İZMİR KÂTİP ÇELEBİ ÜNİVERSİTESİ
2010
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

Artificial Intelligent Based Segmentation on Medical Imaging

Submitted to the Graduate School of Natural and Applied Sciences in partial fulfillment of the requirements for the degree of

Master of Science

in Department of Electric and Electronics Engineering

by

Mahmut AĞRALI

ORCID 0000-0002-5508-2854

June, 2023

This is to certify that we have read the thesis **Artificial Intelligent Based Segmentation on Medical Imaging** submitted by **Mahmut AĞRALI**, and it has been judged to be successful, in scope and in quality, at the defense exam and accepted by our jury as a MASTER'S.

APPROVED BY:

Advisor: **Assoc. Prof. Dr. Volkan KILIÇ**
İzmir Kâtip Çelebi University

Committee Members:

Assoc. Prof. Dr. Aytuğ ONAN
İzmir Kâtip Çelebi University

Assist. Prof. Dr. Emin BORANDAĞ
Manisa Celal Bayar University

Date of Defense: June 02, 2023

Declaration of Authorship

I, **Mahmut AĞRALI**, declare that this thesis titled **Artificial Intelligent Based Segmentation on Medical Imaging** and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for the Master's / Doctoral degree at this university.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this university or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. This thesis is entirely my own work, with the exception of such quotations.
- I have acknowledged all major sources of assistance.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: 02.06.2023

Artificial Intelligent Based Segmentation on Medical Imaging

Abstract

Over the past two decades, the increasing development of technology and theoretical knowledge has led to the growing application of artificial intelligence (AI) in various fields, including healthcare. AI-assisted diagnosis of public health threats, such as COVID-19 and ischemic stroke, leads to improved diagnostic accuracy and reduced workload for healthcare professionals. In addition, the segmentation of disease regions on medical images provides helpful information that can assist healthcare professionals in treatment. However, medical image segmentation still faces challenges such as data scarcity and the small size of disease regions. The thesis has addressed these challenges in three main contribution chapters, using data augmentation methods and new approaches that can focus on disease regions.

First, a framework called DeepChestNet was proposed for COVID-19 lesion detection from medical images. Convolutional Neural Networks (CNN) are still limited in capturing the structured relationships of the lesions since the texture and shape of COVID-19 can be easily confused with other structures in the image. To minimize this limitation, the proposed framework combines the segmentation of the lung, pulmonary lobes, and COVID-19 lesions. The combined segmentation tasks enable DeepChestNet to better capture the structured relationships of the lesions and detect the findings of COVID-19. In addition, new large-scale *DeepChestNet-Lung*, *DeepChestNet-Lobe*, and *DeepChestNet-COVID* datasets containing axial chest computed tomography (CT) images with 9036, 9036, and 1034 annotated pixels, respectively, have been collected for the segmentation tasks in this chapter.

Next, a novel architecture called *U-TranSvision* was proposed due to the poor performance of CNN architectures in extracting the salient features of small COVID-19 regions on CT images. Our architecture leverages transformer and deep supervision approaches focusing on salient features to improve segmentation performance in images containing small COVID-19 lesions. In addition, to improve the segmentation

performance of *U-TranSvision*, the Pix2Pix generative adversarial network (GAN) was utilized for data augmentation. Next pre-processing steps were applied to reduce the noise around the human tissue in the CT images. To train our architecture, a relatively large dataset containing 11,717 axial chest CT images was created by adding images with small lesion regions to the dataset in the previous chapter. Furthermore, the robustness of our architecture was demonstrated by the experiments on three publicly available datasets, including COVID-19-CT-Seg, MosMedData, and MedSeg.

Finally, the ResUNet architecture was modified to improve the performance of ischemic stroke disease segmentation on CT images. Although CNN-based architectures have shown relatively good performance in ischemic stroke disease segmentation, the architectures can be modified to further improve performance. Therefore, a dilation convolution layer has been added to the ResUNet architecture, and it has been shown to increase the accuracy of the model. In addition, traditional data augmentation methods such as brightness changes, Gaussian and blur filters, adding salt and pepper, and Gaussian noise, as well as the Pix2Pix GAN approach, were applied to improve the generalization ability of the model. Furthermore, a public ischemic stroke dataset was used for comparison with similar studies.

Keywords: Artificial intelligence, deep learning, lung segmentation, pulmonary lobe segmentation, COVID-19 detection, COVID-19 lesion segmentation

Medikal Görüntü Üzerinden Yapay Zekâ Tabanlı Bölütleme

Öz

Teknolojinin ve teorik bilginin gelişmesiyle son 20 yıldır günlük hayatımızın her alanında olduğu gibi sağlık alanında da yapay zekâ (YZ)'nın uygulanması giderek artmaktadır. Özellikle koronavirüs (COVID-19) ve iskemik inme gibi toplum sağlığını tehdit eden hastalıkların YZ ile tespit edilmesi sağlık çalışanların iş yüklerini hafifleterek daha doğru tanı sağlamaktadır. Bu hastalıkların tespit edilmesinin yanında medikal görüntü üzerinden hastalık bölgelerinin bölütlenmesi, sağlık çalışanlarına tedavide yardımcı olabilecek bilgi sağlamaktadır. Bununla birlikte medikal görüntü üzerinden bölütlemenin hastalık bölgesinin küçük ve veri sayısının yetersiz olması gibi sorunları mevcuttur. Tez, veri artırma yöntemlerini ve hastalık bölgelerine odaklanabilen yeni yaklaşımları kullanarak üç ana katkı bölümünde bu sorunları ele almıştır.

İlk olarak COVID-19 lezyonunun medikal görüntüleri üzerinden tespiti için DeepChestNet olarak adlandırılan özgün bir çerçeve önerilmiştir. COVID-19'un dokusu ve şekli görüntüdeki diğer yapılarla karıştırılabileceğinden, evrişimli sinir ağları (ESA), lezyonun yapılandırılmış ilişkilerini yakalamada hala sınırlıdır. Bu kısıtlamayı azaltmak için önerilen çerçeve; akciğer, akciğer lob ve COVID-19 lezyon bölütlemesi işlemlerini birleştirmiştir. Bölütleme işlemlerinin birleştirilmesi DeepChestNet'in lezyonların yapılandırılmış ilişkilerini daha fazla yakalayabilmesini ve COVID-19'un bulgularının tespitini sağlamaktadır. Ayrıca, bu bölümde bölütleme işlemleri için sırasıyla 9036, 9036 ve 1034 tane piksel düzeyinde açıklamalı etiketlere sahip eksenel göğüs bilgisayarlı tomografi (BT) görüntüleri içeren yeni geniş ölçekli *DeepChestNet-Lung*, *DeepChestNet-Lobe* ve *DeepChestNet-COVID* veri kümeleri toplanmıştır.

Sonrasında, BT görüntülerindeki küçük COVID-19 bölgelerinin göze çarpan özelliklerini çıkarmada ESA mimarilerinin yetersiz olması nedeniyle *U-TranSvision* adlı özgün bir mimari önerilmiştir. Bu mimarimiz küçük COVID-19 lezyonları içeren

görüntülerde bölütleme başarımını artırmak için göze çarpan özelliklere odaklanabilen transformer ve derin denetim yaklaşımlarını kullanmaktadır. Ayrıca *U-TranSvision*'ın bölütleme başarımını iyileştirmek için veri artırmada Pix2Pix çekişmeli üretici ağ (GAN) kullanılmıştır. Daha sonra, BT görüntülerindeki insan dokusu etrafında bulunan gürültüyü gideren ön işleme adımları uygulanmıştır. Mimarimizi eğitmek için önceki bölümdeki veri kümemize küçük lezyon bölgelerini içeren görüntüleri de ekleyerek toplamda 11.717 eksenel göğüs BT görüntüsü içeren nispeten büyük ölçekli bir veri kümesi oluşturuldu. Ayrıca *U-TranSvision*'ın sağlamlığı, COVID-19-CT-Seg, MosMedData ve MedSeg gibi halka açık üç veri kümesi üzerindeki deneyler ile gösterildi.

Son olarak, iskemik inme hastalığının BT görüntüleri üzerinden tespit edilmesinde başarımı artırmak amacıyla ResUNet mimarisi değiştirilmiştir. ESA tabanlı mimariler iskemik inme hastalığının bölütlenmesinde göreceli olarak iyi başarım göstermesine rağmen, başarımı daha da artırmak için mimariler değiştirilebilmektedir. Bu nedenle, ResUNet mimarisine bir dilation konvolüsyon katmanı eklenmiştir ve bunun modelin doğruluğunu artırdığı gösterilmiştir. Ayrıca, parlaklık değişimi, Gaussian ve blur filtreleri uygulama, tuz biber ve Gauss gürültülerinin eklenmesi gibi geleneksel veri artırma yöntemlerinin yanı sıra Pix2Pix GAN yaklaşımı da kullanılarak modelin genelleme yeteneği artırılmıştır. Ayrıca, benzer çalışmalarla karşılaştırmak için halka açık iskemik inme veri kümesi kullanılmıştır.

Anahtar Kelimeler: Yapay zekâ, derin öğrenme, akciğer bölütlemesi, akciğer lobe bölütlemesi, COVID-19 tespiti, COVID-19 lezyon bölütlemesi

To my family…

Acknowledgment

First and foremost, I would like to thank my advisor, Assoc. Prof. Dr Volkan Kılıç, who guided me with his valuable knowledge and experience during my graduate education at Izmir Kâtip Çelebi University. He also always supported me in realizing my ideals and helped to improve my academic skills. I am grateful to Assoc. Prof. Dr Aytuğ Onan for his moral support for my thesis. I would also like to express my gratitude to the doctors from Izmir Kâtip Çelebi University and Bozyaka Training and Research Hospital for their help in preparing the dataset for my thesis.

I would like to thank my colleagues and friends in the IKCU AI Lab at Izmir Kâtip Çelebi University for their valuable comments and advice on this challenging path. I would also like to thank all my friends and professors who helped me throughout my graduate education.

Finally, I would like to thank my family, who has been by my side in my every decision and has never left me alone with their financial and moral support. I am grateful for their endless support, interest, and patience.

Table of Contents

Declaration of Authorship ii

Abstract iii

Öz… v

Acknowledgment viii

Table of Contents ix

List of Figures xiii

List of Tables xv

List of Abbreviations xvii

List of Symbols xix

1 Introduction 1

1.1 Segmentation Problems and Motivation 1

1.2 Contributions 4

1.3 Literature Survey 5

1.3.1 *DeepChestNet*: Artificial Intelligence Approach for COVID-19 Detection on Computed Tomography Images 5

1.3.2 *U-TranSvision*: Transformer-based Deep Supervision Approach for COVID-19 Lesion Segmentation on Computed Tomography Images 8

1.3.3 Artificial Intelligence in Healthcare: Ischemic Stroke Segmentation 10

1.4 Outline of the Thesis 11

2 Methods 12

2.1 Deep Learning Segmentation Architectures 12

2.1.1 U-Net…. 13

2.1.2 ResUNet 13

2.1.3 DenseU-Net 14
2.1.4 Attention U-Net 14
2.1.5 DeepLabV3+ 14
2.1.6 Backbone Architectures 15
2.2 Data Augmentation 16
2.2.1 Traditional Data Augmentation Approaches 16
2.2.2 GAN-based Data Augmentation Approaches 17
2.3 Image Processing 18
2.3.1 Otsu Method 18
2.3.2 Morphological Operations 19
2.4 Model Evaluation Approaches and Performance Metrics 20
2.4.1 *K*-Fold Cross-validation 20
2.4.2 Performance Metrics 20
2.5 Programming Languages and Tools 21
2.5.1 Python…. 22
2.5.2 MATLAB Image Labeler 22
2.5.3 MedSeg…. 23
3 *DeepChestNet*: Artificial Intelligence Approach for COVID-19 Detection on Computed Tomography Images 24
3.1 Introduction 25
3.2 *DeepChestNet* Framework 26
3.3 Data Acquisition and Pre-processing 28
3.4 DeepChestNet App 30
3.5 Results and Discussion 31
3.5.1 Experimental Settings 31
3.5.2 Artificial Intelligence-Based Segmentation and Detection 32
3.6 Chapter Summary 38

4 *U-TranSvision*: Transformer-based Deep Supervision Approach for COVID-19 Lesion Segmentation on Computed Tomography Images 40

4.1 Introduction 41

4.2 *U-TranSvision* Architecture 44

4.2.1 Encoder…. 44

4.2.2 Bridge…. 46

4.2.3 Transformer-based Decoder 46

4.2.4 Deep Supervision 49

4.2.5 Loss Functions 49

4.3 Data Acquisition and Pre-processing 50

4.4 Results and Discussion 52

4.4.1 Implementation Details 52

4.4.2 Ablation Studies 54

4.4.3 Artificial Intelligence-Based Segmentation Results 55

4.5 Chapter Summary 63

5 Artificial Intelligence in Healthcare: Ischemic Stroke Segmentation 64

5.1 Introduction 65

5.2 ResUNet with a Dilation Convolution 66

5.3 Data Acquisition and Pre-processing 66

5.4 Results and Discussion 67

5.5 Chapter Summary 69

6 Conclusions and Future Research 70

6.1 Conclusions 70

6.1.1 *DeepChestNet*: Artificial Intelligence Approach for COVID-19 Detection on Computed Tomography Images 70

6.1.2 *U-TranSvision*: Transformer-based Deep Supervision Approach for COVID-19 Lesion Segmentation on Computed Tomography Images 71

6.1.3 Artificial Intelligence in Healthcare: Ischemic Stroke Segmentation 72

6.2 Future Research 73

References 75

Appendices 87

Appendix A Publications from the Thesis 88

Curriculum Vitae 89

List of Figures

Figure 3.1 DeepChestNet for COVID-19 lesion detection, including the segmentation tasks using chest CT images. 26

Figure 3.2 The segmented image on *DeepChestNet App* after the detection operation. 30

Figure 3.3 The qualitative results on axial slices for lung segmentation: (a) Original images, (b) DeepLabV3+ (VGG-19), (c) DeepLabV3+ (VGG-16), (d) U-Net (VGG-19), (e) U-Net (VGG-16), (f) DeepLabV3+ (ResNet-50), (g) DenseU-Net (h) U-Net, (i) ResUNet, (j) Ground Truth. 33

Figure 3.4 The qualitative results on axial slices for pulmonary lobe segmentation: (a) Original images, (b) DenseU-Net, (c) DeepLabV3+ (VGG-19), (d) DeepLabV3+ (VGG-16), (e) U-Net (VGG-19), (f) DeepLabV3+ (ResNet-50), (g) U-Net (VGG-16), (h) U-Net, (i) ResUNet, (j) Ground Truth. The colors of the pulmonary lobes are shown for the upper right ,middle right ,lower right ,upper left and lower left 34

Figure 3.5 The qualitative results on axial slices for the COVID-19 lesion segmentation: (a) Original images, (b) DeepLabV3+ (VGG-16), (c) DeepLabV3+ (VGG-19), (d) U-Net (VGG-19), (e) DenseU-Net, (f) U-Net (VGG-16), (g) DeepLabV3+ (ResNet-50), (h) U-Net, (i) ResUNet, (j) Ground Truth..... 36

Figure 4.1 Two examples of COVID-19 positive CT images from our dataset. 42

Figure 4.2 The overall proposed *U-TranSvision* architecture..... 43

Figure 4.3 The detailed view of the transformer layer and its components. 47

Figure 4.4 The detailed statistics about our dataset. 51

Figure 4.5 The pipeline of the sequential pre-processing steps on the dataset. 52

Figure 4.6 The architectures are listed on the horizontal axis, while the metrics are presented on the vertical axes including (a) IoU, (b) Dice, (c) Sensitivity, (d) Specificity, and (e) Precision..... 57

Figure 4.7 The qualitative results of COVID-19 lesion segmentation on axial CT images overlapped corresponding probability maps, including (a) the

original image, (b) U-Net, (c) ResU-Net++, (d) ResUNet, (e) DeepLabV3+ (VGG-16), (f) DeepLabV3+ (VGG-19), (g) DeepLabV3+ (ResNet-50), (h) *U-TranSvision*, and (i) ground truth. 58

List of Tables

Table 3.1 Public datasets for the COVID-19 lesion segmentation or classification. 29
Table 3.2 The performance comparison for lung segmentation. 32
Table 3.3 The experimental results with respect to IoU and DSC scores in pulmonary lobe segmentation. 34
Table 3.4 The performance comparison for the COVID-19 lesion segmentation in terms of IoU and DSC scores. 36
Table 3.5 The quantitative comparison between *DeepChestNet* and state-of-the-art approaches for lung segmentation. 37
Table 3.6 The quantitative comparison for pulmonary lobe segmentation between *DeepChestNet* and state-of-the-art approaches. 37
Table 3.7 The quantitative comparison between *DeepChestNet* and state-of-the-art for the COVID-19 lesion segmentation. 38
Table 4.1 The layers of the proposed architecture and their comprehensive details. 45
Table 4.2 The performance comparison for the ablation study on the test set of our dataset (BL: baseline, OT: one transformer, TT: two transformers, DS: deep supervision) (the results are given as mean $\pm$ standard deviation). 54
Table 4.3 The comparison of COVID-19 lesion segmentation performance on the test set in our dataset (the results are given as mean $\pm$ standard deviation). 56
Table 4.4 The performance comparison with further significant articles in the field (the results are given as mean $\pm$ standard deviation). 59
Table 4.5 The performance comparison on COVID-19-CT-Seg (the results are given as mean $\pm$ standard deviation). 60
Table 4.6 The performance comparison on MosMedData (the results are given as mean $\pm$ standard deviation). 61
Table 4.7 The performance comparison on MedSeg (the results are given as mean $\pm$ standard deviation). 62
Table 5.1 The number and percent of the images in the train, validation, and test sets. 67
Table 5.2 The empirical results of the ischemic stroke disease segmentation..... 68

Table 5.3 The comparison of our approach with the state-of-the-art architectures for ischemic stroke disease segmentation. 68

List of Abbreviations

AI	Artificial Intelligence
API	Application Programming Interface
BL	Baseline
CNN	Convolutional Neural Network
CAD	Computer Aided Diagnosis
COPLE-Net	COVID-19 Pneumonia Lesion Segmentation Network
COVID-19	The Novel Coronavirus Disease
CSNet	Classifier-segmenter Network
CT	Computed Tomography
DICOM	Digital Imaging and Communications in Medicine
DL	Deep Learning
DMDF-Net	Dual Multi-scale Dilated Fusion Network
DRANet	Deep Residual Attention Network
DS	Deep Supervision
DSC	Dice Similarity Coefficient
FCN	Fully-convolutional Network
FLOPs	Floating Point Operations per Second
FN	False Negative
FP	False Positive
GAN	Generative Adversarial Network
IoU	Intersection Over Union
JCS	Joint Classification and Segmentation
LN	Layer Normalization

LwMLA-NET	Lightweight Multi-level Attention-based NETwork
M^2UNet	Multitask Multi-instance Deep Network
ML	Machine Learning
MLP	Multi-layer Perception
MRI	Magnetic Resonance Imaging
MSA	Multi-head Self-attention
OT	One Transformer
PDV-Net	Progressive Dense V-Network
PSPF	Partitioning-stacking Prediction Fusion
ROI	Regions of Interest
RT-PCR	Reverse Transcription-polymerase Chain Reaction
ReLu	Rectifier Linear Unit
ResNet	Deep Residual Network
SA	Self-attention
SSA-Net	Spatial Self-attention Network
SSInfNet	Self-supervised COVID-19 Lung Infection Segmentation Network
TN	True Negative
TP	True Positive
TT	Two Transformers

List of Symbols

$\mathbf{A}$	The predicted image
$\mathbf{B}$	The ground truth
$\mathcal{N} \times \mathcal{N}$	The kernel filter of a matrix
K	The number of sets in K-fold cross-validation
$\mathbf{x}$	The hidden features
N	The number of patches
$n = 1, .., N$	The patch index
x_n	The vectorized patch at the n-th index
$\mathscr{H}$	The height of $\mathbf{x}$
$\mathcal{W}$	The width of $\mathbf{x}$
$\mathcal{C}$	The number of channels in $\mathbf{x}$
D	The size of latent vector embedding space
$P \times P$	The size of each patch
$\mathbf{Q}$	The queries
$\mathbf{K}$	The keys
$\mathbf{V}$	The values
$\mathbf{Z}_0$	The sequences of tokens in the 0-th layer
$\mathbf{E}$	The patch embedding projection
$\mathbf{E}_{pos}$	The position embedding
L	The number of the transformer layer
$\ell = 1, .., L$	The layer index
$\mathbf{Z}_\ell$	The output of the ℓ-th layer
$\mathbf{Z}_{\ell-1}$	The output of the $\ell - 1$-th layer

$\hat{\mathbf{Z}}_{\ell}$	The hidden features in the transformer
$\hat{\mathbf{Z}}_{\ell-1}$	The input sequence at the $\ell - 1$-th layer
$\mathcal{H}$	The number of heads
H	The height of a matrix
W	The width of a matrix
$i = 1, .., H$	The height index of a matrix
$j = 1, .., W$	The width index of a matrix
$k = 1, 2, 3, 4$	The index of deep supervision
$\mathcal{L}_{hybrid}$	The hybrid loss function
$\mathcal{L}_{focal}$	The focal loss function
$\mathcal{L}_{dice}$	The dice loss function
$\widehat{\mathbf{S}}_{\mathrm{k}}$	The output probability map
$\mathbf{S}$	The ground truth for COVID-19 regions
$\mathbf{I}$	The input chest CT image
α	The weight adjust parameter for $\mathcal{L}_{focal}$
γ	The modulating factor for $\mathcal{L}_{focal}$
ε	The smoothing parameter
E	The number of epochs
$\widehat{\mathbf{S}}_{\mathbf{1}}$	The probability maps for first deep supervision
$\widehat{\mathbf{S}}_{\mathbf{2}}$	The probability maps for second deep supervision
$\widehat{\mathbf{S}}_{\mathbf{3}}$	The probability maps for third deep supervision
$\widehat{\mathbf{S}}_{\mathbf{4}}$	The probability maps for fourth deep supervision
$\mathcal{L}_1$	The loss for first deep supervision
$\mathcal{L}_2$	The loss for second deep supervision
$\mathcal{L}_3$	The loss for third deep supervision

$\mathcal{L}_4$	The loss for fourth deep supervision
$\mathcal{L}_T$	The total loss
$\mathbf{W}$	The weights of *U-TranSvision*

Chapter 1

Introduction

This introductory chapter describes segmentation problems for medical imaging and the motivation of the thesis in Section 1.1. Then, the contributions of this thesis are specified in Section 1.2. Next, similar studies associated with segmentation problems are reported to present a literature survey in Section 1.3. Finally, Section 1.4 states the outline of the thesis.

1.1 Segmentation Problems and Motivation

Artificial Intelligence (AI) has evolved significantly over the past 70 years with the development of hardware technology and theoretical knowledge. Although the first applications of AI were generally used for military purposes such as war games, predictive modeling, automatic target identification systems, and war simulations, its applications in civilian life have increased over time. Especially in the last decade, AI-based applications in healthcare, finance, logistics, education, law, and agriculture have become an indispensable daily routine of people, making their work easier. Among these applications, the development and use of AI in healthcare make it easier for healthcare professionals to diagnose diseases faster. In addition, AI in healthcare can prevent possible errors due to the extreme working hours of healthcare professionals. The pandemics and epidemics of recent years demonstrate that AI-based healthcare applications, such as lung, brain, cardiovascular, and liver disease diagnosis based on images or numerical data, are critical to public health. Among these applications, the thesis focuses on the diagnosis of only two diseases associated with the human lung and brain, namely the novel coronavirus disease (COVID-19) and ischemic stroke, due to their negative effect on public health.

COVID-19 has spread rapidly worldwide after first appearing in late 2019, causing millions of confirmed cases and deaths. While the introduction of COVID-19 vaccines

has reduced mortality rates, there are concerns that the mutation of viruses may reduce the effectiveness of existing vaccines. In addition, the spread of the virus continues due to anti-vaccination, vaccination hesitancy, and vaccine inequality. Therefore, early diagnosis, especially in the elderly, is of great importance in terms of their recovery, appropriate treatment, and reducing the negative impact of the disease on society. Although clinical detection of COVID-19 is primarily by reverse transcription-polymerase chain reaction (RT-PCR), it has several limitations. The response time of RT-PCR tests can be slow, typically taking 1-3 hours to detect COVID-19, and have poor sensitivity performance, resulting in a high rate of false negative results [1-3]. Several approaches, such as biosensors, medical imaging, and serology-based tests, are available to alleviate these limitations. Among these approaches, medical imaging techniques, such as magnetic resonance imaging (MRI), X-ray, and computed tomography (CT), are used to complement PCR testing since COVID-19 lesion regions can be distinguished from lung tissue on the medical image [4]. X-ray imaging produces 2D images, while MRI and CT provide much more detail by creating a computer-generated 360-degree view of the organs in the body. CT imaging is the first choice in diagnosing COVID-19 over MRI due to its lower cost and time [5, 6]. In addition, MRI is generally beneficial for particular conditions, including uterine, liver, and prostate diseases, that are challenging to diagnose with CT imaging [7]. Furthermore, it is reported that CT has a sensitivity of 98 % for detecting COVID-19, while the RT-PCR tests have a sensitivity of 71 % [8]. CT can confirm a COVID-19 lesion even when RT-PCR results are negative or weakly positive [9].

