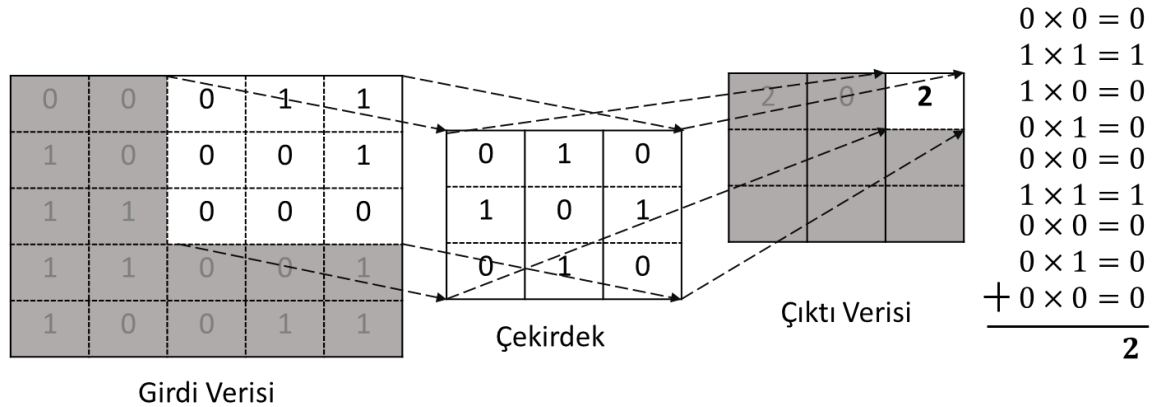
Girdi özellik haritasıyla aynı boyutlara sahip bir çıktı özellik haritası elde edilmek istendiğinde, dolgu (padding) özelliği kullanılmaktadır. Dolgu, her girdi parçasının etrafına uygun sayıda satır ve sütun eklenmesi sonucu oluşan parçadır. Örneğin, Şekil 3.13'te gösterildiği gibi 5x5 olarak verilen girdi verisine 3x3 filtre uygulanıp 5x5 çıktı verisi alınması için, girdinin dört tarafını saracak şekilde satır ve sütun

eklenmelidir. Eklenen satır ve sütunlardan sonra operasyonlar gerçekleştirilmelidir.



Şekil 13. Dolgu özelliğinin kullanılmasına ilişkin basit bir gösterim.

Çıktı boyutunu etkileyebilecek bir diğer faktör ise adım(stride) parametresidir. Ardışık iki pencere arasındaki mesafeye adım adı verilmektedir. Genelde bu değer 1 olarak kabul edilmektedir ve evrişimin önemli bir parametresidir.

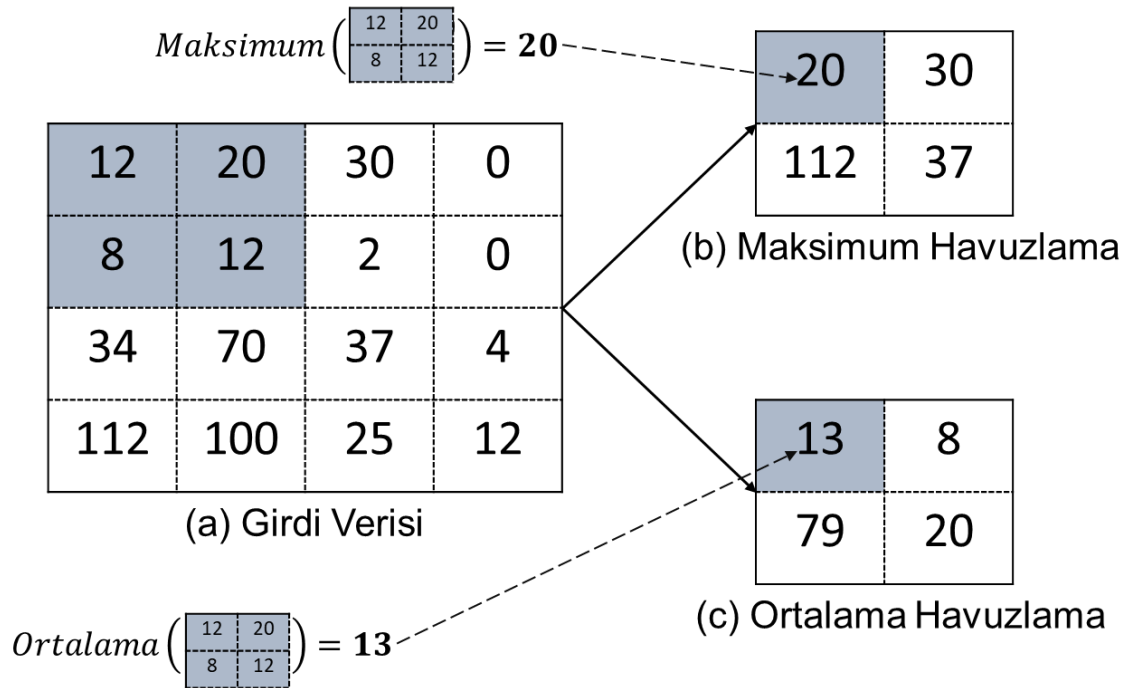


Şekil 14. Dolgu parametresi ve adım parametresi olan evrişim operasyonu.

Şekil 3.14'te dolgu parametresi 0 ve adım parametresi 1 olan evrişim operasyonu verilmiştir. Burada girdi verisi 5x5, çekirdek ise 3x3 olarak verilmiştir. Operasyon, 3x3'lük girdi parçası ile çekirdeğin nokta çarpımının hesaplanması ve hesaplanan öğelerinin toplanması ile gerçekleştirilmiştir. Bu işlem her bir girdi parçası için tekrarlanmıştır. Belirlenen dolgu parametresi ve adım parametresi ile, çıktı verisi 3x3 boyutunda elde edilmiştir.

### 3.6.2. Havuzlama Katmanı

Havuzlama (pooling) katmanı, evrişimsel katmanlar tarafından çıkarılan özellik haritalarının boyutunu azaltmakta kullanılmaktadır. Havuzlama işlemi sayesinde hesaplama maliyeti azalmakta ve takip eden evrişimsel katmanın farklı bir ölçekte özellikler çıkarmasına izin verilmektedir. Havuzlama için, maksimum havuzlama ve ortalama havuzlama olmak üzere iki ana yaklaşım vardır. Maksimum havuzlamada, penceredeki maksimum değer seçilmekte, ortalama havuzlamada ise alanın tüm değerlerinin ortalaması alınmaktadır (Iosifidis & Tefas, 2022). Şekil 3.15'te adım parametresi 2 olan bir havuzlama işlemi verilmiştir. Şekil 3.15 (a) 4x4'lük girdi matrisi, Şekil 3.15 (b)'de maksimum havuzlama işlemi sonucu oluşan matris ve Şekil 3.15 (c)'de ortalama havuzlama işlemi sonucu oluşan matris verilmiştir.

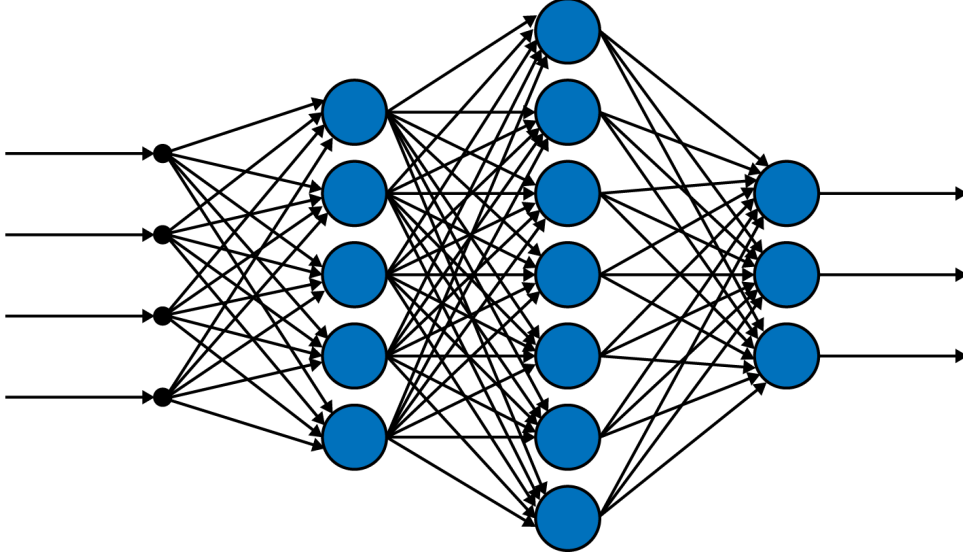


Şekil 3.15. Maksimum ve ortalama havuzlama (Teoh, 2023).

### 3.6.3. Tam Bağlı Katman

Tam bağlı katmanlar (FC, Fully Connected Layer),  $1 \times 1$  boyutunda filtrelere sahip evrişim katmanlarıdır (S. Khan vd., 2018). Tam bağlı bir katmandaki her

birim, bir önceki katmanın tüm birimlerine bağlıdır. Şekil 3.16'da tam bağlı katman yapısı verilmiştir.



Şekil 3.16. Tam bağlı katmanlar.

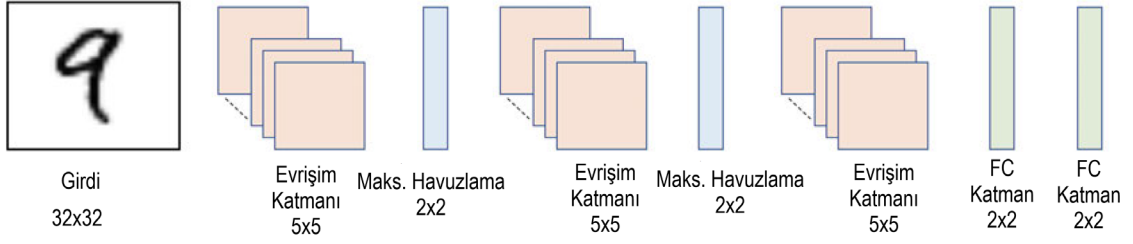
### 3.6.4. Evrimsel Sinir Ağı Mimarileri

Özellikle son 10 yılda birçok CNN mimarisi geliştirilmiştir (Shrestha & Mahmood, 2019). Farklı mimarilerin geliştirilmesindeki temel amaç, modellerin tahmin performansını artırmaktır. Literatürde diğer mimarilere temel oluşturan ve yaygın olarak kullanılan bazı CNN mimarileri aşağıdaki gibi verilebilir.

#### 3.6.3.1. LeNet

CNN mimarilerin tarihi LeNet (Lecun vd., 1998) mimarisi ile başlamaktadır. Bu modelin amacı el yazısı ile yazılmış olan rakamların tahmin etmektir. Bu mimari tasarımının başarılı bir versiyonu, toplamda 5 ağırlık katmanından oluşan LeNet-5 modelidir (S. Khan vd., 2018). Şekil 3.17'de gösterildiği gibi model 3 evrişim katmanı, 2 maksimum havuzlama katmanı ve son olarak 2 tam bağlı katmandan oluşmaktadır.

