Yasin Güzel

Ph.D. Thesis

AGU 2025

DEEP LEARNING APPROACHES FOR SEGMENTATION OF BIOMEDICAL IMAGES

Ph.D. THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF ABDULLAH GUL UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

By

Yasin Güzel

November 2025

DEEP LEARNING APPROACHES FOR SEGMENTATION OF BIOMEDICAL IMAGES

A THESIS
SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING
AND THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE OF ABDULLAH GUL UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Yasin Güzel
November 2025

SCIENTIFIC ETHICS COMPLIANCE

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

Name-Surname: Yasin Güzel

Signature :

REGULATORY COMPLIANCE

Ph.D. thesis titled “Deep Learning Approaches for Segmentation of Biomedical Images” has been prepared in accordance with the Thesis Writing Guidelines of the Abdullah Gül University, Graduate School of Engineering & Science.

Prepared By	Advisor
Yasin Güzel	Assoc. Prof. Dr. Zafer Aydın
Signature	Signature

Head of the Electrical and Computer Engineering Program
Assoc. Prof. Dr. Zafer Aydın
Signature

ACCEPTANCE AND APPROVAL

Ph.D. thesis titled “Deep Learning Approaches for Segmentation of Biomedical Images” and prepared by Yasin Güzel has been accepted by the jury in the Electrical and Computer Engineering Graduate Program at Abdullah Gül University, Graduate School of Engineering & Science.

18/11/ 2025

(Thesis Defense Exam Date)

JURY:

Advisor : Assoc. Prof. Dr. Zafer Aydın …….....

Member : Assoc. Prof. Dr. Mete Çelik ……..…….....

Member : Assoc. Prof. Dr. Rıfat Kurban …….....

Member : Prof. Dr. Bülent Yılmaz .....

Member : Prof. Dr. Muhammet Fatih Talu …….....

APPROVAL:

The acceptance of this Ph.D. thesis has been approved by the decision of the Abdullah Gül University, Graduate School of Engineering & Science, Executive Board dated ….. /….. / …….. and numbered ……..……. .

…….. /…….. / ……..

(Date)

Graduate School Dean
Prof. Dr. İrfan Alan

ABSTRACT

DEEP LEARNING APPROACHES FOR SEGMENTATION OF BIOMEDICAL IMAGES

Yasin Güzel
Ph.D. in Electrical and Computer Engineering
Advisor: Assoc. Prof. Dr. Zafer Aydın
November 2025

Motivated by the crucial role of biomedical image segmentation in diagnostic workflows, this thesis investigates the potential of deep learning-based models across three distinct and challenging clinical domains. The study was conducted across three clinical scenarios; segmentation of microbial dental plaque from intraoral camera images, low-grade glioma tumors from MR, and the prostate gland from PET/CT. For each task, deep learning models were rigorously compared against classic computer vision methods, with systematic hyperparameter optimization performed to ensure a fair evaluation ground. The findings consistently demonstrated the overwhelming superiority of deep learning models over classical methods across all three domains. Notably, the U-Net Transformer model achieved statistically significantly better results than three expert dentists in dental plaque segmentation, underscoring this technology's clinical potential. Similarly, the UNet++ and Attention U-Net architectures delivered the highest performance for low-grade glioma and prostate segmentation, respectively. Furthermore, explainable AI techniques, specifically Grad-CAM and Saliency Maps, confirmed that the decision mechanisms of these high-scoring models correctly focused on the targeted anatomical regions. This process has successfully moved the models beyond the *black box* perception. This thesis provides strong evidence that task-specific deep learning solutions across diverse medical imaging modalities can not only enhance diagnostic accuracy beyond expert-level performance but also improve clinical efficiency through automation and explainability.

Keywords: Biomedical Images, Deep Learning, Explainability, Health, Segmentation

ÖZET

BİYOMEDİKAL GÖRÜNTÜLERİN SEGMENTASYONU İÇİN DERİN ÖĞRENME YAKLAŞIMLARI

Yasin Güzel
Elektrik ve Bilgisayar Mühendisliği Anabilim Dalı Doktora
Tez Danışmanı: Doç. Dr. Zafer Aydın
Kasım-2025

Bu tez çalışması, biyomedikal görüntü segmentasyonunun tanısal süreçlerdeki kritik rolünden hareketle, derin öğrenme tabanlı modellerin potansiyelini üç farklı ve zorlu tıbbi alanda araştırmaktadır. Üç farklı klinik senaryoda yürütülen çalışmalarda, sırasıyla; intraoral kamera görüntülerinden mikrobiyal dental plak, MR görüntülerinden düşük dereceli glioma tümörler ve PET/BT görüntülerinden prostat bezi segmentasyonu gerçekleştirilmiştir. Her bir problem için derin öğrenme mimarileri ile klasik bilgisayarlı görü yöntemleri karşılaştırılmış; hiperparametre optimizasyonları yapılarak adil bir değerlendirme zemini oluşturulmuştur. Bulgular, derin öğrenme modellerinin her üç alanda da klasik yöntemlere ezici bir üstünlük sağladığını tutarlı bir şekilde göstermiştir. Özellikle, diş plağı segmentasyonunda U-Net Transformer modelinin, üç uzman diş hekiminden istatistiksel olarak anlamlı düzeyde daha başarılı sonuçlar elde etmesi, bu teknolojinin klinik potansiyelini vurgulamaktadır. Benzer şekilde, düşük dereceli glioma için UNet++ ve prostat segmentasyonu için Attention U-Net mimarileri en yüksek performansı sergilemiştir. Grad-CAM ve Saliency Map ile açıklanabilirlik teknikleri ile bu yüksek skor veren modellerin karar mekanizmalarının hedeflenen anatomik bölgelere doğru odaklandığı doğrulanmış ve modelleri, *kara kutu* algısının ötesine taşımıştır. Bu tez, farklı tıbbi görüntüleme modalitelerinde probleme özgü önerilen derin öğrenme çözümlerinin, hem otomasyonu sağlayarak verimliliği artırma hem de uzman performansını aşarak tanısal doğruluğu iyileştirme gücünü kanıtlamaktadır.

Anahtar kelimeler: Açıklanabilirlik, Biyomedikal Görüntüler, Derin Öğrenme, Sağlık, Segmentasyon

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Zafer Aydın, for his invaluable guidance and contributions throughout my doctoral studies. He has been an exceptional role model for me, not only academically but also in terms of research discipline, scientific ethics, and professional integrity.

I gratefully acknowledge the jury members, Dr. Bülent Yılmaz, Dr. Mete Çelik, Dr. Muhammet Fatih Talu, and Dr. Rıfat Kurban for their time and insightful contributions. Their constructive critiques played a significant role in refining this thesis.

Last but not least, I owe my deepest gratitude to my beloved wife, *Şeyma*. Her endless love, patience, and encouragement have been my greatest strength throughout this challenging journey. I am grateful for her sacrifices and for always believing in me.

TABLE OF CONTENTS

1. INTRODUCTION 1

1.1 SEGMENTATION 1
1.2 THE ROLE AND APPLICATIONS OF SEGMENTATION IN BIOMEDICAL IMAGING 4
1.2.1 Critical Impact in Diagnostic Processes 4
1.2.2 Role in Prognosis and Disease Progression Monitoring 5
1.2.3 Role in Treatment Planning 5
1.2.4 Role in Surgical Guidance 5
1.2.5 Challenges of Manual Segmentation and the Need for Automation 6
1.3 PURPOSE, CONTRIBUTIONS, AND STRUCTURE OF THE THESIS 7
1.3.1 Purpose 7
1.3.2 Contributions 7
1.3.3 Structure 9

2. BACKGROUND 11

2.1 DEEP LEARNING 11
2.1.1 Introduction 11
2.1.2 Perceptron 12
2.1.3 Multilayer Perceptron 15
2.1.4 Backpropagation Algorithm 19
2.1.5 Optimization Algorithms 24
2.1.6 Convolutional Neural Networks 28
2.1.7 Attention Mechanism 31
2.1.8 Transformer Architecture 33
2.1.9 Evaluation Metrics 35
2.1.10 Hyperparameter Optimization 40
2.2 SEGMENTATION APPROACHES 41
2.2.1 Classical Methods 41
2.2.2 Deep Learning Based Approaches 45
2.3 OVERVIEW OF MEDICAL IMAGING MODALITIES 52
2.3.1 Intraoral Camera 53
2.3.2 Magnetic Resonance Imaging 53
2.3.3 Positron Emission Tomography / Computed Tomography 54
2.4 EXPLAINABILITY 54
2.4.1 The Importance of Explainability in Clinical Practice 55
2.4.2 Explainability Techniques 56

3. MICROBIAL DENTAL PLAQUE SEGMENTATION 58

3.1 INTRODUCTION 58
3.2 RELATED WORKS 59
3.3 MATERIALS AND METHODS 60
3.3.1 Dataset Collection 61
3.3.2 Applied Classical Methods 63
3.3.3 Applied Deep Learning Models 64
3.3.4 Statistical Analysis 64
3.4 RESULTS 65
3.5 DISCUSSION 68
3.6 CONCLUSION 71

4. LOW-GRADE GLIOMA SEGMENTATION IN MAGNETIC RESONANCE IMAGING 73

4.1 INTRODUCTION 73

4.2 Related Works 74
4.3 Materials and Methods 75
4.3.1 The Dataset 75
4.3.2 Applied Classical Methods 77
4.3.3 Applied Deep Learning Models 78
4.3.4 Statistical Analysis 79
4.4 Results 80
4.5 Discussion 83
4.6 Conclusion 85
5. PROSTATE GLAND SEGMENTATION ON ^{68}GA-PSMA-11 PET/CT 86
5.1 Introduction 86
5.2 Related Works 87
5.3 Materials and Methods 92
5.3.1 Dataset Collection 92
5.3.2 Applied Classical Methods 94
5.3.3 Applied Deep Learning Models 95
5.3.4 Statistical Analysis 97
5.4 Results 98
5.5 Discussion 101
5.6 Conclusion 103
6. CONCLUSIONS AND FUTURE PROSPECTS 105
6.1 Conclusions 105
6.2 Societal Impact and Contribution to Global Sustainability 107
6.3 Future Prospects 108

LIST OF FIGURES

Figure 1.1 Examples of different image segmentation types: (A) Input image, (B) Semantic, (C) Instance, and (D) Panoptic segmentation. 3
Figure 2.1 The structure of a perceptron [30] 12
Figure 2.2 Visualization of data separability: linearly separable data, (b) non-linearly separable data [30] 14
Figure 2.3 The architecture of a multilayer perceptron 15
Figure 2.4 Example architecture of a CNN [30] 29
Figure 2.5 The architecture of DeepLabV3+ [73] 46
Figure 2.6 The architecture of Mask-R-CNN [74] 47
Figure 2.7 The architecture of U-Net 48
Figure 2.8 The architecture of Super Vision U-Net 50
Figure 3.1 Flowchart of generating mask image 62
Figure 3.2 Dataset samples of intraoral images and ground truth masks. 63
Figure 3.3 The workflow diagram of the study. 65
Figure 3.4 Visual comparison of dental plaque segmentation results. 66
Figure 3.5 Segmentation of dental plaque by the model and three dentists. 67
Figure 3.6 Heatmap visualization of IoU scores for the model and three dentists 67
Figure 3.7 Examples showing explainability analysis of the model. 68
Figure 4.1 Visualization of multimodal brain MRI of five patients with LGG 76
Figure 4.2 Visualization of the loss and performance metrics for the training process of the seven models 80
Figure 4.3 Example segmentation predictions obtained from the methods. 81
Figure 4.4 Samples showing explainability analysis of the model. 82
Figure 5.1 Samples from the dataset 93
Figure 5.2 The architectures of DE U-Net (A) and U-Net Fusion (B) 97
Figure 5.3 Loss and performance plots during model training 97
Figure 5.4 Prediction samples of deep-learning-based models from test dataset 100
Figure 5.5 Prediction samples of classic methods from test dataset 100
Figure 5.6 Samples showing explainability analysis of the model. 101

LIST OF TABLES

Table 2.1 Confusion Matrix 36
Table 3.1 Eligibility criteria: inclusion and exclusion factors. 61
Table 3.2 Performance of deep learning models on the test dataset. 65
Table 3.3 Optimal hyperparameters of the deep learning models. 65
Table 3.4 Optimal hyperparameters of the classical methods. 65
Table 3.5 Prediction scores of the dentists and the model on 35 test images. 66
Table 3.6 *t*-test results comparing the model with the three dentists and one classical method, SLIC-GC. 66
Table 4.1 Distribution of Tumor Grades and Subtypes in LGG Cohort 76
Table 4.2 Performance of deep learning models on the test dataset. 79
Table 4.3 Optimal hyperparameters of the deep learning models. 81
Table 4.4 Optimal hyperparameters of classic methods 81
Table 5.1 Performance of the models on the test dataset 98
Table 5.2 Optimal hyperparameters of the deep learning models. 98
Table 5.3 Optimal hyperparameters of the classic methods. 98

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
BCE	Binary Cross Entropy
CNN	Convolutional Neural Networks
CT	Computed Tomography
DL	Deep Learning
DSC	Dice Similarity Coefficient
IoU	Intersection Over Union
mIoU	Mean Intersection Over Union
MRI	Magnetic Resonance Imaging
MLP	Multilayer Perceptron
OAR	Organs at Risk
PET	Positron Emission Tomography
RMSProp	Root Mean Square Propagation
SGD	Stochastic Gradient Descent
XAI	Explainable Artificial Intelligence
VGG	Visual Geometry Group
YOLO	You Only Look Once

To my father, Hoca Abdurrahman

Chapter 1

Introduction

1.1 Segmentation

The essence of our endeavors to cognize and explain things in the universe is our capacity to break the entire into useful units [1]. When we observe the world around us, we perceive objects, spaces and patterns that are separated by a clear line instead of an uninterrupted flow of information [2]. This idea, which can be traced back to Ibn al-Haytham's Book of Optics (1011–1021), emphasizes that vision is not experienced as a stream of light, but as objects, spaces, and boundaries that the mind automatically organizes [3]. This inherent capacity of the human mind to discriminate enables us to deal with complexity and to interact with our environment. In the digital era, the objective of teaching this fundamental human skill to machines has resulted in one of the core research areas [4] in the image processing and computer vision domains: *Segmentation*.

The word segmentation comes from the Latin *segmentum*, meaning 'part' or 'section'. It can clearly be defined as the systematic practice of boundary marking and identification. In its simplest definition, segmentation involves breaking down a digital image into a number of homogeneous and meaningful blocks, known as segments, according to common pixel characteristics: color, textures, intensity, or semantic content [5]. The primary aim is to encode a complex image, consisting of raw pixel data, into a more understandable form and one that machines can operate on to perform further processing [4]. Segmentation is not typically the end result, but an initial step, which permits the more intense activities [5].

The applications of this fundamental process are as broad as our imagination will allow. They include delineating a tumor in a biomedical image with more accuracy, or allowing an autonomous vehicle to differentiate between roads and pedestrians and other vehicles; tracking land use changes with satellite images, or detecting defective products

on a manufacturing line [6, 7]. Across this vast spectrum, segmentation is an indispensable first step. Effective segmentation forms the bridge between a machine's ability to *see* and its ability to *understand* [4].

Although this procedure might appear straightforward in principle, it encounters many difficulties in reality, such as noise, shifting light, intricate textures, and object overlap [8]. These difficulties make the ongoing pursuit of more accurate, efficient, and more robust segmentation algorithms a primary focus of research [9].

This ongoing effort has yielded various methods for segmentation, each with its own strengths and weaknesses. A successful segmentation algorithm should be defined mathematically as follows [5].

Consider a digital image represented by a set I of pixels. Image segmentation refers to dividing the set I, yielding n subsets (segments) $S_1, S_2, \dots, S_n$ that satisfy the following conditions:

1. *Completeness*: The union of all segments must cover the entire original image. This means that no pixel is left unassigned to a segment.

$$\bigcup_{i=1}^{n} S_i = I \tag{1.1}$$

2. *Disjointness*: A pixel from one segment cannot belong to another. This means that the intersection of any two different segments must be an empty set.

$$S_i \cap S_j = \emptyset \quad \text{for all } i \neq j \tag{1.2}$$

3. *Intra-region Homogeneity*: All pixels within each segment S_i must be 'similar' according to a predefined homogeneity criterion. This criterion is expressed as a logical proposition, $P(S_i)$. If all pixels within a segment meet this criterion, then $P(S_i)$ is true. Homogeneity criteria can include features like color, intensity, texture, or position.

$$P(S_i) = \text{TRUE} \quad \text{for all } i = 1, 2, \dots, n \tag{1.3}$$

4. *Inter-region Heterogeneity*: Two adjacent segments, S_i and S_j must be 'different' according to the same homogeneity criterion. This is expressed by the proposition P being false for the union of the two adjacent segments.

$$P(S_i \cup S_j) = \text{FALSE} \quad \text{for any two adjacent regions } S_i \text{ and } S_j \tag{1.4}$$

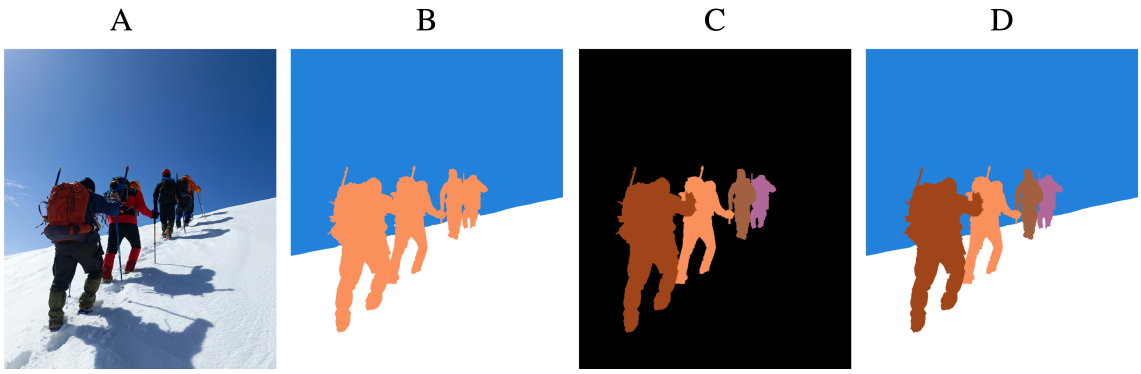


Figure 1.1 Examples of different image segmentation types: (A) Input image, (B) Semantic, (C) Instance, and (D) Panoptic segmentation.

These four conditions form the basis of successful image segmentation, acting as guidelines when designing or evaluating segmentation algorithms [5]. Below, three main types of image segmentation based on different approaches [6] are described in detail (see Figure 1.1):

Semantic segmentation: This approach assigns each pixel of an image to a particular class. For instance, it might label all cars in an image as "car", all people as "person," and all roads as "road." Nevertheless, it does not separate distinct instances of objects within the same class [10].

Instance segmentation: Unlike semantic segmentation, this method identifies and delineates each individual instance of an object within the same class. Given an image that includes four individuals, it would not just label them as "person" but as "person_1", "person_2", “person_3”, and "person_4." This is crucial for tasks that require counting or tracking objects [10].

Panoptic segmentation: This is the most comprehensive type of segmentation, combining instance and semantic segmentation. It assigns all pixels of an image to both a class (e.g., "car," "road," "person") and a separate instance ID (if applicable). This provides a complete understanding of a scene, including both the general regions (stuff like roads and sky) and each countable object (things like cars and people) [10].

1.2 The Role and Applications of Segmentation in Biomedical Imaging

Advances in biomedical imaging technologies have made it possible to examine internal human structures at remarkable resolution [11]. However, extracting clinically meaningful information from these images requires going beyond raw data to precisely identify the targeted anatomical structures or pathological regions. This is where biomedical image segmentation serves a critical function [12]. As the process of dividing a digital image into meaningful and manageable subregions, segmentation has become central to modern medical practice [13]. By transforming raw image data into interpretable and quantifiable information for clinicians and researchers, it has revolutionized many areas of medicine. In particular, within diagnostic radiology, oncology, cardiology, and neurology, segmentation enables the accurate delineation of relevant anatomical structures or lesions, making their quantitative analysis possible [14]. Its critical impact on diagnosis, prognosis, treatment planning, disease monitoring, and surgical guidance has made segmentation one of the cornerstones of clinical decision-making [12, 15].

1.2.1 Critical Impact in Diagnostic Processes

An accurate diagnosis is the first and most important step toward effective treatment [16]. Biomedical image segmentation becomes crucial in helping physicians and radiologists significantly improve diagnostic accuracy [14]. In images obtained from modalities such as Computed Tomography (CT), Positron Emission Tomography (PET), and Magnetic Resonance Imaging (MRI), one of the most fundamental functions of segmentation is to distinguish pathological findings—such as tumors, lesions, and nodules—from healthy tissue [9]. For example, in a brain MRI scan, separating tumor tissue from healthy brain matter, edema, or necrotic areas is crucial for determining the tumor's type and stage [17, 18]. Segmentation not only identifies the pathology but also provides objective measurements of its size, volume, shape, and location, consequently enhancing the precision and reliability of the diagnostic process [19].

1.2.2 Role in Prognosis and Disease Progression Monitoring

Prognosis is the process of predicting a disease's likely course and a patient's future health status [16]. Biomedical image segmentation serves as a powerful tool in this context by enabling the extraction of quantitative biomarkers essential for prognostic assessment and for monitoring the disease over time [20]. In oncology in particular, changes in the volume or shape of a segmented tumor are among the most concrete indicators of a patient's response to therapy [16]. For instance, by segmenting a tumor in pre- and post-treatment images of a cancer patient undergoing chemotherapy or radiotherapy, volumetric comparisons can be made. A reduction in tumor volume typically indicates treatment success and an improved prognosis, whereas an increase in volume or the emergence of new lesions may suggest that the treatment regimen needs to be altered, reflecting a worsening prognosis [16].

1.2.3 Role in Treatment Planning

One of the most concrete and vital roles of segmentation is in treatment planning [16]. Segmentation forms the basis for planning highly sensitive interventions like surgery and radiotherapy [15]. In radiotherapy, a common cancer treatment, the goal is to destroy cancerous cells while protecting surrounding healthy organs and tissues (organs at risk - OARs) as much as possible [16]. To achieve this, treatment planning systems use a patient's CT scans to precisely segment the tumor (the target volume) and surrounding healthy organs (like the heart, lungs, and spinal cord) [21]. This segmentation map helps optimize the intensity and angle of radiation beams. The aim is to maximize the dose to the target while minimizing the dose to OARs. Accurate segmentation directly impacts the effectiveness of the treatment and significantly reduces side effects [12].

1.2.4 Role in Surgical Guidance

Advanced surgical systems combine real-time imaging during an operation with pre-segmented anatomical models [16]. This technology allows surgeons to see the position of their instruments instantly on a pre-constructed 3D digital map of the patient [22]. This is especially useful in procedures like brain surgery or operations near sensitive organs, as it allows the surgeon to move more safely in *blind areas* [23]. The segmented structures act like a GPS, guiding the surgeon and minimizing the risk of damaging critical tissues [23].

In conclusion, biomedical image segmentation is a fundamental technology that reduces diagnostic uncertainty, strengthens prognostic predictions, and makes treatment processes more personalized and safer. Its multifaceted role in clinical practice makes it an indispensable element of contemporary medicine and paves the way for future medical innovations.

1.2.5 Challenges of Manual Segmentation and the Need for Automation

For many years, manual delineation carried out by expert clinicians such as radiologists and oncologists has been the gold standard for biomedical image segmentation. [24]. In this method, an expert uses a mouse or a special pen to manually draw the boundaries of the area of interest on each image slice. However, this approach has significant challenges and limitations:

One major limitation is the time-consuming nature of manual segmentation [7, 24]. In volumetric imaging modalities such as CT and MRI, a single scan may contain hundreds of slices. Segmenting an organ or tumor for just one patient can take hours, making it impractical in high-throughput clinical settings and hindering the analysis of large datasets. This not only consumes valuable clinician time that could otherwise be devoted to patient care but also limits the feasibility of longitudinal studies and large-scale trials.

Another challenge is intra-observer variability [25]. Even when the same expert segments the same image at different times, variations in the results are common. Factors such as fatigue, distraction, or subtle shifts in interpretation can lead to changes in the delineated boundaries. This inconsistency is particularly problematic in disease monitoring, where small changes—such as slight variations in tumor volume—can carry significant clinical implications.

Inter-observer variability further complicates matters [26]. When multiple experts segment the same image, their outputs often differ due to differences in training, experience, and personal interpretation of ambiguous boundaries. Such variability makes it difficult to establish standardized and objective evaluations in research studies and clinical trials.

These challenges have led to a pressing need for faster, more consistent, and more objective approaches in biomedical image analysis. Automatic segmentation methods,

particularly those based on AI, have emerged as a response to this need. By learning from datasets, these algorithms can produce results comparable to, and in some cases more consistent than, those of human experts. Automated systems complete segmentation tasks in minutes—or even seconds—significantly improving efficiency, eliminating inter-observer variability, and providing standardized outputs. This enables clinicians to devote more time to quantitative analysis and decision-making. As a result, automatic segmentation has become an essential instrument in both modern medical practice and research.

1.3 Purpose, Contributions, and Structure of the Thesis

1.3.1 Purpose

This thesis aims to comprehensively investigate the effectiveness and potential of deep learning (DL)–based segmentation approaches in the field of biomedical imaging. In particular, it focuses on challenging and critical clinical applications such as dental plaque, low-grade glioma (LGG), and prostate gland segmentation across different types of biomedical images, including intraoral camera images, MRI, and PET/CT. In addition to evaluating the performance of existing DL models, the study seeks to contribute to the body of knowledge in this field by constructing original datasets and developing innovative model architectures. The ultimate goal is to provide more accurate, reliable, and explainable segmentation solutions for biomedical image segmentation, thereby supporting diagnostic and therapeutic processes.

1.3.2 Contributions

The thesis provides the following original contributions to the literature:

Original Datasets: Two novel datasets have been constructed for important clinical problems that are frequently encountered in practice but lack publicly available standard datasets:

1. *Microbial Dental Plaque Segmentation:*
   An original and comprehensive dataset for dental plaque segmentation was collected using intraoral camera images collected over a six-month period. This dataset provides a critical foundation for evaluating the performance of DL models in real-world clinical scenarios.

2. *Prostate Gland Segmentation:*

   A novel dataset was collected for prostate gland segmentation from PET/CT images, making a significant contribution to research on prostate segmentation, particularly in studies involving ^{68}Ga-PSMA-11 PET/CT imaging.

Novel Deep Learning Architectures: Five new DL architectures were developed within the scope of this thesis:

1. SuperVision U-Net Transformer designed for LGG segmentation
2. Four distinct multimodal DL models developed to perform prostate segmentation by jointly using PET and CT: DE SegResNet, DE UNETR, DE U-Net, and U-Net Fusion.

Comprehensive Deep Learning Model Comparisons and Optimization: For each segmentation task, several widely used and state-of-the-art DL models from the literature were implemented and optimized through hyperparameter tuning. These comprehensive comparisons revealed the strengths and weaknesses of different architectures in addressing specific biomedical segmentation problems.

Explainability Analyses: Explainability techniques such as Grad-CAM and Saliency Map were utilized to understand the decision-making processes of the developed and top-performing DL models. These analyses visually demonstrated that the models focused on relevant anatomical regions or pathologies, thereby enhancing their clinical interpretability and acceptance.

Statistical Validation and Clinical Comparison: In the dental plaque segmentation study, the performance of the best-performing DL model was statistically compared with clinical expert assessments (annotations provided by three different dentists) using *t*-tests, strongly supporting the clinical validity of the model. Furthermore, throughout the studies, statistical comparisons between traditional image processing methods and the best-performing DL approaches clearly demonstrated the superiority of DL techniques.

Related Publications: Parts of this thesis have been published in peer-reviewed journals:

1. Publication [27]: Derived in part from Chapter 3: Microbial Dental Plaque Segmentation.
2. Publication [18]: Derived in part from Chapter 4: Low-Grade Glioma Segmentation.

1.3.3 Structure

This thesis is organized into six main chapters, following a systematic structure that begins with the theoretical background of DL–based biomedical image segmentation and progresses through experimental studies conducted on three different clinical scenarios, culminating in the final evaluations and conclusions.

Chapter 1: Introduction

This chapter presents the fundamental concepts of image segmentation and its significance in the field of biomedical imaging. The role of segmentation in diagnostic processes, disease monitoring, treatment planning, and surgical guidance is discussed, while the challenges of manual segmentation and the need for automation are emphasized. The chapter concludes with an explanation of the thesis objectives, original contributions, and overall structure.

Chapter 2: Background

This chapter provides the theoretical foundations of DL, examining topics such as MLP, CNNs, optimization algorithms, evaluation metrics, and the attention mechanism. Following this, both classic and DL-based segmentation methods are explained in detail. The chapter then offers an overview of the biomedical imaging modalities utilized in the thesis (intraoral camera, MRI, and PET/CT). The chapter concludes with a discussion of Explainable Artificial Intelligence (XAI) methods.

Chapter 3: Microbial Dental Plaque Segmentation

This study focuses on the automatic segmentation of microbial dental plaque from intraoral camera images. Using an original dataset collected over a six-month period, both classical and modern DL models were compared. The performance of the U-Net Transformer model was statistically evaluated against expert dentist annotations and SLIC-GC predictions, while Grad-CAM and Saliency Map analyses visually demonstrated the model's effective focus on plaque regions.

Chapter 4: Low-Grade Glioma Segmentation

In this chapter, the segmentation of low-grade gliomas (LGG) is performed using a publicly available MRI dataset. Various classic and modern DL models were tested. The study also included a novel hybrid model developed within the scope of this thesis, which combined the strengths of two high-performing models from Chapter 3 – Microbial Dental Plaque Segmentation study. The UNet++ model yielded the best results, and this

performance was supported by statistical analyses. The model's explainability was visually analyzed using Grad-CAM and Saliency Map.

Chapter 5: Prostate Gland Segmentation

This study addresses the segmentation of the prostate gland utilizing a newly constructed original PET/CT dataset. A comparison was drawn between models trained solely on CT data and those trained on combined PET+CT data, which included the evaluation of four new DL architectures developed within the scope of this thesis. The CT-based Attention U-Net model achieved the best results, and its explainability analyses visually demonstrated that the model focused on the correct anatomical regions.

Chapter 6: Conclusions and Future Perspectives

This chapter summarizes the findings obtained throughout the thesis and discusses the societal impact and contributions of the developed systems to global sustainability. In addition, recommendations for future research are presented, including enhancing model explainability, improving generalizability, and exploring advanced multimodal learning strategies.

Chapter 2

Background

2.1 Deep Learning

2.1.1 Introduction

We are now living in the 21st century where data is exploding as never before. This huge ocean of data pouring out of our cell phones, social media interactions, science experiments and industrial sensors is both incredibly promising and overwhelmingly complex to handle [28]. Amidst this flood of digital information, we have never more desperately needed tools that can help turn raw data into useful knowledge. This is exactly what the area of *Artificial Intelligence* (AI) is.

AI, broadly defined, is a field of research that aims to develop machines or systems that can carry out activities typically requiring human intelligence, like perception, reasoning, learning, problem solving, and language comprehension [29]. It has historically encompassed rule-based, probabilistic and statistical methods. Today, data-driven methods dominate [7]. The goal is to design appropriate objective functions, representations and optimization strategies for specific tasks.

Machine Learning (ML) is a subset of AI, is concerned with algorithms that allow models to become more efficient with data and experience, without being explicitly programmed [4]. The basic principle is to analyze data and build statistical models that are able to recognize patterns, relationships and trends in that data [30]. These models may then be leveraged to predict or decide on new and previously unseen data.

A more specialized branch of ML is called deep learning (DL). It adopts multilayered artificial neural networks (ANNs), inspired by the functioning and structure of the human brain [31]. The term *deep* denotes the number of layers in these ANNs. Information passed through each layer gets processed by that layer, and more complex

and abstract features are learned. It is this hierarchical manner of learning that enables DL to perform effectively with large amounts of complex and unstructured data, like in image and speech recognition, and natural language processing, compared to traditional ML algorithms.

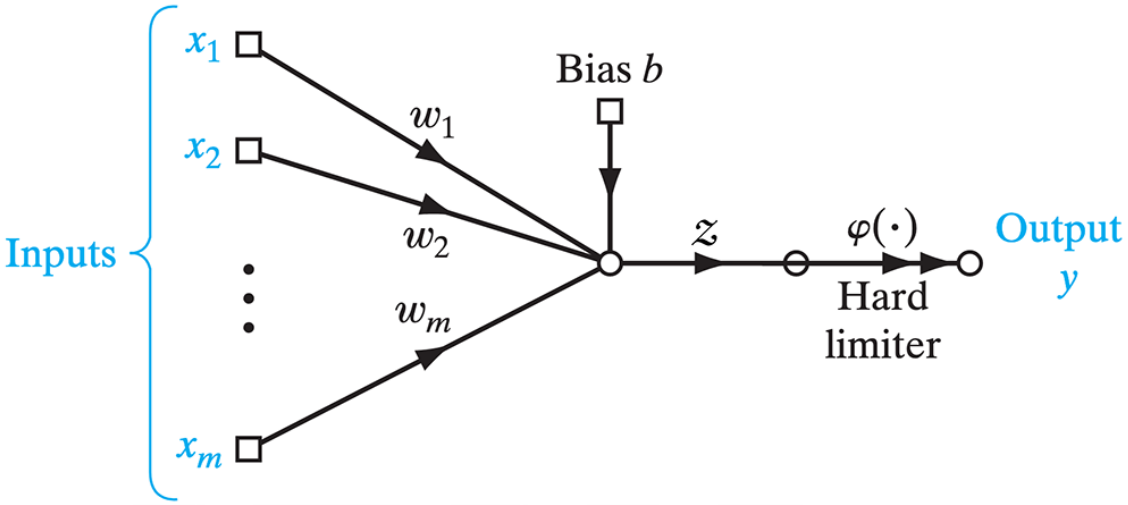


Figure 2.1 The structure of a perceptron [30]

2.1.2 Perceptron

The building block of ANNs is said to be the Perceptron developed by Frank Rosenblatt in 1957 [32]. It is a simple mathematical model inspired by the mechanism of a biological neuron. It is basically an algorithm that receives a group of inputs and provides a binary decision based on given inputs. It may choose to make an email spam or not, or a tumor benign or malignant. The logic and math behind the Perceptron is a good starting point to understanding more complex multilayered neural networks and DL models. The basic architecture of a perceptron is illustrated in Figure 2.1, where the inputs are linearly combined and passed through an activation function.

2.1.2.1 The Mathematical Model of a Perceptron

A Perceptron consists of a small number of components, namely *inputs*, *weights*, a *bias* term, and an *activation function*.

Inputs: These are features on which a model relies to make a decision. They are expressed as an input vector: $\mathbf{x}^{\mathrm{T}} = [x_1, x_2, \ldots, x_n]$. Here $\mathbf{x}^{\mathrm{T}}$ is the transpose of the input vector.