Brain disorders affect the daily routine of the individual and appear as severe physical symptoms, including motor function impairments, memory loss, tremors, seizures, vision problems, speech difficulties, and changes in behavior. Brain disorders are caused by a wide range of diseases that gradually disable brain abilities, such as Alzheimer, Parkinson, dementia, brain cancer, stroke, transient ischemic attack, and epilepsy. Among these diseases, stroke, a major abnormality of the brain, is the leading cause of severe long-term disability worldwide. The World Health Organization reports that stroke is responsible for about 11 % of human diseases and is the second leading cause of death worldwide [10]. Typically, there are two types of stroke: hemorrhagic and ischemic stroke. A hemorrhagic stroke is caused by a ruptured cerebral blood vessel that bleeds into or around the brain, while an ischemic stroke

occurs predominantly due to a lack of blood flow in parts of the brain [11]. Of these two types, ischemic stroke accounts for about 85 % of all strokes, so diagnosing ischemic stroke is vital for the recovery of patients. Several approaches are available to diagnose ischemic stroke, including a cerebral angiogram, echocardiogram, carotid ultrasound, and medical imaging [12]. Among these approaches, medical imaging methods have become functional since they visualize the brain, making it easier to diagnose the disease. In particular, CT is widely used in medical centers as it greatly improves the imaging resolution and diagnosis speed of ischemic stroke disease.

However, the quality of the diagnosis also depends on the expertise of the specialists, causing an increased likelihood of misdiagnosis when manually detecting COVID-19 and ischemic stroke disease on CT images due to factors such as lack of awareness or overwork. Besides detecting the presence of COVID-19 and ischemic stroke disease, the specific affected region on a CT image is crucial in the diagnostic process. Here, detecting regions affected by the diseases or regions of interest (ROI) on CT images is called a segmentation task. On the other hand, manual detection of a small lesion on a CT image can be challenging and time consuming for specialists, potentially resulting in a false diagnosis of COVID-19. Therefore, assistive tools that automatically detect lesions and ischemic stroke on the CT image need to be developed to provide useful diagnostic information. In this regard, AI methods, including machine learning and deep learning, were employed to classify and segment the lesion and ischemic stroke [13, 14]. Machine learning algorithms use statistical models to find patterns and relationships using manually created data features, typically requiring a small amount of structured data for training [15, 16]. On the other hand, deep learning approaches take advantage of multi-layer artificial neural networks to capture complex patterns and relationships in large-scale unstructured datasets [17]. Moreover, various deep learning applications for image segmentation have demonstrated promising results, especially in medical imaging [18, 19]. Deep learning approaches incorporate neural network architectures, such as deep belief nets [20], convolutional neural networks (CNNs) [21-25], transformers [26], and autoencoders [27]. Among these architectures, CNNs and transformers have shown remarkable performance in capturing image features [28].

1.2 Contributions

The main contributions of this thesis are briefly summarized below:

(1) We proposed a novel segmentation framework called DeepChestNet, which combines lung, pulmonary lobe, and COVID-19 lesion segmentation. The proposed framework includes a sequential structure for lung and pulmonary lobe segmentation, leading to improved performance for pulmonary lobe segmentation. In parallel with the sequential structure, DeepChestNet also performs COVID-19 lesion segmentation. The sequential and parallel structure of DeepChestNet helps minimize predictive errors in the COVID-19 lesion segmentation task, leading to accurate diagnoses for findings, such as bilateral, multifocal, and multilobar COVID-19. Moreover, unlike existing datasets, comprehensive datasets were collected, addressing three different segmentations (lung, pulmonary lobe, COVID-19 lesion) with pixel-by-pixel labeling. These presented datasets serve as a comprehensive benchmark for research studies on similar tasks. Finally, lung, pulmonary lobe, and COVID-19 lesion segmentation were integrated into our user-friendly desktop application, *DeepChestNet App*, to support healthcare professionals.

(2) *U-TranSvision* architecture, in which the transformer layer efficiently captures contextual information of images, is proposed to achieve better performance for COVID-19 lesion segmentation and provide faster convergence under deep supervision. Moreover, we extended our dataset in the first contribution to address the lack of publicly available massive datasets on COVID-19 lesion segmentation, which is data-hungry for better generalization ability. In addition, a GAN-based augmentation approach was used for better generalization ability, and two loss functions were combined to minimize the class imbalance problem, causing poor performance. Comprehensive experiments were conducted on our collected dataset and three publicly available datasets to confirm the efficiency of our approach. Furthermore, ablation studies on our dataset have shown that transformers and deep supervision significantly contribute to the segmentation results. Comparative analysis has revealed that the proposed *U-TranSvision* architecture outperforms the state-of-the-art approaches.

(3) Ischemic stroke segmentation is also performed on a public dataset. ResUNet, a CNN-based architecture, was modified with a dilation convolution layer, which allows extracting large contextual features from an image or a feature map to further improve the performance of ischemic stroke segmentation. In addition, various CNN-based architectures were benchmarked to show the effectiveness of ResUNet with a dilation convolution. On the other hand, we combined traditional and unconventional approaches, including a GAN-based method, varying brightness, applying Gaussian and blur filters, and adding salt & pepper and Gauss noise. Finally, ResUNet with a dilation convolution can help similar studies for ischemic stroke disease segmentation, which can reduce the workload and diagnosis time in its clinical use.

1.3 Literature Survey

This section first presents a review of several state-of-the-art approaches for the lung, pulmonary lobe, and COVID-19 lesion segmentation to compare our DeepChestNet framework in Subsection 1.3.1. This is followed by a literature survey of various COVID-19 lesion segmentation methods for our dataset and three publicly available datasets in Subsection 1.3.2. Finally, Subsection 1.3.3 reviews similar studies on ischemic stroke segmentation, an application of AI in healthcare.

1.3.1 *DeepChestNet*: Artificial Intelligence Approach for COVID-19 Detection on Computed Tomography Images

Lung segmentation, identifying and extracting lung regions from medical images, is crucial in various clinical applications, such as COVID-19 detection and computer-aided lung cancer diagnosis. In the following, it is reported several state-of-the-art architectures for lung segmentation from CT images with pulmonary disease. Ardakani et al. [29] utilized CNN architectures such as AlexNet, VGG-16, VGG-19, SqueezeNet, GoogleNet, MobileNet-V2, ResNet-18, ResNet-50, ResNet-101 and Xception to diagnose COVID-19 in CT images while Mobiny et al. [30] proposed detail-oriented capsule networks. Wang et al. [31] proposed a modified inception architecture with transfer learning to improve model performance for COVID-19

diagnosis. Pang et al. [32] reported lung segmentation on high-resolution CT images for interstitial pulmonary disease using the texture feature obtained with GLCM and the deep features extracted via U-Net. In order to improve the robustness of U-Net architecture, Vidal et al. [33] employed transfer learning for lung segmentation on COVID-19 images. Mansoor et al. [34] proposed a framework that employs the fuzzy connectedness algorithm to extract initial lung parenchyma and texture-based features for different abnormality pattern detection. Gill et al. [35] proposed a fusion approach that performs volumetric segmentation using lung CT images via region-growing and model-based algorithms. The approach obtained the differences and combinations between the masks from the two algorithms. The difference between the masks had been utilized to identify the lung region. Finally, the approach integrated the combination and difference. Zhao et al. [36] proposed SP-V-Net, which utilized prior shape information of pathological lung images to use radiomic features in segmentation. Gerard et al. [37] employed a polymorphic training strategy using Seg3DNet architecture on CT images to tackle abnormalities in a lung with a different disease. However, some constraints and limitations were also associated with the studies mentioned above. The models in [32, 34, 36, 37] utilized high-resolution images as an input which increased the computational cost. The approaches in [36, 37] had a large model complexity, and the performance of [34] depended on the severity of the lung diseases. The approach in [35] was based on integrating the outputs of two segmentation algorithms, which can lead to an error if one of the algorithms fails to produce correct results. In addition, running lung segmentation and COVID-19 detection simultaneously increased computational costs, making them impractical for a real-time computer aided diagnosis (CAD) system.

In addition to segmentation of the COVID-19 lesion and lung, the pulmonary lobe segmentation helps diagnose COVID-19 findings such as bilateral, multifocal, and multilobar lesion [38]. The combination of COVID-19 lesion and pulmonary lobe segmentation yields the number of infected regions in each lobe, leading to more findings for interpreting COVID-19 progression. He et al. [39] proposed a framework including a multitask multi-instance deep network (M^2UNet) for COVID-19 classification and pulmonary lobe segmentation tasks. Ferreira et al. [40] presented an FRV-Net for pulmonary lobe segmentation using lung CT images. Tang et al. [41] utilize V-Net architecture to perform pulmonary lobe

segmentation on CT images, applied to threshold-based lung extraction algorithms. However, one drawback of the threshold-based algorithm is the difficulty of obtaining accurate results in segmentation tasks due to the lack of significant differences or a considerable overlap of the gray-scale values [42]. Therefore, these algorithms are prone to errors in extracting lung regions. Imran et al. [43] introduced a progressive dense V-Network (PDV-Net) to increase the speed of the architecture in pulmonary lobe segmentation. Pang et al. [44] employed four deep learning models with nnU-Net for the pulmonary lobe segmentation task on 3D CT images. Park et al. [45] employed pre-processing and post-processing approaches to improve the performance of 3D U-Net architecture. Xie et al. [46] proposed the RTSU-Net approach that contains cascaded relational U-Nets with non-local modules to capture local details and global context for pulmonary lobe segmentation. The feature maps of the architecture [43] had high tensor dimensions, and the architecture employed U-Net based approaches with high training parameters [44, 46]. Hence, they are unsuitable for use in a CAD system with low-end hardware due to high computational cost, which leads to high response time.

Besides lung and pulmonary lobe segmentation for COVID-19 detection, various approaches have been proposed to segment the COVID-19 lesion regions. Ma et al. [47] utilized 3D U-Net for the lung and the COVID-19 lesion segmentation. Oulefki et al. [48] modified multi-level Kapur's entropy-based thresholding approach and used pre-processing steps, enhancing the local contrast of the CT images to increase the segmentation performance. Fan et al. [49] proposed Inf-Net, which provides the COVID-19 lesion segmentation using reverse attention and parallel partial decoder to extract low and high-level features. They also utilized a semi-supervised method to enhance the performance of Inf-Net architecture using labeled and unlabeled images. Zhou et al. [50] employed three U-Net architectures to segment infection regions on the images in different views, including sagittal, coronal, and transverse planes. Zhou et al. [51] added an attention mechanism into U-Net to re-weight the feature representation spatially, and channel-wise, so redundant features were avoided. Amyar et al. [52] proposed a multitask learning architecture using a joint encoder, two decoders, and a multi-layer perceptron (MLP) for reconstructing images, COVID-19 lesion segmentation, and COVID-19 classification tasks. Chen et al. [53] introduced

residual attention U-Net with aggregated residual transformation and a soft attention mechanism to better distinguish various infected regions by enhancing feature representation.

1.3.2 *U-TranSvision*: Transformer-based Deep Supervision Approach for COVID-19 Lesion Segmentation on Computed Tomography Images

In an effort to improve COVID-19 lesion segmentation and minimize the class imbalance problem in DeepChestNet, we have extended our previous chapter through the proposed *U-TranSvision* architecture. Therefore, a comprehensive review of similar studies on COVID-19 lesion segmentation is presented here, along with a brief comparison between these studies and our architecture. Zhang et al. [54] improved the Dense GAN to expand their datasets. They also incorporated a multi-layer attention mechanism into U-Net for COVID-19 lesion segmentation on pulmonary CT images. Pezzano et al. [55] developed a U-shaped architecture called CoLe-CNN+ using the Multiple Convolutional Layers module. Wu et al. [56] proposed a novel joint classification and segmentation (JCS) system, a real-time application that includes explainable AI for classification and COVID-19 lesion segmentation. Wang et al. [57] offered a COVID-19 Pneumonia Lesion Segmentation Network (COPLE-Net) that combines a self-ensembling framework to train their model and a hybrid loss function for better noise-robust segmentation. On the other hand, a drawback of these studies [49, 54-56] is that they have a relatively small dataset for COVID-19 lesion segmentation, which may result in low performance on previously unseen data. In addition, the models [49, 54, 57] have a relatively high number of learning parameters, which can cause an increase in the prediction time for test images.

Furthermore, several studies have used the publicly available COVID-19-CT-Seg dataset that includes a limited number of images to train neural network architectures. Ma et al. [58] combined a U-Net architecture and a few-shot learning approach to deal with limited CT images for COVID-19 lesion segmentation. Wang et al. [59] presented a semi-supervised few-shot iterative segmentation framework to increase the dataset size, and employed it in their proposed Spatial Self-Attention Network (SSA-Net) for

COVID-19 lesion segmentation. Peng et al. [60] introduced a novel COVID-19 segmentation network called Cov-TransNet with auxiliary and U-shaped backbone branches containing transformer layers to capture global information of images. Owais et al. [61] proposed a dual multi-scale dilated fusion network (DMDF-Net) to segment tiny COVID-19 lesions on CT images, using additional pre- and post-processing steps for better segmentation performance. Zhang et al. [62] developed a QC-HC U-Net with a quick connection that combines the residual and dense connections for COVID-19 lesion segmentation to connect the output tensors of sequential convolution layers in the architecture. However, these approaches [58-61] with high-resolution feature maps have relatively high trainable parameters, and quick connections in the study [62] increase model parameters. Therefore, the high model complexity of these architectures causes an increased computational cost for training and testing.

Moreover, various segmentation studies have been performed on the public MosMedData dataset. Shabani et al. [63] proposed an annotation-free pipeline that utilizes a 3D GAN architecture to generate synthesized healthy images, subtracted from the original CT images to produce synthesized ground truths. Next, they employed 2D U-Net architecture and contrastive learning for COVID-19 lesion segmentation with the synthesized ground truths. Mahmud et al. [64] introduced a CovTANet framework that incorporates the tri-level attention-based unit consisting of pixel, spatial, and channel-wise attention for COVID-19 lesion segmentation. Karthik et al. [65] developed a 3D CNN framework that employs a multi-scale feature pyramid, channel-wise attention, and non-local attention to extract the features of COVID-19 lesion regions on CT volumes. Karthik et al. [66] proposed a contour-enhanced attention CNN architecture that enriches the image features with a contour map provided by several sequential image processing approaches to improve COVID-19 lesion segmentation. However, the approach [63] may be error-prone as they trained their model using the synthesized ground truths. In addition, the attention mechanisms in the studies [64, 65] can increase the complexity of its model, and the approach [66] has relatively high learning parameters.

In addition, similar studies have been conducted for COVID-19 segmentation on the public MedSeg dataset. Fung et al. [67] proposed a self-supervised COVID-19 lung infection segmentation network (SSInfNet) that consolidates generative adversarial

image inpainting, lookahead optimization, and focal loss to improve model performance. Goncharov et al. [68] employed the U-Net architecture, which integrates a multitask approach that combines COVID-19 segmentation and classification with a triage approach to categorize CT images based on their severity. Roy et al. [69] offered a Lightweight Multi-Level Attention-based NETwork (LwMLA-NET) framework based on depth-wise separable convolutions to extract COVID-19 lesion regions in CT images. Abdel-Basset et al. [70] introduced a semi-supervised few-shot segmentation approach called FSS-2019-nCov to address the lack of a large-scale dataset for extracting COVID-19 lesion regions. Saeedizadeh et al. [71] proposed TV-Unet, which adds a regularization term into the loss function during the training of the U-Net architecture, utilizing an anisotropic total-variation to improve the segmentation performance. However, COVID-19 lesion segmentation approaches [67-69, 71] either utilize standard methods or do not follow any strategy for data augmentation. Hence, this may be prone to lower generalization ability of their trained model. Self-supervised learning in the study [70] to deal with the lack of annotation for CT images may not apply directly to medical segmentation, as the ground truth generated by the previously trained model could cause errors.

1.3.3 Artificial Intelligence in Healthcare: Ischemic Stroke Segmentation

This subsection presents a comprehensive literature review of the current state-of-the-art architectures for ischemic stroke segmentation, highlighting the major approaches and challenges in the field. It also includes a brief discussion associated with these approaches. In the following, several CNN approaches have been proposed for ischemic stroke segmentation. Hui et al. [72] proposed a partitioning-stacking prediction fusion (PSPF) method based on an improved Attention U-Net. Liu et al. [73] introduced a deep residual attention network (DRANet) that embeds residual blocks into U-Net architecture to segment ischemic stroke disease and white matter hyperintensity lesions in multi-modal MRIs. Kumar et al. [74] employed a classifier-segmenter network (CSNet), including a classifier followed by a segmentation network, which takes advantage of Fractal-Net and U-Net for ischemic stroke disease. Wu et al. [75] utilized an architecture based on a fully-convolutional network (FCN)

with 3D convolution for automatically segmenting subcortical structures in different neural formations within the brain. Rajinikanth et al. [76] proposed an approach using the cuckoo search algorithm, Tsallis entropy-monitored multi-level thresholding, and regularized level set technique, to improve segmentation performance. The partition process utilizes the 3D context that can effectively improve the generalization ability of their proposed method on unseen images for ischemic stroke segmentation. The approaches in [73, 74] employed relatively small-scale datasets to segment ischemic stroke disease, leading to the poor generalization ability of their models. Moreover, the model in [72] is computationally expensive, which causes the stroke lesion segmentation task to be time-consuming. On the other hand, the use of atlas-based segmentation for ischemic stroke in [75] resulted in being time-consuming and error-prone due to the image registration step.

1.4 Outline of the Thesis

This section of the thesis outlines the subsequent chapters, which are organized as follows: Chapter 2 provides a detailed description of the methods and tools employed throughout the thesis. In addition, Chapter 3 contains the proposed DeepChestNet framework, the *DeepChestNet App*, the results, and a discussion of our approach. Chapter 4 is devoted to a comprehensive examination of the proposed *U-TranSvision*, including its structures, our collected dataset, and ablation studies. Chapter 5 presents an application of AI in healthcare in the context of ischemic stroke segmentation. Finally, Chapter 6 concludes the thesis along with recommendations for future research.

Chapter 2
Methods

In this chapter, we first briefly introduce various deep learning segmentation architectures and their details in Section 2.1. An overview of data augmentation, including traditional and modern approaches, is presented in Section 2.2. Furthermore, various image processing approaches are given in Section 2.3, while Section 2.4 describes performance evaluation approaches and metrics that are commonly used for segmentation problems. Finally, the programming languages and tools used in this thesis are mentioned in Section 2.5.

2.1 Deep Learning Segmentation Architectures

AI is a mathematical approach that allows machines to perform sophisticated tasks requiring human intelligence, such as learning, problem-solving, reasoning, and perception. In fact, AI that can operate autonomously or with human intervention is nothing more than involving mathematical algorithms and models represented in code. Even though the applications of AI in our lives have grown become increasingly popular over the past few decades, its history dates back to the 1950s, when systems based on rules and logic were used to simulate human reasoning. However, with the advent of machine learning (ML) in the 1970s, AI applications shifted toward a more data-driven approach to solve various relatively uncomplicated problems, such as classification, regression, and clustering. On the other hand, AI experienced a decline in interest during the 1990s due to the reasons such as the limitations of available computing power and the inability to deal with complicated problems. Interest in AI began to grow again in the 2000s with the advent of deep learning (DL), a subfield of ML that uses multi-layer artificial neural networks to learn and extract complex representations within data. ML is beyond the scope of this thesis since its nature makes it unsuitable for sophisticated tasks, such as neural language processing, fraud detection, speech recognition, and computer vision, including image classification,

object detection, and segmentation. Therefore, DL has been considered in this thesis to deal with massive datasets and segmentation problems on medical images. In this context, CNNs, which can learn and extract features from input images, are commonly used DL approaches for segmentation problems. On the other hand, CNNs for image segmentation are more appropriate to detect COVID-19 and ischemic stroke in medical images; therefore, the following subsections focus on CNN-based architectures described in the context of segmentation.

2.1.1 U-Net

The U-Net architecture is based on an encoder and decoder structure in which image features are extracted at the encoder while the decoder processes the features for object localization [22]. The encoder contains convolution blocks to extract the feature map and uses a rectified linear unit (ReLU) as an activation function. Then, the max-pooling layer is employed to downsample the feature map. The feature maps in the encoder are fed to the decoder for further processing. The decoder provides a semantic mapping of low-level discriminative features in the encoder to the high-level pixel space.

2.1.2 ResUNet

The ResUNet architecture has formed with the combination U-Net and deep residual network (ResNet), aimed to integrate the advantages of both architectures under encoder, bridge, and decoder structures [24]. The encoder extracts the feature maps for the decoder used for pixel-wise classification, like semantic segmentation. The bridge provides the spatial connection between the encoder and decoder. Three residual units are used in each encoder and decoder. An up-sampling and corresponding feature maps are concatenated before each residual block. A 1×1 convolution layer is located after the last residual units in the decoder to project the feature maps into a semantically segmented image as an output.

2.1.3 DenseU-Net

The DenseU-Net architecture consists of U-Net and dense concatenation with down-sampling and up-sampling paths that are symmetrically located [77]. The down-sampling path involves dense and transition blocks for feature extraction to deal with the restricted deepness of U-Net. The dense block contains four convolution blocks concatenated with all output of preceding layers in a feed-forward direction to boost feature reuse. The dense blocks in the down-sampling path can extract the features at 1, 64, 128, 256, and 512 channels. The up-sampling path, which localizes the regions, recovers the input image resolution using the dense block, up-sampling, and merge layers.

2.1.4 Attention U-Net

The architecture comprises a traditional U-Net and attention mechanism to dynamically focus on relevant regions of the input image during the segmentation process [78]. The attention U-Net consists of a symmetric encoder-decoder structure with skip connections. The encoder part uses multiple convolutional blocks to reduce the spatial dimensions and increase the number of filters, capturing the hierarchical features of the input image. Each convolution layer contains two convolutions, batch normalization, and maximum pooling, with LeakyReLu as activation. Batch normalization is employed to normalize the activation of convolution layers for improving segmentation performance. In the decoder part, the feature maps are upsampled to their original dimensions through transposed convolutions. The attention mechanism is incorporated into the decoder, using self-attention blocks that adaptively assign weights to the feature maps from the encoder. These weights selectively concatenate the feature maps from the encoder and decoder for pixel-wise prediction.

2.1.5 DeepLabV3+

The DeepLabV3+ architecture is also created with the encoder and decoder structure [79]. However, in the DeeplabV3+, the encoder takes advantage of dilated convolution that allows adjusting the resolution of the feature maps by changing the distance between kernel pixels to capture multi-scale features [80]. Dilated convolution reduces

the computational complexity of a deep learning model with depth-wise separable convolution, which factorizes a standard convolution into depth-wise convolution that performs a single filter for each input channel. In the decoder structure, low-level features coming from the backbone, which can be any network for feature extraction (i.e., VGG-16), are convoluted to reduce the number of channels. The output feature map of the encoder is upsampled by four. Then, the low-level backbone feature maps and up-sampling are concatenated to produce the output mask. As described in the following section, a backbone architecture can be used in the encoder of DeepLabV3+ to extract low-level features and feed them into the feature maps within the encoder and decoder.

2.1.6 Backbone Architectures

The core components of deep neural networks are backbone architectures, which are responsible for extracting features from input data. Backbone architectures are commonly used in CNN-based architectures since the fact that they extract features effectively leads to improved segmentation performance. The following summarizes several popular backbone architectures for image segmentation problems.

2.1.6.1 VGG-16

VGG-16 architecture is structured with five blocks containing 13 convolution layers with a 3×3 filter [81]. After each convolution layer, a ReLu activation function exists to ensure non-linearity, which avoids acting like a single layer for sequential layers. The last layer in each block involves a 2×2 max-pooling that decreases the spatial resolution of feature maps [25]. The filters of each convolution in the blocks are 64, 128, 256, 512, and 512. After the blocks, three fully connected layers perform classification tasks. However, the five blocks of VGG-16 supply low-level features for semantic segmentation architectures.