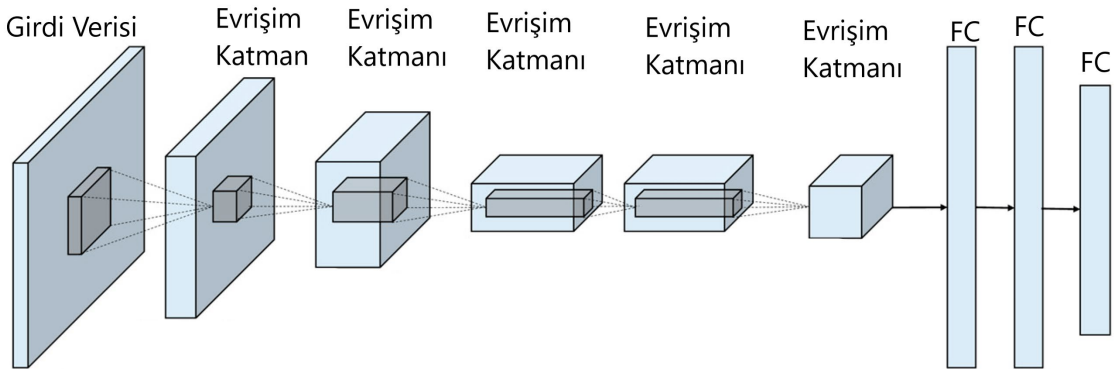




Şekil 3.17. LeNet-5 ağ Mimarisi (S. Khan vd., 2018).

### 3.6.3.2. AlexNet

LeNet (Lecun vd., 1995) derin CNN'lerin tarihini başlatan bir çalışmadır. Ancak o dönemde CNN ile ilgili yapılan çalışmalar el yazısı rakam tanıma görevleriyle sınırlı kalmış ve diğer görüntü sınıflandırma görevlerinde iyi performans sergileyememiştir. AlexNet (Krizhevsky vd., 2012), görüntü sınıflandırma ve nesne tanımlama görevleri için çok iyi sonuçlar üreten ilk derin CNN mimarisi olarak kabul edilmektedir (A. Khan vd., 2020). AlexNet'de, CNN'nin öğrenme kapasitesi derinleştirilerek ve birkaç parametre optimizasyon stratejisi uygulanarak geliştirmiştir (Krizhevsky vd., 2012). Şekil 3.18'de AlexNet mimarisi gösterilmiştir. Mimari, genellikle, 5 evrişim katmanı, aralarda havuzlama katmanları ve son olarak 3 tam bağlı katmandan oluşmaktadır.

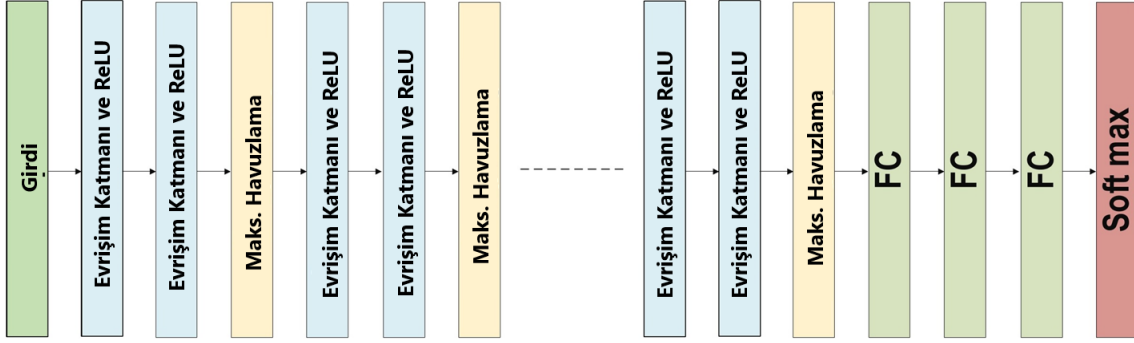


Şekil 3.18. AlexNet ağ Mimarisi (Jiang vd., 2021).

### 3.6.3.3. VGGnet

VGGNet, Oxford Üniversitesi'ndeki Visual Geometry Group ve Google DeepMind tarafından ortaklaşa geliştirilen bir CNN mimarisidir (Simonyan & Zisserman,

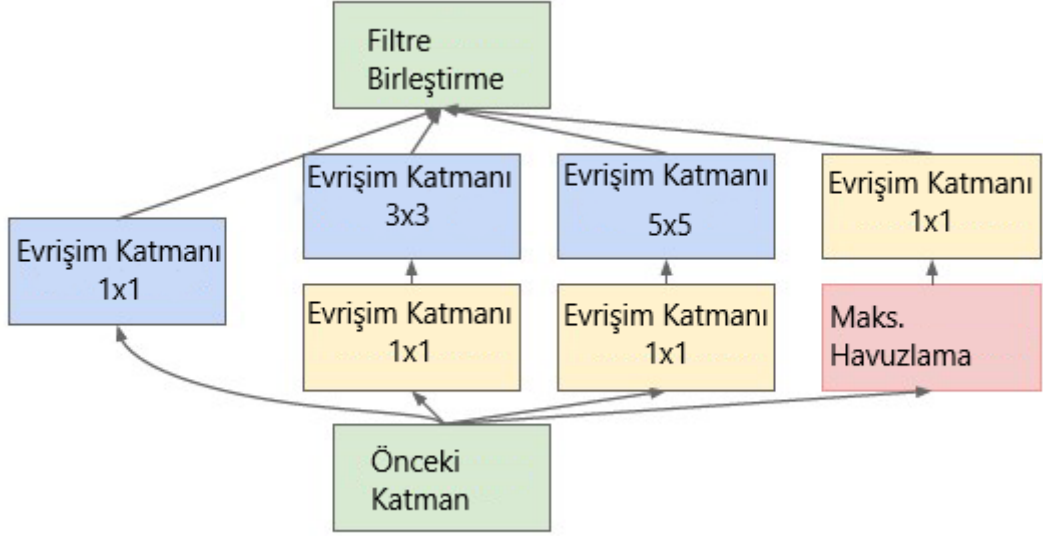
2014). Şekil 3.19'da gösterildiği gibi, VGGNet mimarisi, 3x3 evrişim ve 2x2 havuzlama katmanlarıyla karakterize edilen genişletilmiş bir AlexNet olarak düşünülebilir (Jiang vd., 2021). Ağ mimarisi, özellik öğrenmeyi artırmak için daha küçük evrişim katmanlarının kullanılmasıyla derinleştirilebilir. En yaygın kullanılan iki VGGNet versiyonu VGGNet-16 ve VGGNet-19'dur (Simonyan & Zisserman, 2014).



Şekil 3.19. VGGNet ağ mimarisi (Alzubaidi vd., 2021).

#### 3.6.3.4. GoogleNet

GoogleNet, 2014-ILSVRC yarışmasının birincisidir ve aynı zamanda Inception-V1 olarak da bilinen bir mimaridir. GoogleNet mimarisinin temel amacı, düşük bir hesaplama maliyetiyle yüksek doğruluk elde etmektir (Szegedy vd., 2015). CNN'de inception bloğu adı verilen yeni bir kavramı tanıtmıştır. Bu blok böl, dönüştür ve sonunda tekrar birleştir fikrini kullanarak çok ölçekli evrişim dönüşümlerini içermektedir (A. Khan vd., 2020). Inception bloğunun mimarisi Şekil 3.20'de gösterilmektedir. GoogleNet, büyük boyutlu çekirdekleri kullanmadan önce 1x1 konvolüsyonel filtreli bir darboğaz katmanı ekleyerek hesaplamaları düzenler. Bunun yanı sıra, gereksiz bilgileri atlayarak, gereksiz özellik haritalarını eleyen seyrek bağlantılar kullanılmıştır (A. Khan vd., 2020).



Şekil 3.20. GoogleNet ağ mimarisi (Jiang vd., 2021).

### 3.6.3.5. ResNet

He vd. (2016), ILSVRC 2015'in kazananı olan ResNet'i (Rezidüel Ağ) geliştirdiler. Hedefleri, önceki ağlara kıyasla kaybolan gradyan sorunundan bağımsız ultra-derin bir ağ tasarlamak olmuştur (Alzubaidi vd., 2021). Katman sayısına bağlı olarak birkaç tür ResNet geliştirilmiştir (34 katmandan başlayarak 1202 katmana kadar). En yaygın kullanılan tür ResNet50'dir ve 49 evrişimsel katman ile tek bir FC katmanından oluşur (Alzubaidi vd., 2021).

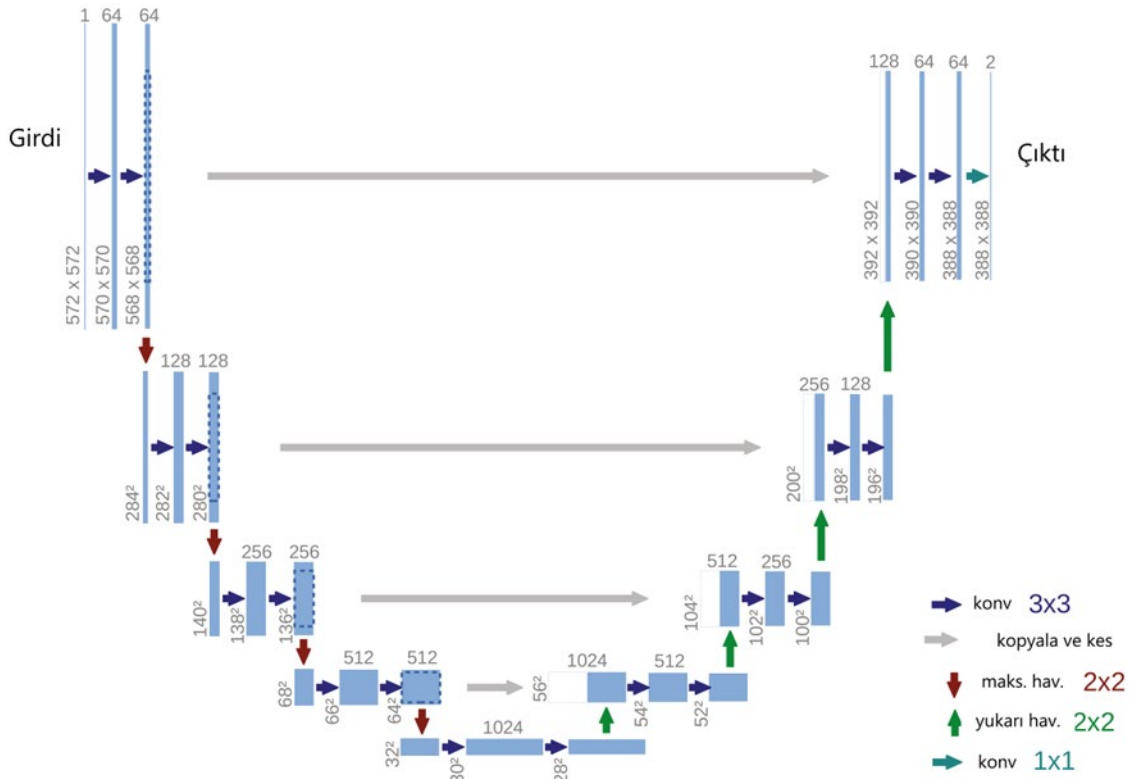
### 3.6.3.2. U-Net

Evrişimsel ağlar genellikle görüntü sınıflandırma görevlerinde kullanılmaktadır. Bir görüntünün neye ait olduğunu belirlemek için evrişimsel ağlar kullanılır. Örneğin, bir evrişimsel ağa bir köpek görüntüsü verildiğinde, ağın çıktısı "köpek" sınıfını temsil eden bir etiket olacaktır. Ancak, birçok görüntü işlemede, özellikle biyomedikal, yer bilimi gibi alanlar ile alakalı görüntülerde, istenen çıktı lokalizasyonu içermelidir, yani her piksele bir sınıf etiketi tanımlanmalıdır (görüntü segmentasyonu). Genellikle, oluşturulan modelleri eğitmek için çok sayıda eğitim görüntüsüne ulaşmak mümkün olmamaktadır. Kısıtlı görüntü ile doğru sonuçlar üretmek için Cirean vd. (2012) az sayıda veri ile çalışılabilen bir model mimarisi

üzerinde çalıştı ve ISBI 2012'deki EM segmentasyon yarışmasını kazandı (Ronneberger vd., 2015).

