



**T.C.
KONYA TEKNİK ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ**

**ÜRETKEN ÇEKİŞMELİ AĞ VE UNET KULLANILARAK
SEGMENTE EDİLMİŞ TOMOGRAFİ GÖRÜNTÜLERDEN
COVID-19 SINIFLANDIRMASINDA FARKLI DERİN
ÖĞRENME MİMARİLERİNİN KULLANIMI**

**Kieleh NGONG IVOLINE CLARISSE
YÜKSEK LİSANS TEZİ
Bilgisayar Mühendisliği Anabilim Dalı**

**Temmuz – 2021
KONYA
Her Hakkı Saklıdır**

TEZ KABUL VE ONAYI

Kieleh Ngong Ivoline Clarisse tarafından hazırlanan “Üretken Çekişmeli Ağ ve UNet kullanılarak segmente edilmiş tomografi görüntülerden Covid-19 sınıflandırmasında farklı derin öğrenme mimarilerinin kullanımı” adlı tez çalışması 08./07./2021 tarihinde aşağıdaki jüri tarafından OY BİRLİĞİ ile Konya Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı’nda YÜKSEK LİSANS TEZİ olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Başkan

Doç. Dr. Hasan Erdinç KOÇER

.....

Danışman

Dr. Öğr. Üyesi Nurdan BAYKAN

.....

Üye

Doç. Dr. Mesut GÜNDÜZ

.....

Yukarıdaki sonucu onaylarım.

Prof. Dr. Saadettin Erhan KESEN
Enstitü Müdürü

TEZ BİLDİRİMİ

Bu tezdeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edildiğini ve tez yazım kurallarına uygun olarak hazırlanan bu çalışmada bana ait olmayan her türlü ifade ve bilginin kaynağına eksiksiz atıf yapıldığını bildiririm.

DECLARATION PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

İmza:

Kieleh NGONG IVOLINE CLARISSE
Tarih: Temmuz – 2021

ÖZET
YÜKSEK LİSANS TEZİ
ÜRETKEN ÇEKİŞMELİ AĞ VE UNET KULLANILARAK
SEGMENTE EDİLMİŞ TOMOGRAFİ GÖRÜNTÜLERDEN
COVID-19 SINIFLANDIRMASINDA
FARKLI DERİN ÖĞRENME MİMARİLERİNİN KULLANIMI

Kieleh NGONG IVOLINE CLARISSE

Konya Teknik Üniversitesi
Lisansüstü Eğitim Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Nurdan BAYKAN

2021, 69 Sayfa

Jüri
Doç. Dr. Hasan Erdiç KOÇER
Doç. Dr. Mesut GÜNDÜZ
Dr. Öğr. Üyesi Nurdan BAYKAN

2019 yılının sonlarında ortaya çıkan yeni koronavirüs (Covid-19), 7 kıtanın tümünde büyük bir küresel sağlık krizine neden olmuştur. Bu virüsle mücadele etme yolunda atılan önemi adımlardan biri etkili tarama, erken teşhis ve enfekte kişilerin izolasyonu yoluyla hastalığın yayılmasını sınırlamaktır. Bilgisayarlı tomografi (BT) taramaları tanıyı belirlemede etkili bir yöntem haline gelirken, BT'nin radyolojik tetkikleri radyologların iş yükünün büyük ölçüde artmasına neden olmuştur. BT görüntülerinden Covid-19'un tespitinde Bilgisayar Destekli Tanı (BDT) sistemlerinin kullanılması radyologların yükünü azaltmakta ve tanı verimliliğini artırmaktadır. Bu çalışmada, Covid-19 enfekte bölgeleri, göğüs BT görüntülerinde otomatik olarak segmentlere ayıran UNet tabanlı ve Koşullu Üretken Çekişmeli Ağa (kÜÇA) dayalı yeni bir sistem önerilmiştir. Segmentasyon amacıyla, üretici (generator), enfekte olmuş bölgelerin etrafındaki sınırları belirleyen ikili maskeler oluşturmayı öğrenirken, ayırıcı (discriminator) ise gerçek ve sentetik maskeleri ayırt etmeyi öğrenir. Dolayısıyla bu çekişmeli eğitim, üretici çok gerçekçi ikili maskeler oluşturarak virüslü bölgeleri tespit etmeye zorlamaktadır. DeneySEL sonuçlar, önerilen yöntemin %92.32 zar skoru (dice score) ve %86.41 Birlik üzerinde Kesişim (BüK) (IoU) skoru ile segmentasyon başarısı sağladığını göstermektedir. Ayrıca, tez kapsamında, önerilen yöntemle üretilen maskeleri kullanarak Covid-19 hastalarını sınıflandırmak için Evrişimsel Sinir Ağı (ESA), Patch Evrişimsel Sinir Ağı (PatchESA) ve Kapsül Sinir Ağı (KapsülSA) olmak üzere üç ayrı sınıflandırıcı kullanılmıştır. Bu sınıflandırıcıların başarısı sırasıyla %99.20, %92.30 ve %73.83'tür. Bu sonuçlara göre en yüksek sınıflandırma başarısı kÜÇA_Unet ve ESA'nın birlikte kullanıldığı sistemde elde edilmiştir.

Anahtar kelimeler: Covid-19 Segmentasyonu, Covid-19 Sınıflandırması, Bilgisayarlı tomografi, Çekişmeli Üretken Ağ, Evrişimsel Sinir Ağı, Patch Evrişimsel Sinir Ağı, Kapsül Sinir Ağı

ABSTRACT

MS THESIS

USING DIFFERENT DEEP LEARNING METHODS FOR COVID-19 CLASSIFICATION FROM CT SCANS SEGMENTED BY GENERATIVE ADVERSARIAL NETWORKS AND UNET

Kieleh NGONG IVOLINE CLARISSE

**Konya Technical University
Institute of Graduate Studies
Department of Computer Engineering**

Advisor: Asst. Prof. Dr. Nurdan BAYKAN

2021, 69 Page

Jury

**Assoc. Prof. Dr. Hasan Erdiñ KOÇER
Assoc. Prof. Dr. Mesut GÜNDÜZ
Asst. Prof. Dr. Nurdan BAYKAN**

The new coronavirus, which emerged at the end of 2019, caused a major global health crisis in 7 continents. An essential step towards fighting this virus is limiting the spread of the disease through effective screening, early detection and isolation of infected persons. While computed tomography (CT) scans have become an effective method to detecting the diagnosis, radiological examination of CT has led to a large increase in the workload for radiologists. Computer Aided Detection (CAD) systems for Covid-19 from CT images can therefore reduce the burden on radiologists and improve efficiency of diagnosis. In this study, a new system based on UNet-based Conditional Generative Adversarial Network (cGAN) that automatically segments infected regions from chest CT slices is proposed. For the purpose of segmentation, the generator learns to create binary masks that outline boundaries around these infected regions while the discriminator learns to distinguish between the real and synthetic masks. This adversarial training therefore forces the generator to detect infected regions by creating very realistic binary masks. Experimental results show that the proposed method achieves a segmentation success with a dice score of 92.32% and IoU score of 86.41%. Furthermore, 3 classifiers which include a Convolutional Neural Network (CNN), a PatchCNN and a Capsule Neural Network (CapsNet) are proposed to classify the generated masks as either Covid-19 or not. Success of these classifiers was 99.20%, 92.30% and 73.83%, respectively. According to these results, the highest success was achieved in the system where cGAN_Unet and CNN are used together.

Keywords: Covid-19 Segmentation, Covid-19 Classification, Generative Adversarial Network, Convolutional Neural Network, PatchCNN, Capsule Neural Network

ÖNSÖZ

Bu çalışmanın gerçekleştirilmesinde benimle değerli bilgilerini ve zamanını paylaştığı için danışmanım Dr. Öğr. Üyesi Nurdan BAYKAN’a müteşekkirim. Eğitim yolculuğum boyunca bana destek olan diğer tüm üniversite öğretmenlerime, aileme ve arkadaşlarıma da teşekkür ederim.

Kieleh NGONG IVOLINE CLARISSE

KONYA-2021

İÇİNDEKİLER

ÖZET	iv
ABSTRACT.....	v
ÖNSÖZ	vi
İÇİNDEKİLER.....	vii
SİMGELER VE KISALTMALAR.....	ix
1. GİRİŞ.....	1
1.1. Tezin Amacı	3
1.2. Tezin Önemi.....	4
1.3. Tezin Organizasyonu	5
2. KAYNAK ARAŞTIRMASI	6
2.1. Covid-19 Segmentasyonu Üzerine Çalışmalar	6
2.2. Covid-19 Sınıflandırması Üzerine Çalışmalar	8
3. MATERYAL VE YÖNTEM.....	11
3.1. Derin Öğrenme.....	11
3.1.1. UNet	18
3.1.2. UNet++	19
3.1.3. SegNet	20
3.1.4. PSPNet.....	21
3.1.5. Üretken Çekişmeli Ağ ve Koşullu Üretken Çekişmeli Ağ.....	22
3.1.6. Evrişimsel Sinir Ağı (ESA).....	25
3.1.7. Yama Evrişimsel Sinir Ağı (PatchESA)	26
3.1.8. Kapsül Sinir Ağları (KSA).....	27
3.2. Metrikler.....	29
3.3. Veri Setleri	32
4. GELİŞTİRİLEN YÖNTEM	34
4.1. Önışlem	35
4.2. Görüntü Segmentasyonu	36
4.2.1. Koşullu Üretken Çekişmeli Ağ ve UNet ile segmentasyon	38
4.3. Sınıflandırma.....	43
4.3.1. Evrişimsel Sinir Ağı ile Sınıflandırma	45
4.3.2. Patch Sinir Ağı ile Sınıflandırma	46
4.3.3. Kapsül Sinir Ağı ile Sınıflandırma	48

5. DENEYSEL SONUÇLAR VE TARTIŞMA	50
5.1. Segmentasyon Sonuçları	50
5.2. Sınıflandırma Sonuçları.....	52
6. SONUÇ VE ÖNERİLER.....	57
6.1. Sonuçlar	57
6.2. Öneriler	60
KAYNAKLAR	64



SİMGELER VE KISALTMALAR

Simgeler

x	: Veri girdileri
ω	: Ağırlıkları
y	: katman çıktısı
b	: Sapma değeri
n	: Girdi sayısı
G	: Üretici
D	: Ayırt edici
L_{BCE}	: İkili Çapraz Entropi Kaybı
L_{DICE}	: Dice Kaybı
L_{DIS}	: Ayırt Edici Kaybı
L_{GEN}	: Üretici Kaybı
L_{L1}	: L1 Kaybı
L_{CGAN}	: cGAN Kaybı

Kısaltmalar

Covid-19	: Koronavirüs
DSÖ (WHO)	: Dünya Sağlık Örgütü (World Health Organization)
PHEIC	: Uluslararası Önemi Haiz Halk Sağlığı Acil Durumu (Public Health Emergencies of International Concern)
RT-PCR	: Ters transkriptaz-polimeraz zincir reaksiyonu testi (Reverse Transcription-Polymerase Chain Reaction)
BT (CT)	: Bilgisayarlı tomografi (Computed Tomography)
MRI	: Manyetik rezonans görüntüleme (Magnetic Resonance Imaging)
CXR	: Göğüs röntgeni (Chest X-Ray)
YZ (AI)	: Yapay Zekâ (Artificial Intelligence)
İB (ROI)	: İlgi Bölgelerini (Region of Interest)
DÖ (DL)	: Derin Öğrenme (Deep Learning)
YSA (ANN)	: Yapay Sinir Ağları (Artificial Neural Network)
ESA (CNN)	: Evrişimsel Sinir Ağları (Convolutional Neural Network)
TEA (FCN)	: Tam Evrişimli Ağ (FCN-Fully Convolution Network)
ÖPA (FPN)	: Özellik Piramit Ağı (Feature Pyramid Network)
YOLO	: You Only Look Once
ReLU	: Rektifiye Edilmiş Doğrusal Birim (Rectified Linear Unit)
PatchESA (PatchCNN)	: Patch Evrişimsel Sinir Ağı (Patch Convolutional Neural Network)
KSA (CapsNet)	: Kapsül Sinir Ağı (CapsNet-Capsule Network)
ROC	: Alıcı İşletim Karakteristiği (Receiver Operating Characteristic)
KSAHE (CLAHE)	: Kontrast Sınırlı Adaptif Histogram Eşitleme

İÇE (BCE) (Contrast Limited Adaptive Histogram Equalization)
: İkili Çapraz Entropi (Binary-Cross Entropy)



Tez kapsamında sayılar ve sayı birimleri SI (The International System of Units – Fransızca Le Syst me International d’Unit s) sistemine uygun olarak verilmiřtir.,

1. GİRİŞ

2019 yılının sonunda, şiddetli akut solunum sendromu Koronavirüs 2'nin (SARS-CoV-2) neden olduğu yeni koronavirüs hastalığı (Covid-19) Çin'in Wuhan şehrinde ortaya çıkmıştır (Ceraolo ve Giorgi, 2020). Çin'den çıkan virüsün uluslararası sınırları aşmasının ardından Dünya Sağlık Örgütü (DSÖ) (WHO-World Health Organization), salgını 30 Ocak 2020 tarihinde Uluslararası Önemi Haiz Halk Sağlığı Acil Durumu (PHEIC- public health emergencies of international concern) olarak ve ardından 11 Mart 2020 tarihinde bir pandemi olarak ilan etmiştir (Cucinotta ve Vanelli 2020, WHO 2021). 5 Mayıs 2021 tarihi itibarıyla, pandeminin başlangıcından bu yana küresel olarak yaklaşık 155 milyon kümülatif vaka ve 3.25 milyon ölüm bildirilmiştir (WHO, 2021). Covid-19'un son derece bulaşıcı doğası göz önüne alındığında, Covid-19 ile mücadele etmeye yönelik en önemli adım; virüsün yayılmasını sınırlamak için enfekte kişilerin etkili bir şekilde taranması, erken tespit edilmesi, izolasyonu ve hızlı bir şekilde tedavisinin başlamasıdır.

Ters transkriptaz-polimeraz zincir reaksiyonu testi (RT-PCR-Reverse Transcription-Polymerase Chain Reaction), Covid-19'u tespit etmek amacıyla dünya çapında yaygın olarak kullanılmaktadır (Jawerth, 2020). Bu işlem, solunum örneklerinde hedeflenen genetik materyalleri tespit etmek için floresan boyalar kullanarak yapılmaktadır. RT-PCR testinin yüksek özgüllüğü, kıyaslama yöntemi olarak kullanılmasına izin vermektedir. Ancak, işlem oldukça kontrollü bir ortamda gerçekleştirilmesi gerektiğinden oldukça zahmetlidir. Ayrıca, test sonuçlarının alınması 4 saatten 1 güne kadar değişmektedir (Herper ve Branswell,2020) ve RT-PCR testi için gereken malzemeleri bulmada sıkıntı yaşanmaktadır (Won ve ark., 2020). Ek olarak, RT-PCR'nin duyarlılığına ilişkin ilk raporlar net veya tutarlı değildir ve Çin'den gelen ilk bulgular, RT-PCR'nin nispeten zayıf bir duyarlılık üretebileceğini göstermiştir (Wang ve ark., 2020). Covid-19 taraması için tamamlayıcı bir yöntem ise bilgisayarlı tomografi (BT) (CT-Computed Tomography), Manyetik rezonans görüntüleme (MRI-Magnetic Resonance Imaging) taramaları veya göğüs röntgeni (CXR-Chest X-Ray) taramaları gibi tıbbi görüntüler üzerinde radyografi incelemesinin kullanılmasıdır. Literatürde bazı klinik semptomlar ortaya çıkmadan önce bile hastaların BT görüntülerinde Covid-19'un gözlemlenebileceğini belirten çalışmalar bulunmaktadır (Lin ve ark., 2020; Wang ve ark.,

2020). BT görüntülerinde Covid-19'un tipik belirtileri arasında buzlu cam opasitesi, küçük bilateral plevral efüzyonlar, pulmoner nodüllerin varlığı, bronşektazi, yuvarlak kistik değişiklikler ve lenfadenopati bulunmaktadır (Lin ve ark., 2020; Meng ve ark., 2020). Bu morbid doku değişikliklerinin hızlı ve doğru tespiti, Covid-19 erken teşhisi için esastır. BT tarama testlerinin geleneksel RT-PCR testlerinden daha yüksek hassasiyet sağladığı literatürde gösterilmiş olsa da (Ai ve ark., 2020; Fang ve ark., 2020), çeşitli zorluklar devam etmektedir. Bu zorluklar arasında, iyi eğitilmiş radyolog sayısının azlığı ve pandemi nedeniyle yaşanan hasta akışının sebep olduğu iş yükündeki artış da yer almaktadır (Mossa-Basha ve ark., 2020). 5 klinik radyolog boşluğundan 3'ünün doldurulamadığı Birleşik Krallık, radyolog eksikliğine bir örnektir. Ayrıca, Amerika Birleşik Devletleri'nde 2025 yılına kadar on binlerce pozisyonu kapsayan ulusal bir radyolog sıkıntısı olacağı tahmin edilmektedir (Survivor 2019). Gelişmekte olan ülkelerde ise, hasta-doktor oranı çok düşüktür ve 55 Afrika ülkesinin tamamında 2,3 milyonluk bir boşluk bulunmaktadır (Stephen ve ark., 2019). Ayrıca Covid-19'un yeni bir hastalık olması nedeniyle, radyologların bilgilerini güncellemeleri ve yeni yorumlama becerileri edinmeleri gerekmektedir. Teşhis koyma, eğitim kaynaklarına erişimin sınırlı olduğu uzak bölgelerde daha da zorlaşmaktadır. Bu nedenle, Covid-19'un teşhisine yardımcı olacak Yapay Zekâ (YZ) (AI-Artificial Intelligence) sistemlerinin ivedilikle geliştirilmesi son derece önem kazanmaktadır.

Son yıllarda, tıbbi görüntülerin segmentasyonu ve sınıflandırılması için Derin Öğrenme (DÖ) (Deep Learning) modellerinin geliştirilmesinde hızlı bir artış olmuştur (Biswas ve ark., 2019). Tıbbi görüntü segmentasyonu; tıbbi görüntülerdeki anormallikleri otomatik olarak tanımlar veya cilt lezyonları (Arsalan ve ark., 2020), akciğerler (Park ve ark., 2019), tümörler (Mittal ve ark., 2019) vb. İlgi Bölgelerini (İB) (ROI-Region of Interest) etiketlerler. Böylece radyologlara tanı koyma, hastalığın ilerlemesini izleme, muayene süresinin azalması ve teşhis doğruluğunun gelişmesi gibi konularda yardımcı olabilirler (Müller ve ark., 2020). Benzer şekilde tıbbi görüntü sınıflandırması, tıbbi bir görüntüde belirli bir hastalığın olup olmadığının belirlenmesidir. Liu ve ark., tarafından yapılan sistematik bir inceleme, derin öğrenme modellerinin tanı performansının sağlık uzmanlarının performansına eşdeğer olabileceğini göstermiştir (Liu ve ark., 2019).

Bu tez çalışmasında, Covid-19 tespiti için öncelikle segmentasyon ve ardından sınıflandırma olmak üzere iki ana aşamaya dayalı derin öğrenme tabanlı bir sistem

önerilmektedir. İlk olarak, orijinal görüntülere bazı ön işleme teknikleri uygulanmış, daha sonra veri artırma yapılmış ve ardından ilgi bölgelerinin (İB) (ROI) segmentasyonu yapılmıştır. Daha sonra ise BT taramasını Covid-19 için pozitif veya negatif olarak sınıflandırmak amacıyla farklı ikili sınıflandırıcılar kullanılmıştır. Bu çalışmada Unet, Unet++, Segnet, PSPNet olarak farklı DÖ yöntemlerin kullanılmasının yanı sıra; bu mimarilerin Koşullu Üretken Çekişmeli Ağ (kÜÇA) (cGAN-Conditional Generative Adversarial Networks) ile hibrit kullanılmasıyla yapılan segmentasyon sonucunda elde edilen segmentlenmiş görüntü ile Covid-19 tespiti, ikili sınıflandırıcı olarak Evrişimsel Sinir Ağı (ESA) (CNN-Convolutional Neural Network), Patch Evrişimsel Sinir Ağı (PatchESA) (PatchCNN) ve Kapsül Sinir Ağı (KSA) (CapsNet-Capsule Neural Network) kullanılarak yapılmıştır. Elde edilen sonuçlar, Covid-19 BT taramalarının Koşullu Üretken Çekişmeli Ağ (kÜÇA) ile Unet hibrit yapısından elde edilen segmentasyon sonuçlarının Evrişimsel Sinir Ağı (ESA) kullanılarak yüksek başarı ile sınıflandırılabilirliğini ortaya koymaktadır.

1.1. Tezin Amacı

Bu tez, Covid19 BT görüntülerinin Üretken Çekişmeli Ağlar (ÜÇA) (GAN) kullanılarak segmentasyonunu ve segmentlenmiş görüntülerin sınıflandırma başarılarını araştırmayı amaçlamaktadır.

Literatürde çeşitli BT taramalarında, Covid-19 dokusunu tespit etmek için koşullu ÜÇA'a dayalı otomatik segmentasyon modelleri geliştirilmiştir. Üretken ağ (generator), tümör bölgesini tanımayı ve ana hatlarını çizerek ikili maskeler oluşturmayı öğrenirken, ayırıcı (discriminator) ise gerçek veri ve üretilen maskeleri ayırt etmeyi öğrenir. Bu durum, üreticinin mümkün olduğunca gerçekçi ikili maskeler oluşturmasını sağlar. Bu tez kapsamında sentetik maskeler oluşturmanın yanı sıra, görüntülerde Covid-19'un olup olmadığını tespit etmek için oluşturulan bu ikili maskeler üzerinde farklı sınıflandırma modellerinin kullanılabilirliğinin gösterilmesi amaçlanmıştır. Sınıflandırma için doku, yoğunluk veya kenar bilgilerini kullanan geleneksel sınıflandırıcıların (Apostolopoulos ve Mpesiana, 2020; Loey ve ark., 2020; Sethy ve ark., 2020; Yang ve ark., 2020) aksine, önerilen yöntem ikili maskelerin yalnızca morfolojik özelliklerini kullanmaktadır. Böylelikle, sadece morfolojik özelliklerle yapılan segmentasyon işlemi ile elde edilen

sonuçlarla BT görüntüleri üzerinde üç ayrı sınıflandırıcının sınıflandırma etkisinin araştırılması amaçlanmıştır.

Tez çalışması süresince yapılan uygulamalar, önerilen tanı sisteminin Covid-19 segmentasyon ve sınıflandırması için başarılı sonuçlar verdiğini göstermektedir. Önerilen model, sınırlı sayıda eğitim örneğiyle bile iyi bir performans sergilemektedir. ÜÇA ile yapılan segmentasyon verileri kullanılarak yapılan sınıflandırma sonuçları, orijinal BT taramaları ile elde edilen sonuçlara kıyasla, sınıflandırıcıların performansında önemli bir artış olduğunu göstermektedir. Covid-19 doku sınırlarının segmentlere ayrılması, radyoterapi ve hastalığın ilerlemesini izlemek için kullanılabilen Bilgisayar Destekli Tanı (BDT) (CAD) sistemlerinde ilgi bölgesini (İB) (ROI) belirlemek açısından faydalı ve kullanılabilir seviyededir.

1.2. Tezin Önemi

Koronavirüsün dünya genelindeki ölüm oranı ve bulaşma hızı, büyük bir endişe vermektedir. 5 Mayıs 2021 tarihi itibarıyla, pandeminin başlangıcından bu yana küresel olarak yaklaşık 155 milyon kümülatif vaka ve 3.25 milyon ölüm bildirilmiştir (WHO, 2021). Yayılmayı kontrol etmek üzere atılan en önemli adımlardan biri, enfekte kişileri mümkün olduğunca erken tespit etmek ve izole etmektir. Ana teşhis aracı olarak kullanılan RT-PCR testinin hassasiyeti, Covid-19'u etkili bir şekilde tespit etmek için hâlâ yeterince yüksek değildir. Test için gereken malzemelerin eksikliğine ek olarak, tarama uzun zaman almakta ve bazen doğru teşhis için testlerin tekrarlanması gerekmektedir. Bu nedenle, daha iyi ve hızlı bir teşhis yöntemine ihtiyaç duyulmaktadır. Röntgenler veya BT taramaları gibi tıbbi görüntüler üzerinden Covid-19 taraması, RT-PCR'ye alternatif bir çözümdür. BT taramalarından Covid-19'u tespit etmek için, derin öğrenme algoritmalarını kullanan bilgisayar destekli tanı (BDT) (CAD) yazılımı uygulanabilir bir yapıdadır. Verimli olmaları halinde, bu sistemler sağlık çalışanlarının iş yükünü azaltma ve optimize etme potansiyeline sahiptir. Ayrıca, daha hızlı sonuç elde edilebileceği için daha kısa sürede daha fazla hastanın bakılmasını sağlayacaktır. Yüksek doğruluk ve hızlı teşhis sayesinde enfekte hastalar izole edilebildiği için hastalığın yayılması da azalacaktır.

BT taramaları üzerinden Covid-19'u segmentlere ayırmak ve sınıflandırmak çeşitli zorlukları da beraberinde getirmektedir. İlk olarak BT taramaları enfeksiyon

karakteristiklerinde yüksek çeşitlilik ve enfeksiyonlar ile normal dokular arasında düşük yoğunluklu kontrast gösterdiği için, enfekte bölgelerin segmentasyonunu bir hayli zorlaştırmaktadır. Ayrıca, derin öğrenme kullanarak görüntü segmentasyonu veya sınıflandırması gerçekleştirmek için genellikle büyük miktarda açıklamalı eğitim ve doğrulama verisi gerekir. Ama, bulaşıcı bir hastalığa ilişkin büyük miktarda veri toplamak, hasta verilerinin paylaşımına ilişkin kısıtlamalar nedeniyle oldukça zordur.

1.3. Tezin Organizasyonu

Tez, beş bölümden oluşmaktadır. İlk bölümde Covid-19 tespitindeki zorluklar ve tanı yöntemleri verilmiş, bu anlamda tez kapsamında önerilen modelin içeriği hakkında bilgi verilmiştir. Tezin amacı ve önemine ilişkin yer alan bilgiler de bu bölüm içerisinde verilmiştir.

İkinci bölümde literatür incelenmiştir. Biyomedikal görüntülerde Covid-19 tanısı için hem segmentasyon hem de sınıflandırma için literatürdeki yöntemler bu bölümde incelenmiştir.

Üçüncü bölüm, tez çalışmasında kullanılan yöntemler ve veri setleri hakkında bilgi içermektedir. Kullanılan veri kümeleri ve derin öğrenme algoritmaları hakkındaki bilgilerin tümü bu bölümde sunulmuştur.

Dördüncü bölümde, geliştirilen yönteme ait uygulama aşamaları verilerek, deneysel sonuç analizleri sunulmuştur.

Son olarak beşinci bölümde ise tez kapsamında yapılan çalışmalar ve elde edilen sonuçlar genel olarak değerlendirilip, literatür ile karşılaştırmalar verilmiş ve sonraki çalışmalar için öneriler sunulmuştur.

2. KAYNAK ARAŞTIRMASI

Son yıllarda makine öğrenmesi, otomatik tıbbi görüntü analizi için en yaygın ve doğru yöntemlerden biri haline gelmiştir (Biswas ve ark., 2019). Derin Öğrenme (DÖ), makine öğrenmesinin bir alt alanı olarak beynin yapısı ve işlevinden esinlenerek oluşturulan “Yapay Sinir Ağları” (YSA) (ANN-Artificial Neural Network) algoritmaları ile ilgilenir. Evrimsel Sinir Ağları (ESA) (CNN-Convolutional Neural Network), verilen bir görüntüden özellik temsilini hiyerarşik ve otomatik bir şekilde öğrenebildikleri için son zamanlarda en çok kullanılan derin öğrenme algoritmalarıdır. ESA, bir görüntünün düşük düzeyden yüksek düzeye kadar temsillerini öğrenmeyi, görüntüdeki çeşitli yönler/nesnelere önem (öğrenilebilir ağırlıklar ve önyargılar) atamayı ve bir nesneyi diğerinden ayırt etmeyi mümkün kılmaktadır. Tez kapsamında da DÖ algoritmaları kullanılarak Covid-19 segmentasyonuna ve sınıflandırmasına odaklanan bir yöntem önerilmiştir.