Weights: Each input (x_i) is linked to a weight (w_i). The weight indicates how much the corresponding input will contribute to the decision. A positive weight indicates that

the input activates the output, while a negative weight indicates that it inhibits the output. Weights are expressed as a vector: $\mathbf{w}^{\mathrm{T}} = [w_1, w_2, \ldots, w_n]$.

Bias: The term bias (b) acts as a threshold that determines how easily or difficult it is for a neuron to activate. Bias shifts the decision boundary and increases the model's flexibility.

The Perceptron computes *the weighted sum* by multiplying all the inputs with their corresponding weights and adding them together. This sum is represented by z:

$$z = w_1 x_1 + w_2 x_2 + \cdots + w_n x_n + b \tag{2.1}$$

A more concise version of this equation is in a vector notation:

$$z = \sum_{i=1}^{n} w_i x_i + b = \mathbf{w}^{\mathrm{T}}\mathbf{x} + b \tag{2.2}$$

Activation function: After computing the weighted sum (z), the value is processed through an *activation function*, and the output is denoted as $a = \varphi(z)$. The classic Perceptron uses a simple activation function called the *step function*. This function produces a value of 1 (class 1) if the calculated z value is positive, and -1 (class 2) otherwise.

Mathematically, the step function $\varphi(z)$ is defined as follows:

$$\hat{y} = \varphi(z) = \begin{cases} 1, & \text{if } \mathbf{w}^T\mathbf{x} + b > 0 \\ -1, & \text{if } \mathbf{w}^T\mathbf{x} + b \leq 0 \end{cases} \tag{2.3}$$

Here, $\hat{y}$ denotes the output predicted by the Perceptron.

2.1.2.2 The Rule of Perceptron

The most impressive aspect of the Perceptron is its ability to *learn* from data. The objective is to determine the optimal weight vector (**w)** and b bias value that will correctly classify the patterns in a given dataset. This process is a supervised learning problem and follows the steps below:

1. The weights ($\mathbf{w}$) and the bias (b) are assigned random initial values.
2. For each sample ($\mathbf{x}, y$) in the training dataset:
    a. The Perceptron's output ($\hat{y}$) is computed.
    b. The $\hat{y}$ is compared to the actual label y.

c. If $\hat{y} \neq y$, meaning a misclassification has occurred, the weights and the bias are updated.

3. Step 2 is repeated over the entire dataset for a fixed number of epochs

The weight update rule is the key aspect of the Perceptron. When a misclassification occurs, the weights are adjusted using the equation given below:

If $y_i(\mathbf{w}^T\mathbf{x} + b) \leq 0$: Update the parameters:

$$\mathbf{w} \leftarrow \mathbf{w} + \eta y_i \mathbf{x}_i, \qquad b \leftarrow b + \eta y_i \tag{2.4}$$

Where $\eta > 0$ is a constant learning rate.

2.1.2.3 The Weaknesses of the Perceptron

The Perceptron's working principle is based on separating data points into two classes using a hyperplane (a line in two dimensions, a plane in three dimensions). The equation $\mathbf{w}^{\mathrm{T}}\mathbf{x} + b = 0$ defines this decision boundary. Therefore, the Perceptron can only perfectly classify linearly separable data [30]. That is, cases where a single line (or hyperplane) can be drawn to completely divide the two classes between the data points. However, for non–linearly separable data, the Perceptron fails to converge and cannot correctly classify the samples. For example, as illustrated in Figure 2.2, the Perceptron can successfully separate the data in part (a), whereas it fails to separate the non-linearly separable data in part (b).

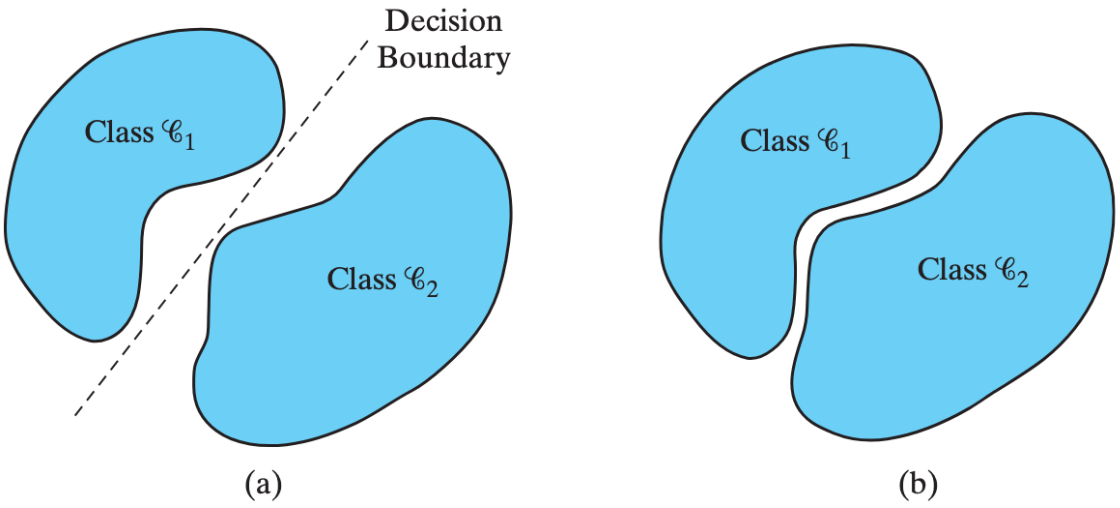


Figure 2.2 Visualization of data separability: linearly separable data, (b) non-linearly separable data [30]

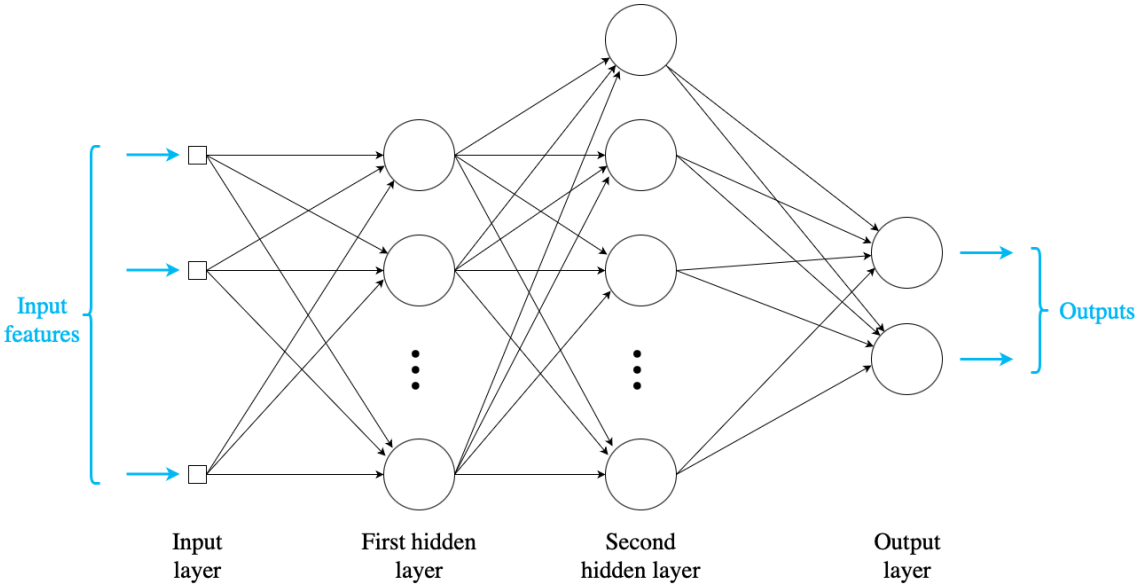


Figure 2.3 The architecture of a multilayer perceptron

However, there are also very simple problems that the Perceptron cannot solve. The XOR operation returns a value of 1 only when one of the inputs is 1 and all other values are 0. As the points of these data are plotted, we cannot divide them into two classes using one straight line.

This weaknesses contributed to an enormous stagnation in the artificially neural network development of the late 1960s. The failure of a single Perceptron to learn such non-linear problems indicated that more powerful models were required [30]. This requirement led to the emergence of *Multilayer Perceptron* (MLP) and the subsequent emergence of *Deep Learning*.

2.1.3 Multilayer Perceptron

The limitation of the single Perceptron model, which can only divide the decision space with a single hyperplane, meaning it is restricted to linearly separable problems, has been a significant obstacle in this field. MLP overcomes this limitation by using a combination of multiple *affine transformations* and *non-linear activations* [33].

An MLP typically consists of an *input layer*, one or more *hidden layers*, and an *output layer* (see Figure 2.3).

Input Layer (ℓ_0): It is the first point of contact that the network has with the data. It takes as input a $\mathbf{x} \in \mathbb{R}^n$ dimensional input vector which describes a point in data with n features. This layer has as many neurons as the dimensions of the input data. No

calculation is performed at this layer it is merely a buffer that forwards the input data to the following layer. This layer's output is the input itself: $\varphi^{(0)} = \mathbf{x}$.

Hidden Layers ($\ell_1, \ldots \ell_{L-1}$): These are the layers that lie at the core of learning of an MLP. The activation vector of the previous layer ($\varphi^{(\ell_{i-1})}$) is received by a hidden layer that has a non-linear activation function in which it undergoes an affine transformation to generate its own activation vector ($\varphi^{(\ell_i)}$). These intermediate layers are trained to produce hierarchical features of the input. Each hidden layer transforms the data space into a richer and often more easily separable form.

Output Layer (ℓ_L): This layer is the final stage of the network that generates the final result as a prediction. It transforms the activations of the previous hidden layer to an output according to the demands imposed by the problem like a probability distribution in a classification or a continuous number in a regression. This layer's output that has *K* neurons is $\hat{y} = \varphi^{(\ell_L)} \in \mathbb{R}^K$.

Mathematically, an MLP can be described as a family of functions, as shown in Equation (2.5):

$$f_{\boldsymbol{\Theta}}: \mathbb{R}^n \to \mathbb{R}^K, \qquad f_{\boldsymbol{\Theta}}(\mathbf{x}) = \varphi^{(\ell_L)}\left(\mathbf{W}^{(\ell_L)}\, \varphi^{(\ell_{L-1})}\left(\cdots \varphi^{(\ell_1)}\left(\mathbf{W}^{(\ell_1)}\mathbf{x} + \mathbf{b}^{(\ell_1)}\right)\cdots\right) + \mathbf{b}^{(\ell_L)}\right) \quad (2.5)$$

Where $\mathbf{x} \in \mathbb{R}^n$ is the input vector, *K* is the output dimension, $\boldsymbol{\Theta} = \{\mathbf{W}^{(\ell)}, \mathbf{b}^{(\ell)}\}_{\ell=1}^{\mathrm{L}}$ represents parameters of the network (the weights and biases for all layers) and $\varphi^{(\ell_i)}$ are layer-specific activation functions. The activation of the last layer $\varphi^{(\ell_L)}$ is the output of the network. In classification $\varphi^{(\ell_L)}$ is mostly used as probability (softmax) or probability-like (sigmoid); in regression, linear (identity) activation is usually preferred in the final layer.

2.1.3.1 Activation Functions

Assuming that the neurons in the hidden layers were merely performing a weighted sum of the inputs, as is done by a Perceptron, there would be no need to use multiple layers. The reason is that a series of multiple linear transformations is the same as a one-linear transformation. In this scenario we would be able to simplify the multilayer structure into one layer and this network would no longer be able to learn non-linear boundaries.

Mathematically, if $\mathbf{W}^{(\ell_1)}$ and $\mathbf{W}^{(\ell_2)}$ are the weight matrices for two layers, then $\mathbf{W}^{(\ell_2)}(\mathbf{W}^{(\ell_1)}\mathbf{x}) = \mathbf{W}^{(\ell_2)}\mathbf{W}^{(\ell_1)}\mathbf{x} = \mathbf{W}^{(net)}\mathbf{x}$.

Activation functions break this linearity by being applied after each layer's linear transformation. The differentiability of these functions is a very important characteristic which enables the network to be trained with Backpropagation. Below are some of the most frequently used activation functions:

Sigmoid [34]: It converts continuous input values into a range between 0 and 1, allowing the output to be interpreted as a probability. Therefore, it is preferred especially in binary classification, but it can lead to vanishing gradients by saturating at large $|z|$ values. The sigmoid function is shown in Equation (2.6).

$$\sigma(z) = \frac{1}{1+e^{-z}} \;\Rightarrow\; \frac{d}{dz}\sigma(z) = \sigma'(z) = \sigma(z)\big(1-\sigma(z)\big) \tag{2.6}$$

Hyperbolic Tangent (tanh) [34]: It, similar to the sigmoid, produces values between -1 and 1 and handles negative inputs better due to its zero-centered nature. This feature contributes to faster convergence during learning. Additionally, it is more resistant than the sigmoid to vanishing gradients thanks to its stronger gradients, and it generally yields more efficient training. The Tanh function is demonstrated in Equation (2.7).

$$\tanh(z) = \frac{e^{z}-e^{-z}}{e^{z}+e^{-z}} \;\Rightarrow\; \frac{d}{dz}tanh(z) = \tanh'(z) = 1-\tanh^2(z) \tag{2.7}$$

Rectified Linear Unit (ReLU) [34]: It thresholds the input at zero, returning a value of zero for negative values and the input itself for positive values [35]. The ReLU function is computationally inexpensive because it requires only a simple thresholding at zero. This allows networks to scale across multiple layers without a significant increase in computational load in comparison to more complex functions such as tanh or sigmoid. The ReLU function is shown in Equation (2.8).

$$\mathrm{ReLU}(z) = \max(0,z) \;\Rightarrow\; \frac{d}{dz}\mathrm{ReLU}(z) = \mathrm{ReLU}'(z) = \max(0,1) \tag{2.8}$$

2.1.3.2 The Universal Approximation Theorem

According to this theorem [36], a Multilayer Perceptron with one hidden layer and an adequate number of neurons is capable of approximating any continuous function with arbitrary precision, with the right non-linear activation function.

The mathematical form of the theorem is the following:

Let $\mathrm{K} \subset \mathbb{R}^{\mathrm{d}}$ be a compact set, and let $g \in C(\mathrm{K})$ be a continuous function. Suppose the activation function $\varphi: \mathbb{R} \to \mathbb{R}$ is *continuous* and *not a polynomial*. Then for every $\varepsilon >$

0, there exists an integer $l \in \mathbb{N}$ and parameters $\{\alpha_j, \mathbf{w}_j, b_j\}_{j=1}^{l}$ such that the single hidden-layer neural network

$$f(\mathbf{x}) = \sum_{j=1}^{l} \alpha_j \, \varphi\left(\mathbf{w}_j^{\top}\mathbf{x} + b_j\right) \tag{2.9}$$

satisfies $\| f - g \|_{\infty} < \varepsilon$, where $\alpha_j \in \mathbb{R}$ represent the output-layer coefficients determining the contribution of each hidden neuron to final output.

This theorem justifies a strong theoretical basis and the high representational power of MLPs. Furthermore, depth (with more than one hidden layer) can be a more efficient representation of the same number of parameters. Where a shallow network could need an exponential number of neurons to perform certain functions, deep architectures could do them with a width-per-layer growth of only a polynomial.

2.1.3.3 Feedforward Mechanism for Prediction

The prediction process of an MLP is the feedforward algorithm [30] where the data flows unidirectionally through the network layers. An example of the feedforward computation for a two hidden layer MLP with m_1 and m_2 neurons is given in Equation (2.10). The notation $h_i^{(\ell_j)}$ denotes the $i^{(th)}$ perceptron of the $j^{(th)}$hidden layer.

$$\begin{aligned} h_i^{(\ell_1)} &= \varphi^{\ell_1}\left(\sum_{j}^{n} w_{ij}^{\ell_1} x_j + b_i^{\ell_1}\right) \\ h_i^{(\ell_2)} &= \varphi^{\ell_2}\left(\sum_{j}^{m_1} w_{ij}^{\ell_2} h_j^{\ell_1} + b_i^{\ell_2}\right) \\ \hat{y}_i &= \varphi^{\ell_3}\left(\sum_{j}^{m_2} w_{ij}^{\ell_3} h_j^{\ell_2} + b_i^{\ell_3}\right) \end{aligned} \tag{2.10}$$

where, $\varphi^{\ell_i}$is the activation function of layer ℓ_i. A different activation function can be used in each layer. When vectorizing this notation, the following is obtained:

$$\mathbf{h}^{(\ell_1)} = \varphi^{\ell_1}\left(\mathbf{W}^{(\ell_1)}\mathbf{x} + \mathbf{b}^{(\ell_1)}\right)$$
$$\mathbf{h}^{(\ell_2)} = \varphi^{\ell_2}\left(\mathbf{W}^{(\ell_2)}\mathbf{h}^{(\ell_1)} + \mathbf{b}^{(\ell_2)}\right) \tag{2.11}$$
$$\hat{\mathbf{y}} = \varphi^{\ell_3}\left(\mathbf{W}^{(\ell_3)}\mathbf{h}^{(\ell_2)} + \mathbf{b}^{(\ell_3)}\right)$$

To simplify the expression, the bias term can be integrated into the weight vector by defining as

$$\tilde{\mathbf{x}}^{\mathrm{T}} = [1, x_1, x_2, \dots, x_n], \qquad \widetilde{\mathbf{w}}^{\mathrm{T}} = [b, w_1, w_2, \dots, w_n] \tag{2.12}$$

With these definitions in Equation (2.12), the affine transformation can be expressed compactly as

$$\mathbf{w}^{\mathrm{T}}\mathbf{x} + b = \widetilde{\mathbf{w}}^{\mathrm{T}}\tilde{\mathbf{x}} \tag{2.13}$$

Here, the formulation above is written for a single sample from the dataset. For the prediction of all samples in the dataset, it can be represented the dataset as a matrix $\mathbf{X} \in \mathbb{R}^{n\times m}$. Furthermore, all neurons' outputs in the $i^{(th)}$ hidden layer for all training samples can be stored in the matrix $\mathbf{H}^{\ell_i}$.

By combining the two simplifications introduced earlier, the Equation (2.11) can be rewritten as

$$\mathbf{H}^{(\ell_1)} = \varphi^{\ell_1}\left(\widetilde{\mathbf{X}}\widetilde{\mathbf{W}}^{(\ell_1)\mathrm{T}}\right)$$
$$\mathbf{H}^{(\ell_2)} = \varphi^{\ell_2}\left(\widetilde{\mathbf{H}}^{(\ell_1)}\widetilde{\mathbf{W}}^{(\ell_2)\mathrm{T}}\right) \tag{2.14}$$
$$\widehat{\mathbf{Y}} = \varphi^{\ell_3}\left(\widetilde{\mathbf{H}}^{(\ell_2)}\widetilde{\mathbf{W}}^{(\ell_3)\mathrm{T}}\right)$$

Equivalently, the expression can be written in a more compact form as

$$\widehat{\mathbf{Y}} = \varphi^{\ell_3}\left(\varphi^{\ell_2}\left(\varphi^{\ell_1}\left(\widetilde{\mathbf{X}}\widetilde{\mathbf{W}}^{(\ell_1)\mathrm{T}}\right)\widetilde{\mathbf{W}}^{(\ell_2)\mathrm{T}}\right)\widetilde{\mathbf{W}}^{(\ell_3)\mathrm{T}}\right) \tag{2.15}$$

2.1.4 Backpropagation Algorithm

The Perceptron, considered the foundational element of ANNs was introduced, and it was shown how these units can be combined into layers to form the MLP architecture in the previous sections. However, the design of an architecture represents only one aspect of the overall task. The question that arises here is how it can be taught to perform a task

such as distinguishing between pictures of cats and dogs on these networks. In this context, the way in which millions of weight (w) and bias (b) parameters are adjusted in order to enable accurate predictions must be considered.

In simplest terms, the learning phase is the network correcting itself based on its errors. In this section, the *Backpropagation algorithm* [33], which constitutes the backbone of DL [34], together with the rationale behind it, will be examined. The process is initiated by measuring the error, an optimization strategy is then determined to minimize this error, and finally the responsibility for the error is efficiently distributed throughout the layers of the network.

2.1.4.1 Loss Functions

Without quantifying how *poorly* a network performs, it is impossible to know how to improve it. This is where *loss functions* or *cost functions* come into play. A loss function is defined as a function quantifies how far the model's predicted outputs ($\hat{y}$) are from the actual targets (y) [30]. The objective is to minimize the value produced by this function, that is, the loss $\mathcal{L}(y, \hat{y})$. To achieve this, all parameters of the network ($\mathbf{W}, \mathbf{b}$) must be optimized.

The selection of loss function is determined by the type of problem under consideration. For example, the Binary Cross-Entropy (BCE) function is commonly adopted for classification problems, whereas the Mean Squared Error (MSE) function is typically employed for regression tasks [34].

1. *Mean Squared Error (MSE)* [37]: It is mainly employed in regression problems (e.g., predicting a patient's blood pressure). It takes the average of the squared difference between the predicted and actual value. Squaring the difference ensures that the error is a positive value and also penalizes large errors more heavily. The formula is shown in Equation (2.16):

$$\mathcal{L}_{\text{MSE}} = -\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2 \tag{2.16}$$

2. *Binary Cross Entropy (BCE)* [37]: It is commonly used in classification problems (e.g., whether a tumor is benign or malignant). It measures the difference between the probability distribution predicted by the model and the

actual probability distribution. The formula for binary classification is given in Equation (2.17):

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{m}\sum_{i=1}^{m}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \tag{2.17}$$

A multilayer perceptron (MLP) capable of processing data and producing an output has been examined, together with the loss functions that measure how *poor* a prediction is. At this point, there is a model that can make predictions and a function that can quantify the associated error. The crucial question that arises is how the error (loss) produced by the model can be minimized, in other words, how the model can *learn* from the data.

The answer to this question lies in two powerful mechanisms that form the basis of DL: *Gradient Descent* and *Backpropagation*, which efficiently performs the calculations required by this mechanism [29, 32].

2.1.4.2 Gradient Descent

Gradient Descent is an optimization algorithm used to find the minimum value of a function [37]. In the context of DL, this function is the loss function. The objective is to adjust weights and biases of the network in such a way that the value of the loss function is minimized as much as possible. Gradient Descent does this through iterative steps.

The gradient ($\nabla\mathcal{L}$) is defined as a vector consisting of the partial derivatives of $\mathcal{L}$ with respect to all parameters, and it indicates the direction of steepest ascent of the function. Let all parameters of the network (all weights and biases) be collected into a single vector $\boldsymbol{\theta} \in \mathbb{R}^P$.

$$\nabla_{\boldsymbol{\theta}}\mathcal{L} = \begin{pmatrix} \frac{\partial\mathcal{L}}{\partial\theta_1} \\ \frac{\partial\mathcal{L}}{\partial\theta_2} \\ \vdots \\ \frac{\partial\mathcal{L}}{\partial\theta_n} \end{pmatrix} \tag{2.18}$$

To reduce the loss, movement is carried out in the opposite direction of this vector. A common training objective is given as

$$J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m} \mathcal{L}(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i) + \lambda\Omega(\boldsymbol{\theta}) \tag{2.19}$$

where J objective (or cost) function, and $\lambda\Omega(\boldsymbol{\theta})$ represents the regularization term. As an update rule, gradient descent updates the parameters by taking small steps along $-\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$ which is the direction opposite to the fastest increase in $J(\boldsymbol{\theta})$ (Equation (2.21)):

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) = \frac{1}{m}\sum_{i=1}^{m} \nabla_{\boldsymbol{\theta}} \mathcal{L}(f_{\boldsymbol{\theta}}(\mathbf{x}_i), y_i) + \lambda\nabla_{\boldsymbol{\theta}}\Omega(\boldsymbol{\theta}) \tag{2.20}$$

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}_t) \tag{2.21}$$

where $\eta > 0$ is the learning rate. This is a hyperparameter that specifies the magnitude of each update step.

2.1.4.3 Backpropagation

The Gradient Descent algorithm requires the gradient $\nabla_{\boldsymbol{\theta}}\mathcal{L}$, that is, the derivative of the loss with respect to each weight and bias [30]. Calculating these gradients manually is impossible in an ANN with millions of parameters. This is where the Backpropagation algorithm comes into play. Backpropagation is, in essence, an algorithmic implementation of the *Chain Rule* in Calculus [33].

Error at the output layer: $\delta_j^{(\ell_i)}$ represents the *error term* in the $j^{(th)}$ neuron in the ℓ_i layer. This is a critical concept and shows the effect of a small change in the pre-activation value of that neuron on the total loss.

The error at the $j^{(th)}$ neuron in the output layer ($\delta_j^{(\ell_L)}$) is the product of the derivative of the loss function with respect to the output activation and the derivative of the activation function with respect to $z_j^{(\ell_L)}$:

$$\begin{aligned} \delta_j^{(\ell_L)} &\triangleq \frac{\partial \mathcal{L}}{z_j^{(\ell_L)}} \\ \delta_j^{(\ell_L)} &= \frac{\partial \mathcal{L}}{\partial a_j^{(\ell_L)}}\frac{\partial a_j^{(\ell_L)}}{\partial z_j^{(\ell_L)}} \\ \delta_j^{(\ell_L)} &= \frac{\partial \mathcal{L}}{\partial a_j^{(\ell_L)}}\varphi'\left(z_j^{(\ell_L)}\right) \end{aligned} \tag{2.22}$$

It can be written in vector notation as:

$$\boldsymbol{\delta}^{(\ell_L)} = \nabla_{\mathbf{a}^{(\ell_L)}} \mathcal{L} \odot \varphi'\left(\mathbf{z}^{(\ell_L)}\right) \tag{2.23}$$

where, $\odot$ denotes element-wise multiplication (Hadamard product). The function φ is applied component-wise to vectors.

Error in the hidden layer: The error in any hidden layer ℓ_i can be expressed in terms of the weighted sum of the errors in the subsequent layer ℓ_{i+1} propagated backward. This constitutes the *propagation* part of the algorithm.

$$\delta_j^{(\ell_i)} = \left(\sum_k w_{kj}^{(\ell_{i+1})}\, \delta_k^{(\ell_{i+1})}\right) \varphi'\left(z_j^{(\ell_i)}\right) \tag{2.24}$$

It can be written in vector notation as:

$$\boldsymbol{\delta}^{(\ell_i)} = \left(\mathbf{W}^{(\ell_{i+1})}\right)^{\mathrm{T}} \boldsymbol{\delta}^{(\ell_{i+1})} \odot \varphi'\left(\mathbf{z}^{(\ell_i)}\right) \tag{2.25}$$

where, $\left(\mathbf{W}^{(\ell_{i+1})}\right)^{\mathrm{T}}$ is the transpose of the weight matrix of layer ℓ_{i+1}.

Gradient with respect to the parameters: The gradient of a weight is equal to the product of the error term of the neuron it is connected to and the activation of the neuron that the connection comes from, whereas the gradient of the bias is equal to the error term of that neuron:

$$\frac{\partial \mathcal{L}}{\partial w_{jk}^{(\ell_i)}} = a_k^{(\ell_{i-1})} \delta_j^{(\ell_i)}, \qquad \frac{\partial \mathcal{L}}{\partial b_j^{(\ell_i)}} = \delta_j^{(\ell_i)} \tag{2.26}$$

In summary, for a given input example **x,** the value $z^{(\ell_i)}$ and $a^{(\ell_i)}$ are computed and stored at each layer. For the output layer, the error vector $\boldsymbol{\delta}^{(\ell_L)}$ is calculated using Equation (2.23). For $\ell_i = \ell_{L-1}, \ell_{L-2}, \ldots, 2$, the error vector of each layer $\boldsymbol{\delta}^{(\ell_i)}$ is computed in terms of the error of the subsequent layer $\boldsymbol{\delta}^{(\ell_{i+1})}$ by means of Equation (2.25). For each layer ℓ_i the required gradients $\frac{\partial \mathcal{L}}{\partial w_{jk}^{(\ell_i)}}$ and $\frac{\partial \mathcal{L}}{\partial b_j^{(\ell_i)}}$ are obtained using Equation (2.26). Finally, the parameters are updated through a Gradient Descent step employing the computed gradients.

2.1.5 Optimization Algorithms

Optimization algorithms are employed at the stage where the gradient vector moves in the opposite direction to minimize the loss function [34]. The task of these algorithms is to determine how to adjust the parameters of the network using the computed gradients. The Gradient Descent algorithm, which is the most basic optimization algorithm, and its variations have some challenges [34]:

- Selecting an appropriate learning rate is challenging. If it is too small, learning is slowed excessively, whereas if it is too large, the optimal point may be overshot, leading to divergence.
- Using the same learning rate for all parameters may not be efficient.
- On high-dimensional and non-convex error surfaces, it can become trapped in suboptimal local minima or saddle points that are not optimal solutions.

More advanced optimization algorithms have been developed to overcome these difficulties. In this section, starting from the basic variations of Gradient Descent, advanced optimization techniques that are frequently employed in modern DL architectures will be examined.

2.1.5.1 Variations of Gradient Descent

Gradient Descent is subdivided into three main variants depending on the amount of data used in the gradient computation:

1. **Batch Gradient Descent**: In this method, the entire training dataset is utilized to compute the gradient. Weight updates are performed only once after the entire data set has been processed. Mathematical formulation:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{X}, \mathbf{y}) \tag{2.27}$$

   where, $\mathbf{X}$ and $\mathbf{y}$ represent the entire dataset. Since the gradient calculation is noise-free, it progresses steadily and smoothly toward the optimal solution. However, it needs to keep the whole dataset loaded in memory, and for very large datasets (e.g., millions of images), each update step is extremely slow and computationally expensive. For this reason, it is almost never used in modern DL applications.

2. **Stochastic Gradient Descent (SGD)**: In this method, only a single data sample randomly selected from training set is used for each update step. Mathematical formulation:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{x}_i, y_i) \tag{2.28}$$

where, $(\mathbf{x}_i, y_i)$ is a single sample selected from the training set. Each update is performed very quickly and is therefore ideal for large datasets. Noise in the gradient computation can help the algorithm escape local minima. However, the update steps are very noisy and irregular. This can slow down convergence to the optimal solution and cause large oscillations in the loss function.

3. **Mini-Batch Gradient Descent**: This approach incorporates desirable properties of both the Batch and Stochastic approaches. Instead of using the entire dataset or a single example, it uses a small subset of data (mini-batch). Mini-batches are typically chosen with sizes that are powers of two, such as 16 or 32. Mathematical formulation:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{X}_{i:i+n}, \mathbf{y}_{i:i+n}) \tag{2.29}$$

where, n denotes the mini-batch size. It is much faster than batch GD and provides more stable convergence than SGD. Moreover, due to vectorized computations, it can take advantage of the hardware architecture of modern GPUs with high efficiency.

2.1.5.2 Momentum

One of the worst shortcomings of Gradient Descent is its slow progress in narrow and elongated valleys (ravines), where the gradient direction changes frequently. Momentum [38] is introduced to address this issue, inspired by the concept of *inertia* in physics. Similar to how a ball rolling down a hill moves not only according to the current slope but also based on the velocity (momentum) it has accumulated, this algorithm takes into account the direction of previous steps.

This is modeled using a velocity vector ($\boldsymbol{v}_t$). This vector maintains the exponentially decreasing mean of previous gradients. The parameter update is then performed using this velocity rather than the current gradient. Update rule (Momentum):

$$\boldsymbol{v}_t = \gamma \boldsymbol{v}_{t-1} + \nabla_{\boldsymbol{\theta}} \mathcal{L} \tag{2.30}$$

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \boldsymbol{v}_t \tag{2.31}$$

where, $\gamma \in [0,1)$ is the momentum coefficient (typically $\gamma \approx 0.9$). This coefficient controls how much the previous steps are *remembered*. Momentum accelerates gradients in the same direction across successive steps while decelerating those in opposite directions, thereby reducing oscillations and enabling faster convergence.

2.1.5.3 Adaptive Gradient Algorithm

The methods discussed so far use the same learning rate (η) for all parameters ($\boldsymbol{\theta}$). However, some parameters may need to be updated more frequently, while others may need to be updated less often. The Adaptive Gradient Algorithm (Adagrad) [39] offers an adaptive learning rate specific to each parameter. The idea is simple: reduce the learning rate for parameters that have received large updates in the past, and increase it for parameters that are updated less frequently.

It does this by storing the sum of the squares of the past gradients for each parameter. Update rule (Adagrad):

$$\mathbf{g}_t = \mathbf{g}_{t-1} + (\nabla_{\boldsymbol{\theta}} \mathcal{L})^2 \tag{2.32}$$

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \frac{\eta}{\sqrt{\mathbf{g}_t + \epsilon}} \nabla_{\boldsymbol{\theta}} \mathcal{L} \tag{2.33}$$

where, $\mathbf{g}_t$, is a vector that holds the sum of the squares of the gradients up to time t, and ϵ (epsilon, typically a very small value like 10^{-8}) is added to prevent division by zero errors.

The major drawback of Adagrad is that the $\mathbf{g}_t$ term in the denominator continuously increases. This can cause the learning rate to decrease significantly over time and eventually halt learning altogether.

2.1.5.4 Root Mean Square Propagation

Root Mean Square Propagation (RMSprop) [40] was proposed to mitigate Adagrad's tendency to shrink the learning rate too quickly. Instead of accumulating all

past gradient squares, it uses an exponentially decaying moving average. This way, only recent gradient magnitudes are considered. Update rule (RMSprop):

$$\mathbf{g}_t = \gamma \mathbf{g}_{t-1} + (1-\gamma)(\nabla_{\boldsymbol{\theta}} \mathcal{L})^2 \tag{2.34}$$

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta \frac{\nabla_{\boldsymbol{\theta}} \mathcal{L}}{\sqrt{\mathbf{g}_t} + \epsilon} \tag{2.35}$$

where, $\mathbf{g}_t$ is the exponentially weighted average of the squared gradients.

2.1.5.5 Adaptive Moment Estimation

Adaptive Moment Estimation (Adam) [41] is perhaps the most widely used optimization algorithm today. It combines the best features of both Momentum and RMSprop. Specifically, it follows the first-order moment of the gradients (as in Momentum) as well as the second-order moment (the squared gradients, as in RMSprop).

1. First Moment (Momentum): Calculates the exponentially decaying average of gradients ($\boldsymbol{m}_t$).
2. Second Moment (Adaptive Learning Rate): Computes the exponentially decaying average of the squared gradients ($\boldsymbol{v}_t$).

Update rule (Adam):

$$\boldsymbol{m}_t = \beta_1 \boldsymbol{m}_{t-1} + (1-\beta_1)\nabla_{\boldsymbol{\theta}} \mathcal{L} \tag{2.36}$$

$$\boldsymbol{v}_t = \beta_2 \boldsymbol{v}_{t-1} + (1-\beta_2)(\nabla_{\boldsymbol{\theta}} \mathcal{L})^2 \tag{2.37}$$

Since m_t and v_t will initially be close to zero, the algorithm includes a *bias correction* step:

$$\widehat{\boldsymbol{m}}_t = \frac{\boldsymbol{m}_t}{1-\beta_1^t} \tag{2.38}$$

$$\widehat{\boldsymbol{v}}_t = \frac{\boldsymbol{v}_t}{1-\beta_2^t} \tag{2.39}$$

Compute the adaptive learning rates:

$$\eta_t = \frac{\eta_{t-1}\sqrt{1-\beta_2^t}}{1-\beta_1^t} \tag{2.40}$$

Finally, the parameter update is performed as follows:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t \frac{\widehat{\boldsymbol{m}}_t}{\sqrt{\widehat{\boldsymbol{v}}_t} + \epsilon} \tag{2.41}$$

where, β_1 (typically 0.9) and β_2 (typically 0.999) are the decay rates. Adam is generally a good starting point for most problems, converges quickly, and its hyperparameters are relatively easy to tune. For this reason, it appears as the default choice in many DL applications.