Cirean vd. (2012)'nin geliştirdikleri mimarinin iki dezavantajı vardır. İlk olarak, her bir alan için ağ ayrı çalıştırılması gerektiğinden oldukça yavaştır. İkinci olarak, araştırılan nesnelerin konumunun doğru belirlenmesinde problemi vardır (Ronneberger vd., 2015).

U-Net mimarisi "tamamen evrişimli ağ" olarak adlandırılan hassas bir mimariye dayanmaktadır (Long vd., 2014). Tamamen evrişimli ağ, tam bağlantılı katmanlardan oluşan yapılardan farklı olarak, yalnızca evrişimli katmanlardan oluşan bir ağ yapısıdır. Bu, ağın girdi boyutuna bağlı olmadan farklı boyutlardaki görüntülerle çalışabilmesini sağlamaktadır (Long vd., 2014). Bu özelliği sayesinde U-Net, piksel düzeyindeki etiketleme görevleri gibi görüntü segmentasyonunda kullanılmak üzere uyarlanmıştır. U-Net mimarisi, çok az sayıda eğitim görüntüsü ile çalışacak şekilde geliştirilmiştir ve böylece daha hassas segmentasyonlar elde edilebilmiştir (Ronneberger vd., 2015). Şekil 3.21'de U-Net mimarisi gösterilmiştir.



Şekil 3.21. U-Net ağ mimarisi (32x32) (Ronneberger vd., 2015).

Bu tez çalışmasında U-Net mimarisi aşağıdaki özelliklerinden dolayı seçilmiştir (Ronneberger vd., 2015);

- Pksel düzeyinde etiketleme: U-Net, piksel düzeyinde etiketleme görevleri için özel olarak tasarlanmış bir mimari olduğundan, sondaj karot sandıklarının segmentasyonu için uygun bir seçenektir. U-Net, görüntünün her pikselini ayrı ayrı etiketleyebilme yeteneği sayesinde, karot sandığının içerdiği farklı bölgeleri (örneğin, mineralizasyon bölgeleri, çatlaklar vb.) ayrıştırma ve analiz etme yeteneği sunabilecektir.
- Hassas segmentasyon: U-Net mimarisi, segmentasyon görevlerinde yüksek hassasiyet sağlama potansiyeline sahiptir. Tamamen evrişimli ağ yapısı ve bağlantıları, karot sandığındaki detaylarını koruyarak, düşük seviyeli ve yüksek seviyeli özellikleri birleştirerek daha hassas segmentasyon sonuçları elde edilmesini sağlayacaktır.
- Kısıtlı veride etkinlik: U-Net, az miktarda eğitim verisiyle bile etkili sonuçlar üretebilen bir mimaridir. Sondaj karot sandığı görüntülerinin, elde edilmesi ve etiketlenmesi zaman alıcı ve maliyetli bir süreçtir. U-Net, sınırlı veriyle çalışırken bile yüksek performans sunabilmektedir ve az sayıda veri olmasından kaynaklı yaşanan zorlukları aşılmasında olacaktır.
- Genellemeye uygunluk: Mimarinin esnek yapısı, farklı boyutlardaki görüntülerle çalışabilme ve genel segmentasyon yetenekleri sayesinde, farklı sondaj alanlarında kullanılabilmeye olanak tanıyacaktır.
- Performans: U-Net, biyomedikal görüntü segmentasyonu alanında etkili bir şekilde kullanılmış ve başarılı sonuçlar vermiştir. Burada biyomedikal görüntülerdeki yüzey ile karot görüntülerinin yüzeyi benzerlik göstermektedir. Bu benzerlikten hareketle modelin iyi sonuçlar üreteceği düşünülmektedir.

## 4. DENEYSEL ÇALIŞMALAR

Numunelerin laboratuvar sonuçlarının elde edilmesi süreci, zaman ve para açısından çok maliyetli olabilmektedir. Bu nedenle süreci hızlandıracak ya da elde edilecek sonuçları destekleyecek teknikler önemli bir rol oynamaktadır. Bu çalışmanın amacı, laboratuvar sonuçlarına bir alternatif ve/veya destekleyici, minimum maliyetli bir yöntem önermektir.

Çalışmada kullanılan veri Demir Export A.Ş. tarafından sağlanmıştır. Sivas'ın Divriği ilçesinde yer alan Demir Export A.Ş. Divriği Demir İşletmesi sahalarında yapılan sondaj çalışmaları neticesinde elde edilen görüntülerdir. Veri kümesini oluşturan görüntülerin örneği Şekil 4.1'de verilmiştir.



Şekil 4.1. Örnek karot sandığı fotoğrafları.

### 4.1. Veri Kümesi

Veri sondaj çalışmaları sırasında elde edilen karot sandığı fotoğraflarıdır. Sandıklar içerisinde karot numuneleri bulunmaktadır. Bir sandık genelde 4 sıra halinde, 1'er metre olmak üzere; 4 metre sondaj örneğini muhafaza etmekte olup bu metraj sandığın ve sondajın tipine göre değişkenlik gösterebilmektedir.

Çalışmada toplam 250 adet görüntü (veri) kullanılmıştır. Kullanılan görüntüler hem karot sandığı tespiti hem de karot üzerindeki cevherin tespiti için iki ayrı aşamada da kullanılmıştır. Bu görüntüler eğitim ve test olmak üzere, farklı sayılarda rasgele ikiye ayrılmıştır. Şekil 4.2'de gösterildiği gibi, görüntülerin 200 adeti eğitim işlemi için, 50 adeti ise modeli test etmek için kullanılmıştır.



Şekil 4.2. Eğitim ve test veri kümeleri.

Çalışmanın amacı görüntüler üzerinden demir cevherini tespit etmek ve “log” işlemini kolaylaştırmaktır. Demir cevheri görüntüler üzerinde, genellikle, ayırt edilebilir formdadır. Demir cevheri, karot numuneleri üzerinde, genellikle, koyu kahverengi olarak karşılaşılmaktadır.

#### 4.1.1. Görüntülerde Karşılaşılan Zorluklar

Sondaj işlemleri sonrasında alınan görüntüler her zaman aynı standartlarda ve yüksek kalitede olmayabilir. Bunun sebepleri; sondaj çalışmasının yapıldığı saha, ortam şartları (güneş, gölge vb.), görüntüleme cihazının yetersizliği, ölçek farklılığı vb. olabilir.

Şekil 4.3’te görüntünün çekildiği ortam nedeni ile ışık açısından etkilenen ve yerel gölgeler oluşan bir görüntü örneği verilmiştir.



Şekil 4.3. Gölge faktörünün önemli olduğu görüntü örneği.

Şekil 4.4’de 2 adet karot sandığının yan yana olduğu duruma ait görüntü örneği verilmiştir. Sahada sandıklar düzenli bir şekilde her zaman yerleştirilememektedir. Yan yana veya arka arkaya sandıklar sıralanabilmektedir. Geliştirilecek modelin bu duruma uyum sağlaması ve doğru sonuçlar üretmesi gerekmektedir.



Şekil 4.4. Birden fazla karot sandığının olduğu görüntü örneği.

Sahada, her zaman görüntüler uygun bir ortamda, ortam değişkenlerinden en az etkilenecek şekilde alınmamış olabilmektedir. Şekil 4.5'te bu duruma ait bir görüntü örneği yer almaktadır. Burada; açık havada, direk güneş ışığı altında ve karotlar kuru olarak görüntü alınmıştır.



Şekil 4.5. Açık ortamda alınan görüntü örneği.

Yukarıdaki gibi durumlar ile sık karşılaşılmaktadır. Çalışmada kullanılan modelin genelleştirme kabiliyetini artırmak ve farklı ortamdaki görüntüleri doğru



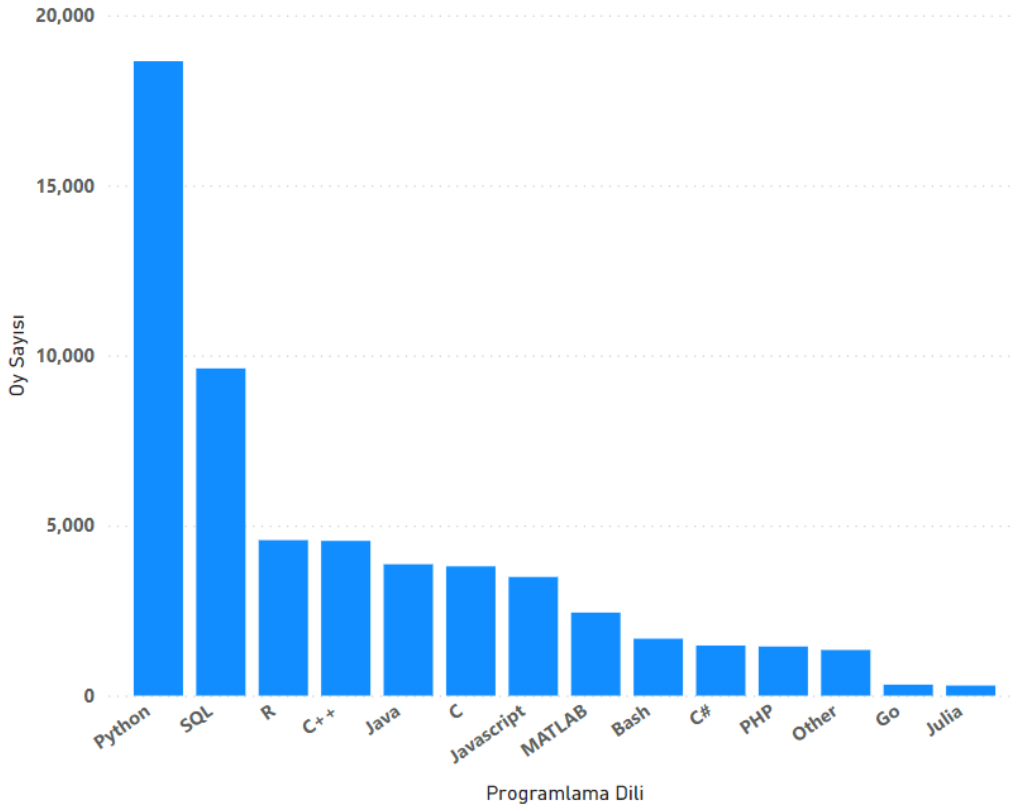
işleyebilmek için, birbirlerinden farklı karakterde görüntülerden oluşan bir veri kümesi oluşturulmuştur.