2.1. Covid-19 Segmentasyonu Üzerine Çalışmalar

Semantik segmentasyonun amacı, tıbbi bir görüntüdeki her pikseli etiketleyerek organlar veya tıbbi anormallikler gibi İlgi Bölgelerini (İB) (ROI) belirlemektir. Segmentasyon; tanı koyma, hastalığın ilerlemesini izleme, hız ve verimlilik gibi konuların iyileştirilmesinde, tıbbi görüntüdeki hastalıklı alanların doğru olarak belirlenerek radyologlara yardımcı olması sebebiyle Covid-19 görüntü analizinde çok önemli bir adımdır. Segmentasyon amacıyla son yıllarda kullanılan ve başarılı sonuçlar üreten DÖ modellerinin başarılı bir şekilde sonuç üretebilmesi için büyük miktarlarda veri kullanılarak eğitilmeleri ve bununla birlikte görüntü segmentasyonu gibi belirli görevlerde, modele verilecek verilerin iyi etiketlenmiş veriler olması gerekmektedir. Covid-19’un son derece bulaşıcı olması nedeniyle sağlık personelinin işlem sürecinde iyi korunması ve bu süreçte kullanılan ekipmanın sık sık dezenfekte edilmesi gerektiğinden, veri toplama süreci özellikle zorlaşmaktadır (Müller ve ark., 2020). Ayrıca, Covid-19 ile ilgili toplanmış olan bazı veri kümelerinin halka açık olmaması nedeniyle de veriler bu alandaki araştırmacılar tarafından da kullanılamamaktadır. Bu zorluklara rağmen, Covid-19 segmentasyonu için toplanan biyomedikal görüntüler kullanılarak, hastalığın teşhisi için çeşitli yaklaşımlar

önerilmiştir. Kodlayıcı-kod çözücü mimarisi (encoder-decoder architecture), semantik segmentasyon için yaygın olarak kullanılmaktadır (Xing ve ark., 2020). Bir kodlayıcı-kod çözücü yapısında kodlayıcı, girdinin özellik temsillerini öğrenirken kod çözücü, bu özellik temsilini alarak havuzlama işlemi sırasında kaybolan konum bilgisini kurtarır ve ikili bir maske üretir. İyi bilinen bu tür mimarilerden birisi, kodlayıcı ve kod çözücü katmanları arasına atlama bağlantıları ekleyerek girdi görüntülerinden önemli bilgileri elde eden UNet'tir (Ronneberger ve ark., 2015). UNet ve türevleri, Covid-19'un segmentasyonu için geliştirilmiştir. Zhou ve ark.(2020), yapmış oldukları çalışmada UNet'i değiştirerek kodlayıcı ve kod çözücü yolu arasına iç içe bir Evrişimsel yapı eklemiş ve UNet++'ı önermişlerdir. Bu ağ, Chen ve ark. (2020) tarafından lezyon segmentasyonu için uygulanmıştır. Lezyonun segmentasyonundan sonra, pozitif ve negatif taramaları belirlemek için 3 ardışık kesit (3-consecutive slice) ile kadrant-bazlı işlem sonrası (quadrant based post-processing) yöntemler uygulanmıştır. Kadrant bazlı işlemde, ardışık 3 çıktı görüntüsündeki lezyonlar eğer aynı kadranda olursa çıktı kabul edilmiş; aksi takdirde reddedilmiştir. UNet'in bir başka türevi olan 3D UNet (Çiçek ve ark., 2016), geleneksel UNet'in katmanlarını bir 3D versiyonla değiştirerek, segmentasyon için kesitler arası bilgiyi kullanmaktadır.

Semantik segmentasyon için iyi bilinen bir başka mimari, ESA'nın tamamen bağlı katmanlarının kaldırıldığı Tam Evrişimli Ağ'dır (TEA) (FCN-Fully Convolution Network) (Chen ve ark., 2020). Milletari ve diğerleri (2016), temel evrişim bloğu olarak artık blokları olan bir TESA kullanmışlardır. İleri sürdükleri V-Net isimli mimarileri, 3D hacimler için yapılmıştır ve ağı optimize etmek için Zar kaybı (Dice Loss) kullanılmıştır. Shan ve ark.(2020), BT taramalarında Covid-19 enfeksiyon bölgelerini segmentlere ayırmak için VB-Net sinir ağını kullanmışlardır. Ağı eğitmek için human-in-the-loop yaklaşımını entegre etmişlerdir; böylece, radyologlarla etkileşim içinde kalmışlardır. Buna karşılık Oktay ve ark. (2018), Covid-19 görüntülerinde ince yapıları yakalamak için gelişmiş dikkat mekanizmalarını kullanan; ağ, lezyonları ve akciğer nodüllerini segmentlere ayırabilen Attention UNet'i önermişlerdir.

Açıklamalı tıbbi görüntülerin eksikliği nedeniyle, bazı araştırmalar denetimsiz veya yarı denetimli yöntemleri içermektedir. Zheng ve ark. (2020), Covid-19 BT taramalarından sahte segmentasyon maskeleri üreten denetimsiz bir öğrenme tekniği sunmuşlardır. Fan ve ark. (2020), BT kesitlerinde enfekte bölgelerin tanımlanması için Inf-Net adlı yarı

denetimli bir yaklaşım önermişler, enfekte görüntülerde üst düzey özellikleri ve model sınırlarını toplamak için kısmi bir kod çözücü ve ters dikkati kullanmışlardır.

T. Zhou ve ark. (2020), kodlayıcılardan yararlı özellikleri çıkarmak için dikkat mekanizmasını içeren UNet tabanlı bir bölümlendirme ağı önermişlerdir. Özellik temsilini yeniden ağırlıklandırmak ve zengin bağlamsal ilişkileri yakalamak için uzamsal ve kanal dikkatini kullanmışlardır. Sonuç olarak %83,1 zar skoru (dice score) elde etmişlerdir. Chen ve ark. (2020), ise çalışmalarında birden fazla Covid-19 enfeksiyon bölgesinin otomatik segmentasyonu için bir derin öğrenme algoritması önermişlerdir. Daha sonra Wu ve ark. (2021), gerçek zamanlı ve açıklanabilir Covid-19 teşhisi gerçekleştiren bir Ortak Sınıflandırma ve Segmentasyon (JCS) sistemi geliştirmişlerdir. Kendi oluşturmuş oldukları ve 144167 BT görüntüsünden oluşan büyük ölçekli bir veri kümesinde %78'lik bir zar skoru elde etmişlerdir.

Literatürde görüntü segmentasyonu için Üretken Çekişmeli Ağları (ÜÇA) kullanan çeşitli yaklaşımların da önerildiği görülmektedir. Ancak bu yaklaşımlar, Covid-19 BT taramalarını sentezleyerek veri kümesini büyötmeye dayanmaktadır. Zhang ve ark. (2020), enfekte bölgeleri açıklama maskesi ile tek bir radyolojik görüntüden öğrenilebilen CoSinGAN'ı önermişlerdir. Model, girdi koşullarına uyan, yüksek çözünürlüklü ve çeşitli yapılarıdaki görüntüleri sentezleyebilmektedir.

2.2. Covid-19 Sınıflandırması Üzerine Çalışmalar

Tıbbi görüntü sınıflandırması, tam bir görüntüyü önceden tanımlanmış sınıflara etiketlemeyi amaçlar ve bu nedenle teşhis için kullanılabilir. Literatürde Covid-19'u sınıflandırmak için birçok yaklaşım önerilmiştir. Zhang ve ark. Covid-19 ve Covid-19 olmayan röntgen görüntüleri arasında anormallik algılayan ve sınıflandıran ResNet tabanlı bir model önermişlerdir (Zhang ve ark., 2019). Anomali skoru, 10078 görüntü içeren veri kümesini sınıflandırmayı optimize etmek için kullanılmıştır. Modelde; duyarlılık (sensitivity), özgüllük (specificity) ve AUC skoru %96.0, %70.7 ve %95.2 olarak elde edilmiştir.

Loey ve ark., koronavirüs tespiti için derin transfer öğrenmeli bir ÜÇA uygulamışlardır (Loey ve ark., 2020). Kullanılan veri setindeki görüntülerden daha fazla sayıda görüntü oluşturmak için ÜÇA kullanmışlardır. Görüntüler sırasıyla %66.67, %69.67

ve %80.56 doğruluk (accuracy) üreten AlexNet, ResNet18 ve GoogleNet kullanılarak ilk senaryoda sınıflandırılmıştır. Apostolopoulos ve Mpesiana yapmış oldukları çalışmada, Vgg19 ile Covid-19'un tespiti için transfer öğrenmeyi kullanmışlar ve %96.78 oranında doğruluk elde etmişlerdir (Apostolopoulos ve Mpesiana, 2020). Yang ve ark. ise çalışmalarında, açık kaynaklı bir veri kümesi oluşturmuş ve bir ESA modelinin Covid-19'u teşhis etmek için kullanılabileceğini göstermişlerdir (Yang ve ark., 2020). Önerdikleri model ile %89 doğruluk ve %90 F1-skoru elde etmişlerdir. Hemdan ve ark.(2020), Covid-19 teşhisi için yedi farklı transfer öğrenme mimarisi içeren CovidX-Net'i önermişlerdir. Çalışmada kullanılan Vgg19 ve DenseNet, %90 doğruluk ile en iyi performansı göstermişlerdir.

Sethy ve ark. (2020), çeşitli ESA'lardan elde edilen özellikleri sınıflandırmak için Destek Vektör Makineleri (DVM) (SVM-Support Vector Machine) kullanmışlardır. Yaptıkları çalışmada, DVM sınıflandırıcısı ile çalışan ResNet50 modeli %95.33 doğrulukla en iyi sonucu elde etmiştir. Benzer şekilde Song ve ark. (2020), BT görüntülerinden özellikleri çıkarmak için bir Özellik Piramit Ağı (ÖPA) (FPN-Feature Pyramid Network) kullanmışlar, çıkarılan özellikler ile ResNet-50 modeliyle sınıflandırma yapmışlardır. Çalışmada %86 oranında doğruluk elde edilmiştir. Öztürk ve ark. (2020), röntgen görüntülerinden Covid-19'u %98.08 doğrulukla otomatik olarak tespit eden DarkCovidNet'i sunmuşlardır. DarkCovidNet, en gelişmiş gerçek zamanlı nesne algılama sistemi YOLO'nun (You Only Look Once) omurgası olan Darknet-19'dan esinlenmiştir. Öte yandan Wang ve ark. tarafından önerilen Covid-Net, 13975 röntgen görüntüsünden oluşan büyük bir veri kümesi üzerinde eğitilmiştir(Wang ve ark., 2020). Model mimarisi, gelişmiş temsil kapasitesi sağlayan “projeksiyon-genişleme-projeksiyon tasarım deseni” tasarım modelini (projection-expansion-projection design pattern) kullanmıştır.

Wang ve ark., Covid-19 ve viral pnömoni arasında sınıflandırma gerçekleştiren bir 2D ESA modeli önermişlerdir (Wang ve ark., 2020). Deneysel sonuçlar, önerilen modelin %73.1 doğruluk, %74.0 duyarlılık ve %67.0 özgüllük sağladığını göstermiştir. Derin ESA tabanlı bir model olan Covid-Net, %83.5'lik bir test doğruluğu sağlayarak Wang ve ark. (2020) tarafından, röntgen görüntülerinde Covid-19'u tespit etmek için önerilmiştir. Narin ve ark. (2020) çalışmalarında farklı transfer öğrenme yöntemleri kullanmışlardır. Çalışmada Covid-19'u tespit etmek amacıyla iki farklı veri kümesinde ResNet50, InceptionV3 ve Inception-ResNetV2 ikilisini kullanmışlardır. ResNet50 %98 doğrulukla en

iyi performansı elde ederken, InceptionV3 ve Inception-ResNetV2 sırasıyla %97 ve %87 doğruluk elde etmiştir. Ghoshal ve Tucker, standart Vgg16 modelinin sınıflandırma doğruluğunu %85.7'den %92.8'e yükselten bir Bayes Evrişimsel Sinir ağı sunmuşlardır (Ghoshal ve Tucker 2020). Yazarlar, modelin karar vermek için odaklandığı belirli konumları belirtmek için belirginlik haritalarını (saliency maps) kullanarak açıklama bilirliliği artırmışlardır.

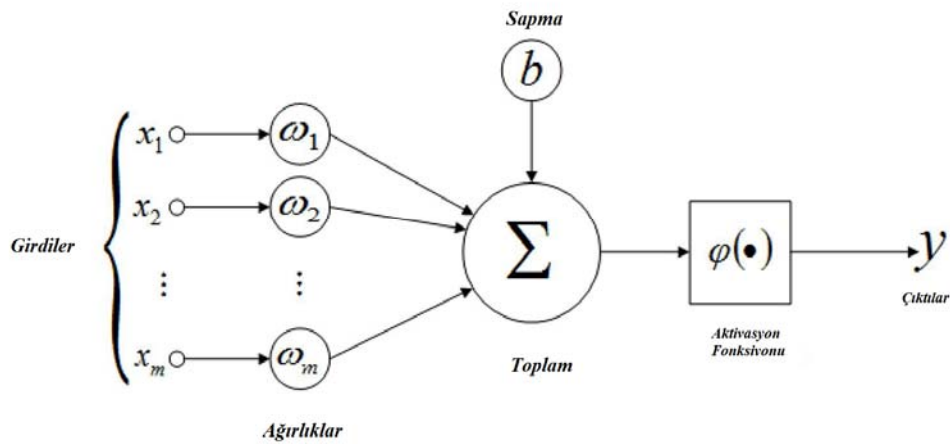
Li ve Zhu (2020) tarafından önerilen Covid-19 taraması için kullanılabilecek hafif derin sinir ağı Z Model, önceden eğitilmiş bir yoğun ağı (DenseNet-121), yerleşik bir arkadaş ağı ve tıp öğrencisi ağından oluşan 3 oyunculu bir bilgi aktarımı ve damıtma (KTD) çerçevesinden oluşmaktadır.

Covid-19 teşhisinde önerilen birçok yöntem ile hem segmentasyon hem de sınıflandırma gerçekleştirilmiştir. Zheng ve ark. (2020), akciğer segmentasyonu için bir UNet modeli kullanmış ve ortaya çıkan sonuç, Covid19'un varlığını tahmin etmek için bir 3D ESA'ya girdi olarak kullanılmıştır. Bu model ile %90.7 duyarlılık, %91.1 özgüllük ve %95.9 AUC oranı elde edilmiştir. Benzer şekilde Chen ve ark.(2020), BT görüntülerinden segmentli lezyonlar elde etmek ve Covid-19 ve diğer viral pnömoni türlerini tahmin etmek için UNet++'ı kullanmışlardır. Önerilen model sayesinde radyologların okuma süresi %65 oranında kısaltılmıştır. Xu ve ark. (2020), aday enfekte bölgeleri segmentlere ayırmak için bir V-Net kullanmışlar ve elde edilen çıktıyı el yapımı özelliklerle birleştirmişlerdir. Bu kombinasyon daha sonra %86.7'lik sınıflandırma doğruluğuna ulaşan bir ResNet-18 ağına gönderilmiştir. Li ve ark. (2020) , akciğer bölgelerini çıkarmak için UNet ve ResNet50'nin bir kombinasyonunu kullanmışlar ve %96 özgüllük, %90 hassasiyet ve %96'lık bir AUC ile sınıflandırma gerçekleştirmişlerdir. Amyar ve ark. (2020), BT görüntülerinden Covid-19 lezyonlarını ortaklaşa sınıflandıran, yeniden yapılandıran ve segmentlere ayıran çok görevli bir derin öğrenme modeli önermişlerdir. Önerilen model ortak bir kodlayıcı, 2 kod çözücü ve çok katmanlı bir algılayıcıdan oluşmaktadır ve %96 doğruluk oranına sahip bir zar katsayısı üretmiştir. Karakanis ve Leontidis (2021), Covid-19'u tespit etmek için ÜÇA ile üretilen sentetik görüntüleri kullanan hafif bir derin sinir ağı önermişlerdir. Önerilen modelden %98.7 doğruluk oranı elde edilmiştir.

3. MATERYAL VE YÖNTEM

3.1. Derin Öğrenme

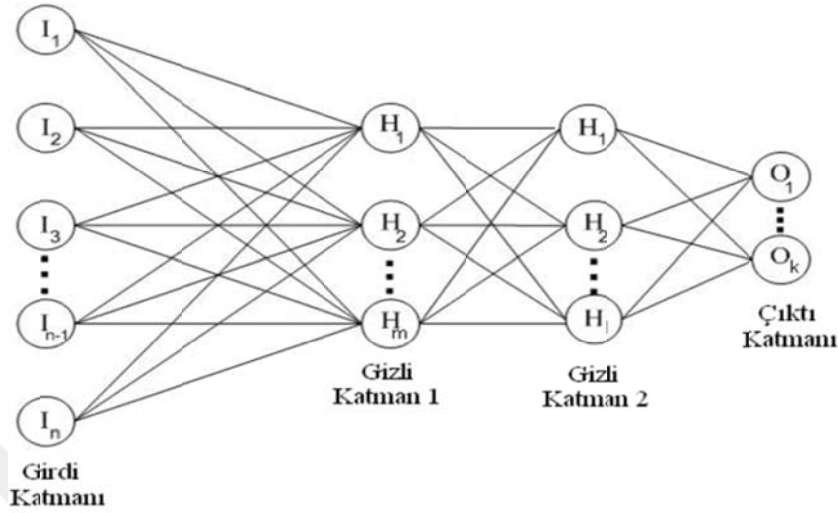
Derin öğrenme, yapay sinir ağlarını kullanarak insan beyninin çalışma şeklini modellemek için tasarlanmış bir tür makine öğrenme sistemidir. Bir Yapay Sinir Ağı (YSA), birbirine yoğun şekilde bağlı binlerce hatta milyonlarca basit işlem nöronundan (sinir hücresi) oluşur. Her yapay nöron, bir diğerine bağlanır ve ilişkili bir “ w ” ağırlığına ve “ b ” sapmaya (bias) sahiptir. “Algılayıcı” olarak adlandırılan yapay bir nöronun en basit modeli, ilk olarak 1957’de R. Rosenblatt tarafından önerilmiş (Rosenblatt, 1957) ve Şekil 3.1’de şematik olarak özetlenmiştir. Bir algılayıcıda, alınan x veri girdileri, w ağırlıkları ile çarpılır ve ardından toplanır. Toplam, belirli bir eşik değerinden fazlaysa, nöron “ateşler” ve bağlı olduğu nöronları harekete geçirir. Bu eşik, “aktivasyon fonksiyonu” tarafından belirlenmektedir. YSA’nın örüntü tanıma, görüntü sınıflandırma ve karar verme gibi belirli görevlerde oldukça başarılı olduğu kanıtlanmıştır (Francesco ve ark., 2019).



Şekil 3.1. Bir yapay nöron yapısı

En yaygın YSA yapısı; bir girdi katmanı, bir veya daha fazla gizli katman ve bir çıktı katmanından oluşmaktadır. Bir sinir ağındaki gizli katman sayısı arttıkça derin sinir ağları oluşur. Şekil 3.2; 1 girdi katmanı (3 nöron), 2 gizli katman (4’er nöron) ve 1 çıktı katmanı (1 nöron) olan bir yapay sinir ağını (3x4x4x1) göstermektedir. Bir katmandaki tüm

nöronlar bir sonraki katmandaki tüm nöronlara bağlı olduğu için, bu ağı “tam bağlı ağ” (fully connected) denir.



Şekil 3.2. Yapay Sinir Ağı örneği (Kutlu ve Badur, 2009)

Girdi verileri, girdi katmanı aracılığıyla sinir ağına, ardından gizli katmanlara iletilir. İşlem, ağırlıklı bir bağlantı sistemi aracılığıyla gizli katmanlarda gerçekleşmektedir. Gizli katmandaki nöronlara rastgele w ağırlıkları atanır ve bunlar daha sonra girdi verileriyle çarpılır. Tüm girdiler ağırlıklar ile çarpıldıktan sonra Denklem 3.1’e göre nöronun çıktısını üretmek için elde edilen sonuca b sapma değeri eklenir:

$$y = b + \sum_{i=1}^n w_i x_i \quad (3.1)$$

Bu denklemde n girdi sayısı, y nöron çıktısı ve b ise sapma değerini ifade etmektedir. Aktivasyon fonksiyonunun net girdisini artırmak veya azaltmak için her katmanın girdisine bir sapma eklenir ve bunun değişmesine izin verilir (Goodfellow ve ark., 2016).

Aktivasyon, bir sinir ağının gizli katmanlarındaki değerleri hesaplamak için gereklidir. Girdiyi bir sonraki nöron veya çıktı katmanına göndermeden önce yapılan doğrusal olmayan bir dönüşümdür. Temel olarak bir nöronun aktive edilip edilmeyeceğine karar vermek için kullanılır. Böylece net girdinin değeri sınırlanmış olur. Yaygın olarak

kullanılan aktivasyon fonksiyonlarından birisi, Denklem 3.2’de gösterilen ReLU adı verilen Rektifiye Edilmiş Doğrusal Birim’dir (Rectified Linear Unit).

$$f(z) = \begin{cases} z & \text{if } z \geq 0 \\ 0 & \text{if } z < 0 \end{cases} \quad (3.2)$$

ReLU, sinir ağlarında kaybolan gradyan problemini önler ve bu nedenle sinir ağlarında sıklıkla kullanılır. Normal ReLU'nun eğimi, negatif kısımda her zaman sıfır değerini almaktadır. Aktivasyon fonksiyonunun negatif tarafında bir nöron sıkışır, her zaman sıfır verir ve nöron ölü olarak kabul edilir. LeakyReLU, sıfır eğimi olan bir kısma sahip olmayarak bu sorunu çözer. LeakyReLU, Denklem 3.3’te tanımlanmaktadır.

$$f(z) = \begin{cases} z & \text{if } z \geq 0 \\ 0.01z & \text{otherwise} \end{cases} \quad (3.3)$$

Sigmoid, YSA’da kullanılan diğer bir aktivasyon fonksiyonudur. Sigmoid fonksiyonunun kullanılmasının temel nedeni 0 ile 1 arasında $([0,1])$ bulunmasıdır. Özellikle çıktının bir olasılık olarak tahmin edildiği modeller için kullanılmaktadır. Herhangi bir şeyin olasılığı sadece 0 ile 1 aralığında var olduğu için sigmoid doğru bir seçimdir. Denklem 3.4, fonksiyonun formal tanımını göstermektedir.

$$\text{sigmoid}(z) = \frac{1}{1 + e^{-z}} \quad (3.4)$$

Sigmoid aktivasyon fonksiyonunun aksine, hiperbolik tanjant (tanh olarak da bilinir) $[-1, 1]$ arasında bir aralığa sahiptir. Tanh için formal gösterim Denklem 3.5’te verilmiştir.

$$\text{tanh}(z) = \frac{2}{1 + e^{-2z}} - 1 \quad (3.5)$$

Bir aktivasyon fonksiyonu olarak tanh kullanıldığında, $[-1, 1]$ arasındaki aralık nedeniyle sadece sıfıra yakın değerler sıfıra eşlenecektir.

YSA’da “düzenleştirme” (regularization); “aşırı öğrenmeyi” (overfitting) önlemek için kullanılan bir teknik olup, modelin eğitim verileri üzerinde test verilerine kıyasla çok daha iyi performans gösterdiği durumlarda meydana gelir. Bir ağda düzenleştirme gerçekleştirmenin veri artırma (data augmentation), seyreltme (dropout), yığın normalleştirme (batch normalization) ve örnek normalleştirme (Instance or Contrast normalization) gibi birkaç farklı yolu bulunmaktadır (Goodfellow ve ark., 2016).

Veri artırma, bir ağın performansını iyileştirmenin etkili bir yoludur. Ağın eğitimine daha fazla veri eklenerek, sinir ağının daha iyi genelleme yapmayı öğrenmesi sağlanır. Bu yaklaşım, sentetik (yapay) veriler oluşturularak ve bunları eğitim verilerine ekleyerek gerçekleştirilebilir. Görüntüleri kullanırken, görüntülerin kırılması (cropping), yakınlaştırılması (zooming) ve döndürülmesi (rotating) artırma için faydalı tekniklerdir (Goodfellow ve ark., 2016).

Seyreltme (Dropout), ağın her yinelemesinde bazı düğümleri rastgele seçerek girdi ve çıktıları kaldırma yöntemidir. İptal edilen düğümler birbirinden bağımsız olarak örneklenir ve olasılık bir hiper parametre ile seçilir (Goodfellow ve ark., 2016). Böylece, düğümlerin çok fazla uyum sağlamasının önüne geçilebilir. Tüm ağ bir kerede eğitirken, diğer düğümlerin neden olduğu hatalar nedeniyle bazı düğümler değişebilir ve bu da düğümler arasında ortak uyumlara neden olmaktadır. Ağ üzerinden yayılırken sürekli olarak farklı düğümler seçerek, belirli bir düğümün parametrelerini yalnızca kendi hatalarını düzeltmeye zorlar ve böylece aşırı öğrenmeyi önler (Srivastava ve ark., 2014).

Yığın normalleştirme (Batch normalization), ‘yığın’ olarak adlandırılan her bir girdi eğitim verisi kümesini normalleştirerek, ağın performansının istikrarını artırmaktadır. Ağ, yığın ortalamasını çıkararak ve yığın standart sapmasına bölerek önceki bir aktivasyon katmanının çıktısını normalleştirir.

Örnek normalleştirme (Instance normalization), yığın normalleştirme ile toplu olarak yapılan normalleştirme yerine, her eğitim örneğindeki her bir kanalı normalleştirerek ağın performansının istikrarını artırır. Bu, ortalamanın çıkarılması ve her kanaldaki standart sapmaya bölünmesiyle yapılmaktadır (Ulyanov ve ark., 2016).

Kayıp (Loss) fonksiyonu, tasarlanan modelin hata oranını ve aynı zamanda başarımını ölçen fonksiyondur. Derin ağların son katmanı kayıp (loss) fonksiyonun tanımlandığı katmandır. Kayıp fonksiyonu hata hesaplama işini, problemi bir optimizasyon problemine dönüştürerek yaptığı için optimizasyon terminolojisinde kullanılan “objective

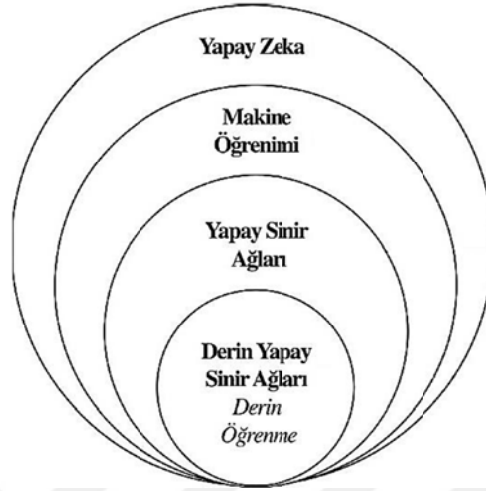
function” (amaç fonksiyonu) ya da “cost function” (maliyet fonksiyonu) isimleriyle de anılmaktadır. Kayıp fonksiyonu temelde modelin yaptığı tahminin (predicted), gerçek değerden (ground truth) ne kadar farklı olduğunu hesaplamaktadır. Bu nedenle iyi tahmin eden bir model oluşturulmamışsa, gerçek değer (ground truth) ile tahmin edilen değer (predicted) arasındaki fark dolayısıyla kayıp (loss) değeri yüksek olacaktır. Ancak iyi bir modelde kayıp (loss) değeri az olacaktır. Gerçek değer ile ağırlık çıktısındaki tahmin değeri birebir aynı olduğu durumda kayıp değeri sıfır (0) olacaktır. İyi bir modelden sıfıra yakın kayıp değerine sahip olması beklenmektedir (Çarkacı, 2017).