2.1.6 Convolutional Neural Networks

The Multilayer Perceptrons (MLPs) examined in the previous sections are powerful general-purpose learning models. However, when confronted with data that exhibit spatial structure, such as images, they encounter significant limitations. To feed an image into an MLP, it is typically flattened into a single vector. This transformation destroys vital spatial information, such as the neighborhood relationship between pixels [34]. Furthermore, a high-resolution image translates to a massive input vector, which means an astronomical number of parameters (weights) even in the first layer [30]. This increases computational cost and makes the model highly prone to overfitting.

At this point, Convolutional Neural Networks (CNNs) [42], inspired by the human visual cortex, are introduced. CNNs are a specialized architecture designed to leverage the nature of image data. By sharing parameters intelligently and preserving the spatial hierarchy of the data, CNNs have brought major advancements to computer vision tasks, particularly in image recognition, object detection, and segmentation.

Consider a color image represented as a tensor $\mathbf{X} \in \mathbb{R}^{C \times H \times W}$, where C denotes the number of channels, and $H$and W denote the height and width, respectively. CNNs produce feature maps by passing small spatial filters (e.g., 3×3) over this tensor. Each filter is sensitive to a specific visual pattern (such as edges, corners, or textures). As the network progresses to deeper layers, the receptive field increases, allowing the extraction of more abstract features (such as parts or objects).

One of the main reasons CNNs are indispensable is their ability to dramatically decrease the number of parameters. For example, in a fully connected layer with an input of size $C \times H \times W$, even a single neuron would require CHW weights. In contrast, a

$k \times k$ convolution shares the same k^2C weights across all positions, which drastically reduces both the number of parameters and the computational cost.

As illustrated in Figure 2.4, this Convolutional Neural Network (CNN) consists of convolutional and pooling layers

2.1.6.1 Convolution Layer

The convolution layer forms the core of the CNN and is in charge of feature extraction [42]. The purpose of this layer is to capture fundamental features such as textures, or edges in the input image (or in the feature map from the previous layer).

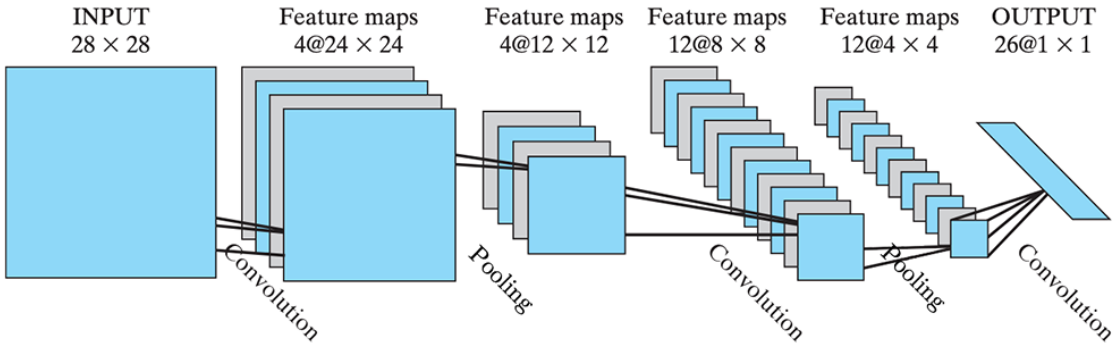


Figure 2.4 Example architecture of a CNN [30]

It performs this operation using small, learnable weight matrices called filters (kernels). The filter is slid step by step over the input image. At each position, an element-wise multiplication is performed between the elements of the filter and the pixel values in the region of the image covered by the filter, and the results are summed. This mathematical operation is called convolution. This approach dramatically decreases the number of parameters compared to fully connected layers while remaining sensitive to spatial structures.

Let the input be $\mathbf{X} \in \mathbb{R}^{C_{in} \times H \times W}$, the filter bank be $\mathbf{W} \in \mathbb{R}^{C_{out} \times C_{in} \times K_h \times K_w}$, the bias vector be $\mathbf{b} \in \mathbb{R}^{C_{out}}$, and the output be $\mathbf{Y} \in \mathbb{R}^{C_{out} \times H_{out} \times W_{out}}$. Here, K_h and K_w denote the kernel height and width, s_h, s_w denote the stride values, and p_h, p_w denote the padding values. Forward propagation (0-based indexing):

$$\mathbf{Y}[c,i,j] = \mathbf{b}[c] + \sum_{c'=0}^{C_{in}-1} \sum_{u=0}^{K_h-1} \sum_{v=0}^{K_w-1} \mathbf{W}[c,c',u,v]\mathbf{X}[c', is_h + u - p_h, js_w + v - p_w] \quad (2.42)$$

Output dimensions:

$$H_{out} = \left(\frac{H + 2p_h - K_h}{s_h}\right) + 1, \qquad W_{out} = \left(\frac{W + 2p_w - K_w}{s_w}\right) + 1 \tag{2.43}$$

Backpropagation of convolutional layer: The gradients are computed as follows:

$$\mathbf{G} = \frac{\partial \mathcal{L}}{\partial \mathbf{Y}} \in \mathbb{R}^{C_{out} \times H_{out} \times W_{out}} \tag{2.44}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{W}[c, c', u, v]} = \sum_{i=0}^{H_{out}-1} \sum_{j=0}^{W_{out}-1} \mathbf{G}[c, i, j]\mathbf{X}[c', is_h + u - p_h, js_w + v - p_w] \tag{2.45}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{b}[c]} = \sum_{i=0}^{H_{out}-1} \sum_{j=0}^{W_{out}-1} \mathbf{G}[c, i, j] \tag{2.46}$$

$$\frac{\partial \mathcal{L}}{\partial \mathbf{X}[c', p, q]} = \sum_{i=0}^{H_{out}-1} \sum_{j=0}^{W_{out}-1} \mathbf{G}[c, i, j]\mathbf{W}[c, c', p - is_h + p_h, q - js_w + p_w] \tag{2.47}$$

2.1.6.2 Pooling Layer

A pooling layer performs downsampling to lower the spatial dimensions of the feature maps, thereby decreasing the computational burden and improving the model's robustness to small shifts and distortions [34]. It is parameter-free: there are no weights to learn; only hyperparameters such as window size and stride exist.

Pooling operates independently for each channel. A window of size $K_h \times K_w$ is shifted over the input tensor $\mathbf{X} \in \mathbb{R}^{C \times H \times W}$ with stride values s_h, s_w. The output is given by $\mathbf{Y} \in \mathbb{R}^{C \times H_{out} \times W_{out}}$, where the dimensions are computed as follows:

$$H_{out} = \left(\frac{H - K_h}{s_h}\right) + 1, \qquad W_{out} = \left(\frac{W - K_w}{s_w}\right) + 1 \tag{2.48}$$

For each (i, j) position, the values under the window are reduced to a single scalar value; how this value is calculated determines the pooling type. The most commonly used pooling types are Max Pooling and Average Pooling.

In Max Pooling, the largest value within the window is taken, with each channel being independent. It tends to preserve sharp edges and textures. The process is performed as follows:

$$\mathbf{Y}[c,i,j] = \max(\boldsymbol{X}[c, is_h + u, js_w + v]) \tag{2.49}$$

In Average Pooling, the arithmetic mean of the values within the window is taken:

$$\mathbf{Y}[c,i,j] = \frac{1}{K_h K_w} \sum_{u=0}^{K_h-1} \sum_{v=0}^{K_w-1} \boldsymbol{X}[c, is_h + u, js_w + v] \tag{2.50}$$

Backpropagation of pooling layer: For Max Pooling, the gradient is only 1 at the argmax position for each window; other positions receive a gradient of 0. For Average Pooling: The gradient is divided equally among all positions within the window.

2.1.7 Attention Mechanism

In the context of DL, especially in natural language processing (NLP) tasks, the processing of sequential data has been carried out for many years using recurrent neural networks (RNNs) and their derivatives, such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) models [43–45]. These models rely on the principle of sequentially processing words or elements in a sequence and accumulating contextual information in a hidden state vector [46]. However, these architectures face certain fundamental limitations. In particular, when working with long sequences, it becomes increasingly difficult for the model to carry information from the beginning of the sequence all the way to the end. As a result, an important detail that appears early in a sentence may be forgotten or its influence diminished when producing outputs toward the end.

The Attention Mechanism was introduced to overcome this limitation [47]. Rather than depending on a fixed summary, attention allows the model to dynamically and directly *look at* any part of the input when needed. Inspired by the phenomenon of attention in the human cognitive system, this mechanism allows a model to determine that it needs to *pay more attention* to or concentrate on particular regions of the input [47].

At the core of the attention mechanism lies a weighting process that determines which elements of the input sequence a model should focus on. This process is typically shaped around three fundamental concepts: *Query*, *Key*, and *Value* [48]. The Query represents the current position or context. It is the vector that asks “What should I focus on?” when generating output. The Key acts as a kind of *label* for each item in the sequence. It is used in comparison with the Query to compute an attention score. The

Value carries the actual content or meaning of each element in the input sequence. This is the vector that will be aggregated as a result of weighting.

For each input element, the model learns these three distinct vector representations. Let a single word embedding be denoted by **x**. In this case, the query **q**, key **k** and value **v** vectors are obtained as different linear transformations of **x**. These transformations are performed using learnable weight matrices $\mathbf{W}^Q$, $\mathbf{W}^K$, and $\mathbf{W}^V$:

$$\mathbf{q} = \mathbf{W}^Q\mathbf{x}, \qquad \mathbf{k} = \mathbf{W}^K\mathbf{x}, \qquad \mathbf{v} = \mathbf{W}^V\mathbf{x} \tag{2.51}$$

Here, the matrices $\mathbf{W}^Q$, $\mathbf{W}^K$, and $\mathbf{W}^V$ are the learnable parameters of the model during the training process. The attention mechanism operates as follows to produce an output vector ($\mathbf{z}_j$) for the $j^{(th)}$ position:

1. **Computation of Similarity Scores**: A similarity score is calculated between the query vector ($\mathbf{q}_j$)) at the $j^{(th)}$ position and the key vector ($\mathbf{k}_i$) at each $i^{(th)}$ position in the input sequence. This is typically carried out using dot product

$$e_{ji} = \mathbf{q}_j^{\mathrm{T}}\mathbf{k}_i \tag{2.52}$$

   This scalar value (e_{ji}) is a raw score indicating how much attention the $j^{(th)}$ position should pay to the $i^{(th)}$ position.

2. **Scaling and Softmax**: The scores are divided by the square root of the dimension of the key vectors (d_k) to prevent gradients from becoming excessively small. Then, a softmax function is applied to these scaled scores to obtain attention weights (α).

$$\alpha_{ji} = \mathrm{softmax}\left(\frac{e_{ji}}{\sqrt{d_k}}\right) = \frac{e^{\left(\frac{e_{ji}}{\sqrt{d_k}}\right)}}{\sum_{l=1}^{n} e^{\left(\frac{e_{ji}}{\sqrt{d_k}}\right)}} \tag{2.53}$$

   The α_{ji} weights are positive numbers whose sum is 1, and they indicate how much importance is assigned to the $i^{(th)}$ input position when producing the $j^{(th)}$ output.

3. **Weighted Sum**: Finally, the output vector $\mathbf{z}_j$ is obtained by multiplying the value vectors ($\mathbf{v}_i$) of all input positions with the computed attention weights (α_{ji}) and summing them up.

$$\mathbf{z}_j = \sum_{i=1}^{n} \alpha_{ji}\, \mathbf{v}_i \tag{2.54}$$

These steps can be performed simultaneously and efficiently for all positions using matrix operations. When all query, key, and value vectors are combined into the Q, K, and V matrices, respectively, the output of the attention mechanism, the matrix Z, is formulated as follows:

$$\mathbf{Z} = \text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^{\mathrm{T}}}{\sqrt{d_k}}\right)\mathbf{V} \tag{2.55}$$

where, d_k, is the dimension of the key vectors. This formula is used to enable the attention mechanism to be computed in parallel for all queries, which is one of the keys to the efficiency of the Transformer architecture.

2.1.8 Transformer Architecture

The Transformer architecture [48] is a model built entirely on the attention mechanism, without using recurrent or convolutional layers. This architecture has become the cornerstone of DL models because of its parallel processing capability and superior success in modeling long-range dependencies.

2.1.8.1 Multi-Head Attention

A single attention mechanism allows a query to learn the relationships within the input sequence from a single perspective [48]. However, considering that a word or sentence can have different meanings or establish different relationships in different contexts, a single attention mechanism may not be sufficient. Therefore, the Multi-Head Attention mechanism was developed.

Multi-Head Attention is fundamentally based on the principle of running the attention mechanism multiple times in parallel and enabling each *head* to learn to concentrate on various subspaces of the input representation. Each attention head applies an independent attention function over the input query, key, and value vectors. This

allows the model to capture relationships across different *representation subspaces* using the same input information. Mathematically, each head h computes an attention output with its own weight matrices $\mathbf{W}_h^Q$, $\mathbf{W}_h^K$, and $\mathbf{W}_h^V$:

$$\text{head}_h = \text{Attention}\left(\mathbf{QW}_h^Q, \mathbf{KW}_h^K, \mathbf{VW}_h^V\right) \tag{2.56}$$

The outputs of these heads are concatenated and passed through an additional weight matrix ($\mathbf{W}^O$) to form the final output:

$$\text{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_H)\mathbf{W}^O \tag{2.57}$$

2.1.8.2 Positional Encoding

Unlike sequential models, the Transformer architecture does not have recurrent connections and therefore does not naturally process the order of inputs [48]. However, in sequential data such as language, the position and order of words are very important. To prevent this loss of information, the Transformer adds *positional encodings* to the input embedding vectors. These encodings are constructed using sinusoidal functions that help the model recognize the position of each word in the sequence. Positional encodings are not learnable parameters; instead, they are calculated in a fixed manner using the following formulas [48]:

$$PE_{(pos,2i)} = \sin\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \tag{2.58}$$

$$PE_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{\frac{2i}{d_{model}}}}\right) \tag{2.59}$$

where, pos is the position of the word, i is the dimension index of the embedding vector. d_{model} is the predefined dimensionality of the token embeddings. This operation provides the model with essential information about the ordering of words.

2.1.8.3 Encoder Block

The Encoder processes the input sequence (e.g., the words in a sentence) to produce a semantic representation (context vectors). Each Encoder layer is composed of two main sub-layers:

1. *Multi-Head Self-Attention layer*: This sublayer enables each element in the input sequence to learn its relationship with all other elements in the sequence. Here, **Q**, **K**, and **V** come from the same source (the output of the previous layer).
2. *Position-wise Feed-Forward Network*: This network is composed of two linear layers and a ReLU activation applied independently to each position:

$$FFN(\mathbf{x}) = \max(0, \mathbf{x}\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2 \quad (2.60)$$

This network learns higher-level abstractions by further transforming the complex relationships captured by the attention mechanism.

2.1.8.4 Decoder Block

The decoder block is similar to the Encoder Block but additionally contains a second Multi-Head Attention layer that focuses on the encoder's output. In this layer, the **K** and **V** matrices come from the encoder's output, while the **Q** matrix comes from the decoder's own previous layer. This structure ensures that the decoder takes the entire original input into account when generating output.

The Transformer architecture, by combining these components, has produced a highly parallelizable and scalable model capable of capturing relationships in long and complex sequences with extraordinary success. Thanks to these features, it has revolutionized not only the context of NLP but also computer vision.

2.1.9 Evaluation Metrics

Quantitatively measuring how successful the developed model is in practice, identifying its weaknesses, and developing strategies to address these weaknesses is at least as important as the model's design. This section will cover the fundamental metrics utilized to assess the performance of DL models and the confusion matrix on which these metrics are based.

2.1.9.1 Confusion Matrix

The Confusion Matrix [49], a major tool for assessing model performance, provides a tabular summary that compares a model's predictions with actual labels, particularly in binary and multi-class classification problems. This matrix allows us to visualize which

classes are predicted correctly and which it predicted incorrectly, and it forms the basis for calculating performance metrics.

The Confusion Matrix consists of four fundamental components in a binary classification task:

1. *True Positive (TP)*: Situations in which the model correctly classifies an observation as positive. An instance that is actually positive is also classified as positive by the model.
2. *True Negative (TN)*: Situations in which the model correctly classifies an observation as negative. An instance that is actually negative is also classified as negative by the model.
3. *False Positive (FP)*: Situations in which the model incorrectly classifies a negative observation as positive (Type I error). An instance that is actually negative is classified as positive by the model
4. *False Negative (FN):* Situations in which the model incorrectly classifies a positive observation as negative (Type II error). An instance that is actually positive is classified as negative by the model.

These components are represented in the Table 2.1 as follows:

Table 2.1 Confusion Matrix

		Predicted Values	
		Positive	**Negative**
Actual	**Positive**	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

This matrix serves as an essential tool for identifying the types of errors a model tends to make. Particularly imbalanced datasets, where accuracy metrics can be misleading, the Confusion Matrix allows for separate examination of the prediction performance for each class.

2.1.9.2 Recall

Recall [50], also referred to as true positive rate or sensitivity, measures how effectively the model identifies positive instances. It represents the proportion of all examples that are actually positive that the model correctly classified as positive. Recall

is particularly important in situations where the cost of false negatives (FN) is high (e.g., medical diagnosis, fraud detection). Its mathematical formula is as follows:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2.61}$$

The Recall value ranges between 0 and 1. A Recall value close to 1 indicates that the model has a low probability of missing the positive class. For example, a high Recall in a cancer diagnosis model means that a large proportion of cancer patients are correctly identified, which is crucial for early diagnosis and treatment. Though, evaluating Recall alone can be misleading; because a model could achieve high Recall by predicting all examples as positive, but in this case, the Precision value will likely be very low. Therefore, Recall is usually evaluated together with Precision.

2.1.9.3 Precision

Precision [50] is a metric that shows how many of the examples the model correctly predicted as positive are actually positive. It measures the reliability of the model's positive predictions. In particular, the Precision metric is of great importance in situations where the cost of false positives (FP) is high (e.g., spam email detection). Mathematically, it is expressed as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2.62}$$

The Precision value indicates that the majority of instances classified as positive by the model are actually positive. For example, in a spam email filter, a high Precision means that the likelihood of legitimate emails being incorrectly marked as spam is low. As with Recall, Precision alone is not sufficient. A model could achieve high Precision by predicting only a few instances as positive when it is very confident, but in that case, the Recall value would likely be very low.

2.1.9.4 Dice Similarity Coefficient

The Dice Similarity Coefficient (DSC) [51, 52], also known as the Sørensen-Dice coefficient or F1-Score, is a harmonic mean that measures the balance between Precision and Recall, particularly in binary classification problems. This metric is widely used, especially in image segmentation tasks, to evaluate the similarity between the predicted

segmentation mask and the ground true mask. The DSC provides a more reliable performance indicator in imbalanced datasets.

The mathematical formulas of the DSC are as follows:

$$\mathrm{DSC}(\mathbf{A}, \mathbf{B}) = \frac{2 \cdot |\mathbf{A} \cap \mathbf{B}|}{|\mathbf{A}| + |\mathbf{B}|} \tag{2.63}$$

where, **A** represents the set of pixels predicted by the model, while **B** represents the ground truth pixel set. When expressed in terms of the confusion matrix:

$$\mathrm{DSC} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{2.64}$$

This formula can also be expressed in terms of Precision and Recall:

$$\mathrm{DSC} = 2 \cdot \frac{\mathrm{Precision} \cdot \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \tag{2.65}$$

The DSC value ranges between 0 and 1. A value close to 1 indicates that the model both captures positive examples well (high Recall) and that most of its positive predictions are correct (high Precision). In image segmentation, the DSC measures the overlap between the predicted pixel set and the actual pixel set. A high DSC value indicates that the model can accurately determine the boundaries of objects and correctly classify both the interior and exterior of the object. This metric is preferred in fields such as biomedical imaging, where the accurate segmentation of lesions or organs is of vital importance.

2.1.9.5 Intersection Over Union

Intersection over Union (IoU) [53], also known as Jaccard Similarity, is a fundamental evaluation metric used particularly in object detection and image segmentation tasks. This metric measures the overlap ratio between the predicted bounding box or segmentation mask and the actual (ground truth) bounding box or mask. IoU is a specific version of the Jaccard index, which is used to evaluate the similarity and diversity between two sets. Its mathematical formula is as follows:

$$\mathrm{IoU}(\mathbf{A}, \mathbf{B}) = \frac{|\mathbf{A} \cap \mathbf{B}|}{|\mathbf{A} \cup \mathbf{B}|} \tag{2.66}$$

Its expression in terms of the confusion matrix is as follows:

$$\mathrm{IoU} = \frac{TP}{TP + FP + FN} \tag{2.67}$$

IoU and DSC are related and generally yield similar results. Both effectively measure the spatial accuracy of a model in segmentation tasks. The IoU and the DSC can be converted into one another as follows:

First, let

$$a = TP, \qquad b = TP + FP + FN \tag{2.68}$$

It follows that

$$\mathrm{IoU} = \frac{TP}{TP + FP + FN} = \frac{a}{b} \tag{2.69}$$

and

$$\mathrm{DSC} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} = \frac{2a}{a + b} \tag{2.70}$$

Hence, the DSC can be expressed in terms of the IoU as

$$\mathrm{DSC} = \frac{\frac{2a}{b}}{\frac{a+b}{b}} = \frac{2 \cdot \frac{a}{b}}{1 + \frac{a}{b}} = \frac{2 \cdot \mathrm{IoU}}{1 + \mathrm{IoU}} \tag{2.71}$$

When IoU is isolated, the following equation is obtained:

$$\mathrm{IoU} = \frac{\mathrm{DSC}}{2 - \mathrm{DSC}} \tag{2.72}$$

From this relationship, the following inequality can be derived:

$$\frac{\mathrm{DSC}}{2} < \mathrm{IoU} \leq \mathrm{DSC} \tag{2.73}$$

This inequality indicates that IoU is always bounded above by DSC, while remaining strictly greater than half of DSC.

2.1.10 Hyperparameter Optimization

Hyperparameters are parameters defined prior to model training and not directly learned during the learning process. In addition to the model architecture and parameters learned throughout training, hyperparameters significantly influence the performance of models [54]. They are adjustable configurations that directly affect a model's learning capability and generalization performance. Examples include learning rate, optimization algorithm, activation function, number of neurons per layer, number of layers, mini-batch size, regularization coefficients, and dropout rate [34].

Hyperparameter optimization refers to the process of finding the most optimal set of hyperparameters to maximize a model's performance on a given task [54]. This process typically involves training the model and evaluating its performance on a validation set. Optimal hyperparameters prevent the model from overfitting to the training data or underfitting, thereby improving its generalization ability to new and unseen data. Incorrectly chosen hyperparameters can slow down the model's convergence speed, cause it to get stuck in local minima, or generally lead to poor performance [55].

Optimizing hyperparameters is a vital step for the success of DL projects and is typically performed using trial-and-error, heuristic approaches, or more systematic search strategies [34]. These strategies include methods such as Bayesian Optimization [56], Random Search [55], and Grid Search [55]. Each method has its own advantages and disadvantages, and the selection depends on the size of the search space, the available computational resources, and the complexity of the task.

2.1.10.1 Grid Search

One of the most common and simplest approaches to hyperparameter optimization is Grid Search [55]. This method systematically searches through a predefined set of hyperparameter values, testing every possible hyperparameter combination and evaluating the performance of the model on the validation set. The goal is to find the hyperparameter combination that maximizes the specified performance metric (e.g., F1-Score).

Grid Search constructs a *grid* containing the values of each hyperparameter that will be tested within a specific range and at specific steps. For example, if the values [0.1, 0.01] are specified for the learning rate and [2, 3, 4] for the number of layers, Grid Search

tries all combinations of these values (0.1-2, 0.1-3, 0.1-4, 0.01-2, ...) one by one. For each combination, the model is trained and its performance is measured on the validation set. The combination achieving the best performance is identified as the optimal hyperparameter configuration.

Since it tests all combinations within the specified range, it has the potential to reach the global optimum (if the optimal values are within the specified range and steps). Because each combination can be tested independently, Grid Search can be easily parallelized, which can reduce computation time. As the number of hyperparameters and the number of values to be tested for each hyperparameter increases, the number of combinations to be tested increases exponentially. This can be time-consuming and computationally expensive, especially when there are many hyperparameters or wide ranges.

2.2 Segmentation Approaches

Many techniques and algorithms have been developed for segmentation of image [5]. These are commonly categorized into two main groups [7]: classical methods and DL based approaches.

2.2.1 Classical Methods

Classical image segmentation methods, also known as conventional methods, provide effective and straightforward solutions in scenarios with limited data and computational resource [7]. Techniques such as thresholding and clustering group pixels based on density, color, or texture, while region growing and edge-based techniques determine object boundaries [57]. Graph cut separates foreground and background by using energy minimization principles, whereas the watershed method bounds regions using a water-basin analogy [58]. Thanks to their simplicity and computational efficiency, these methods remain in use; however, they exhibit limited performance in complex and dynamic situations when compared to modern DL techniques [59].

2.2.1.1 SLIC + GrabCut

This approach begins by partitioning the image into regular, compact regions—superpixels—using the SLIC algorithm [60]. This step reduces computational load while

enhancing regional consistency. Hyperparameters such as the number of superpixels to be produced (N_{sp}) and compactness (c) are critical in determining the quality of region boundaries.

Next, the mean values of each superpixel are computed in the LAB color space. Each superpixel is scored using an intuitive color rule (bmax, combo, etc.) assumed to be relevant to the segmentation target (plate region). These scores are then used to build an automatic seeding mask for the GrabCut algorithm [61] which is a graph-based optimization algorithm. High-scoring superpixels are labeled as *foreground*, while others are labeled as *background*.

Finally, this automatic seeding mask and the image are fed as input to GrabCut. The algorithm refines the segmentation boundaries based on these seeds and produces the final mask.

2.2.1.2 HSV Based Thresholding

In this method, each pixel is classified as background or foreground (object) based on whether their color properties fall within a specific range [5]. The HSV (Hue, Saturation, Value) color space is often preferred for such applications, as it is more robust against variations in lighting conditions compared to the RGB color space [62]. Hue represents the color itself; Saturation represents the vividness of the color; and Value represents the brightness of the color. Since objects are typically within a specific Hue and Saturation range, thresholding based on these channels can yield effective results [63].

The method is fast, interpretable, and easy to implement, and it's particularly effective when the target class has a distinctive color range [63]. However, performance can be affected by illumination changes, white balance shifts, or color drift across scenes. Therefore, it is important to choose the hue range and saturation threshold appropriately based on the data and capture conditions.

2.2.1.3 Marker-Controlled Watershed

The Watershed Transform is a classic segmentation method that views a grayscale image as a topographical surface where pixel intensities represent altitude [64]. The algorithm, based on the analogy of *filling* basins with water on this surface, identifies the lines where different basins merge as object boundaries. However, the standard approach

is sensitive to noise and texture variations in the image, often leading to over-segmentation, i.e., the division of an object into numerous meaningless fragments.

The Marker-Controlled Watershed algorithm, developed to overcome this problem, guides the process using predefined markers [65]. These markers define a core region (foreground) known to belong to the object and a surrounding area (background) known not to belong to the object. The segmentation process is initiated only from these reliable regions, not from random local minima. This approach prevents over-segmentation and allows for more accurate boundary determination [65].

The implementation of the algorithm involves a multi-step pipeline. First, the image is prepared through preprocessing (e.g., noise reduction). Next—and most crucial—foreground and background markers are generated. Here, the foreground and background regions are automatically detected using the image's intensity characteristics. A gradient image is then computed to emphasize object boundaries, and the watershed transform is applied to this gradient under the guidance of the markers. Finally, the resulting raw segmentation mask is refined with post-processing techniques, such as morphological operations, to yield the final result.

2.2.1.4 Multi-Atlas Based Segmentation

Multi-Atlas based segmentation is a powerful classic technique that leverages a pre-labeled collection of images (an *atlas library*) to segment a target image. The core assumption is that anatomical structures exhibit comparable patterns across individuals [66]. The method transfers expert knowledge (i.e., manually annotated segmentation masks) from one or more atlases to a new, unlabeled target image.

The essence of the process lies in achieving a precise alignment between the target image and each atlas. This alignment is achieved through a critical step called *image registration*. Registration estimates the spatial transformation that geometrically aligns an atlas image with the target [67]. These transformations can range from simple geometric motions (rotation, scaling) to complex, flexible deformations. To increase accuracy and efficiency, this computationally intensive process is usually performed by selecting only the atlases that most closely resemble the target image.

Once the optimal alignment transformation is calculated for each atlas, it's directly applied to the atlas's segmentation mask. In this way, the labeled structures from the atlas are projected onto the target, producing multiple segmentation proposals. Combining

these often non-identical proposals into a single, more robust result is known as label fusion [68]. During fusion, algorithms such as simple Majority Voting heuristics or more advanced probabilistic fusion methods that account for observer reliability are used to decide the final label for each pixel.

Finally, the fused mask is refined through post-processing, typically using morphological operations, to produce a clean, usable segmentation. What makes Multi-Atlas based segmentation especially powerful is that it exploits the rich anatomical and structural prior information embedded in the atlas library, rather than relying solely on superficial features like color or intensity. This makes it particularly effective for segmenting complex and variable structures.

2.2.1.5 Graph-Cut with Shape Prior

Graph-Cut is a powerful classical method that approaches the image segmentation problem from an energy minimization perspective [69]. The algorithm assigns each pixel to a class by balancing two competing forces: fidelity to the observed image features (the data term, e.g., color or intensity) and the tendency to share consistent labels with neighboring pixels (the smoothness term). The point where this balance is optimally achieved defines the optimum segmentation boundary.

In addition to this core structure, the Graph-Cut with Shape Prior algorithm introduces a third layer of information that encourages the segmentation result to conform to an anatomically or structurally expected geometry [70]. This geometric expectation is typically incorporated into the energy function via a Signed Distance Function (SDF) learned from labeled examples in the training set. The SDF encodes each pixel's distance from the expected shape boundary, preventing the segmentation from deviating excessively from this prior information and enabling more consistent results.

The practical application of the algorithm begins with the automatic derivation of these energy terms from the image data. First, using the shape prior, a *target seed* is determined at the object's core and *background seeds* are determined at the image boundaries. These seeds are used to learn the intensity profiles of the foreground and background. These models form *the data term* that calculates the label probability of each pixel. At the same time, *the smoothness term*, which considers the gradient differences between neighboring pixels, allows the preservation of distinct edges in the image while promoting smooth segmentation in homogeneous regions. The total energy defined by all

these constraints is minimized using the s-t min-cut/max-flow algorithm to find the mathematically optimal solution. The resulting mask is then cleaned using morphological operations to achieve its final form.

2.2.2 Deep Learning Based Approaches

DL based segmentation methods utilize powerful neural networks to classify pixels, thereby delivering revolutionary contributions to the field of modern image processing [34]. CNNs [42] and architectures such as U-Net [71] offer detailed and precise segmentation, while methods like Mask R-CNN [72] and DeepLabV3+ [73] stand out for their object detection and learning capabilities. These approaches provide high accuracy and efficiency in various domains, including satellite analysis, autonomous vehicles, and biomedical imaging, surpassing traditional techniques [6]. The limited generalization performance of classical segmentation approaches, contrasted with the robust generalization exhibited by DL models trained on large datasets, has rapidly positioned DL solutions as the dominant approach in the field of segmentation [59].

CNN-based networks commonly employ an encoder-decoder architecture that filters spatial information from images. In this framework, the encoder extracts features by encoding the image into low-dimensional representations, while the decoder utilizes these features to reconstruct a high-resolution output, generating the segmentation mask in the process [34]. Because objects' distinguishing characteristics may occur in any part of the image, convolutional filters are applied across all image regions during model training.

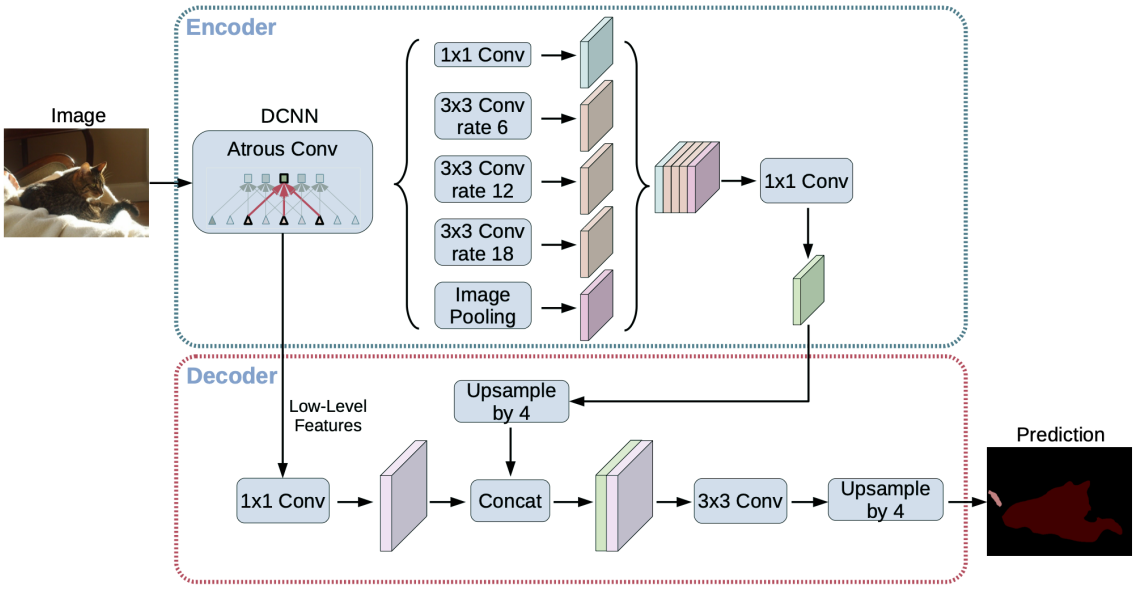


Figure 2.5 The architecture of DeepLabV3+ [73]

2.2.2.1 DeepLabV3+

DeepLabV3+ [73] is a widely adopted DL architecture developed for semantic image segmentation, whose purpose is to classify each individual pixel by its corresponding class (for architecture, see Figure 2.5). Building upon its predecessor, DeepLabV3, this model introduces an encoder-decoder structure that significantly improves the delineation of object boundaries while preserving the ability to capture rich contextual information. The encoder utilizes atrous, or dilated, convolutions to enlarge the receptive field while preserving spatial resolution and maintaining parameter efficiency. This design choice enables the model to capture multi-scale contextual features efficiently. At the core of the encoder lies the Atrous Spatial Pyramid Pooling (ASPP) module, which applies parallel atrous convolutions with different dilation rates to effectively aggregate contextual information at multiple scales. The decoder component of DeepLabV3+ is relatively simple yet effective; it refines the coarse segmentation maps generated by the encoder by incorporating low-level features from earlier layers of the backbone network. This fusion of low-level spatial information with high-level semantic features enhances the accuracy of object boundaries, which is particularly important for precise segmentation tasks. The model architecture is typically built upon powerful backbone networks such as Xception or ResNet, further enhancing its performance. Due to its robustness and flexibility, DeepLabV3+ has been successfully applied to various practical situations, including autonomous driving, biomedical image analysis, satellite image interpretation, and other domains requiring fine-grained pixel-

level classification. Its ability to balance accuracy and computational efficiency makes it a widely adopted and highly influential model in the field of computer vision.