#### 4.1.2. Görüntü Özellikleri

Tez çalışmasının uygulamasında kullanılan görüntülerin dosya uzantısı “jpeg” formatındadır. Görüntülerin bit derinliği 24 (8x3) Bit'tir. Görüntüler, 5152 piksel genişlik ve 3864 piksel yükseklik olmak üzere büyük boyuta sahiptir. Görüntülerin ortalama boyutu ise 15 mb'tır.

#### 4.2. Veri İşlemede Kullanılan Programlama Dili ve Ekipman

Çalışma, Python programlama dili ile yapılmıştır. Son yıllarda Python programlama dili, veri analizi projelerinde oldukça sık kullanılmaktadır (Mooney, 2022). Şekil 4.6'da Python programlama dilinin yaygın kullanımına ilişkin grafik verilmiştir. Çalışmada Python programlama dili versiyonu olarak “3.9” kullanılmıştır.



Şekil 4.6. Veri bilimi uygulamalarında kullanılan programlama dilleri (Mooney, 2022).

Görüntü işleme çalışmalarında veri okutma, ön işleme, eğitim, skorlama gibi aşamalar bulunmaktadır. Bu aşamaları daha hızlı hale getirmeye yönelik Python kütüphaneleri vardır. Bunlardan bazıları aşağıdaki gibi listelenebilir;

- Numpy kütüphanesi, bilimsel hesaplamalar yapmak için kullanılmaktadır. Çok boyutlu bir dizi nesnesi, çeşitli türetilmiş nesnelere (maskelenmiş diziler ve matrisler gibi) ve diziler üzerinde hızlı işlemler için matematiksel, mantıksal, şekil işleme, sıralama, seçme dahil çeşitli kolaylıklar sağlayan bir Python kütüphanesidir (Harris vd., 2020).
- PyTorch kütüphanesi, çok boyutlu tensörler için veri yapılarını içerir ve bu tensörler üzerinden matematiksel işlemleri gerçekleştirir (Paszke vd., 2019).
- OpenCV kütüphanesi, görüntü işleme uygulamaları için ortak bir altyapı sağlamak ve kullanımı hızlandırmak amacıyla kullanılmaktadır (Bradski, 2000).
- Pillow kütüphanesi, görüntüleri okumak ve işlemek için, kapsamlı dosya formatı desteği olan bir kütüphanedir (Umesh, 2012).

Çalışma mobil iş istasyonu olan bir bilgisayarda yapılmıştır. Bilgisayar 32 GB belleğe (RAM) sahiptir ve işlemci 2.40 GHz hıza sahip Intel® Core™ i9-10885H işlemcisidir. Bilgisayarın ekran kartı, NVIDIA Quadro Serisi RTX 3000 modelidir. Ekran kartının CUDA desteği bulunmaktadır (NVIDIA vd., 2020). Bu da görüntülerin ekran kartı üstünde hızlıca işlenmesine olanak sağlamaktadır.

### **4.3. Görüntülerin Etiketlenmesi**

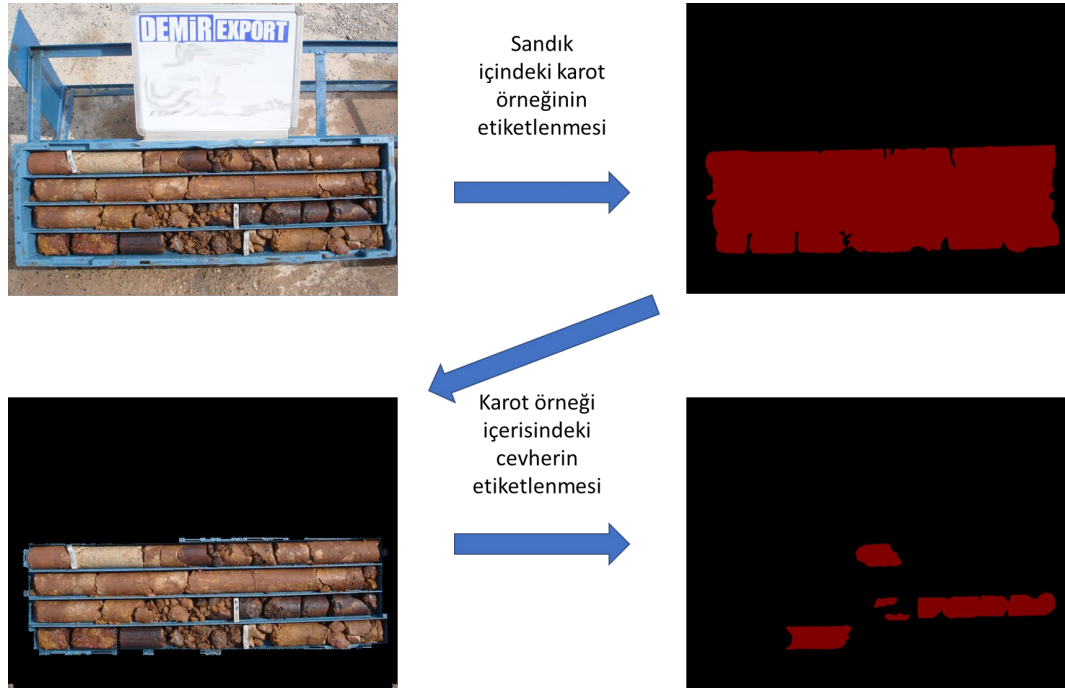
Görüntü işleme algoritmaları, görüntüleri anlamlandırmak ve içerdikleri özellikleri tanımlamak için kullanılır. Ancak bu algoritmaların etkili bir şekilde çalışabilmesi için, görüntülerin etiketlenmesi aşaması büyük bir öneme sahiptir.

Görüntülerin etiketlenmesi aşaması, görüntülerin içerdikleri nesne veya özelliklerin tanımlanması ve uygun etiketlerin verilmesi sürecidir. Bu süreçte, genellikle alan uzmanlarına (etiketleyicilere) ihtiyaç duyulmaktadır. Alan uzmanı, görüntülerde bulunan nesnelere belirleyerek etiketleri atarlar.

Görüntülerin etiketlenmesi aşaması, bazı zorlukları da beraberinde getirmektedir. Öncelikle, etiketleyicilerin kendi içlerinde tutarlılık sağlaması son derece önemlidir. Aynı görüntüyü farklı etiketleyiciler değerlendirdiğinde aynı sonuçları

vermeleri beklenmektedir. Büyük veri kümeleri üzerinde çalışıldığında, onlarca, binlerce hatta milyonlarca görüntüyü etiketlemek çok zaman alan bir süreç olabilmektedir.

Çalışmada kullanılan görüntüler, alan uzmanları (jeologlar) tarafından etiketlenmiştir. Etiketleme işlemi iki aşamalı gerçekleştirilmiştir. İlk aşama görüntü içerisinde karot sandığının etiketlenmesidir. Bu aşama, kurulacak olan model sayesinde görüntü içerisindeki fazla alanlardan kurtulmasını ve sadece istenilen alana odaklanmayı sağlamıştır. Diğer aşama ise sandığın içerisinde yer alan numunedeki cevherli bölgelerin tespitini gerçekleştirecek modelde kullanılacak görüntülerin etiketlenmesidir. Şekil 4.7’de görüntüler üzerindeki etiketlenen alanlar, iki aşamalı olarak gösterilmiştir. Çalışmada kullanılan 250 adet görüntünün (her iki aşama için) tümü bu aşamalardan geçerek etiketlenmiştir.



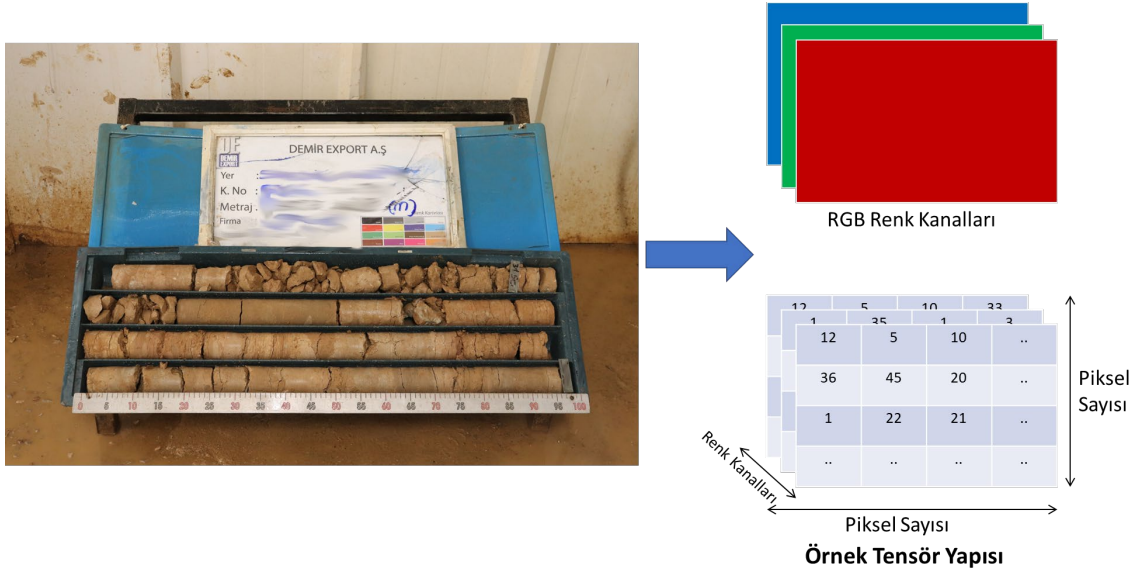
Şekil 4.7. Görüntü etiketleme aşamaları.

#### 4.4. Görüntülerin Okutulması

Görüntü okutma işlemi kolaylaştırmak ve standart hale getirmek için OOP (Nesne Tabanlı Programlama, Van Rossum & Drake, 2009) mimarisi kullanılarak bir kod betiği yazılmıştır. Kod betiğinin içerisinde, görüntü okutma işlemleri için Pillow kütüphanesinin fonksiyonları kullanılmıştır.