Optimizasyon / Kayıp (amaç) işlevini en aza indirmek ve modeldeki ağırlık ve sapmaları güncellemek için bir optimizasyon stratejisi kullanılmaktadır. Doğrusal olmayan problemlerin çözümünde optimum değeri bulmak için optimizasyon yöntemleri kullanılmaktadır. Derin öğrenme uygulamalarında yaygın olarak Stokastik Gradyan İnişi (stochastic gradient descent), Adagrad, Adadelata, Adam (Adaptive Moment Estimation), Adamax gibi optimizasyon algoritmaları kullanılmaktadır. “Öğrenme hızı” parametresiyle çarpılması, çıkan sonucun ağırlık değerlerinden çıkarılarak yeni ağırlık değerinin hesaplanmasıyla yapılmaktadır. “Öğrenme oranı” parametresi sabit değer olarak belirlenebilirken, adım adım artan bir değer olarak (0.001 adımdan sonra 0.01 gibi) veya momentum değerine (β_1 ve β_2) bağlı olarak belirlenebilir. Stokastik Gradyan İniş gibi çok fazla salınım oluşturan, teknik olarak normalize edilemez salınımlar azaltılmaktadır. Bu normalizasyon işlemi momentum beta katsayıları (β_1 ve β_2) ile yapılmaktadır. Bu teknikte yeni bir değeri olduğu gibi almak yerine, yeni değer bir önceki beta katsayısı değerinin içine katılarak hesaplanır. Bu işlem, gürültü ve grafikteki salınımları azaltarak daha hızlı bir yöntem sunmaktadır (Çarkacı, 2018).

$L1$ ve $L2$ regülasyonları, model performansını iyileştirebilecek diğer teknikler arasında sayılmaktadır. $L1$ (Lasso) regülasyonu verinin ortanca (medyan) değerini tahmin etmeye çalışırken, $L2$ (Ridge) regülasyonu aşırı öğrenmeyi önlemek için verinin ortalamasını tahmin etmeye çalışmaktadır. Ayrıca $L1$, önemli olmayan özellikleri ortadan kaldırarak özellik seçiminde yardımcı olur. Bu, özellik noktalarının sayıca fazla olduğu durumlarda yararlı olmaktadır.

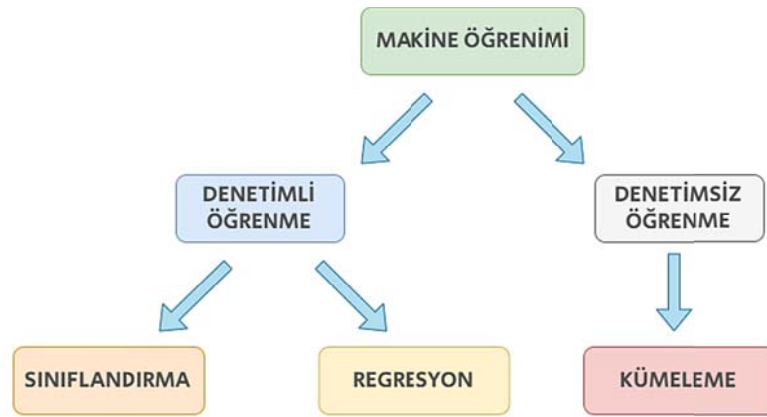
DÖ, yapay sinir ağlarını temel alan makine öğrenmesinin bir alt dalıdır. DÖ, belirsizlik altında tahmin yapmak veya karar vermek için kullanılan, büyük miktarda veriden örüntü çıkarmak için kullanılabilen yapay zekâ sistemidir (Şekil 3.3). Tıbbi teşhis,

yüz tanıma, makine çevirisi vb. sorunları çözmek için günümüzde farklı algoritmalar yaygın olarak kullanılmaktadır.



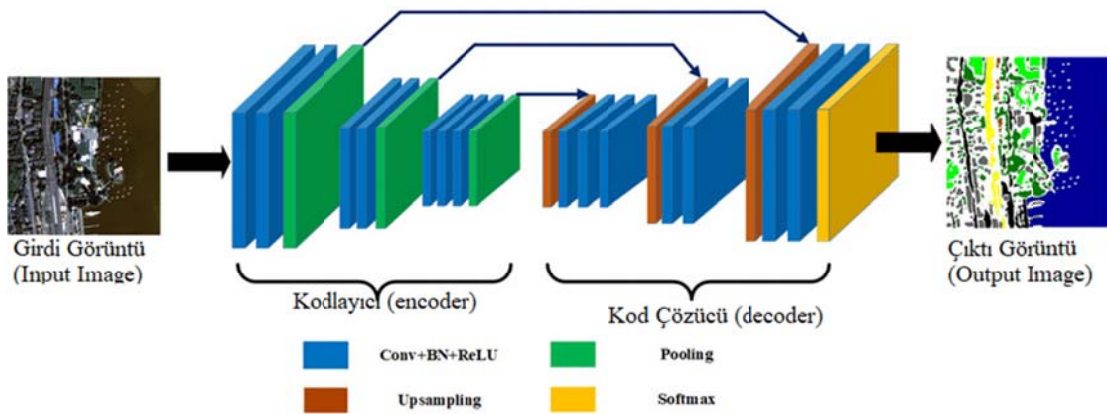
Şekil 3.3. Yapay Zeka ve alt alanlarının gösterimi

Temel olarak denetimli, denetimsiz ve takviyeli olarak üç tür makine öğrenmesi algoritması vardır. Denetimli öğrenme, etiketli bir girdi-çıkı çifti veri kümesi kullanarak girdilerden çıktıları eşlemeyi içerirken; denetimsiz öğrenmede etiketli bir veri kümesi gerekli değildir. Bunların yanında, eğer veri setinde az sayıda etiketlenmiş veri ve buna karşın çok daha fazla sayıda etiketlenmemiş veri varsa, bu durumda az sayıdaki etiketlenmiş veriden hareketle etiketlenmemiş veriler hakkında çıkarım yapmak da takviyeli öğrenmeye dayanmaktadır. En yaygın denetimli öğrenme türü sınıflandırmadır. Burada algoritmanın amacı, bir dizi girdi almak ve her girdi için doğru sınıf etiketini üretmektir (Cucinotta ve Vanelli, 2020). Denetimsiz öğrenmede ise, kümeleme önemlidir. Kümeleme, genel anlamda başlangıçta kesin belirlenmiş kategorileri olmayan verileri, benzerliklerine göre gruplamaktır. Görüntü işlemede bu işleme “segmentasyon” örnektir (Şekil 3.4).



Şekil 3.4. Makine öğrenmesi ve alt dalları

Derin öğrenmede çoğu semantik segmentasyon modeli, bir Kodlayıcı (encoder) ve bir Kod Çözücü (decoder) olmak üzere iki parça içerir. Kodlayıcı, görüntüden özelliklerin çıkarılmasından sorumludur. Kod çözücü ise, sonuç görüntüsünde pikselin sınıfını tahmin edendir. Segmentasyon görevi için tipik bir Kodlayıcı-Kod Çözücü, Şekil 3.5’te verilmiştir.



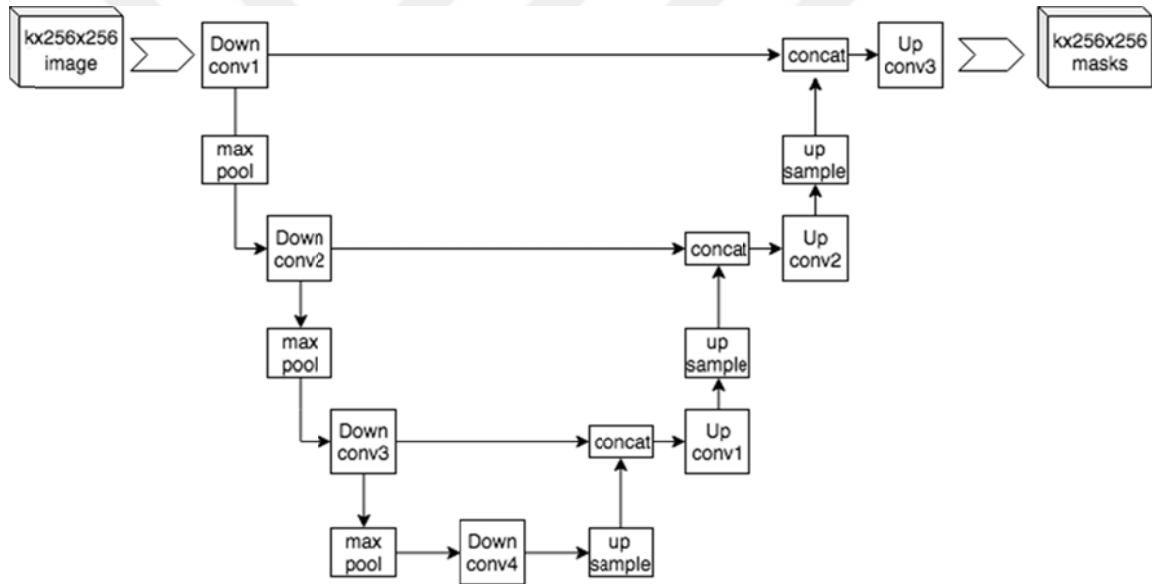
Şekil 3.5. Semantik segmentasyon için Kodlayıcı-Kod çözücü yapısı (Peng ve ark., 2019)

Temel olarak Şekil 3.5’te verilen yapı kullanılarak, derin öğrenmede literatürde yaygın olarak kullanılan temel segmentasyon modelleri UNet, UNet++, SegNet, PSPNet ve sınıflandırma için kullanılan temel modeller ise Evrişimsel Sinir Ağı (ESA) (CNN-Convolutional Neural Network), Patch ESA (PatchCNN) ve Kapsül Sinir Ağı (KSA) (CapsNet-Capsule Network)’dır.

3.1.1. UNet

UNet biyomedikal görüntü segmentasyonunda yaygın olarak kullanılan bir mimaridir ve veri artırma teknikleri kullanarak sınırlı sayıda eğitim örneğinde bile iyi tahminler yapabilecek kadar güçlü olduğu belirtilmiştir (Ronneberger ve ark., 2015). UNet yapısı temel 3 ana bölümden oluşan “U” şekilli bir mimaridir (Şekil 3.6). Şekil 3.6’da giriş görüntüsündeki k değeri görüntü katman sayısı ve 256×256 ise görüntü boyutlarıdır. UNet bölümleri:

- 1) Daralma bölümü (kodlayıcı) (contracting path-encoder)
- 2) Dar geçit bölümü
- 3) Genişleme bölümü (kod çözücü) (expansive path-decoder)



Şekil 3.6. UNet mimari yapısı (Contributors, 2021)

Darlama bölümü tipik bir evrişimsel sinir ağı yapısını izler ve birçok daralma bloğundan oluşur. Her blok, bir girdi görüntüsünü evrişimsel katmandan geçirir ve her birini rektifiye edilmiş doğrusal bir birim (ReLU) ve alt örnekleme için bir adım kaydırma (stride) ile maksimum birleştirme katmanı takip eder. Adım kaydırma, uygulanan filtrenin ana görüntü üzerinde kaç piksel boyunca kayacağını belirler.

Dar geçit bölümü, darlama ve genişleme katmanını birleştirir.

Genişleme bölümü de daralma bölümü gibi birkaç bloktan oluşmaktadır. Her blokta girdi görüntüsü, önce evrişimsel katman ve ardından üst örnekleme katmanından geçirilir. Daralma bloklarının aksine, burada simetriyi korumak için özellik haritalarının sayısı yarı yarıya azalır. Girdi görüntüsü, karşılık gelen daralma katmanına aralıklarla eklenir. Böylelikle, daralma sırasında öğrenilen özelliklerin yeni görüntüyü yapılandırmak için kullanılması sağlanmaktadır. Genişleme bloklarının sayısının, daralma bloklarının sayısı ile aynı olduğuna dikkat edilmelidir. Genişleme bloğunun sonucu daha sonra başka bir evrişim katmanından geçirilir ve burada özellik haritalarının sayısı istenen sayıda segmente veya sınıfa eşittir.

Küçük veri kümelerinde iyi performans göstermesinin yanı sıra, UNet'in bir başka avantajı da farklı boyutlardaki görüntülerin girdi olarak kullanılabilmesidir. UNet'in başarısı, büyük ölçüde kodlayıcıdan sık ve düşük seviyeli özellikleri toplayan ve bunları kod çözücüde aynı çözünürlüğü paylaşan derin ve yüksek seviyeli özelliklere dağıtan atlama bağlantılarından kaynaklanmaktadır.

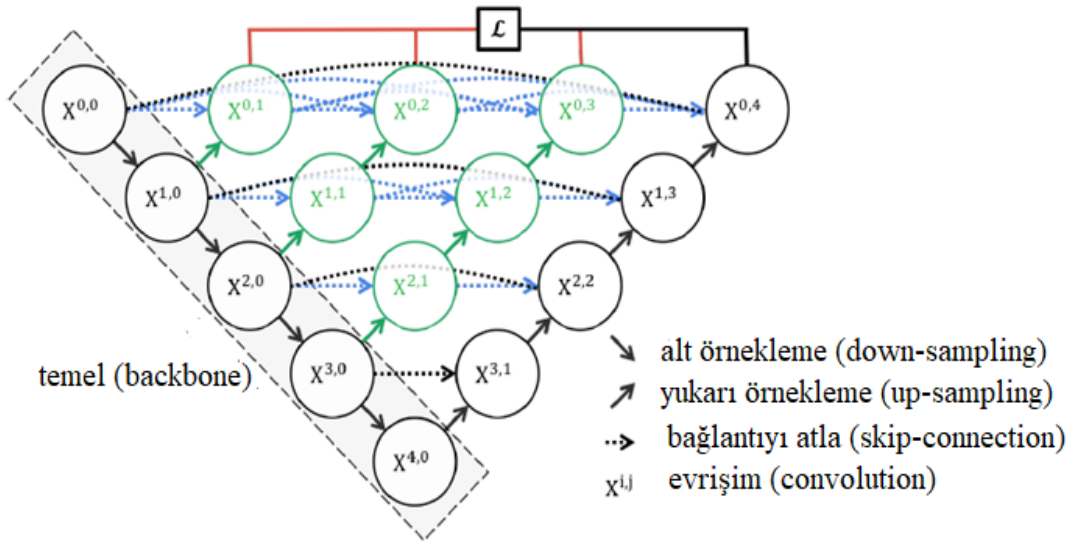
3.1.2. UNet++

Nested UNet olarak da bilinen UNet++, Zhou ve ark.(2019) tarafından UNet mimarisinin segmentasyon performansının iyileştirilmesi amacıyla ileri sürülmüştür. Tasarımının arkasındaki ana fikir, evrişimsel bloklar arasındaki atlama bağlantılarının sayısını artırma yoluyla iç içe geçmiş evrişimsel blokları dahil ederek ve ağ yapısını yeniden tasarlayarak kodlayıcının karşılık gelen evrişimsel blokları ile kod çözücü yolları arasındaki semantik boşluğu arasında köprü oluşturma ihtiyacıdır. Derin ağların mimarisi, özellikle medikal görüntülerin segmentasyonu için uyarlanmış ve geliştirilmiştir ve UNet++'da kodlayıcı ve kod çözücü bölümleri, bu iki bölümün özellik haritaları arasındaki semantik mesafenin azalacağı şekilde yoğun atlama yolları ile bağlanmıştır. Çeşitli tıbbi görüntü görevlerine ilişkin deneysel sonuçlar, UNet++'nın geleneksel UNet mimarisinden daha iyi performans sergilediğini göstermiştir (Zhou ve ark., 2020). UNet++, orijinal UNet'ten üç şekilde farklıdır:

- 1) Kodlayıcı ve kod çözücü özellik haritaları arasındaki semantik boşluğu dolduran atlama yollarına evrişimsel katmanların eklenmesiyle yeniden tasarlanan atlama yollarının eklenmesi

- 2) Gradyan akışını iyileştirmek amacıyla eklenen yoğun atlama bağlantılarının olması
- 3) Model budamasını sağlayan ve yalnızca bir kayıp katmanı kullanmakla karşılaştırılabilir performans sağlayan derin denetimin olması

UNet++'ın genel mimarisi Şekil 3.7'de verilmektedir.

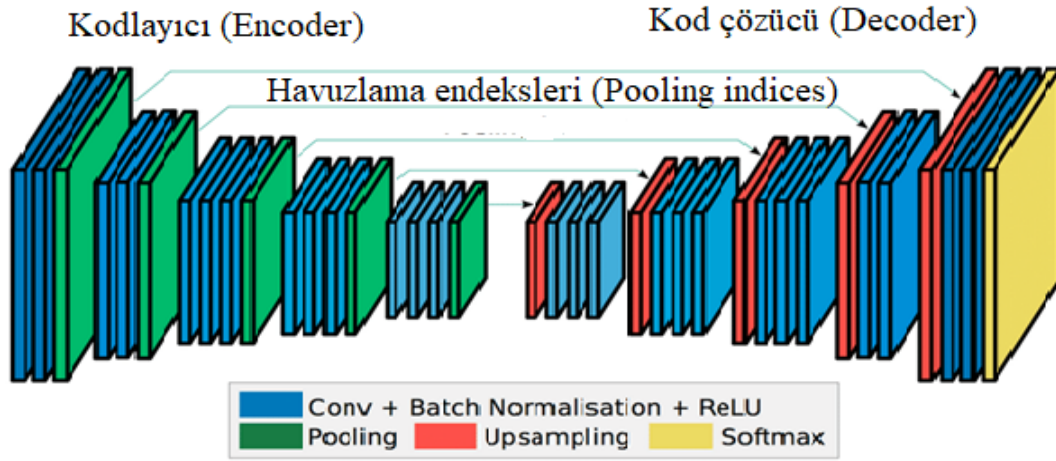


Şekil 3.7. UNet++ mimari yapısı (Zhou ve ark., 2019)

3.1.3. SegNet

SegNet, Şekil 3.8'de gösterildiği üzere, çok sınıflı piksel esaslı segmentasyon için derin bir kodlayıcı-kod çözücü yapısıdır (Badrinarayanan ve ark., 2017). SegNet yapısı, bir kodlayıcıdan, onu takip eden bir kod çözücüden ve son olarak da piksel esaslı bir sınıflandırıcıdan oluşmaktadır. Kodlayıcı; tipik olarak bir veya daha evrişimsel katman, onu takip eden yığın normalizasyon (batch normalization), bir ReLU aktivasyon fonksiyonu ve son olarak maksimum havuzlamadan (maximum or max pooling) oluşur. Kodlayıcı çıktılarına daha yüksek çözünürlüklü özellik haritalarını korumak için doğrusal katmanlar bulunmamaktadır. Kodlayıcı ve kod çözücü, her kodlayıcı katmanının kod çözücüde karşılık gelen bir katmanı olduğundan, aynı sayıda katmana sahiptir. Havuzlama işlemi sırasında meydana gelen özellik haritalarında uzamsal çözünürlük kaybı, seyrek kodlamaya yol açmaktadır. Bu durum, kod çözücüde yukarı örnekleme (upsampling)

yapmak için kodlama dizisindeki maksimum havuzlama indeksler kullanılarak düzeltilir. Bölümlenmiş görüntülerde yüksek frekans detaylarını korumanın yanı sıra, bu işlem kod çözücündeki toplam eğitilebilir parametre sayısını da azaltmaktadır. Ayrıca bu üst örnekleme biçimi, yalnızca küçük bir değişiklikle herhangi bir kodlayıcı-kod çözücü mimarisine dahil edilebilir.

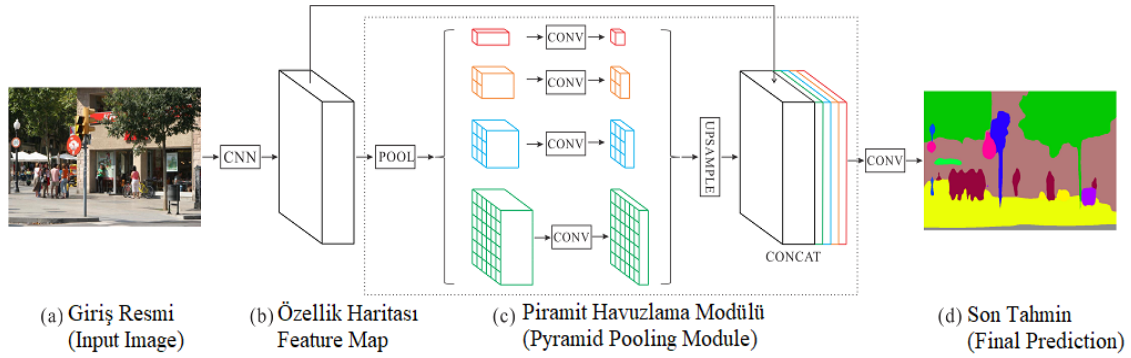


Şekil 3.8. SegNet mimari yapısı (Badrinarayanan ve ark., 2017)

3.1.4. PSPNet

Long ve ark.(2015) tarafından öne sürülen Tam Evrişimli Ağ (TEA) (FCN-Fully Convolution Network), düşük seviyeli görüntü bilgilerini yüksek seviyeli semantik bilgilerle birleştirerek segmentasyon gerçekleştiren bir sinir ağı mimarisidir. TEA, her pikseli başarılı bir şekilde etiketlemesine rağmen, uzamsal tutarlılıktan yoksun olduğu için pikseller arasındaki ilişkiyi tam olarak dikkate almamaktadır. Zhao ve ark. (2017), bu sınırlamayı gidermek için 2016 yılında Piramit Sahne Ayrıştırma Ağı'nı (PSPNet) önermişlerdir. Önerilen yöntemde, segmentasyonu gerçekleştirmek için ResNet ve genişletilmiş bir ağ kombinasyonu kullanılmaktadır. İlk olarak, girdi görüntüsü üzerinde özellik çıkarımı yapılır ve çıkarılan özellik haritasının boyutu orijinal görüntünün belirlenen bir oranı kadar olur. Özellik haritası daha sonra piramit havuzlama modülüne verilir ve son olarak özellik yüzeyini elde etmek için kademelendirilir. Öğrenme, hiperparametreleri ayarlayarak ve geriye yayılım algoritması ile kayıp hesaplayarak yapılır.

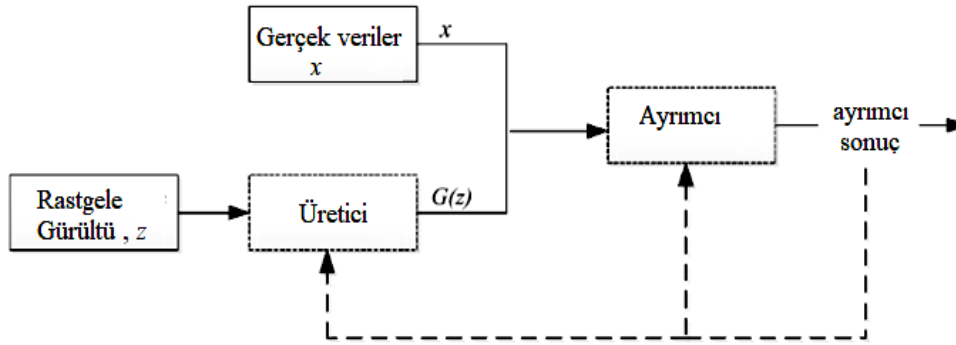
Şekil 3.9, PSPNet mimarisini göstermektedir. Ön işlemeden sonra, özellik haritası giriş görüntüsünden çıkarılır ve ardından piramit havuzlama katmanına gönderilir. Piramit havuzlama katmanı, özellik haritalarını farklı boyutlarda evriştirir, ardından çıktıyı kademelendirir ve entegre eder. Sonrasında diğer evrişimlerden ve seyreltmelerden (dropout) geçerek pikselleri etiketlemek ve son olarak segmentasyon haritasını elde etmek için bir sınıflandırıcı kullanılır.



Şekil 3.9. PSPNet mimari yapısı (Zhao ve ark., 2017)

3.1.5. Üretken Çekişmeli Ağ ve Koşullu Üretken Çekişmeli Ağ

Bir Üretken Çekişmeli Ağ (ÜÇA) (GAN-Generative Adversarial Networks), birbiriyle rekabet etmek üzere eğitilmiş bir “üretici” (generator) ve bir “ayırıcı” (discriminator) oluşur (Çelik ve Talu, 2021). Üretici (G), ayırıcıyı (D) kandırmak veya şaşırtmak için gerçek verilerin (x) dağılımından, gerçeğe yakın ama sahte veriler üretmek üzere eğitilir. Ayırıcı (D)’nin amacı ise gerçek veri ile üretici (G) tarafından üretilen sahte veri arasında ayırım yapmaktır. Üretici, yalnızca ayırt ediciden geri yayılan gradyanlar aracılığıyla güncellenmektedir. ÜÇA’da, hem üretici (G) hem de ayırıcı (D) yinelemeli olarak güncellenerek; ayırıcının, üretici tarafından üretilen gerçek ve sahte verileri birbirinden ayırt edemediği bir dengeye ulaşılır. Şekil 3.10’da ÜÇA’nın temel yapısı olan Vanilya ÜÇA görülmektedir.



Şekil 3.10. Vanilya ÜÇA (Hu ve ark. n.d.)

Matematiksel terimlerle üretici G ve ayrımcı D , Denklem 3.6 ile verilen amaç fonksiyonu ile iki oyunculu bir min-max algoritması uygulamaktadır (Goodfellow ve ark., 2014; Denton ve ark., 2015)

$$L_{\text{ÜÇA}}(G, D) = \mathbb{E}_{x \sim P_{\text{data}(x)}} [\log(D(x))] + \mathbb{E}_{z \sim p_{\text{data}(z)}} [1 - \log(D(G(z)))] \quad (3.6)$$

Burada z rastgele gürültü ve x giriş veri bilgileridir.

Evrişimsel Sinir Ağ'ların (ESA) (CNN), ÜÇA'lar aracılığıyla ayırt edici bir şekilde eğitilmesinin birçok avantajı vardır ve iyi sonuçlar vermektedir; ancak bunun da zorlukları mevcuttur:

Mod Çöküşü: Mod çöküşü, üreticinin girdiden bağımsız olarak sınırlı bir çeşitlilikte örnek veya aynı örneği üretmesidir. Bu, üreticinin girilen gürültüden bağımsız olarak, üretim yapmanın en uygun nokta olduğunu düşündüğü tek bir sabit nokta olduğu anlamına gelir.