2.2.2.2 Mask R-CNN

Mask R-CNN [72] is a DL framework for instance segmentation, capable of both detecting objects in an image and producing accurate segmentation masks for each individual instance (Figure 2.6). It is an extension of the Faster R-CNN model, enhancing its capabilities by adding a parallel branch that predicts segmentation masks for each detected object. The model architecture typically includes a convolutional backbone such as ResNet, often combined with a Feature Pyramid Network (FPN), to extract multi-scale features from the input image. A Region Proposal Network (RPN) then identifies candidate object regions, which are refined through RoIAlign—a technique that accurately preserves spatial alignment during feature pooling. Each Region of Interest (RoI) is then processed to produce three outputs: the object class, bounding box coordinates, and a binary mask that represents the shape of the object at the pixel level. By combining object detection and segmentation in a unified framework, Mask R-CNN provides a comprehensive solution for tasks that require both localization and detailed object boundaries.

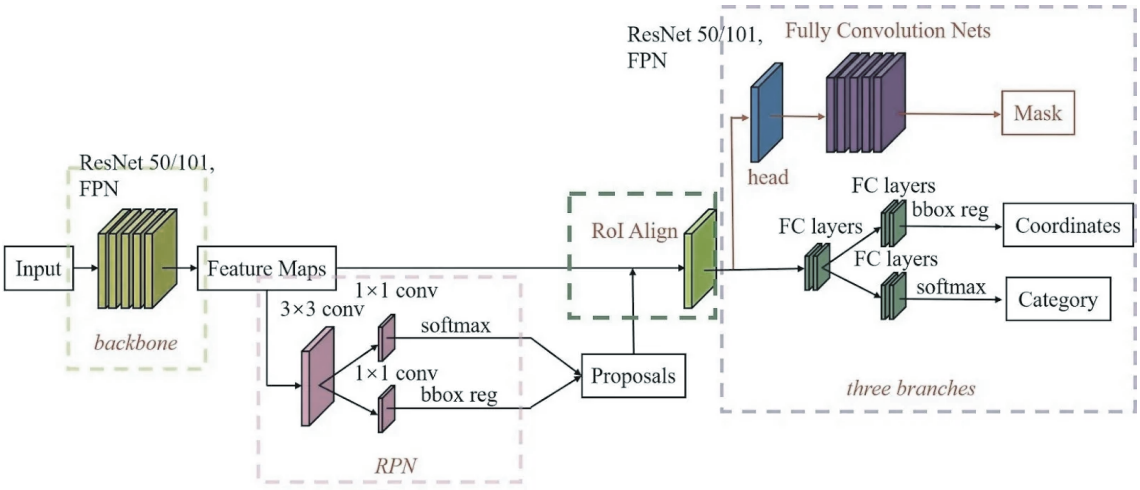


Figure 2.6 The architecture of Mask-R-CNN [74]

2.2.2.3 SegResNet

SegResNet [75], is a modern and robust semantic segmentation model, particularly well-suited for sensitive applications like biomedical image analysis. Its architecture combines two proven approaches: a deep Residual Network (ResNet) and an encoder-

decoder structure. The encoder extracts high-level semantic features from the input image, while the decoder uses these features to generate a precise segmentation map at the original image resolution.

The core of the model consists of ResNet blocks, which solve the *vanishing gradient* problem. This allows the network to be much deeper, enabling it to learn complex patterns and ensuring an efficient training process. One of the most critical elements of the architecture is the skip connections. These connections transfer detailed spatial information (like edges and textures) from the early layers of the encoder directly to the decoder. This mechanism significantly improves the accuracy of segmentation boundaries and overall precision.

Thanks to these architectural advantages, SegResNet is a flexible and reliable solution for challenging tasks in 2D and 3D medical imaging, such as delineating tumors, lesions, or organs. The model effectively combines the power of DL with a smart architecture to produce highly accurate results.

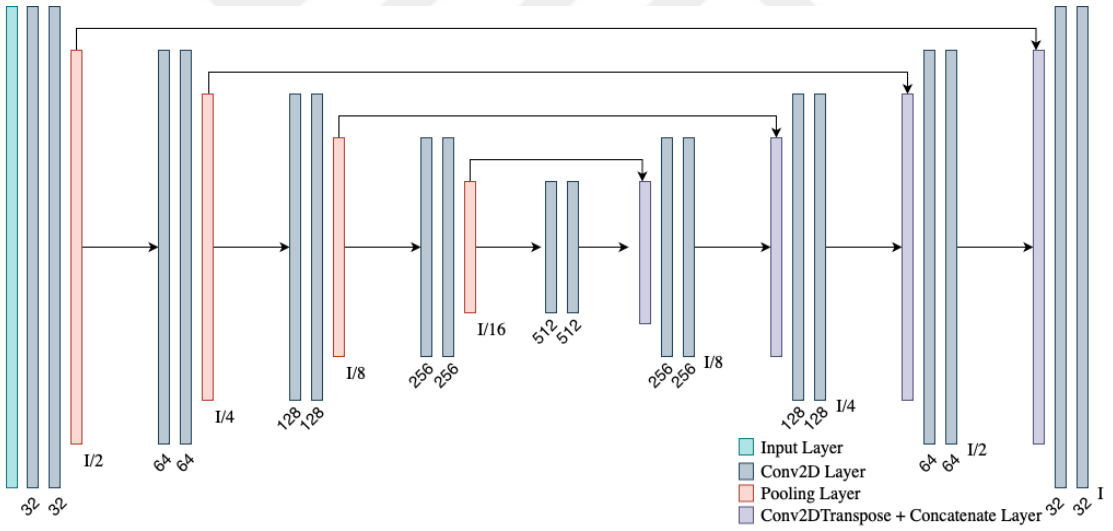


Figure 2.7 The architecture of U-Net

2.2.2.4 U-Net

U-Net [71] is a specialized convolutional neural network architecture originally designed to address the challenges of biomedical image segmentation, and it has since become one of the most widely used models for dense prediction tasks (see Figure 2.7). Its architecture follows a symmetrical encoder–decoder structure that enables the network to capture both coarse semantic context and fine-grained spatial features. The encoder, or contracting path, is composed of repeated convolutional operations followed by ReLU

activations and max pooling, which gradually decrease spatial resolution while increasing the depth of the feature maps. This process allows the model to extract rich contextual features from the input image. The decoder, or expansive path, performs upsampling operations through transposed convolutions and concatenates the resulting feature maps with corresponding feature maps from the encoder via skip connections. These skip connections are a defining feature of U-Net, as they enable the direct transfer of spatial information from the encoder to the decoder, significantly improving localization accuracy. The final layer of the network applies a 1×1 convolution to produce a pixel-wise classification map. U-Net is fully convolutional and can operate on images of arbitrary size, making it highly versatile. Its ability to deliver high segmentation performance even with limited annotated data, combined with its relatively simple yet powerful design, has made it a standard model in medical imaging, satellite imagery analysis, and various other domains requiring precise segmentation.

2.2.2.5 Super Vision U-Net

Super Vision U-Net [76] is an advanced variant of the traditional U-Net architecture, designed to enhance semantic segmentation performance by integrating mechanisms that improve spatial detail preservation and feature representation (see Figure 2.8). While preserving the vanilla U-Net's basic encoder-decoder structure and skip connections, Super Vision U-Net incorporates additional refinement modules that allow for more effective feature fusion and multi-scale representation learning. A key enhancement in this architecture is the integration of deep supervision strategies that guide the model toward learning more discriminative features at different stages of the decoding process. These additions enable the network to concentrate on the most significant spatial and contextual information, particularly in regions where object boundaries are ambiguous or fine-grained details are critical. Furthermore, the architecture may utilize residual connections, dense blocks, or multi-scale feature extraction modules to further increase the model’s expressive power. By doing so, Super Vision U-Net improves upon the limitations of classical U-Net in capturing fine structures and reducing over-smoothing in segmentation maps. This makes it an excellent choice for segmentation tasks in complex visual environments, such as satellite data analysis, medical imaging, and industrial inspection.

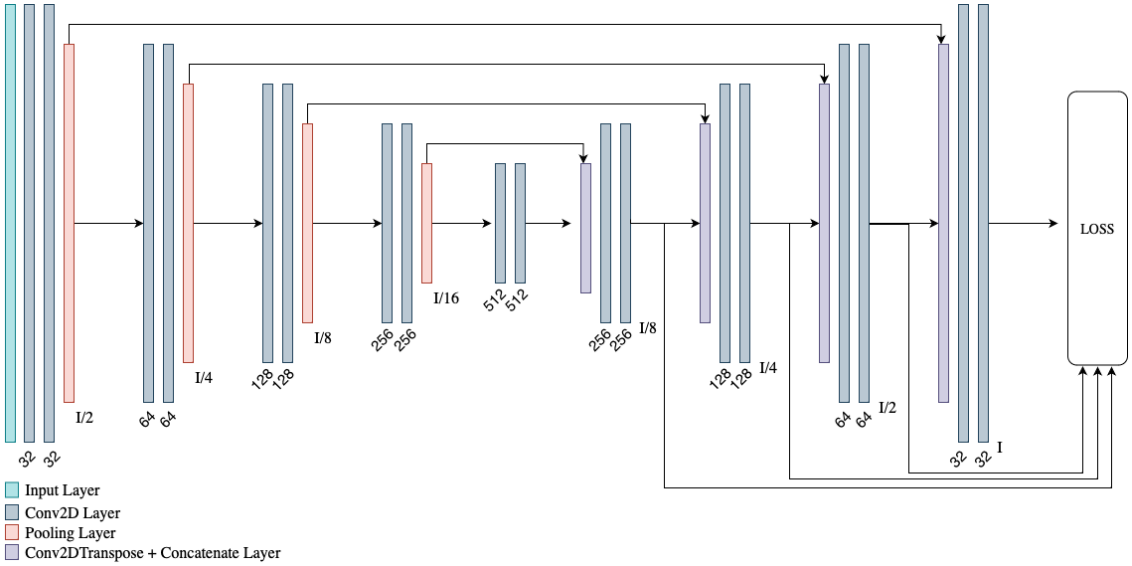


Figure 2.8 The architecture of Super Vision U-Net

2.2.2.6 U-Net Transformer

U-Net Transformer [77] is a hybrid DL model that integrates the advantages of convolutional neural networks (CNNs) and Transformer models for biomedical image segmentation and other dense prediction tasks. It builds upon the classical U-Net structure by replacing or augmenting parts of the encoder with a Transformer module to better capture long-range relationships and global context within the data. In this architecture, the input data is first handled by a series of convolutional layers to extract low-level spatial features, which are then flattened and passed to a Vision Transformer (ViT) that models global relationships between image patches using self-attention mechanisms. The Transformer encoder's output is reshaped and fed into a decoder path, similar to that of traditional U-Net, where upsampling and skip connections from earlier convolutional layers help recover spatial resolution and local detail. This combination allows the model to overcome the locality limitations of CNNs while preserving their efficiency in capturing fine-grained features. By leveraging both global and local context, U-Net Transformer achieves superior segmentation accuracy, particularly in tasks that involve complex structures or subtle anatomical variations.

2.2.2.7 Attention U-Net

Attention U-Net [78] is a DL model that enhances the standard U-Net architecture with attention mechanisms to achieve more accurate results in tasks such as biomedical image segmentation. The fundamental innovation of this architecture is the *attention*

gates integrated into the skip connections between the encoder and decoder paths. In traditional U-Net, feature maps from the encoder are directly passed to the decoder, which can result in irrelevant or noisy background features being carried over. In Attention U-Net, attention gates learn to focus on important regions in the feature maps from the encoder and suppress irrelevant regions. This allows the model to concentrate its resources only on objects that are critical for segmentation. As a result, Attention U-Net demonstrates higher performance than standard U-Net, particularly in the segmentation of small and complex structures, by reducing false positives and determining object boundaries more accurately.

2.2.2.8 UNet++

UNet++ [79], also known as Nested U-Net, is an advanced version of the U-Net architecture designed to bridge the semantic gap between the encoder and decoder paths. The most distinctive feature of this model is its redesigned skip connections. Instead of the direct and long skip connections found in the standard U-Net, UNet++ uses a network composed of nested and dense convolutional blocks. This structure more flexibly combines low-level, fine-grained feature maps from the encoder with high-level, semantically rich feature maps in the decoder. This gradual feature fusion facilitates the network's optimization process and enables it to produce more accurate segmentation results, particularly at object boundaries.

2.2.2.9 YOLO

YOLO (You Only Look Once) [80] is a groundbreaking approach in DL-based object detection. Traditionally, object detection was performed using two-stage systems that first identified potential object regions in an image and then classified these regions. While these methods produced accurate results, they were often computationally intensive and not fast enough for real-time applications. YOLO solved this problem by offering a unified, single-stage architecture that analyzes an image in a single pass and directly predicts the locations and classes of objects. This *single-shot* approach revolutionized object detection speed, making the algorithm ideal for low-latency applications such as autonomous vehicles, security cameras, and robotics.

At its core, YOLO is based on the principle of dividing the input image into a fixed-size grid (e.g., 7x7). If the center of an object falls within one of these grid cells, that cell becomes responsible for detecting the object. Each grid cell makes a certain number of

bounding box predictions. For each bounding box, a confidence score indicating whether the box contains an object and a probability distribution indicating which class the object belongs to are predicted. The confidence score reflects both the likelihood that the box contains an object and how well the predicted box overlaps with the actual object. This way, YOLO determines both where the object is and what it is at the same time and in a single operation.

2.2.2.10 Dynamic U-Net

DynUNet is a segmentation backbone that preserves the encoder–decoder structure of the U-Net architecture while dynamically adjusting depth, kernel sizes, and stride sequences according to the spatial characteristics of the data.

At each stage, residual blocks, normalization, and nonlinearities ensure stable gradient flow. The decoder combines multi-scale features with skip connections; it recovers boundary details by upscaling via transposed convolution or interpolation+convolution. Deep supervision support accelerates learning and improves convergence by producing auxiliary outputs at intermediate scales; full-resolution headers are used for prediction.

The architecture maintains a compute-memory balance by incrementally increasing and symmetrically decreasing channel widths; it easily adapts to different patch sizes and resolutions. This flexibility offers high accuracy, stable training, and practical deployment advantages for 2D/3D binary or multi-class segmentation tasks.

2.3 Overview of Medical Imaging Modalities

Understanding the internal structures and functions of the body is crucial for identifying diseases and determining appropriate treatments [81]. Medical imaging is a field dedicated to visualizing the body's internal anatomy and physiological processes. It assists physicians in diagnosing diseases and monitoring treatment progress [82].

Different medical imaging methods operate on different principles and each has its own specific applications. For example, while some methods are more effective in visualizing bones, others provide clearer insights into chemical processes or soft tissues within the body. In this thesis, three main types imaging modalities were utilized: Intraoral Camera, Magnetic Resonance Imaging (MRI), and PET/CT [82].

2.3.1 Intraoral Camera

Intraoral cameras [83] are specialized devices used in dentistry. They have a pen-like structure. They feature a lens and a light source for illumination at their tip. This design allows for clear imaging of even hard-to-reach areas of the mouth. They take high-resolution, color, and magnified photographs. These images show the surface of the teeth and gums in detail and help detect tooth decay, cracks, the condition of old fillings, and gum problems. The images may vary in quality due to glare, saliva, and blurring.

These cameras enhance communication between the patient and the dentist. Patients can clearly see the problems in their mouths on the screen. This increases their participation in the treatment process. In addition, the photos taken are used to document the treatment process. The before and after of the treatment can be compared. This method is very useful for detecting structures such as dental plaque. It clearly reveals differences in color and texture. Therefore, it is an ideal imaging tool for dental plaque segmentation studies.

2.3.2 Magnetic Resonance Imaging

Magnetic Resonance Imaging (MRI) is the most widely used method in clinics for soft tissue differentiation. This method uses a large magnet and radio waves instead of harmful rays such as X-rays. This makes it a safe method, especially for repeated scans [82].

MRI primarily targets water molecules in the body. The magnetic field aligns the protons within these molecules. Radio waves are then sent to momentarily disrupt this alignment. When the signal is turned off, the protons return to their original alignment. During this return, they emit an energy signal into the environment. The device's sensitive receivers collect these signals. Different tissues contain water at different rates. Therefore, different signals come from different tissues. The computer processes these signals to generate detailed, cross-sectional images.

MRI is particularly superior at displaying soft tissues. The brain, spinal cord, muscles, and organs are visualized with great clarity. It is the gold standard for diagnosing brain tumors [26]. It clearly shows the location, size, and relationship of the tumor to surrounding tissues.

2.3.3 Positron Emission Tomography / Computed Tomography

Positron Emission Tomography/Computed Tomography (PET/CT) is a hybrid imaging method that combines nuclear medicine and radiology. It provides both anatomical and metabolic information in a single scan. The fusion of these two different types of modalities is extremely powerful in diagnosing diseases. The device incorporates a CT scanner and a PET scanner within the same system [84].

Computed Tomography (CT) uses X-rays to generate detailed anatomical images of the body. Multiple cross-sectional images are taken from different angles of the body. These *slices* are combined by a computer to form a three-dimensional map. This map reveals the shape, size, and location of organs and tissues [81].

Positron Emission Tomography (PET) shows the body's biological functions. A radioactive tracer is administered to the patient intravenously. This substance mimics a specific molecule in the body [84]. For prostate cancer, this is usually a Gallium-68 (Ga-68) tracer that binds to a protein called PSMA (Prostate-Specific Membrane Antigen) [20]. Prostate cancer cells contain much more PSMA protein on their surface than normal cells. Therefore, the radioactive tracer accumulates intensely in these cancerous cells. The PET scanner detects the positrons emitted from these accumulation areas. These areas appear as bright *hot spots* in the images.

A PET/CT scan overlays these two images, resulting in the precise localization of cancerous activity highlighted by PET onto the anatomical map provided by CT. This integration provides valuable information for the initial diagnosis of prostate cancer, determining its spread (staging), and evaluating response to treatment. Therefore, Ga-PSMA PET/CT scans constitute a rich and meaningful dataset for prostate segmentation [85].

2.4 Explainability

DL models often operate like a *black box*: they take an input, make a decision, but do not reveal *why* they made that decision [86]. This is not sufficient in high-risk areas such as clinics. Physicians, technicians, and patients want to see the logic behind the decision in order to trust the model [87]. Explainable AI (XAI) is a field that seeks to answer precisely these questions. The primary goal of XAI is to make the inner workings, decision-making processes, and outcomes of AI models, particularly DL models,

understandable to humans. This is of great importance not only for technical experts but also for end users who are affected by the model's outputs or who must rely on them (e.g., physicians, financial experts, lawyers). Understanding the reasons behind a model's decisions increases trust in the model, facilitates the detection of potential errors or biases, and, most importantly, paves the way for the more effective and responsible integration of AI into real-world problems [87]. This section discusses why explainability is critical in clinical practice and addresses the most commonly used visualization approaches in practice, namely *Grad-CAM*, and *Saliency Maps*.

2.4.1 The Importance of Explainability in Clinical Practice

The applications of DL models in the healthcare domain are rapidly increasing. From medical imaging (radiology, pathology) to patient data analysis, from drug discovery to personalized treatment, AI provides powerful tools to physicians and researchers in many areas [19]. However, in this critical field, it is not enough for a model to simply make accurate predictions; it is also vital that it can explain why it made that prediction [86]. Here are the fundamental reasons why explainability is so important in clinical settings:

Building Trust [88]: Physicians carry great responsibility when making decisions that affect patients' lives. In order to trust the recommendation of an AI model, they need to understand the rationale behind it. Knowing what findings the model relies on when making a diagnosis or recommending a treatment increases clinicians' trust in the model and allows them to combine it with their own expertise. An unexplained *black box* model, no matter how accurate, cannot be easily adopted in clinical practice.

Error Detection and Bias Mitigation [86]: DL models can learn biases present in the data they are trained on and reflect these biases in their decisions. For instance, they may make biased diagnoses against patients belonging to a certain ethnic group or gender. Explainability techniques reveal which features the model relies on when making decisions, thereby helping to identify such biases. When the model makes an incorrect decision, explainability tools help us understand why the error occurred, which in turn guides the model's improvement.

Legal and Ethical Requirements [88]: The healthcare sector is subject to strict legal and ethical regulations on matters such as data security, patient privacy, and treatment responsibility. With the introduction of AI systems into clinical practice, the decisions

made by these systems must be transparent and auditable. Regulations such as the European Union's Artificial Intelligence Act explicitly stipulate explainability requirements for AI systems used in high-risk domains such as healthcare [89].

Integration into Clinical Decision-Making Processes [90]: Explainable models enable physicians to compare AI recommendations with their own clinical judgment. The explanations provided by the model can help clinicians validate their own diagnoses or gain a different perspective. This integration enables more informed and comprehensive clinical decisions.

Potential for New Medical Discoveries [87]: Sometimes, a DL model can uncover complex relationships or patterns that human experts overlook or fail to notice. Explainability techniques can reveal how the model makes such new discoveries, contributing to the development of new hypotheses in medical science and a better understanding of diseases.

In summary, explainability is an indispensable element for AI to reach its maximum capability and achieve widespread adoption in clinical practice. Reliability, transparency, and accountability are the cornerstones of AI applications in healthcare.

2.4.2 Explainability Techniques

Various techniques have been developed to interpret the internal mechanisms of DL models. These techniques attempt to visualize or quantify which parts of the input the model focuses on when making a decision, or which features are more influential in the decision.

2.4.2.1 Gradient-Weighted Class Activation Mapping

Gradient-weighted Class Activation Mapping (Grad-CAM) [91] is a popular explainability technique developed specifically for CNN based segmentation models. Its purpose is to visualize which regions of an image the CNN attends to, or which areas contribute most significantly, when making a specific classification decision. As the name suggests, Grad-CAM computes the gradients of the feature maps in the final convolutional layer of the model [91]. These gradients show how much each feature map contributes to the activation of the corresponding class. The gradients are then used as weights for the corresponding feature maps, and the weighted maps are aggregated to

produce a heatmap. When overlaid on the original image, this heatmap visually highlights the regions most influence the decision of the model for that class.

2.4.2.2 Saliency Maps

Saliency Maps [92] are another explainability technique that shows which pixels or features of the input a DL model relies on most when producing an output. They typically work by identifying which pixels in the input require the least change to alter the model's prediction. In other words, it highlights which pixels are *salient* or *important* when the model makes a decision.

The basic principle of generating a Saliency Map is to calculate the gradient of the output of the model score (e.g., the probability of a certain class in a classification model) with respect to each pixel in the input [92]. Pixels with high gradient values have a greater impact on the model's output. These gradient values are typically visualized as a grayscale heatmap at the original image size. This heatmap highlights the most critical pixels or regions that influence the decision of the model.

Chapter 3

Microbial Dental Plaque Segmentation

3.1 Introduction

Dental plaque is defined as a microbial community embedded in a matrix composed of polymers derived from bacteria and the content of saliva that develops on the surface of the teeth [93]. Microbial dental plaque is adsorbed onto the tooth surface within seconds after dental cleaning and persists functionally [94]. These molecules primarily exist in the fluid of the subgingival sulcus, along with saliva, and demonstrate settlement in this area [95, 96]. The primary etiological factor for gingivitis and periodontitis is bacterial plaque, which can lead to the destruction of gingival tissues and periodontal attachment [97]. In children, inadequate oral hygiene following tooth eruption and the absence of regular brushing habits facilitate the accumulation of bacterial biofilm on tooth surfaces and gingival margins. This accumulation, in turn, triggers gingival inflammation and increases the risk of periodontal disease [98].

The early detection and treatment of periodontal diseases in children are clinically crucial, as these conditions can progress and lead to adverse long-term outcomes [99]. Bacterial plaque is the primary etiological factor in gingival diseases among pediatric patients. However, identifying and distinguishing microbial dental plaque can be challenging for patients. Plaque detection is commonly performed in routine clinical practice using periodontal probes and/or plaque-disclosing solutions. While these methods are widely utilized, they may produce subjective results [100]. Moreover, such assessment techniques can be cumbersome, time-consuming, and ineffective, particularly in uncooperative children. Additionally, plaque-disclosing solutions may cause the temporary staining of the oral mucosa and lips, which can be a disadvantage in clinical practice.

Several plaque assessment indices, such as the Plaque Index [101] and the Quigley–Hein Index [102], are widely employed to evaluate dental plaque accumulation. However, these indices rely on subjective evaluations, limiting their consistency and reproducibility. In contrast, deep learning (DL) –based artificial intelligence (AI) models offer a more objective and automated approach to plaque detection. Manual plaque assessment is not only time-intensive but also prone to human error, particularly in high-volume clinical settings.

To address these challenges, digital technologies, including intraoral scanners and fluorescence-based techniques, have been explored for three-dimensional plaque detection. Additionally, advanced digital imaging methods, such as laser-induced autofluorescence spectroscopy and HIS color space analysis, have been investigated for microbial dental plaque detection. However, the widespread adoption of these techniques is hindered by limitations such as high equipment costs and the need for technical standardization [103, 104].

For these reasons, this study aims to develop an affordable and easily accessible DL–based AI model for the early and accurate diagnosis of microbial dental plaque in children. The aim is to prevent various periodontal problems and provide motivation for oral hygiene by evaluating the diagnostic and detection performance of this AI model.

With advancements in AI and image processing, research on plaque detection, segmentation, and quantification in dental camera images has gained momentum [105–115]. However, despite these advancements, plaque detection and segmentation using AI have not yet become a gold standard. This has encouraged researchers to conduct further studies in this field. Therefore, this study aims to propose an AI model for predicting dental plaque in young permanent teeth of children. Finally, to evaluate the clinical feasibility of the model, statistical hypothesis tests are conducted to compare its predictions with assessments made by three experienced dentists.

3.2 Related Works

Several recent studies have explored DL architectures for dental plaque segmentation, each reporting varying degrees of success. Sudheera et al. [106] attempted to detect dental plaque using Enhanced K-Means, which is a machine learning algorithm. Imangaliyev et al. [107] proposed a Convolutional Neural Network (CNN)-based DL

model to classify dental red autofluorescence plaque on quantitative light-induced fluorescence images, achieving an F1 score of 0.75 on the test dataset. Liu et al. [108] developed a Mask R-CNN-based dental health IoT platform to classify seven different oral diseases, including dental plaque. Their platform has a 100% accuracy rate for dental plaque recognition, but not for segmentation. Li et al. [109] presented a novel low-shot learning method for dental plaque segmentation using oral endoscope images. It conducts low-shot learning at the super-pixel level, integrating local-to-global features for accurate pixel-level segmentation. The mean Intersection over Union (mIoU) score of their model is 0.8585. In their subsequent work [111], they improved their model with a novel network featuring a self-attention module, achieving an mIoU score of 0.7364. You et al. [110] utilized DeepLabV3+ for segmenting dental plaque in primary teeth and achieved a clinically acceptable mIoU of 0.726 compared to a pediatric dentist. Similarly, Yüksel et al. [112] applied DeepLabV3+ on intraoral images from pediatric patients, surpassing a dentist's performance by achieving an IoU of 0.76, whereas the dentist achieved 0.71. Nantakeeratipat et al. [113] focused on classifying plaque severity with Google Cloud Vertex AI AutoML, reporting 90.7% overall accuracy and notably high precision (98.3%) for the heavy plaque class. Chen et al. [114] combined YOLOv8 for tooth detection, the Segment Anything Model for segmentation, and a CNN-based system called DeepPlaq for classification, achieving 94.1% accuracy in tooth detection and 84% in plaque classification. Another approach is Song et al.'s CenterFormer [115], a transformer-based model integrating Cluster Center Encoder and Multiple Granularity Perceptions, which attained 60.91% IoU and 76.81% pixel accuracy, particularly benefiting low-contrast and variable plaque appearances.

3.3 Materials and Methods

In this study, a privately collected dataset was used to train six AI models, incorporating variations of the U-Net architecture, for the prediction of dental plaque in the young permanent teeth of children. These models' performance was then systematically assessed and comprehensively analyzed. Finally, to assess the clinical applicability of the best-performing model, statistical hypothesis tests were conducted to compare its predictions with the assessments of three experienced dentists and a classical method.

Table 3.1 Eligibility criteria: inclusion and exclusion factors.

Type	Criteria
Inclusion Criteria	Children aged 8–13 years Individuals in the mixed dentition phase Patients presenting at the Pediatric Dentistry Clinic, Hamidiye Faculty of Dental Medicine Patients randomly selected from those attending routine dental check-ups at a public hospital Individuals who were unaware of the study beforehand (i.e., had not received any prior oral health education or motivation)
Exclusion Criteria	Anterior young permanent teeth with enamel defects such as caries, hypoplasia, or hypomineralization Teeth with restorations or prosthetic treatments Young permanent teeth located in the posterior region Primary teeth

3.3.1 Dataset Collection

During the five-month data collection period, 31 pediatric patients aged 8 to 13 years, in the mixed dentition phase, were included in the study based on predefined inclusion criteria. These patients presented at the Pediatric Dentistry Clinic of Hamidiye Faculty of Dental Medicine, University of Health Sciences, and were randomly selected from those attending routine check-up appointments at a public hospital. Importantly, participants were not previously informed about the study and had not received any oral or dental hygiene education or motivational training prior to their participation.

Exclusion criteria included anterior young permanent teeth with enamel tissue integrity disruptions, such as decay, hypoplasia, and hypomineralization, as well as restored or prosthetically treated teeth. Additionally, young permanent teeth in the posterior region and primary teeth were not included in the study (see Table 3.1).

The Scientific Research Ethics Committee of Hamidiye at the University of Health Sciences granted approval for this investigation, and legal guardians provided informed consent on behalf of the children. The official trial protocol is publicly available on ClinicalTrials.gov under the ID NCT06603233.

After the teeth were isolated from saliva using cotton rolls, a saliva ejector, and an air spray, their initial photographs were captured using an intraoral camera (1280 × 720 pixels, TPC Ligang, China). To eliminate a deficiency in the study of You et al., images of the relevant tooth taken from different angles were also included in the dataset [110]. Subsequently, to reveal dental plaque, patients were instructed to chew a dental plaque-disclosing tablet (TePe-PlaqSearch™, Malmö, Sweden), which changes to a pink color,

making the presence of plaque visible. Following this, the teeth were photographed for the second time from the same angles as the initial images.

The dataset utilized for this research was developed based on the O'Leary Plaque Score Index, a widely recognized and standardized method for assessing dental plaque accumulation [116]. To ensure consistency and reliability, all plaque annotations were performed following this index's criteria. Plaque-disclosing agents were applied to the dental surfaces, and plaque presence was evaluated on three surfaces per tooth (mesial, distal, and buccal). Based on this assessment, the dataset was labeled in a binary manner as "plaque present" or "plaque absent". These labeled images were then used to train the DL model, ensuring that it was developed using clinically validated ground truth data.

Each photograph was cropped so that only a single complete tooth was visible in the image. After that, the plaque on the teeth in the initial photographs was annotated using the Visual Geometry Group (VGG) Image Annotator tool (version 2.0.11) by a specialized dentist based on the second set of photographs where the dental plaque was revealed [117]. The ground truth masks generated from the output of the VGG tool were used as labels during the training of DL models. Finally, the study included 506 photographs involving young permanent anterior teeth with corresponding ground truth masks. The step-by-step workflow illustration of generating dataset is given in Figure 3.1.

The dataset is randomly divided into three sets: ~70% training (354 images), ~15% validation (73 images), and ~15% test (79 images). This division follows a tooth-based approach to ensure that images captured from different angles of the same tooth are included within the same subset. The 70-15-15 split is chosen as it provides an optimal balance for small to medium-sized datasets, allowing the model to learn effectively while ensuring sufficient data for validation and testing without compromising generalization.

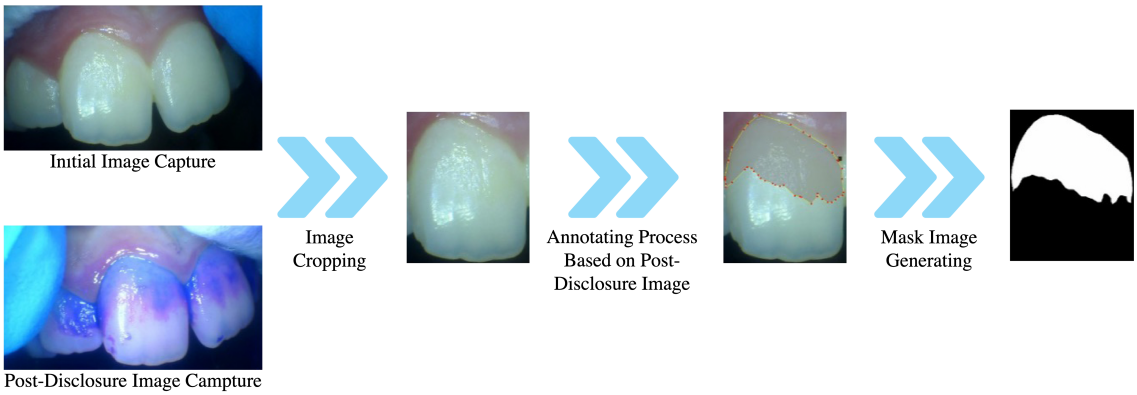


Figure 3.1 Flowchart of generating mask image

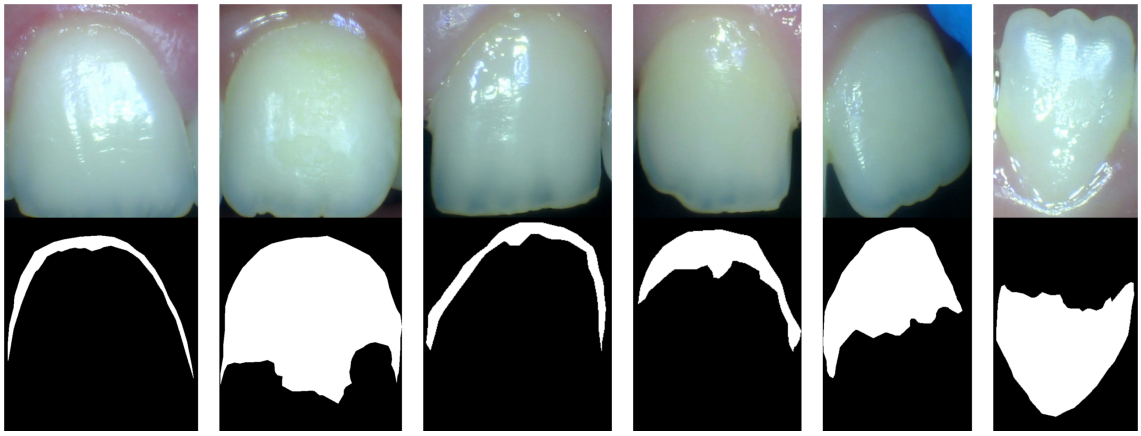


Figure 3.2 Dataset samples of intraoral images and ground truth masks.