2.1.6.2 VGG-19

VGG-19 architecture contains five blocks with 16 convolution layers and has a small receptive field of 3×3 [25]. Each convolution layer follows a ReLu activation

function, while a max-pooling is used in the last convolution layer of each block. After the last convolution layer, three fully connected layers follow for classification problems, but they are removed to perform the segmentation task.

2.1.6.3 ResNet-50

In the encoder of DeepLabV3+, ResNet-50 architecture can be utilized as a backbone to extract low-level features of images. However, the fact that ResNet-50 is deep compared to VGG-16 and VGG-19 causes a performance decrease called the degradation problem. The residual network, which involves the identity connection between the input and output of the residual blocks, is utilized in ResNet-50 to avoid the problem [82]. The residual blocks with three convolution layers supply more connection between the layers to improve the performance of ResNet-50. This increases the effect of the deep layers during backpropagation. ResNet-50 contains three fully connected layers later in the last convolution block for classification, but these layers were removed to feed the DeepLabV3+.

2.2 Data Augmentation

Training deep learning architectures with small-scale datasets results in a lack of generalization due to overfitting caused by model bias. Therefore, large-scale datasets are required to ensure the generalization capability of the CNN-based architectures. In this regard, data augmentation is applied to small-scale datasets, which generate modified copies and new synthetic data from existing data [83]. Data augmentation could be classified into traditional data augmentation approaches and GANs, which create virtual synthetic images. The distinction between these approaches is described in the next subsection.

2.2.1 Traditional Data Augmentation Approaches

Commonly used traditional data augmentation approaches for image datasets are basic image manipulations with kernel filters and geometric transformations, including color space transformation and noise injection.

2.2.1.1 Kernel Filters

For image manipulations, a $\mathcal{N} \times \mathcal{N}$ matrix called kernel filter is multiplied pixel-wise by sliding over the whole image, which transforms the image into a sharpener or blurrier based on the type of kernel filter. A vertical or horizontal edge filter ensures for sharpening of an image, while a Gaussian blur filter provides a blurred image.

2.2.1.2 Color space transformation

A digital image consists of a three-dimension tensor denoted as height, width, and color channels. A simple way of color space augmentation is using gray-scale images without color channels. The gray-scale images can be adjusted through basic matrix operations to increase or decrease the brightness. A scalar value can be added to each matrix value to increase the brightness and subtracted to decrease the brightness.

2.2.1.3 Noise injection

Noise can be injected into an image to create a new modified image using a random matrix that can be Gaussian distribution, salt & pepper noises, or other distributions. The method is called noise injection, which helps CNNs architecture learn more robust features from an image [83].

2.2.2 GAN-based Data Augmentation Approaches

The following briefly describes GAN-based data augmentation approaches, a deep learning method used to generate synthetic images similar to their original images. An advantage of these approaches is their ability to generate more realistic and diverse synthetic data than traditional approaches, which can be particularly beneficial in tasks where real images are scarce or costly to acquire. In addition, these approaches have the potential to reduce overfitting by providing a wider range of samples on which to train deep learning models.

2.2.2.1 Pix2Pix GAN Architecture

In this thesis, the Pix2Pix GAN architecture [84], widely used for image-to-image translation tasks, was implemented to generate synthetic chest images only from the test sets. The architecture consists of two structures a generator and a discriminator. The generator typically employs a U-Net [22] architecture to create a new synthetic image, while the discriminator takes advantage of PatchGAN [84] to classify the synthetic image as real or fake. To augment the datasets, the ground truths were employed in the input of the Pix2Pix GAN architecture, and their corresponding original images were utilized as the target for training.

2.3 Image Processing

Image processing involves analyzing, manipulating, and enhancing digital images using various mathematical operations. The primary purpose of image processing is to improve the visual quality of images and to extract useful information from them. Moreover, image processing approaches can be helpful for a wide range of applications, including visualization, recognition, retrieval, pattern recognition, sharpening & restoration, feature extraction, segmentation, and pre-processing. The following sections describe various image processing techniques used in this thesis for the purpose of segmentation and pre-processing.

2.3.1 Otsu Method

The Otsu method is a thresholding technique that can automatically segment an image into two parts called foreground and background [85]. The idea behind this method is to find an optimal threshold value that maximizes the separability between these two parts. Its algorithm calculates the variance of pixel intensities in the image for all possible threshold values, trying to find the optimum that minimizes the intra-class variance and maximizes the inter-class variance. After calculating the optimal threshold, the pixels with intensity values greater than the threshold are classified as foreground, while the remaining pixels are assigned as background. This method is commonly used in image processing applications, including object recognition and

image segmentation. Therefore, it has been used as an auxiliary method to segmentation for the pre-processing in this thesis.

2.3.2 Morphological Operations

As an image processing technique, morphological operations can modify the shape or texture of objects in an image. The operations rely on mathematical morphology, a branch of mathematical theory that analyzes and manipulates geometric structures. Morphological operations use a structuring element, a small binary image that is a pattern to be applied to the image. Due to their easy computation, the size of structuring elements is usually an odd number, like 3×3, 5×5, or 7×7. The structuring element is moved across the image, and the overlapped pixels are compared with the corresponding reference pixels within the structuring element. Next, the output of the operation at each location is decided according to the comparison result. In the following, various morphological operations, such as erosion, dilation, opening, and closing, are presented for pre-processing.

(1) The erosion operation removes the boundary of the objects in a binary image to make them smaller. It is used in image processing to separate the objects by eliminating the noise and smoothing the boundaries.
(2) The working principle of the dilation operation is to add a new pixel to the objects in a binary image in order to expand their region. It can help to fill the gaps between objects, connect broken lines and smooth the edges of things.
(3) The opening operation, a combination of erosion followed by dilation, can be useful for eliminating small objects in a binary image and smoothing the edges of larger things.
(4) The closing operation, used to fill small holes in the object within a binary image and smooth the boundary of things, combines a dilation followed by erosion.

2.4 Model Evaluation Approaches and Performance Metrics

In a machine learning task, the model created after the training should be evaluated using several approaches, such as K-fold cross-validation, the holdout method, or stratified sampling. The holdout method divides a dataset into two parts for training and testing, while stratified sampling ensures that the proportion of each sample from each class is the same in both the train and test sets. Among these approaches, K-fold cross-validation is considered in this thesis since it is commonly used for model evaluation. On the other hand, performance metrics are also an essential aspect of model evaluation for machine learning. They provide a quantitative measure of whether a model performs well for a given task. The commonly used performance metrics for segmentation tasks are introduced in the following subsections.

2.4.1 K-Fold Cross-validation

In machine learning, K-fold cross-validation, an approach to evaluating model performance, is based on randomly dividing the dataset into K sets. In this approach, one set is used as the test set, and the remaining K-1 folds are allocated for training. This process is repeated K times, and a different test set is used in each iteration. In this way, the performance of the model can be better measured by using the entire data set as both training and test sets.

2.4.2 Performance Metrics

Intersection over union (IoU), dice similarity coefficient (DSC), sensitivity, specificity, and precision are widely used in medical image segmentation tasks to evaluate the performance of CNN-based architectures [60]. IoU quantifies the level of overlap between the predicted image A and the ground truth B, while DSC computes the similarity between these two sets using the following equations:

$$\text{IoU}\ (\mathbf{A}, \mathbf{B}) = \frac{|\mathbf{A}\ \cap \mathbf{B}|}{|\mathbf{A}\ \cup \mathbf{B}|} \tag{2.1}$$

$$\mathrm{DSC}\,(\mathbf{A},\mathbf{B}) = \frac{2|\mathbf{A} \cap \mathbf{B}|}{|\mathbf{A}| + |\mathbf{B}|}$$

Sensitivity and specificity are also used to evaluate the segmentation performance for distinct positive and negative pixels computed with the predicted image and the ground truth. Precision measures the ability of CNN-based architectures to correctly identify positive pixels classified as positive. These evaluation metrics can be expressed as follows:

$$\begin{aligned} \mathrm{Sensitivity} &= \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \\ \mathrm{Specificity} &= \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \\ \mathrm{Precision} &= \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \end{aligned} \tag{2.2}$$

where TP represents true positive, the pixels correctly labeled as the ROI in both the predicted image and the ground truth. FP stands for false positive that the regions marked as the ROI in the predicted image but not present in the ground truth. TN denotes true negative, the pixels marked as the complement to the union between the two sets. Finally, FN represents false negative, the pixels present in the ground truth but not labeled as the ROI in the predicted image.

The values of the performance metrics take values between zero and one, with a greater value representing a higher similarity in their segmented region.

2.5 Programming Languages and Tools

Programming languages are tools that allow humans to give instructions to computers. Various programming languages, including Python, MATLAB, C, and C++, exist for different purposes, such as increasing productivity in software development, scientific computing, data analysis, or artificial intelligence. Among them, using a programming language for artificial intelligence has increased with theoretical knowledge and technological development. Therefore, this thesis considered Python language since it is a favorite for artificial intelligence. On the other hand, tools can be beneficial for

segmentation tasks to label images, develop an interface, and train models. For this purpose, several tools, including TensorFlow & Keras, PyQt5, MATLAB Image Labeler, and MedSeg, are presented in the following.

2.5.1 Python.

Python is an open-source programming language developed by Guido van Rossum in 1991. It is an object-oriented, interpreted, and high-level language popular due to its large libraries, ease of learning, and cross-platform support. The language can be used in various fields, such as data science, AI, machine learning, web development, embedded systems, scientific computing, and network programming. In particular, it is widely employed in data science and AI. Moreover, AI-based applications can be developed in Python through its various functional libraries, including Tensorflow and Keras, as well as PyQt5, which is also helpful in designing an interface that deploys a trained model.

TensorFlow, developed by Google in 2015, is a machine learning framework designed for AI applications. Its various capabilities include optimizing computational graphs and handling large datasets. Besides, Keras is a high-level application programming interface (API) designed to create deep learning models. This API employs TensorFlow as its default tensor backend engine. Moreover, Tensorflow and Keras provide many pre-trained models for deep learning and machine learning applications, leading to easy implementation in Python.

As previously stated, PyQt5, a Python version of the Qt library written in C++, is a graphical user interface library. PyQt5 provides various Qt tools, such as drawings, buttons, text boxes, menus, dialog boxes, graphics, animations, and videos. These tools offer easy use in various applications, including data analysis, visualization, scientific computing, and desktop application development. In addition, the cross-platform support in PyQt5 allows deep learning models to be deployed on multiple platforms.

2.5.2 MATLAB Image Labeler

MATLAB is a high-level programming language for mathematical operations such as matrix manipulation, data analysis, and scientific computing. It has many toolboxes:

signal processing, computer vision, image processing, optimization, control system design, and artificial intelligence. Among these, the computer vision toolbox is practical for AI-based solutions through its algorithms, functions, and applications, including the Image Labeler, which automatically labels objects within an image in a dataset. Image Labeler also allows manual labeling, used in image processing applications based on deep learning or machine learning to create ground truths of data for segmentation tasks.

2.5.3 MedSeg.

MedSeg is a web-based 2D and 3D segmentation tool that automatically distinguishes between different tissues or divergent structures in medical images. Its deep learning-based automatic segmentation feature provides an efficient and faster method to create a ground truth image based on disease or abnormality regions. MedSeg is employed in medical fields, such as COVID-19 diagnosis, tumor or lesion identification, brain image analysis, and cardiac image analysis.

Chapter 3

DeepChestNet: Artificial Intelligence Approach for COVID-19 Detection on Computed Tomography Images

In this chapter, a novel framework called DeepChestNet is proposed that leverages structured relationships by jointly performing segmentation and classification on the lung, pulmonary lobe, and COVID-19 lesion, leading to enhanced detection of COVID-19 with findings. In DeepChestNet, ResUNet architecture is performed for lung and pulmonary lobe segmentation tasks sequentially, and their outputs are combined with ResUNet based COVID-19 lesion segmentation results to improve the COVID-19 diagnosis performance. To train ResUNet architecture in DeepChestNet, new large-scale *DeepChestNet-Lung*, *DeepChestNet-Lobe*, and *DeepChestNet-COVID* datasets, including 9036, 9036, and 1034 axial chest CT images with pixel-level annotated labels, are collected. The performance of DeepChestNet in terms of DSC is 99.35 %, 99.73 %, and 97.89 %, while it is 98.71 %, 99.47 %, and 95.89 % in terms of IoU for the lung, pulmonary lobe, and COVID-19 lesion segmentation, respectively. The experimental investigations on *DeepChestNet-Lung*, *DeepChestNet-Lobe* and *DeepChestNet-COVID* datasets, and comparison with several state-of-the-art approaches reveal the great potential of DeepChestNet for diagnosis of COVID-19 disease.

In what follows, we begin with an introduction to our proposed framework in Section 3.1. This is followed by Section 3.2, which describes the DeepChestNet framework. Our collected datasets are also mentioned in Section 3.3, and the *DeepChestNet App* is introduced in Section 3.4. Then, we continue with the results and discussion in Section 3.5. Finally, this chapter is concluded in Section 3.6.

3.1 Introduction

COVID-19 detection can be performed through segmentation on medical imaging, identifying whether an individual is infected with COVID-19. Recent studies [50, 86] for segmenting infected regions in a medical image have worked on essential information related not only to COVID-19 detection but also to the progression of the disease. In this context, AI-based COVID-19 detection applications can provide this information, reducing the burden on healthcare professionals as well as leading to faster and more accurate diagnoses. On the other hand, detecting COVID-19 on medical images can be challenging due to restrictive reasons such as the complex structure of the infected region and its high contrast. We proposed the DeepChestNet framework, which employs the ResUNet architecture for extracting the features in a medical image, to reduce these challenges and achieve better performance by combining lung, pulmonary lobe, and COVID-19 segmentation. In addition to COVID-19 detection, it provides further COVID-19 findings such as bilateral, multifocal, or multilobar. To the best of our knowledge, although approaches [87, 88] combining lung segmentation and COVID-19 classification have been proposed previously, no study has combined lung, pulmonary lobe, and COVID-19 for segmentation. In addition to the limitations, there are a few publicly available datasets with a small number of images, so a relatively large-scale dataset was collected for lung, pulmonary lobe, and COVID-19 lesion segmentation tasks. This led to improved performance of our model in these tasks.

Our proposed framework DeepChestNet (Figure 3.1) differs substantially from previous works on COVID-19 detection with respect to lung, lobe, and lesion segmentation. DeepChestNet employs segmentation tasks both sequentially and in parallel to improve predictive performance for lung, lobe, and COVID-19 lesion. The lung and lobe segmentation tasks were performed sequentially while the COVID-19 lesion segmentation was run in parallel with lung and pulmonary lobe segmentation, and complementary results led to more finding information for COVID-19. Several public datasets [89-91] are available for the segmentation or classification tasks, but most of the datasets are offered images with case-level annotations, i.e., class labels. However, we created three datasets, *DeepChestNet-Lung*, *DeepChestNet-Lobe*, and *DeepChestNet-COVID*, with pixel-level annotations including all of the lung,

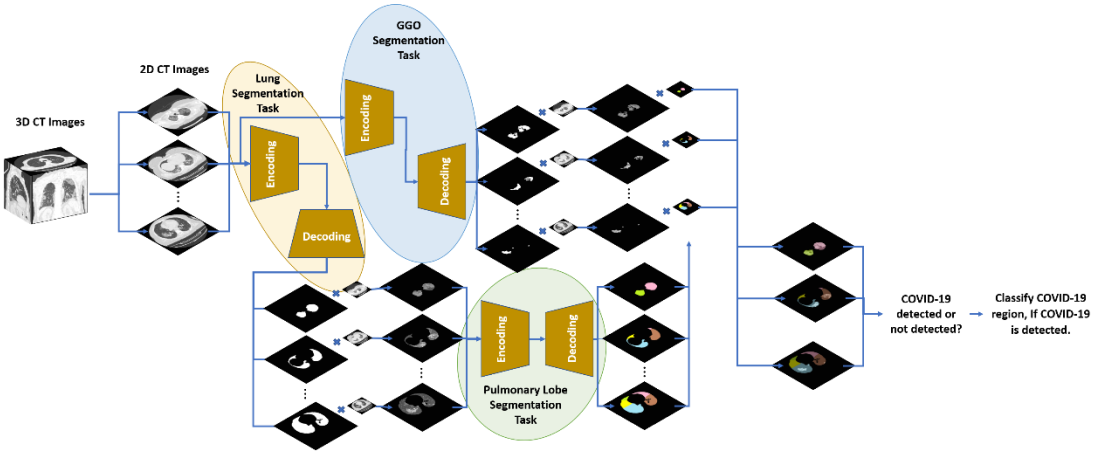


Figure 3.1: DeepChestNet for COVID-19 lesion detection, including the segmentation tasks using chest CT images.

pulmonary lobe, and COVID-19 lesion segmentation. Contrary to [39, 41], deep learning techniques have been employed before pulmonary lobe segmentation to minimize the error in lung region extraction. In addition, DeepChestNet offers more generalization ability for COVID-19 lesion segmentation as it was trained with the large-scale dataset. It was also integrated into our custom-designed desktop application, *DeepChestNet App* (Figure 3.2), which runs the lung, pulmonary lobe, and COVID-19 lesion segmentation tasks under a fast and user-friendly interface.

3.2 *DeepChestNet* Framework

A new framework, called DeepChestNet, shown in Figure 3.1, is proposed for COVID-19 detection with a pipeline containing the lung, pulmonary lobe, and COVID-19 lesion segmentation tasks. The pipeline, which leads to improving the detection performance of COVID-19, consists of sequential and parallel steps to take advantage of each task. The sequential step involves the lung and pulmonary lobe segmentation tasks, while the COVID-19 lesion segmentation task is performed in the parallel step. The output images in COVID-19 lesion and pulmonary lobe segmentation tasks are overlapped to get the number of infected regions at each pulmonary lobe. Thus, COVID-19 findings that provide information about the course of the disease are determined. Each segmentation task has been performed on DeepChestNet using CNN architectures explained in Section 2.2 to create a benchmark. The input resolutions of the CNN architectures in whole tasks have been fixed to 256×256 to decrease computation time. To train the CNN architectures, large-scale datasets, described in

Section 3.3, have been created for lung, pulmonary lobe, and COVID-19 lesion segmentation tasks. DeepChestNet has been tested on the test subsets of each dataset for lung, lobe, and COVID-19 lesion segmentation tasks and evaluated via the metrics presented in Subsection 2.4.2.

First, the lung segmentation task, belonging to the sequential step of DeepChestNet, has been performed using the CNN architecture trained on *DeepChestNet-Lung* dataset to separate the lung region from the non-lung pixel to improve the performance of the architectures in the pulmonary lobe segmentation task. The segmented lung image is used as a mask and overlapped with the corresponding original image to remove redundant pixels on the image. The CNN architectures, ResUNet, DenseU-Net, U-Net (VGG-16 and VGG-19 backbones), and DeepLabV3+, which runs ResNet-50, VGG-16, and VGG-19 as a backbone, have been trained with *DeepChestNet-Lung* dataset, including the lung CT images and its ground truth images. Among them, ResUNet outperforms the U-Net and DeepLabV3+ in terms of IoU and DSC scores, making it selected to run in *DeepChestNet App*. Thus, the output of the ResUNet is fed to the pulmonary lobe segmentation task.

A lung consists of the right upper, right lower, left upper, left middle, and left lower lobes. The pulmonary lobe segmentation task in DeepChestNet allows a pixel-wise classification of five lobes and background, leading to the segmentation of lobe regions. The number of lesions in the pulmonary lobe regions helps to identify COVID-19 findings about the disease level so that correct treatment can be determined. The pulmonary lobe segmentation task in the sequential step of DeepChestNet utilizes the segmented lung image generated by the lung segmentation task. Thus, the noises belonging to non-lung regions in the image are removed, leading to an increase in the performance of the pulmonary lobe segmentation task [41]. Besides, the segmented lung image forces CNN architecture to utilize only the lung region, so this improves the reliability and quality of the pulmonary lobe segmentation task [88]. *DeepChestNet-Lobe* dataset has been created as explained in Section 3.3, and the CNN architectures, including ResUNet, DenseU-Net, U-Net (VGG-16 and VGG-19 backbones), DeepLabV3+, which contains ResNet-50, VGG-16, and VGG-19, as a backbone have been trained. Here, ResUNet shows the best performance again

with respect to IoU and DSC scores. After the image is processed with ResUNet to segment the lung, it is reprocessed with ResUNet to extract the lobe regions.

In DeepChestNet, the segmentation of COVID-19 lesions provides a pixel-wise separation between the infected region and the background on chest CT images, resulting in the detection of COVID-19. The COVID-19 lesion segmentation task is performed in parallel with the sequential step to detect COVID-19 on CT images. To train the CNN architectures, *DeepChestNet-COVID* dataset has been created and used after the pre-processing steps explained in Section 3.3. The CNN architectures, including ResUNet, DenseU-Net, U-Net (VGG-16 and VGG-19 backbones), DeepLabV3+ (ResNet-50, VGG-16, and VGG-19 backbones), have been trained, and the highest performance was obtained with ResUNet again. The outputs of the images for COVID-19 lesion and lobe segmentation tasks have been overlapped to get the number of infected regions at each pulmonary lobe.

3.3 Data Acquisition and Pre-processing

The dataset plays a significant role in deep learning-based systems to achieve a robust model; however, a few datasets for COVID-19 are available, as listed in Table 3.1. The datasets [89-91], which can be used for COVID-19 lesion segmentation and classification, are small-scale for pixel-level annotation, resulting in a lack of generalization capability. In addition, these datasets are not suitable for lung, pulmonary lobe, and COVID-19 lesion segmentation tasks. Therefore, new datasets, called *DeepChestNet-Lung*, *DeepChestNet-Lobe*, and *DeepChestNet-COVID*, have been created to address these issues. Chest CT images for the dataset were collected by two radiologists from 231 cases, including 86 normal cases and 145 cases of COVID-19 lung involvement. CT images were captured with Siemens SOMATOM (Hannover, Germany) and Toshiba Alexion (Kyoto, Japan) machines based on the following technical specifications: axial and sagittal images, automatic tube current modulation with a voltage of 120 kVp, matrix size of 512×512, slice number of 320-420, pixel spacing of 0.74-0.82 mm, increment and thickness of 1-5 mm. The fact that some CT images do not include the lung causes performance degradation in training CNN architectures. Therefore, these images were removed from the datasets. *DeepChestNet-Lung*, *DeepChestNet-Lobe*, and *DeepChestNet-COVID* include 9,036,

Table 3.1: Public datasets for the COVID-19 lesion segmentation or classification.