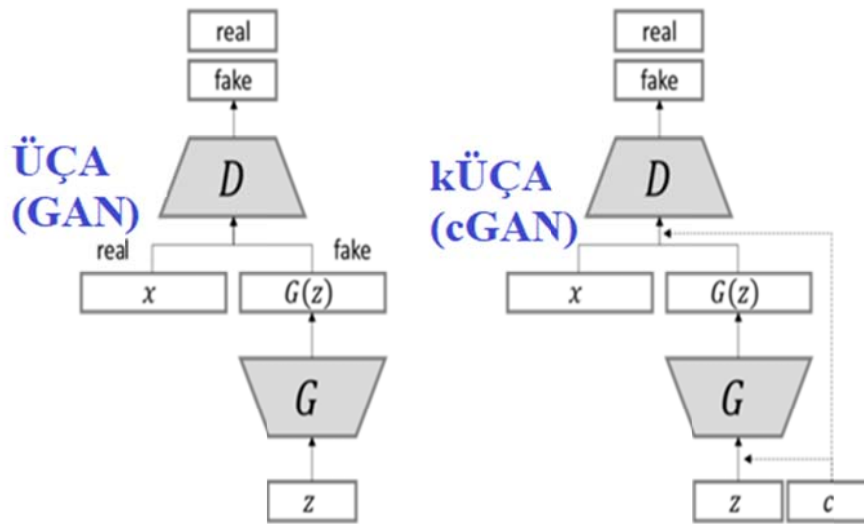
İstikrarsızlık: İstikrarsızlık, aynı girdi için farklı çıktılar üretilmesidir. Bu, optimizasyon sürecinde kaybolan gradyanlar nedeniyle gerçekleşen bir durumdur.

Yığın normalleştirme (Batch normalization) her katmana girişin sıfır ortalaması ve birim varyansı olmasını sağlar. Yığın normalleştirme ÜÇA'lardaki istikrarsızlık için bir çözüm olarak ileri sürülmüştür. Ancak ÜÇA performansını en üst düzey istikrara getirmek için yeterli değildir. Bu nedenle, bu sorunu çözmeye çalışmak için birçok ÜÇA varyasyonu öne sürülmüştür ve kÜÇA yapılan çalışmalarda kullanılan bu varyasyonlardan biridir.

ÜÇÄ'nin kısıtlamalarından biri, gerçek görüntü üretimi üzerinde hiçbir kontrolünün olmamasıdır. Mirza ve Osidero tarafından ileri sürülen kÜÇÄ, bu sorunun üstesinden gelmeye çalışmaktadır. kÜÇÄ yoluyla, tür etiketleri gibi ek bilgiler görüntü oluşturma sürecine dâhil edilebilir (Mirza and Osidero, 2014). kÜÇÄ'da üreticinin 2 girişi vardır. Bunlar, rastgele gürültü z ve ön bilgiler c 'dir. Daha sonra ön bilgi c , karşılık gelen gerçek veya sahte verilerle birlikte ayırıcıya yüklenir. Matematiksel olarak kÜÇÄ yapısı Denklem 3.7 ile ifade edilir.

$$L_{kÜÇÄ}(G, D) = \mathbb{E}_{x \sim P_{data(x)}}[\log(D(x|c))] + \mathbb{E}_{x \sim p_{z(z)}}[1 - \log(D(G(z|c)))] \quad (3.7)$$

Şekil 3.11, ÜÇÄ ve Koşullu Üretken Çekişmeli Ağ (kÜÇÄ) arasındaki mimari farklılığı göstermektedir.

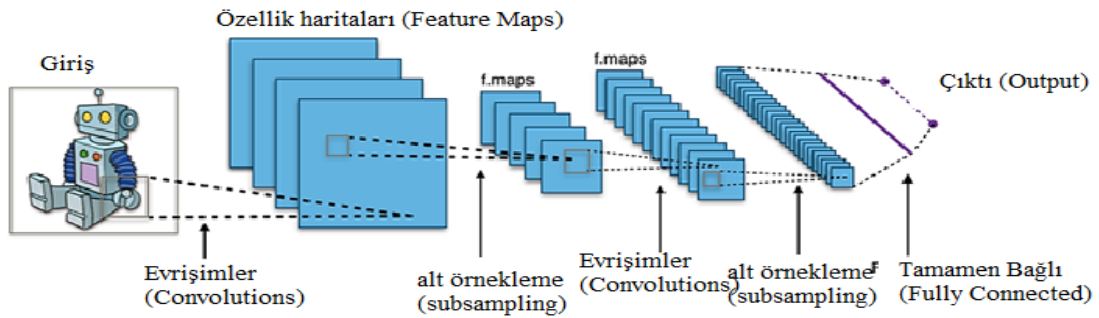


Şekil 3.11. ÜÇÄ ve kÜÇÄ arasındaki mimari farklılık (Mino and Spanakis, 2018)

Araştırmacılar, kÜÇÄ'nin yalnızca ayrıntılı özelliklerin üretimini değil, aynı zamanda eğitim sırasındaki istikrarı da arttırdığını belirtmektedirler. Bu nedenle tez kapsamında kÜÇÄ tercih edilmiştir.

3.1.6. Evrişimsel Sinir Ağı (ESA)

Evrişimsel Sinir Ağı (ESA) (CNN-Convolutional Neural Network), her katmanda bir dizi özellik (kenarlar, köşeler, şekiller vb.) biriktirmesi sebebiyle görüntü sınıflandırması için yaygın olarak kullanılan derin öğrenme mimarisidir. Bir ESA temel olarak üç ana katmandan oluşur (Şekil 3.12).



Şekil 3.12. Temel ESA yapısı (contributors, 2021)

1) Evrişim katmanı (Convolutional Layer): Genellikle paralel olarak birkaç evrişimi gerçekleştirmektedir. Özellikleri saptamak için kullanılır.

2) Havuzlama katmanı (Pooling Layer): ESA'ların bir diğer önemli kavramı, doğrusal olmayan alt örneklemenin (subsampling) bir biçimi olan havuzlamadır. Maksimum havuzlamanın en yaygın olduğu havuz oluşturmayı uygulamak için birkaç doğrusal olmayan işlev vardır. Girdi görüntüsünü bir dizi dikdörtgene böler ve her bir alt bölge için uygunluğu kontrol ederek, maksimum çıktıyı verir.

3) Tam bağlantılı katman (Fully Connected Layer): Sınıflamada kullanılan standart sinir ağı mimarisidir.

Evrişim katmanı, ESA'nın yapı taşıdır. Görüntüye belirli bir filtre uygulayarak, bazı görüntülerdeki düşük seviyeli ve sonrasında daha yüksek seviyeli öznitelikleri tespit etmek için kullanılmaktadır. Düşük seviyeli özellikler (low level features), çizgiler veya noktalar gibi görüntünün küçük ayrıntıları iken, görüntüdeki nesneleri ve daha büyük şekilleri algılamak için düşük seviyeli özelliklerden yüksek seviyeli özellikler (high level features) oluşturulur.

“Filtre” genellikle, özellik haritaları oluşturmak için bir girdi verisi dizisiyle çarpılan çok boyutlu piksel değerleri dizisini ifade etmektedir. Adım kaydırma (stride) ve piksel ekleme (padding), evrişimli bir katmanda kullanılan diğer hiper parametrelerdir. Piksel ekleme, filtreler giriş görüntüsünün boyutuna uymadığında kullanılırken; adım kaydırma filtrenin giriş görüntüsü üzerinde hareket ettiği piksel sayısıdır.

Bir özellik haritası oluşturulduğunda, özellik haritasındaki her bir değerin doğrusal olmayanlığını ağa tanıtan ReLU gibi bir aktivasyon fonksiyonundan geçirilebilir. Aktivasyon fonksiyonunun çıktısı, havuzlama katmanına iletilir. Havuzlama katmanı, temsilin uzaysal boyutunu küçültmek ve böylece ağdaki parametre sayısını azaltmak için kullanılmaktadır. Bazı havuzlama türlerine örnek olarak Maksimum Havuzlama, Ortalama Havuzlama, Global Havuzlama vb. sayılabilmektedir. Havuzlama katmanındaki özellik haritaları, Düzleştirme (Flattening) katmanında tek boyutlu dizi değerlerine dönüştürülür. Düzleştirme katmanının çıkışı, ağın son katmanı olan Tam Bağlantılı Katmana iletilir. Sınıflandırma bu katmanda gerçekleştirilir.

3.1.7. Yama Evrişimsel Sinir Ağı (PatchESA)

ESA, genellikle tüm görüntüleri sınıflandırmak için kullanılmaktadır. PatchESA’da, önemli teşhis bilgilerini taşıyan uygun boyutlardaki “yama”lar orijinal görüntüden çıkarılır ve her bir yama ayrı ayrı sınıflandırılır. Her bir test görüntüsü öncelikle $\mathbb{P}$ gibi bir dizi yamaya bölünür ve her yama $i \in \mathbb{P}$ için ESA kullanılarak olasılıksal tahmin yapılır. Bu tahminler daha sonra bir oylama şeması kullanılarak nihai görüntü etiketiyle birleştirilir. Çoğunluk oylaması, maksimum olasılık ve olasılıkların toplamı; yama etiketlerini birleştirmek için kullanılabilecek oylama şemalarına örnekler olarak verilebilmektedir. Çoğunluk oylamasında, en çok oyu alan etiket, nihai görüntü sınıfı olarak seçilir (Denklem 3.8).

$$v_k = \sum_{i \in \mathbb{P}} (l_i = k) \quad (3.8)$$

Burada k , sınıf etiketine oy veren yamaların sayısını, l_i , i yamasının sınıf etiketini temsil etmektedir. En çok oy alan yama etiketi Denklem 3.9 kullanılarak seçilmektedir.

$$\check{k} = \arg \max_{k \in \mathbb{K}} (v_k) \quad (3.9)$$

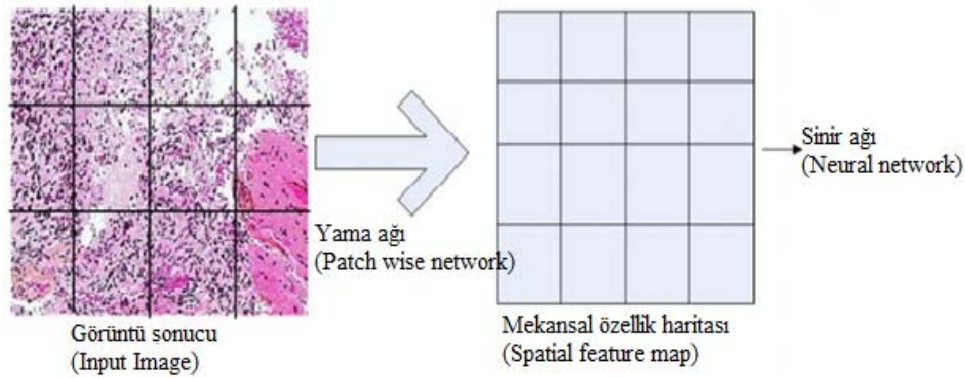
Maksimum olasılıkta (Denklem 3.10); olasılıklar toplamıyla yama sınıfı olasılıkları toplanır ve en yüksek toplama sahip sınıf, Denklem 3.11’de gösterildiği gibi seçilirken, en yüksek sınıf olasılığına sahip olan yama, görüntü sınıfına karar vermektedir.

$$\check{k} = \arg \max_{k \in \mathbb{K}} \left(\max_{i \in \mathbb{P}} (p_i[k]) \right) \quad (3.10)$$

$$\check{k} = \arg \max_{k \in \mathbb{K}} \left(\sum_{i \in \mathbb{P}} p_i[k] \right) \quad (3.11)$$

Burada, $p_i[k] = \mathbb{P}$ (yama $i \in$ sınıf k).

Şekil 3.13, bir görüntüden çıkarılan ve bir sinir ağına giren yamaları göstermektedir.



Şekil 3.13. Temel patch çıkarımı ve sinir ağına giriş (Sumi ve Delhibabu, 2019)

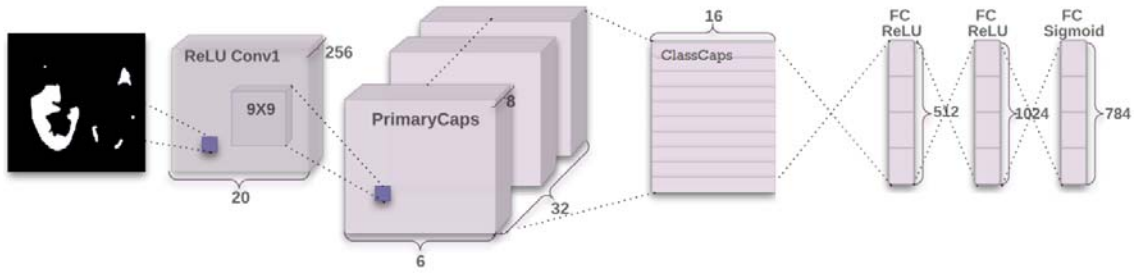
3.1.8. Kapsül Sinir Ağları (KSA)

Evrimsel Sinir Ağları (ESA) nesnelerin kesin konumu ve duruşu gibi bilgileri kaybetme eğiliminde olan birkaç havuzlama katmanına sahiptir. Kapsül Sinir Ağı (KSA) (CapsNet), ters grafik fikrini kullanarak ESA’lardaki bu dezavantajların üstesinden gelmek için 2017 yılında Geoffrey Hinton (Sabour ve ark., 2017) tarafından ileri sürülmüştür. ESA’ların aksine, KSA ağ boyunca bir nesnenin konumu ve duruşu hakkında ayrıntılı bilgileri muhafaza etmektedir. Bir kapsül ağı, kapsüllerden oluşur. Kapsül, bir görüntüdeki bir nesneyi veya bir nesnenin parçalarını algılamayı öğrenen bir grup nöronudur. Çıktısı,

uzunluđu bir görüntüdeki bir nesnenin varlığını temsil eden ve yönelimi bir görüntüdeki nesnenin konumunu, dönüşünü, boyutunu veya ölçeğini temsil eden bir vektördür. Bu durum, özellikler arasındaki uzamsal ilişkileri görmezden gelen ESA’lardan farklıdır.

KSA mimarisi bir kodlayıcı ve bir kod çözücünden oluşmaktadır. Kodlayıcı; Evrişim Katmanı, Birincil Kapsül Katmanı ve ClassCaps Katmanı olmak üzere 3 katman içermektedir. Evrişim Katmanı, $9 \times 9 \times 1$ boyutunda ve 1 adımlı 256 çekirdeğe ve ardından ReLU aktivasyon fonksiyonuna sahiptir. Bu katmanda, görüntü piksel yoğunlukları yerel özellik algılayıcılara dönüştürülmektedir. Bu katmanın çıktısı daha sonra, her bir kapsülün boyutu 6×6 olan 32 birincil kapsül içeren bir evrişimsel katman olan Birincil Kapsül Katmanına girdi olarak verilir. Görüntüdeki en düşük seviyeli varlıklar bu katmanda yakalanmaktadır. Bu katmanın çıktısı daha sonra dinamik bir yönlendirme algoritması aracılığıyla ClassCaps katmanına gönderilir. Dinamik yönlendirme algoritması, kapsüllerin sonraki uygun katmanlara yönlendirilmesini kolaylaştırırken, Birincil Kapsül ve ClassCaps katmanları arasındaki bağlantıyı ölçmek için bir bağlantı katsayısı hesaplar. CapsNet modeli, eğitim sürecinde güncellendiği için katsayı değeri aracılığıyla öğrenir ve kaybı daha hızlı optimize etmek ve ezberlemeyi önlemek için yalnızca 3 yönlendirme yinelemesi kullanılır. ClassCaps katmanında, her sınıf için bir tane olmak üzere 2 sınıf kapsülü bulunur. Buradaki her girdi vektörüne, girdi alanını 16 boyutlu kapsül çıktı alanına eşleyen bir 8×16 ağırlık matrisi uygulanır.

KSA’daki kod çözücü ağıının amacı, vektörleri ClassCaps katman temsilinden yeniden oluşturmaktır. Bunu, girdi görüntüsü ile kod çözücünün çıktısı arasındaki Öklid mesafesini en aza indirerek yapar. Kod çözücü 3 tam bağlantılı katmandan oluşur; birinci katman toplam 512 nöron, ikinci katman 1024 nöron ve üçüncü katman 784 nöron içermektedir. KSA parametreleri, her çıktı sınıfı için bir marj kaybı kullanılarak optimize edilir. Marj kaybına ek olarak, girdi görüntüsünün parametrelerini kodlamak amacıyla her bir kapsülü artırmak için bir düzenleme yöntemi olarak $L2$ yeniden oluşturma kaybı (ortalama kare hatası) kullanılır. KSA mimarisi Şekil 3.14’te gösterilmektedir.



Şekil 3.14. Temel KSA mimari yapısı (Sabour ve ark., 2017)

3.2. Metrikler

Performans ölçümleri, öğrenme algoritmalarının performansını yani başarısını değerlendirmede kullanılmaktadır. Farklı ölçüm metrikleri, modelleri nicel ve nitel olarak değerlendirmek için kullanılmaktadır. Görüntü analizinde kullanılan temel nicel analiz metrikleri Zar Benzerlik Skoru (Zar Skoru) (Dice Score), Birlik Üzerinden Kesişme (BüK) (IoU-Intersection over Union), Kesinlik (Precision), Duyarlılık (Recall), Özgüllük (Specifity), F1-skor (F1-score) ve Alıcı İşletim Karakteristiği (ROC-Receiver Operating Characteristic) ölçütleridir. Nitel analiz ise elde edilen görsel sonuçların temel gerçek görsellerle karşılaştırılmasıyla gerçekleştirilir.

Zar Skoru ve BüK metrikleri, literatürde çok bilinen ve özellikle tıbbi görüntü segmentasyonu için değerlendirmede kullanılan temel metriklerdir (Voiculescu ve Yeghiazaryan, 2015). Her iki ölçümün daha yüksek puanlar sergilemesi, daha iyi segmentasyon performansına işaret etmektedir. F1-skoru gibi Zar Skoru, hedef görüntü ile segmentasyon sonucunun karşılaştırması için kullanılan bir örtüşme ölçüsüdür ve Denklem 3.12’de olduğu gibi tanımlanmaktadır.

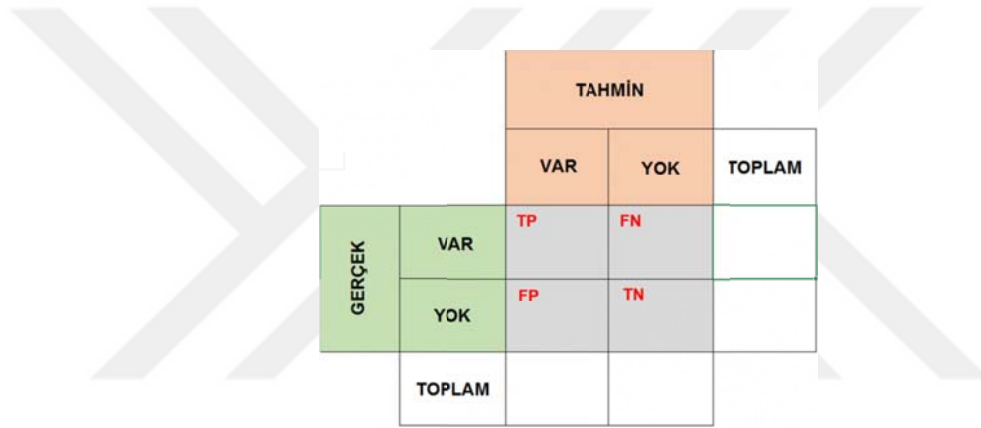
BüK ise, temel gerçek görüntü ile segmentasyon sonucu arasındaki örtüşme yüzdesini ölçmek için kullanılan yaygın bir ölçümdür. Aynı zamanda “Jaccard indeksi” olarak da adlandırılır ve Denklem 3.13’te tanımlandığı gibi gerçek ve sistem sonucu arasındaki benzer piksellerin sayısının her iki görüntüde bulunan toplam piksel sayısına bölünmesiyle ölçülür.

$$Zar\ Skoru = \frac{2 * (A \cap B)}{A + B} \quad (3.12)$$

$$BüK = \frac{A \cap B}{|A| \cup |B|} * 100 \quad (3.13)$$

Denklem 3.12 ve 3.13’de, A ve B sırasıyla gerçek ve segmentasyon sonucu olan görüntüyü ifade etmektedir.

Kesinlik ve duyarlılık metrikleri, makine öğrenmesi alanında elde edilen sonuç (tahmin) değerlerin gerçek değerler ile benzerliklerini yansıtmaktadır. Bu değerler “karışıklık matrisi” (confusion matrix) adı verilen bir matris yapısında değerlendirilmektedir (Şekil 3.15). Görüntü analizinde bu ölçüm sonuçları, temel olarak yanlış ve doğru sınıflandırılan piksellerin sayısına dayanmaktadır.



The diagram illustrates the structure of a confusion matrix. It features a central 2x2 matrix with a green header 'GERÇEK' on the left and an orange header 'TAHMİN' on top. The rows are labeled 'VAR' and 'YOK' under the 'GERÇEK' header, and the columns are labeled 'VAR' and 'YOK' under the 'TAHMİN' header. The cells contain 'TP' (True Positive), 'FN' (False Negative), 'FP' (False Positive), and 'TN' (True Negative) in red text. To the right of the central matrix is a column labeled 'TOPLAM' (Total). Below the central matrix is a row labeled 'TOPLAM' (Total). The entire matrix is set against a background with large, light gray 'X' marks.

		TAHMİN		TOPLAM
		VAR	YOK	
GERÇEK	VAR	TP	FN	
	YOK	FP	TN	
TOPLAM				

Şekil 3.15. Karışıklık matris yapısı

Bu matris yapısında sonuçlar, Gerçek Pozitif (GP) (TP-True Positive), Gerçek Negatif (GN) (TN-True Negative), Yanlış Pozitif (YP) (FP-False Positive) ve Yanlış Negatif (YN) (FN-False Negative) cinsinden ifade edilebilmektedir. Burada:

- 1) GP: Modelin pozitif sınıfına ait örneklerin kaç tanesini doğru bir şekilde pozitif olarak tanımladığını,
- 2) GN: Modelin negatif sınıfa ait örneklerin kaç tanesini doğru bir şekilde negatif olarak tanımladığını,
- 3) YP: Modelin negatif sınıfa ait örneklerin kaç tanesini yanlış bir şekilde pozitif olarak tanımladığını,
- 4) YN: Modelin pozitif sınıfa ait örneklerin kaç tanesini yanlış bir şekilde negatif olarak tanımladığını göstermektedir.

Kesinlik değeri (Denklem 3.14), modelde doğru tanımlanmış pozitif örneklerin (GP) toplam sayısının, pozitif sınıfa (GP+YP) ait olarak tanımlanan toplam eleman sayısına oranıdır. Yüksek hassasiyet, pozitif olarak tanımlanan bir örneğin gerçekten de pozitif olduğunu göstermektedir. Ancak, görüntü analizinde alt segmentasyon hataları kesinliğe yansımamaktadır. Duyarlılık ise (Denklem 3.15), modelde doğru tanımlanmış pozitif örneklerin (TP) toplam sayısının gerçek pozitif sınıfa (GP+YN) ait olan toplam eleman sayısına oranı olarak tanımlanabilir.

Karışıklık matrisinde kullanılan bir diğer karşılaştırma metriği ise özgüllüktür. Özgüllük, modelin gerçek negatif verileri doğru olarak tanımlama oranıdır. Özgüllük (Denklem 3.16), doğru tanımlanmış negatif örneklerin (GN) toplam sayısının, gerçekte negatif sınıfa (GN+YP) ait olan toplam eleman sayısına oranıdır.

Görüntü analizinde, duyarlılık, segmentasyondaki aşırı segmentasyon hatalarını hesaba katmasa da (Voiculescu ve Yeghiazaryan., 2015) yüksek bir duyarlılık, düşük sayıda yanlış negatif olduğunu göstermektedir.

F1-skoru, kesinlik ve duyarlılığın harmonik ortalamasıdır ve yalnızca hem kesinlik hem de duyarlılık yüksek olduğunda yüksek değer almaktadır (Denklem 3.17). Özellikle görüntü segmentasyonunda gerçek görüntü ile sonuç görüntü arasındaki örtüşme başarısını ölçmek için F1-skor kullanılması, model sonucunu değerlendirmek için daha güvenlidir.

Alıcı işletim karakteristiği (AİK) (ROC) eğrisi, iki sınıflı sistemlerde, sınıf ayırım eşik değerinin farklılık gösterdiği durumlarda, hassasiyetin kesinlik değerine olan oranıyla hesaplanmaktadır. AİK, daha basit anlamda doğru pozitiflerin, yanlış pozitiflere olan oranı diye tanımlanabilir. AİK eğrisi altında kalan alan, farklı modeller için ölçülmektedir ve çeşitli eşik değerlerinde, model performans ölçümünü gösteren bir grafik çizimidir.

$$Kesinlik = \frac{GP}{GP + YP} \quad (3.14)$$

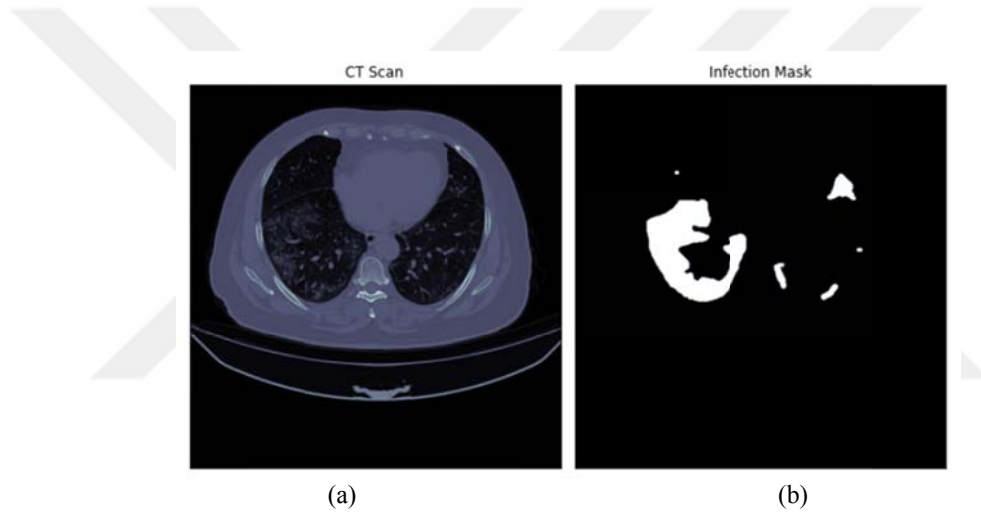
$$Hatırlama(Duyarlılık) = \frac{GP}{GP + YN} \quad (3.15)$$

$$Özgüllük = \frac{GN}{GN + YP} \quad (3.16)$$

$$F1 Skoru = \frac{2 \cdot GP}{2 \cdot GP + YP + YN} \quad (3.17)$$

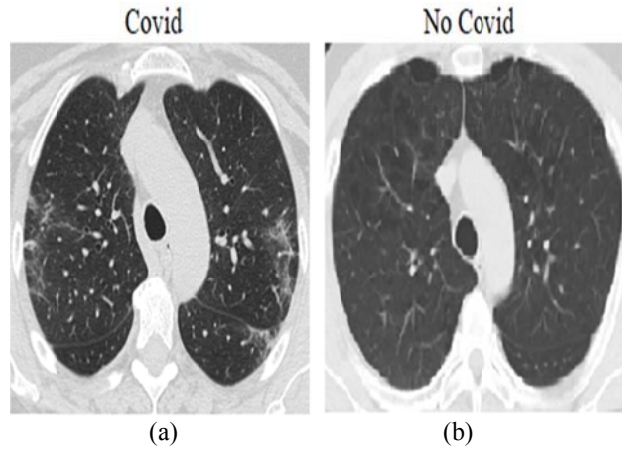
3.3. Veri Setleri

Tez çalışmasında önerilen yöntemin segmentasyon ve sınıflandırma başarısını doğrulamak için, halka açık iki farklı veri kümesi kullanılmıştır. Segmentasyon için kullanılan ilk veri kümesi (Veri Kümesi-1) (Paiva, 2020), 20 Bilgisayarlı Tomografi (BT) görüntüsünün yanı sıra uzmanlar tarafından yapılan akciğer segmentasyonları ve enfeksiyon maskelerini içermektedir. Veri Kümesi-1’de toplam 2112 kesit bulunmaktadır. Bunların 1615 tanesi Covid-19 teşhisi konulmuş kişilerden alınan kesitler ve 497 tanesi ise sağlıklı kişilere ait görüntülerdir. Şekil 3.16, veri kümesinden alınan bir BT görüntü örneğini ve buna karşılık gelen maskeyi göstermektedir.