Samples from the dataset are illustrated in Figure 3.2. The first row contains intraoral input images, and the second row presents the corresponding manually annotated ground truth masks for dental plaque segmentation.

3.3.2 Applied Classical Methods

For comparison with DL approaches, two classical baselines were implemented: SLIC superpixels with GrabCut (SLIC-GC) and HSV-based thresholding (HSV-T). Implementations used Python (3.10) with OpenCV (4.11) and Scikit-Image (0.23). Because classical methods are sensitive to parameterization, hyperparameter optimization was conducted exclusively on the training set via a systematic search; after selecting the optimal configuration, a single evaluation on the test set was performed. For each configuration, Recall, Precision, Dice Similarity Coefficient (DSC), and IoU were recorded.

HSV-T: Hue bounds were swept as $H_{\text{low}} \in [0,25]$ and $H_{\text{high}} \in [20,50]$ with step 5; invalid pairs with $H_{\text{low}} \geq H_{\text{high}}$ were skipped. The saturation threshold (S_{low}) was chosen from {0.15, 0.30, 0.45, 0.60}, and the median-blur kernel size (k_{blur}) from {0,3,5} (with 0 indicating no blur).

SLIC-GC: The number of superpixels $N_{\text{sp}} \in \{100,200,300,500\}$, compactness $c \in \{5,10,20\}$, gradient Gaussian $\sigma_g \in \{0,1\}$, median-blur kernels $k_{\text{blur}} \in \{0,3\}$, and morphological-closing kernels $k_{\text{morph}} \in \{0,3\}$ were explored. For each image, SLIC labels were computed and region scores were derived in CIELAB using four heuristics—bmax, lmin, aabsmax, and combo—to generate GrabCut seeds. Object/background seeds were selected via quantiles ($q_{\text{object}} = 0.75, q_{\text{BG}} = 0.25$) with a minimum seed size of 50

pixels. GrabCut was then run with $n_{\mathrm{GC}} \in \{2,3\}$; if $k_{\mathrm{morph}} > 0$, binary closing was applied to remove small gaps.

3.3.3 Applied Deep Learning Models

As DL models, DeepLabV3+, Mask R-CNN (Detectron2), YOLOv8, U-Net, Super Vision U-Net and U-Net Transformer models were selected. The implementation of these models was carried out in Python (version 3.10) with the TensorFlow (version 2.11) and PyTorch (version 1.12) libraries. Due to the insufficient size of the training set containing 354 images for the training of these 6 models, data augmentation techniques were employed over the training dataset during training to increase this number.

For hyperparameter optimization, a grid search method was employed to tune the input image size (128, 192, and 256), batch size (2, 4, and 8), learning rate (1×10^{-2}, 1×10^{-4}, 1×10^{-6}), and optimization algorithm (RMSProp, SGD, and Adam) based on the performance of the model on the validation dataset. The Detectron2 framework was used to implement the Mask R-CNN. However, since Detectron2 does not allow modifications to the optimization algorithm, SGD was used as the default optimizer. Additionally, the baseline models in Detectron2 (R101-C4, R50-C4, R50-DC5, and R50-FPN) were also optimized as part of the hyperparameter tuning process. The training epochs and the patience parameter for early stopping were uniformly set to 120 and 15, respectively, across all models.

3.3.4 Statistical Analysis

Using the prior knowledge ($\alpha = 0.05$, $\beta = 0.2$) and an effect size of 0.61, the actual power of the comparison between U-Net Transformer and dentists on 34 test images is at least 80%, which is deemed sufficient. Therefore, 35 randomly selected images on the test dataset were labeled by three dentists without seeing the ground truth and were predicted by U-Net Transformer. Then, the IoU score of these labeled and predicted images were calculated. To confirm clinical feasibility and to compare with classical segmentation method, four *t*-tests were performed on the IoU scores, and a p value < 0.05 was considered statistically significant. The workflow diagram of the study is depicted in Figure 3.3.

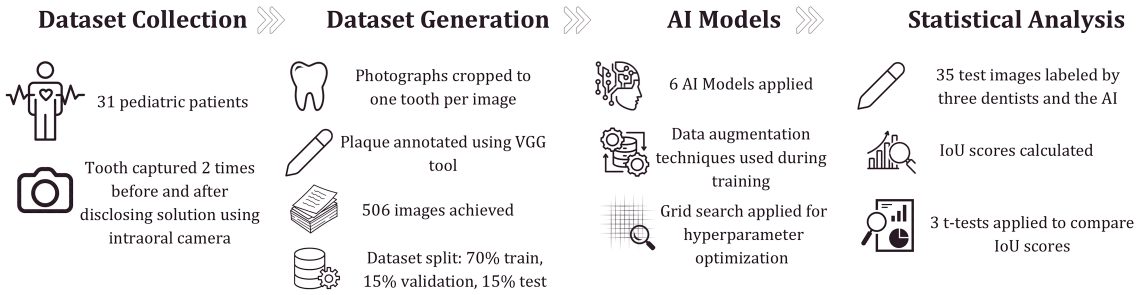


Figure 3.3 The workflow diagram of the study.

Table 3.2 Performance of deep learning models on the test dataset.

	Model Name	#Params	Recall	Precision	DSC	IoU
Deep Learning Models	DeepLabV3+	11.8M	0.7606	0.6664	0.7081	0.6575
	Mask R-CNN	44M	0.8027	0.7471	0.7395	0.7229
	Super Vison U-Net	1.9M	<u>0.8277</u>	0.8203	<u>0.8240</u>	0.7793
	U-Net	1.9M	0.8095	0.8006	0.8037	0.7607
	U-Net Transformer	100M	0.7718	<u>0.8782</u>	0.8215	<u>0.7845</u>
	YOLOv8	27M	0.5409	0.6600	0.5799	0.6157
Classic Methods	HSV-T	-	0.4975	0.5315	0.4582	0.3255
	SLIC-GC	-	0.7304	0.4926	0.5418	0.4040

Table 3.3 Optimal hyperparameters of the deep learning models.

Model Name	Image Size	Batch Size	Optimizer	Learning Rate
DeepLabV3+	192	2	Adam	1×10^{-2}
Mask R-CNN	256	8	SGD	1×10^{-2}
Super Vision U-Net	128	2	RMSProp	1×10^{-4}
U-Net	192	4	Adam	1×10^{-4}
U-Net Transformer	256	4	RMSProp	1×10^{-4}
YOLOv8	128	4	Adam	1×10^{-4}

Table 3.4 Optimal hyperparameters of the classical methods.

Model Name	Optimum Values
HSV-T	$H_{\text{low}} = 15$; $H_{\text{high}} = 69$; $S_{\text{low}} = 0.1$; $k_{\text{blur}} = 0$
SLIC-GC	$N_{\text{sp}} = 200$; $c = 20$; $\sigma_g = 1$; $k_{\text{blur}} = 3$; $k_{\text{morph}} = 3$; rule = bmax; $q_{\text{object}} = 0.75$; $q_{\text{BG}} = 0.25$; $n_{\text{GC}} = 3$

3.4 Results

DeepLabV3+, Mask R-CNN, YOLOv8, U-Net, Super Vision U-Net, and U-Net Transformer were trained on 354 images and tested on 79 images. The scores of the six DL models and two classic methods for dental plaque segmentation on the test dataset are shown in Table 3.2. Among the six DL models, U-Net Transformer yielded the best results, with an IoU of 0.7845 and a DSC of 0.8215. Compared to the DL models, classical approaches showed considerably lower performance. Among them, SLIC-GC achieved the best segmentation quality, with an IoU of 0.4040 and a DSC of 0.5418, whereas HSV-T performed significantly worse. The optimum hyperparameters of the models are given in Table 3.3 and Table 3.4.

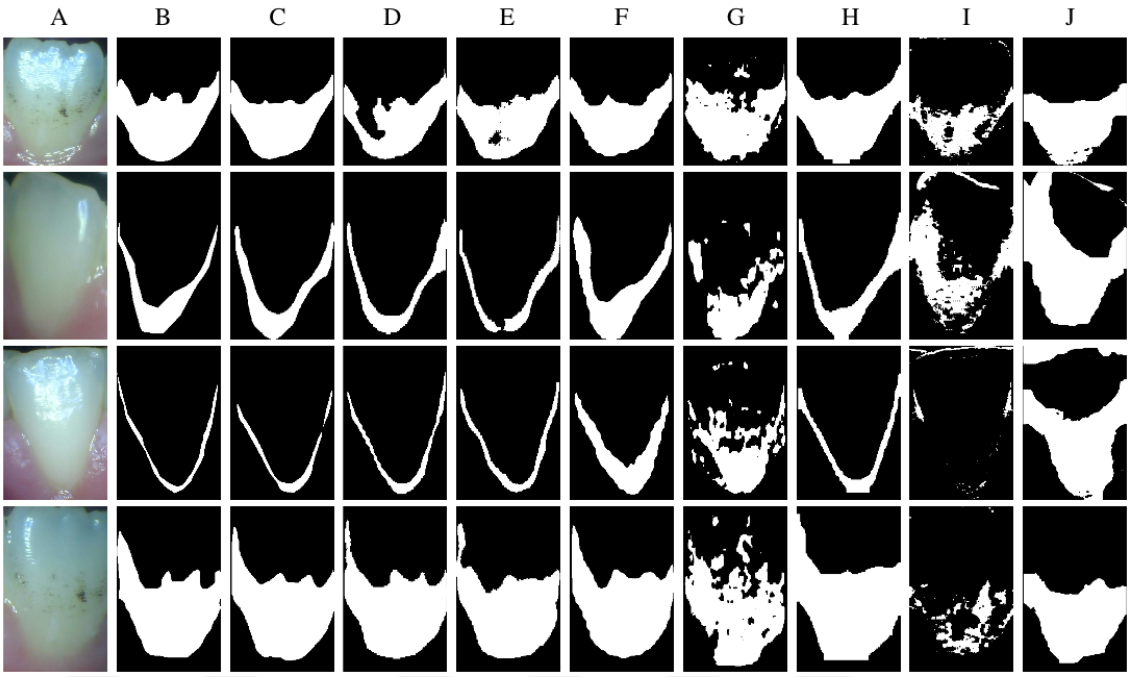


Figure 3.4 Visual comparison of dental plaque segmentation results.

Table 3.5 Prediction scores of the dentists and the model on 35 test images.

	Recall	Precision	DSC	IoU
Dentist A	0.5324	0.8661	0.6122	0.6565 ± 0.204
Dentist B	0.4405	0.8652	0.5304	0.6065 ± 0.196
Dentist C	0.6352	0.8494	0.6785	0.6898 ± 0.170
The model	0.7796	0.8398	0.7942	0.7783 ± 0.115

Table 3.6 *t*-test results comparing the model with the three dentists and one classical method, SLIC-GC.

	t	df	*p*
The model & Dentist A	−3.077	53.742	0.003
The model & Dentist B	−4.467	55,009	0.000
The model & Dentist C	−2.549	59.799	0.013
The model & SLIC-GC	-7.561	138.799	0.000

A qualitative comparison of segmentation results is shown in Figure 3.4. The columns are defined as follows: (A) Input image; (B) ground truth segmentation mask; and (C) through (J) the predicted segmentation results from the comparative models, which include (C) U-Net Transformer, (D) Super Vision U-Net, (E) U-Net, (F) Mask R-CNN, (G) DeepLabV3+, (H) YOLOv8, (I) HSV-T, and (J) SLIC-GC.

The prediction scores of the three dentists and the model (U-Net Transformer) on 35 test images are summarized in Table 3.5, which demonstrates that the model achieves superior IoU scores compared to the three dentists. Although the AI model performed best in recall, DSC, and IoU scores, it lagged behind the dentists in the precision score. The results of the *t*-tests are presented in Table 3.6. The model is effective, as evidenced by the IoU scores in Table 3.5 and by *t*-tests yielding *p*-values less than 0.05.

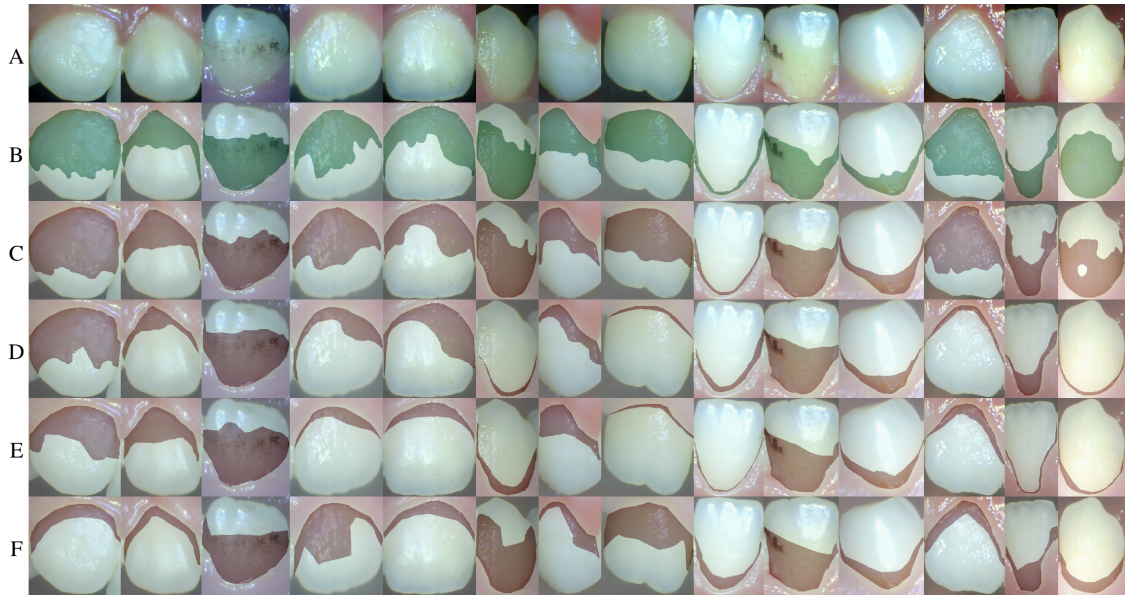


Figure 3.5 Segmentation of dental plaque by the model and three dentists.

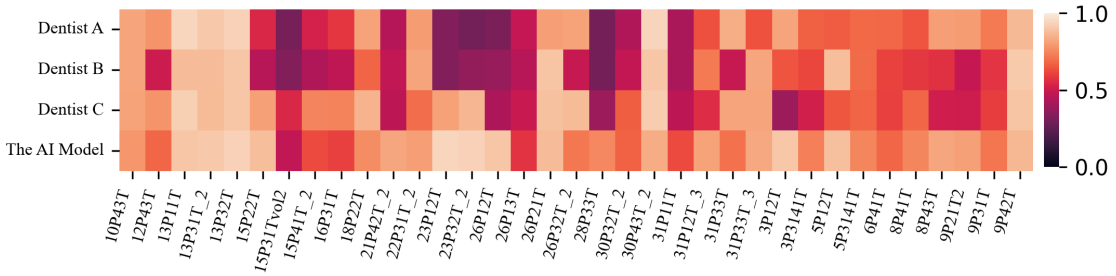


Figure 3.6 Heatmap visualization of IoU scores for the model and three dentists

The sample images predicted by the model and the three dentists, along with their corresponding ground truths, are depicted in Figure 3.5. The rows are defined as follows: (A) input image, (B) ground truth, (C) predictions of the AI model (U-Net Transformer), (D) predictions of dentist A, (E) predictions of dentist B, and (F) predictions of dentist C. According to this figure, the dental plaque predictions of the AI model are significantly closer to the ground truth compared to the predictions made by the three dentists. Figure 3.6 is the heatmap visualization of IoU scores for the AI model and three dentists on the test dataset. The color intensity represents the IoU values, where darker shades indicate lower scores and weaker segmentation agreement, while lighter shades correspond to higher IoU values and better agreement with the ground truth.

Figure 3.7 demonstrates the model's performance and interpretability on samples selected from the test dataset. The five rows are structured as follows: (A) Input images, (B) ground truth masks, (C) model predictions, (D) Grad-CAM outputs, and (E) Saliency Maps. It can be seen that the model predictions are highly consistent with the ground truth masks and successfully segment the dental plaque. Grad-CAM maps show that the model

focuses on areas with particularly dense plaque accumulation and strongly aligns with the ground truth mask. The saliency maps reveal that the model concentrates on plaque regions and, in particular, the boundaries of these regions during the decision-making process.

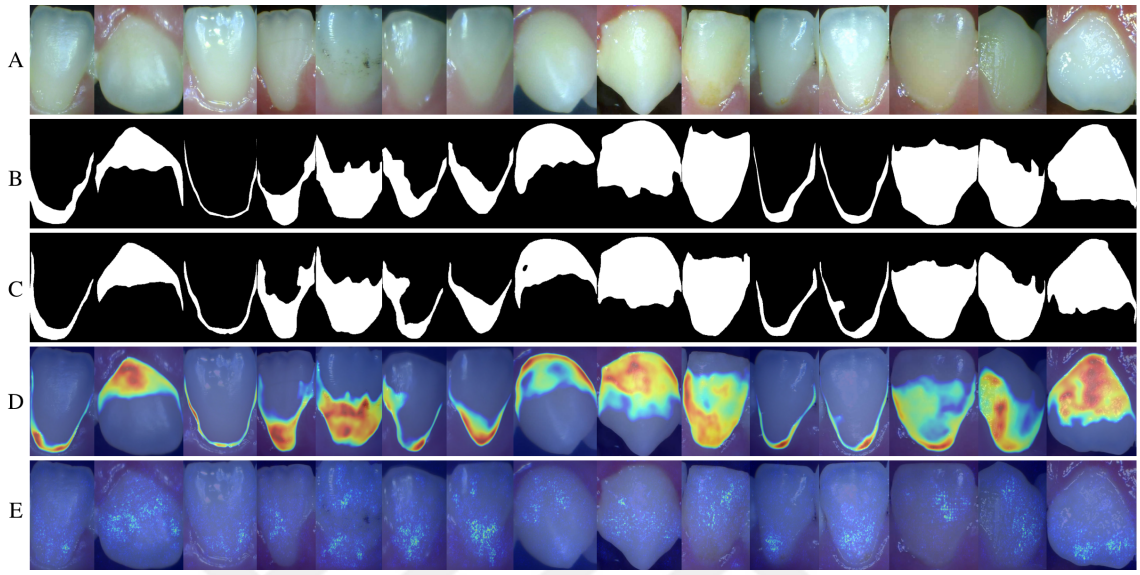


Figure 3.7 Examples showing explainability analysis of the model.

3.5 Discussion

Identifying dental plaque is essential to ensure that preventive and intervention treatments are safely provided to patients. Residual plaque can become more structured, making it harder to remove and more likely to harbor bacteria, which may contribute to oral diseases. If plaque is not properly eliminated, it can lead to gum inflammation (gingivitis), which may eventually progress to more severe conditions characterized by bone loss [98].

Identifying plaque in the young permanent teeth of children can be tricky for parents due to the similarity in color between the tooth surface and plaque. While staining the plaque with a disclosing agent helps visualize it, these agents have the potential to cause trauma in pediatric patients through changing the color of certain materials utilized in the restoration of young permanent teeth, resulting in the temporary staining of both the teeth, lips, tongue, hands, and clothes. The staining caused by disclosing agents may make individuals, especially children, feel self-conscious about their appearance, leading to discomfort in social settings. At the same time, these agents may provoke an unpleasant taste sensation in the oral cavity and elicit allergic reactions.

To overcome these limitations, AI-based dental plaque segmentation techniques offer a non-invasive, efficient, and automated alternative for plaque detection. Compared to conventional disclosing agents, AI-powered models, such as the U-Net Transformer used in this study, provide a precise, real-time, and objective assessment of plaque presence without the need for staining agents. This approach eliminates patient discomfort, making plaque detection more accessible, especially in pediatric dentistry.

Furthermore, traditional methods often rely on visual inspection by dentists, which is subjective and may vary based on expertise and lighting conditions. In contrast, AI-based models ensure consistent and standardized plaque detection, reducing the risk of human error. These advantages can enhance clinical workflow by reducing the time required for plaque assessment and supporting dentists in making more accurate treatment decisions. Additionally, integrating AI into routine dental examinations could help clinicians track changes in plaque accumulation over time, aiding in personalized preventive care and better patient education.

An analysis of the results indicates that U-Net and its variants, namely Super Vision U-Net and U-Net Transformer, outperform other models in dental plaque segmentation. U-Net Transformer achieves the best performance with an IoU of 0.7845 and a DSC of 0.8215, demonstrating the effectiveness of Transformer-based approaches in segmentation tasks. While Super Vision U-Net attains the highest recall, it falls behind U-Net Transformer in terms of precision, reflecting the trade-off between detecting more plaque regions and accurately identifying them. In contrast, YOLOv8 and DeepLabV3+ show relatively weak performance, likely due to their architectures being less suited for capturing fine structural details. Classical methods such as HSV, and SLIC-GC also perform poorly, often producing noisy or incomplete segmentations. This limitation may stem from the fact that dental plaque does not always exhibit a clear color contrast or distinct edge against the tooth surface, making intensity- or boundary-based methods unreliable. These findings highlight the limitations of traditional image processing in handling the complex texture and intensity variations of dental plaque, underlining the superiority of DL–based approaches for this task.

The visual comparison in Figure 3.4 further illustrates these differences. DL models (C–H) generate segmentations closer to the ground truth (B) than classical methods (I, J). Among them, U-Net Transformer (C) and Super Vision U-Net (D) provide the most accurate delineation of plaque regions, with well-defined boundaries and fewer false

predictions. On the other hand, YOLOv8 (H), DeepLabV3+ (G), and classical methods (I–J) struggle to capture plaque areas precisely, consistent with their lower quantitative scores. Finally, U-Net Transformer not only surpasses other models but also achieves higher IoU, recall, and DSC values than three dentists, as shown in Table 3.5. Figure 3.5 supports this observation by directly comparing U-Net Transformer predictions with expert annotations, highlighting the model's consistency relative to human performance.

As shown in Figure 3.7, the obtained Grad-CAM and Saliency images demonstrate that the model produces reliable outputs in plaque segmentation not only in terms of prediction accuracy but also in terms of the decision-making process. Specifically, Grad-CAM maps show that the model focuses on areas where plaque formation is concentrated, while saliency maps highlight plaque boundaries and plaque accumulation regions. This allows for a partial explanation of the model's *black-box* nature and increases its clinical interpretability.

Our findings are consistent with previous studies in the literature, highlighting both the advantages and limitations of existing methods. While Transformer- and U-Net-based models demonstrate superior performance compared to DeepLabV3+ models, it remains evident that further improvements are needed in dental plaque segmentation. Moreover, the results of this study align with recent research utilizing the O'Leary Plaque Score Index. For instance, Ramírez-Pedraza et al. categorized plaque accumulation into clinically interpretable levels, whereas this study enhances the accuracy and consistency of deep-learning-based plaque detection by ensuring alignment with established clinical standards [116].

The limitations of this study can be outlined as follows. First, although data augmentation techniques were applied to the training dataset, the overall number of images remained relatively small. Second, since the AI model was trained using images captured by a single type of camera, its performance with images from different camera brands and models remains uncertain. To address this limitation, domain adaptation techniques could be implemented alongside the inclusion of images taken with various camera brands. Third, due to the absence of a publicly available dental plaque dataset, a direct performance comparison with other models in the literature could not be conducted. Fourth, this study focused solely on predicting the presence of dental plaque; however, it did not assess plaque density, as the dataset lacked density information. Training the model with a dataset containing plaque density labels could enable the development of an

AI system capable of predicting both the presence and density of plaque, potentially bringing AI-based methods closer to the gold standard in dental plaque assessment.

Although an intraoral camera with an integrated flash was used to capture images, ambient illumination was not further standardized with a studio flash before imaging. Variations in lighting conditions may have influenced image quality, potentially affecting the accuracy of plaque detection and model performance. While the intraoral camera provided a controlled imaging environment, differences in equipment and photographic techniques could still lead to variations in image color and resolution, impacting the reproducibility of results. Standardization of imaging was more feasible in the anterior region, as positioning teeth correctly before and after staining was easier in this area. However, achieving similar standardization in posterior teeth remains challenging. Future studies could explore the use of a stabilization device to ensure fixed-angle imaging for both anterior and posterior teeth. Further research is needed to refine this approach and validate its effectiveness.

3.6 Conclusion

In this study, six DL models were trained for dental plaque segmentation on intraoral images of young permanent teeth, and their performance was compared both with each other and with classical image processing methods. The findings revealed that DL–based models demonstrated significantly superior performance, with U-Net Transformer emerging as the most effective solution. Furthermore, the model predictions showed clinically significant higher accuracy when compared to the predictions of three experienced pediatric dentists. This finding demonstrates that AI-based systems can be reliable tools that can support clinicians' decision-making processes and improve the quality of patient care.

The model's reliability was further established by examining its explainability using Grad-CAM and Saliency Maps. These visual tools confirmed that the model's decision-making mechanism aligns precisely with clinically relevant features such as high plaque density and distinct plaque borders. Thus, the model's *black box* nature has been partially made transparent, enhancing its potential for clinical use.

The study presents a non-invasive and objective alternative that can eliminate patient comfort issues and subjective evaluation differences that may arise from

traditional methods used in dental plaque detection (e.g., disclosing agents). Particularly in pediatric dentistry, AI-based plaque segmentation systems can contribute to the development of a safer, faster, and more patient-friendly approach.

Another significant contribution of this study is that it raises the possibility of individuals monitoring their own oral hygiene through AI models that will be trained on images captured with mobile devices in the future. Thus, patients and parents will be able to detect plaque using only a smartphone, without the need for any additional equipment.

In conclusion, this research demonstrates that DL models built on Transformer- and U-Net-based architectures have strong potential in dental plaque segmentation. The findings pave the way for the development of next-generation AI applications that can support clinical workflows and contribute to patient education and personalized preventive care processes.

Chapter 4

Low-Grade Glioma Segmentation in Magnetic Resonance Imaging

4.1 Introduction

Brain tumors are abnormal growths that can seriously affect the central nervous system and significantly reduce a patient's quality of life [118]. These tumors are typically categorized into two main types: low-grade gliomas (LGGs) and high-grade gliomas (HGGs). Gliomas, which arise from glial cells, constitute the most frequently occurring brain tumors. Based on their growth rate and level of malignancy, gliomas are divided into different grades. According to the World Health Organization (WHO) classification, they are categorized into four grades, from Grade I to Grade IV [119].

Among them, glioblastomas (Grade IV), which are a type of HGG, are the most aggressive and have the lowest survival rates [120]. Because of this, most research and funding have focused on HGGs, while LGGs have received less attention [121]. However, LGGs can slowly grow and spread into surrounding brain tissue. If not treated early, they may develop into HGGs, which makes them more dangerous over time [118]. This highlights the need for more research on the diagnosis and treatment of LGGs.

Features such as the tumor's size, shape, grade, and location are very important for deciding on the right treatment [118]. One of the first steps in diagnosing a brain tumor is medical imaging, especially Magnetic Resonance Imaging (MRI) [121]. MRI is a non-invasive method that provides detailed images of the brain and is widely used to detect brain tumors [25].

A critical part of analyzing MRI scans is the segmentation process. Segmentation means identifying and outlining the tumor area in the image. This helps doctors

understand where the tumor is and how big it is, which is useful for diagnosis and treatment planning [26]. Although manual segmentation by experts is still considered the most reliable method, it takes a lot of time, requires expertise, and may vary between observers. To address these challenges, researchers have increasingly focused on developing automated segmentation systems.

In recent years, deep learning (DL) methods—especially models based on U-Net and transformer architectures—have demonstrated remarkable performance in biomedical image analysis and segmentation. These technologies allow for fast and accurate tumor detection. In this study, the performance of U-Net and its variants in segmenting LGGs from brain MRI images is evaluated.

4.2 Related Works

This section presents previous studies focused on the segmentation of LGGs. Naser and Deen [24] developed a DL–based approach for segmenting brain tumors and grading LGGs using magnetic resonance imaging (MRI). In their study, a U-Net-based convolutional neural network (CNN) was used for tumor segmentation, while a VGG16-based transfer learning model was employed for tumor grading. The segmentation model was trained on a dataset consisting of T1-precontrast, FLAIR, and T1-postcontrast MRI scans from 110 patients, achieving a Dice Similarity Coefficient (DSC) of 84%.

Lefkovits and Szilágyi [122] developed a CNN-based system for the segmentation of both high-grade (HGG) and low-grade gliomas (LGG) using the Amazon SageMaker platform. Six different CNN models were trained on BraTS 2017–2020 datasets, and the best-performing model was selected through hyperparameter optimization. Furthermore, an ensemble model was developed based on the weighted average of these models to improve the accuracy of tumor boundary detection. The results were compared with the DSC scores from the BraTS 2020 challenge, and the proposed method ranked within the top 25%.

Shomirov, Zhang, and Billah [17] addressed the class imbalance problem in tumor segmentation by introducing a 3D U-Net model with a Weighted Focal Loss (WFL) function. Designed for both HGG and LGG segmentation, the model was tested on the BraTS 2019 and 2020 datasets. It demonstrated high accuracy in segmenting tumor core (TC), whole tumor (WT), and enhancing tumor (ET) regions.

BabaAhmadi and FallahPour [123] conducted a comparative study on three models for LGG segmentation: DeepLabV3+, U-Net, and a novel Transformer-based approach. Their findings showed that the MobileNetV3-based DeepLabV3+ model achieved the best overall performance. However, the Transformer-based model demonstrated superior segmentation accuracy, memory efficiency, and adaptability to different image sizes. Additionally, transfer learning techniques were applied to further improve the models' performance.

Similarly, Wan et al. [21] proposed a segmentation model based on DeepLabV3+ and RegNet, which incorporated not only MRI scans but also patients' genetic data and clinical information for a more comprehensive analysis. The model included a novel loss function called outlier-region loss, designed to reduce the influence of small, misclassified areas. Experiments on the Lower-Grade Glioma (LGG) Segmentation dataset yielded a DSC score of 94.36% and an Intersection over Union (IoU) score of 91.83%, outperforming several existing methods.

Finally, Dattangire, Biradar, and Joon [124] proposed a U-Net-based artificial intelligence (AI)-assisted model for LGG segmentation. Aiming to address the challenges of manual segmentation caused by the heterogeneous nature of tumors, their model was designed to support radiologists through automated segmentation. Trained on the BraTS dataset, the model was evaluated using accuracy, DSC score, and IoU metrics, and the results indicated that it could be effectively integrated into clinical decision support systems.

4.3 Materials and Methods

4.3.1 The Dataset

The dataset utilized in this research was sourced from The Cancer Imaging Archive (TCIA) and consists of clinical and imaging data from 110 patients diagnosed with low-grade gliomas (LGGs) [125]. The patients were collected from five different medical institutions. Among them, 50 were classified as Grade II, 58 as Grade III, while 2 patients had unknown tumor grades and 1 had an unspecified tumor subtype. The tumors were categorized into three subtypes: Astrocytoma, Oligoastrocytoma, and Oligodendroglioma. The distribution of these tumor types is shown in Table 4.1.

Table 4.1 Distribution of Tumor Grades and Subtypes in LGG Cohort

	Grade II	Grade III	Unknown
Tumor sub-types			
Astrocytoma	8	26	-
Oligoastrocytoma	14	14	1
Oligodendroglioma	28	18	-
Unknown	-	-	1
Total	50	58	2

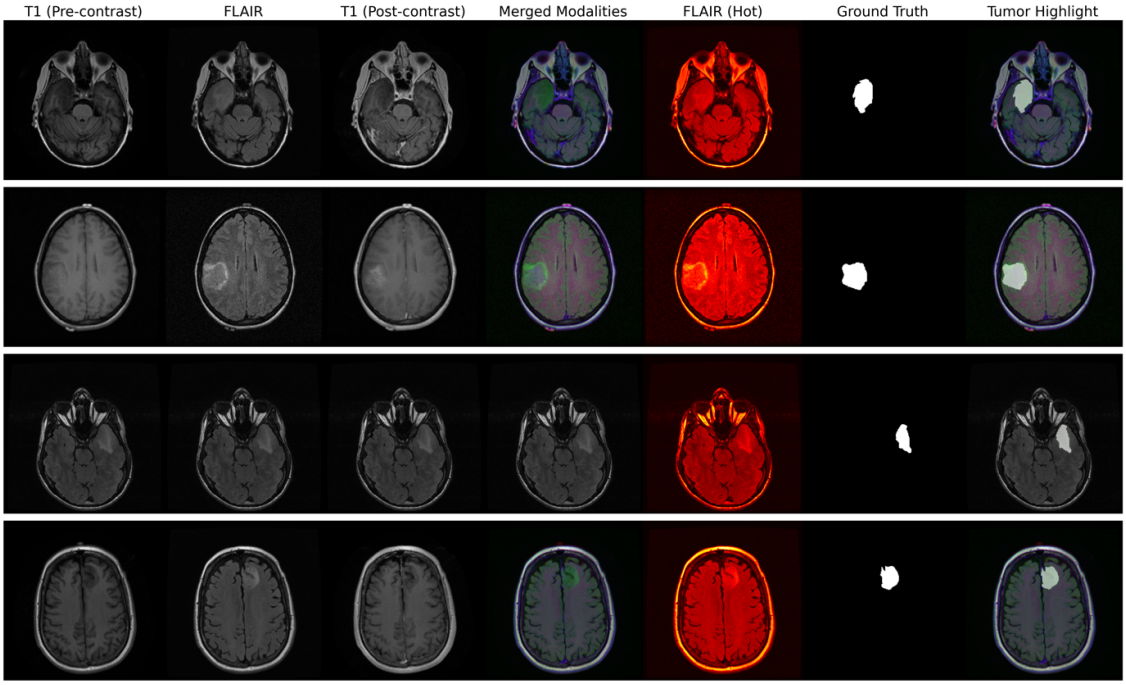


Figure 4.1 Visualization of multimodal brain MRI of five patients with LGG

Preoperative imaging data include three MRI sequences: non-contrast-enhanced T1-weighted (T1), FLAIR, and contrast-enhanced T1-weighted (T1CE) images. Each patient has between 20 and 88 slices, each with a resolution of 256×256 pixels. A total of 3,929 images were used, comprising 1,373 images containing tumors and 2,556 without visible tumor regions. Figure 4.1 presents example images from five patients in the dataset. These include the non-contrast T1, FLAIR, and contrast-enhanced T1 sequences. To aid interpretation, the Figure 4.1 also shows a merged visualization of the three modalities, a FLAIR image with a hot colormap for improved contrast, and the tumor region overlaid on the merged image. Each row represents a different patient. The columns show: (1) T1-weighted (pre-contrast), (2) FLAIR, (3) T1-weighted (post-contrast), (4) a merged visualization of the three modalities, (5) FLAIR with a hot colormap for enhanced contrast, (6) the ground truth tumor mask, and (7) the tumor region overlaid on the merged image.