Dataset	**Type**	**Images/Cases**
PLXR [89]	X-ray	98/70
CTSeg [90]	CT	110/60
COVID-CT [91]	CT	812/216
DeepChestNet-Lung (Ours)	CT	9,036/145
DeepChestNet-Lobe (Ours)	CT	9,036/145
DeepChestNet-COVID (Ours)	CT	1,034/145

9,036, and 1,034 images for lung, pulmonary lobe, and COVID-19 lesion segmentation tasks, respectively. The images have been annotated at pixel-level based on the lung, pulmonary lobe, and COVID-19 lesion segmentation tasks through MATLAB Image Labeler App. The images in *DeepChestNet-Lobe* have been labeled according to five pulmonary lobe regions enumerated from '1' to '5'. However, the images of *DeepChestNet-Lung* and *DeepChestNet-COVID* have been labeled with the '1' to indicate the foreground. The '0' label corresponds to the background for all annotated images in three datasets. After the first training for the COVID-19 lesion task, the performance of the CNN has shown that tiny lesion regions cause poor performance. Thus, *DeepChestNet-COVID* has been rearranged by removing the images, the ratio of lesion region to background ratio of which is smaller than 5 %. Several pre-processing methods, such as data normalization and augmentation processes, have been applied to optimize the datasets before training the deep learning architectures. The images have been normalized through min-max normalization [92]. Besides, each dataset has been divided five-fold via the five-fold cross-validation to overcome the effect of data leakage that causes poor predictive performance in the CNN architectures. Each time four folds were combined into a training subset, and the remaining one was employed for testing. The training subset was split 80 by 20 into training and validation subsets. The experimental results on the testing subset were averaged after each training. Next, the data augmentation process, offered in Section 2.2, has been applied to the training subsets of *DeepChestNet-Lung* and *DeepChestNet-COVID* to increase the generalization capability of the deep learning architectures. The datasets have been augmented via image processing methods, such as increasing/decreasing intensity, adding Gauss and salt & pepper noises, and

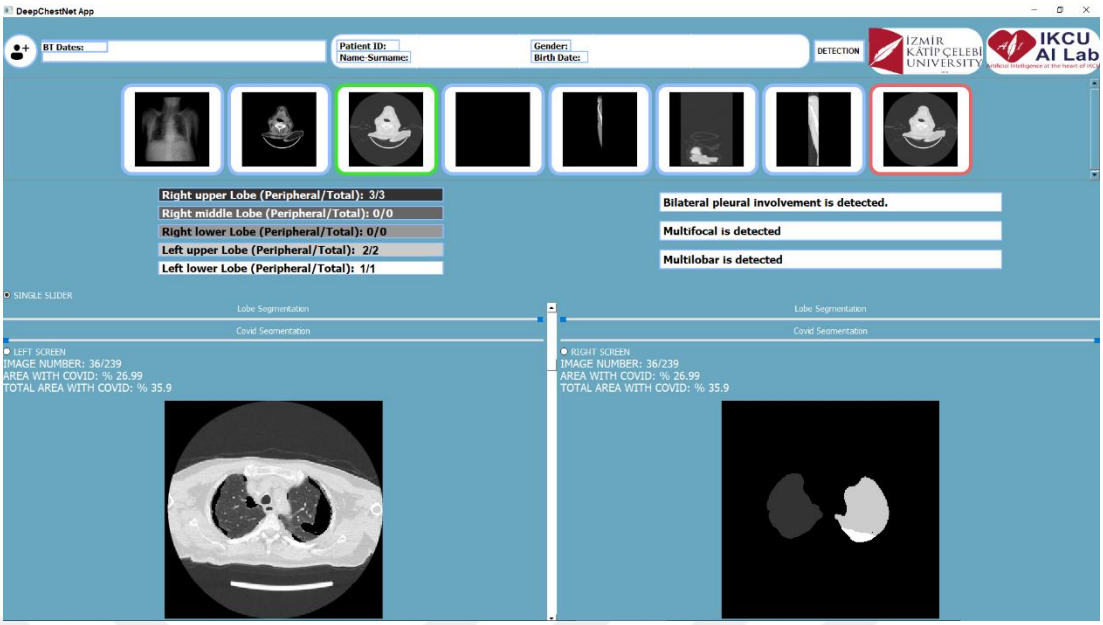


Figure 3.2: The segmented image on *DeepChestNet App* after the detection operation.

applying blur and Gaussian filters on the images. The settings used for the image processing techniques are as follows: the intensity values of the images are increased or decreased by 40, and the kernel size of the blur and Gaussian filters is 7×7. After applying the data augmentation process, *DeepChestNet-Lung* and *DeepChestNet-COVID* were enlarged to 63,252 and 7,238, respectively.

3.4 DeepChestNet App

A desktop application called *DeepChestNet App*, shown in Figure 3.2, was developed by integrating DeepChestNet to support healthcare professionals in treating COVID-19. *DeepChestNet App* offers a login screen with a specific password for professionals to prevent the access of unauthorized persons. CT images of the patient are uploaded by clicking the patient add button on the left upper of the main screen. The images can be displayed on the right or left side of the screen after selecting the category (axial, coronal, or sagittal CT images) from the top row, and then the volume of images can be examined using the index slider by clicking the radio buttons. As COVID-19 is detected on the axial CT images, the detection process can be started after selecting them. The predicted masks can be shown via a new red button, which appears after the process, and the radio buttons to examine the infected region on the corresponding images. The lung and pulmonary lobe masks are overlapped with varying opacity

values for better analysis. Healthcare professionals can adjust the opacity values for predicted pulmonary lobe and COVID-19 lesion masks on original images via the sliders on the upper sides to analyze the COVID-19 lesions effectively. Moreover, it is possible to swap the screen from left to right, which allows the user to view and compare the original or mask images on both sides. Besides, the number of lesions in each pulmonary lobe is calculated and presented on *DeepChestNet App* to determine the findings such as bilateral, multifocal, or multilobar. Thus, healthcare professionals can specify the category of COVID-19, such as typical, atypical, indeterminate, and negative for the patient based on the findings.

3.5 Results and Discussion

So far, this chapter has presented an introduction, our DeepChestNet framework, the collected datasets, and the developed desktop application. Here, the experimental settings of the study are given, followed by the results of the CNN-based architectures on our dataset. Additionally, a comprehensive discussion of comparable studies is provided in the following subsections.

3.5.1 Experimental Settings

The experiments were performed on Keras 2.8.0 [93], including TensorFlow 2.8.0 [94] as the backend, in Python to train the CNN architectures for the lung, pulmonary lobe, and COVID-19 lesion segmentation tasks. The optimizer, kernel initializer, and epoch were set as Adam, with 0.0005 learning rate, he-normal, and 300, respectively. The batch size used for DeepLabV3+ (all backbones), DenseU-Net, ResUNet, and U-Net (with and without the backbones) is 48, 4, 32, and 64, respectively, due to a lack of memory resources. The kernel parameters of the convolution layers were randomly initialized with a fixed seed number to ensure the repeatability of the experiments. The dice and focal losses used for segmentation were added as a loss function in training to increase the performance of the CNN architectures [84]. The experiments are performed on a workstation with Windows 10 operating system and NVIDIA RTX A5000 24 GB graphics processing unit through Python 3.8 environment.

Table 3.2: The performance comparison for lung segmentation.

Method	**IoU (%)**	**DSC (%)**
DeepLabV3+ (VGG-19)	94.54 ± 0.03	97.15 ±0.10
DeepLabV3+ (VGG-16)	94.89 ± 0.16	97.38 ± 0.24
U-Net (VGG-19)	95.05 ± 0.09	97.44 ± 0.14
U-Net (VGG-16)	95.18 ± 0.25	97.58 ± 0.12
DeepLabV3+ (ResNet-50)	95.48 ± 0.09	97.86 ± 0.15
DenseU-Net	97.51 ± 0.06	98.73 ± 0.03
U-Net	97.33 ± 0.30	98.65 ± 0.16
ResUNet	**98.71 ± 0.35**	**99.35 ± 0.18**

3.5.2 Artificial Intelligence-Based Segmentation and Detection

The CNN architectures U-Net, DenseU-Net, ResUNet, and DeepLabV3+ with their extensions using ResNet-50, VGG-16, and VGG-19 as a backbone have been trained by respective datasets and tested for the lung, pulmonary lobe, and COVID-19 lesion segmentation tasks.

Table 3.2 lists the quantitative comparison of the CNN architectures for lung segmentation in terms of DSC and IoU scores. The experimental results indicate that ResUNet architecture yields the highest performance in terms of evaluation metrics, with an IoU score of 98.71 ± 0.35 % and a DSC score of 99.35 ± 0.18 %. The second highest results have been achieved by U-Net architecture. Based on the experiments, ResUNet architecture has outperformed the other compared architectures. ResUNet architecture is based on U-Net and ResNet. The empirical results indicate that U-Net with ResNet can enhance the predictive performance for segmentation tasks by robust feature extraction. The third-highest predictive performance in terms of the IoU score has been achieved by DenseU-Net architecture, and the third-highest predictive performance in terms of DSC score has been achieved by DeepLabV3+ (ResNet-50). For the experimental results, DeepLabV3+ with ResNet-50 architecture outperforms other architectures where DeepLabV3+ has been employed. In addition, DeepLabV3+ showed better performance with VGG-16 compared to VGG-19 due to the degradation

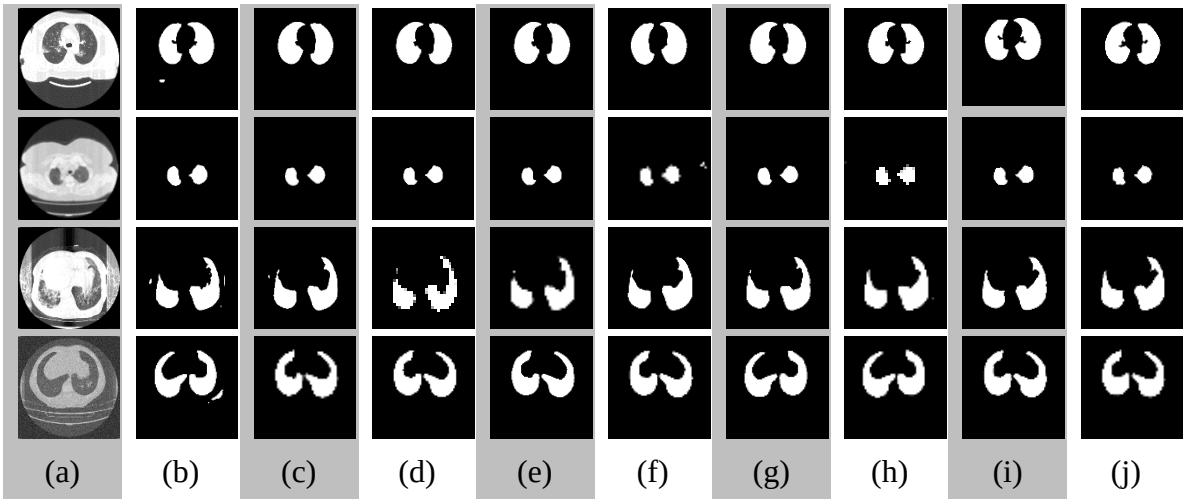


Figure 3.3: The qualitative results on axial slices for lung segmentation: (a) Original images, (b) DeepLabV3+ (VGG-19), (c) DeepLabV3+ (VGG-16), (d) U-Net (VGG-19), (e) U-Net (VGG-16), (f) DeepLabV3+ (ResNet-50), (g) DenseU-Net (h) U-Net, **(i) ResUNet**, (j) Ground Truth.

problem in the deeper network. The qualitative results for U-Net, ResUNet, DenseU-Net, and DeepLabV3+ (with ResNet50, VGG-16, and VGG-19) on axial CT images are shown in Figure 3.3. Although all approaches usually do not produce false-negative pixels for a lung region, ResUNet mostly generates a smoother lung boundary; therefore, it was integrated into the *DeepChestNet App*.

The pulmonary lobe segmentation performance in terms of IoU and DSC scores has been reported in Table 3.3. For the experimental results, ResUNet and U-Net yield relatively high performance for pulmonary lobe segmentation. The U-Net with VGG-16 yields higher performance compared to the U-Net with VGG-19. The DeepLabV3+ has achieved the fourth-highest predictive performance for pulmonary lobe segmentation with ResNet-50. The Dense U-Net underperformed in pulmonary lobe segmentation. The DeepLabV3+ with ResNet-50 outperformed other extensions of DeepLabV3+ through the residual network. Thus, ResUNet was used in DeepChestNet due to its superior performance on pulmonary lobe segmentation. DenseU-Net yields promising results for lung segmentation, while it underperforms for pulmonary lobe segmentation. Besides, the qualitative results of U-Net (without backbone, with VGG-16 and VGG-19), ResUNet, DeepLabV3+ (with ResNet-50, VGG-16, and VGG-19), and DenseU-Net for axial CT images are given in Figure 3.4 where separated lobes with ResUNet are more similar to ground truth results.

Table 3.3: The experimental results with respect to IoU and DSC scores in pulmonary lobe segmentation.

Method	**IoU (%)**	**DSC (%)**
DenseU-Net	94.46 ± 0.20	97.13 ± 0.08
DeepLabV3+ (VGG-19)	97.84 ± 0.43	98.99 ± 0.17
DeepLabV3+ (VGG-16)	98.12 ± 0.35	99.08 ± 0.12
U-Net (VGG-19)	98.75 ± 0.51	99.38 ± 0.23
DeepLabV3+ (ResNet-50)	98.96 ± 0.45	99.46 ± 0.19
U-Net (VGG-16)	99.19 ± 0.52	99.57 ± 0.25
U-Net	99.24 ± 0.57	99.62 ± 0.29
ResUNet	**99.47 ± 0.67**	**99.73 ± 0.34**

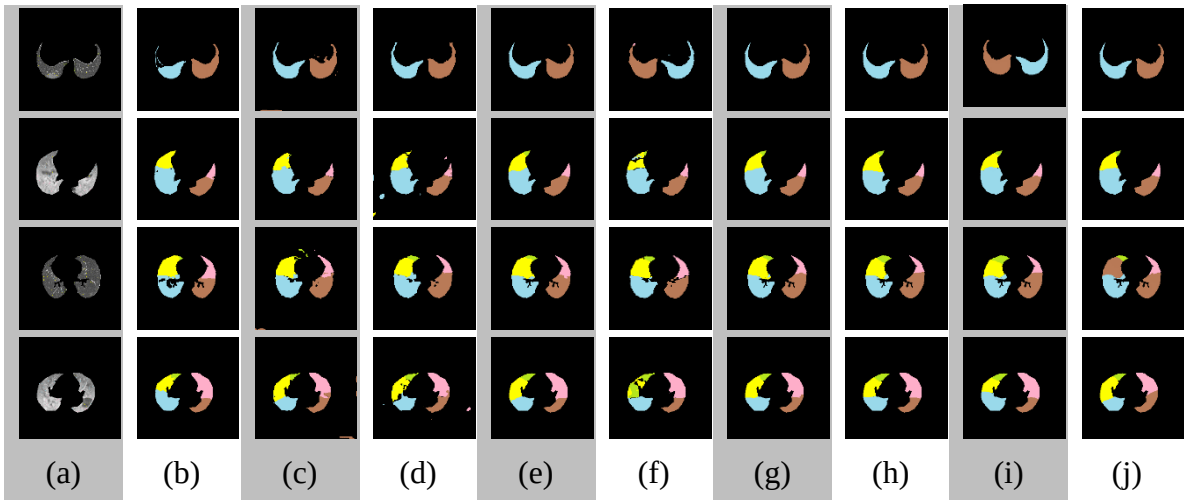


Figure 3.4: The qualitative results on axial slices for pulmonary lobe segmentation: (a) Original images, (b) DenseU-Net, (c) DeepLabV3+ (VGG-19), (d) DeepLabV3+ (VGG-16), (e) U-Net (VGG-19), (f) DeepLabV3+ (ResNet-50), (g) U-Net (VGG-16), (h) U-Net, (i) ResUNet, (j) Ground Truth. The colors of the pulmonary lobes are shown for the upper right ,middle right ,lower right ,upper left and lower left .

Table 3.4 shows the performance comparison on *DeepChestNet-COVID* for DSC and IoU scores. ResUNet has achieved the highest performance for the COVID-19 lesion segmentation due to the residual network. U-Net without a backbone performs better than U-Net with VGG-16 and VGG-19 through its shallower network, as a deeper network leads to the degradation problem. DeepLabV3+ with ResNet-50 is better than the other extensions of DeepLabV3+ due to the residual network. DenseU-Net relatively underperformed. Moreover, the qualitative results of U-Net (without backbone, with VGG-16 and VGG-19), ResUNet, DenseU-Net, and DeepLabV3+ (with ResNet50, VGG-16, and VGG-19) are illustrated in Figure 3.5. The ResUNet usually does not involve false-negative pixels and segments more accurately lesion regions.

DeepChestNet, with ResUNet, was also compared with recent lung segmentation approaches and results given in Table 3.5 in terms of DSC and IoU scores. The approach [35] underperformed due to failing both region-growing and model-based segmentation methods. The method [32], which employs U-Net, is insufficient to extract the features of images due to the degradation problem observed in the deeper network. The performance of the method [36, 37] on COVID-19 CT images are close to DeepChestNet, but they show less performance due to the poor feature extraction of Seg3DNet and SP-V-Net.

Similarly, the pulmonary lobe segmentation results using DeepChestNet with ResUNet were compared with the state-of-the-art approaches in terms of DSC and IoU scores in Table 3.6. The model in [40] lacks generalization capability as the number of CT images used to train the model is insufficient for 3D pulmonary lobe segmentation. In [39, 41], pre-processing with a threshold-based algorithm before training reduces the quality of lung extraction in the images where the lesion region is dense, resulting in degradation in the model performance.

Last, comparisons for COVID-19 lesion segmentation with state-of-the-art approaches are given in Table 3.7. The models in [49, 51-53] underperform than DeepChestNet as a small-scale dataset was used in training. Besides, a relatively small 3D dataset leads to poor generalization for the model in [47]. Thus, DeepChestNet outperformed the recent approaches in COVID-19 lesion segmentation.

Table 3.4: The performance comparison for the COVID-19 lesion segmentation in terms of IoU and DSC scores.

Method	**IoU (%)**	**DSC (%)**
DeepLabV3+ (VGG-16)	71.76 ± 2.15	82.93 ± 1.13
DeepLabV3+ (VGG-19)	74.46 ± 2.30	84.74 ± 1.23
U-Net (VGG-19)	80.02 ± 1.79	88.28 ± 0.93
DenseU-Net	84.42 ± 1.84	91.09 ± 0.96
U-Net (VGG-16)	87.98 ± 1.45	93.30 ± 0.83
DeepLabV3+ (ResNet-50)	89.60 ± 2.01	94.32 ± 0.98
U-Net	91.94 ± 2.26	95.77 ± 1.19
ResUNet	**95.89 ± 1.38**	**97.89 ± 0.79**

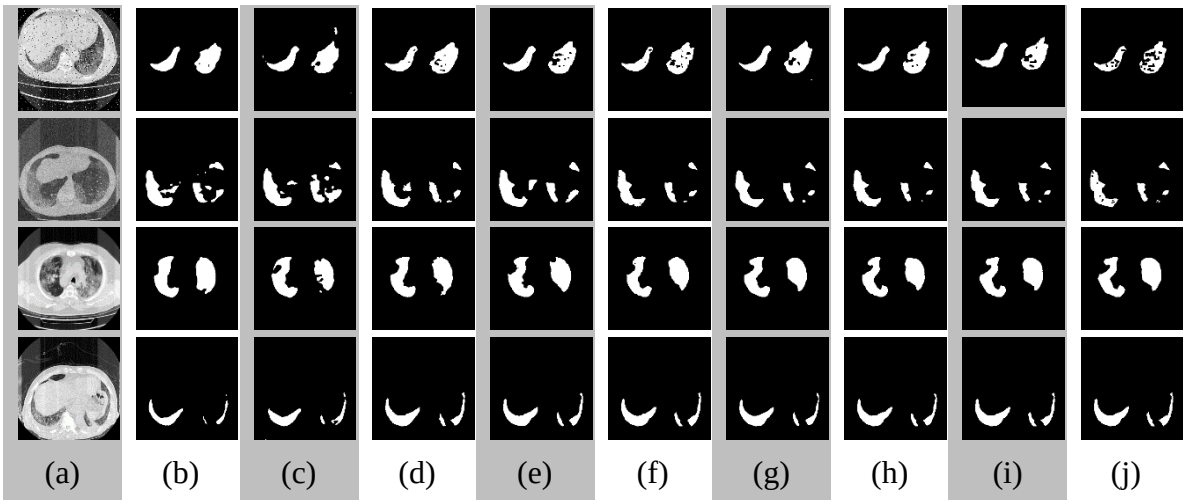


Figure 3.5: The qualitative results on axial slices for the COVID-19 lesion segmentation: (a) Original images, (b) DeepLabV3+ (VGG-16), (c) DeepLabV3+ (VGG-19), (d) U-Net (VGG-19), (e) DenseU-Net, (f) U-Net (VGG-16), (g) DeepLabV3+ (ResNet-50), (h) U-Net, (i) ResUNet, (j) Ground Truth.

Table 3.5: The quantitative comparison between *DeepChestNet* and state-of-the-art approaches for lung segmentation.

Method	**IoU (%)**	**DSC (%)**
GLCM + U-Net [32]	-	89.42
U-Net + Transfer Learning [33]	-	94.47
Initial Lung Segmentation [34]	-	96.80
Fusion Approach [35]	-	97.16
SP-V-Net [36]	-	97.96
Seg3DNet [37]	-	98.45
DeepChestNet	**98.71**	**99.35**

Table 3.6: The quantitative comparison for pulmonary lobe segmentation between *DeepChestNet* and state-of-the-art approaches.

Method	**IoU (%)**	**DSC (%)**
M^2UNet [39]	-	78.50
FRV-Net [40]	-	93.00
V-Net [41]	-	94.17
PDV-Net [43]	-	95.00
nnU-Net [44]	-	96.40
3D U-Net [45]	94.00	97.00
RTSU-Net [46]	94.90	-
DeepChestNet	**99.47**	**99.73**

Furthermore, *DeepChestNet App* was developed with DeepChestNet, which completes three tasks in 12 seconds for around 235 chest CT images, which can include several lesion regions in each lobe. The number of lesion regions provides additional information if a lesion is bilateral, multifocal, or multilobar. A bilateral lesion is defined as the presence of the lesion region in the right and left lungs. Having more

Table 3.7: The quantitative comparison between *DeepChestNet* and state-of-the-art for the COVID-19 lesion segmentation.

Method	**IoU (%)**	**DSC (%)**
3D U-Net [47]	-	67.32
Kapur's entropy [48]	57.30	71.40
Semi-Inf-Net [49]	-	73.90
U-Net [50]	-	78.30
U-Net [51]	-	83.10
Multitask [52]	-	88.00
U-Net + RA [53]	-	94.00
DeepChestNet	**95.89**	**97.89**

than one lesion in one pulmonary lobe is called a multifocal lesion. The fact that a lesion region appears in more than one pulmonary lobe is called a multilobar lesion. The information on bilateral, multifocal, or multilobar lesions provides the COVID-19 categories, such as typical, atypical, indeterminate, and negative of COVID-19 [38]. Healthcare professionals can interpret the findings, such as bilateral, multifocal, or multilobar lesions, to obtain the category of COVID-19, leading to applying proper treatment for the patient.

3.6 Chapter Summary

In this chapter, we propose a new framework called DeepChestNet for COVID-19 detection under a user-friendly desktop application, *DeepChestNet App*, by jointly performing lung, lobe, and lesion segmentation and classification with detected findings on CT images. In DeepChestNet, we first employ lung segmentation on each CT image to deal with the challenging problem of pulmonary lobe segmentation. We further develop a ResUNet based network to detect lesion regions, where the context information provided by pulmonary lobe segmentation can be employed to improve the performance of diagnosing COVID-19 findings such as bilateral, multifocal, and multilobar lesions. The CNN architectures have been trained with our *DeepChestNet-Lung*, *DeepChestNet-Lobe*, and *DeepChestNet-COVID* datasets collected for the lung,

lobe, and COVID-19 lesion segmentation due to the lack of a large-scale dataset in the literature. The performance of DeepChestNet in terms of DSC is 99.35 $\pm$ 0.18 %, 99.73 $\pm$ 0.34 %, and 97.89 $\pm$ 0.79 % for the lung, pulmonary lobe, and COVID-19 lesion segmentation, respectively. The lung, pulmonary lobe, and COVID-19 lesion segmentation experiments present the benchmark performance in terms of IoU as 98.71 $\pm$ 0.35 %, 99.47 $\pm$ 0.67 %, and 95.89 $\pm$ 1.38 %, respectively. Experimental results on *DeepChestNet-Lung*, *DeepChestNet-Lobe*, and *DeepChestNet-COVID* demonstrate that DeepChestNet achieves promising results in COVID-19 detection, compared with several state-of-the-art approaches.

Chapter 4

U-TranSvision: Transformer-based Deep Supervision Approach for COVID-19 Lesion Segmentation on Computed Tomography Images

In the previous chapter, lung, pulmonary lobe, and COVID-19 lesion segmentation are studied, assuming that the infected regions on CT images are large to minimize the class imbalance problem causing poor predictive performance. We extended our dataset with images of tiny infected regions to further improve the previous study. Here, we focused only on COVID-19 lesion segmentation using our extended dataset, which also includes images with tiny infected regions, to address the limitation in the previous chapter.

AI-assisted COVID-19 detection in chest CT images has an important role in the early diagnosis and appropriate treatment of infected patients. CNN-based AI approaches have shown significant performance in segmenting COVID-19 lesion regions. However, they have several limitations to deal with the complexity of the lesion characteristics, low or high image contrast, and small lesion regions. To address these limitations, we propose a novel architecture called *U-TranSvision*, which leverages transformers and deep supervision to improve segmentation performance by focusing on the salient features of small COVID-19 lesions. Furthermore, Pix2Pix GAN was used in data augmentation to improve the performance of *U-TranSvision*, and pre-processing steps were applied to remove the noise around human tissue on an image. In addition, we expanded our previous dataset to a relatively large-scale dataset of 11,717 axial chest CT images, along with their corresponding pixel-level annotations. Based on extensive experimental evaluations, *U-TranSvision* achieved a DSC of 85.57

% and an IoU of 74.82 %. The experiments were also conducted on three publicly available datasets, such as COVID-19-CT-Seg, MosMedData, and MedSeg, to demonstrate the robustness of *U-TranSvision*. The qualitative and quantitative results proved that *U-TranSvision* had promising performance compared to the state-of-the-art architectures for COVID-19 lesion segmentation. In addition, *U-TranSvision* has relatively low learning parameters, which results in low computational costs.