Şekil 3.16. (a) Veri kümesi-1’den örnek BT görüntüsü, (b) görüntüye ait ikili maske

Önerilen yöntemin sınıflandırma başarısını doğrulamak için kullanılan diğer veri seti (Veri Kümesi-2), Angelov ve Soares (2020) tarafından oluşturulmuştur ve veriler Brezilya hastanelerindeki gerçek hastalardan toplanmıştır (Sars-Cov-2 CT-Scan-Ver2). Toplamda 2481 BT görüntüsü içermektedir. Bunların 1252 tanesi Covid-19 için pozitifken, 1229 tanesinin BT taraması negatiftir. Ancak Veri Kümesi-2’de sınıflandırılmış görüntüler bulunurken, görüntü segmentasyonu için ikili maskeler bulunmamaktadır. Bu nedenle Veri Kümesi-2’de sınıflandırma başarısını test etmek için, Veri Kümesi-1’deki ikili maskeler kullanılarak yapılan segmentasyon sonuçlarından en iyi sonucu veren derin öğrenme yapıları segmentasyon için kullanılmıştır. Şekil 3.17, Covid-19 ile enfekte olmuş ve olmamış iki hastaya ait örnek görüntüler göstermektedir.



Şekil 3.17. Veri kümesi-2’den (a) Covid-19 enfekte BT görüntüsü, (b) Covid-19 enfekte olmamış BT görüntüsü

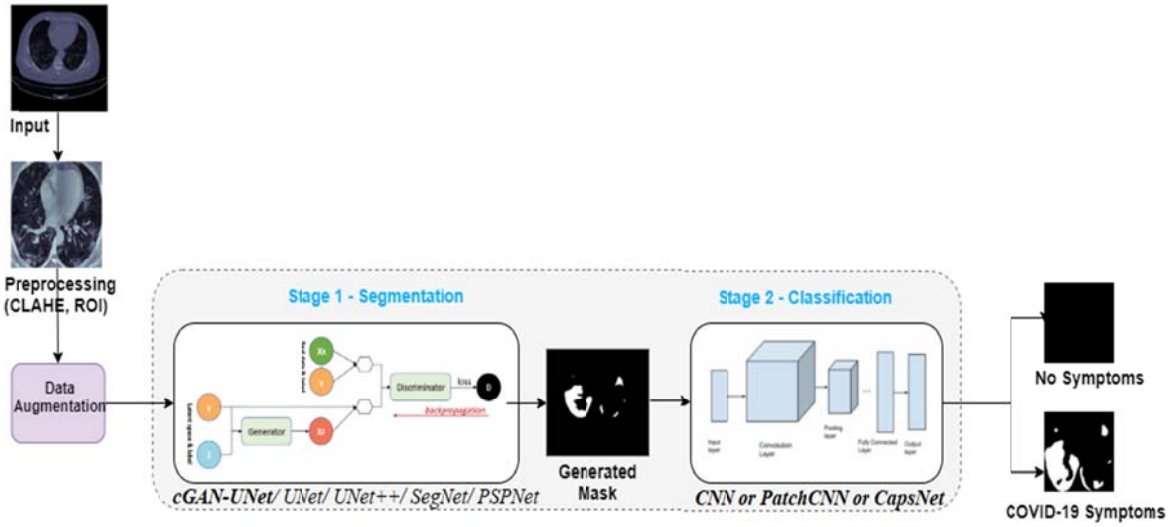
Veri Kümesi-1, segmentasyon modelinin performansını eğitmek ve değerlendirmek için kullanılmıştır. Elde edilen segmentasyon modeli, daha sonra Veri Kümesi-2’de sınıflandırma işleminde, segmentasyon aşamasında kullanılan maskeyi oluşturmak amacıyla kullanılmıştır. Çizelge 3.1, her iki veri kümesinin özetini göstermektedir.

Çizelge 3.1. Tez kapsamında kullanılan veri setlerindeki görüntü bilgileri

	Covid-19 hasta	Sağlıklı	Toplam	Kullanılma amacı
Veri Kümesi-1	1615	497	2112	Segmentasyon
Veri Kümesi-2	1252	1229	2481	Sınıflandırma

4. GELİŞTİRİLEN YÖNTEM

Tez çalışmasında önerilen yöntem, segmentasyon ve sınıflandırma olarak iki ana aşamada (Şekil 4.1) yapılmıştır. Hem segmentasyon hem de sınıflandırma, farklı derin öğrenme teknikleri kullanılarak gerçekleştirilmiştir.



Şekil 4.1. Önerilen yöntemin adımları

Geliştirilen derin öğrenme modelinde veriler, modele girmeden önce ön işleme aşamalarına tabi tutulmuştur. Tez çalışmasında, ön işlemten sonra veriler, segmentasyon aşamasında Covid-19 BT görüntülerinde ikili bir lezyon maskesi elde etmek üzere öğrenme modeline verilmiştir. Segmentasyon modeli olarak UNet, UNet++, SegNet, PSPNet ve kÜÇA-UNet modelleri kullanılmıştır. Segmentasyon modelinden elde edilen segmentlenmiş görüntüler, daha sonra sınıflandırma aşamasına girdi olmuştur. Bu aşamada, bir hastanın Covid-19 olup olmadığını tahmin etmek için Evrişimsel Sinir Ağı (ESA) (CNN), PatchESA (PatchCNN) ve Kapsül Sinir Ağı (KSA) (CapsNet) modelleri kullanılmıştır. Covid-19 içeren segmentlenmiş görüntü siyah zemin üzerine beyaz alanlara sahipken, Covid-19 içermeyen segmentlenmiş görüntüler tamamen siyahtır. Algoritmanın adımları Şekil 4.2’te verilmiştir.

Algoritma Adımları
1) Veri Kümesi-1'den görüntü al 2) Orijinal görüntülerden 2D görüntü dilimlerini çıkar: $x \in D: \{x_i\}_{i=1}^{2112}$ 3) Görüntüleri gri ölçeğe dönüştür, 256x256 piksele yeniden boyutlandır ve normalleştirme ile siyah-beyaz görüntüye çevir. ($x_{pre} = \text{Önişlem}(x)$) 4) Görüntüleri geliştirmek için KSAHE uygulama ($x_{KS} = \text{KSAHE}(x_{pre})$) 5) Görüntü üzerine konturlar çizerek ve ardından kapalı alanı kırparak İB'yi çıkar ($x_{IB} = \text{İB}(x_{KS})$) 6) Döndürme, kesme, yakınlaştırma ve yatay çevirme gibi standart yöntemleri kullanarak veri artırma stratejilerini uygula 7) Verileri eğitim için %80 ve test için %20 olacak şekilde böl 8) Segmentasyon modelini oluştur (UNet/UNet++/SegNet/PSPNet/kÜÇA-UNet), sınıflandırıcıya girdi olarak kullanılacak segmentasyon maskelerini elde etmek için eğitim veri kümesini kullanarak modeli eğit 9) Segmentasyon modelini değerlendir, test et ve diğer temel modellerle karşılaştırarak, sonuçları değerlendir 10) Seçilen eğitilmiş segmentasyon modeli ile Veri Kümesi-2'de segmentasyon maskeleri oluştur 11) Oluşturulan ikili maskeleri kullanarak sınıflandırma modellerini (ESA/PatchESA/KSA) eğit 13) Tüm sınıflandırıcıların performansını değerlendir, test et ve karşılaştırma yap

Şekil 4.2. Tez kapsamındaki algoritmanın adımları

4.1. Önişlem

Görüntü işlemede önişlem aşaması, ham görüntülerin, yapılan uygulamanın sonraki adımlarında sorunsuz kullanılabilmesi için çeşitli düzeltme işlemlerine tabi tutulmalarıdır. Tez çalışmasında, ilk aşama olarak veri setindeki görüntülere bazı önişlem adımları uygulanmıştır.

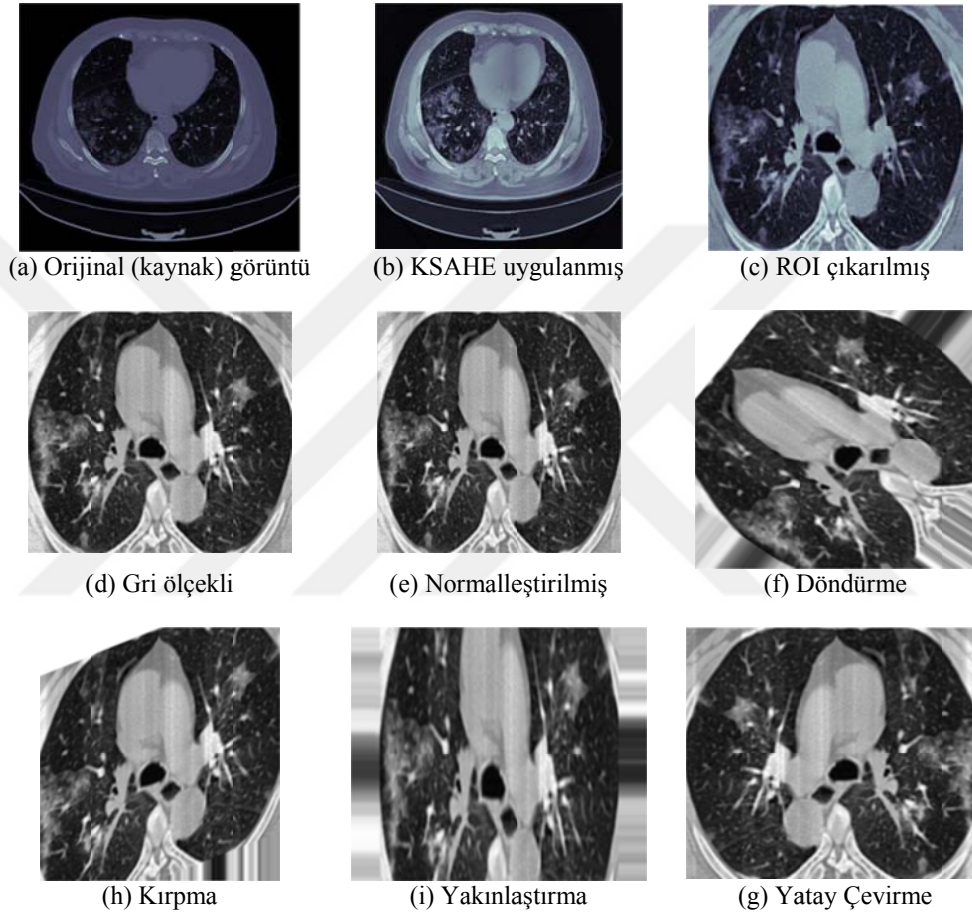
Önişlem aşamasında ilk olarak, alınan Bilgisayarlı Tomografi (BT) taramalarından 2D kesitler çıkarılmıştır. Tıbbi görüntülerin düşük kontrast, gürültü, sinyal düşmeleri ve karmaşık anatomik yapılar gibi özellikleri nedeniyle tıbbi görüntü analizi genellikle zor olmaktadır (Dawood ve ark., 2011). Bu sorunu çözmek için BT taramalarına Kontrast Sınırlı Adaptif Histogram Eşitleme (KSAHE) (CLAHE-Contrast Limited Adaptive Histogram Equalization) uygulanmıştır. Bu uygulama küçük ayrıntıları, dokuları ve görüntülerin lokal kontrastını geliştirmektedir (Zuiderveld, 1994). KSAHE, kontrast amplifikasyonunun gürültü amplifikasyonunu azaltmak için sınırlandırıldığı bir adaptif

histogram eşitleme türevidir. Bu işlem, her piksel değerindeki bir histogramın değerini kırparak ve ardından tüm histogram kutuları arasında eşit olarak yeniden dağıtarak yapılmaktadır (Contributors, 2020). KSAHE uygulanmasından sonra, Veri Kümesi-1'deki görüntülerden ilgi bölgesi (İB) (ROI) çıkarımı yapılmıştır. Bunun nedeni, Veri Kümesi-1'deki görüntülerin çoğu, akciğerlerin altında çok fazla siyah alan ve bölümler içermektedir ve bu durum görüntü analizinde gereksiz alanların analiz edilmesine ve dolayısıyla zaman kaybına yol açmaktadır. İB çıkarımında, görüntünün üzerine konturlar çizilerek ve ardından en büyük kapalı sınıra sahip dikdörtgen kırpılarak, işlem gerçekleştirilmiştir. Ortaya çıkan İB, görüntüdeki her iki akciğeri de kapsamaktadır. Daha sonra İB görüntüleri 256x256 piksel olarak yeniden boyutlandırılıp, gri seviye görüntülerde piksel değerleri [0,1] arasına normalize edilmiştir. Bu aşamadan sonra, derin öğrenme algoritmasının uygulamasında aşırı öğrenmeyi önlemek ve görüntülerinin sayısını artırmak için verilere döndürme (rotating), kırpma (shearing), yakınlaştırma (zooming) ve yatay çevirme (horizontal flipping) gibi veri büyütme stratejileri uygulanmıştır. Bu işlem adımlarında görüntülere karşılık gelen segmentasyon maskelerine de aynı işlem adımları sırasıyla uygulanmıştır. Şekil 4.3, ön işleme aşamasının çıktılarını göstermektedir.

4.2. Görüntü Segmentasyonu

Görüntü segmentasyonu, dijital bir görüntünün genellikle bir nesne ve arka plandan oluşan anlamlı bölümlere ayrıldığı süreçtir. Bu süreç, görüntüyü basitleştirir ve görüntü analizini geliştirir (Chhawchharia ve ark., 2016). Bu nedenle görüntü segmentasyonu, pikselleri benzerliklere veya farklılıklara göre gruplandırmayı amaçlayan bir görüntü işleme problemidir. Segmentasyon, görüntüyü daha anlamlı hale getirir ve sonraki aşamaların daha hızlı olmasını sağlar (Saglam ve Baykan, 2017). Görüntü segmentasyonu, otonom araçlar, yüz tanıma, tıbbi görüntüleme vb. birçok görüntüleme uygulamasının önemli bir parçasıdır. Tıbbi görüntü analizinde vazgeçilmez bir adım olarak segmentasyon, görüntünün daha hızlı ve verimli analiz edilmesini ve daha çok bilgi sağlamasını amaçlamaktadır. Tıbbi görüntü segmentasyonu, diğer segmentasyon uygulamalarından farklıdır çünkü segmentlenecek nesnenin anatomik yapısına uygun olarak işlem yapılması gerekmektedir. BT ve MRI (Manyetik Rezonans Görüntüleri) görüntüleri gibi yaygın görüntü formatları; manuel incelemeleri çok zorlaştıran, yüksek çözünürlüklere ve çok

sayıda görüntü kesitlerine sahiptir. Bu nedenle, otomatik segmentasyon araçları, özel öneme sahip nesneleri vurgulayarak bu tür büyük görüntülerde tarama yapma konusunda tıbbi personele yardımcı olabilir. Ayrıca tıbbi görüntü segmentasyonu, daha gelişmiş analizler ve simülasyonlar için kullanılabilecek ilgili anatomik veya patolojik yapıların çıktısını verebilir (Läthén. 2010).



Şekil 4.3. Ön işlem aşaması adımları

Tez kapsamında BT görüntülerinin segmentasyonu için literatürde yaygın olarak kullanılan UNet, UNet++, SegNet ve PSPNet mimarileri kullanılmıştır. Daha sonra ise başarı arttırımı için bu mimariler, Koşullu Üretken Çekişmeli Ağ (kÜÇA) (cGAN) ile hibrit kullanılarak çalışmalar yapılmıştır. Bu hibrit mimari yapılardan kÜÇA ile UNet++, SegNet ve PSPNet ile yapılan segmentasyon işlemlerinde, segmente edilmiş görüntüler tamamen siyah sonuç verdiği için dolayı, bir sonuç alınamamış ve bu hibrit modeller

kullanılmamıştır. kÜÇA-UNet hibrit modeli ile yapılan segmentasyon işleminde ise başarılı sonuçlar elde edilmiştir.

Çizelge 4.1. Temel derin öğrenme mimarileri ile yapılan segmentasyon işlemi için kullanılan parametre değerleri

Mimari	Epoch sayısı	Optimize edici	Öğrenme oranı	Momentum	Kayıp fonksiyonu
UNet	100	Adam	0.0005	$\beta_1 = 0.9$, $\beta_2 = 0.999$	İkili çapraz entropi + Zar kaybı
UNet++	100	Adam	0.0005	$\beta_1 = 0.9$, $\beta_2 = 0.999$	İkili çapraz entropi + Zar kaybı
SegNet	100	Adam	0.001	$\beta_1 = 0.9$, $\beta_2 = 0.999$	Ortalama Kare Hatası (MSE)
PSPNet	100	Adam	0.001	$\beta_1 = 0.9$, $\beta_2 = 0.999$	Ortalama Kare Hatası (MSE)

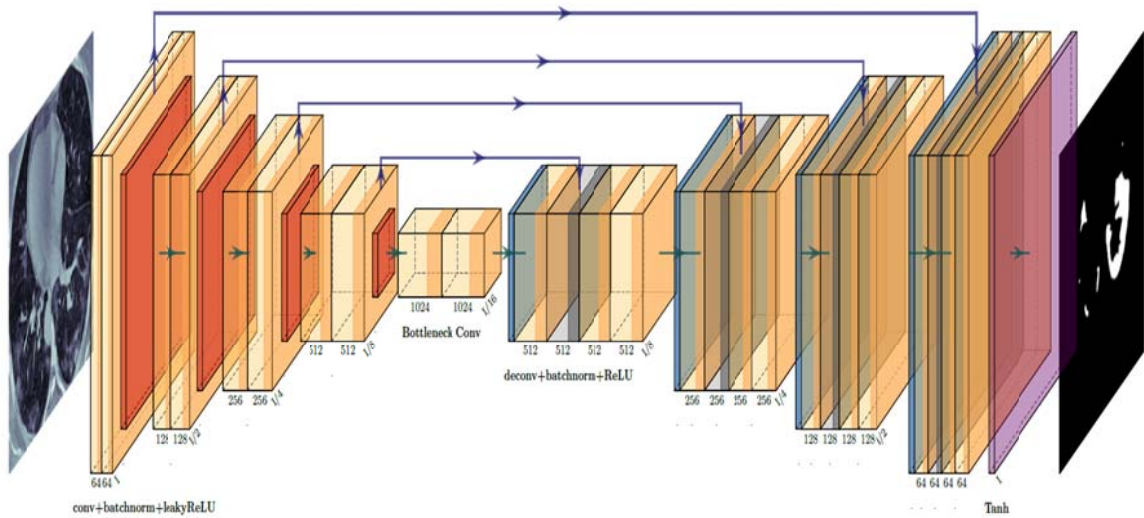
4.2.1. Koşullu Üretken Çekişmeli Ağ ve UNet ile segmentasyon

Bu tez çalışmasında BT görüntülerinden Covid-19 lezyonlarını segmentlere ayırmak için Koşullu Üretken Çekişmeli Ağlar (kÜÇA) (cGAN) ile UNet mimarisi hibrit kullanılmıştır. Önerilen mimaride, görüntüden görüntüye çeviri için kullanılan Pix-2-Pix ÜÇA yapısı kullanılmıştır (Isola ve ark., 2017). Bu ağ yapısında Üretken Ağ, Alıcı işletim karakteristiği (AİK) (ROC) tanımayı öğrenir ve sınırlarını çevreleyen ikili bir maske oluşturur. Ayırıcı ise gerçek segmente edilmiş maskeler ile sentetik maskeler arasında ayırım yapmayı öğrenir. kÜÇA için kullanılan “üretici” ve “ayırıcı” yapıları UNet ile modifiye edilerek kullanılmıştır.

kÜÇA için Üretici Mimarisi: Tez çalışmasında kullanılan kÜÇA üreticisi, modifiye edilmiş bir UNet mimarisidir ve kÜÇA-UNet ismi buradan gelmektedir. kÜÇA-UNet mimarisinin üreticisindeki kodlama bölümü, tipik Evrişimsel Sinir Ağı (ESA) (CNN) mimarisini takip eder ve birçok kodlayıcı bloğundan oluşmaktadır. Her kodlayıcı bloğu, bir girdi görüntüsünü bir evrişimsel katmandan geçirir, ardından toplu normalleştirme ve sızdıran doğrultulmuş doğrusal birim (LeakyReLU) (eğim 0.2) gelir. Orijinal UNet’in aksine, bu yaklaşımda tüm maksimum havuzlama katmanları kaldırılır. Her bloktan sonra, mimarinin karmaşık yapıları etkili bir şekilde öğrenebilmesi için çekirdek sayısı iki katına çıkar.

UNet mimarisinde dar geçit bölümü; daralma ve genişleme katmanını birbirine bağlar. Daralma bölümü 2x2 adım kaydırma işlemi (stride), bir adet 4x4 evrişim

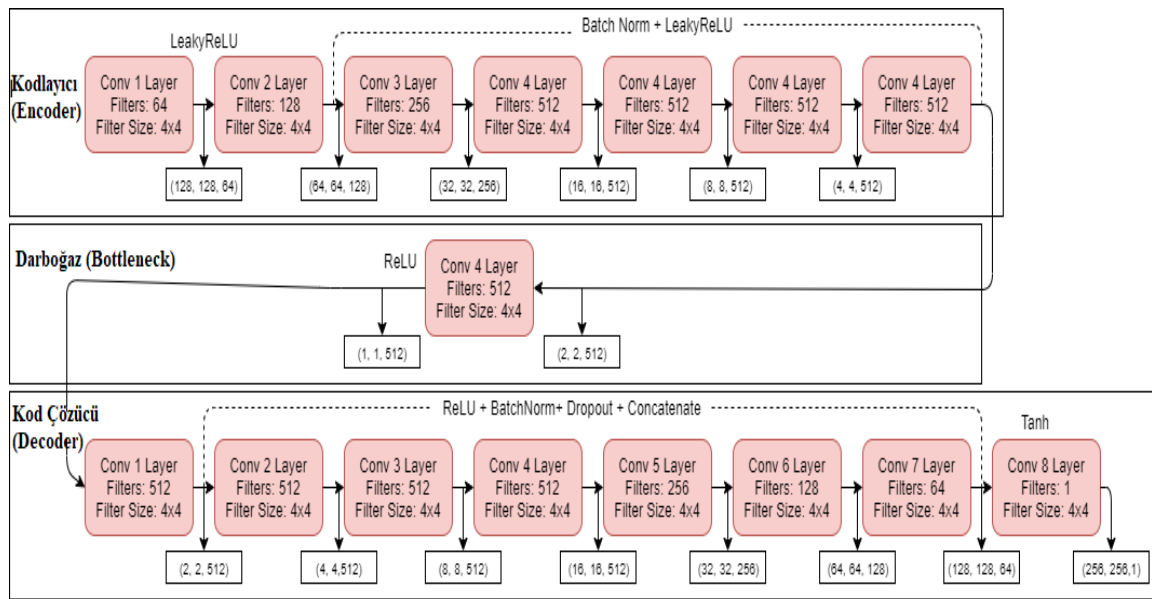
katmanından ve ardından bir ReLU aktivasyon katmanından oluşur. Bu genellikle “atlama bağlantıları” (skip connection) olarak adlandırılır. Daralma bölümü gibi, genişleme bölümü de birkaç bloktan oluşmaktadır. Her kod çözücü bloğunda girdi görüntüsü bir ters evrişim katmanından geçirilir, ardından toplu normalleştirme, bir seyreltme katmanı, bir birleştirme katmanı ve son olarak bir etkinleştirme katmanı gelmektedir. Kodlayıcı bloklarından farklı olarak burada simetriyi korumak için özellik haritalarının sayısı yarı yarıya azaltılmaktadır. Birleştirme katmanında, karşılık gelen daralma katmanına girdi eklenir. Bu yaklaşım, daralma sırasında öğrenilen özelliklerin yeni görüntüyü yeniden oluşturmak için kullanılmasını sağlamaktadır. Kodlayıcı bloklarının sayısının, kod çözücü bloklarının sayısı ile aynı olduğuna dikkat edilmelidir. Genişletme bloğundan elde edilen sonuç, daha sonra özellik haritalarının sayısının istenen segment veya sınıf sayısına eşit olduğu başka bir ters evrişim katmanından geçirilir. Tüm Evrişimsel ve evrişimsiz katmanlar, 4x4 çekirdek ve 2x2’lik bir adımla tanımlanır ve bunlar sırasıyla özellik haritalarını aşağı ve yukarı örneklemektedir. Bu mimari Şekil 4.4’te gösterilmektedir.



Şekil 4.4. Tez çalışmasındaki kÜÇA-UNet yapısındaki üretici mimari yapısı

kÜÇA-UNet’in performansını iyileştirmek amacıyla, tez kapsamında literatürde yer alan çeşitli stratejiler uygulanmıştır (Salimans ve ark., 2016; Brownlee, 2019). Bu stratejilerin arasında toplu normalleştirme, LeakyReLU kullanma ve hem üretici hem de ayırıcı da 0.002’lik bir standart sapma ile rastgele Gauss ağırlık ilklendirmesi

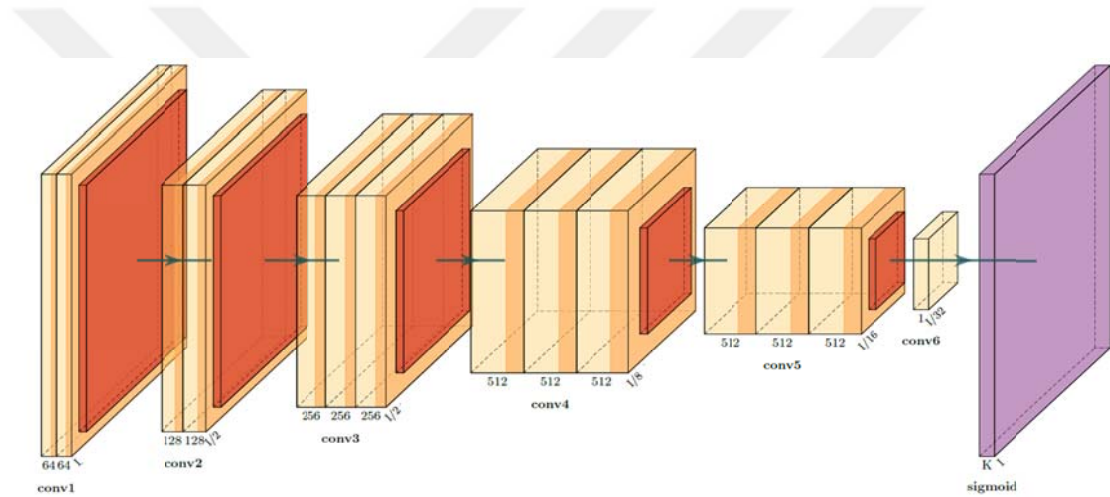
bulunmaktadır. Ek olarak, ayırıcının tüm katmanlarına $L2$ düzenlenmesi eklenmiştir. Ayrıca tüm sınıf etiketlerine etiket yumuşatma uygulanmıştır. Etiket yumuşatmada, gerçek ve sahte etiketleri temsil etmek için 1 ve 0 gibi sert etiketleri kullanmak yerine, gerçek görüntüleri temsil etmek için 0.9 ile 1.0 arasında rastgele değerler ve sahte görüntüleri temsil etmek için 0 ile 0.1 arasında değerler üreterek yumuşak etiketler kullanılmıştır. Bu değişiklikleri uyguladıktan sonra, modelin birleşme hızında gözle görülür bir değişiklik gözlenmiştir. Üreticinin hem girdi hem de çıktı katmanlarının tüm özellik vektörlerinin boyutları Şekil 4.5'te verilmektedir.