Segmentation masks were manually annotated by Buda et al. [126] based on FLAIR images. These annotations are publicly available in TIF format along with the corresponding MRI scans. Each sample in the dataset consists of a three-channel image, where the channels represent the non-contrast T1, FLAIR, and contrast-enhanced T1 sequences. In cases where imaging data were incomplete, the missing channels were filled using available FLAIR scans to ensure data consistency.

To minimize bias and enhance the reliability of model evaluation, the dataset was partitioned at the patient level into three distinct subsets: training, validation, and testing. Specifically, 70% of the patients (77 patients) were assigned to the training set, 15% (16 patients) to the validation set, and 15% (17 patients) to the test set. As a result, the training set contains 2,738 images, the validation set includes 579 images, and the test set comprises 612 images. Dividing the dataset by patient ensures that images from the same individual are not split across different subsets, thereby allowing a more accurate assessment of the model's generalization performance.

4.3.2 Applied Classical Methods

For comparison with DL approaches, two classical baselines were implemented: Marker-Controlled Watershed (MC-Watershed) and Multi-Atlas Segmentation (registration + label fusion). Implementations used Python (version 3.10) with Scikit-Image (version 0.23), and SimpleITK (version 2.5). Because classical methods are sensitive to parameterization, hyperparameter optimization was conducted exclusively on the training set via systematic search; after selecting the optimal configuration, a single evaluation on the test set was performed. For each configuration, Recall, Precision, DSC, and IoU were recorded.

MC-Watershed: Preprocessing applied robust z-score normalization per channel and produced a coarse brain/ROI mask using Otsu thresholding followed by hole filling and small-object removal. Target/background markers were generated from multi-channel cues (high FLAIR relative to $\mu + k\sigma$ and low post-contrast T1 below the q-th percentile within the ROI), then refined by morphological erosion/dilation. After optional Gaussian smoothing, a gradient magnitude image (Scharr or Sobel) was computed and watershed was run on the gradient surface within the anatomical mask with optional compactness. Post-processing constrained predictions to the brain core (via a distance-transform margin) and applied opening/closing plus size filtering. The search explored:

k-sigma multiplier $k_\sigma \in \{1.0, 1.2, 1.5\}$; percentile $q \in \{0.5, 0.6, 0.7\}$; erosion radius $r_{\text{erode}} \in \{1,2,3\}$; dilation radius $r_{\text{dilate}} \in \{4,6,8\}$; gradient operator $o_{\text{grad}} \in \{\text{Scharr}, \text{Sobel}\}$; gradient Gaussian $\sigma_g \in \{0,1\}$; compactness $c \in \{0,10\}$; minimum component size $m_{\text{min}} \in \{150,300,500\}$ pixels; and core margin $m_{\text{core}} \in \{4,6,8\}$.

Multi-Atlas Segmentation: The pipeline selected the top-N_A atlases by similarity (measured on a single scalar channel), registered each atlas to the target with an initial similarity transform (translation–rotation–scale) and, when enabled, a local B-spline refinement. Registration used Mattes mutual information or correlation as the metric, random metric sampling, and a multi-resolution scheme optimized by Regular Step Gradient Descent. Atlas masks were warped to the target space with nearest-neighbor interpolation and combined via label fusion using STAPLE (with a probability threshold) or majority voting; light morphological clean-up (closing/opening and small component/hole removal) produced the final mask. The search varied: number of atlases $N_A \in \{6,8,10\}$; registration metric $M \in \{\text{MattesMI}, \text{CC}\}$; sampling percentage $\rho_s \in \{0.1,0.2\}$; iterations $T \in \{120,180\}$; B-spline grid spacing $g_{\text{bs}} \in \{8,10,12\}$; B-spline iterations $T_{\text{bs}} \in \{40,60\}$; fusion method $F \in \{\text{staple}, \text{majority}\}$ with threshold $\tau \in \{0.4,0.5\}$; morphological radius $r_{\text{morph}} \in \{0,1,2\}$; minimum component size $m_{\text{min}} \in \{20,30,50\}$; and the scaler channel used for registration $s_{reg} \in \{\text{flair}, \text{flair_mix}\}$.

4.3.3 Applied Deep Learning Models

In this study, six of the models employed in this study (Attention U-Net, UNet++, DynUNet, Super Vison U-Net, U-Net, and U-Net Transformer) are established architectures with proven effectiveness in the literature. The seventh model, Super Vision U-Net Transformer, is presented as a novel approach developed within the scope of this thesis. All models were implemented using Python (v3.10) and the TensorFlow framework (v2.11).

To effectively train the models, several data augmentation methods were carried out during preprocessing, including geometric transformations (horizontal flip, vertical flip, random rotation, transpose, affine transformations), pixel-level augmentations (random brightness and contrast adjustment, gamma correction, blurring), and deformation-based techniques (elastic transform, grid distortion, optical distortion).

These augmentations aimed to improve model performance, prevent overfitting, and enhance generalization ability.

For hyperparameter optimization, a grid search strategy was adopted. Specifically, batch sizes of 2, 4, 8, and 16, loss functions including Binary Cross Entropy (BCE), Dice Loss, BCE Dice Loss (a combination of BCE and Dice loss), IoU loss and Tversky loss, and learning rates of 1×10^{-3}, 1×10^{-4},and 1×10^{-5} were tested. The optimal combinations were selected based on the models' performance on the validation set. In all models, the number of training epochs was set to 200, and an early stopping mechanism with a patience value of 15 was employed. This configuration was designed to ensure a balanced and effective training process, maximizing both model accuracy and generalization capacity.

4.3.4 Statistical Analysis

U-Net, which is widely used as a reference model in the biomedical image segmentation literature and provides a strong baseline, was compared with UNet++, which achieved the highest performance in this study. Additionally, UNet++ was compared against MC-Watershed segmentation method. The DSC was computed per test sample, yielding paired observations on the same test set (n = 612). Normality was evaluated on the paired differences (U-Net − UNet++ and MC-Watershed – UNet++) using the Shapiro–Wilk test with visual inspection (histogram, Q–Q plot). As normality of differences was rejected (U-Net−UNet++: $W = 0.2899, p < 0.001$; MC-Watershed – UNet++: $W = 0.6883, p < 0.001$), with skewness and kurtosis values also being outside the ± 1 range, the primary comparison was conducted using the Wilcoxon signed-ranks test (two-sided). Effect size was summarized using the rank-biserial correlation.

Table 4.2 Performance of deep learning models on the test dataset.

	Model Name	#Params	Recall	Precision	DSC	IoU
Deep Learning Models	Attention U-Net	1.9M	0.8958	<u>0.9615</u>	0.8931	0.8654
	UNet++	2.41M	0.9411	0.9472	<u>0.9204</u>	<u>0.8946</u>
	DynUNet	21M	0.9326	0.9385	0.9054	0.8775
	Super Vison U-Net	1.9M	0.9400	0.9450	0.9172	0.8904
	U-Net	1.9M	0.9296	0.9362	0.9032	0.8749
	U-Net Transformer	116M	<u>0.9477</u>	0.9258	0.9070	0.8791
	SV U-Net Transformer	116M	0.8911	0.9393	0.8730	0.8411
Classic Methods	MC-Watershed	-	0.8116	0.7925	0.6653	0.6373
	Multi-Atlas	-	0.5581	0.4745	0.5129	0.3449

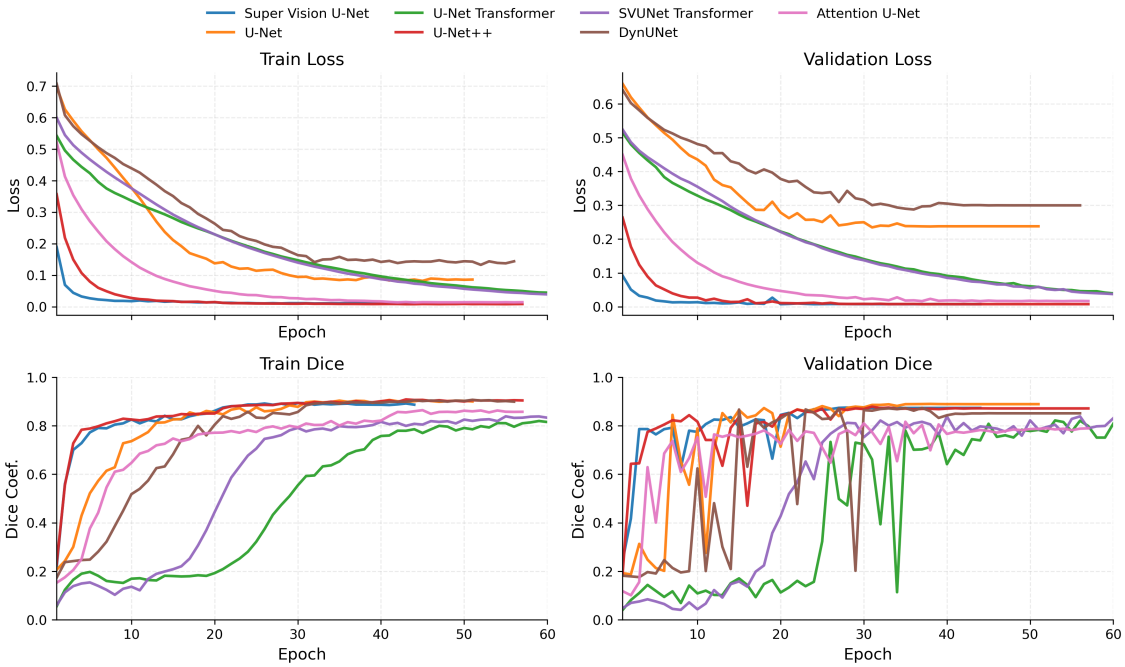


Figure 4.2 Visualization of the loss and performance metrics for the training process of the seven models

4.4 Results

The losses and performance metrics for the seven DL models during training are presented in Figure 4.2. Table 4.2 summarizes test-set performance across nine segmentation approaches. Among the models, UNet++ delivers the best overall accuracy with the top DSC (0.9204) and IoU (0.8946), closely followed by Super Vision U-Net (DSC 0.9172, IoU 0.8904) and U-Net Transformer (DSC 0.9070, IoU 0.8791). DynUNet (DSC 0.9054, IoU 0.8775) and the baseline U-Net (DSC 0.9032, IoU 0.8749) also perform strongly.

Regarding Recall, the U-Net Transformer (0.9477) demonstrated the highest sensitivity, closely followed by UNet++ (0.9411) and Super Vision U-Net (0.9400). Conversely, for Precision values, the Attention U-Net (0.9615) delivered the highest predictive accuracy. However, when considering the DSC and IoU values, the Attention U-Net exhibited lower overall performance compared to these models.

On the other hand, the Super Vision U-Net Transformer (SV U-Net Transformer) model lagged behind other models in terms of segmentation success, with lower DSC (0.8730) and IoU (0.8411) values.

Table 4.3 Optimal hyperparameters of the deep learning models.

Model Name	Batch Size	Loss Function	Learning Rate
Attention U-Net	16	BCE	1×10^{-4}
UNet++	8	BCE	1×10^{-4}
DynUNet	16	BCE Dice Loss	1×10^{-4}
Super Vision U-Net	4	BCE	1×10^{-4}
U-Net	16	BCE	1×10^{-5}
U-Net Transformer	4	Dice Loss	1×10^{-5}
SV U-Net Transformer	8	BCE	1×10^{-5}

Table 4.4 Optimal hyperparameters of classic methods

Model Name	Hyperparameters
MC-Watershed	$k_\sigma = 1;\ q = 0.7;\ r_{\text{erode}} = 1; r_{\text{dilate}} = 8; o_{\text{grad}} = \text{Sobel};$ $\sigma_g = 1; c = 10; m_{\text{min}} = 150$ px; $m_{\text{core}} = 4$ px
Multi-Atlas	$N_A = 6; M = \text{MattesMI}; \rho_s = 0.1; T = 120; g_{\text{bs}} = 8; T_{\text{bs}} = 40;$ $F = \text{staple with } \tau = 0.4; r_{\text{morph}} = 0; m_{\text{min}} = 30$ px; $s_{reg} = \text{flair_mix}$

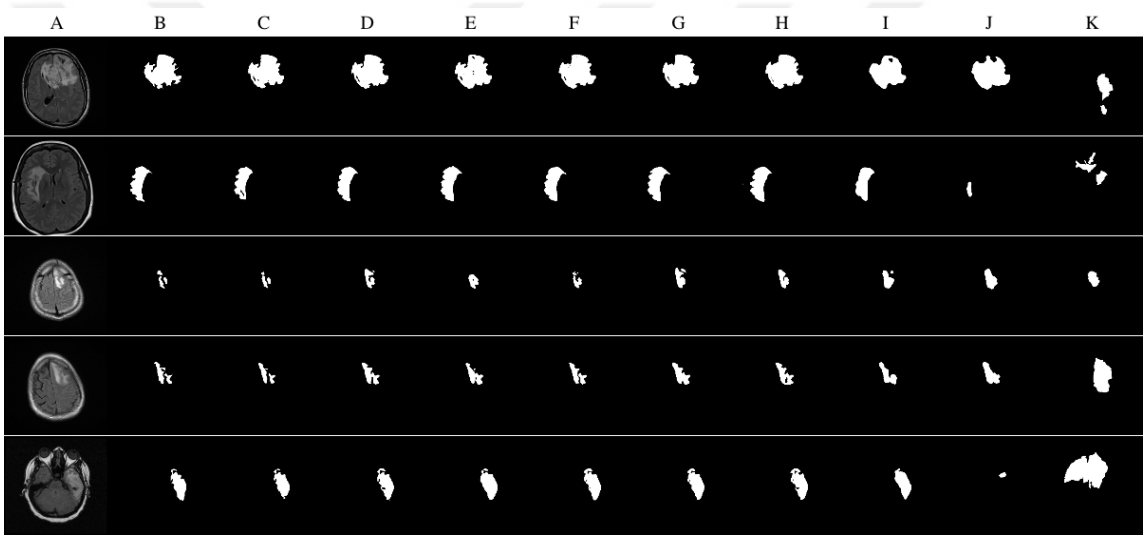


Figure 4.3 Example segmentation predictions obtained from the methods.

The two classical methods—MC-Watershed and Multi-Atlas—trail the deep models by a wide margin (best DSC 0.6653 vs. ≥0.8730 for deep models), underscoring the clear advantage of deep-learning-based approaches for LGG segmentation on this dataset. Overall, all DL models achieve high accuracy, with UNet++ offering the most balanced and state-of-the-art performance across metrics.

Table 4.3 and Table 4.4 present the optimum hyperparameters for the models. An analysis of the tables shows that the most successful model, UNet++, achieved its best performance using a batch size of 8, the Binary Cross-Entropy (BCE) loss function, and a learning rate of 1×10−4.

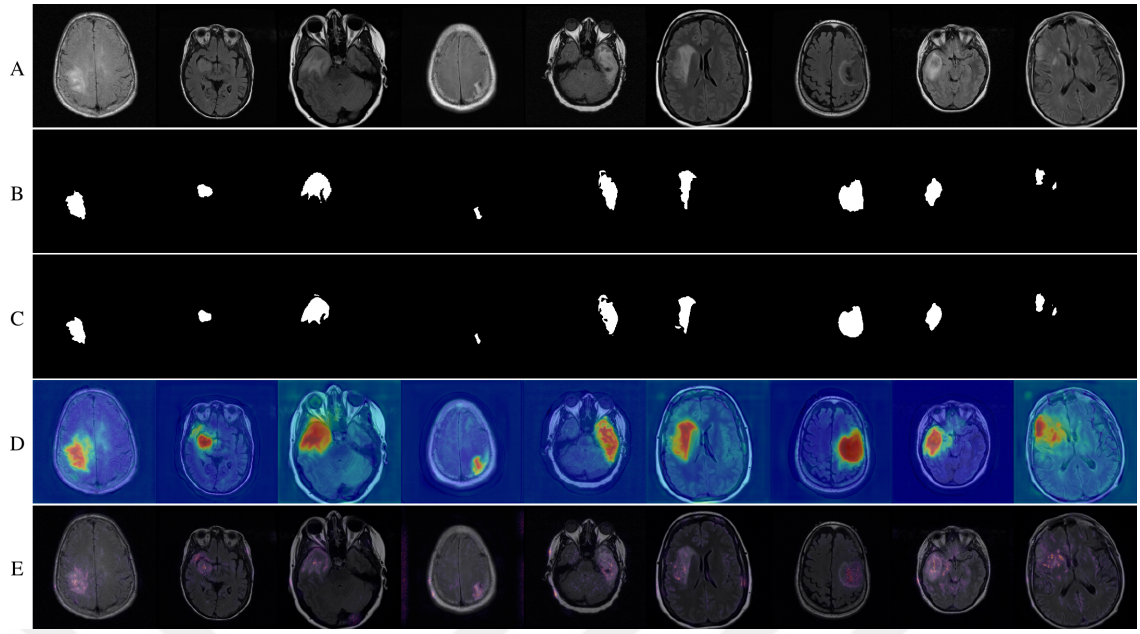


Figure 4.4 Samples showing explainability analysis of the model.

Sample prediction results obtained from the models used in the study are presented in Figure 4.3. The first two columns establish the baseline for evaluation: the input image (A) is followed immediately by the ground truth mask (B). The remaining nine columns present the predictions from the models, ordered as follows: Attention U-Net (C), U-Net++ (D), DynUNet (E), Super Vision U-Net (F), U-Net (G), U-Net Transformer (H), SV U-Net Transformer (I), MC-Watershed (J), and Multi-Atlas (K). These visual examples are provided to reveal the general trends of each model and the qualitative appearance of the segmentation results.

The Wilcoxon Signed-Rank Test indicated that UNet++ performed statistically and significantly better than both U-Net and the MC-Watershed segmentation method. The comparison between U-Net and UNet++ revealed a significant difference ($W = 7652.5$, $p < 0.001$), supported by a moderate effect size (rank-biserial $r_{rb} = -0.40$). The performance advantage of UNet++ over the MC-Watershed method was far more substantial ($W = 1038,\ p < 0.001$), with a very large effect size (rank-biserial $r_{rb} = -0.95$) suggesting an almost complete superiority in performance.

The UNet++ model achieved superior results on the test dataset compared to the other models. Consequently, to investigate the model's explainability, Figure 4.4 presents sample cases, including (A) the input images, (B) ground truth masks, (C) model predictions, and explainability methods: (D) Grad-CAM and (E) Saliency Map. An examination of these visualizations reveals that the model can capture tumor boundaries

with particularly high accuracy. The Grad-CAM and Saliency maps further highlight the regions the model focuses on during its decision-making process, drawing attention to areas consistent with the segmentation output.

4.5 Discussion

The findings indicate that different DL architectures offer various strengths for LGG segmentation. Specifically, the UNet++ model outperformed other models in terms of DSC and IoU values, indicating that the model can achieve high structural similarity in low-grade glioma segmentation. This is consistent with findings in the literature that UNet++'s ability to produce richer feature maps improves segmentation accuracy [79].

The U-Net Transformer, with its high recall, was able to capture a larger portion of the tumor regions. However, its relatively lower precision indicates the presence of some false positive segmentations. This suggests that although transformer-based architectures are effective in modeling global context, they may be prone to errors in capturing fine structural details. Although the Attention U-Net model appears powerful in reducing the false positive rate by achieving the highest precision value, the relatively low DSC and IoU values indicate that it may be limited in determining the exact boundaries of the tumor.

As expected, the SV U-Net Transformer model—which combines the strengths of the two highest-performing models from the first study, U-Net Transformer and Super Vision U-Net— was anticipated to achieve strong results in LGG segmentation. However, the results did not meet this expectation; the model yielded lower DSC and IoU scores compared to the other methods. This can be attributed to potential feature mismatches and conflicts in learning strategies that may arise when merging different architectural designs. In particular, the inability of the U-Net Transformer's global context capture capacity and the Super Vision U-Net's more locally focused feature extraction mechanisms to integrate effectively in the same model may have made the model's optimization difficult. Furthermore, the combination of a high parameter count and a relatively low number of samples in the dataset likely prevented the model from effectively optimizing its parameters, thus failing to achieve the desired learning level. Future work could help the model realize its expected performance by using larger

datasets, experimenting with different loss functions, or employing advanced regularization techniques.

Furthermore, an analysis of the architectural design reveals that model size is not a primary driver of accuracy: compact networks (~1.9–2.4M parameters) such as UNet++ and Super Vision U-Net match or surpass much larger models (DynUNet at 21M and Transformer variants at 116M), highlighting strong parameter efficiency for this task.

When examining hyperparameter optimization, most models yield the best results with a learning rate of 1×10^{-4} and BCE, while Transformer-based models outperform with 1×10^{-5} and Dice/Dice+BCE loss; this indicates that attention components require smaller steps and overlap-sensitive losses. Batch size generally ranges from 8 to 16 in U-Net variants and drops to 4–8 in more complex or transformer-based architectures. This pattern reflects both memory constraints associated with larger models and the regularizing effect of using smaller batches.

In contrast to DL models, the classical approaches—MC-Watershed and Multi-Atlas—performed substantially worse, with DSC scores of 0.6653 and 0.5129, respectively. This large performance gap highlights the limitations of traditional, rule-based segmentation techniques in capturing the heterogeneous appearance and irregular boundaries of LGG tumors. Unlike deep architectures that can learn hierarchical and context-aware features directly from data, classical methods rely on predefined intensity thresholds or atlas-based correspondences, which are often insufficient for complex and variable tumor structures. These results reinforce the transformative role of DL in biomedical image segmentation and emphasize that modern architectures not only provide higher accuracy but also greater adaptability to diverse clinical scenarios.

The findings reveal that the UNet++ model demonstrates strong performance not only in terms of accuracy but also in explainability. Visualizations generated by Grad-CAM and Saliency Map methods indicate that the model focuses intensely on the actual tumor regions and that false positive areas are limited. This robustness significantly enhances the model's reliability and offers a crucial advantage for its potential use in clinical applications. However, some examples showed partial deviations along the segmentation boundaries. This suggests that the model needs further improvement, especially in handling the heterogeneous regions of the tumor. Crucially, the explainability methods contribute to understanding these limitations, thereby guiding future work aimed at model refinement and enhancement.

4.6 Conclusion

This study confirms the clear superiority of DL over classical methods for LGG segmentation. The UNet++ model offered the most balanced and reliable performance in terms of overall accuracy, IoU, and explainability. While Transformer-based models achieved higher recall, providing wider tumor coverage, they consequently showed a trade-off in precision. Importantly, model size was not a primary determinant of success; compact architectures were found to be competitive with much larger models. Grad-CAM and Saliency visualizations support UNet++'s potential for clinical use, yet partial deviations at tumor boundaries suggest the need for further refinement. Future work should focus on utilizing larger, heterogeneous datasets compiled from different institutions, designing specialized loss functions, implementing advanced regularization, incorporating multi-modal learning, and developing strategies to harmoniously merge global and local feature representations.

Chapter 5

Prostate Gland Segmentation on ^{68}Ga-PSMA-11 PET/CT

5.1 Introduction

Prostate cancer (PCa) is the second most commonly diagnosed cancer in men. In 2022, it caused approximately 1.46 million new cases and 396,000 deaths worldwide. Due to an aging population, this number is expected to rise to 2.4 million new cases by 2040 [127]. In the United States alone, about 313,780 new cases are predicted for 2025, making up around 15.4% of all expected cancer diagnoses [128, 129].

Despite progress in diagnosis and treatment, local recurrence and metastatic progression remain as major clinical challenges. To reduce treatment-related side effects and improve outcomes, highly accurate imaging-based planning methods are essential. Positron emission tomography/computed tomography (PET/CT), which combines functional and anatomical imaging, is increasingly being used for staging, monitoring treatment response, and especially for target volume definition in radiotherapy for prostate cancer [130]. Both PSMA-based tracers (such as ^{68}Ga-PSMA-11) and non-PSMA tracers (like ^{18}F-fluciclovine) are capable of detecting lesions with high sensitivity, even in cases of multifocal involvement or suspected biochemical recurrence.

However, to make full use of these imaging benefits in clinical workflows, the contours drawn on these images must be accurate, consistent, and efficient. In current clinical practice, the boundaries of the prostate, nearby seminal vesicles, and surrounding critical organs (like the bladder and rectum) are mostly outlined manually. This process is time-consuming and often shows significant variability between observers. Inaccurate contours can negatively affect both the biological dose delivery and quantitative metrics

such as SUV values [131]. Moreover, modern approaches like adaptive radiotherapy (ART) and radioligand therapy often require repeated segmentations on a daily or weekly basis, making manual methods impractical. Recent studies have shown that automatic segmentation of organs at risk and tumors can significantly shorten ART planning time and reduce clinical workload [132].

However, differentiating the prostate from surrounding soft tissue on CT is challenging due to low contrast, and PET alone does not provide clear anatomical boundaries. Therefore, using multimodal PET and CT modalities together helps match functional activity with anatomical structures, improving segmentation accuracy and supporting advanced planning strategies such as biologically targeted dose escalation. An automatic segmentation model could provide clinicians with reliable volume definitions for: (i) real-time re-planning in adaptive radiotherapy, (ii) tumor burden and dosimetry analysis in radiopharmaceutical treatments, (iii) targeting in biopsy or focal therapy, and (iv) radiomics-based big data research.

This study aims to propose an automated prostate segmentation model using state-of-the-art DL architectures trained on both PET and CT.

5.2 Related Works

Litjens et al. [133] introduces a standardized benchmark to assess prostate MRI segmentation algorithms. Using a dataset of 100 T2-weighted MRI scans from four centers, the challenge evaluated 11 methods, including active appearance models and multi-atlas registration, through metrics like Dice Similarity Coefficient (DSC) and Hausdorff distance. The top-performing algorithms (Imorphics and ScrAutoProstate) achieved scores above 84, surpassing most competitors in both accuracy and efficiency. Results highlight that active appearance model-based approaches generally outperform atlas-based methods, though combining algorithms could further enhance performance. The study underscores the continuing need for robust, generalizable segmentation solutions across multi-center and multi-protocol data.

Bhandary et al. [134] systematically evaluates U-Net and its variants for automatic prostate segmentation using multi-modal biomedical images. The authors highlight the limitations of manual delineation, such as time consumption and observer variability, and emphasize the role of DL, particularly U-Net-based convolutional neural networks, in

achieving state-of-the-art segmentation performance. Five architectures—U-Net, V-Net, Attention U-Net, SegResNet, and UNet++—were benchmarked on four datasets, including PROMISE-12 and MSD-prostate, using DSC and Hausdorff distance metrics. Results show that nnU-Net and Attention U-Net generally outperform the baseline U-Net, especially on smaller datasets, while hybrid or attention-based approaches offer improved robustness. The study provides an open-source framework for standardized evaluation and underscores the importance of dataset diversity and rigorous validation for clinical applicability.

Tian et al. [135] presents an end-to-end DL framework for automated prostate segmentation on MRI. The proposed model, PSNet, is based on a fully convolutional network (FCN) that is fine-tuned from a pretrained network to enhance segmentation accuracy. Unlike traditional patch-wise CNN approaches, PSNet enables pixel-wise prediction over the entire image, reducing computational redundancy. The model was evaluated on 140 MRI volumes from three datasets, including ISBI2013 and PROMISE12, using metrics such as DSC, relative volume difference (RVD), and Hausdorff distance (HD). PSNet achieved a mean DSC of 85.0%, outperforming state-of-the-art models like U-Net and V-Net. Its weighted cross-entropy loss further improved segmentation balance between prostate and background voxels. The study highlights PSNet's robustness, efficiency, and potential applicability to broader biomedical image segmentation tasks.

Zhang et al. [136] introduces Z-Net, a novel convolutional neural network (CNN) designed for automatic prostate segmentation from MRI scans. Z-Net extends the U-Net architecture by incorporating Z-blocks and decoder Z-blocks with dense connections and feature concatenation, allowing for more effective multi-level feature extraction and improved boundary detection. The network was trained on 45 MRI images from the PROMISE12 dataset and tested on 30 images. Three image preprocessing strategies—pad & cut, 2D resize, and 3D resize—were evaluated, with 2D resizing yielding the best performance. Quantitative evaluation showed that Z-Net outperformed U-Net and Ensemble DCNN, achieving a higher DSC (90.49%), lower Hausdorff distance. The results indicate that Z-Net provides more accurate and robust segmentation for prostate MRI, making it promising for clinical applications.

Zhu et al. [137] proposes a deeply-supervised CNN to accurately segment the prostate from MRI scans, addressing challenges such as fuzzy boundaries and patient-

specific variations. The architecture builds on U-Net but integrates 1×1 convolutions for deeper feature extraction and eight additional supervision layers to improve gradient flow and reduce information loss during training. The model is trained end-to-end with the DSC as the loss function to handle class imbalance. Evaluated on a dataset of 81 patients, it outperformed U-Net and Fully Convolutional Networks (FCNs), achieving a mean DSC score of 0.885 compared to 0.865 for U-Net. The results demonstrate improved boundary detection and smoother segmentation contours.

Lei et al. [138] introduces a deeply supervised V-Net (DS-CR-V-Net) designed for automatic prostate segmentation in transrectal ultrasound (TRUS) images, addressing the low contrast and noise challenges of TRUS. The model integrates deep supervision at multiple stages with a hybrid loss function (binary cross-entropy + Dice loss) to enhance feature learning and convergence. A multidirectional contour refinement step combines segmentations from transverse, sagittal, and coronal planes to improve accuracy, especially at the prostate apex and base. The network was trained and evaluated on TRUS images from 44 patients, achieving a DSC of 0.919, outperforming U-Net, V-Net, and CNN-based methods. This framework demonstrates robust segmentation suitable for clinical use in prostate cancer interventions.

Jiang et al. [139] introduces MicroSegNet, a transformer-based U-Net variant designed for prostate segmentation in micro-ultrasound (micro-US) images, which have higher resolution but suffer from artifacts and indistinct boundaries. Built on the TransUNet architecture, MicroSegNet employs multi-scale deep supervision and a novel annotation-guided binary cross-entropy (AG-BCE) loss, which assigns higher weights to challenging regions where expert and non-expert annotations disagree. The model was trained on data from 55 patients and evaluated on 20 patients, achieving a DSC of 0.939 and a Hausdorff distance of 2.02 mm, outperforming U-Net, Dilated Residual U-Net (DRUNet), TransUNet, and even human annotators. The results highlight MicroSegNet's ability to produce accurate and robust segmentations, making it promising for real-time prostate biopsy guidance and cancer staging.

Lei et al. [140] presents a synthetic MRI-aided deep attention fully convolution network (sMRI-DAFCN) for automatic prostate segmentation on CT, addressing the challenge of poor soft-tissue contrast in CT. The approach first generates synthetic MRI (sMRI) from CT using a CycleGAN, then applies a deep attention FCN to enhance boundary detection by focusing on relevant features. Evaluated on data from 49 patients

(leave-one-out) and an additional 50 patients (hold-out test), the method achieved a DSC score of 0.92 ± 0.09 and a Hausdorff distance of 4.38 mm, outperforming standard U-Net approaches. By eliminating the need for MRI acquisition and registration, this framework offers a faster and more accurate solution for radiotherapy planning.

Li et al. [141] proposes FocalUNETR, a transformer-based architecture for accurate prostate segmentation on CT. The model integrates focal self-attention (SA), which captures both fine-grained local features and global context, addressing the limitations of conventional CNN-based methods. To tackle the challenge of unclear prostate boundaries in CT scans, an auxiliary boundary-aware contour regression task is incorporated using Gaussian-based heatmaps as additional supervision. The network employs a hybrid loss combining Dice and cross-entropy for segmentation and mean squared error for boundary regression. Tested on two large CT datasets (400 private scans and 300 AMOS scans), FocalUNETR achieved a DSC of 89.23% on the private dataset, outperforming U-Net, Swin-UNet, and UNETR. The results highlight its ability to deliver precise boundary delineation while maintaining global shape consistency.

He et al. [142] introduces HF-UNet, a multi-task learning framework designed for prostate segmentation in CT, where soft tissue contrast is typically low. The method employs a two-stage pipeline: (1) a lightweight U-Net to localize the prostate region, followed by (2) a hierarchically-fused U-Net that performs both prostate segmentation and contour delineation as complementary tasks. A novel Task Consistency Learning (TCL) block facilitates feature sharing between the segmentation and boundary branches while preserving task-specific representations. The boundary delineation task is formulated as a regression problem with Gaussian-based contour labels to improve accuracy on unclear boundaries. Evaluated on a 339-patient CT dataset, HF-UNet significantly outperformed U-Net, V-Net, and other multi-task networks, achieving a DSC score of 0.88 and a marked reduction in surface distance errors.

Zhong et al. [143] introduces ProSegDiff, a novel diffusion-model-based framework for prostate segmentation, addressing the challenges of ambiguous boundaries and low tissue contrast. Unlike traditional CNNs and transformers, ProSegDiff employs a conditional network with a Transformer encoder, an Adapter module, and a denoising network. The Adapter dynamically adjusts injected features during each denoising step, progressively refining global and boundary details. Additionally, a selection strategy filters high-quality outputs from multiple inferences to enhance final segmentation

accuracy. The model was evaluated on four benchmark datasets (MRI and ultrasound) using metrics such as DSC, Intersection over Union (IoU), Hausdorff Distance, and ASD, outperforming both discriminative and generative baselines, including MicroSegNet and CriDiff. The approach demonstrates strong boundary preservation and robustness in challenging cases.

Ma et al. [144] proposes ResGNet, a new encoder–decoder architecture combining residual and gated connections in a multi-path and multi-directional learning strategy to improve prostate segmentation on MRI. The ResGNet block splits feature maps into four branches, enabling simultaneous focus on global attention and local boundary features. A dynamic weighted fusion loss (Dice + Focal Tversky) is used to optimize both class balance and hard boundary regions. Evaluated on multiple public datasets (PROMISE12, NCI-ISBI13, PI-CAI, Prostate158), ResGNet achieved state-of-the-art performance, including a DSC score of 0.944 on PROMISE12. It also demonstrated strong robustness across varying MRI scanners and prostate shapes.

Kot et al. [145] presents a hybrid approach that combines YOLOv4 for object detection and U-Net for prostate segmentation, trained on both single-modality (MRI) and multimodal MRI+CT datasets. YOLOv4 is first used to localize the prostate, after which U-Net performs fine segmentation on cropped and preprocessed regions. Data from 120 patients (MRI and CT scans) were used, with experiments evaluating the effects of data augmentation and modality fusion. The best performance was achieved using a mixed MRI+CT dataset, reaching a DSC score of 0.9685 at a 0.6 threshold, outperforming single-modality models. This pipeline demonstrates the benefit of combining detection and segmentation steps, as well as leveraging multimodal data for improved generalization.