In the following, we introduce COVID-19 lesion segmentation, its limitations, and our approach in Section 4.1. Then, Section 4.2 presents our proposed *U-TranSvision* architecture, including its structure, transformer layer, deep supervision, and hybrid loss function. Section 4.3 describes our collected dataset, while Section 4.4 contains the experimental results using CNN-based architectures on our dataset and three public datasets. Next, we discuss the experiments on COVID-19 lesion segmentation follows, and the chapter is summarized in Section 4.5.

4.1 Introduction

A comprehensive approach for better performance of CNN-based architectures in COVID-19 lesion segmentation on chest CT images is to use various approaches, such as data augmentation, transfer learning, deep supervision, attention mechanism, transformer, and residual connections. Data augmentation applies random transformations to the training set to increase the amount of data, while transfer learning uses a pre-trained model as a starting weight for a new task. Deep supervision adds intermediate outputs at multiple levels to a CNN-based architecture, while attention mechanisms allow the model to focus on salient features of the input data. Transformers use the attention mechanism to process input sequences. Residual connections allow information to bypass certain layers in a CNN-based architecture and pass directly to subsequent layers. Among them, the popularities of transformers and deep supervision have increased for better model performance, while data augmentation is a common approach to improve generalization ability. In the previous chapter, residual connections were employed in the segmentation tasks, but this increased the complexity of our model. In addition, transfer learning was not applied since we do not have a similar dataset with a sufficient number of CT images.

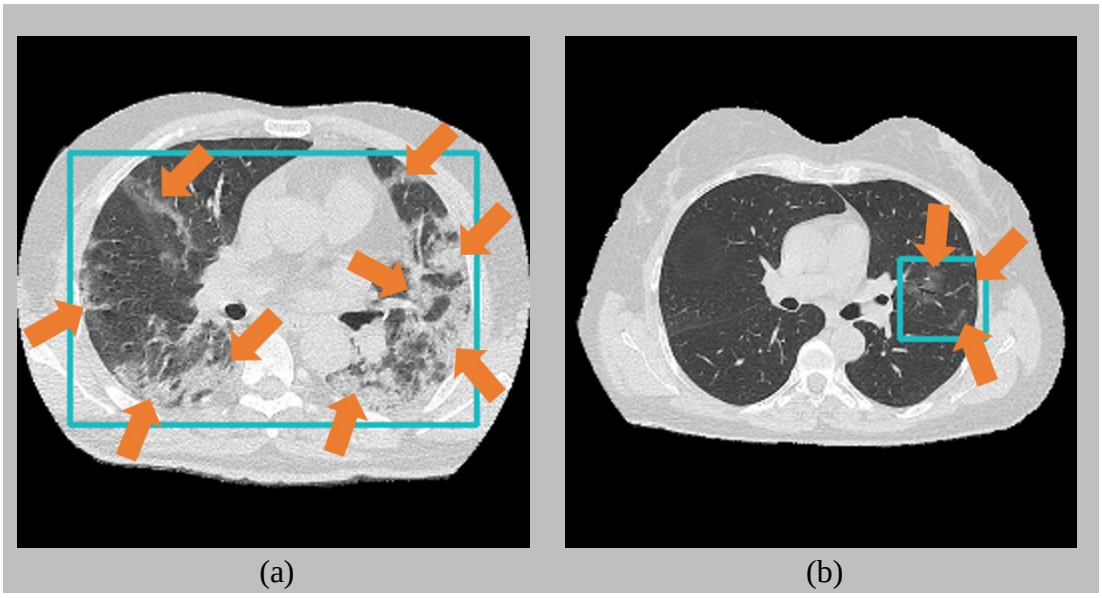


Figure 4.1: Two examples of COVID-19 positive CT images from our dataset.

Therefore, this chapter considers transformers, deep supervision, and data augmentation for COVID-19 lesion segmentation.

Despite theoretical and algorithmic advances in CNN-based image segmentation approaches, there are several challenges associated with segmenting COVID-19 lesions on CT images. First, typically infected CT images can have divergent characteristics, such as complex appearances, noisy, fuzzy borders, and irregular shapes. In particular, the COVID-19 lesion regions have low contrast, and visually unclear boundaries, making them complicated to segment, as shown in Figure 4.1. Second, COVID-19 datasets generally have a long-tailed distribution due to small and uneven infected regions on CT images, usually seen in the early stages of the disease. It is called a class imbalance problem, causing the model to perform poorly. Third, although small-scale datasets are publicly available, large datasets with pixel-wise annotation for COVID-19 lesion segmentation are scarce. In addition, datasets with an insufficient number of images can cause weak generalization ability, lack of robustness, and poor model performance on previously untrained data for CNN architectures. To handle the data scarcity of public datasets, a relatively large-scale dataset has been collected in the previous chapter. However, it showed poor performance on images with small infected regions due to the class imbalance problem. Therefore, training has been performed on images that contain only large COVID-19 regions. To overcome this limitation, we first expanded the dataset with more images, including small infected regions. Then, the Pix2Pix GAN

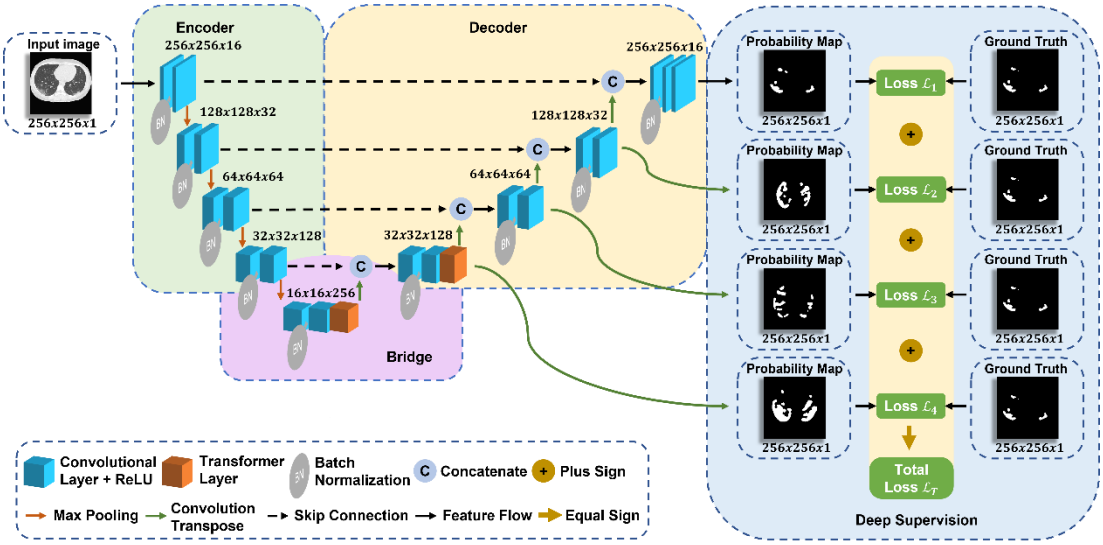


Figure 4.2: The overall proposed *U-TranSvision* architecture.

architecture [84] was used instead of traditional data augmentation approaches [60, 65, 66] to augment the images in the datasets, which improves the generalization ability of our model and leads to a better segmentation performance. As a pre-processing, several image processing algorithms, including thresholding, binarizing, masking, and contour detection, were applied to reduce noise outside the lung regions. Then, a new architecture called *U-TranSvision* was proposed to minimize class imbalance using hybrid loss functions that combine focal [95] and dice [96] losses with transformers under a U-shaped structure. Transformers have a global modeling capability based on attention mechanism to capture long-range dependencies and improve COVID-19 lesion segmentation. Therefore, the latest trends in transformers combine the superiority of transformers with CNN-based architectures to further capture salient and contextual features of images [60]. Hence, our approach leverages a U-shaped structure and transformers to capture features of COVID-19 infections in CT images, focusing on small regions, blurred appearance, and noisy characteristics. Our approach employs deep supervision to achieve better segmentation performance, accelerate model convergence, and reduce time consumption. Besides our expanded dataset, *U-TranSvision* was also tested with three publicly available datasets to demonstrate its robustness.

4.2 *U-TranSvision* Architecture

This section provides a comprehensive overview of the end-to-end *U-TranSvision*, including information on the network architecture, core components, and the loss function for COVID-19 lesion segmentation. Our proposed *U-TranSvision* architecture is designed to achieve the following goals: high segmentation performance, a limited number of trainable parameters, and optimal memory consumption. To achieve these goals, *U-TranSvision* leverages the benefits of the U-shaped architecture, the transformer layer, and the deep supervision approach, as described in Figure 4.2. Further details of each layer in *U-TranSvision* are also given in Table 4.1, including layer size and the number of trainable parameters. The proposed architecture consists of the encoder, bridge, and decoder structures borrowed from U-Net [22] to segment COVID-19 lesion regions on CT images. The encoder downsamples the input image to capture the high-level features in different resolution levels, while the decoder recovers the spatial dimensions of the feature maps for object localization. The bridge connects the encoder to the decoder for high-level feature flow.

4.2.1 Encoder

In *U-TranSvision*, the encoder, which captures high-level features of an input image, consists of several blocks, including convolutions, activation functions, pooling operations, and batch normalization. The input image is processed by the blocks using standard convolution (padded convolution) operations to compute feature maps. Each block contains two 3×3 convolution layers followed by a rectifier linear unit as an activation function and a batch normalization. The ReLU function ensures non-linearity for the layers to efficiently capture complex features at multiple scales. Batch normalization reduces the internal covariate shift, which is the change in the distribution of the feature maps, to improve segmentation performance on unseen data, stabilize training, and lead to faster convergence. After each block, a 2×2 max-pooling operation with stride two is added to downsample the feature maps. The convolutions in the first block have 16 filters, followed by convolutions with 32, 64, and 128 filters to capture image patterns, such as edges, corners, and dots.

Table 4.1: The layers of the proposed architecture and their comprehensive details.

	Layer	Input Size	Kernel Size	Kernel Depth	Stride	Params
Encoder	Input Layer	$256 \times 256 \times 1$	-	-	-	0
	Convolution 1	$256 \times 256 \times 1$	3×3	16	1×1	160
	Batch Normalization 1	$256 \times 256 \times 16$	-	-	-	64
	Convolution 2	$256 \times 256 \times 16$	3×3	16	1×1	2,320
	Max pooling 1	$256 \times 256 \times 16$	2×2	16	2×2	0
	Convolution 3	$128 \times 128 \times 16$	3×3	32	1×1	4,640
	Batch Normalization 2	$128 \times 128 \times 32$	-	-	-	128
	Convolution 4	$128 \times 128 \times 32$	3×3	32	1×1	9,248
	Max pooling 2	$128 \times 128 \times 32$	2×2	32	2×2	0
	Convolution 5	$64 \times 64 \times 32$	3×3	64	1×1	18,496
	Batch Normalization 3	$64 \times 64 \times 64$	-	-	-	256
	Convolution 6	$64 \times 64 \times 64$	3×3	64	1×1	36,928
	Max pooling 3	$64 \times 64 \times 64$	2×2	64	2×2	0
	Convolution 7	$32 \times 32 \times 64$	3×3	128	1×1	73,856
	Batch Normalization 4	$32 \times 32 \times 128$	-	-	-	512
	Convolution 8	$32 \times 32 \times 128$	3×3	128	1×1	147,584
	Max pooling 4	$32 \times 32 \times 128$	2×2	128	2×2	0
Bridge	Convolution 9	$16 \times 16 \times 128$	3×3	256	1×1	295,168
	Batch Normalization 5	$16 \times 16 \times 256$	-	-	-	1024
	Convolution 10	$16 \times 16 \times 256$	3×3	256	1×1	590,080
	Transformer 1	$16 \times 16 \times 256$	-	-	-	597,248
	Convolution Transpose 1	$16 \times 16 \times 256$	2×2	128	2×2	131,200
	Concatenate 1	$32 \times 32 \times 128$	-	-	-	0
Decoder	Convolution 11	$32 \times 32 \times 256$	3×3	128	1×1	295,040
	Batch Normalization 6	$32 \times 32 \times 128$	-	-	-	512
	Convolution 12	$32 \times 32 \times 128$	3×3	128	1×1	147,584
	Transformer 2	$32 \times 32 \times 128$	-	-	-	332,672
	Convolution Transpose 2	$32 \times 32 \times 128$	2×2	64	2×2	32,832
	Concatenate 2	$64 \times 64 \times 64$	-	-	-	0
	Convolution 13	$64 \times 64 \times 128$	3×3	64	1×1	73,792
	Batch Normalization 7	$64 \times 64 \times 64$	-	-	-	256
	Convolution 14	$64 \times 64 \times 64$	3×3	64	1×1	36,928
	Convolution Transpose 3	$64 \times 64 \times 64$	2×2	32	2×2	8,224
	Concatenate 3	$128 \times 128 \times 32$	-	-	-	0
	Convolution 15	$128 \times 128 \times 64$	3×3	32	1×1	18,464
	Batch Normalization 8	$128 \times 128 \times 32$	-	-	-	128
	Convolution 16	$128 \times 128 \times 32$	3×3	32	1×1	9,248
	Convolution Transpose 4	$128 \times 128 \times 32$	2×2	16	2×2	2,064
	Concatenate 4	$256 \times 256 \times 16$	-	-	-	0
	Convolution 17	$256 \times 256 \times 32$	3×3	16	1×1	4,624
	Batch Normalization 9	$256 \times 256 \times 16$	-	-	-	64
	Convolution 18	$256 \times 256 \times 16$	3×3	16	1×1	2,320
	Convolution 19 (Output)	$256 \times 256 \times 16$	3×3	1	1×1	17
	Total params: 2,874,580		**Trainable params: 2,869,524**			

*-: Not applicable.

4.2.2 Bridge

The bridge between the encoder and decoder of *U-TranSvision* is the connection to flow high-level features. It consists of two convolution layers with 256 filters, two ReLU functions, a batch normalization, and a transformer layer. The bridge also includes the skip connection, which allows feature flow at different resolution levels, between the encoder and decoder. The multi-head self-attention (MSA) in the transformer layer allows the model to focus on the most relevant parts of the feature maps. In addition, the transformer can capture the long-range dependencies that represent the relationship between pixels within a single image, as well as the relationships between voxels in stacked images.

4.2.3 Transformer-based Decoder

Segmentation architectures create an output probability map containing an ROI and matching the resolution of an input image. In that sense, the decoder, including four blocks, gradually increases the resolution of the encoded features from the encoder. Each block has two convolutional layers with the ReLU function, batch normalization, and concatenation, followed by a transposed convolution that increases the receptive field of the feature map. To preserve the fine details that might be lost during the max-pooling operation, the feature maps fed by the skip connection are combined with the output of the transposed convolutions through the concatenation layers. The feature channels of each block in the decoder must be consistent with the number of filters in the encoder, starting at 128 and decreasing to 64, 32, and 16 filters, respectively. In the block with 128 feature channels, a transformer layer is employed to focus on the ROI in the feature maps from the encoder and mitigate the effect of long-range spatial dependencies.

The global modeling capability of the transformers offers to capture contextual and salient features within an image, which leads to improved segmentation performance on previously unseen data. Hence, transformer layers were used in *U-TranSvision* to address the above concerns. The number of transformers is limited to one in the decoder of *U-TranSvision* since the transformer increases the number of learning parameters and requires high computational power in training. The structure of the

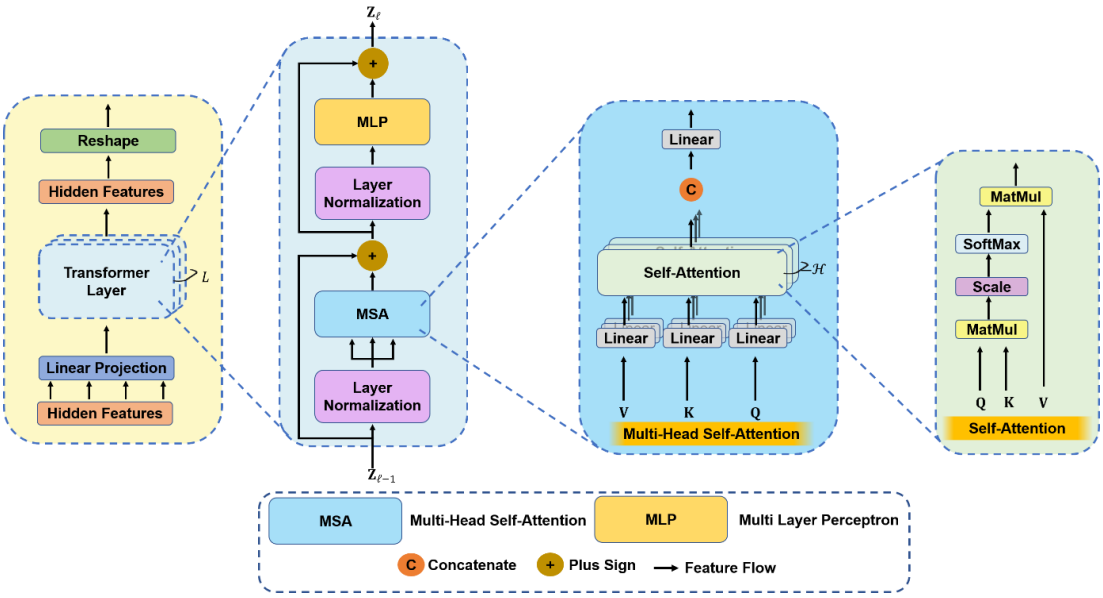


Figure 4.3: The detailed view of the transformer layer and its components.

transformer layer was adapted from [97] as depicted in Figure 4.3. A standard transformer layer uses 1D sequences of token embeddings as an input. Therefore, the hidden features in the bridge and decoder of *U-TranSvision* are reshaped into a sequence of vectorized 2D patches. A trainable linear projection maps the vectorized patches to a latent vector embedding space. The patch and position embeddings are summed to preserve position information. After this process, the transformer layer is fed with the embedded sequences. A layer normalization (LN) is used to normalize the embedded sequences, and it follows the MSA block, which separately computes attention weights over multiple heads for focusing on salient features of images. Then, the embedded sequences are added to the output of the MSA block. An LN normalizes the hidden features created after the adding operation, and its output feeds a multi-layer perception (MLP) block, which can capture complex features of sequences. At the end of a transformer layer, the output of the MLP block is summed with the hidden features. The transformer layer can be used recursively, but only two transformer was employed in *U-TranSvision* due to computational limitations. Finally, the hidden features in the output of the transformer are reshaped to the resolution level for the transformer layer.

The input of a typical transformer layer is a 1D sequence of token embeddings. Therefore, the hidden features $\mathbf{x} \in \mathbb{R}^{\mathcal{H}\times\mathcal{W}\times\mathcal{C}}$ is reshaped into a sequence of flattened 2D patches $\{x_n \in \mathbb{R}^{P^2.\mathcal{C}} \mid n = 1,..,N\}$. Here, $\mathcal{H} \times \mathcal{W}$ represents the spatial

resolution of the hidden features, $\mathcal{H}$ is the height of the hidden features, $\mathcal{W}$ states the width of the hidden features, $\mathcal{C}$ denotes the number of channels, x_n is a vectorized patch at the n-th index, $P \times P$ is the size of each patch, and $N = \frac{\mathcal{H}\mathcal{W}}{P^2}$ is the number of patches. The vectorized patches x_n are mapped into a latent vector embedding space of size D using a trainable linear projection. Next, specific position embeddings are added to the patch embeddings to preserve position information as described:

$$\mathbf{Z_0} = [\mathrm{x}_1\mathbf{E}; \mathrm{x}_2\mathbf{E}; \dots; \mathrm{x}_\mathrm{N}\mathbf{E}] + \mathbf{E}_{\boldsymbol{pos}} \tag{4.1}$$

where $\mathbf{Z}_0$, $\mathbf{E} \in \mathbb{R}^{(P^2.\mathcal{C})\times D}$ and $\mathbf{E}_{pos} \in \mathbb{R}^{N\times D}$ are the sequence of tokens in the 0-th layer, the patch embedding projection, and position embedding, respectively. The transformer is designed as L-layers, including MSA and MLP blocks, so its ℓ-th layer can be defined as follows:

$$\begin{aligned} \widehat{\boldsymbol{Z}}_\ell &= \mathbf{MSA}\big(\mathbf{LN}(\boldsymbol{Z}_{\ell-1})\big) + \boldsymbol{Z}_{\ell-1} \\ \boldsymbol{Z}_\ell &= \mathbf{MLP}\left(\mathbf{LN}\big(\widehat{\boldsymbol{Z}}_\ell\big)\right) + \widehat{\mathbf{Z}}_\ell \end{aligned} \tag{4.2}$$

where LN(·) represents the layer normalization, which normalizes the mean and variance of each layer across the feature dimension. Here, $\mathbf{Z}_\ell$ denotes a sequence that is the output of the ℓ-th layer, $\mathbf{Z}_{\ell-1}$ represents the $\ell - 1$-th layer, and $\widehat{\mathbf{Z}}_{\ell-1}$ means the hidden features in the transformer. However, the transformer layer in *U-TranSvision* contains only one layer to reduce computational complexity. In the transformer layer, the MLP block includes a two-layer feedforward network to capture complex sequence features and uses a Gaussian error linear unit as its activation function for non-linearity.

A conventional self-attention (SA) uses a single attention mechanism to compute the attention weights. However, the MSA block splits the attention mechanism into multiple heads, each of which computes attention weights separately. First, the MSA creates queries $\mathbf{Q}$, keys $\mathbf{K}$, and values $\mathbf{V}$ from the input sequence $\widehat{\mathbf{Z}}_{\ell-1}$, in which the layer normalization is applied to $\mathbf{Z}_{\ell-1}$, using a trainable linear projection. Next, attention weights are computed through SA using $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ to focus on more

meaningful features. The MSA block involves executing $\mathcal{H}$ times SA operations, called heads, in parallel and linearly projecting their concatenated outputs using a trainable linear projection.

4.2.4 Deep Supervision

In deep learning, deep supervision refers to a training strategy where hidden layers of a network are used as supervision for better learning performance. The idea is based on adding supervision at multiple levels of spatial resolution, which can lead to improved performance and faster convergence [98]. Moreover, with deep supervision, the gradient of a network can be backpropagated through more layers, which allows for effective learning and helps mitigate the vanishing or exploding gradient problem. Therefore, *U-TranSvision* takes advantage of deep supervision to further improve predictive performance. In the *U-TranSvision*, the output at different resolution levels in the decoder is applied to a transpose convolution with different strides (changing from the bottom to the top of the architecture as 8, 4, 2, 1) and a kernel size of 2×2 to expand receptive fields based on ground truth. In this way, a probability map is created for each resolution level via a sigmoid activation function. Next, the loss functions are computed using each probability map and the ground truth, following which they are summed to produce a total loss, as discussed next.