Şekil 4.5. kÜÇA-UNet yapısındaki kullanılan üretici modelinin girdi ve çıktı katmanlarındaki özellik vektörlerinin boyutları

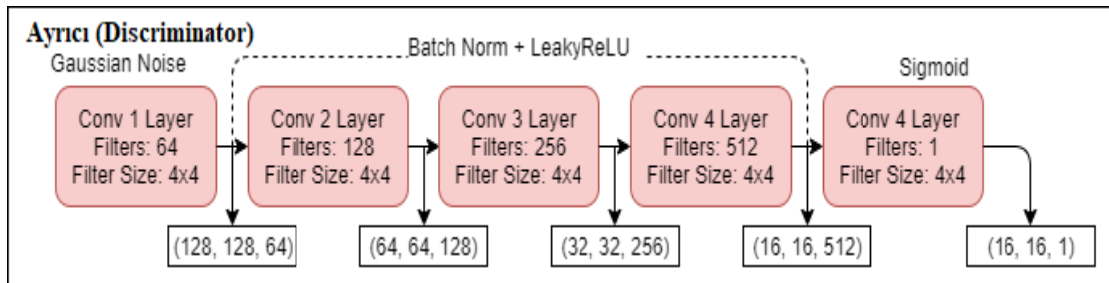
kÜÇA için Ayırıcı Mimarisi: Ayırıcı, sisteme verilen ikili maskenin gerçek mi yoksa sahte mi olduğunu belirlemektedir. (Isola ve ark., 2017). Yama ayırıcı, girdi görüntüsünü bir dizi yamaya böler ve her yamayı tek bir sayısal çıktıyla eşler. Bir görüntünün tamamı üzerinde tahmin gerçekleştiren normal bir görüntü ayırıcının aksine, yama ayırıcı her bir yama için bir tahmin yapar ve son tahmin tüm yama tahminlerinin aritmetik ortalamasıdır. Ayrıca, yama ayırıcı daha az parametre gerektirir, çok büyük ve bulanık görüntülerle iyi çalışır ve hesaplama süresi daha düşüktür. Isola ve ark. (2017), tarafından yapılan çalışmada 70x70 yama boyutunun küresel özellikleri barındırabileceği

için iyi bir seçim olduğunu gösterildiğinden, yapılan tez çalışmasında da yama ayırıcı için, 70x70 piksellik bir yama boyutu kullanılmıştır. Uygulanan model, özellikle birlikte birleştirilmiş bir kaynak ve bir hedef görüntü olmak üzere 2 adet girdi görüntüsü almaktadır. Aşırı öğrenmeyi önlemek için, birinci evrişim katmana geçmeden önce bu birleştirilmiş girdiye Gauss gürültüsü ($\sigma = 0.2$) eklenmiştir. Model, her biri toplu normalleştirme ve LeakyReLU tarafından takip edilen 5 ana evrişim katmandan oluşmaktadır. Her evrişim katmanı 4x4 filtreye ve 2 adıma sahiptir. Ayrıca ayırıcı, L2 düzenlemesi kullanılarak gradyanlarının normlarını sınırlayarak düzenlenir. Bu nedenle, tüm evrişim katmanlarına bir L2 düzenleyici eklenir. Ayırıcı mimarisi Şekil 4.6'da gösterilmektedir.



Şekil 4.6. Tez çalışmasındaki kÜÇA-UNet yapısındaki ayırıcı mimari yapısı

Ayırıcının hem girdi hem de çıktı katmanlarının tüm özellik vektörlerinin boyutları Şekil 4.7’de verilmektedir.



Şekil 4.7. kÜÇA-UNet yapısındaki kullanılan ayırıcı modelinin girdi ve çıktı katmanlarındaki özellik vektörlerinin boyutları

Ayırıcı, ikili çapraz entropi (içe) (bce-binary-cross entropy) kaybı (Denklem 4.1) ve zar kaybının (Dice) (Denklem 4.2) bir kombinasyonu kullanılarak optimize edilir. Ayırıcı kaybı, Denklem 4.3 kullanılarak hesaplanır ve bu kayıp üretici aracılığıyla geri yayılır. İkili çapraz entropi kaybı, ESA (CNN) sınıflandırması için en yaygın kullanılan kayıptır (Kodym ve ark., 2018).

$$L_{BCE} = -\frac{1}{N} \sum_{i=1}^N y_i \cdot \log \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i)) \quad (4.1)$$

$$L_{ZAR} = 1 - \frac{2 \times (y \cap p(y))}{y + p(y)} \quad (4.2)$$

$$L_{DIS} = L_{BCE} + L_{ZAR} \quad (4.3)$$

Bu denklemlerde y gerçek veri, $p(y)$ veri kümesindeki görüntü sayısı olan N 'nin tüm değerleri için öngörülen maskesi ve L_{DIS} ise ayırıcı kaybı temsil etmektedir. L_1 kayıp fonksiyonu, gerçek değer ile tahmin edilen değer arasındaki tüm mutlak farkların toplamı olan hatayı en aza indirmek için kullanılır.

Üretici G ise, Isola ve ark. (2017) tarafından önerildiği üzere, ayırıcı kaybı ve L_{L1} kaybı (L_1 Loss) (Denklem 4.4) kullanılarak L_{L1} kaybı lehine $[1, 100]$ arasında bir ağırlıkla güncellenir. Üretici için nihai amaç fonksiyonu Denklem 4.5'te gösterilmektedir.

$$L_{L1}(G) = \sum_{x,y,z} ||y - G(x,z)|| \quad (4.4)$$

$$L_{ÜRETİCİ} = \argmin G \max D L_{kÜÇA}(G, D) + \lambda L_{L1}(G) \quad (4.5)$$

Burada λ , L_{L1} kaybının ağırlığını arttırmak için kullanılan sabit değeri temsil etmektedir ve genelde ($\lambda=100$) olarak kullanılmaktadır. $L_{L1}L_1$ kaybını, $L_{ÜRETİCİ}$ üretici kaybını ve $L_{kÜÇA}$ ise çekişme kaybını temsil etmektedir.

Üretici ve ayırıcı; üreticinin makul bir ikili maske oluşturmayı öğrendiği ve ayırıcının ise sentetik ve gerçek segmentasyon arasında nasıl ayrım yapılacağını öğrendiği noktada optimize edilir. Burada içe (bce) zar kaybı, Gauss gürültüsü ve $L2$ düzensileştirmesinin tanıtılmasının yanı sıra, farklı öğrenme oranları ve kayıp fonksiyonları

kullanılmıştır. kÜÇA-Unet'in için kullanılan paramatere değerleri Çizelge 4.2'de verilmiştir.

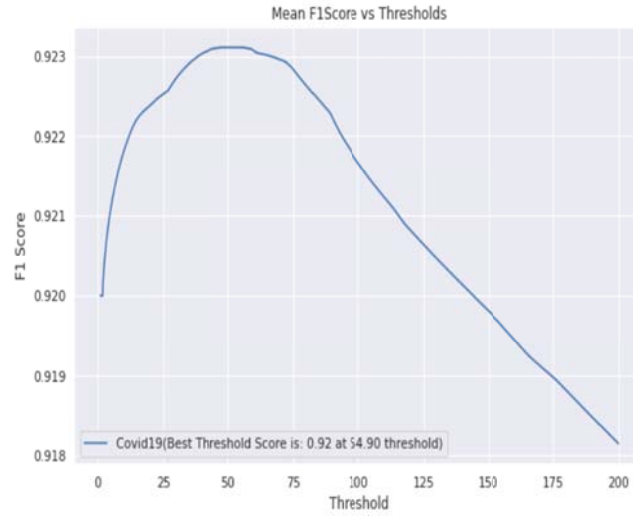
Çizelge 4.2. kÜÇA-UNet için kullanılan parametre değerleri

Epoch sayısı	Optimize edici	Öğrenme oranı	Momentum	Kayıp fonksiyonu
131000	Adam	0.0001	$\beta_1 = 0.5,$ $\beta_2 = 0.999$	İkili çapraz entropi (bce), zar (dice) ve L_{L1}

Segmentasyon maskeleri elde edildikten sonra, ikili maskeler oluşturmak için basit bir eşikleme gerçekleştirildiğine dikkat edilmelidir. Eşikte, eşik olarak bilinen ölçüm için belirli bir değer alınır. Bir görüntüdeki pikselin değeri eşik değerinden büyük veya ona eşitse, bu bir nesne pikselidir veya aksi takdirde bir arka plan pikselidir. Eşikleme, paraziti, yani çok büyük veya çok küçük pikselleri kaldırarak görüntüyü filtreler. Şekil 4.8'de gösterildiği gibi 0.9223 (%92.23) F1-skor değerinin elde edildiği en iyi eşik değeri olarak 54.90 seçilmiştir.

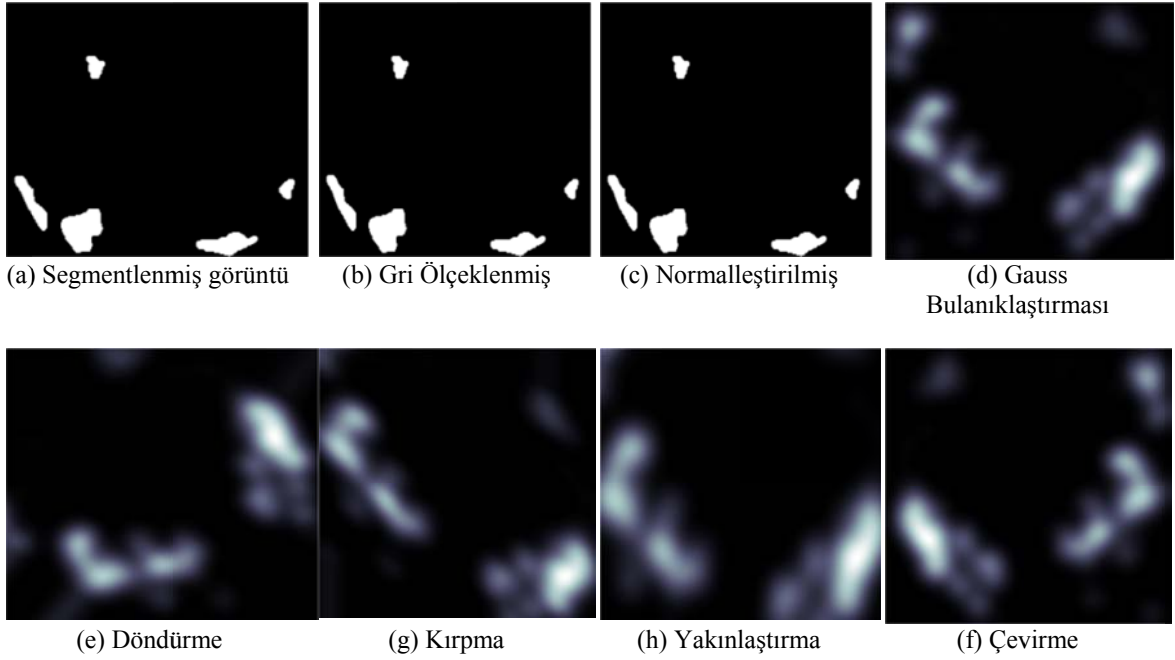
4.3. Sınıflandırma

Sınıflandırma, verilen veri noktalarının sınıfını tahmin etme sürecidir. Burada bir model, verilen veri kümesinden veya gözlemlerden öğrenmekte ve ardından yeni gözlemleri var olan sınıfa veya gruba sınıflandırmaktadır. Tıbbi görüntü analizinde önemli bir adım olan görüntü sınıflandırmanın amacı tıbbi görüntüleri, görüntüleme formatları veya klinik patolojiler gibi belirli kriterlere göre ayırt etmektir. İyi bir görüntü sınıflandırma sistemi, tıbbi görüntülerin hızlı ve doğru bir şekilde incelenmesi ve yorumlanmasında sağlık personeline yardımcı olmalıdır (Zhang ve ark., 2019). Derin öğrenme, geleneksel yöntemlere göre kanıtlanmış avantajlar gösterse de, görüntüleme formatlarının ve klinik patolojilerin çeşitliliğinin neden olduğu önemli sınıf içi varyasyon ve sınıflar arası benzerlik nedeniyle zorlayıcıdır.



Şekil 4.8. Eşik değeri ve F1-skor grafiği

Sınıflandırma için, segmentasyon işlemi sonrasında elde edilen segmentlenmiş görüntüler, görüntülerde Covid-19 olup olmadığını tahmin etmek için çeşitli sınıflandırıcılara girdi olarak verilmiştir. Segmentlenmiş görüntülere, sınıflandırıcılara girdi olarak vermeden önce, Şekil 4.9’da gösterildiği gibi döndürme, kırpma, yakınlaştırma ve Gauss bulanıklaştırması gibi basit ön işleme ve veri artırma yöntemleri uygulanmıştır.

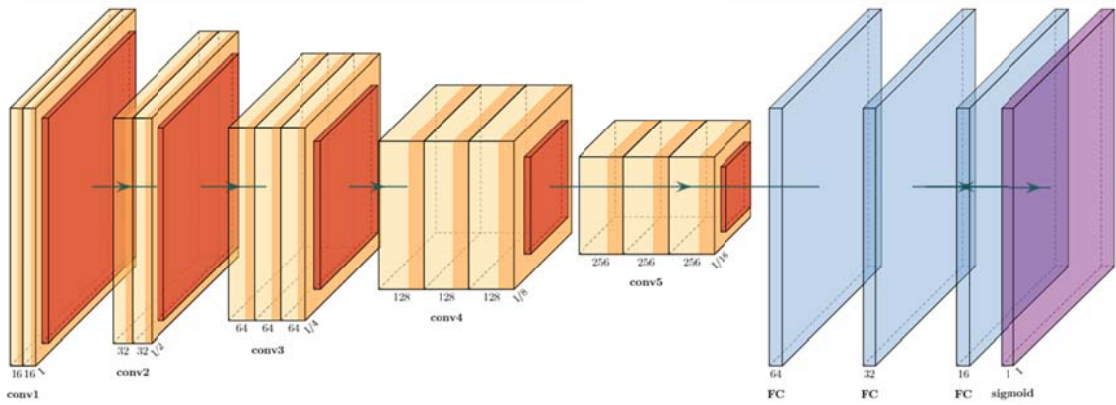


Şekil 4.9. Segmentlenmiş görüntüler için veri artırma işlemleri

Tez kapsamında sınıflandırma için Evrişimsel Sinir Ağı (ESA) (CNN), PatchESA (PatchCNN) ve Kapsül Sinir Ağı (KSA) (CapsNet) olarak üç farklı sınıflandırma yöntemi uygulanmış ve performansları karşılaştırılmıştır.

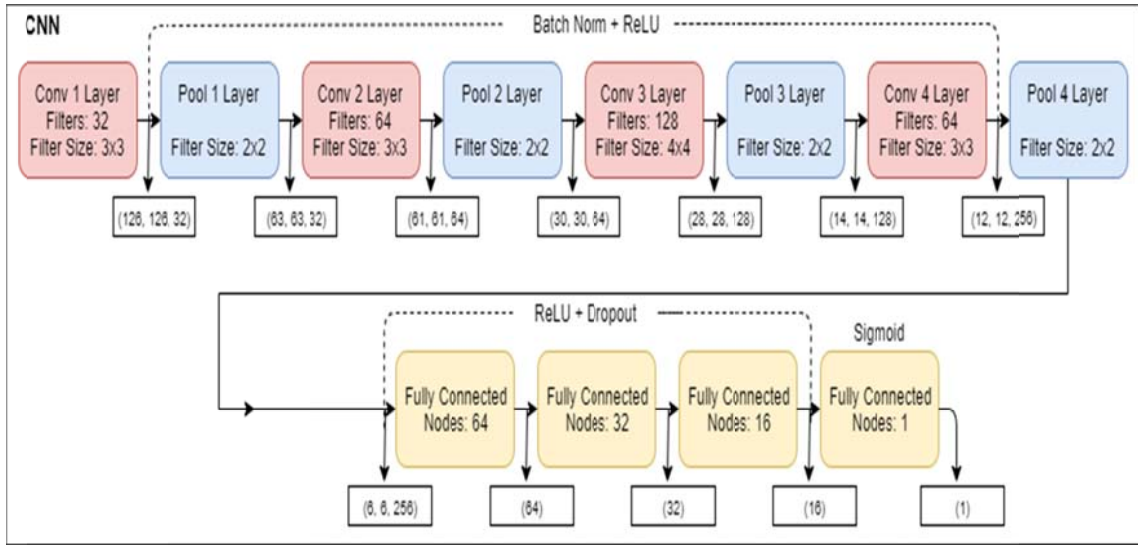
4.3.1. Evrişimsel Sinir Ağı ile Sınıflandırma

Tez kapsamında kullanılan Evrişimsel Sinir Ağı (ESA) (CNN) Şekil 4.10'da gösterildiği üzere, 5 evrişim katmanından ve 3 tam bağlı (fully connected) katmandan oluşmaktadır. Her evrişimsel katmanı yığın normalleştirme (batch normalization) ve bir maksimum havuzlama (max pooling) katmanı takip ederken; tam bağlı katmanları, seyreltme katmanları (dropout) takip eder. ReLU aktivasyon fonksiyonu, sigmoid kullanan son katman dışındaki tüm katmanlara uygulanmıştır.



Şekil 4.10. Tez kapsamında kullanılan Evrişimsel Sinir Ağı (ESA) yapısı

ESA'da evrişimsel katmanlar 3x3 çekirdek kullanır ve 1 adımdır. Ağ, 150 iterasyon boyunca eğitilmiştir. Aşırı öğrenmeyi önlemek için yığın normalleştirme ve seyreltme katmanları eklenmiştir. Kullanılan ESA'nın katmanları ve katmanların boyutları Şekil 4.11'de ve kullanılan parametre değerleri de Çizelge 4.3'de verilmektedir.



Şekil 4.11. Tezde kullanılan ESA ve PatchESA modelinin girdi/çıkış katmanları ve boyutları

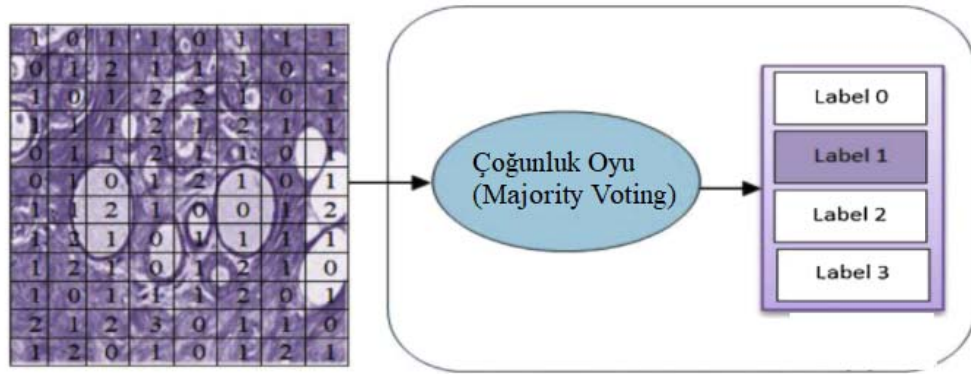
Tez kapsamında sınıflandırma için kullanılan ESA ve PatchESA modellerinin katmanları ve katmanların boyutları aynıdır.

Çizelge 4.3. Tez kapsamında kullanılan ESA ve PatchESA için parametre değerleri

Epoch sayısı	Optimize edici	Öğrenme oranı	Momentum	Kayıp fonksiyonu
150	Adam	0.0005	$\beta_1 = 0.5,$ $\beta_2 = 0.999$	İkili çapraz entropi (bce)

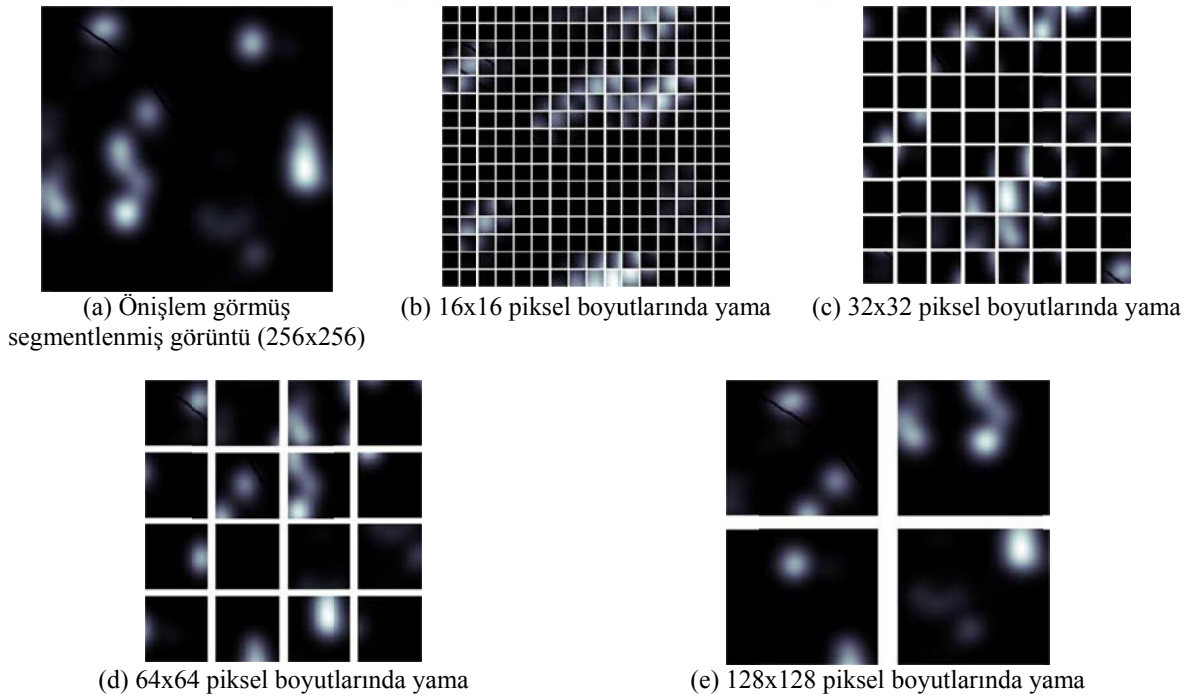
4.3.2. Patch Sinir Ağı ile Sınıflandırma

PatchESA mimarisi, ESA mimarisine çok benzemektedir. Ancak tüm görüntünün ağı gönderildiği ESA'dan farklı olarak, bir görüntüden belirlenen sayıda ya da boyutta “yama” çıkarılmakta ve ardından her yama ağı gönderilmektedir. Önerilen yöntemde, her bir yama için sınıf etiketi model tarafından tahmin edilmekte, ardından çoğunluk oylama şeması uygulanarak, en çok oyu alan sınıf, görselin son sınıfı olmaktadır (Şekil 4.12).



Şekil 4.12. Önerilen PatchESA mimarisi için yama etiketleri ve oylama ile sonucun belirlenmesi (Roy ve ark., 2019)

Segmentasyon aşamasında kÜÇA-UNet modelinden üretilen ve boyutu 256x256 piksel boyutlarında olan segmentlenmiş görüntüler üzerinde sınıflandırma aşamasından önce veri artırma uygulandıktan sonra çeşitli yama boyutları denenmiştir. Yamalar, görüntünün belirli bir yama boyutundaki daha küçük görüntülere bölünmesiyle oluşturulmaktadır. Şekil 4.13’de görüldüğü gibi, 16x16, 32x32, 64x64 ve 128x128 piksel boyutunda yamalar denenmiştir.



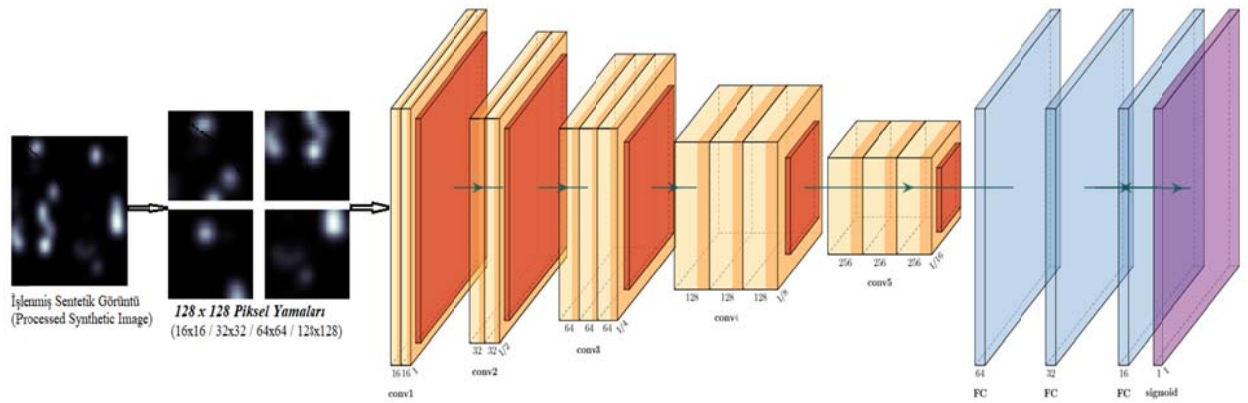
Şekil 4.13. Ön işleme tabi tutulmuş girdi görüntüsü ve farklı boyutlarda elde edilen yamalar

Farklı boyutlarda yamalar denenerek, en iyi sonuç veren yama boyutu tespit edilmiştir. Çizelge 4.4'te yama boyutunun ne kadar büyük olursa, model performansının o kadar iyi olacağı gösterilmektedir. Bu durum; daha küçük yama boyutlarının yeterli tanı bilgisi taşımazken, daha büyük yama boyutlarının yeterli ayırt edici özellikler taşıdığını göstermektedir.

Çizelge 4.4. PatchESA için farklı yama boyutlarına göre elde edilen başarılar

Yama (patch) boyutu	16x16	32x32	64x64	128x128
Doğruluk (accuracy)	% 50.59	% 68.41	% 89.93	%92.56

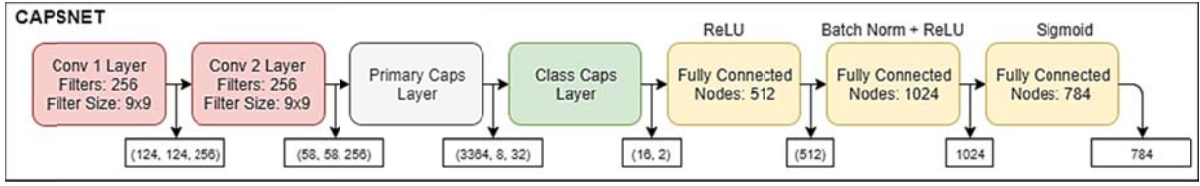
Yama boyutu başarısına göre, tüm PatchESA modellerinde 128x128 piksel boyutunda yamalar ile çalışılmıştır. PatchESA için kullanılan katman ve parametreler ESA ile aynı olup; katmanlar ve katmaların boyutları Şekil 4.11'de ve parametre değerleri de Çizelge 4.3'de verilmiştir. Şekil 4.14 ise, PatchESA mimarisinin yapısını göstermektedir.



Şekil 4.14. 128x128 piksel boyutlarındaki yama ile PatchESA mimari yapısı

4.3.3. Kapsül Sinir Ağı ile Sınıflandırma

Bu çalışmada kullanılan KSA mimarisinin yapısı Şekil 3.14'de verilen temel KSA mimarisi yapısı ile aynıdır. KSA için kullanılan parametreler ise Şekil 4.15 ve Çizelge 4.5'te gösterilmiştir.