Görüntülerin orjinal boyutları çok yüksek olduğundan, görüntülerin işlenmesi zor ve bazı durumlarda imkansız olacaktır. Bu sebepten görüntülerin boyutları işlenebilir olacak şekilde küçültülmüştür. Yapılan denemeler ve tecbureler sonucunda, görüntü karakteristiğini değiştirmeyecek ve en-boy oranını en iyi temsil edecek şekilde boyut 400×600 piksel olarak belirlenmiştir.

Pytorch kütüphanesi ile görüntüleri işleyebilmek için, görüntüler tensör formatına çevrilmiştir. Şekil 4.8’de örnek renk kanalları ve tensör yapısı verilmiştir.



Şekil 4.8. Görüntü örnek tensör gösterimi.

Burada görüntülerin okutulması işlemi, tıpkı görüntü etiketleme işleminde olduğu gibi iki aşamalı gerçekleştirilmiştir. İlk aşamada karot sandıklarının içerisindeki numunelerin tespitini gerçekleştirmek amacıyla Şekil 4.9’daki örnekte olduğu gibi görüntülerin tamamı okutulmuş ve kırmızı ile gösterilen alanlar hedef olarak okutulmuştur.



Şekil 4.9. Karot sandığı içerisindeki numunenin tespiti için etiketlenmiş görüntü örneği.

İkinci aşamada ise Şekil 4.10'daki örnekte olduğu gibi ilk aşamada karot numunesi içerisinde bulunan cevherli alanı tespit etmek için, sadece karot numunesi ve numune üzerinde kırmızı ile belirtilen cevherli bölge okutulmuştur.



Şekil 4.10. Numunedeki cevherli bölgenin tespiti için etiketlenmiş görüntü örneği.

#### 4.5. Model Parametreleri ve Eğitim

Diğer yapay sinir ağıları modellerinde olduğu gibi U-Net modelini eğitmek için de bazı önemli parametrelerin belirlenmesi gerekmektedir. Bu parametreler, modelin eğitim sürecini kontrol etmek ve optimize etmek için kullanılmaktadır. Hem karot sandığı içerisindeki karot numunesinin tespiti hem de numune üzerindeki cevherli bölgelerin tespiti için; model parametreleri yapılan denemeler ve tecrübelerden hareketle belirlenmiş ve aşağıda detayları verilmiştir.

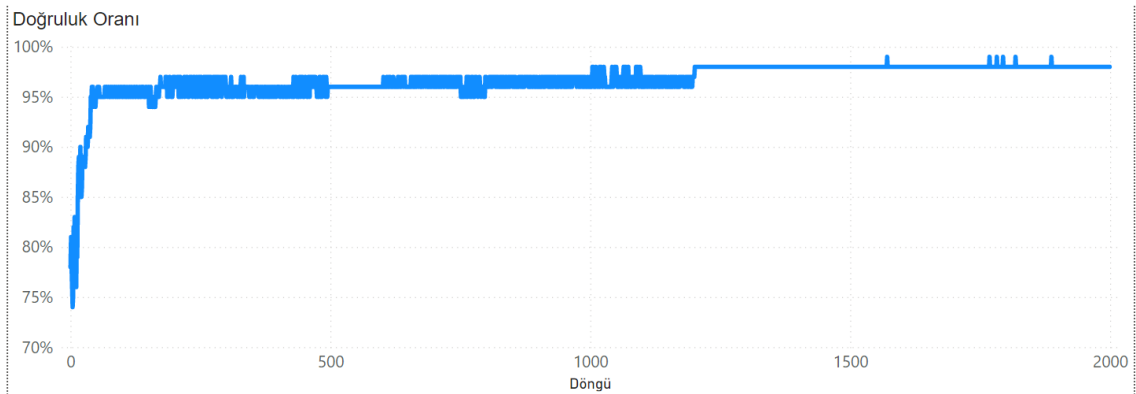
- Öğrenme Hızı (learning rate): Öğrenme hızını belirleyen bir parametredir. Modelin ne kadar hızlı veya yavaş öğreneceğini kontrol etmektedir. Parametre değeri 0.00001 olarak belirlenmiştir.
- Cihaz (device): Modelin hangi cihazda çalışacağını belirler. Eğer CUDA destekli bir GPU kullanılıyorsa, "cuda" değeri atanmaktadır; aksi halde "cpu" olarak atanmaktadır. Çalışmanın gerçekleştirildiği bilgisayarda CUDA desteğine sahip bir ekran kartı bulunduğundan parametre "cuda" olarak belirlenmiştir.
- Toplu iş boyutu (batch size): Her eğitim adımında işlenecek veri örneklerinin sayısını belirler. Burada, 1 olarak ayarlanmıştır, yani her adımda yalnızca bir örnek işlenecektir.
- Döngü sayısı (epoch): Eğitim süresince kaç döngü yapılacağını belirler. Burada; süre ve diğer kısıtlardan dolayı döngü sayısı 2000 olarak belirlenmiştir.
- Bellek sabitleme (pin memory): Veri yüklemesi sırasında belleği tutmak için özel bir bellek alanı ayrılmasını sağlamaktadır. Bu bellek alanı, GPU belleği ile daha hızlı veri transferini gerçekleştirmek için kullanılmaktadır. Burada, parametre değeri "True" olarak, yani aktif olarak ayarlanmıştır.

Eğitim sürecinin temel adımı, modelin eğitim döngüsüdür. Her bir döngü, belirlenen parametrelerle beraber, veri setinin tamamının model üzerinden geçirilmesini sağlar. Her veri örneği, model tarafından işlenir ve tahminler elde edilir. Elde edilen tahminler, gerçek etiketlerle karşılaştırılır ve kayıp fonksiyonu kullanılarak hata hesaplanır (Ronneberger vd., 2015). Hata hesaplandıktan sonra, gradyanlar geriye doğru yayılır ve ağırlıklar güncellenir (Ronneberger vd., 2015). Bu adımlar, modelin öğrenme sürecini ve hata azaltma yeteneğini

uygulamasını sağlamaktadır. Eğitim süreci ilerledikçe, modelin giderek daha iyi sonuçlar üretmesi beklenmektedir.

İlk olarak karot sandığı içerisindeki karot numunesinin tespiti için oluşturulan modelin eğitimi, yukarıda verilen parametrelerle başlatılmıştır. Eğitim gerçekleşirken eğitim süresi geçtikçe ekran kartı sıcaklığı yaklaşık 87°C'yi bulmuştur. Donanım kısıtı, zaman ve döngü sayısına göre doğruluk (accuracy) grafiğinden hareketle modelin eğitimi yaklaşık 72 saat sürmüştür. Buna göre;

- Eğitim boyunca modelin ulaştığı en yüksek doğruluk değeri %98.53 olarak hesaplanmıştır.
- Eğitim boyunca modelin ulaştığı en yüksek zar (dice) değeri %95.7 olarak hesaplanmıştır.
- Şekil 4.11'den hareketle, model yerel optimum noktalardan kurtulma kabiliyetine sahiptir.
- Şekil 4.11'den hareketle, ortalama 1200'üncü döngüden sonra doğruluk oranında çok fazla değişim görülmemiştir.
- En yüksek doğruluk yüzdesine sahip adımda hata fonksiyonu değeri 0.015 olarak hesaplanmıştır.

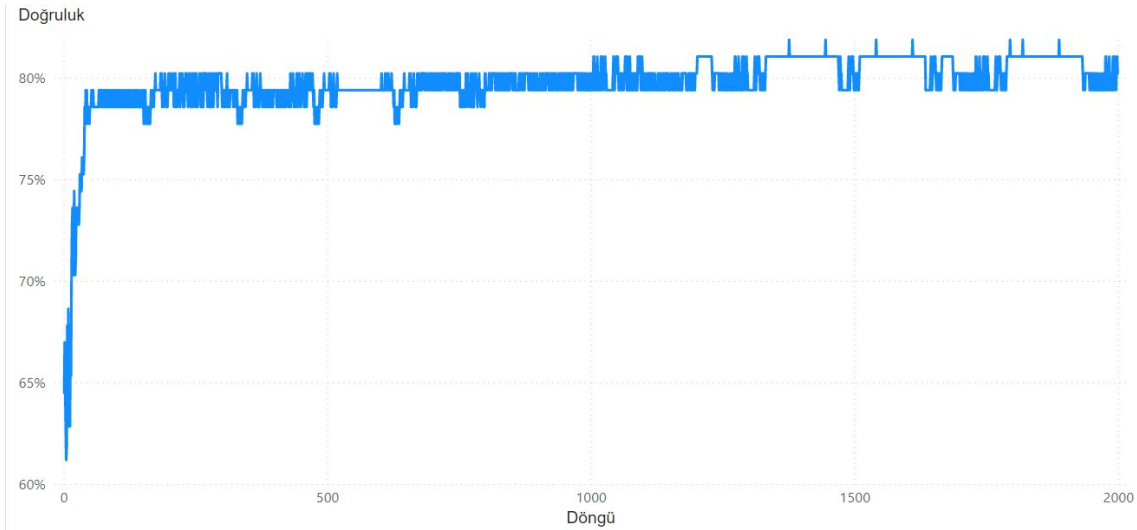


Şekil 4.11. Döngü sayısına göre doğruluk oranının değişimi.

Sonraki adımda karot numunesi içerisindeki cevherli bölgelerin tespiti için oluşturulan modelin eğitimi, yukarıda verilen parametrelerle başlatılmıştır. Eğitim gerçekleşirken eğitim süresi geçtikçe ekran kartı sıcaklığı yaklaşık 84°C'yi bulmuştur. Donanım kısıtı, zaman ve döngü sayısına göre doğruluk grafiğinden hareketle modelin eğitimi yaklaşık 80 saat sürmüştür. Buna göre;

- Eğitim boyunca modelin ulařmış olduđu en yüksek dođruluk deđeri %81.86 olarak hesaplanmıřtır.
- Eğitim boyunca modelin ulařmış olduđu en yüksek zar deđeri %73.70 olarak hesaplanmıřtır.
- Őekil 4.12'den hareketle, model yerel optimum noktalardan kurtulma kabiliyetine sahiptir.
- Őekil 4.12'den hareketle, ortalama 1400'üncü döngüden sonra döngülerin birbirini benzer periyotta tekrar etmeye bařladıđı görülmüřtür.
- En yüksek dođruluk yüzdesine sahip adımda hata fonksiyonu deđeri 0.102 olarak hesaplanmıřtır.

Eđitim süreçlerinin sonunda karot sandıđı içerisindeki karot numunesini tespit eden ve karot numunesi içerisindeki cevherli bölgeleri tespit eden 2 model elde edilmiřtir.