4.2.5 Loss Functions

The class imbalance problem causes poor predictive performance on minority classes in segmentation and classification tasks. To mitigate this issue, a hybrid loss function, including focal loss [95] and dice loss [69, 96], was used for the COVID-19 lesion segmentation. As a kind of cross-entropy loss [95], the focal loss can adjust the weights of the majority and minority classes to reduce the class imbalance problem. Therefore, focal loss, which can effectively focus on the minority classes, helps to improve the segmentation performance [67]. In addition, dice loss, which highlights true positive predictions, computes the overlap ratio between the probability map and the ground truth. This also contributes to the performance of *U-TranSvision* by minimizing the total loss function. The hybrid loss function can be formulated as follows:

$$\mathcal{L}_{hybrid}(\widehat{\mathbf{S}}_{\mathbf{k}}, \mathbf{S}) = \mathcal{L}_{focal}(\widehat{\mathbf{S}}_{\mathbf{k}}, \mathbf{S}) + \mathcal{L}_{dice}(\widehat{\mathbf{S}}_{\mathbf{k}}, \mathbf{S})$$

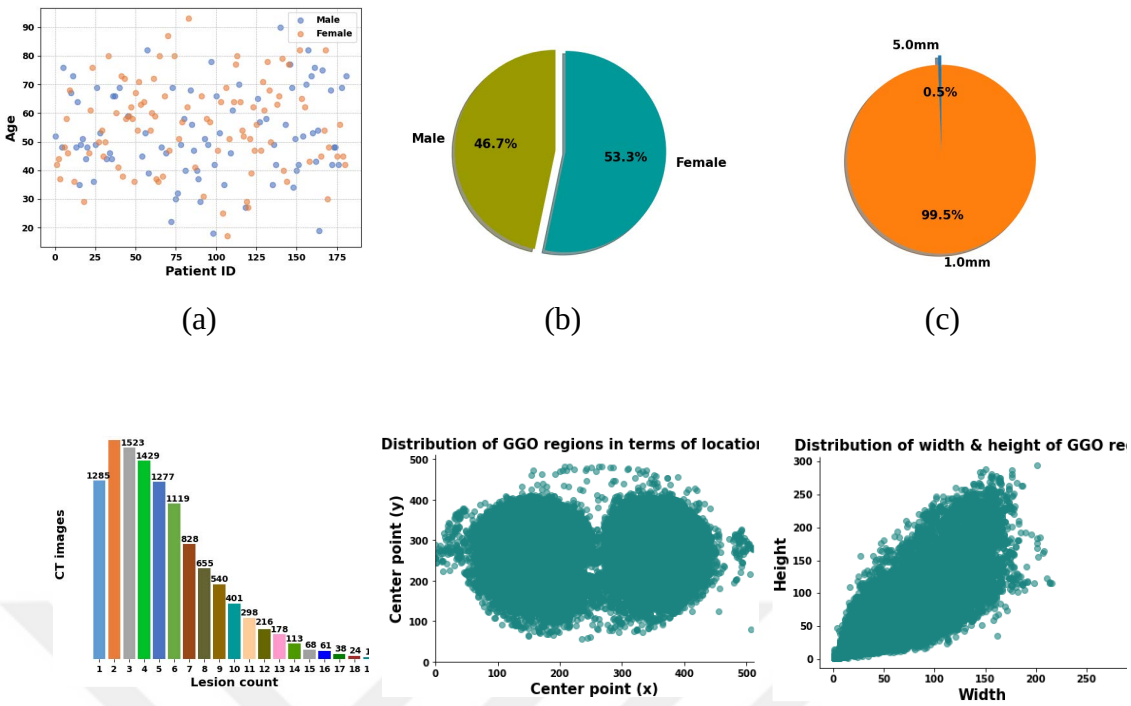
$$\mathcal{L}_{focal}(\widehat{\mathbf{S}}_{\mathrm{k}}, \mathbf{S}) = -\frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W}\mathbf{S}^{i,j}\cdot\alpha\cdot(1-\widehat{\mathbf{S}}_{k}^{i,j})^{\gamma}\cdot\log(\widehat{\mathbf{S}}_{k}^{i,j}) \tag{4.3}$$

$$\mathcal{L}_{dice}(\widehat{\mathbf{S}}_{\mathrm{k}}, \mathbf{S}) = 1 - \frac{\sum_{i=1}^{H}\sum_{j=1}^{W}\widehat{\mathbf{S}}_{k}^{i,j} + \varepsilon}{\sum_{i=1}^{H}\sum_{j=1}^{W}\widehat{\mathbf{S}}_{k}^{i,j} + \sum_{i=1}^{H}\sum_{j=1}^{W}\mathbf{S}^{i,j} + \varepsilon}$$

where α is used to adjust the weight of the background and foreground, while γ is a modulating factor for the focal loss. The output probability map, denoted by $\widehat{\mathbf{S}}_{\mathrm{k}}$, is generated for each input image **I** at different resolution levels of deep supervision, where k can vary between one and four due to four resolution levels in the decoder, and **S** represents the ground truth. Here, the resolution of $\widehat{\mathbf{S}}_{\mathrm{k}}$ and **S** is represented by the height H, and the width W. Besides, ε is a smoothing parameter that avoids division by zero and set to one.

4.3 Data Acquisition and Pre-processing

A large and diverse dataset is crucial to building a robust deep learning model, leading to a better generalization ability on previously unseen data. Although several datasets for COVID-19 are publicly available, either lesion labels or small-scale samples are missing. To address this limitation, the size of our previous dataset [99] was expanded from 1,034 images to 11,717 CT images belonging to 182 cases with COVID-19 lung involvement. Semi-automated segmentation was then performed to label lesion regions on CT images using MedSeg [100], a web-based segmentation tool. After performing the lesion segmentation with MedSeg, two experts manually confirmed the ground truth by consensus. In the ground truth, the '1' label stands for the foreground, while the '0' label represents the background regions. A comprehensive description of our expanded dataset is explained as follows. The age distribution of the patients ranges from 17 to 93 years old, as presented in Figure 4.4(a), while the pie chart about the gender percentages of the cases, including 85 males and 97 females, is shown in Figure 4.4(b). The distribution of image thickness is shown as a pie chart in Figure 4.4(c). Figure 4.4(d) shows the distribution of lesion counts in CT images, which vary from 1 to 19. In Figure 4.4(e), the center points of the lesions are scattered over the image due to noise effects during imaging, but they are usually concentrated in the lung regions. Figure 4.4(f) shows that the width and



(a) (b) (c)

(d) (e) (f)

Figure 4.4: The detailed statistics about our dataset.

height of the lesion regions range from 1 to 293 and 1 to 294, respectively. In addition to our dataset, three publicly available datasets were employed in this chapter. One of the public datasets is COVID- 19-CT-Seg[1], which contains 20 pixel-wise annotated chest CT scans with 3,520 images and their corresponding ground truth for infection regions. MedSeg (Medical Segmentation)[2], the second public dataset, contains 350 labeled axial chest images from 9 CT scans. MosMedData[3], which contains 1,110 CT scans, is the third publicly available dataset used in this chapter.

To improve the capability of deep learning for more distinguishing features, pre-processing steps can be performed on the dataset. In this context, the pixel intensity values of CT images with a window width and window center of 1600 and -550 Hounsfield units were normalized to lie within the range of [0, 1] for all datasets. The

[1] https://zenodo.org/record/3757476#.ZAsrVHZByUk

[2] https://medicalsegmentation.com/covid19/

[3] https://mosmed.ai/datasets/covid191110/

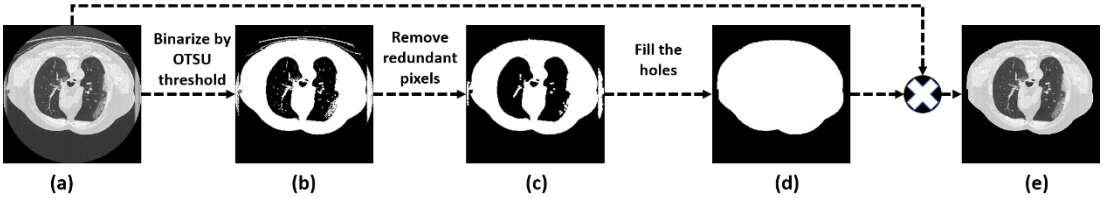


Figure 4.5: The pipeline of the sequential pre-processing steps on the dataset.

fact that outside of human tissue on CT images contains plenteous noisy pixels due to the scanning process can impact segmentation performance. To mitigate these effects, the noisy pixels outside the human tissue were removed to focus on the ROI using a pipeline based on the study [60]. First, the Otsu algorithm was employed to compute the threshold value for each image. As shown in Figure 4.5(b), the images in Figure 4.5(a) were binarized using the Otsu algorithm, where each pixel value was compared to the threshold. Pixels with values greater than the threshold were considered foreground, while pixel values less than or equal to the threshold represented the background. Although some pixels associated with non-human parts of the images are considered foreground, they can be easily removed with a morphological opening operation on the binarized image, as shown in Figure 4.5(c). Next, the hole-filling algorithm was employed to create a mask of the segmented human tissue, followed by two recursive morphological opening operations as illustrated in Figure 4.5(d). Finally, in Figure 4.5(e), the mask and the initial image were overlapped to ensure that the CNN-based architectures were focused on the ROI.

4.4 Results and Discussion

In this section, the implementation details, including hyperparameters and the algorithm, are presented in Subsection 4.4.1, while Subsection 4.4.2 discuss the experimental results for ablation studies of *U-TranSvision*. Subsection 4.4.3 details quantitative and qualitative results for COVID-19 lesion segmentation. These results are followed by a comparison between our approach and state-of-the-art architectures.

4.4.1 Implementation Details

U-TranSvision was implemented using Python, which includes Keras 2.8.0 with Tensorflow 2.8.0 as the backend for the lesion segmentation task. The experimental

settings include the Adam optimization algorithm with a learning rate of 0.0005, a batch size of 32, and kernel initialization with he-normal initialization. In addition, the γ and α hyper-parameters in the hybrid loss function were set to 2 and 0.25, respectively. All experiments were conducted using an early stop of 100 epochs to avoid the overfitting problem, on a computer with NVIDIA RTX A5000 24 GB graphics processing unit and Windows 10 operating system running Python 3.8 environment. To quantitatively evaluate the performance of the lesion segmentation, five-fold cross-validation [50] was conducted on the datasets. Therefore, the dataset was split into five-fold to alleviate the data leakage that causes to decrease in the predictive performance of the model. In each iteration, the four-fold were merged into a training subset, while the rest was used for testing. The training subset was then divided into training and validation subsets. All CNN-based architectures were trained and tested with the same five-fold to ensure a fair comparison. In addition, the pseudocode of the entire training and optimization process for *U-TranSvision* to

Algorithm 4.1**:** The pseudocode of *U-TranSvision* architecture for training.

Input: Chest CT image, **I**; COVID-19 lesion ground truth, **S**; Number of epochs, E; Probability maps, $\widehat{\boldsymbol{S}}_{\mathbf{1}}, \widehat{\boldsymbol{S}}_{\mathbf{2}}, \widehat{\boldsymbol{S}}_{\mathbf{3}}, \widehat{\boldsymbol{S}}_{\mathbf{4}}$

Output: COVID-19 lesion probability map, $\widehat{\boldsymbol{S}}_{\mathbf{1}}$.

/***Train** *U-TranSvision* ***/**

Initialize weights W.

while $Epoch \leq E$ **do**

$\quad$ **Calculate** $\widehat{\boldsymbol{S}}_{\mathbf{1}}, \widehat{\boldsymbol{S}}_{\mathbf{2}}, \widehat{\boldsymbol{S}}_{\mathbf{3}}, \widehat{\boldsymbol{S}}_{\mathbf{4}} = \mathbf{W}(\mathbf{I})$

$\quad$ **Calculate Loss:**

$\quad \mathcal{L}_1 = \mathcal{L}_{hybrid}(\widehat{\mathbf{S}}_{\mathbf{1}}, \mathbf{S})$

$\quad \mathcal{L}_2 = \mathcal{L}_{hybrid}(\widehat{\mathbf{S}}_{\mathbf{2}}, \mathbf{S})$

$\quad \mathcal{L}_3 = \mathcal{L}_{hybrid}(\widehat{\mathbf{S}}_{\mathbf{3}}, \mathbf{S})$

$\quad \mathcal{L}_4 = \mathcal{L}_{hybrid}(\widehat{\mathbf{S}}_{\mathbf{4}}, \mathbf{S})$

$\quad$ **Calculate Total Loss:**

$\quad \mathcal{L}_T = \mathcal{L}_1 + \mathcal{L}_2 + \mathcal{L}_3 + \mathcal{L}_4$

$\quad$ **Run Adam optimizer to update W**

end

generate the output probability map is summarized in Algorithm 4.1.

4.4.2 Ablation Studies

This section argues that the proposed architecture is effective and performs well regarding quantitative results. This claim is supported by several ablation experiments conducted to evaluate the impact of each component in *U-TranSvision*, as listed in Table 4.2. To investigate the contribution of the transformer layer and deep supervision, the experiments were run with different combinations, including baseline, one and two transformer layers, and deep supervision. Here, a single transformer layer in the bridge is called one transformer. Similarly, two transformers mean using one transformer layer in the bridge and decoder. Based on the ablation studies, the better predictive performance in the proposed architecture can be attributed to the transformer layers in the bridge and the decoder, where transformers draw attention to salient features of the COVID-19 regions on high-level feature maps from the encoder. Likewise, the ablation studies with deep supervision revealed that the supervision approach, which effectively minimizes the loss function, has a role in further improving the predictive performance of our architecture. It is concluded that *U-TranSvision* benefits from the complementary characteristics of transformer and deep supervision to highlight their respective strengths strengths and eliminate their weaknesses in medical image segmentation.

Table 4.2: The performance comparison for the ablation study on the test set of our dataset (BL: baseline, OT: one transformer, TT: two transformers, DS: deep supervision) (the results are given as mean ± standard deviation).

BL	OT	TT	DS	IoU (%)	DSC (%)	Sensitivity (%)	Specificity (%)	Precision (%)
✓				71.33±3.64	83.22±2.47	79.40±4.11	99.70±0.06	87.71±1.93
✓			✓	72.77±4.34	84.18±2.93	79.62±5.08	**99.75±0.09**	**89.66±2.67**
✓	✓			73.07±4.28	84.38±2.87	80.93±5.41	99.71±0.09	88.53±2.54
✓		✓		74.36±3.13	85.27±2.07	81.82±2.84	99.73±0.05	89.18±1.71
✓	✓		✓	73.93±3.62	84.97±2.39	**83.18±2.90**	99.66±0.11	87.06±3.30
✓		✓	✓	**74.82±3.06**	**85.57±2.01**	82.95±3.60	99.70±0.09	88.63±2.27

4.4.3 Artificial Intelligence-Based Segmentation Results

A benchmark for COVID-19 lesion segmentation on the test set in our dataset was created using various CNN-based architectures, including U-Net, ResUNet++, ResUNet, DeepLabV3+ (as a backbone VGG-16), DeepLabV3+ (as a backbone VGG-19), and DeepLabV3+ (as a backbone ResNet-50). Moreover, the experimental results on the test sets of the datasets, including ours and three publicly available datasets, were described and compared with the state-of-the-art architectures.

Table 4.3 quantitatively compares the CNN-based architectures using our dataset for COVID-19 lesion segmentation based on IoU, DSC, sensitivity, specificity, and precision. The performance comparison between *U-TranSvision* and the CNN-based architectures is also presented in Figure 4.6 as violin plots for each evaluation metric to provide further detail on the distribution of the results. For a fair comparison, the results were evaluated with identical hardware and software configurations. Note that, the proposed *U-TranSvision* shows the highest performance among the CNN-based architectures in terms of IoU, DCS, and sensitivity. The performance of the proposed architecture can be attributed to the transformer layer, which efficiently focuses on and captures salient features in images. In addition, the deep supervision has contributed to minimizing the loss function and faster convergence during training, leading to improved predictive performance. On the other hand, this table shows that the proposed architecture has relatively fewer learning parameters and floating point operations per second (FLOPs) than the other CNN-based architectures except for U-Net. Therefore, *U-TranSvision* has the capability to give a faster response due to low complexity. Moreover, Figure 4.7 displays the qualitative results of U-Net, ResUNet++, ResUNet, DeepLabV3+ (VGG-16), DeepLabV3+ (VGG-19), DeepLabV3+ (ResNet-50), and *U-TranSvision* on axial CT images from the test set in our dataset. As demonstrated, *U-TranSvision* produces probability maps with fewer false-negative pixels and provides more accurate COVID-19 lesion segmentation.

The performance of *U-TranSvision* was also compared with similar state-of-the-art studies in Table 4.4. The evaluation results demonstrated that our approach achieved outstanding performance in COVID-19 lesion segmentation, with high scores in terms

Table 4.3: The comparison of COVID-19 lesion segmentation performance on the test set in our dataset (the results are given as mean ± standard deviation).

Method	**Params / FLOPs**	**IoU (%)**	**DSC (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**
U-Net [22]	1.943 M / 6.7 G	72.35 ± 3.98	83.91 ± 2.68	79.59 ± 4.65	99.73 ± 0.10	89.10 ± 2.91
ResUNet++ [23]	4.026 M / 28.8 G	72.15 ± 1.66	83.82 ± 1.13	78.90 ± 1.75	99.76 ± 0.06	89.58 ± 2.28
ResUNet [24]	8.052 M / 80.1 G	71.37 ± 4.37	83.23 ± 3.06	76.75 ± 5.33	**99.80 ± 0.06**	**91.32 ± 2.23**
DeepLabV3+ (VGG-16) [25]	20.687 M / 60.5 G	71.65 ± 4.88	83.41 ± 3.37	79.71 ± 5.90	99.71 ± 0.07	88.14 ± 2.14
DeepLabV3+ (VGG-19) [25]	25.996 M / 71.4 G	72.38 ± 3.44	83.94 ± 2.32	80.54 ± 5.11	99.70 ± 0.06	88.29 ± 1.43
DeepLabV3+ (ResNet-50) [25]	11.846 M / 23.3 G	72.95 ± 4.80	84.28 ± 3.29	82.44 ± 3.22	99.66 ± 0.09	86.71 ± 3.22
***U-TranSvision* (Ours)**	2.874 M / 7.4 G	**74.82 ± 3.06**	**85.57 ± 2.01**	**82.95 ± 3.60**	99.70 ± 0.09	88.63 ± 2.27

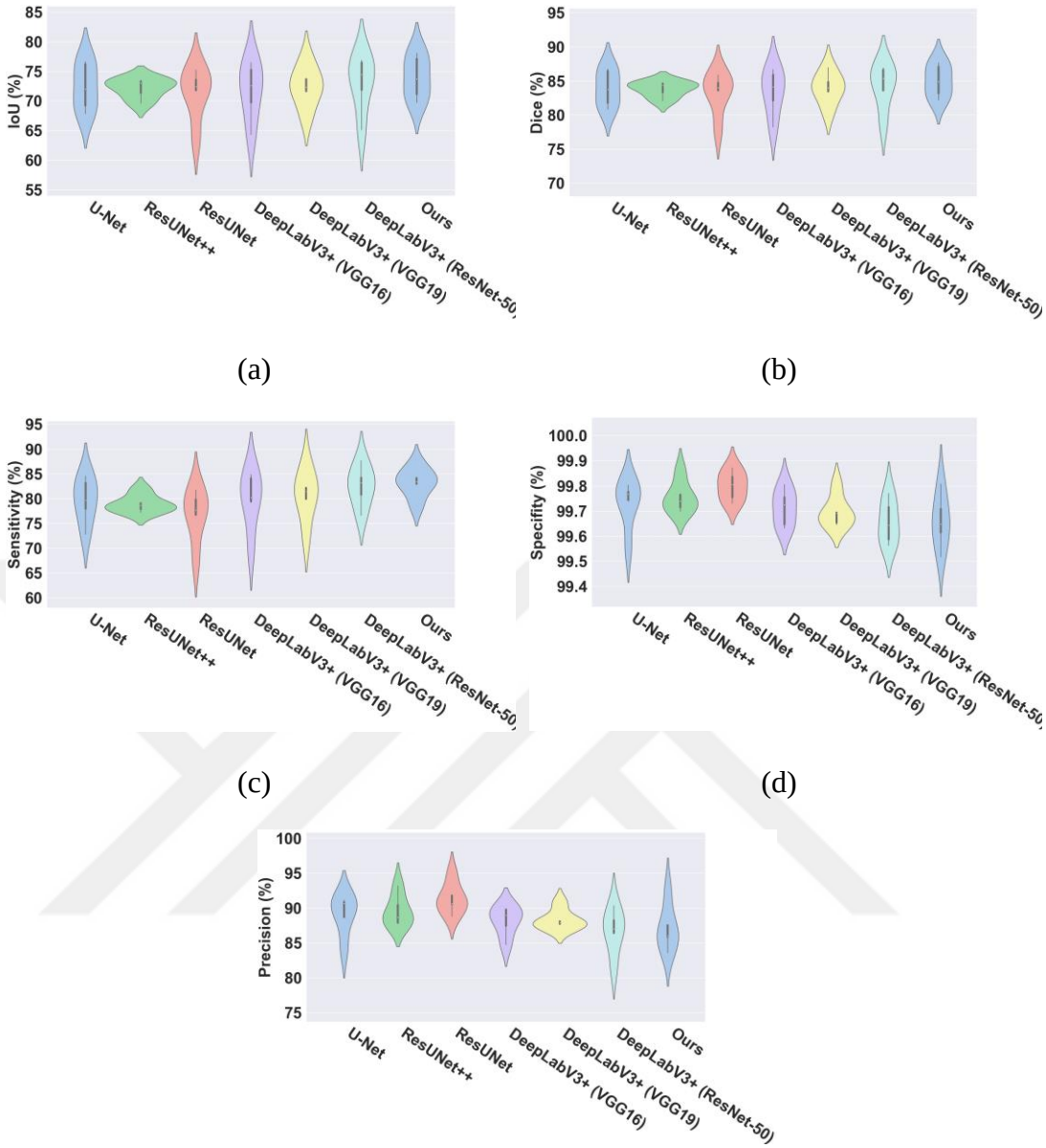


Figure 4.6: The architectures are listed on the horizontal axis, while the metrics are presented on the vertical axes including (a) IoU, (b) Dice, (c) Sensitivity, (d) Specificity, and (e) Precision.

of IoU, DSC, and precision. U-Net [54], CoLe-CNN+ [55], and Inf-Net [49] show poor performance compared with *U-TranSvision*, which can be attributed to their relatively small-scale datasets. Similarly, JCS [56] has relatively few labeled images for the segmentation of COVID-19 lesions, which may cause poor generalization ability, even though JCS uses a large classification dataset. In addition, Inf-Net [49], U-Net [54], and COPLE-Net [57] have 28.54 million, 33.12 million, and 10.52 million

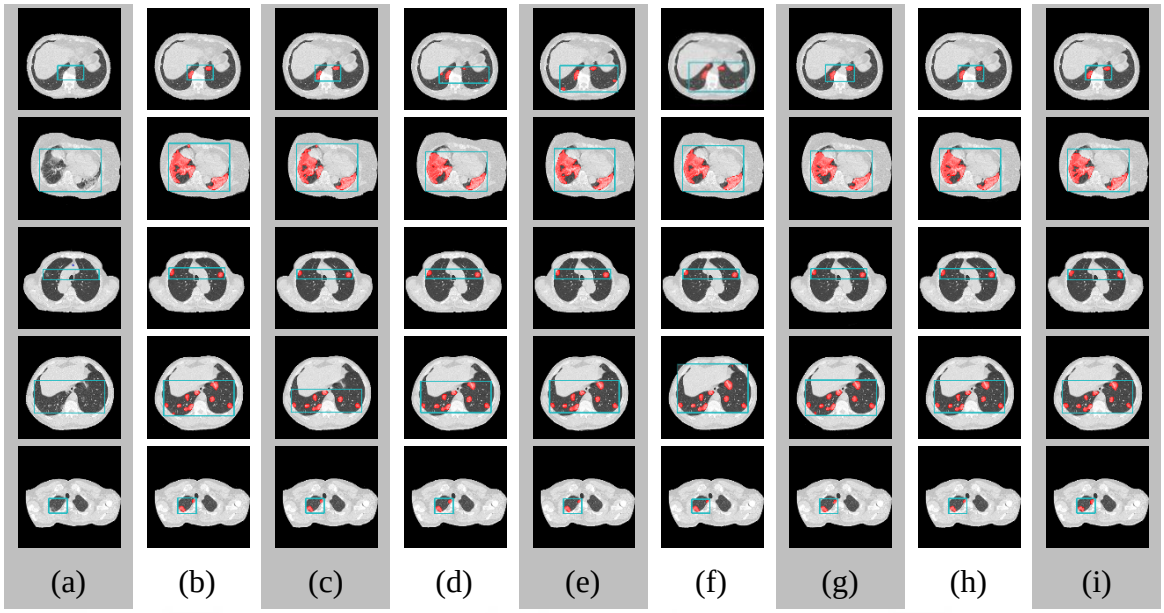


Figure 4.7: The qualitative results of COVID-19 lesion segmentation on axial CT images overlapped corresponding probability maps, including (a) the original image, (b) U-Net, (c) ResU-Net++, (d) ResUNet, (e) DeepLabV3+ (VGG-16), (f) DeepLabV3+ (VGG-19), (g) DeepLabV3+ (ResNet-50), (h) *U-TranSvision*, and (i) ground truth.

trainable parameters, respectively, indicating that these architectures require relatively large computational power.

Next, the comparison was then performed on publicly available datasets to demonstrate the robustness and accuracy of the proposed architecture, regardless of the dataset selection. In that sense, further experiments were performed on the COVID-19-CT-Seg, MosMedData, and MedSeg datasets, and our results were compared with the reported studies in Table 4.5, Table 4.6, and Table 4.7, respectively.

In Table 4.5, the relatively poor performance of U-Net-based architectures [58, 62] may stem from a deficiency within their structural configuration, namely the absence of a specialized component to prioritize salient features. Furthermore, the low performance of the studies [59, 61] can be attributed to the fact that they did not use any data augmentation approach for their training set.

In Table 4.6, the low performance in [63, 64] can be attributed to inadequate generalization ability for their model due to the absence of data augmentation approaches. Table 4.7 shows the performance comparison on MedSeg for several recent studies. The U-Net-based approaches [68, 71] have relatively low segmentation

Table 4.4: The performance comparison with further significant articles in the field (the results are given as mean ± standard deviation).