Şekil 4.15. Tez kapsamında kullanılan Kapsül Sinir Ağı (KSA) yapısı

Çizelge 4.5. Tez kapsamında kullanılan KSA için parametre değerleri

Epoch sayısı	Optimize edici	Öğrenme oranı	Momentum	Kayıp fonksiyonu
100	Adam	0.0001	$\beta_1 = 0.9$, $\beta_2 = 0.999$	Marj + L2 kaybı

5. DENEYSEL SONUÇLAR VE TARTIŞMA

Literatürde önerilen mevcut Covid-19 BT segmentasyon yöntemlerinin çoğu, farklı veri kümeleri üzerinde çalışılmıştır. Bu veri kümeleri çoğunlukla özel veri kümeleridir ve farklı ortamlarda çalıştırılıp, farklı ölçüler kullanılarak değerlendirilmiştir (Ma ve ark., 2020). Bu durum, önerilen yöntemler arasında performans karşılaştırması yapılmasını zorlaştırmaktadır. Bu nedenle, tez kapsamında karşılaştırmaların doğruluğu için hem segmentasyon hem de sınıflandırma için önerilen tüm yöntemler tez kapsamında kullanılan veri setlerine (segmentasyon için Veri Kümesi-1 ve sınıflandırma için Veri Kümesi-2) ayrı ayrı uygulanarak, sonuçlar karşılaştırılmıştır. Elde edilen sonuçların literatür ile karşılaştırılmasında da, tez kapsamında en iyi başarıyı veren yöntemlerin literatür ile karşılaştırılmaları yapılmıştır.

Tez kapsamında yapılan tüm deneyler, NVIDIA GPU Computing Toolkit 10.1 GPU'yu destekleyen Intel® Core™ i7-4720HQ CPU @ 2.60GHz × 8 işlemci, NVIDIA GeForce GTX 960M / PCIe / SSE2 grafik kartı kullanılarak yapılmıştır. Uygulama, Ubuntu 18.0.4 platformunda uygulanmış ve Tesla T4 GPU, yaklaşık 12.6 GB RAM ve 33 GB disk alanı ile ücretsiz bir GPU ve jupyter dizüstü bilgisayar ortamı sağladığı için Google Colab entegre geliştirme ortamı (IDE) olarak tercih edilmiştir. Platformlar arası olması, yaygın olarak kullanılması, aktif bir geliştirici topluluğuna sahip olması ve kullanımı kolay yerleşik kütüphaneler sağlaması nedeniyle kullanılan programlama dili Python'dur (2019). Ayrıca, Keras 2.3.1 ve Tensorflow 2.0 açık kaynaklı makine öğrenmesi kütüphaneleri Python'da uygulanmaktadır. Keras 2.3.1, hata ayıklamayı kolaylaştıran ve istekli yürütmeye (eager execution) sahip API'ler ile kolay model oluşturma olanağı sağlamaktadır. Tensorflow 2.0, hem makine öğrenmesi yöntemlerinin kullanımı hem de araştırma için güçlü bir deney ortamı sağlar (Abadi ve ark., 2016).

5.1. Segmentasyon Sonuçları

Tez çalışmasında, BT görüntüler için segmentasyon ve sınıflandırma modellerini değerlendirmek için ayrı deneyler yapılmıştır. İlk olarak segmentasyon için, literatürde kullanılan dört derin öğrenme modeli olan UNet, UNet++, SegNet, PSPNet ve önerilen hibrit model KÜÇA-UNet, Veri Kümesi-1'deki BT görüntülerini segmentlere ayırmak için

kullanılmıştır. Bu aşamada Veri Kümesi-1'deki segmentlenmiş görüntüler için sonuçlar analiz edilmiş ve sistemin doğruluğu ortaya konmuştur. Çalışmada, Veri Kümesi-1'deki veriler önce literatürde kullanıldığı gibi %80 eğitim ve %20 test olarak ayrılarak çalışmalar yapılmıştır. Daha sonra ise, verilerde 5-kat çapraz doğrulama (5-fold cross validation) uygulanarak da sonuçlar elde edilmiştir. Segmentasyon için kullanılan eğitim ve test veri sayılarına ilişkin veriler Çizelge 5.1'de verilmiştir.

Çizelge 5.1. Segmentasyon için Veri Kümesi-1'de kullanılan görüntü sayıları

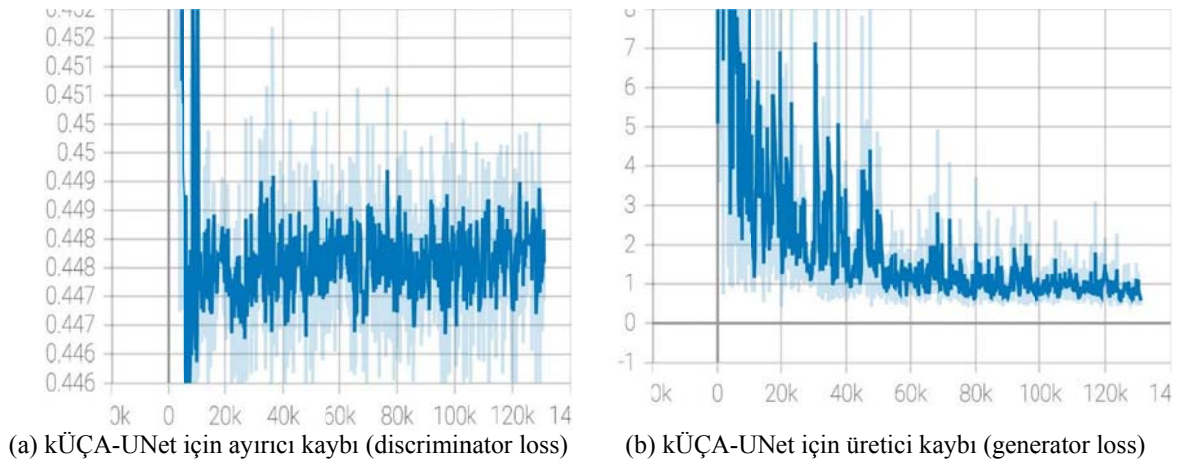
	Covid-19 hastası	Sağlıklı	Toplam
Eğitim (%80)	1292	398	1690
Test (%20)	323	99	422
Veri Kümesi-1 Toplam	1615	497	2112

Segmentasyon için kullanılan dört temel algoritma ve tez kapsamında önerilen kÜÇA-UNet yönteminin Veri Kümesi-1'deki segmentasyon başarısı için karşılaştırmalar Zar Skoru (Dice score), Birlik Üzerinden Kesişme (BüK) (IoU-Intersection over Union), Kesinlik (Precision), Duyarlılık (Recall) ve F1-skor (F1-score) metriklerine göre Çizelge 5.2'de verilmektedir.

Çizelge 5.2. Veri Kümesi-1 kullanılarak yapılan segmentasyon işleminde elde edilen sonuçlar (%)

Model	Zar skoru	BüK	Kesinlik	Duyarlılık	F1-skoru
UNet(Ronneberger, 2015)	86.90	77.74	86.64	86.65	86.29
UNet++ (T. Zhou, 2020)	85.12	75.20	84.11	86.39	84.50
SegNet (Badrinarayanan, 2017)	77.37	65.08	74.87	74.87	77.37
PSPNet (Zhao, 2017)	51.20	37.66	61.55	53.96	51.37
kÜÇA-UNet (önerilen yöntem) (5-kat çapraz doğrulama olmadan)	92.31	86.41	92.2	92.70	92.31
kÜÇA-UNet (önerilen yöntem) (5-kat çapraz doğrulama ile)	92.21	86.20	92.4	92.03	92.21

Çizelge 5.2'ye göre, tez kapsamında önerilen yöntem tüm metriklerde diğer algoritmalarından daha iyi performans göstermektedir. kÜÇA ile diğer temel modellerin hibrit kombinasyonları olan kÜÇA-UNet++, kÜÇA-SegNet ve kÜÇA-PSPNet; eğitim sürecinde öğrenememiş ve tamamen siyah görüntüler üretmiştir. Bu nedenle bu hibrit modellerin sonuçları alınamamıştır. Şekil 5.1, önerilen kÜÇA-UNet modelinin eğitim sırasında ayırıcı ve üretici kaybını göstermektedir.



Şekil 5.1. Epoch sayısını göre kÜÇA-UNet kayıp fonksiyon grafikleri

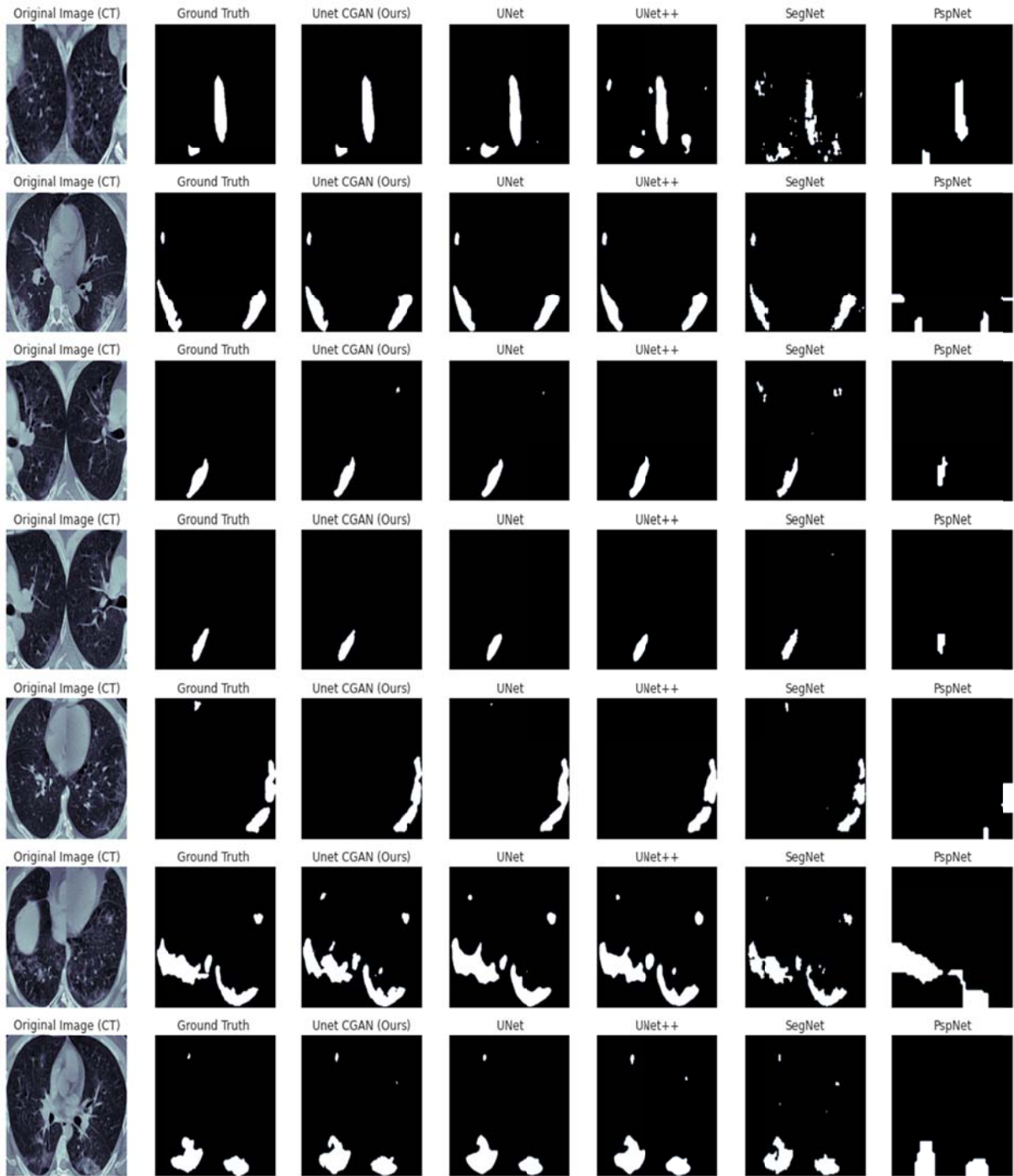
Segmentasyon için, her model tarafından üretilen segmentasyon sonuçları Şekil 5.2’de verilmiştir. kÜÇA-UNet modeline ait sonuçlar, gerçeğe en yakın sonuçları net bir şekilde vermiştir.

5.2. Sınıflandırma Sonuçları

Sınıflandırma işleminde, sağlıklı ve Covid19 hastalarının BT taramalarını içeren Veri Kümesi-2 üzerinde sınıflandırma yapılmıştır. Veri Kümesi-2’nin görüntü segmentasyonu için gerçek ikili maskelere sahip olmadığına dikkat etmek önemlidir. Bu nedenle, segmentasyon aşamasında elde edilen sonuçlara göre, en yüksek başarıyı veren ve tez kapsamında önerilen kÜÇA-UNet modeli ile Veri Kümesi-2’de segmentasyon işlemi yapılmış ve elde edilen segmentlenmiş görüntüler ESA (CNN), PatchESA (PatchCNN) ve KSA (CapsNet) olarak üç farklı derin öğrenme sınıflandırıcısı kullanılarak sınıflandırılmıştır. Segmentasyonda olduğu gibi veri kümesi Çizelge 5.3’te gösterildiği şekilde, eğitim için önce % 80 ve test için % 20 olarak bölünmüştür. Daha sonra ise, sınıflandırma başarısını da test etmek için verilere 5-kat çapraz doğrulama uygulanmıştır.

Çizelge 5.3. Sınıflandırma için Veri Kümesi-2’de kullanılan görüntü sayıları

	Covid-19 hastası	Sağlıklı	Toplam
Eğitim (%80)	1001	983	1984
Test (%20)	251	246	497
Veri Kümesi-2 Toplam	1252	1229	2481

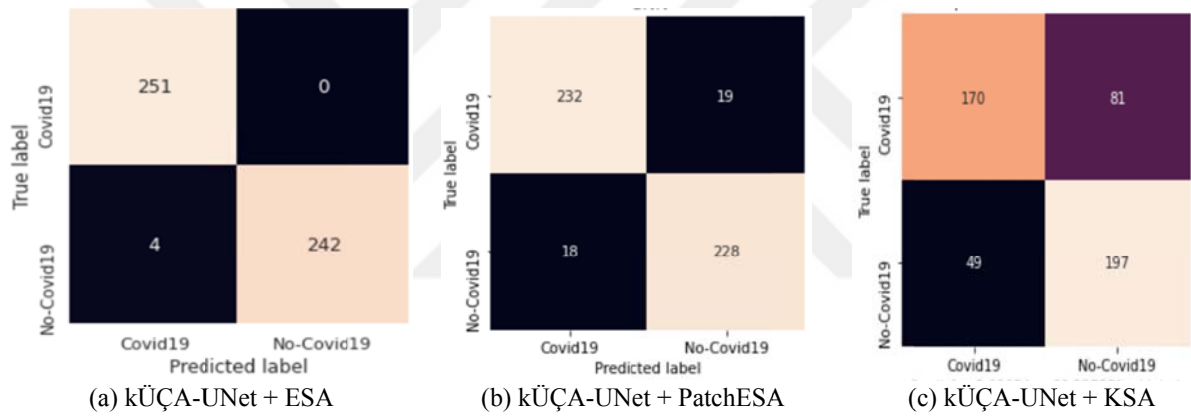


Şekil 5.2. Segmentasyon için kullanılan modellere ait görsel sonuç örnekleri

Sınıflandırma modellerinin performansını değerlendirmek ve sistemlerin başarısını test etmek için doğruluk (accuracy), Kesinlik (Precision), Duyarlılık (Recall), Özgüllük (Specificity) ve F1-skor (F1-score) kullanılmıştır. Ayrıca, sınıflandırma başarısı için Alıcı İşletim Karakteristiği (AİK) (ROC-Receiver Operating Characteristic) eğrisi kullanılmıştır.

ESA, PatchESA ve KSA kullanılarak yapılan sınıflandırma sonuçları Çizelge 5.4'te gösterilmektedir. Çizelge 5.2'de verilen her segmentasyon modelinden elde edilmiş segmentlenmiş görüntüler kullanılarak sınıflandırma yapılmıştır. Ayrıca, orijinal BT görüntüleri de karşılaştırma için her sınıflandırıcıya girdi olarak verilerek, sonuçlar alınmıştır.

Çizelge 5.4'ten görüleceği gibi %99.2 doğruluk ve F1-skoru ile en iyi sonuçlar, kÜÇA-UNet modeli ile segmentasyon sonucunda elde edilen görüntülerle ESA ile yapılan sınıflandırma işleminde elde edilmiştir. PSPNet segmentasyon modeli ile tüm sınıflandırıcılar en kötü sonuçları vermiştir. Tüm sınıflandırıcılar için en iyi sonuçları veren segmentasyon modeli kÜÇA-UNet olmuştur. Bu nedenle kÜÇA-UNet ile sınıflandırıcılara ait karışıklık (confusion) matrisleri Şekil 5.3'te verilmiştir.

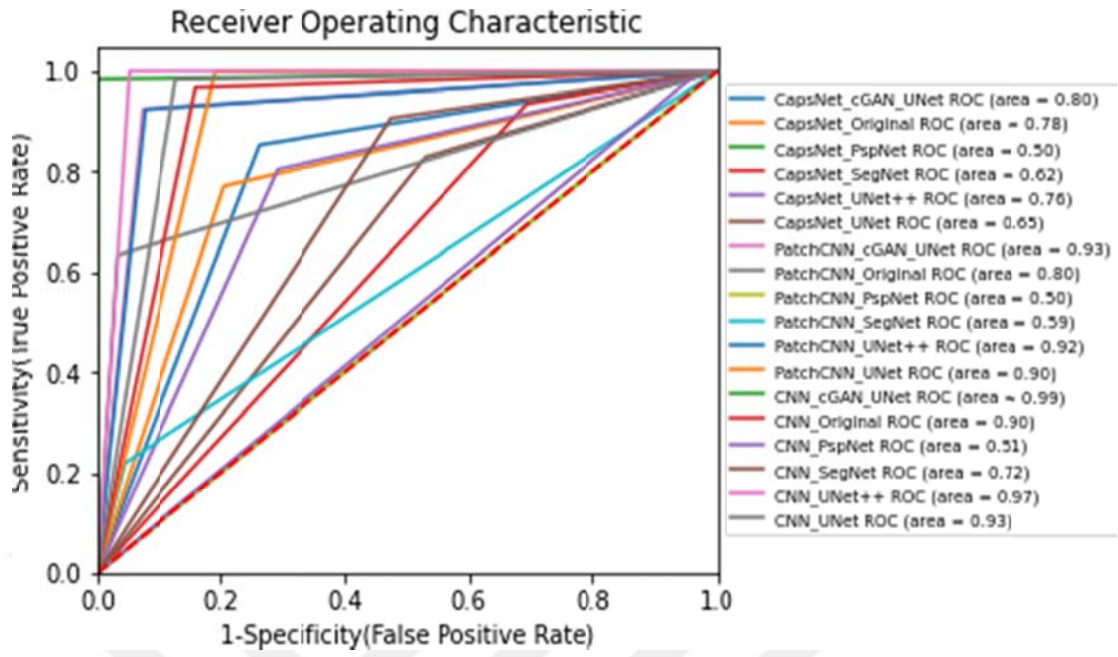


Şekil 5.3. kÜÇA-UNet segmentasyon modeli kullanan sınıflandırıcılara ait karışıklık matrisleri

Şekil 5.4 ise, tüm sınıflandırma modellerinin Alıcı İşletim Karakteristiği (AİK) (ROC-Receiver Operating Characteristic) eğrilerini göstermektedir.

Çizelge 5.4. Veri Kümesi-2’de farklı segmentasyon modelleri ile elde edilen segmentlenmiş görüntüler kullanılarak yapılan sınıflandırma işlemlerinden elde edilen sonuçlar (%)

Sınıflandırıcı	Segmentasyon Yöntemi	Doğruluk	Keskinlik	Duyarlılık	Özgüllük	F1 skoru
ESA (CNN)	Orijinal BT	90.34	96.34	84.06	96.7	89.79
	UNet	92.76	98.20	87.25	98.37	92.40
	UNet++	97.38	100	94.82	100	97.34
	SegNet	71.43	85.16	52.59	90.65	65.02
	PSPNet	50.90	88.88	0.03	99.59	0.06
	kÜÇA-UNet (5-kat çapraz doğrulama olmadan)	99.20	98.43	100	98.37	99.20
	kÜÇA-UNet (5-kat çapraz doğrulama ile)	98.75	98.99	98.56	98.95	98.76
PatchESA (PatchCNN)	Orijinal BT	79.88	72.12	96.75	63.55	82.64
	UNet	90.54	100	80.89	100	89.44
	UNet++	92.26	92.53	92.23	92.40	92.46
	SegNet	58.35	54.52	95.52	21.91	69.42
	PSPNet	50.50	26.0	51.02	50.1	34.01
	kÜÇA-UNet (5-kat çapraz doğrulama olmadan)	92.55	92.80	92.43	92.68	92.61
	kÜÇA-UNet (5-kat çapraz doğrulama ile)	90.95	91.2	90.84	91.06	91.02
KSA (CapsNet)	Orijinal BT	78.29	76.87	79.54	77.09	78.18
	UNet	64.79	73.75	47.01	82.93	57.42
	UNet++	75.77	77.77	70.87	80.49	74.17
	SegNet	61.77	82.79	30.68	93.50	44.76
	PSPNet	50.50	50.50	1.0	0.0	67.11
	kÜÇA-UNet (5-kat çapraz doğrulama olmadan)	73.84	77.62	67.73	80.08	72.34
	kÜÇA-UNet (5-kat çapraz doğrulama ile)	69.11	75.33	57.62	78.89	62.60



Şekil 5.4. Sınıflandırma işlemi için tüm modellerin AİK (ROC) eğrileri

6. SONUÇ VE ÖNERİLER

6.1. Sonuçlar

2019 yılının sonlarında Covid-19'un ortaya çıkmasıyla birlikte, dünya büyük bir sağlık kriziyle karşı karşıya kalmıştır. Covid-19'un son derece bulaşıcı doğası göz önüne alındığında, Covid-19 ile mücadele etmeye yönelik en önemli adım; virüsün yayılmasını sınırlamak için enfekte kişilerin etkili bir şekilde taranması, erken tespit edilmesi, izolasyonu ve hızlı bir şekilde tedavisinin başlamasıdır. Bu amaçla, röntgenler veya BT taramaları gibi tıbbi görüntüler üzerinden Covid-19 taraması, hastalığın hızlı teşhisinde alternatif bir çözümdür. Bu tez çalışmasında, BT görüntülerinden Covid-19 lezyonlarının segmentasyonu için yeni bir hibrit derin öğrenme yöntemi önerilmiştir. Ayrıca Covid-19 hastası ve sağlıklı kişilerin sınıflandırılması amacıyla hem literatürdeki temel segmentasyon yöntemleri hem de önerilen hibrit segmentasyon yöntemi kullanılarak, üç farklı derin öğrenme sınıflandırıcısı ile de görüntü sınıflandırması yapılmıştır.

Tez kapsamında ilk olarak; teşhis koyma, hastalığın ilerlemesini izleme, muayene süresinin azalması ve doğruluğun iyileştirilmesi gibi konularda radyologlara yardımcı olabilecek ikili maskeler oluşturmak amacıyla literatürde yaygın olarak kullanılan UNet, UNet++, SegNet ve PSPNet segmentasyon modelleri kullanılmıştır. Ardından Koşullu Üretken Çekişmeli Ağ (kÜÇA) (cGAN-Conditional Generative Adversarial Networks) ile bu modeller hibrit kullanılarak segmentasyon başarısının arttırılması amaçlanmıştır. Hibrit modellerde kÜÇA ile UNet++, SegNet ve PSPNet ile yapılan segmentasyon işlemlerinde, segmente edilmiş görüntüler tamamen siyah görüntü sonuçları verdiği için dolayı, bir sonuç alınamamış ve bu hibrit modeller kullanılamamıştır. kÜÇA-UNet hibrit modeli ile yapılan segmentasyon işleminde ise başarılı sonuçlar elde edilmiştir.

Tez kapsamında önerilen kÜÇA-UNet yöntemi ile yapılan segmentasyon işleminden elde edilen sonuçlar, karşılaştırma metriklerine göre UNet, UNet++, SegNet ve PSPNet temel segmentasyon modellerin tek başına kullanılmasına kıyasla enfeksiyon bölgelerini saptamak için daha yüksek bir segmentasyon doğruluğu sağlamıştır. Elde edilen sonuçlara göre, segmentasyonda en çok kullanılan UNet modeline göre önerilen yöntem BT görüntü segmentasyonunda, Bük (IoU) metriğine göre % 9, diğer metriklerde ise (Zar Skoru (Dice score), Kesinlik (Precision), Duyarlılık (Recall) ve F1-skor (F1-score)) % 6

daha iyi sonuç vermiştir. En kötü performans ölçütüne sahip model, yalnızca %51.20 Zar skoru ve %37.66 Bük (IoU) skoru ile PSPNet olmuştur. Yapılan çalışmalardan elde edilen yüksek başarı sonuçları, UNet gibi mevcut modellere çekışmeli eğitimin eklenmesinin, segmentasyonun doğruluğunu arttırdığını göstermektedir.

Segmentasyon işleminin ardından teşhis sürecini daha da kolaylaştırmak adına, BT görüntülerine sınıflandırma yapılmıştır. Covid-19 tespiti için Evrişimsel Sinir Ağı (ESA) (CNN), PatchESA (PatchCNN) ve Kapsül Sinir Ağı (KSA) (CapsNet) olarak üç sınıflandırma modeli kullanılmıştır. Bu sınıflandırıcılar, segmentasyon sonucunda elde edilen ikili maskeyi girdi olarak alarak, Covid-19'un var olup olmadığının tahmin edilmesi için kullanılmıştır.

Sınıflandırma modellerinin performansını değerlendirmek ve sistemlerin başarısını test etmek için Doğruluk (accuracy), Kesinlik (Precision), Duyarlılık (Recall), Özgüllük (Specifity) ve F1-skor (F1-score) kullanılmıştır. Ayrıca, sınıflandırma başarısı için Alıcı İşletim Karakteristiğı (AİK) (ROC-Receiver Operating Characteristic) eğrisi de kullanılmıştır. Yapılan sınıflandırma işlemlerinden elde edilen sonuçlar, BT taramaları üzerinden lezyonların segmentlere ayrılmasının ve derin öğrenme ile sınıflandırılması ile Covid-19 hastaları ile sağlıklı hastaların ayırt edebileceğini göstermiştir.

PatchESA; “yama” olarak tanımlanan, girdi görüntüsünün belirli bir boyutta ya da sayıda daha küçük görüntülere bölünmesiyle çalışan bir mimaridir. Sınıflandırma aşamasında PatchESA için farklı yama boyutları denenmiştir. Elde edilen sonuçlar, PatchESA modelinin performansının, yama boyutu arttıkça arttığını göstermektedir. Bu sonuç, büyük yamaların aksine daha küçük yama boyutlarında yeterli tanı bilgisinin taşınmadığına işaret etmektedir. Bununla birlikte, PatchESA sınıflandırıcısının sonuçları kötü olmasa da ESA sınıflandırıcısı çok daha iyi performans sergilemiştir. PatchESA ile yapılan sınıflandırma işleminde, kÜÇA-UNet ve UNet++ ile elde edilen sonuçlar çok yakındır ve hepsi yaklaşık %92 doğruluk göstermiştir.

KSA sınıflandırıcısından elde edilen sonuçlar ise, sonuçların diğer sınıflandırıcılardan önemli ölçüde daha düşük olması nedeniyle modelin ikili maskeler üzerinde eğitim için uygun olmadığını göstermektedir. KSA sınıflandırıcısı ile orijinal BT görüntüleri ile yapılan sınıflandırma işleminde %78.29 oranında en iyi doğruluk elde edilmiş, ardından %75.77'lik doğrulukla UNet++ ve %73.84'lük doğrulukla kÜÇA-UNet gelmiştir.