Song et al. [146] establishes a multi-institutional expert consensus dataset of prostate contours on MRI to serve as a benchmark for artificial intelligence (AI)-based auto-segmentation tools. A panel of two radiologists and two radiation oncologists developed slice-by-slice consensus contours for 68 MRI cases from six institutions, using high-resolution T2-weighted images. Six AI tools (three commercial and three academic) were evaluated against this reference standard using DSC score, boundary errors, and volume differences. The best-performing AI achieved a median DSC of 0.94 and a mean surface error of 1.3 mm, while the worst model reached 0.80 with errors up to 9.4 mm. Although top AI models exceeded the typical accuracy of CT-based contours by

clinicians, none were flawless—manual review remains necessary for clinically critical tasks like radiotherapy planning. The paper also provides a detailed contouring atlas to guide standardized prostate segmentation.

Overall, the existing literature demonstrates substantial progress in prostate segmentation using MRI, CT, ultrasound, and more recently transformer- or diffusion-based approaches. However, segmentation pipelines specifically designed to exploit the complementary strengths of PET and CT remain largely unexplored. While PET provides valuable functional information and CT offers high-resolution anatomical detail, an integrated framework that leverages both modalities has not been systematically investigated. To address this gap, the present work evaluates prostate segmentation on PET/CT through two complementary strategies: (i) CT-only segmentation as a baseline, and (ii) multimodal PET+CT segmentation using dual-encoder and fusion-based architectures. This comparative analysis highlights the potential of PET information for improved boundary delineation and provides insights into effective strategies for multimodal integration.

5.3 Materials and Methods

5.3.1 Dataset Collection

For this study, a dataset of ^{68}Ga-PSMA-11 PET/CT scans was retrospectively collected from the archives of the Nuclear Medicine Department at Kayseri City Hospital. It includes 593 scans from 589 patients with prostate cancer, with a mean age of 73.63 ± 10.20 years. Patients who underwent radical prostatectomy were excluded from the study. All data were approved for use by the hospital's ethics committee (Decision No. 375, dated March 11, 2025). Personally identifiable information was removed, and the data will be used solely for research purposes.

The imaging data were archived in a format suitable for segmentation of the prostate region. All scans were acquired according to clinical standards using a ^{68}Ga-PSMA-11 PET/CT protocol. Cases were excluded if the images were non-diagnostic, had artifacts or low resolution, or if the prostate borders were unclear due to surgery, congenital malformations, or widespread metastasis.

A two-stage normalization process was applied to the DICOM images. In the first stage, pixel values were converted to Hounsfield Units (HU) using the rescale slope and intercept parameters extracted from each DICOM file. Subsequently, intensity values were windowed by clipping them to a predefined range based on the window center and window width, ensuring that only clinically relevant contrast levels were preserved. In the second stage, a min-max normalization was performed, where the pixel values of each individual image were scaled to the range of 0.0 to 1.0 based on their own minimum and maximum values. This normalization step facilitates the learning process during model training and ensures consistency across different images by standardizing intensity distributions.

Manual segmentation of the prostate gland was conducted in a structured multi-step process to ensure high annotation quality. For each patient, an experienced physician first identified the axial slice in which the prostate was most clearly visible. Subsequently, a second physician manually delineated the prostate boundaries on the selected slice using the Label Studio annotation platform. Finally, a third expert physician independently reviewed the annotated masks for accuracy and consistency. This multi-expert validation workflow ensured reliable ground truth annotations for a total of 593 PET/CT scans. Representative examples from the annotated dataset are presented in Figure 5.1. Each column represents a different patient. The rows show: (A) CT, (B) PET, (C) the ground truth mask, and (D) the prostate region overlaid on the CT.

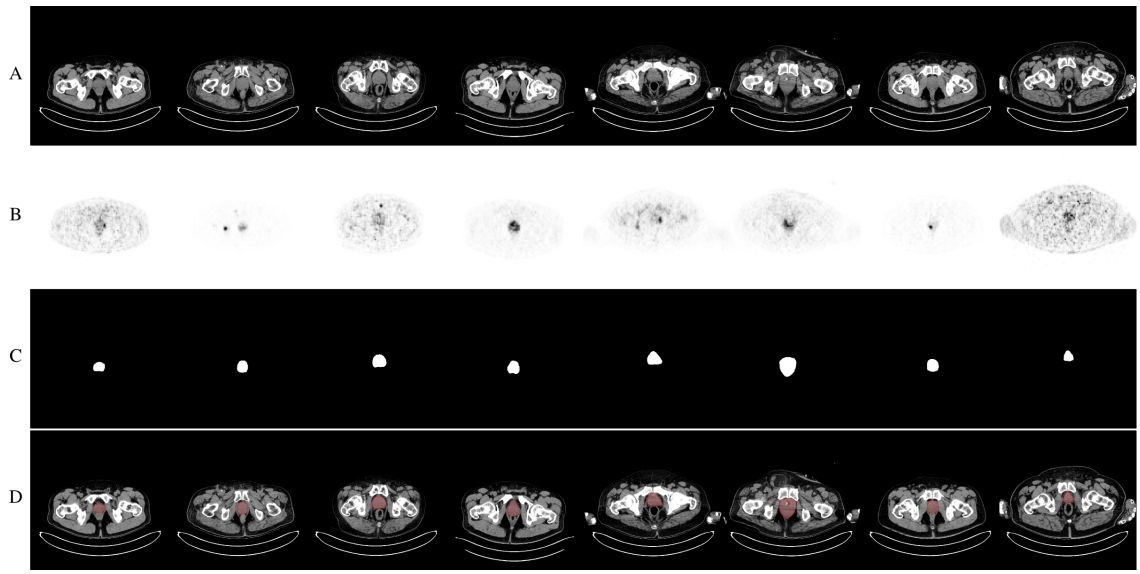


Figure 5.1 Samples from the dataset

To train and evaluate the model, the dataset was split randomly into three parts at the patient level: 15% (83 patients) was set aside as an independent test set; the remaining 85% (506 patients) was used for training and validation. Within this, 85% (427 patients) formed the training set, while 15% (83 patients) was used for hyperparameter tuning during validation. Only the training and validation sets were used during model development. The test set was reserved for final performance evaluation, completely independent of training.

5.3.2 Applied Classical Methods

Two classical methods were applied for comparison with DL approaches: Multi-Atlas based segmentation and Graph-Cut with Shape Prior. The implementations were developed in Python (version 3.10) using SimpleITK (version 2.5) and Scikit-Image (version 0.23). Due to the parameter sensitivity of classical methods, hyperparameter optimization was performed exclusively on the training set using a systematic grid search. After identifying the optimal configuration, a single evaluation was performed on the test set. For each configuration, the Recall, Precision, DSC, and IoU metrics were reported.

Multi-Atlas: Initially, each image slice underwent robust min-max normalization to the [0,1] range. For a given target slice, a subset of atlas exemplars was selected from the training set. Each atlas was coarsely aligned to the target using a 2D similarity transform (translation–rotation–scale) and, when enabled, finely aligned using a local B-spline free-form deformation. During registration, either Mattes Mutual Information or correlation was used as the similarity metric. Transformations were estimated using a multi-resolution scheme (involving downscaling and smoothing) coupled with a Regular Step Gradient Descent optimizer. The registered atlas masks were transferred to the target space using nearest-neighbor interpolation. These masks were then combined to generate a single consensus mask using either STAPLE (with a probability threshold) or majority voting. In the final stage, minor morphological operations, like opening and closing, were applied to remove small particles and produce a compact, coherent binary segmentation. The search varied: number of atlases $N_A \in \{3,5,7\}$; registration metric $M \in \{\text{MattesMI}, \text{CC}\}$; sampling percentage $\rho_s \in \{0.2,0.4\}$; iterations $T \in \{100,150\}$; B-spline grid spacing $g_{\text{bs}} \in \{8,10,12\}$; B-spline iterations $T_{\text{bs}} \in \{40,60\}$; fusion method $F \in \{\text{staple}, \text{majority}\}$ with threshold $\tau \in \{0.4,0.5\}$; morphological radius $r_{\text{morph}} \in \{0,2\}$.

Graph-Cut with Shape Prior: The image was first normalized to the [0–1] range using min-max normalization. From the training masks, a prior probability map and the corresponding Signed Distance Field (negative inside the object, positive outside) were derived. Using this prior, automatic seeds were generated for the target: a small disk in the core area for the object and strip-shaped regions at the image edges for the background. The total energy was optimized via the *s–t* min-cut / max-flow algorithm. The resulting binary mask was then refined using minor morphological operations, such as closing/opening and small component/hole removal, to produce the final segmentation. The search varied: unary shape-prior weight $w_3 = \{0.2, 0.5, 1.0\}$ (with w_1=1.0 and w_2=1.0 fixed); edge (smoothness) weights $\beta \in \{10, 25, 30\}$ and $\gamma \in \{20, 30, 50, 80\}$; object seed radius $r_{\text{object}} \in \{4, 6, 8\}$; background band width $w_{\text{BG}} \in \{4, 6\}$; and morphological clean-up $m_{\min} \in \{80, 120, 200\}$, $r_{\text{close}} \in \{1, 2, 3\}$, $r_{\text{open}} \in \{0, 1\}$.

5.3.3 Applied Deep Learning Models

In this study, two different approaches were considered for the automatic segmentation of PET/CT scans of the prostate gland. The first approach performed segmentation using CT only. The second approach, conversely, involved developing novel multi-modal models—originally designed within the scope of this thesis—to simultaneously leverage the anatomical information from CT and the functional information provided by PET.

For hyperparameter optimization, a grid search strategy was adopted. Specifically, batch sizes of 2, 4, 8, and 16, loss functions including Dice Loss, IoU loss and Tversky loss, and learning rates of 1×10^{-3}, 1×10^{-4},and 1×10^{-5} were tested. The optimal combinations were selected based on the models' performance on the validation set. In all models, the number of training epochs was set to 200, and an early stopping mechanism with a patience value of 15 was employed. This configuration was designed to ensure a balanced and effective training process, maximizing both model accuracy and generalization capacity.

5.3.2.1 Segmentation Based on CT Only

The first approach utilized CT as the sole modality for segmentation. Accordingly, several models commonly adopted in the literature—and representing diverse

architectural families—were utilized: Attention U-Net, DynUNet, SegResNet, SV U-Net Transformer, U-Net, UNet++, U-Net Transformer, and YOLO11.

The selection aimed to cover different segmentation strategies, including attention mechanisms, residual blocks, transformer-based designs, and conventional convolutional architectures.

5.3.2.2 Segmentation Utilizing Both PET and CT Modalities

In the second approach, the segmentation process was designed to jointly leverage information from both PET and CT modalities. Specifically, it combines the complementary strengths of the two modalities—leveraging PET's functional sensitivity and CT's anatomical detail—to improve segmentation. Within the scope of this thesis, novel multimodal models were developed, and two distinct strategies were adopted.

- *Dual Encoder – Single Decoder Strategy:* In this strategy, CT and PET were processed by separate encoder architectures. The features extracted from both modalities were then merged through a shared decoder to produce the final segmentation output. The models designed under this architecture include: *DE SegResNet*, *DE UNETR*, and *DE U-Net*.
- *Fusion Strategy:* The architecture called *U-Net Fusion* is a U-Net that accepts both CT and PET as inputs. The CT (512×512) drives the network from the input, and the encoder progressively downsamples until the feature maps reach 128×128. At this stage, the PET (128×128) is introduced and fused with the CT stream—e.g., by concatenating PET with the CT feature maps and applying a 1×1 convolution to align channels.

These two strategies allowed a comprehensive comparison between single-modality and multimodal DL architectures for prostate segmentation.

Figure 5.2 presents schematic diagrams of two representative models designed in this study: DE-U-Net (A) and U-Net Fusion (B).

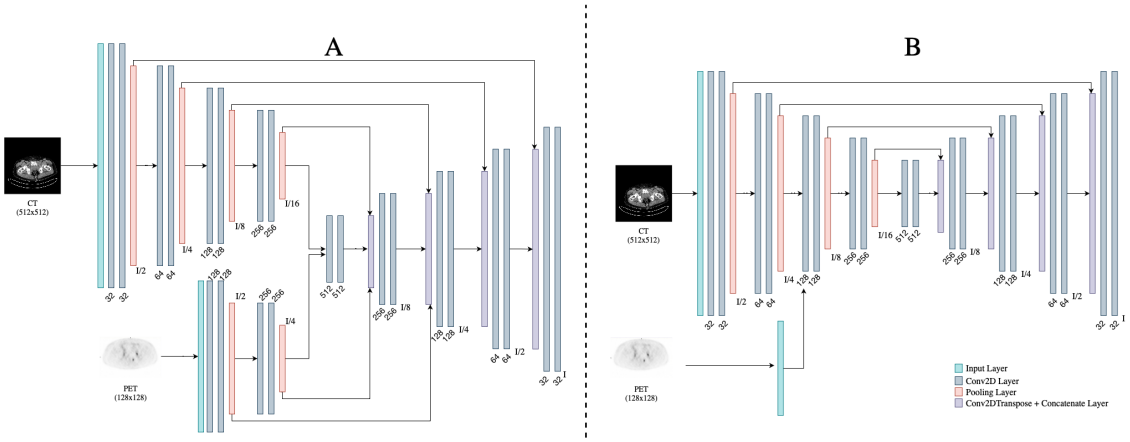


Figure 5.2 The architectures of DE U-Net (A) and U-Net Fusion (B)

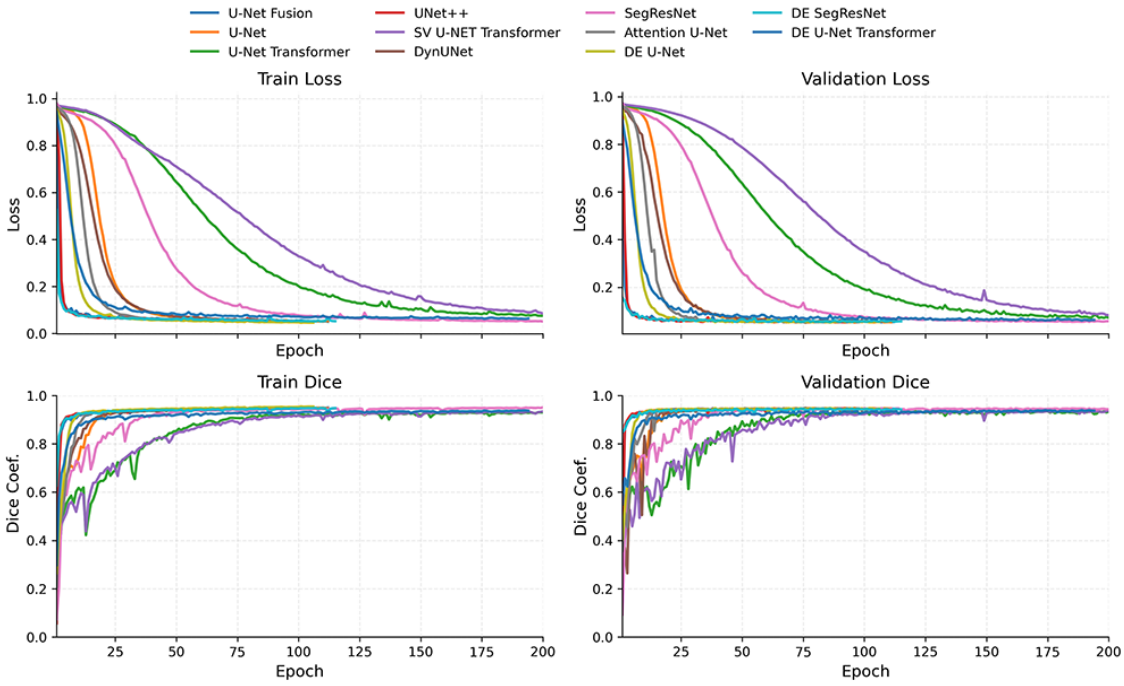


Figure 5.3 Loss and performance plots during model training

5.3.4 Statistical Analysis

U-Net, which is widely used as a reference model in the biomedical image segmentation literature and provides a strong baseline, was compared with Attention U-Net, which achieved the highest performance in this study. Additionally, Attention U-Net was compared against Multi-Atlas segmentation method. The DSC was computed per test sample, yielding paired observations on the same test set ($n = 83$). Normality was evaluated on the paired differences (U-Net − Attention U-Net and Multi-Atlas – Attention U-Net) using the Shapiro–Wilk test with visual inspection (histogram, Q–Q plot). As normality of differences was rejected both pairs ($p < 0.001$), with skewness

and kurtosis values also being outside the ± 1 range, the primary comparison was conducted using the Wilcoxon signed-ranks test (two-sided). Effect size was summarized using the rank-biserial correlation.

5.4 Results

All models were trained on the training dataset and subsequently evaluated on the test dataset to measure their performance. The models' loss and performance metrics during the training phase are displayed in Figure 5.3. Their final performance results on the test dataset are summarized in Table 5.1, while the outputs of the hyperparameter optimization process are presented in Table 5.2 and Table 5.3.

Table 5.1 Performance of the models on the test dataset.

		Model Name	#Params	Recall	Precision	DSC	IoU
Deep Learning Models	Only CT	Attention U-Net	7.9M	0.9516	0.9374	0.9426	0.8936
		DynUNet	1.9M	0.9487	0.9395	0.9422	0.8929
		SegResNet	1.5M	0.9524	0.9362	0.9424	0.8934
		SV U-Net Transformer	116M	0.9489	0.9081	0.9256	0.8643
		U-Net	1.9M	0.9527	0.9152	0.9316	0.8746
		UNet++	2.4M	0.9552	0.9341	0.9425	0.8936
		U-Net Transformer	116M	0.9297	0.9213	0.9227	0.8595
		YOLO11	22M	0.9645	0.8750	0.9163	0.8467
	PET/CT	DE SegResNet	8.9M	0.9490	0.9345	0.9398	0.8888
		DE UNETR	231M	0.9436	0.9147	0.9259	0.8655
		DE U-Net	14M	0.9527	0.9328	0.9407	0.8902
		U-Net Fusion	8.1M	0.9468	0.9386	0.9407	0.8907
Classic Methods		Graph-Cut	-	0.7567	0.4604	0.4024	0.3029
		Multi-Atlas	-	0.7351	0.8645	0.7752	0.6409

Table 5.2 Optimal hyperparameters of the deep learning models.

	Model Name	Batch Size	Loss Function	Learning Rate
Only CT	Attention U-Net	2	Dice Loss	1×10^{-4}
	DynUNet	4	Dice Loss	1×10^{-4}
	SegResNet	8	Dice Loss	1×10^{-4}
	SV U-Net Transformer	4	Dice Loss	1×10^{-4}
	U-Net	2	Dice Loss	1×10^{-5}
	UNet++	4	Dice Loss	1×10^{-3}
	U-Net Transformer	4	Dice Loss	1×10^{-4}
PET/CT	DE SegResNet	4	Dice Loss	1×10^{-3}
	DE UNETR	2	Dice Loss	1×10^{-4}
	DE U-Net	2	Dice Loss	1×10^{-4}
	U-Net Fusion	4	Dice Loss	1×10^{-3}

Table 5.3 Optimal hyperparameters of the classic methods.

Model Name	Hyperparameters
Graph-Cut	$w_3 = 1.0;\ \beta = 25$ and $\gamma = 30;\ r_{\text{object}} = 6;\ w_{\text{BG}} = 4;$ $m_{\min} = 200; r_{\text{close}} = 1; r_{\text{open}} = 1$
Multi-Atlas	$N_A = 5;\ M = \text{CC};\ \rho_s = 0.4;\ T = 100;\ g_{\text{bs}} = 12;\ T_{\text{bs}} = 60;$ $F = \text{staple with } \tau = 0.5;\ r_{\text{morph}} = 2$

The results indicate that the eight models involved in the first approach, trained solely on CT images, yielded very close performance values when assessed by the DSC and IoU metrics. Among these, UNet++, Attention U-Net, and SegResNet demonstrated the highest and most balanced performance. Specifically, the top segmentation results were achieved by Attention U-Net (DSC:0.9426, IoU:0.8936), UNet++ (DSC:0.9425, IoU:0.8936), and SegResNet (DSC:0.9424, IoU:0.8934). These models were closely followed by DynUNet (DSC:0.9422).

Within the second approach, the DE U-Net and U-Net Fusion models demonstrated the best performance, achieving a DSC score of 0.9407 and IoU values of 0.8902 and 0.8907, respectively. DE SegResNet, another model utilizing the Dual Encoder architecture, closely followed with a DSC score of 0.9398. However, the most complex model in this group, the DE UNETR, which contains 231M parameters, performed below expectations, yielding a DSC score of only 0.9259.

Among the classical methods, Graph-Cut yielded the lowest results, with a DSC score of 0.4024 and an IoU of 0.3029. The Multi-Atlas method demonstrated higher performance, achieving a DSC score of 0.7752 and an IoU of 0.6409, but still trailed significantly behind all DL-based approaches.

Across the DL models, the Dice Loss function was observed to produce the best results, and the optimal learning rates generally clustered within the $1\times10-4$ to $1\times10-3$ range.

Figure 5.4 and Figure 5.5 display the segmentation results obtained by all methods. To enhance visual clarity and ensure direct comparability of structural details, all presented images are crops, derived by extracting a 220×220 pixel window centered on the original data. It should be noted that the training and quantitative evaluations were conducted on the full Field of View (FOV). Figure 5.4 presents the outputs of the DL models. The columns show (A) the input image, (B) the ground truth mask, followed by the prediction masks from the 12 deep models: (C) Attention U-Net, (D) DynUNet, (E) SegResNet, (F) SV U-Net Transformer, (G) U-Net, (H) UNet++, (I) U-Net Transformer, (J) YOLOv11, (K) DE SegResNet, (L) DE UNETR, (M) DE U-Net, and (N) U-Net Fusion. Figure 5.5 summarizes the results from the classical methods. This figure organizes the images by row, presenting (A) the input image, (B) the ground truth mask, (C) the Multi-Atlas predictions, and (D) the Graph-Cut predictions.

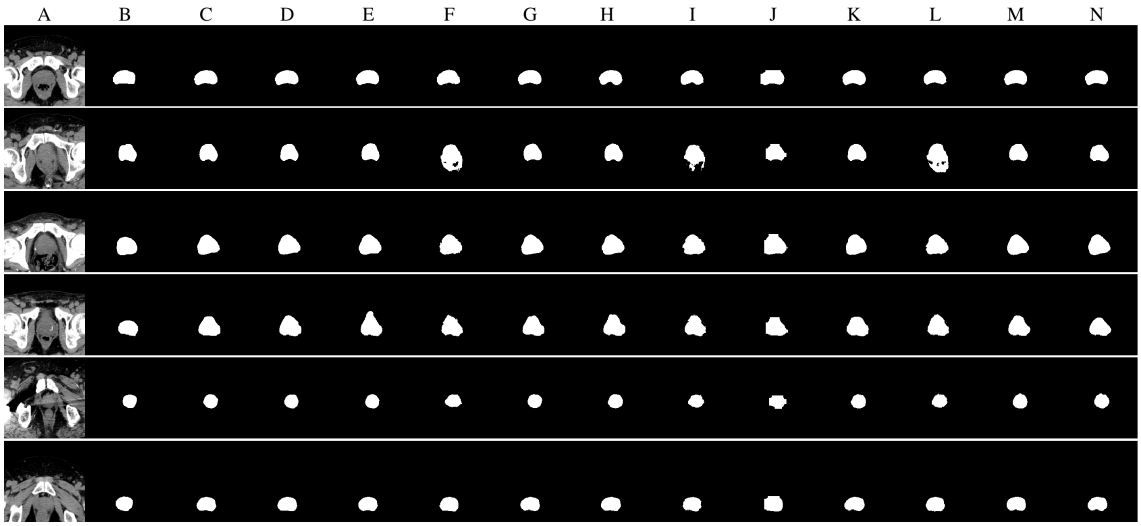


Figure 5.4 Prediction samples of deep-learning-based models from test dataset

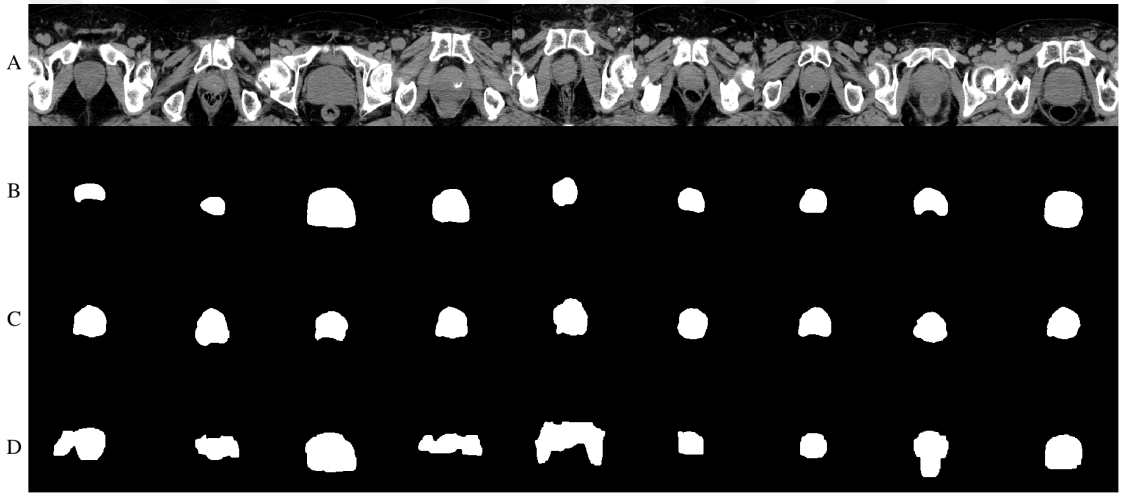


Figure 5.5 Prediction samples of classic methods from test dataset

The Wilcoxon Signed-Rank Test indicated that Attention U-Net performed statistically and significantly better than both U-Net and the Multi-Atlas segmentation method. The comparison between U-Net and Attention U-Net revealed a significant difference ($W = 684,\ p < 0.001$), supported by a large effect size (rank-biserial $r_{rb} = -0.6075$). The performance advantage of Attention U-Net over the Multi-Atlas method was far more substantial ($W = 24,\ p < 0.001$), with a very large effect size ($r_{rb} = -0.9862$) suggesting an almost complete superiority in performance.

The qualitative performance and explainability analyses of Attention U-Net, which achieved the highest IoU and DSC scores among models trained for prostate segmentation, are presented in Figure 5.6. Rows show (A) input images, (B) ground truth masks, (C) model predictions, (D) Grad-CAM visualizations, and (E) Saliency Maps. To

enhance the readability of the visualizations, each example was cropped using a fixed 256×256 pixel window centered on the prostate; this same cropping was applied to the input, ground truth, prediction, Grad-CAM, and Saliency Maps. Visual inspections confirm that the predictions of the Attention U-Net model exhibit a high degree of structural and spatial consistency with the ground truth masks. The model was observed to detect the boundaries of the prostate gland with high accuracy. Grad-CAM heatmaps show that the model correctly focuses its attention on the anatomical location of the prostate gland when making its segmentation decision. High activation regions (red/yellow) in the heatmaps clearly correspond to prostate tissue. Similarly, Saliency Map analyses reveal that the most influential pixels forming the model's prediction are predominantly located within the prostate boundaries. This visual evidence demonstrates that the model's quantitative success is supported by its ability to focus accurately on the targeted anatomical structure.

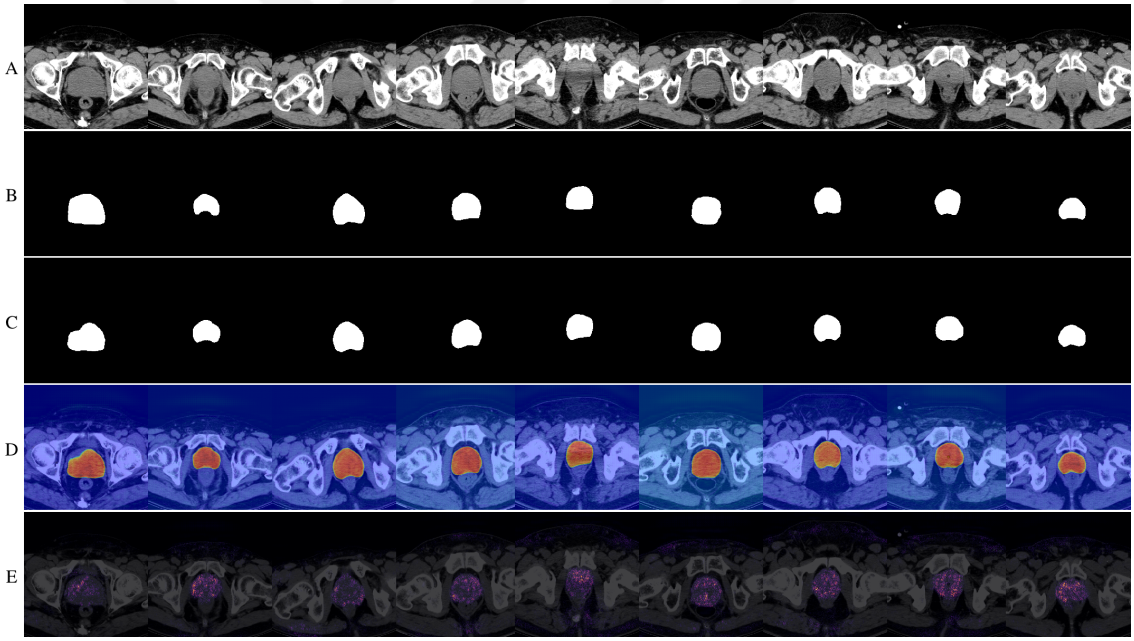


Figure 5.6 Samples showing explainability analysis of the model.

5.5 Discussion

Accurate prostate segmentation is a necessity in modern oncology, serving as a critical prerequisite for precise dose planning in radiotherapy and reliable volume assessment during active surveillance. This study affirms the transformative role of DL in this domain, demonstrating that highly competitive performance can be achieved

across various architectures, a result that sharply contrasts with the markedly lower performance observed in classical segmentation methods.

Firstly, the results indicate that CNN-based architectures yield better results than Transformer-based models in CT-only models. This finding suggests that models with a relatively smaller number of parameters can learn more efficiently when trained on limited datasets. Although Transformer-based models are theoretically capable of capturing long-range dependencies more effectively, their high parameter count and the restricted data size may have led to issues such as overfitting or poor generalization.

Another key finding of the study is that models trained solely with CT, such as Attention U-Net, UNet++, and SegResNet, achieved partially higher DSC scores than the proposed multimodal PET/CT models. The initial hypothesis was that the metabolic activity information captured in PET scans would provide an additional advantage for delineating the prostate boundaries. However, the results suggest that, for the architectures and dataset used in this study, the high-resolution anatomical detail available in CT alone is sufficient for accurate prostate segmentation. There may be several possible reasons why adding PET data did not significantly improve performance. The metabolic activity shown in PET (e.g., areas of high SUV values) may not always align perfectly with the prostate's anatomical boundaries, potentially introducing confusing signals for the model. Moreover, dual-encoder architectures designed to process multimodal inputs might require a larger dataset or alternative architectural designs to effectively learn the complex relationships between these two distinct imaging modalities.

Among the multimodal models, DE U-Net (DSC: 0.9407, IoU: 0.8902) and U-Net Fusion (DSC: 0.9407, IoU: 0.8907) yielded the highest results, suggesting these architectures are more effective than others at integrating PET and CT data. Conversely, the relatively lower performance of DE UNETR (DSC: 0.9259, IoU: 0.8655) may indicate that transformer architectures require significantly larger datasets or specialized pre-training strategies to succeed in biomedical image fusion tasks. A key finding is the consistent advantage demonstrated by multimodal models over their single-modality counterparts (DE UNETR vs. U-Net Transformer comparison (DSC: 0.9227); DE U-Net/U-Net Fusion vs. U-Net comparison (DSC: 0.9316)). This demonstrates that PET information provides a benefit in slightly improving overall segmentation performance.

From the perspective of model efficiency, the results are quite clear. Models with very low parameter counts, such as SegResNet (1.5M parameters) and DynUNet (1.9M

parameters), outperformed extremely large Transformer-based architectures like SV U-Net Transformer (116M) and DE UNETR (231M), which have nearly 100 times more parameters. This finding demonstrates that overly complex models are not always the optimal solution for specific tasks like biomedical image segmentation. In fact, when the dataset is limited, such models are more prone to overfitting and may generalize poorly to unseen test data. For clinical applications, where faster training times, less demanding hardware requirements, and easier deployment are crucial, the superior performance of these efficient and compact models represents a significant advantage.

The explainability analyses presented in Figure 5.6 demonstrate that the Attention U-Net not only achieved high quantitative performance but also focused accurately on the anatomically relevant prostate region. This outcome proves that the model's success is not coincidental but is based on a clinically meaningful learning process. Grad-CAM heatmaps reveal that the model does not operate as a *black box*, instead directing its attention toward the correct anatomical location — the prostate gland. This focused behavior is further confirmed at the pixel level through saliency maps, which highlight the regions most influential in the decision-making of the model, thereby reinforcing the reliability of the developed AI solution. In summary, Attention U-Net stands out not only for its strong quantitative results but also for its interpretability and logical decision patterns, providing a solid foundation for clinical acceptance and systematic error analysis.

5.6 Conclusion

This study provides important insights into prostate segmentation by evaluating both classical methods and DL approaches under single-modal and multimodal approaches. The findings demonstrate that, particularly with limited datasets, CNN-based architectures with smaller parameter sets perform more successfully than highly complex Transformer models.

Surprisingly, incorporating PET data did not improve performance, likely because the high-resolution anatomical detail in CT alone was sufficient. This finding points to the need for larger datasets or specialized architectures to more effectively leverage multimodal integration in future work. Nevertheless, given that the Dual Encoder or Fusion approaches enhanced the performance of the U-Net and U-Net Transformer

models, future research should apply these successful multi-modal strategies to the top-performing single-modality architectures, such as Attention U-Net and UNet++. This extension is hypothesized to further leverage the PET functional information and potentially yield state-of-the-art results by combining superior single-modality performance with effective multi-modality fusion.

Ultimately, the validation of successful models like Attention U-Net must extend beyond quantitative metrics. The proof of reliability offered by explainability analyses (such as Grad-CAM) is a critical step in establishing clinical trust. Demonstrating that the model focuses attention on the anatomically correct areas proves that its success is based on a logically sound learning process, thereby significantly enhancing its reliability for clinical adoption and facilitating effective error analysis. Overall, this work establishes a strong foundation, indicating that parameter-efficient, deep-learning-based models, supported by interpretable decision-making, are the most promising avenue for developing robust, clinically viable prostate segmentation solutions.

Chapter 6

Conclusions and Future Prospects

6.1 Conclusions

This thesis explores three distinct segmentation challenges within biomedical imaging: dental plaque, low-grade glioma (LGG), and the prostate gland. These problems were tackled utilizing both DL models and traditional image processing techniques. The findings demonstrate the effectiveness and limitations of DL architectures across various data types and clinical scenarios, providing valuable contributions to the existing body of knowledge in this domain.