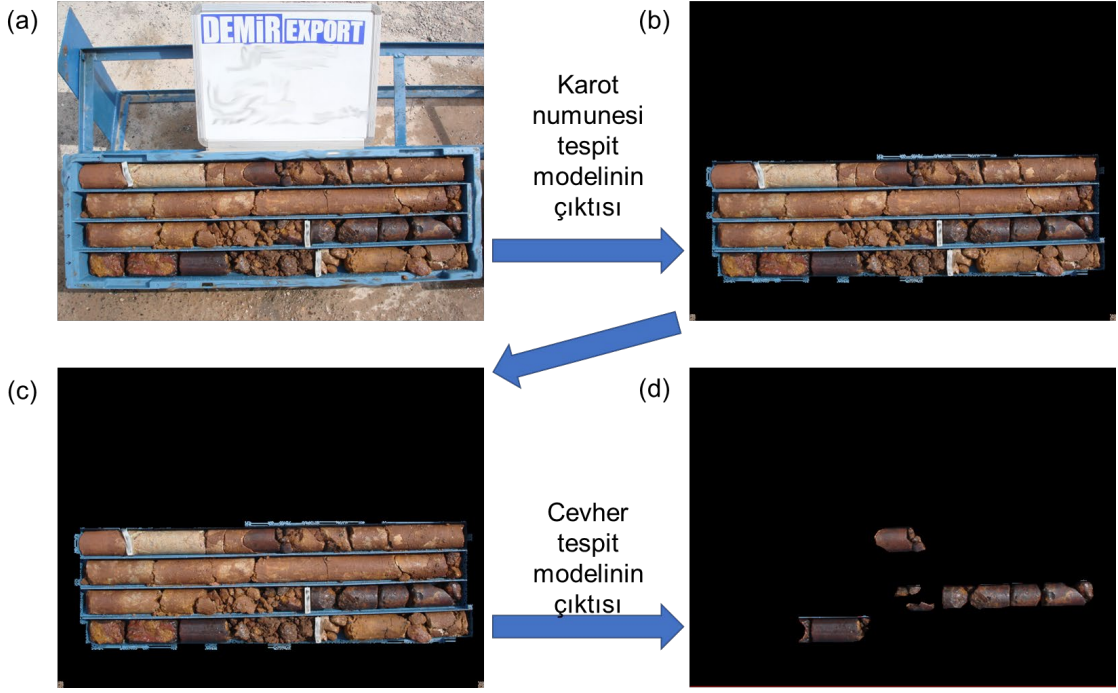


Şekil 4.12. Döngü sayısına göre dođruluk oranının deđişimi.

#### 4.6. Modellerin Entegrasyonu ve Uygulama

Oluřturulan modellerin cevher tespitinde kullanılabilir olması için birbirlerine entegre bir şekilde çalışması ve çıktı üretmesi gerekmektedir. Şekil 4.13'de modeller arasındaki iliřki gösterilmektedir. Karot numunelerini tespit eden modelin çıktısı, cevher tespitini gerçekleřtiren modelin girdisi olmaktadır.

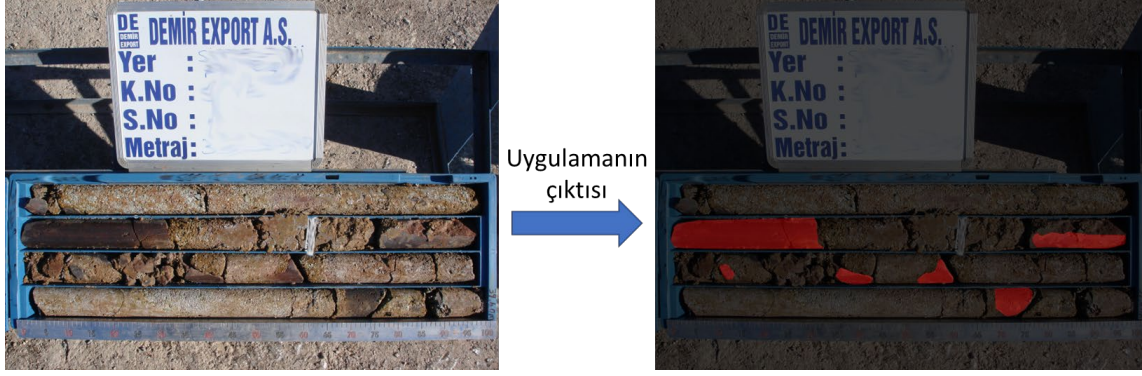




Şekil 4.13. Geliştirilen modeller arası ilişki.

İlk olarak modeller ayrı ayrı olarak çalıştırılmıştır. Önce karot numunelerinin tespiti gerçekleştirilip kaydedilmiştir. Sonrasında ise kaydedilen görüntüler üzerinden cevher tespiti gerçekleştirilip kaydedilmiştir. Cevherli bölge tespiti yapılırken aynı zamanda cevherli bölgenin numuneye oranı hesaplanarak, karot numunesinin cevherli bölge oranı (tenör) da kaydedilmiştir. Tenör hesabını gerçekleştirmek için PyTorch kütüphanesi kullanılmıştır. Örneğin Şekil 39 (d)'de hesaplanan tenör %13.25'dir.

Karot numunesi ve cevher tespitinin, iki ayrı süreç olarak pratikte kullanılması çok zahmetlidir. Bu iki ayrı süreci birleştirmek üzere Python programlama dili ile bir uygulama geliştirilmiştir. Geliştirilen uygulama sayesinde karot numunesinin ve cevherin tespiti için sadece bir fotoğrafın girdi olarak sağlanması yeterli olmuştur. Uygulama otomatik olarak karot numunesi tespit ettikten sonra, Şekil 4.14'deki gibi numune üzerindeki cevherli bölgelerin tespitini ve tenör hesabını gerçekleştirip kaydetmektedir. Tenör hesabını gerçekleştirmek için PyTorch kütüphanesi kullanılmıştır.



Şekil 4.14. Uygulama çıktısı örneği.

Geliştirilen uygulama, iki ayrı modeli birbirine entegre etmiş olsa da görüntüler manuel olarak bir yerde depolanıp uygulamaya girdi olarak verilmek zorundadır. Sondaj işlemleri veya incelemeler sırasında hızlıca görüntüleri analiz etmek çok zaman almaktadır.

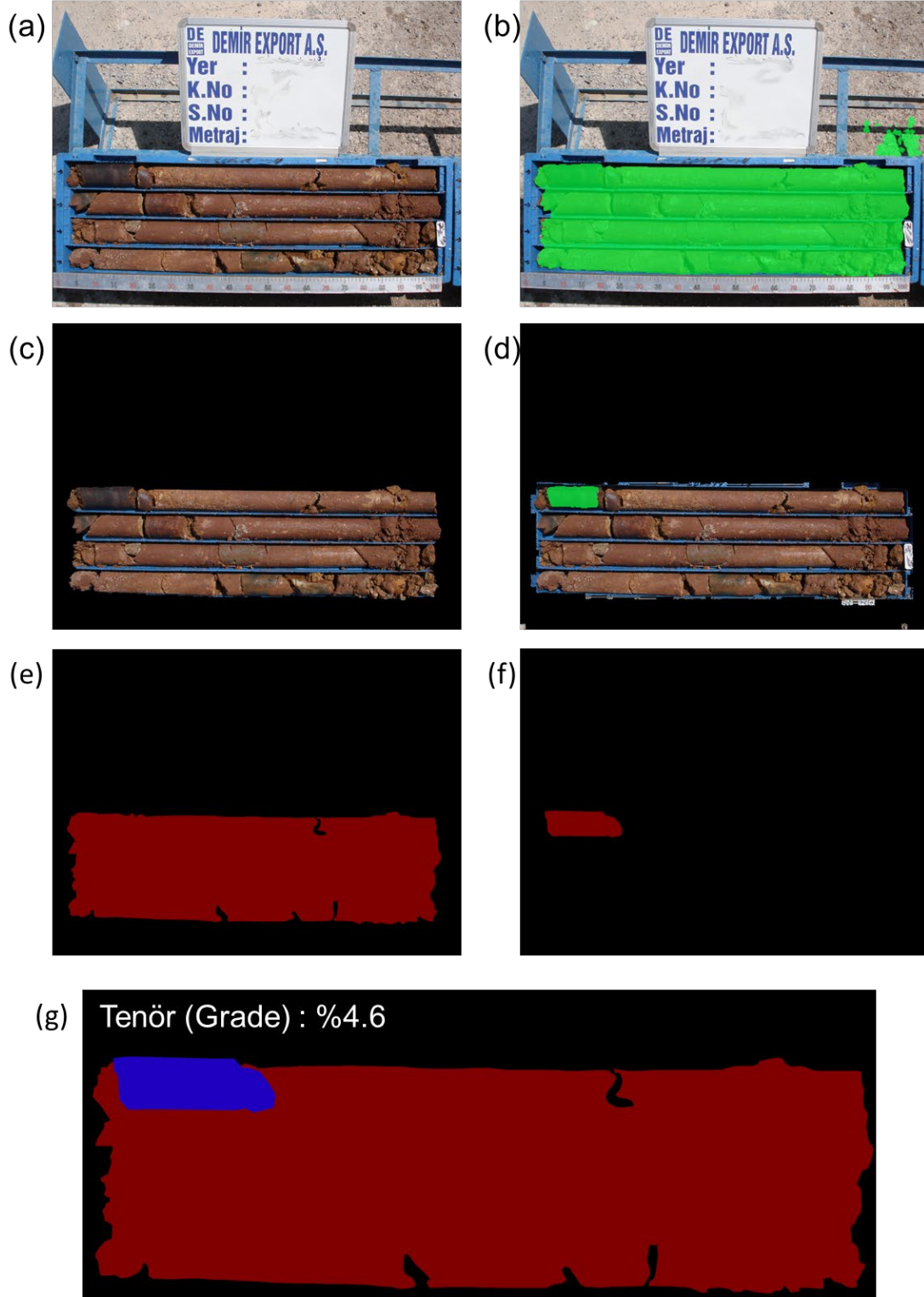
Gelişen teknolojiler ve ihtiyaçlardan dolayı, bazı durumlarda süreci hızlandırmak için gerçek zamanlı (real-time) veya gerçek zamanlıya yakın (near real-time) sonuçlar elde edilmek istenmektedir. Bundan hareketle uygulama girdi olarak görüntüleri gerçek zamanlı okuyacak, işleyecek ve sonuç üretecek şekilde tekrar geliştirilmiştir. Burada gerçek zamanlı görüntüleri okuyabilmek için OpenCV kütüphanesi kullanılmıştır. Okunan görüntüleri, oluşturulan modellerin işleyebileceği ve çıktı üretebileceği formata çevirmek için Pillow ve PyTorch kütüphaneleri kullanılmıştır. Sonrasında model çıktılarını gerçek zamanlı sunabilmek için OpenCV kütüphanesi kullanılmıştır. Tenör hesabını gerçekleştirmek için PyTorch kütüphanesi kullanılmıştır. Ayrıca uygulama gerçek zamanlı görüntü üzerinden sadece gösterim değil görüntü kaydı da gerçekleştirip depolayacak şekilde geliştirilmiş ve bu işlemleri gerçekleştirmek için Pillow kütüphanesi kullanılmıştır.