Method	**IoU (%)**	**DSC (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**
Dense GAN + U-Net [54]	n/a	68.30	69.80	n/a	**94.60**
CoLe-CNN+ [55]	61.30 ± 3.47	73.90 ± 3.13	76.80 ± 5.94	99.50 ± 0.17	77.50 ± 4.52
Inf-Net [49]	n/a	73.90	72.50	96.00	n/a
JCS [56]	66.40	78.50	n/a	n/a	n/a
COPLE-Net [57]	n/a	80.72 ± 9.96	n/a	n/a	n/a
Ours	**74.82 ± 3.06**	**85.57 ± 2.01**	**82.95 ± 3.60**	**99.70 ± 0.09**	88.63 ± 2.27

*n/a: Not available

Table 4.5: The performance comparison on COVID-19-CT-Seg (the results are given as mean $\pm$ standard deviation).

Method	**IoU (%)**	**DSC (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**
U-Net + few-shot learning [58]	n/a	67.30 $\pm$ 22.30	62.00 $\pm$ 23.70	**99.90 $\pm$ 15.90**	84.00 $\pm$ 19.60
SSA-Net [59]	n/a	65.22	n/a	n/a	n/a
Cov-TransNet [60]	67.30	80.30	77.20	99.50	84.20
DMDF-Net [61]	73.85 $\pm$ 7.59	81.99 $\pm$ 7.24	**90.51 $\pm$ 5.21**	98.95 $\pm$ 1.01	n/a
QC-HC U-Net [62]	n/a	85.31	83.60	99.96	n/a
Ours	**80.77 $\pm$ 8.74**	**89.15 $\pm$ 5.54**	89.04 $\pm$ 6.58	89.47 $\pm$ 5.13	**99.77 $\pm$ 0.08**

*n/a: Not available

Table 4.6: The performance comparison on MosMedData (the results are given as mean $\pm$ standard deviation).

Method	**IoU (%)**	**DSC (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**
The annotation-free pipeline [63]	n/a	58.40 ± 19.00	76.80 ± 16.7	97.70 ± 2.4	n/a
CovTANet [64]	51.70 ± 0.29	62.30 ± 0.18	$\mathbf{99.60 \pm 0.09}$	n/a	84.80 ± 0.26
Cov-TransNet [60]	60.70	75.10	80.00	**99.90**	71.70
The 3D CNN framework [65]	66.70 ± 2.39	80.10 ± 1.59	75.92 ± 1.95	n/a	$\mathbf{84.88 \pm 2.58}$
Contour-enhanced attention CNN [66]	71.51	83.71	84.58	99.75	82.43
Ours	$\mathbf{73.40 \pm 15.19}$	$\mathbf{83.89 \pm 10.97}$	84.76 ± 11.95	99.86 ± 10.18	$\mathbf{83.22 \pm 0.07}$

*n/a: Not available

Table 4.7: The performance comparison on MedSeg (the results are given as mean $\pm$ standard deviation).

Method	**IoU (%)**	**DSC (%)**	**Sensitivity (%)**	**Specificity (%)**	**Precision (%)**
SSInfNet [67]	32.00	n/a	82.00	n/a	38.00
U-Net + Multitask [68]	n/a	61 $\pm$ 2.00	n/a	n/a	n/a
LwMLA-NET [69]	n/a	76.90	76.50	n/a	77.10
FSS-2019-nCov [70]	n/a	79.80	80.30	98.60	n/a
TV-Unet [71]	n/a	86.30	86.70	n/a	85.90
Ours	**85.81 $\pm$ 5.46**	**92.30 $\pm$ 3.19**	**90.87 $\pm$ 3.88**	**99.85 $\pm$ 0.13**	**94.00 $\pm$ 3.33**

*n/a: Not available

performance since the models may be biased towards massive COVID-19 lesion regions on CT images compared to tiny lesions. FSS-2019-nCov [70] is based on a semi-supervised learning approach, which employs labeled and unlabeled data to segment COVID-19 regions on CT images. This architecture also uses weakly supervised output probability maps, which may not be as accurate as direct supervision, resulting in lower model performance.

Finally, *U-TranSvision* differs substantially from state-of-the-art architectures with respect to the evaluation metrics and the number of trainable parameters. Cov-TransNet [60] has 18.987 million, while DMDF-Net [61] has approximately 5.85 million parameters, which increases the computational complexity for COVID-19 lesion segmentation. Besides, the architecture [65] contains about 11 million learning parameters, whereas the framework [66] has 30 million. These can be considered reasonable, but their model complexity is relatively high compared to *U-TranSvision*, which may lead to limited use in medical applications because of hardware limitations.

4.5 Chapter Summary

Here, a novel architecture called *U-TranSvision* is proposed for COVID-19 lesion segmentation utilizing a U-shaped structure with the transformer layer and the deep supervision approach, offering to detection of small-size lesions on CT images, faster convergence, and better segmentation performance. Pix2Pix GAN was employed to improve the predictive performance of *U-TranSvision* for data augmentation. The pre-processing steps were also applied to mitigate the negative effect of possible noise outside the lung border on the segmentation performance. Besides, *U-TranSvision* benefited from a hybrid loss function to mitigate the class imbalance problem. Moreover, we collected a relatively large-scale dataset to create a model with higher generalization ability. The CNN-based architectures were trained with our dataset created for COVID-19 lesion segmentation due to the scarcity of a relatively large public dataset. The performance of *U-TranSvision* on our dataset in terms of DSC and IoU was achieved at 85.57 % and 74.82 % for the lesion segmentation, respectively. The *U-TranSvision* was also tested on three publicly available datasets, and the experimental results show the advantage of our proposed architecture over the state-of-the-art for COVID-19 lesion segmentation.

Chapter 5

Artificial Intelligence in Healthcare: Ischemic Stroke Segmentation

So far, the thesis has investigated pixel-wise classification for various tasks such as lung, pulmonary lobe, and COVID-19 lesion segmentation on chest CT images. Now, this chapter covers an application of AI in healthcare, specifically for ischemic stroke segmentation on brain CT images. In this context, stroke is brain cell death because of either lack of blood flow (ischemic) or bleeding (hemorrhagic) that prevents the brain from functioning properly in both conditions. Ischemic stroke is a common type of stroke caused by a blockage in the cerebrovascular system that prevents blood from flowing to brain regions and directly blocks blood vessels. CT scanning is frequently used in the evaluation of stroke, and rapid and accurate diagnosis of ischemic stroke with CT images is critical for determining the appropriate treatment. The manual diagnosis of ischemic stroke can be error-prone due to several factors, such as the busy schedules of specialists and the large number of patients admitted to healthcare facilities. Therefore, in this paper, a deep learning-based interface was developed to automatically diagnose the ischemic stroke through segmentation on CT images leading to a reduction in the diagnosis time and workload of specialists. CNNs allow automatic feature extraction in ischemic stroke segmentation, utilized to mark the disease regions from CT images. CNN-based architectures, such as U-Net, U-Net VGG16, U-Net VGG19, Attention U-Net, and ResUNet, were used to benchmark the ischemic stroke disease segmentation. To further improve the segmentation performance, ResUNet was modified, adding a dilation convolution layer after the last layer of the architecture. In addition, data augmentation was performed to increase the number of images in the dataset, including the ground truths for the ischemic stroke disease region. Based on the experimental results, our modified ResUNet with a dilation convolution provides the highest performance for ischemic stroke segmentation in DSC and IoU with 98.45 % and 96.95 %, respectively. The

experimental results show that our modified ResUNet outperforms the state-of-the-art approaches for ischemic stroke disease segmentation.

Section 5.1 introduces a segmentation problem in the context of ischemic stroke. This chapter follows with our modified ResUNet using a dilation convolution in Section 5.2. The dataset and pre-processing steps are explained in Section 5.3. Next, the experimental results are presented and discussed in Section 5.4. Finally, the chapter concludes in Section 5.5, including a summary of ischemic stroke segmentation.

5.1 Introduction

Ischemic stroke segmentation, which refers to a pixel-wise classification of ROI on medical brain images, is critical for effective diagnosis. In recent years, with the development of technology and theoretical background, AI-based approaches have shown promising performance in ischemic stroke segmentation [74, 75]. Recent studies [101, 102] attempt to mitigate the overfitting problem in their models due to the lack of large-scale datasets. To address this problem, data augmentation approaches such as image processing and GAN-based methods presented in Chapter 2 are helpful for ischemic stroke segmentation. In addition, CNN-based architectures can employ various structures, such as dilation convolution, pooling layers, residual connections, and skip connections, to further improve segmentation performance. Unlike standard convolution, dilation convolution has space between pixels in its filter or kernel, while pooling layers downsamples feature maps. Residual connections allow adding output from a previous layer to the output of the current layer, while skip connections add feature maps to the following layers. ResUNet, which we examined in this chapter, contains already pooling layers, residual connections, and skip connections. ResUNet does not include a dilation convolution, so it was employed in ResUNet to capture larger-scale contextual information in an input image.

In this chapter, the publicly available dataset [103] has been utilized for ischemic stroke disease segmentation. However, the dataset, including the brain CT images and the corresponding masks on which the disease region is marked, is relatively small for training CNN architectures. Therefore, several data augmentation approaches, such as Pix2Pix GAN, varying brightness, applying Gaussian and blur filters, and adding salt

& pepper and Gauss noise, were used to increase the number of images in the dataset. These augmentation approaches reduce the overfitting problem in small-scale datasets and increase the diversity of images, improving the generalization ability of CNN architectures. After dataset augmentation, several CNN architectures were trained, including ResUNet modified with a dilation convolution to improve the segmentation performance of ischemic stroke disease.

5.2 ResUNet with a Dilation Convolution

A dilation convolution has been added to the ResUNet to improve segmentation performance. Dilation convolution utilizes a dilation rate parameter, which controls the spacing between the values in the kernel. The dilation convolution expands the receptive field in each kernel without increasing the parameters, thereby allowing for larger contextual feature extraction from the feature maps. Dilation convolution is particularly useful in tasks such as semantic segmentation and object detection, where preserving the spatial context of features is essential. Therefore, convolution with a dilation rate of 10 has been used at the last layer of ResUNet to capture large contextual features of the feature maps.

5.3 Data Acquisition and Pre-processing

In deep learning, the dataset is essential to make the model robust for previously unseen images. The publicly available dataset [103], including 1,130 brain CT images with ischemic stroke in digital imaging and communications in medicine (DICOM) format, was employed to perform segmentation for the disease diagnosis. Data augmentation, which creates modified versions of existing data samples, was performed on the dataset with limited training images to improve the robustness and generalization ability of CNN architectures. The number of images in the dataset was doubled using the Pix2Pix GAN architecture for data augmentation. In addition to this approach, basic image processing approaches, such as varying brightness, applying Gaussian and blur filters, and adding salt & pepper and Gauss noise, were applied to the dataset. After the augmentation, the dataset was enlarged to 5,628 images for the training of CNN architectures. The images in the augmented dataset have been resized from a resolution of 512×512 to 256×256after cropping them without disturbing

Table 5.1: The number and percent of the images in the train, validation, and test sets.

Dataset	Train	Validation	Test
Number	3601	1127	900
Percent	64	16	20

the brain region. Next, the images were normalized using the min-max scaler [92]. Finally, the dataset has been divided into the train, validation, and test sets for ischemic stroke segmentation, as presented in Table 5.1.

5.4 Results and Discussion

The CNN-based architectures, including U-Net, ResUNet, U-Net (backbone VGG-16), U-Net (backbone VGG-19), and ResUNet with a dilation convolution and attention U-Net, were trained and tested on the dataset as mentioned in Section 5.3. The experiments for the CNN architectures were performed in a Python environment using TensorFlow with the parameters Adam optimizer, learning rate=0.0005, he-normal kernel initializer, batch size=32, and the sum of dice and focal losses as the loss function [104]. The CNN architectures showed higher performance with these parameters based on the experiments. In addition, batch normalization was used instead of dropout in the CNN architectures to further improve the robustness of the model.

The results of the CNN architectures for the ischemic stroke disease segmentation with respect to DSC and IoU are listed in Table 5.2. Based on the evaluation results, ResUNet with dilation convolution outperforms other architectures in terms of DSC and IoU scores, achieving 98.45 % and 96.45 %, respectively. The superior performance of ResUNet with a dilation convolution can be attributed to its use of dilation convolution, which allows the extraction of larger contextual features from the images. The fact that Attention U-Net can automatically focus on ischemic stroke regions in brain images without additional supervision lead to showing the second highest score. The U-Net with VGG-16 architecture achieved the third highest result due to its shallower depth compared to the U-Net (VGG-19). Next, U-Net (VGG-19) followed ResUNet since the residual connections in ResUNet deal with the problem

Table 5.2: The empirical results of the ischemic stroke disease segmentation.

Method	**IoU (%)**	**DSC (%)**
U-Net	86.16	92.41
U-Net (VGG-19)	88.44	93.80
ResUNet	88.94	94.10
U-Net (VGG-16)	91.25	95.41
Attention U-Net	94.77	97.30
ResUNet with a dilation convolution	**96.95**	**98.45**

of vanishing or exploding gradients observed in deep layers. Finally, U-Net shows the poorest results for ischemic stroke segmentation due to the gradient vanishing or exploding gradient problems.

In addition, the modified ResUNet with a dilation convolution architecture was compared with the latest studies on the segmentation of ischemic stroke disease with respect to the DSC and IoU values in Table 5.3. The model in [72] is unsuitable for embedding in a desktop application that can assist specialists since the architecture requires more computation power, causing more time consumption. The fact that the approaches in [73, 74] use relatively small-scale datasets for ischemic stroke disease segmentation; causes the poor generalization ability of the model. The method in [75] is time-consuming and error-prone due to the image registration approach in the atlas-based ischemic stroke disease segmentation.

Table 5.3: The comparison of our approach with the state-of-the-art architectures for ischemic stroke disease segmentation.

Method	**IoU (%)**	**DSC (%)**
PSPF [72]	-	75.40
DRANet [73]	-	76.39
CSNet [74]	-	82.14
FCN [75]	-	91.86
Tsallis Entropy [76]	-	95.18
Our approach	**96.95**	**98.45**

5.5 Chapter Summary

In this chapter, CNN architectures were compared using the evaluation metrics to segment ischemic stroke disease on brain images. A public dataset with labeled ischemic stroke masks was used to train CNN architectures, including U-Net, ResUNet, U-Net VGG-16, U-Net VGG-19, Attention U-Net, and ResUNet with dilation convolution. The modified ResUNet with a dilation convolution achieved the highest segmentation performance with a 98.45 % DSC and a 96.95 % IoU score on the test set. Compared to the state-of-the-art methods, the experiments showed that the modified architecture had an advantage in detecting ischemic stroke disease.

Chapter 6

Conclusions and Future Research

So far, we have investigated several medical segmentation problems, including lung, pulmonary lobe, COVID-19 lesions, and ischemic stroke. This chapter briefly summarizes our significant contributions, such as DeepChestNet, *U-TranSvision*, and ischemic stroke segmentation. Furthermore, suggestions for future studies are presented to help other researchers in medical image segmentation.

6.1 Conclusions

This thesis examines the technical challenges and possible solutions associated with segmentation tasks in medical imaging to provide a perspective for further studies. Three main contributions of the thesis have been discussed in Chapter 3, 4, and 5, respectively: (1) the fact that our DeepChestNet framework performs lung, pulmonary lobe, and lesion segmentation provides an increase in model performance, (2) our *U-TranSvision* architecture focuses on salient features of the images and minimizes the class imbalance problem in the dataset, which leads to an improved generalization ability for the model, (3) the modified ResUNet with a dilation convolution helped to increase the performance in ischemic stroke segmentation. These contributions are briefly summarized below.

6.1.1 *DeepChestNet*: Artificial Intelligence Approach for COVID-19 Detection on Computed Tomography Images

The combination of the lung, pulmonary lobe, and COVID-19 lesion segmentation for COVID-19 detection can provide healthcare professionals with greater insight into disease progression. AI approaches are employed to perform these segmentations, especially CNN architectures, which are beneficial for COVID-19 detection. Moreover, CNN-based architectures used for lung, pulmonary lobe, and COVID-19

segmentation in medical imaging suffer from the complex structures associated with COVID-19 lesion regions due to the nature of their convolutions in two-dimensional images. The texture and shape of the lesions may be confused with other structures that appear in the foreground or background of the image. To alleviate these problems, we propose a novel framework called DeepChestNet, which combines the lung, pulmonary lobe, and COVID-19 lesion segmentation.

Our DeepChestNet framework is proposed that leverages structured relationships by jointly performing segmentation and classification on the lung, pulmonary lobe, and lesion to detect COVID-19 with findings. DeepChestNet employs images overlapped with segmented lung masks and their corresponding original images to improve the quality of pulmonary lobe segmentation. Besides, COVID-19 lesion segmentation is performed in parallel with lung and pulmonary lobe segmentation. Finally, the outputs of the COVID-19 segmentation are combined with the contextual information generated by pulmonary lobe segmentation, which allows the diagnosis of the findings, such as bilateral, multifocal, and multilobar lesions. Furthermore, new large-scale *DeepChestNet-Lung*, *DeepChestNet-Lobe*, and *DeepChestNet-COVID* datasets, including 9036, 9036, and 1034 axial chest CT images, were annotated with pixel-level labels, are created to further improve the generalization ability of our models for COVID-19 detection. Experimental results on *DeepChestNet-Lung*, *DeepChestNet-Lobe*, and *DeepChestNet-COVID* demonstrate that DeepChestNet achieves promising results in COVID-19 detection compared with several state-of-the-art approaches.

The DeepChestNet framework differs significantly from similar studies on the diagnosis of COVID-19 since it combines the lung, pulmonary lobe, and lesion segmentation, which improves segmentation performance and provides more findings.

6.1.2 *U-TranSvision*: Transformer-based Deep Supervision Approach for COVID-19 Lesion Segmentation on Computed Tomography Images

In COVID-19 lesion segmentation, the CNN-based approaches have several limitations, including the complexity of the COVID-19 characteristics, low or high image contrast, and small infected regions, which cause the class imbalance problem.

To minimize these limitations, there is a need to use an advanced approach like transformers, which can focus on the salient features of small COVID-19 lesions. In addition, a deep supervision approach can also contribute to the predictive performance of a model. To apply these approaches, we propose a new architecture called *U-TranSvision* that takes advantage of both approaches and investigate how these approaches affect the performance of our model.

U-TranSvision employs transformations and a deep supervision approach to improve predictive performance by capturing the dependencies between pixels in an input image. Our approach using the hybrid loss function minimizes the class imbalance problem for COVID-19 lesion segmentation. In addition, we have expanded our dataset in Chapter 3 to a relatively large dataset, including 11,717 axial CT images and their ground truths. The images in our expanded dataset have applied the pre-processing steps to eliminate noises around human tissue. *U-TranSvision* also performed data augmentation using Pix2Pix GAN to further improve the generalizability ability of the model. The ablation studies and experiments on our dataset revealed that *U-TranSvision* outperforms the CNN-based architectures for COVID-19 lesion segmentation. The experiments on three public datasets using *U-TranSvision* were also performed to further prove the robustness of our model and showed promising results.

The use of transformers and a deep supervision approach together in the *U-TranSvision* architecture makes our model more robust. In addition, our approach performs better on previously unseen images and leads to less trainable parameters than several state-of-the-art architectures.

6.1.3 Artificial Intelligence in Healthcare: Ischemic Stroke Segmentation

The CNN-based architectures can show low performance for ischemic stroke segmentation when the training set of datasets is small-scale, which results in poor generalization ability on the testing set. To minimize this issue, data augmentation approaches can be beneficial for segmentation tasks. Moreover, CNN-based architectures can be modified in their hyperparameters and added new layers to

improve the predictive performance. In this context, we modified the ResUNet architecture by adding a dilation convolution for ischemic stroke segmentation.

The Pix2Pix GAN architecture and basic image processing approaches were employed to augment the public dataset for ischemic stroke segmentation, improving the predictive performance of our segmentation model. Eight synthetic images were generated from each image in the training set of the dataset. In addition, a dilation convolution was added to the last layer of ResUNet to increase the receptive field in the convolutional layer. The dilation convolution allows ResUNet to capture more global contextual information while preserving fine-grained details in the probability output of the architecture. Based on the experimental results, ResUNet with a dilation convolution performs better than similar studies.

In conclusion, the Pix2Pix GAN architecture is helpful for better generalization ability of our model. Besides, we added a dilation convolution into the ResUNet architecture to further improve the performance of ischemic stroke segmentation.

6.2 Future Research

Although our contributions are valuable for segmentation tasks using medical images, the approaches in this thesis have several problems and limitations. In this section, we would like to point out the limitations of the thesis and its drawbacks. In addition, the following includes possible future studies for our approaches to help similar works.

- The DeepChestNet framework for COVID-19 detection assumes that the images in the *DeepChestNet-COVID* have a ratio of lesion region-to-background ratio greater than 5 % to ignore the class imbalance problem. Therefore, further improvements are needed to detect small infected regions for COVID-19 lesion segmentation. In addition, we attempted to minimize the class imbalance in our extended dataset using *U-TranSvision* in Chapter 4. However, the experiments were not performed in the DeepChestNet framework, which uses *U-TranSvision* for COVID-19 lesion segmentation on the extended dataset. Therefore, the future direction of this thesis could be to combine the *U-TranSvision* architecture with the DeepChestNet framework.

- Our DeepChestNet framework and *U-TranSvision* architecture have been proposed for COVID-19 lesion segmentation in chest CT images. Moreover, we have modified the ResUNet architecture using a dilation convolution to segment ischemic stroke in brain CT images. On the other hand, these approaches could be extended to other lung or brain diseases, such as lung cancer, tuberculosis, pneumonia, traumatic brain injury, cerebral aneurysms, and brain tumors, which can be interesting directions for future studies. To investigate the efficient way of segmenting these diseases on medical images, our approaches could also be tried in various configurations, such as DeepChestNet and *U-TranSvision* or DeepChestNet and ResUNet with a dilation convolution.
- In Chapter 3 and Chapter 4, we collected relatively large datasets for lung, pulmonary lobe, and COVID-19 lesion segmentation from one hospital in Izmir. Therefore, our datasets may be biased toward characteristics of infected regions in CT images collected only from this hospital. Our datasets have little diversity regarding patient factors, COVID-19 lesion types, and scanner parameters or acquisition protocols. Thus, for the future direction of the chapters, our datasets could be expanded to include CT images collected from hospitals in various regions worldwide.
- Our *U-TranSvision* architecture employs only two transformer layers for COVID-19 lesion segmentation due to their computational cost. To minimize this cost, our architecture could be optimized in future studies. Besides, we would like to recommend a future study for this thesis in the context of medical segmentation. In this thesis, the CT imaging modality was employed for segmentation tasks due to its lower response time and resource requirements than MRI. However, MRI can be beneficial for detecting early changes of ischemic stroke. It also provides detailed information about the location, size, and severity of ischemic stroke. Therefore, future studies may include MRI modality to detect even tiny ischemic stroke regions in the early stage.

References

[1] Z. Sidiq, M. Hanif, K. K. Dwivedi, and K. Chopra, Benefits and limitations of serological assays in COVID-19 infection, Indian Journal of Tuberculosis, vol. 67, no. 4, pp. S163-S166, 2020.

[2] M. Teymouri, S. Mollazadeh, H. Mortazavi, Z. N. Ghale-Noie, V. Keyvani et al., Recent advances and challenges of RT-PCR tests for the diagnosis of COVID-19, Pathology-Research Practice, vol. 221, p. 153443, 2021.

[3] B. Katipoğlu, L. Ö. Sönmez, H. Vatansev, N. Yüce, M. Sabak et al., Can hematological and biochemical parameters fasten the diagnosis of COVID-19 in emergency departments?, Disaster Emergency Medicine Journal, vol. 5, no. 4, pp. 175-181, 2020.

[4] Y. Wang, H. Hou, W. Wang, and W. Wang, Combination of CT and RT-PCR in the screening or diagnosis of COVID-19, 2020.

[5] P. M. Leth, The use of CT scanning in forensic autopsy, Forensic Science, Medicine, Pathology, vol. 3, pp. 65-69, 2007.

[6] S. C. Harvey, P. A. Di Carlo, B. Lee, E. Obadina, D. Sippo et al., An abbreviated protocol for high-risk screening breast MRI saves time and resources, Journal of the American College of Radiology, vol. 13, no. 11, pp. R74-R80, 2016.

[7] J. Zhang, S. L. Chua, and B. L. Khoo, Worm-based microfluidic biosensor for real-time assessment of the metastatic status, Cancers, vol. 13, no. 4, p. 873, 2021.

[8] Y. Fang, H. Zhang, J. Xie, M. Lin, L. Ying et al., Sensitivity of chest CT for COVID-19: comparison to RT-PCR, Radiology, 2020.