Sınıflandırma sonucunda elde edilen sonuçlar incelendiğinde, sınıflandırıcıların kÜÇA-UNet, UNet ve UNet++’dan oluşturulan ikili maskeye uygulanmasının, orijinal BT kullanılarak yapılan sınıflandırma işlemine göre daha iyi performans gösterdiği gözlenmiştir. Öte yandan SegNet ve PSPNet modellerinden elde edilen segmentlenmiş görüntüler kullanılarak yapılan sınıflandırma sonuçları, orijinal BT görüntüleri kullanılarak yapılan sınıflandırma sonuçlarından daha düşüktür.

Tez kapsamında önerilen kÜÇA-UNet modeli, ayrıca Covid-19 segmentasyonu ve sınıflandırması için literatürdeki yöntemlerle de karşılaştırılmıştır. Karşılaştırma yapılan çalışmalarda farklı veri setleri kullanılmıştır. Ancak segmentasyon için literatürde aynı veri türü kullanılmış olan çalışmalar seçilmiştir. Segmentasyon sonuçlarına ait karşılaştırmalar Çizelge 6.1’de verilmiştir.

Çizelge 6.1. Segmentasyon sonuçlarının literatür ile karşılaştırılması

Yöntem	Kullanılan görüntü formatı	Görüntü sayısı	Zar skoru (%) (Dice score)
Inf-Net (Fan ve ark., 2020)	BT	100	68.2
RC-CoSinGAN (Zhang ve ark., 2020)	BT	100	71.3 ± 19
Semi-Inf-Net (Fan ve ark., 2020)	BT	100	73.9
JCS (sınıflandırmadan gelen özellik olmadan) (Wu ve ark., 2021)	BT	3855	77.5
JCS (sınıflandırma modelinden gelen özellikler ile) (Wu ve ark., 2021)	BT	3855	78.3
Kodlayıcı-Kod çözücü (Amyar ve ark., 2020)	BT	100	88.0
VB-Net (Shan ve ark., 2020)	BT	300	91.6 ± 10
UNet (Ma ve ark., 2020)	BT	2112	67.3
kÜÇA-UNet (önerilen yöntem)	BT	2112	92.31

Çizelge 6.1’de gösterildiği üzere BT görüntüleri için literatür çalışmaları; Fan ve ark. (2020) çalışmasında elde edilen %68.2’lik bir Zar skorundan, Shan ve ark. (2020) çalışmasında elde edilen %91.6’lık bir skora kadar farklılık göstermektedir. Segmentasyon sonuçlarına göre tez kapsamında önerilen yöntem literatürdeki yöntemlerden daha yüksek başarı sağlamıştır. Ayrıca %91.6 başarı sağlayan yönteme göre, Veri Kümesi-1 kullanılarak fazla sayıda görüntü üzerinde bile başarının artmış olduğu görülmektedir.

Sınıflandırma işlemi ile elde edilen sonuçlar da literatür ile karşılaştırılmıştır. Literatürde de sınıflandırma çalışmalarında da farklı veri setleri ve türleri kullanılmıştır. Ancak tez kapsamında elde edilen sonuçların literatürdeki sonuçlar ile karşılaştırmalı analiz edilebilmesi için sonuçlar Çizelge 6.2’de verilmiştir.

Çizelge 6.2’de verilen sonuçlar, Loey ve ark. tarafından gerçekleştirilen (Loey ve ark., 2020) çalışmada elde edilen %66.67’lik bir doğruluktan, Karakanis ve Leontidis tarafından yapılan çalışmada (Karakanis ve Leontidis, 2021) elde edilen %98.7’lik bir doğruluk arasında değişen en gelişmiş sınıflandırma modellerini göstermektedir. Elde edilen sonuçlara göre, segmentasyonda olduğu gibi sınıflandırmada da literatüre göre daha başarılı sonuçlar alındığı görülmektedir. Elde edilen bu sonuçlar, BT görüntülerine doğru segmentasyon modelinin uygulanmasının sınıflandırma modelinin performansını önemli ölçüde artırabileceğini göstermektedir.

6.2. Öneriler

Tez kapsamında önerilen yöntemler, temel ve en gelişmiş modellerden daha iyi performans sergiledikleri için umut verici sonuçlar sağlamıştır. Önerilen segmentasyon yöntemi, %92.31 Zar skoru (Dice score) ve %86.41 Bük (IoU) skoru elde etmiştir. Sınıflandırma sonuçlarına gelince en iyi model, gerçek ikili maskesi olmayan bağımsız bir veri kümesinde %99.20’lik bir doğruluk ve F1 skoru elde etmiştir. Böylelikle önerilen yöntem, sağlık çalışanlarına özellikle Covid-19 teşhisinde yardımcı olma potansiyeline sahiptir ve sağlık tesislerine sınırlı erişimi olan bölgelerde kullanılabilir.

Önerilen segmentasyon modeli, diğer tüm temel modellerden daha iyi bir performans elde etmesine rağmen, modeli eğitmek için geçen süre (yaklaşık 72 saat) uzundur. Ancak, tüm modeller için sonuç alma süresi, bir görüntü için takribi 2 saniye olarak aynıdır. Daha yüksek bir bilgisayar gücü ile önerilen yöntem, daha hızlı sonuçlar üretebilir.

Çizelge 6.2. Sınıflandırma sonuçlarının literatür ile karşılaştırılması

Yöntem	Kullanılan görüntü formatı	Görüntü sayısı	Doğruluk (%)	Kesinlik (%)	Duyarlılık (%)	Özgüllük	F1 skor
UNet+DeCoVNet (Zhang ve ark., 2019)	BT	70: Covid-19 1431: Sağlıklı 1008: Pnömoni	90.8	---	96	70.7	---
AlexNet (Loey ve ark., 2020)	Röntgen	60: Covid-19 70: Sağlıklı	66.7	---	66.7	---	---
ResNet18 (Loey ve ark., 2020)		70: Pnömoni_bakteri	69.47	---	66.67	---	---
GoogleNet (Loey ve ark., 2020)		70: Pnömoni viral	80.56	---	80.56	---	---
ESA+DVM (Sethy ve ark., 2020)	Röntgen	127: Covid-19 127: Pnömoni 127: Sağlıklı	95.33	---	95.33	---	95.34
ESA (Yang ve ark., 2020)	BT	349: Covid-19 483: Sağlıklı	89	---	---	---	90
Deep Pneumonia (Song ve ark., 2020)	BT	88: Covid-19 101: Pnömoni 86: Sağlıklı	86	79	96	---	87
DarkCovidNet (Ozturk ve ark., 2020)	Röntgen	125: Covid-19 500: Sağlıklı 500: Pnömoni	87.02	89.96	85.35	92.18	87.37
DarkCovidNet (Ozturk ve ark., 2020)	Röntgen	125: Covid-19 500: Sağlıklı	98.08	98.03	95.13	95.3	96.51
VGG-19 (Apostolopoulos ve Mpesiana, 2020)	Röntgen	224: Covid-19 700: Pnömoni 504: Sağlıklı	96.78	---	98.66	96.46	---
Covid-Net (Wang ve ark., 2020)	Röntgen	53: Covid-19 5526: Covid-19 olmayan hasta 8066: Sağlıklı	92.4	---	---	---	---
Covid-MobileXpert (Li ve Zhu, 2020)	Röntgen	179: Covid-19 179: Pnömoni 37: Sağlıklı	88.9	---	---	---	---
Covidx-Net (Hemdan ve ark., 2020)	Röntgen	25: Covid-19 25: Sağlıklı	90	91.5	90	---	91
UNet+3D DNN (Zheng ve ark., 2020)	BT	313: Covid-19 229: Sağlıklı	90.8	---	90.7	91.11	---

Çizelge 6.2. Sınıflandırma sonuçlarının literatür ile karşılaştırılması (devam)

Kodlayıcı-kod çözücü (Amyar ve ark., 2020)	BT	449: Covid-19 98: Akciğer hastası 425: Sağlıklı 2738: Covid-19 olmayan hasta	94.67	---	96	92	---
(UNet++)+CNN(Chen ve ark., 2020)	BT	20886: Covid-19 14469: Sağlıklı	92.59	---	100	81.82	---
ResNet20 (Ghoshal ve Tucker, 2020)	Röntgen	68: Covid-19 1583: Sağlıklı 2786: Pnömoni bakteri 1504: Pnömoni viral	92.8	---	---	---	---
Inception (Wang ve ark., 2020)	BT	325: Covid-19 740: Pnömoni	79.3	---	83	67	---
ResNet50 (Narin ve ark., 2020)	Röntgen	50: Covid-19 50: Sağlıklı	96.1	76.5	91.8	96.6	83.5
ResNet+Location Attention (Xu ve ark., 2020)	BT	219: Covid-19 175: Sağlıklı 224: Viral Pnömoni	86.7	86.9	86.7	---	86.7
CovNet (Li ve ark., 2020)	BT	468: Covid-19 1551: Pnömoni 1173: Pnömoni değil	95	---	87	92	---
JCS (Wu ve ark., 2021)	BT	64711: Covid-19 68041: Sağlıklı	---	---	94.5	93.5	---
ESA (Karakanis ve Leontidis, 2021)	BT	275: Covid-19 275: Sağlıklı	98.7	---	100	98.3	---
GoogleNet (Angelov ve Soares, 2020)	BT	1252: Covid-19 1229: Sağlıklı	88.6	---	89.7	88.6	89.2
kÜÇA-UNet + ESA	BT	1252: Covid-19 1229: Sağlıklı	99.2	98.43	100	98.37	99.2

Aşırı öğrenmeyi önlemek için toplu normalleştirme, seyreltme, katmanlı çapraz doğrulama gibi çeşitli yöntemler kullanılmıştır. Sonuçları daha fazla doğrulamak için gelecekte hastanelerden daha büyük bir veri kümesi elde edilebilir. Dolayısıyla, modelin farklı konumlardaki genelleme derecesi değerlendirilebilir. Ayrıca çalışmada kullanılan çok daha fazla katmana sahip bir KSA (CapsNet) modelinin, daha iyi sonuçlar doğurabileceği öngörülmektedir.

Buna ek olarak, üretilen segmentlenmiş görüntülerden gürültülü küçük yamaları çıkarabilen, daha başarılı son işlem yöntemleri uygulamak, elde edilen sonuçları iyileştirebilir. Son olarak bu çalışmada kullanılan tüm modellerin temel birimi, kara kutu modeli olan “sinir ağıdır” ve açıklanabilirliği üzerine çalışmalar hâlâ devam etmektedir. İlerleyen zamanlarda derin öğrenme model yapıları daha net bir şekilde açıklanabilirse, çok farklı ve daha başarılı çalışmalar gerçekleştirilebileceği düşünülmektedir.

KAYNAKLAR

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., 2016, Tensorflow: A system for large-scale machine learning, in: *Symposium on Operating Systems Design and Imp. (OSDI'16)*. syf. 265–283.
- Ai, T., Yang, Z., Hou, H., Zhan, C., Chen, C., Lv, W., Tao, Q., Sun, Z., ve Xia, L., 2020, Correlation of chest CT and RT-PCR testing in coronavirus disease 2019 (COVID-19) in China: a report of 1014 cases, *Radiology* 200642.
- Amyar, A., Modzelewski, R., Li, H., Ruan, S., 2020, Multi-task deep learning based CT imaging analysis for COVID-19 pneumonia: Classification and segmentation, *Computers in Biology and Medicine* 126, 104037.
- Angelov, P., ve Soares, E., 2020, Explainable-by-design approach for covid-19 classification via ct-scan, *medRxiv*.
- Apostolopoulos, I.D., ve Mpesiana, T.A., 2020, Covid-19: automatic detection from x-ray images utilizing transfer learning with convolutional neural networks, *Physical and Engineering Sciences in Medicine* 43, syf. 635–640.
- Arsalan, M., Kim, D.S., Owais, M., Park, K.R., 2020, OR-Skip-Net: Outer residual skip network for skin segmentation in non-ideal situations, *Expert Systems with Applications* 141, 112922.
- Badrinarayanan, V., Kendall, A., Cipolla, R., 2017, Segnet: A deep convolutional encoder-decoder architecture for image segmentation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, syf. 2481–2495.
- Biswas, M., Kuppili, V., Saba, L., Edla, D.R., Suri, H.S., Cuadrado-Godia, E., Laird, J.R., Marinho, R.T., Sanches, J.M. ve Nicolaides, A., 2019, State-of-the-art review on deep learning in medical imaging, *Frontiers in bioscience (Landmark edition)* 24, syf. 392–426.
- Brownlee, J., 2019, How to Implement GAN Hacks in Keras to Train Stable Models, *Machine Learning Mastery*.
- Çarkacı, N., 2018, Derin Öğrenme Uygulamalarında En Sık kullanılan Hiper-parametreler, <https://medium.com/deep-learning-turkiye/derin-ogrenme-uygulamalarinda-en-sik-kullanilan-hiper-parametreler-ece8e9125c4> [Erişim: Ocak-2021].
- Çarkacı, N., 2017, Derin öğrenme uygulamalarında temel kavramlar : perceptron, skor fonksiyonu ve hata hesaplaması(loss function) <https://www.linkedin.com/pulse/derin-ogrenme-uygulamalarinda-temel-kavramlar-skor-ve-çarkacı/> [Erişim: Mayıs-2021].
- Ceraolo, C. ve Giorgi, F.M., 2020, Genomic variance of the 2019-nCoV coronavirus, *Journal of Medical Virology*.
- Chen, J., Wu, L., Zhang, J., Zhang, L., Gong, D., Zhao, Y., Hu, S., Wang, Y., Hu, X., Zheng, B., 2020, Deep learning-based model for detecting 2019 novel coronavirus pneumonia on high-resolution computed tomography: a prospective study, *MedRxiv*.
- Chen, X., Yao, L., Zhang, Y., 2020, Residual attention UNet for automated multi-class segmentation of covid-19 chest ct images, *arXiv preprint arXiv:2004.05645*.
- Chhawahharia, S., Mohapatra, S., Sahoo, G., 2016, Improved Lymphocyte Image Segmentation Using Near Sets for ALL Detection, in: *Handbook of Research on Emerging Perspectives in Intelligent Pattern Recognition, Analysis, and Image Processing*. IGI Global, syf. 317–334.
- Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016, 3D UNet: learning dense volumetric segmentation from sparse annotation, in: *International*

- Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, syf. 424–432.
- Contributors., W., 2020, Adaptive histogram equalization, *Wikipedia, The Free Encyclopedia*.
- Cucinotta, D., Vanelli, M., 2020, WHO declares COVID-19 a pandemic, *Acta Biomedica*.
- Dawood, F.A.A., Rahmat, R.W., Dimon, M.Z., Nurliyana, L., Kadiman, S. Bin, 2011, Automatic boundary detection of wall motion in two-dimensional echocardiography images, *Journal of Computer Science* 7(8), syf.1261-1266.
- Denton, E.L., Chintala, S. ve Fergus, R., 2015, Deep generative image models using a laplacian pyramid of adversarial networks, *Advances in Neural Information Processing Systems* 28, syf. 1486–1494.
- Fan, D.-P., Zhou, T., Ji, G.-P., Zhou, Y., Chen, G., Fu, H., Shen, J. ve Shao, L., 2020, Inf-Net: Automatic COVID-19 Lung Infection Segmentation from CT Images, *IEEE Transactions on Medical Imaging* 39(8), syf. 2626-2637.
- Fang, Y., Zhang, H., Xie, J., Lin, M., Ying, L., Pang, P., Ji, W., 2020, Sensitivity of chest CT for COVID-19: comparison to RT-PCR, *Radiology* 200432.
- Francesco, T., Chiara, M., Dario, G., Daniele, B., 2019, An artificial neuron implemented on an actual quantum processor, *NPJ Quantum Information* 5 (26)
- Ghoshal, B. ve Tucker, A., 2020, Estimating uncertainty and interpretability in deep learning for coronavirus (COVID-19) detection, *arXiv preprint arXiv:2003.10769*.
- Goodfellow, I., Bengio, Y., Courville, A. ve Bengio, Y., 2016, Deep learning. MIT press Cambridge.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y., 2014, Generative adversarial nets, *Advances in Neural Information Processing Systems* 27, syf. 2672–2680.
- Hemdan, E.E.-D., Shouman, M.A., Karar, M.E., 2020, Covidx-net: A framework of deep learning classifiers to diagnose covid-19 in x-ray images, *arXiv preprint arXiv:2003.11055*.
- Herper, M. ve Branswell, H., 2020, Shortage of crucial chemicals creates new obstacle to US coronavirus testing, *STAT News*.
- Hu, Z., Fu, Y., Luo, Y., Xu, X., Xia, Z., Zhang, H., 2020, Speaker recognition based on short utterance compensation method of generative adversarial networks, *International Journal of Speech Technology* 23, syf. 443-450.
- Isola, P., Zhu, J.-Y., Zhou, T. ve Efros, A.A., 2017, Image-to-image translation with conditional adversarial networks, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. syf. 1125–1134.
- Jawerth, N., 2020, How is the COVID-19 virus detected using real time RT-PCR?, *IAEA Bulletin*.
- Karakanis, S., Leontidis, G., 2021, Lightweight deep learning models for detecting COVID-19 from chest X-ray images, *Computers in Biology and Medicine* 130, 104181.
- Kodym, O., Španěl, M., Herout, A., 2018, Segmentation of head and neck organs at risk using cnn with batch dice loss, in: *German Conference on Pattern Recognition*. Springer, syf. 105–114.
- Kutlu, B., Badur, B. , 2009, Yapay Sinir Ağları ile Borsa Endeksi Tahmini. *Yönetim Dergisi*, 20(63), syf. 40-45.

- Läthén, G., 2010, Segmentation methods for medical image analysis, *Linköping Studies in Science and Technology Thesis 2*, syf. 1403–1408.
- Li, L., Qin, L., Xu, Z., Yin, Y., Wang, X., Kong, B., Bai, J., Lu, Y., Fang, Z., Song, Q., 2020, Artificial intelligence distinguishes COVID-19 from community acquired pneumonia on chest CT, *Radiology* 296 (2), E65-E71.
- Li, X. ve Zhu, D., 2020, Covid-xpert: An ai powered population screening of covid-19 cases using chest radiography images, *arXiv preprint arXiv:2004.03042*.
- Lin, C., Ding, Y., Xie, B., Sun, Z., Li, X., Chen, Z., Niu, M., 2020, Asymptomatic novel coronavirus pneumonia patient outside Wuhan: The value of CT images in the course of the disease, *Clinical Imaging* 63, syf. 7–9.
- Liu, X., Faes, L., Kale, A.U., Wagner, S.K., Fu, D.J., Bruynseels, A., Mahendiran, T., Moraes, G., Shamdass, M., Kern, C., 2019, A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: a systematic review and meta-analysis, *The Lancet Digital Health* 1, e271–e297.
- Loey, M., Smarandache, F., M. Khalifa, N.E., 2020, Within the lack of chest COVID-19 X-ray dataset: a novel detection model based on GAN and deep transfer learning, *Symmetry* 12, 651.
- Long, M., Cao, Y., Wang, J., Jordan, M., 2015, Learning transferable features with deep adaptation networks, in: *International Conference on Machine Learning*. PMLR, syf. 97–105.
- Ma, J., Wang, Y., An, X., Ge, C., Yu, Z., Chen, J., Zhu, Q., Dong, G., He, J., He, Z., 2020, Towards Efficient COVID-19 CT Annotation: A Benchmark for Lung and Infection Segmentation, *arXiv preprint arXiv:2004.12537*.
- Meng, H., Xiong, R., He, R., Lin, W., Hao, B., Zhang, L., Lu, Z., Shen, X., Fan, T., Jiang, W., 2020, CT imaging and clinical course of asymptomatic cases with COVID-19 pneumonia at admission in Wuhan, China, *Journal of Infection* 81(1), E33-E39.
- Milletari, F., Navab, N., Ahmadi, S.-A., 2016, V-net: Fully convolutional neural networks for volumetric medical image segmentation, in: *2016 Fourth International Conference on 3D Vision (3DV)*. IEEE, syf. 565–571.
- Mino, A., Spanakis, G., 2018, LoGAN: Generating logos with a generative adversarial neural network conditioned on color, in: *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, syf. 965–970.
- Mirza, M., Osindero, S., 2014, Conditional generative adversarial nets, *arXiv preprint arXiv:1411.1784*.
- Mittal, M., Goyal, L.M., Kaur, S., Kaur, I., Verma, A., Hemanth, D.J., 2019, Deep learning based enhanced tumor segmentation approach for MR brain images, *Applied Soft Computing* 78, syf. 346–354.
- Mossa-Basha, M., Meltzer, C.C., Kim, D.C., Tuite, M.J., Kolli, K.P., Tan, B.S., 2020, Radiology department preparedness for COVID-19: radiology scientific expert panel, *Radiology* 200988.
- Müller, D., Rey, I.S., Kramer, F., 2020, Automated Chest CT Image Segmentation of COVID-19 Lung Infection based on 3D UNet, *arXiv preprint arXiv:2007.04774*.
- Narin, A., Kaya, C., Pamuk, Z., 2020, Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks, *arXiv preprint arXiv:2003.10849*.

- Oktaý, O., Schlemper, J., Folgoc, L. Le, Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N.Y., Kainz, B., 2018, Attention UNet: Learning where to look for the pancreas, *arXiv preprint arXiv:1804.03999*.
- Paiva, O., 2020, Helping Radiologists To Help People In More Than 100 Countries!, *Coronavirus Cases*.
- Park, B., Park, H., Lee, S.M., Seo, J.B., Kim, N., 2019, Lung segmentation on HRCT and volumetric CT for diffuse interstitial lung disease using deep convolutional neural networks, *Journal of Digital Imaging* 32, syf. 1019–1026.
- Peng, D., Zhang, Y., Guan, H., 2019, End-to-end change detection for high resolution satellite images using improved UNet++. *Remote Sensing* 11(11), 1382.
- Ronneberger, O., Fischer, P., Brox, T., 2015, UNet: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer, syf. 234–241.
- Rosenblatt, F., 1957, The perceptron, a perceiving and recognizing automaton Project Para. Cornell Aeronautical Laboratory.
- Roy, K., Banik, D., Bhattacharjee, D., Nasipuri, M., 2019, Patch-based system for classification of breast histology images using deep learning. *Computerized Medical Imaging and Graphics*, 71, syf. 90-103.
- Sabour, S., Frosst, N., Hinton, G.E., 2017, Dynamic routing between capsules, *arXiv preprint arXiv:1710.09829*.
- Saglam, A., Baykan, N.A., 2017, Effects of color spaces and distance norms on graph-based image segmentation, in: 2017 3rd International Conference on Frontiers of Signal Processing (ICFSP). IEEE, syf. 130–135.
- Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X., 2016, Improved techniques for training gans, *arXiv preprint arXiv:1606.03498*.
- Sethy, P.K., Behera, S.K., Ratha, P.K., Biswas, P., 2020, Detection of coronavirus disease (COVID-19) based on deep features and support vector machine. *International Journal of Mathematical, Engineering and Management Sciences* 5(4), syf. 643-651.
- Shan, F., Gao, Y., Wang, J., Shi, W., Shi, N., Han, M., Xue, Z., Shi, Y., 2020, Lung infection quantification of covid-19 in ct images with deep learning, *arXiv preprint arXiv:2003.04655*.
- Song, Y., Zheng, S., Li, L., Zhang, Xiang, Zhang, Xiaodong, Huang, Z., Chen, J., Zhao, H., Jie, Y., Wang, R., 2020, Deep learning enables accurate diagnosis of novel coronavirus (COVID-19) with CT images, *MedRxiv*.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014, Dropout: a simple way to prevent neural networks from overfitting, *The Journal of Machine Learning Research* 15, syf. 1929–1958.
- Stephen, O., Sain, M., Maduh, U.J., Jeong, D.-U., 2019, An efficient deep learning approach to pneumonia classification in healthcare, *Journal of Healthcare Engineering* 2019, 4180949.
- Sumi, P. S., Delhibabu, R., 2019, Glioblastoma Multiforme Classification On High Resolution Histology Image Using Deep Spatial Fusion Network. in: CEUR Workshop Proceedings , syf. 109-120.
- Survivor, P., 2019, Shortage of Radiologists, *Pseudomyxoma Survivor*.
- Ulyanov, D., Vedaldi, A., Lempitsky, V., 2016, Instance normalization: The missing ingredient for fast stylization, *arXiv preprint arXiv:1607.08022*.

- Voiculescu, I., Yeghiazaryan, V., 2015, An Overview of Current Evaluation Methods Used in Medical Image Segmentation.
- Wang, K., Kang, S., Tian, R., Zhang, X., Wang, Y., 2020, Imaging manifestations and diagnostic value of chest CT of coronavirus disease 2019 (COVID-19) in the Xiaogan area, *Clinical Radiology* 75(5), syf. 341-347.
- Wang, L., Lin, Z.Q., Wong, A., 2020, COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images, *Scientific Reports*.
- Wang, S., Kang, B., Ma, J., Zeng, X., Xiao, M., Guo, J., Cai, M., Yang, J., Li, Y., Meng, X., 2020, A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19), *MedRxiv*.
- WHO, 2021, WHO Coronavirus(Covid-19) Dashboard <https://covid19.who.int/> [Eriřim: Mayıs-2021]
- WHO, 2020, Considerations for quarantine of individuals in the context of containment for coronavirus disease (COVID-19), *Who*.
- Won, J., Lee, S., Park, M., Kim, T.Y., Park, M.G., Choi, B.Y., Kim, D., Chang, H., Kim, V.N., Lee, C.J., 2020, Development of a laboratory-safe and low-cost detection protocol for SARS-CoV-2 of the Coronavirus Disease 2019 (COVID-19), *Experimental Neurobiology* 29(2), syf. 107-119.
- Wu, Y.-H., Gao, S.-H., Mei, J., Xu, J., Fan, D.-P., Zhang, R.-G., Cheng, M.-M., 2021, Jcs: An explainable covid-19 diagnosis system by joint classification and segmentation, *IEEE Transactions on Image Processing* 30, syf. 3113–3126.
- Xing, Y., Zhong, L., Zhong, X., 2020, An Encoder-Decoder Network Based FCN Architecture for Semantic Segmentation, *Wireless Communications and Mobile Computing* 2020.8861886
- Xu, X., Jiang, X., Ma, C., Du, P., Li, X., Lv, S., Yu, L., Ni, Q., Chen, Y., Su, J., 2020, A deep learning system to screen novel coronavirus disease 2019 pneumonia, *Engineering* 6(10), syf. 1122-1129.
- Yang, X., He, X., Zhao, J., Zhang, Y., Zhang, S., Xie, P., 2020, Covid-ct-dataset: a ct scan dataset about covid-19, *arXiv preprint arXiv:2003.13865*.
- Zhang, J., Xie, Y., Wu, Q., Xia, Y., 2019, Medical image classification using synergic deep learning, *Medical Image Analysis* 54, syf. 10–19.
- Zhang, P., Zhong, Y., Deng, Y., Tang, X., Li, X., 2020, CoSinGAN: Learning COVID-19 Infection Segmentation from a Single Radiological Image, *Diagnostics* 10, 901.
- Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J., 2017, Pyramid scene parsing network, in: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. syf. 2881–2890.
- Zheng, C., Deng, X., Fu, Q., Zhou, Q., Feng, J., Ma, H., Liu, W., Wang, X., 2020, Deep learning-based detection for COVID-19 from chest CT using weak label, *MedRxiv*.
- Zhou, T., Canu, S., Ruan, S., 2020, An automatic COVID-19 CT segmentation based on UNet with attention mechanism, *arXiv preprint arXiv:2004.06673*.
- Zhou, Z., Siddiquee, M.M.R., Tajbakhsh, N., Liang, J., 2019, Unet++: Redesigning skip connections to exploit multiscale features in image segmentation, *IEEE Transactions on Medical Imaging* 39, syf. 1856–1867.
- Zuiderveld, K., 1994, Contrast limited adaptive histogram equalization, *Graphics gems* 474–485.