Şekil 4.15'te uygulamanın üretebildiği gerçek zamanlı çıktıların örnekleri bulunmaktadır. Bu görüntülerin detayları aşağıda verilmiştir.

- Şekil 4.15 (a) görüntünün orijinal halini göstermektedir.
- Şekil 4.15 (b) orijinal görüntü üzerinde, sandık içerisindeki karot numunesinin tahminini (yeşil renk) göstermektedir.
- Şekil 4.15 (c) karot numunesinin tahminini göstermektedir.

- Şekil 4.15 (d) karot numunesi üzerindeki cevherli bölgeyi(yeşil renk) göstermektedir.
- Şekil 4.15 (e) karot numunesini ayırt etmek için tekil renkler şeklinde verilmiştir. Kırmızı alan karot numunesini ifade etmektedir. Siyah alan ise görüntüden atılan (silinen) kısımları ifade etmektedir.
- Şekil 4.15 (f) görüntüdeki cevherli bölgeyi ayırt etmek için tekil renkler şeklinde verilmiştir. Kırmızı alan cevherli bölgeyi ifade etmektedir. Siyah alan ise görüntüden atılan (silinen) kısımları ifade etmektedir.
- Şekil 4.15 (g) karot numunesi üzerindeki cevherli bölgeyi ayırt etmek için tekil renkler şeklinde verilmiştir. Kırmızı alan karot numunesini, mavi alan ise cevherli bölgeyi ifade etmektedir. Mavi alanın piksel büyüklüğünün, kırmızı alanın piksel büyüklüğüne bölümünden tenör yüzdesi hesaplanmıştır.

Gerek görüldüğü takdirde, Şekil 4.15’de verilen uygulama çıktılarından sadece istenilen görüntüler gerçek zamanlı olarak takip edilebilmektedir. Örneği uygulamadan sadece Şekil 4.15 (g) çıktısı görüntülenebilir, diğer çıktılar gizlenebilir. Bu özellik uygulamadaki fazla çıktıların yol açabileceği karışıklığın azalmasına ve uygulama performansının da artmasına katkıda bulunacaktır.



Şekil 4.15. Uygulamanın gerçek zamanlı çıktı örnekleri.

Yapılan geliştirmeler sonucunda uygulama, pratik olması ve ek donanımlar gerektirmemesi için basit bir bilgisayar kamerası (webcam) yardımı ile görüntüleri okuyabilecek ve çıktı üretilebilecek duruma getirilmiştir. Şekil 4.16'da basit bir

bilgisayar kamerası ile uygulamanın açık havada yapılan saha testine ait bir düzenek görünmektedir. Farklı karot numuneleri üzerinde, numuneler kuru ve ıslak olacak şekilde testler gerçekleştirilmiştir.



Şekil 4.16. Geliştirilen uygulamanın saha çalışmalarına ait bir örnek.

## 5. SONUÇLAR VE TARTIŞMA

Madencilik sektörü, kaynakların etkin bir şekilde kullanılması ve verimli çıkarılması için sürekli olarak yenilikçi yaklaşımlara ihtiyaç duymaktadır. Bu tez, evrimsel sinir ağları kullanarak sondaj karot sandıklarının incelenmesi ve tenör tahmininin yapılması konusunda yeni bir yaklaşım sunmaktadır. Ülkemiz madencilik endüstrisinde daha doğru ve verimli bir kaynak tahmini yapmanın önemli bir adımı olacağı düşünülmektedir.

Tezde kullanılan evrimsel sinir ağları, veri analitiği ve makine öğrenimi tekniklerini birleştiren güçlü bir araçtır. Bu çalışma, veri analitiği yöntemlerini kullanarak sondaj karot verilerini işlemekte, karotların içerdiği bilgileri anlamaya yardımcı olmakta ve bunları tenör tahmini için kullanmaktadır. Bu entegrasyonun, madencilik endüstrisinde verimlilik artışı ve karar verme süreçlerinde iyileştirme sağlayacağı düşünülmektedir.

Karot sandıklarının doğru bir şekilde incelenmesi ve tenörün doğru bir şekilde tahmin edilmesi, madencilik operasyonlarının verimliliği açısından kritik bir öneme sahiptir. Bu tez, evrimsel sinir ağlarının kullanımıyla daha hassas ve doğru tenör tahminleri yapmanın mümkün olduğunu göstermiştir. Bu da madencilik şirketlerinin kaynak tahmini ve planlama süreçlerini iyileştirerek verimliliklerini artırmalarına yardımcı olacaktır.

Sondaj karot verilerinin etkin bir şekilde analiz edilmesi, madencilik faaliyetlerinin sürdürülebilirliği açısından önemlidir. Bu çalışma, doğru tenör tahmini ve kaynak tahminlerinin yapılmasıyla madencilik şirketlerinin kaynakların daha sürdürülebilir bir şekilde yönetilmesine ve çevresel etkilerin azaltılmasına katkı sağlamaktadır.

Çalışmada geliştirilen uygulama sayesinde yenilenemez kaynakların daha verimli kullanımını ve maden atıklarının asgari düzeye düşürülmesi hedeflenmektedir. Yapılan sondajlardan ne kadar çok bilgi edinilebilirse, aynı bölgedeki sondaj tekrarlarına o kadar az ihtiyaç duyulacaktır. Bu etki türünün özellikle sürdürülebilir çevre kriterine katkı sağlayacağı öngörülmektedir.

Madencilik sektöründe inovasyon ve rekabet önemli bir rol oynamaktadır. Evrimsel sinir ağları gibi ileri makine öğrenimi tekniklerini kullanmak, madencilik şirketlerine diğerlerinden öne çıkmak ve daha iyi sonuçlar elde etmek için bir fırsat

sunmaktadır. Bu çalışma, madencilik endüstrisinde teknolojik ilerlemeyi teşvik etmekte ve şirketlerin rekabet avantajı elde etmesini amaçlamaktadır.

Tez kapsamında geliştirilen uygulama sondaj karot verilerinin otomatik olarak işlenmesini ve tenör tahminlerinin yapılmasını sağlamaktadır. Bu, insan hatasını ve işgücü maliyetlerini azaltırken aynı zamanda hızlı ve doğru sonuçlar elde edilmesini sağlamaktadır. Bu otomatikleştirme, madencilik sektöründe veri analitiği ve tahminleme süreçlerini optimize etmek için büyük bir potansiyele sahiptir. Geliştirilen uygulama, özellikle ülkemiz madencilik sektöründe, iş süreçlerine dahil edildikçe sadece maden arama faaliyetleri değil, üretim, planlama gibi diğer madencilik faaliyetlerine de öncü olması beklenmektedir.

Bu tezde geliştirilen uygulama ve bulgular, madencilik sektörünün yanı sıra jeoloji, jeofizik, doğal kaynaklar yönetimi ve diğer ilgili alanlarda, uygulamanın geliştirilmesi durumunda potansiyel uygulama alanlarına sahiptir. Benzer veri analitiği ve tahminleme problemlerinin çözümünde evrimsel sinir ağlarının kullanımını teşvik edip ve bu alanda daha fazla araştırma ve geliştirme fırsatları sunacaktır. Çalışmada kullanılan görüntülerin tedarik edildiği Demir Export A.Ş. bu çalışmayı kısmen iş planlarına dahil etmeye ve kullanmaya başlamıştır.

Bu tez çalışmasında kullanılan veriler Demir Export A.Ş. tarafından sağlanmıştır. Görüntülere yalnızca bu tez çalışmasında kullanılmak üzere izin verilmiştir. Görüntüler Demir Export A.Ş.'ye ait Sivas'ın Divriği ilçesinde bulunan Demir Export Divriği Demir Madeni işletmesine aittir. Görüntüler, bu bölgede yapılan sondajlara ait görüntüler içerisinde seçilmiştir. Çalışmada kullanılmak üzere 250 adet görüntü alınmıştır. Alan uzmanları tarafından ilk aşamada görüntülerdeki karot numuneleri etiketlenmiştir. Sonrasında aynı alan uzmanları tarafından aynı görüntülerinde bulunan karot numuneleri üzerindeki cevherler etiketlenmiştir.

Karot sandığı içerisinde numune tespitini gerçekleştiren U-Net modeli, test verileri üzerinde %98.53 doğruluk oranını yakalamıştır, bu da modelin girdi verilerini doğru bir şekilde sınıflandırma yeteneğini göstermektedir. Ayrıca zar skorunda yakalanan %95.7 oranı, modelin segmentasyon başarısını göstermektedir. Şekil 37'den hareketle, modelin yerel optimum noktalardan kurtulma kabiliyetine sahip olduğu söylenebilir. Kurulan bu model tek başına ayrı bir uygulama gibi kullanıma potansiyeline sahiptir. Bu model sayesinde görüntü üzerindeki gereksiz

alanlardan kurtulup, görüntülerin temizlenmiş bir şekilde saklanması sağlanabilmektedir.

Karot numunesi üzerindeki cevherli bölgenin tespitini gerçekleştiren U-Net modeli, test verileri üzerinde %81.86 doğruluk oranını yakalamıştır, karot numune tespit modeli kadar başarılı olmasa da, modelin girdi verilerini doğru bir şekilde sınıflandırma yeteneğini göstermektedir.

Uygulama basit bir kamera yardımıyla karotların görüntülenmesi sureti ile çalışmaktadır. Diğer tüm işlemler arka planda otomatik olarak gerçekleştirilmekte ve herhangi bir müdahale gerektirmemektedir. Ayrıca uygulama çıktılarından sadece istenilen çıktılar görüntülenirken diğerleri kapatılabilmektedir.

Gelecek çalışmalarda daha iyi özelliklere sahip bilgisayar, iş istasyonu veya sunucu kullanarak modeller daha uzun eğitime tabi tutulabilir. Ayrıca daha güçlü donanımlarla çalışılması durumunda, daha yüksek boyutlara sahip görüntülerle çalışılabilir. Daha yüksek boyutlu görüntüler, görüntüdeki nesnelere daha detaylı özelliklerini barındıracağından daha doğru sonuçların üretileceği düşünülmektedir.

Cevherli bölge tespitini gerçekleştiren modelin ulaştığı doğruluk oranı artırılabilir. Bunun için daha fazla etiketli görüntü ile çalışılabilir, model daha uzun eğitim sürecine tabi tutulabilir.

Uygulama içerisindeki modellere yeterli sayıda etiketli görüntü örnekleri sağlandığında ve gerekli parametre optimizasyonları yapıldığı takdirde demir madeni dışında diğer madenlere de uygulanabilecektir. Örneğin, altın madenciliği, bakır madenciliği, kurşun-çinko madenciliği gibi diğer madencilik sektörlerinde de kendine yer edinebileceği öngörülmektedir.