[9] X. Xie, Z. Zhong, W. Zhao, C. Zheng, F. Wang et al., Chest CT for typical coronavirus disease 2019 (COVID-19) pneumonia: relationship to negative RT-PCR testing, Radiology, vol. 296, no. 2, pp. E41-E45, 2020.

[10] S. Khezrpour, H. Seyedarabi, S. N. Razavi, and M. Farhoudi, Automatic segmentation of the brain stroke lesions from MR flair scans using improved U-net framework, Biomedical Signal Processing Control, vol. 78, p. 103978, 2022.

[11] H. Kirshner and M. Schrag, Management of intracerebral hemorrhage: update and future therapies, Current Neurology Neuroscience Reports, vol. 21, pp. 1-5, 2021.

[12] R. Karthik, R. Menaka, A. Johnson, and S. Anand, Neuroimaging and deep learning for brain stroke detection-A review of recent advancements and future prospects, Computer Methods Programs in Biomedicine, vol. 197, p. 105728, 2020.

[13] F. Shi, J. Wang, J. Shi, Z. Wu, Q. Wang et al., Review of artificial intelligence techniques in imaging data acquisition, segmentation, and diagnosis for COVID-19, IEEE Reviews in Biomedical Engineering, vol. 14, pp. 4-15, 2020.

[14] A. Tursynova, B. Omarov, A. Sakhipov, and N. Tukenova, Brain Stroke Lesion Segmentation Using Computed Tomography Images based on Modified U-Net Model with ResNet Blocks, International Journal of Online Biomedical Engineering, vol. 18, no. 13, 2022.

[15] M. Şen, E. Yüzer, V. Doğan, İ. Avcı, K. Ensarioğlu et al., Colorimetric detection of H2O2 with Fe3O4@ Chi nanozyme modified µPADs using artificial intelligence, Microchimica Acta, vol. 189, no. 10, p. 373, 2022.

[16] V. Doğan, T. Isık, V. Kılıç, and N. Horzum, A field-deployable water quality monitoring with machine learning-based smartphone colorimetry, Analytical Methods, vol. 14, no. 35, pp. 3458-3466, 2022.

[17] E. Yüzer, V. Doğan, V. Kılıç, and M. Şen, Smartphone embedded deep learning approach for highly accurate and automated colorimetric lactate analysis in sweat, Sensors Actuators B: Chemical, vol. 371, p. 132489, 2022.

[18] K. B. Girum, G. Créhange, R. Hussain, and A. Lalande, Fast interactive medical image segmentation with weakly supervised deep learning method, International Journal of Computer Assisted Radiology Surgery, vol. 15, pp. 1437-1444, 2020.

[19] K. Suzuki, Overview of deep learning in medical imaging, Radiological Physics Technology, vol. 10, no. 3, pp. 257-273, 2017.

[20] J. Tian, Y. Liu, W. Zheng, and L. Yin, Smog prediction based on the deep belief-BP neural network model (DBN-BP), Urban Climate, vol. 41, p. 101078, 2022.

[21] Ö. B. Mercan and V. Kılıç, Deep Learning based Colorimetric Classification of Glucose with Au-Ag nanoparticles using Smartphone, in 2020 Medical Technologies Congress (TIPTEKNO), 2020, pp. 1-4: IEEE.

[22] O. Ronneberger, P. Fischer, and T. Brox, U-net: Convolutional networks for biomedical image segmentation, in International Conference on Medical Mmage Computing and Computer-assisted Intervention, 2015, pp. 234-241: Springer.

[23] D. Jha, P. H. Smedsrud, M. A. Riegler, D. Johansen, T. De Lange et al., Resunet++: An advanced architecture for medical image segmentation, in 2019 IEEE International Symposium on Multimedia (ISM), 2019, pp. 225-2255: IEEE.

[24] Z. Zhang, Q. Liu, and Y. Wang, Road extraction by deep residual u-net, IEEE Geoscience Remote Sensing Letters, vol. 15, no. 5, pp. 749-753, 2018.

[25] S. Das, A. A. Fime, N. Siddique, and M. Hashem, Estimation of road boundary for intelligent vehicles based on deepLabV3+ architecture, IEEE Access, vol. 9, pp. 121060-121075, 2021.

[26] R. Ghali, M. A. Akhloufi, and W. S. Mseddi, Deep learning and transformer approaches for UAV-based wildfire detection and segmentation, Sensors, vol. 22, no. 5, p. 1977, 2022.

[27] B. Palsson, J. R. Sveinsson, and M. O. Ulfarsson, Blind hyperspectral unmixing using autoencoders: A critical comparison, IEEE Journal of Selected Topics in Applied Earth Observations Remote Sensing, vol. 15, pp. 1340-1372, 2022.

[28] J. Li, J. Chen, Y. Tang, C. Wang, B. A. Landman et al., Transforming medical imaging with Transformers? A comparative review of key properties, current progresses, and future perspectives, Medical Image Analysis, p. 102762, 2023.

[29] A. A. Ardakani, A. R. Kanafi, U. R. Acharya, N. Khadem, and A. Mohammadi, Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: Results of 10 convolutional neural networks, Computers in Biology Medicine, vol. 121, p. 103795, 2020.

[30] A. Mobiny, P. Yuan, P. A. Cicalese, and H. Van Nguyen, Decaps: Detail-oriented capsule networks, in Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part I 23, 2020, pp. 148-158: Springer.

[31] S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao et al., A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19), European Radiology, vol. 31, pp. 6096-6104, 2021.

[32] T. Pang, S. Guo, X. Zhang, and L. Zhao, Automatic lung segmentation based on texture and deep features of HRCT images with interstitial lung disease, BioMed Research International, vol. 2019, 2019.

[33] P. L. Vidal, J. de Moura, J. Novo, and M. Ortega, Multi-stage transfer learning for lung segmentation using portable X-ray devices for patients with COVID-19, Expert Systems with Applications, vol. 173, p. 114677, 2021.

[34] A. Mansoor, U. Bagci, Z. Xu, B. Foster, K. N. Olivier et al., A generic approach to pathological lung segmentation, IEEE Transactions on Medical Imaging, vol. 33, no. 12, pp. 2293-2310, 2014.

[35] G. Gill and R. R. Beichel, An approach for reducing the error rate in automated lung segmentation, Computers in Biology Medicine, vol. 76, pp. 143-153, 2016.

[36] C. Zhao, Y. Xu, Z. He, J. Tang, Y. Zhang et al., Lung segmentation and automatic detection of COVID-19 using radiomic features from chest CT images, Pattern Recognition, vol. 119, p. 108071, 2021.

[37] S. E. Gerard, J. Herrmann, Y. Xin, K. T. Martin, E. Rezoagli et al., CT image segmentation for inflamed and fibrotic lungs using a multi-resolution convolutional neural network, Scientific Reports, vol. 11, no. 1, pp. 1-12, 2021.

[38] S. Simpson, F. U. Kay, S. Abbara, S. Bhalla, J. H. Chung et al., Radiological Society of North America expert consensus statement on reporting chest CT findings related to COVID-19. Endorsed by the Society of Thoracic Radiology, the American College of Radiology, and RSNA, Journal of Thoracic Imaging, 2020.

[39] K. He, W. Zhao, X. Xie, W. Ji, M. Liu et al., Synergistic learning of lung lobe segmentation and hierarchical multi-instance classification for automated severity assessment of COVID-19 in CT images, Pattern Recognition, vol. 113, p. 107828, 2021.

[40] F. T. Ferreira, P. Sousa, A. Galdran, M. R. Sousa, and A. Campilho, End-to-end supervised lung lobe segmentation, in 2018 International Joint Conference on Neural Networks (IJCNN), 2018, pp. 1-8: IEEE.

[41] H. Tang, C. Zhang, and X. Xie, Automatic pulmonary lobe segmentation using deep learning, in 2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019), 2019, pp. 1225-1228: IEEE.

[42] Y. Song and H. Yan, Image segmentation techniques overview, in 2017 Asia Modelling Symposium (AMS), 2017, pp. 103-107: IEEE.

[43] A.-A.-Z. Imran, A. Hatamizadeh, S. P. Ananth, X. Ding, N. Tajbakhsh et al., Fast and automatic segmentation of pulmonary lobes from chest CT using a progressive dense V-network, Computer Methods in Biomechanics Biomedical Engineering: Imaging Visualon, vol. 8, no. 5, pp. 509-518, 2020.

[44] H. Pang, Y. Wu, S. Qi, C. Li, J. Shen et al., A fully automatic segmentation pipeline of pulmonary lobes before and after lobectomy from computed tomography images, Computers in Biology Medicine, vol. 147, p. 105792, 2022.

[45] J. Park, J. Yun, N. Kim, B. Park, Y. Cho et al., Fully automated lung lobe segmentation in volumetric chest CT with 3D U-Net: validation with intra-and extra-datasets, Journal of Digital Imaging, vol. 33, pp. 221-230, 2020.

[46] W. Xie, C. Jacobs, J.-P. Charbonnier, and B. Van Ginneken, Relational modeling for robust and efficient pulmonary lobe segmentation in CT scans, IEEE Transactions on Medical Imaging, vol. 39, no. 8, pp. 2664-2675, 2020.

[47] J. Ma, Y. Wang, X. An, C. Ge, Z. Yu et al., Towards efficient covid-19 ct annotation: A benchmark for lung and infection segmentation, 2020.

[48] A. Oulefki, S. Agaian, T. Trongtirakul, and A. K. Laouar, Automatic COVID-19 lung infected region segmentation and measurement using CT-scans images, Pattern Recognition, vol. 114, p. 107747, 2021.

[49] D.-P. Fan, T. Zhou, G.-P. Ji, Y. Zhou, G. Chen et al., Inf-net: Automatic covid-19 lung infection segmentation from ct images, IEEE Transactions on Medical Imaging, vol. 39, no. 8, pp. 2626-2637, 2020.

[50] L. Zhou, Z. Li, J. Zhou, H. Li, Y. Chen et al., A rapid, accurate and machine-agnostic segmentation and quantification method for CT-based COVID-19 diagnosis, IEEE Transactions on Medical Imaging, vol. 39, no. 8, pp. 2638-2652, 2020.

[51] T. Zhou, S. Canu, and S. Ruan, Automatic COVID-19 CT segmentation using U-Net integrated spatial and channel attention mechanism, International Journal of Imaging Systems Technology, vol. 31, no. 1, pp. 16-27, 2021.

[52] A. Amyar, R. Modzelewski, H. Li, and S. Ruan, Multi-task deep learning based CT imaging analysis for COVID-19 pneumonia: Classification and segmentation, Computers in Biology Medicine, vol. 126, p. 104037, 2020.

[53] X. Chen, L. Yao, and Y. Zhang, Residual attention u-net for automated multi-class segmentation of covid-19 chest ct images, arXiv preprint arXiv:.05645, 2020.

[54] J. Zhang, L. Yu, D. Chen, W. Pan, C. Shi et al., Dense GAN and multi-layer attention based lesion segmentation method for COVID-19 CT images, Biomedical Signal Processing Control, vol. 69, p. 102901, 2021.

[55] G. Pezzano, O. Díaz, V. R. Ripoll, and P. Radeva, CoLe-CNN+: Context learning-Convolutional neural network for COVID-19-Ground-Glass-Opacities detection and segmentation, Computers in Biology Medicine, vol. 136, p. 104689, 2021.

[56] Y.-H. Wu, S.-H. Gao, J. Mei, J. Xu, D.-P. Fan et al., Jcs: An explainable covid-19 diagnosis system by joint classification and segmentation, IEEE Transactions on Image Processing, vol. 30, pp. 3113-3126, 2021.

[57] G. Wang, X. Liu, C. Li, Z. Xu, J. Ruan et al., A noise-robust framework for automatic segmentation of COVID-19 pneumonia lesions from CT images, IEEE Transactions on Medical Imaging, vol. 39, no. 8, pp. 2653-2663, 2020.

[58] J. Ma, Y. Wang, X. An, C. Ge, Z. Yu et al., Toward data-efficient learning: A benchmark for COVID-19 CT lung and infection segmentation, Medical Physics, vol. 48, no. 3, pp. 1197-1210, 2021.

[59] X. Wang, Y. Yuan, D. Guo, X. Huang, Y. Cui et al., SSA-Net: Spatial self-attention network for COVID-19 pneumonia infection segmentation with semi-supervised few-shot learning, Medical Image Analysis, vol. 79, p. 102459, 2022.

[60] Y. Peng, T. Zhang, and Y. Guo, Cov-TransNet: Dual branch fusion network with transformer for COVID-19 infection segmentation, Biomedical Signal Processing Control, vol. 80, p. 104366, 2023.

[61] M. Owais, N. R. Baek, and K. R. Park, DMDF-Net: Dual multiscale dilated fusion network for accurate segmentation of lesions related to COVID-19 in lung radiographic scans, Expert Systems with Applications, vol. 202, p. 117360, 2022.

[62] Q. Zhang, X. Ren, and B. Wei, Segmentation of infected region in CT images of COVID-19 patients based on QC-HC U-net, Scientific Reports, vol. 11, no. 1, p. 22854, 2021.

[63] S. Shabani, M. Homayounfar, V. Vardhanabhuti, M.-A. N. Mahani, and M. Koohi-Moghadam, Self-supervised region-aware segmentation of COVID-19 CT images using 3D GAN and contrastive learning, Computers in Biology Medicine, vol. 149, p. 106033, 2022.

[64] T. Mahmud, M. J. Alam, S. Chowdhury, S. N. Ali, M. M. Rahman et al., CovTANet: a hybrid tri-level attention-based network for lesion segmentation, diagnosis, and severity prediction of COVID-19 chest CT scans, IEEE Transactions on Industrial Informatics, vol. 17, no. 9, pp. 6489-6498, 2020.

[65] R. Karthik, R. Menaka, M. Hariharan, and D. Won, CT-based severity assessment for COVID-19 using weakly supervised non-local CNN, Applied Soft Computing, vol. 121, p. 108765, 2022.

[66] R. Karthik, R. Menaka, M. Hariharan, and D. Won, Contour-enhanced attention CNN for CT-based COVID-19 segmentation, Pattern Recognition, vol. 125, p. 108538, 2022.

[67] D. L. Fung, Q. Liu, J. Zammit, C. K.-S. Leung, and P. Hu, Self-supervised deep learning model for COVID-19 lung CT image segmentation highlighting putative causal relationship among age, underlying disease and COVID-19, Journal of Translational Medicine, vol. 19, pp. 1-18, 2021.

[68] M. Goncharov, M. Pisov, A. Shevtsov, B. Shirokikh, A. Kurmukov et al., CT-Based COVID-19 triage: Deep multitask learning improves joint identification and severity quantification, Medical Image Analysis, vol. 71, p. 102054, 2021.

[69] K. Roy, D. Banik, D. Bhattacharjee, O. Krejcar, and C. Kollmann, LwMLA-NET: A lightweight multi-level attention-based network for segmentation of COVID-19 lungs abnormalities from CT images, IEEE Transactions on Instrumentation Measurement, vol. 71, pp. 1-13, 2022.

[70] M. Abdel-Basset, V. Chang, H. Hawash, R. K. Chakrabortty, and M. Ryan, FSS-2019-nCov: A deep learning architecture for semi-supervised few-shot segmentation of COVID-19 infection, Knowledge-Based Systems, vol. 212, p. 106647, 2021.

[71] N. Saeedizadeh, S. Minaee, R. Kafieh, S. Yazdani, and M. Sonka, COVID TV-Unet: Segmenting COVID-19 chest CT images using connectivity imposed Unet, Computer Methods Programs in Biomedicine Update, vol. 1, p. 100007, 2021.

[72] H. Hui, X. Zhang, F. Li, X. Mei, and Y. Guo, A partitioning-stacking prediction fusion network based on an improved attention U-Net for stroke lesion segmentation, IEEE Access, vol. 8, pp. 47419-47432, 2020.

[73] L. Liu, L. Kurgan, F.-X. Wu, and J. Wang, Attention convolutional neural network for accurate segmentation and quantification of lesions in ischemic stroke disease, Medical Image Analysis, vol. 65, p. 101791, 2020.

[74] A. Kumar, N. Upadhyay, P. Ghosal, T. Chowdhury, D. Das et al., CSNet: A new DeepNet framework for ischemic stroke lesion segmentation, Computer Methods Programs in Biomedicine, vol. 193, p. 105524, 2020.

[75] J. Wu and X. Tang, Brain segmentation based on multi-atlas guided 3D fully convolutional network ensembles, arXiv preprint arXiv:.01381, 2019.

[76] V. Rajinikanth, S. L. Fernandes, B. Bhushan, and N. R. Sunder, Segmentation and analysis of brain tumor using Tsallis entropy and regularised level set, in Proceedings of 2nd International Conference on Micro-Electronics, Electromagnetics and Telecommunications: ICMEET 2016, 2018, pp. 313-321: Springer.

[77] S. Cai, Y. Tian, H. Lui, H. Zeng, Y. Wu et al., Dense-UNet: a novel multiphoton in vivo cellular image segmentation model based on a convolutional neural network, Quantitative Imaging in Medicine Surgery, vol. 10, no. 6, p. 1275, 2020.

[78] O. Oktay, J. Schlemper, L. L. Folgoc, M. Lee, M. Heinrich et al., Attention u-net: Learning where to look for the pancreas, arXiv preprint arXiv:.03999, 2018.

[79] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 801-818.

[80] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Transactions on Pattern Analysis, vol. 40, no. 4, pp. 834-848, 2017.

[81] K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:.05645, 2014.

[82] K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770-778.

[83] C. Shorten and T. M. Khoshgoftaar, A survey on image data augmentation for deep learning, Journal of Big Data, vol. 6, no. 1, pp. 1-48, 2019.

[84] S. Jadon, A survey of loss functions for semantic segmentation, in 2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB), 2020, pp. 1-7: IEEE.

[85] D. Liu and J. Yu, Otsu method and K-means, in 2009 Ninth International Conference on Hybrid Intelligent Systems, 2009, vol. 1, pp. 344-349: IEEE.

[86] J. O. Diniz, D. B. Quintanilha, A. C. Santos Neto, G. L. da Silva, J. L. Ferreira et al., Segmentation and quantification of COVID-19 infections in CT using pulmonary vessels extraction and deep learning, Multimedia Tools Applications, vol. 80, no. 19, pp. 29367-29399, 2021.

[87] R. Hertel and R. Benlamri, A deep learning segmentation-classification pipeline for x-ray-based covid-19 diagnosis, Biomedical Engineering Advances, vol. 3, p. 100041, 2022.

[88] L. O. Teixeira, R. M. Pereira, D. Bertolini, L. S. Oliveira, L. Nanni et al., Impact of lung segmentation on the diagnosis and explanation of COVID-19 in chest X-ray images, Sensors, vol. 21, no. 21, p. 7116, 2021.

[89] N. Sajid, Covid-19 patients lungs x ray images 10000, 2020.

[90] H. B. Jenssen, Covid-19 ct segmentation dataset, 2020.

[91] J. Zhao, Y. Zhang, X. He, and P. Xie, Covid-ct-dataset: a ct scan dataset about covid-19, 2020.

[92] K. H. Abdulkareem, M. A. Mohammed, A. Salim, M. Arif, O. Geman et al., Realizing an effective COVID-19 diagnosis system based on machine learning and IOT in smart hospital environment, IEEE Internet of Things Journal, vol. 8, no. 21, pp. 15919-15928, 2021.

[93] A. Gulli and S. Pal, Deep learning with Keras. Packt Publishing Ltd, 2017.

[94] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis et al., Tensorflow: a system for large-scale machine learning, in Osdi, 2016, vol. 16, no. 2016, pp. 265-283: Savannah, GA, USA.

[95] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, Focal loss for dense object detection, in Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2980-2988.

[96] F. Milletari, N. Navab, and S.-A. Ahmadi, V-net: Fully convolutional neural networks for volumetric medical image segmentation, in 2016 Fourth International Conference on 3D Vision (3DV), 2016, pp. 565-571: Ieee.

[97] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, Segmenter: Transformer for semantic segmentation, in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 7262-7272.

[98] K. Hu, Y. Huang, W. Huang, H. Tan, Z. Chen et al., Deep supervised learning using self-adaptive auxiliary loss for COVID-19 diagnosis from imbalanced CT images, Neurocomputing, vol. 458, pp. 232-245, 2021.

[99] M. Ağralı, V. Kilic, A. Onan, E. M. Koç, A. M. Koç et al., DeepChestNet: Artificial intelligence approach for COVID-19 detection on computed tomography images, International Journal of Imaging Systems and Technology, vol. 33, no. 3, 2023.

[100] J. S. Suri, S. Agarwal, A. Carriero, A. Paschè, P. S. Danna et al., COVLIAS 1.0 vs. MedSeg: artificial intelligence-based comparative study for automated COVID-19 computed tomography lung segmentation in Italian and Croatian Cohorts, Diagnostics, vol. 11, no. 12, p. 2367, 2021.

[101] R. Liu, L. O. Hall, D. B. Goldgof, M. Zhou, R. A. Gatenby et al., Exploring deep features from brain tumor magnetic resonance images via transfer learning, in 2016 International Joint Conference on Neural Networks (IJCNN), 2016, pp. 235-242: IEEE.

[102] O. Shafaat, J. D. Bernstock, A. Shafaat, V. S. Yedavalli, G. Elsayed et al., Leveraging artificial intelligence in ischemic stroke imaging, Journal of Neuroradiology, vol. 49, no. 4, pp. 343-351, 2022.

[103] U. Koç, E. A. Sezer, Y. A. Özkaya, Y. Yarbay, O. Taydaş et al., Artificial Intelligence in Healthcare Competition (Teknofest-2021): Stroke Data Set, The Eurasian Journal of Medicine, vol. 54, no. 3, p. 248, 2022.

[104] A. S. Dina, A. Siddique, and D. Manivannan, A deep learning approach for intrusion detection in Internet of Things using focal loss function, Internet of Things, p. 100699, 2023.

Appendices

Appendix A

Publications from the Thesis

Journal Articles

1. Ağralı, M., & Kilic, V. U-TranSvision: Transformer-based deep supervision approach for COVID-19 lesion segmentation on Computed Tomography image. (under review).

2. Ağralı, M., Kilic, V., Onan, A., Koç, E. M., Koç, A. M., Büyüktoka, R. E., ... & Adıbelli, Z. DeepChestNet: Artificial intelligence approach for COVID-19 detection on computed tomography images. International Journal of Imaging Systems and Technology, vol. 33, no. 3, 2023.

3. Uçkun, S., Ağralı, M., & Kılıç, V. (2023). Deep Learning-Based Ischemic Stroke Segmentation on Brain Computed Tomography Images. Avrupa Bilim ve Teknoloji Dergisi, (50), 105-112.

4. Sayracı, B., Ağralı, M., & Kılıç, V., (2023). Artificial Intelligence Based Instance-Aware Semantic Lobe Segmentation on Chest Computed Tomography Images. European Journal of Science and Technology, (46), 109-115.

Projects

1. The project called “Artificial Intelligent Based Segmentation on Medical Imaging” was supported by the scientific research projects coordination unit of Izmir Katip Celebi University (project no. 2023-TYL-FEBE-0003).

Curriculum Vitae

Name Surname : Mahmut Ağralı

Education:

2016–2021 İzmir Kâtip Çelebi University, Dept. of Electrical and Electronics Eng.

2021–2023 İzmir Kâtip Çelebi University, Electric-Electronics Eng.

Publications:

1. Ağralı, M., & Kilic, V. U-TranSvision: Transformer-based deep supervision approach for COVID-19 lesion segmentation on Computed Tomography image. (under review).

2. Ağralı, M., Kilic, V., Onan, A., Koç, E. M., Koç, A. M., Büyüktoka, R. E., ... & Adıbelli, Z. DeepChestNet: Artificial intelligence approach for COVID-19 detection on computed tomography images. International Journal of Imaging Systems and Technology, vol. 33, no. 3, 2023.

3. Uçkun, S., Ağralı, M., & Kılıç, V. (2023). Deep Learning-Based Ischemic Stroke Segmentation on Brain Computed Tomography Images. Avrupa Bilim ve Teknoloji Dergisi, (50), 105-112.

4. Sayracı, B., Ağralı, M., & Kılıç, V., (2023). Artificial Intelligence Based Instance-Aware Semantic Lobe Segmentation on Chest Computed Tomography Images. European Journal of Science and Technology, (46), 109-115.

5. Ağralı, M., Soydemir, M. U., Gökçen, A. & Sahin, S. (2021). Deep Reinforcement Learning Based Controller Design for Model of The Vertical Take off and Landing System. Avrupa Bilim ve Teknoloji Dergisi, Ejosat Special Issue 2021 (HORA), 358-363. DOI: 10.31590/ejosat.957216