## 6. KAYNAKLAR

- Acosta, I. C. C., Khodadadzadeh, M., Tolosana-Delgado, R., & Gloaguen, R. (2020). Drill-core hyperspectral and geochemical data integration in a superpixel-based machine learning framework. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 13, 4214-4228. <https://doi.org/10.1109/JSTARS.2020.3011221>
- Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1), 53. <https://doi.org/10.1186/s40537-021-00444-8>
- Arbib, M. A. (1987). *Brains, Machines, and Mathematics*. Springer US. <https://doi.org/10.1007/978-1-4612-4782-1>
- Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford University Press, USA.
- Bishop, C. M. (2007). *Pattern Recognition and Machine Learning (Information Science and Statistics)* (1. bs). Springer. <http://www.amazon.com/Pattern-Recognition-Learning-Information-Statistics/dp/0387310738%3FSubscriptionId%3D13CT5CVB80YFWJEPWS02%26tag%3Dws%26linkCode%3Dxm2%26camp%3D2025%26creative%3D165953%26creativeASIN%3D0387310738>
- Bradshaw, D. (2014, Mart). The role of “process mineralogy” in improving the process performance of complex sulphide ores. *IMPC 2014 - 27th International Mineral Processing Congress*.
- Bradski, G. (2000). The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.
- Bueno, M., Foggatto, B., & Lane, G. (2015, Mart). *Geometallurgy Applied in Comminution to Minimize Design Risks*.
- Chollet, F. (2017). *Deep Learning with Python* . Manning.

- Cirean, D., Giusti, A., Gambardella, L. M., & Schmidhuber. (2012). Deep Neural Networks Segment Neuronal Membranes in Electron Microscopy Images. *Proceedings of Neural Information Processing Systems*, 25.
- Cosido, O., Iglesias, A., Galvez, A., Catuogno, R., Campi, M., Terán, L., & Sainz, E. (2014). Hybridization of convergent photogrammetry, computer vision, and artificial intelligence for digital documentation of cultural heritage-a case study: the magdalena palace. *2014 International Conference on Cyberworlds*.
- da Silva, E., & Mendonca, G. (2005). *Digital Image Processing* (ss. 891-910). <https://doi.org/10.1016/B978-012170960-0/50064-5>
- Dastres, R., & Soori, M. (2021). Advanced Image Processing Systems. *International Journal of Imaging and Robotics*, 21.
- Duan, K. (2022). Center Net. İçinde *GitHub repository*. GitHub.
- Fandrich, R., Gu, Y., Burrows, D., & Moeller, K. (2007). Modern SEM-based mineral liberation analysis. *International Journal of Mineral Processing*, 84(1), 310-320. <https://doi.org/https://doi.org/10.1016/j.minpro.2006.07.018>
- Gandhi, S. M., & Sarkar, B. C. (2016). Essentials of Mineral Exploration and Evaluation. İçinde *Drilling* (1st bs, ss. 199-234). Elsevier. <https://doi.org/10.1016/C2015-0-04648-2>
- Gonzalez, R. C., & Woods, R. E. (2008). *Digital image processing*. Prentice Hall. <http://www.amazon.com/Digital-Image-Processing-3rd-Edition/dp/013168728X>
- Goodfellow, I. J., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.
- Gottlieb, P. (2008, Mart). *The Revolutionary Impact of Automated Mineralogy on Mining and Mineral Processing*.
- Haldar, S. K. (2013). *Mineral Exploration: Principles and Applications*.
- Hapugoda, P., Krahenbuhl, G., O'Brien, G., & Warren, K. (2015, Mart). *A New Method for Obtaining Detailed Mineral Information on Individual Coal Particles at the size that they are used in Coke Making*.

- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., David Cournapeau, Wieser, E., Taylor, J., Sebastian Berg, Smith, N. J., Kern, R., Hoyer, M. P. and S., van Kerkwijk, M. H., Matthew Brett, Haldane, A., del Río, J. F., Wiebe, M., Peterson, P., Pierre Gérard-Marchant, ... Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, 585(7825), 357-362. <https://doi.org/10.1038/s41586-020-2649-2>
- Haykin, S. S. (2009a). *Neural Networks and Learning Machines*. Prentice Hall/Pearson.
- Haykin, S. S. (2009b). *Neural networks and learning machines* (Third). Pearson Education.
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770-778. <https://doi.org/10.1109/CVPR.2016.90>
- Hu, Y. H., Hwang, J.-N., & Hwang, J.-N. (2000). *Handbook of Neural Network Signal Processing* (1st bs). CRC Press, Inc.
- Huang, S.-C., & Le, T.-H. (2021). *Principles\_and\_Labs\_for\_Deep\_Learning*.
- Hyder, Z., Siau, K., & Nah, F. (2019). Artificial Intelligence, Machine Learning, and Autonomous Technologies in Mining Industry. *Journal of Database Management*, 30(2), 67-79. <https://doi.org/10.4018/JDM.2019040104>
- Iosifidis, A., & Tefas, A. (2022). *Deep Learning for Robot Perception and Cognition*. Elsevier. <https://doi.org/10.1016/C2020-0-02902-6>
- Javaid, M., Haleem, A., Singh, R. P., & Suman, R. (2022). Artificial Intelligence Applications for Industry 4.0: A Literature-Based Study. *Journal of Industrial Integration and Management*, 07(01), 83-111. <https://doi.org/10.1142/S2424862221300040>
- Jiang, Z.-P., Liu, Y.-Y., Shao, Z.-E., & Huang, K.-W. (2021). An Improved VGG16 Model for Pneumonia Image Classification. *Applied Sciences*, 11(23). <https://doi.org/10.3390/app112311185>
- Kelleher, J. D. (2019). *Deep Learning*. The MIT Press.

- Khan, A., Sohail, A., Zahoora, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, 53(8), 5455-5516. <https://doi.org/10.1007/s10462-020-09825-6>
- Khan, S., Rahmani, H., Shah, S. A. A., & Bennamoun, M. (2018). A Guide to Convolutional Neural Networks for Computer Vision. *Synthesis Lectures on Computer Vision*, 8(1), 1-207. <https://doi.org/10.2200/S00822ED1V01Y201712COV015>
- Kollmannsberger, S., D'Angella, D., Jokeit, M., & Herrmann, L. (2021). *Deep Learning in Computational Mechanics* (C. 977). Springer International Publishing. <https://doi.org/10.1007/978-3-030-76587-3>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks. İçinde F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Ed.), *Advances in Neural Information Processing Systems 25* (ss. 1097-1105). Curran Associates, Inc. <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>
- Layton-Matthews, D., Hamilton, C., & McClenaghan, M. B. (2014). Mineral chemistry: modern techniques and applications to exploration. *Geological Survey of Canada, Open File*, 7553, 9-18.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436.
- Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324. <https://doi.org/10.1109/5.726791>
- Lecun, Y., Jackel, L. D., Bottou, L., Cortes, C., Denker, J. S., Drucker, H., Guyon, I., Muller, U. A., Sackinger, E., Simard, P., & Vapnik, V. (1995). Learning algorithms for classification: A comparison on handwritten digit recognition. İçinde J. H. Oh, C. Kwon, & S. Cho (Ed.), *Neural networks* (ss. 261-276). World Scientific.
- Long, J., Shelhamer, E., & Darrell, T. (2014). *Fully Convolutional Networks for Semantic Segmentation*.

- Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2012). *Foundations of Machine Learning*. The MIT Press.
- Mooney, P. (2022). *2022 Kaggle Machine Learning & Data Science Survey*. Kaggle. <https://kaggle.com/competitions/kaggle-survey-2022>
- Murphy, K. P. (2013). *Machine learning : a probabilistic perspective*. MIT Press. [https://www.amazon.com/Machine-Learning-Probabilistic-Perspective-Computation/dp/0262018020/ref=sr\\_1\\_2?ie=UTF8&qid=1336857747&sr=8-2](https://www.amazon.com/Machine-Learning-Probabilistic-Perspective-Computation/dp/0262018020/ref=sr_1_2?ie=UTF8&qid=1336857747&sr=8-2)
- Nelson, H. (2023). *Essential Math for AI: Next-Level Mathematics for Efficient and Successful AI Systems* (1. bs, C. 1). O'Reilly Media.
- Nikonow, W., & Rammlmair, D. (2017). Automated mineralogy based on micro-energy-dispersive X-ray fluorescence microscopy ( $\mu$ -EDXRF) applied to plutonic rock thin sections in comparison to a mineral liberation analyzer. *Geoscientific Instrumentation, Methods and Data Systems*, 6(2), 429-437. <https://doi.org/10.5194/gi-6-429-2017>
- NVIDIA, Vingelmann, P., & Fitzek, F. H. P. (2020). *CUDA, release: 10.2.89*. <https://developer.nvidia.com/cuda-toolkit>
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., ... Chintala, S. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. İçinde *Advances in Neural Information Processing Systems* 32 (ss. 8024-8035). Curran Associates, Inc. <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf>
- Ro, S.-H., & Kim, S.-H. (2021). An image stitching algorithm for the mineralogical analysis. *Minerals Engineering*, 169, 106968. <https://doi.org/https://doi.org/10.1016/j.mineng.2021.106968>
- Ronneberger, O., Fischer, P., & Brox, T. (2015). *U-Net: Convolutional Networks for Biomedical Image Segmentation*.
- Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65 6, 386-408.

- Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, 323(6088), 533-536.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., & Fei-Fei, L. (2015). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3), 211-252. <https://doi.org/10.1007/s11263-015-0816-y>
- Shanmuganathan, S., & Samarasinghe, S. (2016). *Artificial Neural Network Modelling* (S. Shanmuganathan & S. Samarasinghe, Ed.; 1. bs). Springer International Publishing.
- Shepherd, G. M. (1990). *The Synoptic Organization of the Brain*. Oxford University Press.
- Shrestha, A., & Mahmood, A. (2019). Review of Deep Learning Algorithms and Architectures. *IEEE Access*, 7, 53040-53065. <https://doi.org/10.1109/ACCESS.2019.2912200>
- Simonyan, K., & Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv 1409.1556*.
- Sonka, M., Hlavac, V., & Boyle, R. (2007). *Image Processing, Analysis, and Machine Vision*. Thomson-Engineering.
- S&P Global Market Intelligence. (2019). *PDAC Special Edition: World Exploration Trends 2018*.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1-9. <https://doi.org/10.1109/CVPR.2015.7298594>
- Teoh, T. T. (2023). *Convolutional Neural Networks for Medical Applications*. Springer Nature Singapore. <https://doi.org/10.1007/978-981-19-8814-1>
- Tyagi, V. (2018a). *Understanding Digital Image Processing*. <https://doi.org/10.1201/9781315123905>

- Tyagi, V. (2018b). *Understanding Digital Image Processing*. CRC Press.  
<https://doi.org/10.1201/9781315123905>
- Umesh, P. (2012). Image Processing in Python. *CSI Communications*, 23.
- Van Rossum, G., & Drake, F. L. (2009). *Python 3 Reference Manual*. CreateSpace.
- Zhu, X. (2005). *Semi-Supervised Learning Literature Survey* (Sayı 1530).  
[http://pages.cs.wisc.edu/~jerryzhupubssl\\_survey.pdf](http://pages.cs.wisc.edu/~jerryzhupubssl_survey.pdf)