The most fundamental and consistent finding of this thesis is that DL–based models have demonstrated an overwhelming superiority over classical image processing methods across all three investigated problems. Classical approaches proved inadequate in capturing complex patterns such as color and edge ambiguities in dental plaque, as well as heterogeneous tumor structures and irregular boundaries in LGG. In contrast, DL models exhibited significantly higher accuracy and consistency, owing to their ability to learn hierarchical and contextual features directly from the data.

A second key finding of the thesis is that the best-performing model architecture varied according to the specific problem and the characteristics of the dataset. In dental plaque segmentation, the Transformer-based U-Net Transformer model achieved the highest accuracy, highlighting the potential of such architectures in capturing fine structural details. In contrast, for LGG and prostate segmentation, more compact CNN-based architectures such as UNet++ and Attention U-Net delivered more balanced and efficient results than large-scale Transformer models. This suggests that model complexity and the number of parameters are not the primary determinants of performance—particularly when dealing with limited datasets—and that overly complex models may be more susceptible to overfitting. This is a major practical benefit for

clinical deployment, as it implies reduced hardware demands and accelerated training cycles.

The third major finding of the thesis is that, in the LGG study, a novel hybrid architecture named Super Vision U-Net Transformer was developed by combining the strengths of the U-Net Transformer and Super Vision U-Net models, both of which had achieved strong performance in Chapter 3 – Microbial Dental Plaque Segmentation study. However, this hybrid model did not yield the expected level of success on the LGG dataset. This outcome demonstrates that merging two high-performing models does not necessarily lead to improved performance; on the contrary, interactions between architectural components can produce complex and sometimes counterproductive effects across different data types and problem domains. This finding highlights an important insight — that greater architectural complexity or model fusion does not inherently guarantee better results, emphasizing the need for problem-specific model design in biomedical image segmentation.

The prostate gland segmentation study revealed valuable and unexpected findings regarding the use of multimodal data. Contrary to the initial hypothesis, incorporating metabolic information from PET did not consistently lead to a significant performance improvement compared to models trained solely on CT. This suggests that the high-resolution anatomical detail in CT alone may be sufficient, or that larger datasets and specialized architectural designs are required to learn the complex relationship between the two distinct modalities. Nevertheless, the observation that dual-encoder approaches such as DE U-Net achieved slightly better performance than single-modality models indicates that, with appropriate integration strategies, multimodal data can still offer meaningful benefits.

One of the strongest aspects of the thesis is that the models were evaluated not only through quantitative metrics but also using explainability (XAI) techniques such as Grad-CAM and Saliency Maps. In all three studies, the best-performing models were shown to focus their decision-making processes on clinically meaningful regions—such as plaque accumulation areas, tumor tissue, and the prostate gland. These analyses helped demystify the models' *black-box* nature, demonstrating that their success was not coincidental but grounded in a logical and interpretable learning process. Establishing model reliability in this manner represents a critical step toward clinical adoption and systematic error analysis.

In conclusion, this thesis strongly demonstrates the revolutionary potential of DL models in biomedical image segmentation. The studies highlight the importance of task-specific architectural selection and parameter efficiency, draw attention to the challenges of multimodal data integration, and underscore explainability as an essential component for the clinical reliability of AI systems. The developed frameworks have the potential to enhance clinical decision-support mechanisms and improve the quality of patient care across a wide range of applications—from providing a non-invasive and objective alternative for monitoring dental hygiene to improving the accuracy of treatment planning in oncology.

6.2 Societal Impact and Contribution to Global Sustainability

The DL–based segmentation models presented in this thesis not only represent scientific and technological advancements but also hold the potential to generate significant positive societal impacts and contribute to global sustainability goals. In the healthcare domain in particular, such AI applications can deliver broad social benefits by accelerating diagnostic processes, optimizing treatment planning, and enhancing the overall quality of patient care.

AI-powered diagnostic and screening tools can improve access to healthcare, particularly in rural or underserved regions where access to medical specialists is limited. As demonstrated in the dental plaque segmentation example, AI models integrated into mobile devices could enable individuals to monitor their oral hygiene regularly, promoting early diagnosis and preventive care. Such applications have the potential to reduce health disparities and help broader populations gain access to high-quality healthcare services.

DL models enhance resource efficiency within healthcare systems by automating manual and time-consuming processes. For instance, the rapid and accurate analysis of medical images can significantly reduce clinicians' workloads, allowing them to focus on a greater number of patients. Such automation contributes to more effective use of human resources while also helping to lower diagnostic and treatment costs. These cost

reductions, in turn, can make healthcare services more affordable and support the development of a more sustainable healthcare system.

The early and accurate diagnosis of critical diseases such as cancer (e.g., LGG and prostate cancer) can significantly enhance treatment success and improve patients' quality of life. Segmentation models facilitate the precise delineation of tumors or lesions, potentially reducing the need for invasive procedures and enabling the implementation of more effective treatments with fewer side effects. By helping individuals lead longer and healthier lives, these technologies ultimately contribute to improved societal well-being.

The integration of AI into healthcare services supports the development of more resilient and inclusive health infrastructures through the development and widespread adoption of innovative technologies. Furthermore, it can contribute to sustainability efforts indirectly by enabling more efficient use of resources and reducing the environmental footprint.

In conclusion, this thesis demonstrates that DL–based segmentation is not merely a technical achievement but also holds the potential to transform healthcare, enhance societal well-being, and play a crucial role in advancing global sustainability goals. The ethical and responsible development and implementation of these technologies will be essential in building a healthier, more equitable, and more sustainable world for the future.

6.3 Future Prospects

The findings presented in this thesis establish a strong foundation, yet they also highlight several key avenues for future research. Subsequent studies should focus on advancing the accuracy, generalizability, and clinical integration of the developed models to fully realize their potential.

Enrichment and Diversification of Datasets

The studies in this thesis have underscored the critical impact of dataset scope and quality on model performance. A crucial next step is the collection of larger and more diverse datasets.

- *Multi-Center Data Collection*: For studies conducted with custom-collected data, such as dental plaque and prostate segmentation, gathering data from multiple institutions and different devices (e.g., various intraoral camera

brands) would generate more heterogeneous datasets, thereby enhancing the models' generalization capabilities.

- *Detailed Annotation*: Enriching datasets with information beyond segmentation masks could unlock new possibilities. For instance, adding labels for plaque density to the dental plaque dataset would enable AI systems to assess not only the presence but also the severity of the plaque.

Advanced Model Architectures and Optimization Strategies

This thesis has shown that the most complex or largest model is not always the best solution; instead, task-specific and parameter-efficient architectures often achieve superior performance.

- *Hybrid Architectures*: Future work should explore novel hybrid architectures that combine the global context understanding of Transformers with the efficiency of CNNs in extracting local features. As suggested in the LGG study, strategies that harmoniously merge global and local feature representations could leverage the best of both worlds.
- *Customized Loss Functions*: Designing specialized loss functions can improve model accuracy, particularly in cases with ambiguous tumor boundaries or small segmentation targets. Such functions could place greater emphasis on boundaries or better manage tissue imbalances.

Deepening Multimodal Integration

The prostate segmentation study revealed both the challenges and the potential of using multimodal data.

- *Advanced Fusion Techniques*: A concrete next step is to apply the successful fusion strategies from this thesis to the top-performing single-modality architectures, such as Attention U-Net and UNet++. This approach is hypothesized to leverage PET's functional information more effectively and yield state-of-the-art results.

Clinical Integration and Patient-Centric Applications

To transform these models into practical value, the focus must shift toward clinical workflow integration and end-user applications.

- *Mobile Health and Personal Monitoring*: One of the most exciting future directions from the dental plaque study is the potential for individuals to monitor their oral hygiene using smartphone cameras. This would require making models robust to variations in lighting and camera types, necessitating the use of techniques like domain adaptation.
- *The Role of Explainability*: The use of methods like Grad-CAM and Saliency Maps across all three studies enhanced clinical trust by making the models' decision-making processes transparent. Future work could transform these explainability outputs into interactive interfaces that provide direct feedback to clinicians, accelerating model adoption and facilitating effective error analysis.
- *Longitudinal Analysis*: The developed models can be used to monitor patient conditions over time. For instance, objectively tracking plaque accumulation between dental visits or assessing changes in tumor volume during treatment would greatly contribute to personalized care and preventive medicine.

In summary, the findings of this thesis provide a robust roadmap for the future of AI in biomedical image segmentation. Future research built upon richer datasets, efficient and intelligent model architectures, deeper multimodal integration, and patient-centric clinical applications will continue to accelerate the transformation of this field.

BIBLIOGRAPHY

[1] N. Chater, "The logic of human learning," *Nature*, vol. 407, no. 6804, pp. 572–573, Oct. 2000.

[2] S. Zeki, *A vision of the brain.* in A vision of the brain. Cambridge, MA, US: Blackwell Scientific Publications, 1993, pp. xi, 366.

[3] al-Ḥasan ibn al-Ḥasan Ibn al-Hayṯam and ʿAbd al-Ḥamīd Ṣabrah, *The Optics of Ibn Al-Haytham*. in Studies of the Warburg Institute, no. 40. London: Warburg Institute, University of London, 1989.

[4] A. I. Khan and S. Al-Habsi, "Machine learning in computer vision," *Procedia Computer Science*, vol. 167, pp. 1444–1451, 2020.

[5] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. USA: Prentice-Hall, Inc., 2006.

[6] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: a survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, pp. 1–1, 2021.

[7] K. K. Brar *et al.*, "Image segmentation review: Theoretical background and recent advances," *Information Fusion*, vol. 114, p. 102608, Feb. 2025.

[8] A. Rezaei and F. Asadi, "Systematic review of image segmentation using complex networks," 2024, *arXiv*.

[9] J. Zhang *et al.*, "Advances in attention mechanisms for medical image segmentation," *Computer Science Review*, vol. 56, p. 100721, May 2025.

[10] S. Borse, H. Park, H. Cai, D. Das, R. Garrepalli, and F. Porikli, "Panoptic, instance and semantic relations: a relational context encoder to enhance panoptic segmentation," 2022, *arXiv*.

[11] A. Maier, S. Steidl, V. Christlein, and J. Hornegger, Eds., *Medical Imaging Systems: An Introductory Guide*, vol. 11111. in Lecture Notes in Computer Science, vol. 11111. Cham: Springer International Publishing, 2018.

[12] B. Sahiner *et al.*, "Deep learning in medical imaging and radiation therapy," *Medical Physics*, vol. 46, no. 1, Jan. 2019.

[13] D. Battalapalli, B. V. V. S. N. P. Rao, P. Yogeeswari, C. Kesavadas, and V. Rajagopalan, "An optimal brain tumor segmentation algorithm for clinical MRI dataset with low resolution and non-contiguous slices," *BMC Med Imaging*, vol. 22, no. 1, p. 89, May 2022.

[14] K. Suzuki, "Overview of deep learning in medical imaging," *Radiol Phys Technol*, vol. 10, no. 3, pp. 257–273, Sept. 2017.

[15] M. Li, Y. Jiang, Y. Zhang, and H. Zhu, "Medical image analysis using deep learning algorithms," *Front. Public Health*, vol. 11, p. 1273253, Nov. 2023.

[16] J. D. Firth, C. P. Conlon, and T. M. Cox, *Oxford textbook of medicine*, 6th ed. Oxford: Oxford university press, 2020.

[17] A. Shomirov, J. Zhang, and M. M. Billah, "Brain tumor segmentation of HGG and LGG MRI images using WFL-based 3D U-net," *JBiSE*, vol. 15, no. 10, pp. 241–260, 2022, doi: 10.4236/jbise.2022.1510022.

[18] Y. Güzel and Z. Aydın, "A comparative study of UNet variants for low-grade glioma segmentation in magnetic resonance imaging," *Journal of İnönü University Vocational School of Health Services*, vol. 13, no. 2, pp. 344–355, June 2025.

[19] Md. E. Rayed, S. M. S. Islam, S. I. Niha, J. R. Jim, M. M. Kabir, and M. F. Mridha, "Deep learning for medical image segmentation: State-of-the-art advancements

and challenges," *Informatics in Medicine Unlocked*, vol. 47, p. 101504, 2024, doi: 10.1016/j.imu.2024.101504.
[20] Z. Vosoughi *et al.*, "Prediction of the gleason score of prostate cancer patients using 68Ga-PSMA-PET/CT radiomic models," *J. Med. Biol. Eng.*, Oct. 2024, doi: 10.1007/s40846-024-00906-3.
[21] B. Wan, B. Hu, M. Zhao, K. Li, and X. Ye, "Deep learning-based magnetic resonance image segmentation technique for application to glioma," *Front. Med.*, vol. 10, p. 1172767, Nov. 2023, doi: 10.3389/fmed.2023.1172767.
[22] I. Isikay, E. Cekic, B. Baylarov, O. Tunc, and S. Hanalioglu, "Narrative review of patient-specific 3D visualization and reality technologies in skull base neurosurgery: enhancements in surgical training, planning, and navigation," *Front. Surg.*, vol. 11, p. 1427844, July 2024, doi: 10.3389/fsurg.2024.1427844.
[23] R. Pratt, J. Deprest, T. Vercauteren, S. Ourselin, and A. L. David, "Computer-assisted surgical planning and intraoperative guidance in fetal surgery: a systematic review," *Prenatal Diagnosis*, vol. 35, no. 12, pp. 1159–1166, Dec. 2015, doi: 10.1002/pd.4660.
[24] M. A. Naser and M. J. Deen, "Brain tumor segmentation and grading of lower-grade glioma using deep learning in MRI images," *Computers in Biology and Medicine*, vol. 121, p. 103758, June 2020, doi: 10.1016/j.compbiomed.2020.103758.
[25] I. Despotović, B. Goossens, and W. Philips, "MRI Segmentation of the Human Brain: Challenges, Methods, and Applications," *Computational and Mathematical Methods in Medicine*, vol. 2015, pp. 1–23, 2015, doi: 10.1155/2015/450341.
[26] A. Verma, S. N. Shivhare, S. P. Singh, N. Kumar, and A. Nayyar, "Comprehensive Review on MRI-Based Brain Tumor Segmentation: A Comparative Study from 2017 Onwards," *Arch Computat Methods Eng*, vol. 31, no. 8, pp. 4805–4851, Dec. 2024, doi: 10.1007/s11831-024-10128-0.
[27] B. Tez, Y. Güzel, B. Kızıltan Eliaçık, and Z. Aydın, "Deep-Learning-Based AI-Model for Predicting Dental Plaque in the Young Permanent Teeth of Children Aged 8–13 Years," *Children*, vol. 12, no. 4, p. 475, Apr. 2025.
[28] X.-W. Chen and X. Lin, "Big Data Deep Learning: Challenges and Perspectives," *IEEE Access*, vol. 2, pp. 514–525, 2014, doi: 10.1109/ACCESS.2014.2325029.
[29] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*, Third edition, Global edition. in Prentice Hall series in artificial intelligence. Boston Columbus Indianapolis: Pearson, 2016.
[30] S. S. Haykin, *Neural networks and learning machines*, 3. ed., Internat. ed. Upper Saddle River, NJ Munich: Pearson, 2009.
[31] I. Goodfellow, A. Courville, and Y. Bengio, *Deep learning*. in Adaptive computation and machine learning. Cambridge, Massachusetts: The MIT Press, 2016.
[32] F. Rosenblatt, "The perceptron: A probabilistic model for information storage and organization in the brain.," *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958, doi: 10.1037/h0042519.
[33] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986, doi: 10.1038/323533a0.
[34] I. Goodfellow, A. Courville, and Y. Bengio, *Deep learning*. in Adaptive computation and machine learning. Cambridge, Massachusetts: The MIT Press, 2016.

[35] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Proceedings of the 27th International Conference on International Conference on Machine Learning*, in ICML’10. Madison, WI, USA: Omnipress, 2010, pp. 807–814.
[36] K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural Networks*, vol. 2, no. 5, pp. 359–366, Jan. 1989, doi: 10.1016/0893-6080(89)90020-8.
[37] C. M. Bishop, *Pattern recognition and machine learning*. in Information science and statistics. New York: Springer, 2006.
[38] N. Qian, “On the momentum term in gradient descent learning algorithms,” *Neural Networks*, vol. 12, no. 1, pp. 145–151, Jan. 1999, doi: 10.1016/S0893-6080(98)00116-6.
[39] J. Duchi, E. Hazan, and Y. Singer, “Adaptive Subgradient Methods for Online Learning and Stochastic Optimization,” *Journal of Machine Learning Research*, vol. 12, no. 61, pp. 2121–2159, 2011.
[40] T. Tieleman, “Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude,” *COURSERA: Neural networks for machine learning*, vol. 4, no. 2, p. 26, 2012.
[41] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
[42] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, “Gradient-based learning applied to document recognition,” *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998, doi: 10.1109/5.726791.
[43] S. Hochreiter and J. Schmidhuber, “Long Short-Term Memory,” *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.
[44] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling,” 2014, *arXiv*. doi: 10.48550/ARXIV.1412.3555.
[45] J. L. Elman, “Finding Structure in Time,” *Cognitive Science*, vol. 14, no. 2, pp. 179–211, Mar. 1990, doi: 10.1207/s15516709cog1402_1.
[46] A. Sherstinsky, “Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network,” *Physica D: Nonlinear Phenomena*, vol. 404, p. 132306, Mar. 2020, doi: 10.1016/j.physd.2019.132306.
[47] D. Bahdanau, K. Cho, and Y. Bengio, “Neural Machine Translation by Jointly Learning to Align and Translate,” 2014, *arXiv*. doi: 10.48550/ARXIV.1409.0473.
[48] A. Vaswani *et al.*, “Attention Is All You Need,” 2017, *arXiv*. doi: 10.48550/ARXIV.1706.03762.
[49] K. M. Ting, “Confusion Matrix,” in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds., Boston, MA: Springer US, 2010, pp. 209–209. doi: 10.1007/978-0-387-30164-8_157.
[50] K. M. Ting, “Precision and Recall,” in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds., Boston, MA: Springer US, 2010, pp. 781–781. doi: 10.1007/978-0-387-30164-8_652.
[51] T. Sorensen, “A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons,” *Biologiske skrifter*, vol. 5, pp. 1–34, 1948.
[52] L. R. Dice, “Measures of the Amount of Ecologic Association Between Species,” *Ecology*, vol. 26, no. 3, pp. 297–302, July 1945, doi: 10.2307/1932409.

[53] P. Jaccard, “The Distribution Of The Flora In The Alpine Zone,” *New Phytologist*, vol. 11, no. 2, pp. 37–50, Feb. 1912, doi: 10.1111/j.1469-8137.1912.tb05611.x.
[54] Y. Bengio, “Practical recommendations for gradient-based training of deep architectures,” 2012, *arXiv*. doi: 10.48550/ARXIV.1206.5533.
[55] J. Bergstra and Y. Bengio, “Random search for hyper-parameter optimization,” *J. Mach. Learn. Res.*, vol. 13, no. null, pp. 281–305, Feb. 2012.
[56] J. Snoek, H. Larochelle, and R. P. Adams, “Practical Bayesian optimization of machine learning algorithms,” in *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, in NIPS’12. Red Hook, NY, USA: Curran Associates Inc., 2012, pp. 2951–2959.
[57] A. Bali and S. N. Singh, “A Review on the Strategies and Techniques of Image Segmentation,” in *2015 Fifth International Conference on Advanced Computing & Communication Technologies*, 2015, pp. 113–120. doi: 10.1109/ACCT.2015.63.
[58] Y. Yu *et al.*, “Techniques and Challenges of Image Segmentation: A Review,” *Electronics*, vol. 12, no. 5, p. 1199, Mar. 2023, doi: 10.3390/electronics12051199.
[59] N. Ofir and J.-C. Nebel, “Classic versus deep learning approaches to address computer vision challenges,” 2021, *arXiv*. doi: 10.48550/ARXIV.2101.09744.
[60] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, “SLIC Superpixels Compared to State-of-the-Art Superpixel Methods,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.
[61] C. Rother, V. Kolmogorov, and A. Blake, “‘GrabCut’: interactive foreground extraction using iterated graph cuts,” *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, Aug. 2004.
[62] D. J. Bora, A. K. Gupta, and F. A. Khan, “Comparing the Performance of L*A*B* and HSV Color Spaces with Respect to Color Image Segmentation,” 2015, *arXiv*. doi: 10.48550/ARXIV.1506.01472.
[63] A. Vadivel, M. Mohan, S. Sural, and A. K. Majumdar, “Segmentation Using Saturation Thresholding and Its Application in Content-Based Retrieval of Images,” in *Image Analysis and Recognition*, vol. 3211, A. Campilho and M. Kamel, Eds., in Lecture Notes in Computer Science, vol. 3211. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 33–40.
[64] L. Vincent and P. Soille, “Watersheds in digital spaces: an efficient algorithm based on immersion simulations,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583–598, 1991.
[65] B. Li, M. Pan, and Z. Wu, “An improved segmentation of high spatial resolution remote sensing image using Marker-based Watershed Algorithm,” in *2012 20th International Conference on Geoinformatics*, 2012, pp. 1–5.
[66] T. Rohlfing, D. B. Russakoff, and C. R. Maurer, “Performance-based classifier combination in atlas-based image segmentation using expectation-maximization parameter estimation,” *IEEE Transactions on Medical Imaging*, vol. 23, no. 8, pp. 983–994, 2004.
[67] J. E. Iglesias and M. R. Sabuncu, “Multi-atlas segmentation of biomedical images: A survey,” *Medical Image Analysis*, vol. 24, no. 1, pp. 205–219, Aug. 2015.
[68] H. Wang, J. W. Suh, S. R. Das, J. B. Pluta, C. Craige, and P. A. Yushkevich, “Multi-Atlas Segmentation with Joint Label Fusion,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 3, pp. 611–623, 2013.

[69] Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.

[70] D. Freedman and T. Zhang, “Interactive graph cut based segmentation with shape priors,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, 2005, pp. 755–762 vol. 1.

[71] O. Ronneberger, P. Fischer, and T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation,” in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds., Cham: Springer International Publishing, 2015, pp. 234–241. doi: 10.1007/978-3-319-24574-4_28.

[72] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[73] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, “Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation,” presented at the Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 2018, pp. 801–818.

[74] Zhī Huá Zhàn lán, “Mask R-CNN Architecture.” [Online]. Available: https://zhuanlan.zhihu.com/p/62492064

[75] M. J. Cardoso *et al.*, “MONAI: An open-source framework for deep learning in healthcare,” 2022, *arXiv*. doi: 10.48550/ARXIV.2211.02701.

[76] J. Le’Clerc Arrastia *et al.*, “Deeply Supervised UNet for Semantic Segmentation to Assist Dermatopathological Assessment of Basal Cell Carcinoma,” *Journal of Imaging*, vol. 7, no. 4, Art. no. 4, Apr. 2021, doi: 10.3390/jimaging7040071.

[77] O. Petit, N. Thome, C. Rambour, L. Themyr, T. Collins, and L. Soler, “U-Net Transformer: Self and Cross Attention for Medical Image Segmentation,” in *Machine Learning in Medical Imaging*, C. Lian, X. Cao, I. Rekik, X. Xu, and P. Yan, Eds., Cham: Springer International Publishing, 2021, pp. 267–276. doi: 10.1007/978-3-030-87589-3_28.

[78] O. Oktay *et al.*, “Attention U-Net: Learning Where to Look for the Pancreas,” 2018, *arXiv*. doi: 10.48550/ARXIV.1804.03999.

[79] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, “UNet++: A Nested U-Net Architecture for Medical Image Segmentation,” in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, vol. 11045, D. Stoyanov, Z. Taylor, G. Carneiro, T. Syeda-Mahmood, A. Martel, L. Maier-Hein, J. M. R. S. Tavares, A. Bradley, J. P. Papa, V. Belagiannis, J. C. Nascimento, Z. Lu, S. Conjeti, M. Moradi, H. Greenspan, and A. Madabhushi, Eds., in Lecture Notes in Computer Science, vol. 11045. , Cham: Springer International Publishing, 2018, pp. 3–11. doi: 10.1007/978-3-030-00889-5_1.

[80] G. Jocher and J. Qiu, *Ultralytics YOLO11*. (2024). [Online]. Available: https://github.com/ultralytics/ultralytics

[81] M. Mahesh, “The Essential Physics of Medical Imaging, Third Edition.,” *Medical Physics*, vol. 40, no. 7, p. 077301, July 2013, doi: 10.1118/1.4811156.

[82] E. Samei, D. J. Peck, and W. R. Hendee, *Hendee’s physics of medical imaging*, Fifth edition. Hoboken, NJ: John Wiley & Sons, Inc, 2019.

[83] K.-C. Pentapati and H. Siddiq, “Clinical applications of intraoral camera to increase patient compliance - current perspectives,” *CCIDE*, vol. Volume 11, pp. 267–278, Aug. 2019, doi: 10.2147/CCIDE.S192847.

[84] S. R. Cherry, J. A. Sorenson, and M. E. Phelps, *Physics in nuclear medicine*, 4th ed. Philadelphia: Elsevier/Saunders, 2012.
[85] E. Yazdani *et al.*, "Automated segmentation of lesions and organs at risk on [68Ga]Ga-PSMA-11 PET/CT images using self-supervised learning with Swin UNETR," *Cancer Imaging*, vol. 24, no. 1, p. 30, Feb. 2024, doi: 10.1186/s40644-024-00675-x.
[86] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why Should I Trust You?': Explaining the Predictions of Any Classifier," 2016, *arXiv*. doi: 10.48550/ARXIV.1602.04938.
[87] W. Samek, T. Wiegand, and K.-R. Müller, "Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models," 2017, *arXiv*. doi: 10.48550/ARXIV.1708.08296.
[88] A. Holzinger, C. Biemann, C. S. Pattichis, and D. B. Kell, "What do we need to build explainable AI systems for the medical domain?," 2017, *arXiv*. doi: 10.48550/ARXIV.1712.09923.
[89] European Parliament and Council, "Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 on laying down harmonised rules on artificial intelligence (Artificial Intelligence Act) and amending Regulations (EC) No 45/2001 and (EU) 2019/1896 and Directive (EU) 2022/2488," *Official Journal of the European Union*, no. L 2024/1689, pp. 1–352, 2024.
[90] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, D. Pedreschi, and F. Giannotti, "A Survey Of Methods For Explaining Black Box Models," 2018, *arXiv*. doi: 10.48550/ARXIV.1802.01933.
[91] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice: IEEE, Oct. 2017, pp. 618–626. doi: 10.1109/ICCV.2017.74.
[92] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps," 2013, *arXiv*. doi: 10.48550/ARXIV.1312.6034.
[93] P. Marsh and M. Martin, *Oral microbiology*, 5th ed. Edinburgh ; New York: Elsevier, 2009.
[94] C. Hannig, M. Hannig, and T. Attin, "Enzymes in the acquired enamel pellicle," *European J Oral Sciences*, vol. 113, no. 1, pp. 2–13, Feb. 2005, doi: 10.1111/j.1600-0722.2004.00180.x.
[95] R. Bos, "Physico-chemistry of initial microbial adhesive interactions – its mechanisms and methods for study," *FEMS Microbiology Reviews*, vol. 23, no. 2, pp. 179–229, Apr. 1999, doi: 10.1016/S0168-6445(99)00004-2.
[96] H. J. Busscher, W. Norde, and H. C. Van Der Mei, "Specific Molecular Recognition and Nonspecific Contributions to Bacterial Interaction Forces," *Appl Environ Microbiol*, vol. 74, no. 9, pp. 2559–2564, May 2008, doi: 10.1128/AEM.02839-07.
[97] R. 5. Science and Therapy Committee of the American Academy of Periodontology, "Treatment of Plaque-Induced Gingivitis, Chronic Periodontitis, and Other Clinical Conditions," *Journal of Periodontology*, vol. 72, no. 12, pp. 1790–1800, Dec. 2001, doi: 10.1902/jop.2001.72.12.1790.
[98] J. E. Botero, C. K. Rösing, A. Duque, A. Jaramillo, and A. Contreras, "Periodontal disease in children and adolescents of Latin America," *Periodontology 2000*, vol. 67, no. 1, pp. 34–57, Feb. 2015, doi: 10.1111/prd.12072.

[99] M. Çalışır and A. Akpınar, "Çocuklarda ve adolesanlarda periodontal hastalıklar," *Cumhuriyet Dent J*, vol. 16, no. 3, pp. 226–234, July 2013, doi: 10.7126/cdj.2013.1421.
[100] B. R. D. Gillings, "Recent developments in dental plaque disclosants," *Australian Dental Journal*, vol. 22, no. 4, pp. 260–266, Aug. 1977, doi: 10.1111/j.1834-7819.1977.tb04509.x.
[101] J. Silness and H. Löe, "Periodontal Disease in Pregnancy II. Correlation Between Oral Hygiene and Periodontal Condition," *Acta Odontologica Scandinavica*, vol. 22, no. 1, pp. 121–135, Jan. 1964, doi: 10.3109/00016356408993968.
[102] S. Turesky, N. D. Gilmore, and I. Glickman, "Reduced Plaque Formation by the Chloromethyl Analogue of Victamine C," *Journal of Periodontology*, vol. 41, no. 41, pp. 41–43, Sept. 1970, doi: 10.1902/jop.1970.41.41.41.
[103] B. Joseph, C. S. Prasanth, J. L. Jayanthi, J. Presanthila, and N. Subhash, "Detection and quantification of dental plaque based on laser-induced autofluorescence intensity ratio values," *J. Biomed. Opt*, vol. 20, no. 4, p. 048001, Apr. 2015, doi: 10.1117/1.JBO.20.4.048001.
[104] T. Yiğit, F. Karaaslan, U. Yiğit, and A. Dikilitaş, "Comparison of the plaque removal efficacy of chewable, electric, and manual toothbrushes: A randomized clinical trial," *Süleyman Demirel Üniversitesi Sağlık Bilimleri Dergisi*, vol. 14, no. 2, pp. 170–177, Aug. 2023, doi: 10.22312/sdusbed.1279913.
[105] K. H. Zou *et al.*, "Statistical validation of image segmentation quality based on a spatial overlap index1," *Academic Radiology*, vol. 11, no. 2, pp. 178–189, Feb. 2004, doi: 10.1016/S1076-6332(03)00671-8.
[106] P. Sudheera, V. R. Sajja, S. Deva Kumar, and N. G. Rao, "Detection of dental plaque using enhanced K-means and silhouette methods," in *2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT)*, Ramanathapuram, India: IEEE, May 2016, pp. 559–563. doi: 10.1109/ICACCCT.2016.7831702.
[107] S. Imangaliyev, M. H. van der Veen, C. M. C. Volgenant, B. J. F. Keijser, W. Crielaard, and E. Levin, "Deep Learning for Classification of Dental Plaque Images," in *Machine Learning, Optimization, and Big Data Proceedings of the Second International Workshop, MOD 2016*, P. M. Pardalos, P. Conca, G. Giuffrida, and G. Nicosia, Eds., Cham: Springer International Publishing, 2016, pp. 407–410. doi: 10.1007/978-3-319-51469-7_34.
[108] L. Liu, J. Xu, Y. Huan, Z. Zou, S.-C. Yeh, and L.-R. Zheng, "A Smart Dental Health-IoT Platform Based on Intelligent Hardware, Deep Learning, and Mobile Terminal," *IEEE J. Biomed. Health Inform.*, vol. 24, no. 3, pp. 898–906, Mar. 2020, doi: 10.1109/JBHI.2019.2919916.
[109] S. Li *et al.*, "Low-Shot Learning of Automatic Dental Plaque Segmentation Based on Local-to-Global Feature Fusion," in *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, Iowa City, IA, USA: IEEE, Apr. 2020, pp. 664–668. doi: 10.1109/ISBI45749.2020.9098741.
[110] W. You, A. Hao, S. Li, Y. Wang, and B. Xia, "Deep learning-based dental plaque detection on primary teeth: a comparison with clinical assessments," *BMC Oral Health*, vol. 20, no. 1, p. 141, Dec. 2020, doi: 10.1186/s12903-020-01114-6.
[111] S. Li *et al.*, "Automatic Dental Plaque Segmentation Based on Local-to-Global Features Fused Self-Attention Network," *IEEE J. Biomed. Health Inform.*, vol. 26, no. 5, pp. 2240–2251, May 2022, doi: 10.1109/JBHI.2022.3141773.

[112] B. Yüksel, N. Özveren, and Ç. Yeşil, "Evaluation of Dental Plaque Area with Artificial Intelligence Model," *Nigerian Journal of Clinical Practice*, vol. 27, no. 6, pp. 759–765, June 2024, doi: 10.21203/rs.3.rs-3423771/v1.
[113] T. Nantakeeratipat, N. Apisaksirikul, B. Boonrojsaree, S. Boonkijkullatat, and A. Simaphichet, "Automated machine learning for image-based detection of dental plaque on permanent teeth," *Front. Dent. Med*, vol. 5, p. 1507705, Nov. 2024, doi: 10.3389/fdmed.2024.1507705.
[114] X. Chen, Y. Shen, J.-S. Jeong, H. Perinpanayagam, K.-Y. Kum, and Y. Gu, "DeepPlaq: Dental plaque indexing based on deep neural networks," *Clin Oral Invest*, vol. 28, no. 10, p. 534, Sept. 2024, doi: 10.1007/s00784-024-05921-x.
[115] W. Song, X. Wang, Y. Guo, S. Li, B. Xia, and A. Hao, "CenterFormer: A Novel Cluster Center Enhanced Transformer for Unconstrained Dental Plaque Segmentation," *IEEE Trans. Multimedia*, vol. 26, pp. 10965–10978, 2024, doi: 10.1109/TMM.2024.3428349.
[116] A. Ramírez-Pedraza *et al.*, "Deep Learning in Oral Hygiene: Automated Dental Plaque Detection via YOLO Frameworks and Quantification Using the O'Leary Index," *Diagnostics*, vol. 15, no. 2, p. 231, Jan. 2025, doi: 10.3390/diagnostics15020231.
[117] A. Dutta, A. Gupta, and A. Zissermann, *VGG Image Annotator (VIA)*. (2016). University of Oxford, Oxford, UK. [Online]. Available: http://www.robots.ox.ac.uk/vgg/software/via/
[118] Q. T. Ostrom *et al.*, "The epidemiology of glioma in adults: a 'state of the science' review," *Neuro-Oncology*, vol. 16, no. 7, pp. 896–913, July 2014, doi: 10.1093/neuonc/nou087.
[119] D. N. Louis *et al.*, "The 2016 World Health Organization Classification of Tumors of the Central Nervous System: a summary," *Acta Neuropathol*, vol. 131, no. 6, pp. 803–820, June 2016, doi: 10.1007/s00401-016-1545-1.
[120] N. Sanai and M. S. Berger, "Glioma Extent Of Resection and Its Impact on Patient Outcome," *Neurosurgery*, vol. 62, no. 4, pp. 753–766, Apr. 2008, doi: 10.1227/01.neu.0000318159.21731.cf.
[121] B. R. J. Van Dijken, P. J. Van Laar, G. A. Holtman, and A. Van Der Hoorn, "Diagnostic accuracy of magnetic resonance imaging techniques for treatment response evaluation in patients with high-grade glioma, a systematic review and meta-analysis," *Eur Radiol*, vol. 27, no. 10, pp. 4129–4144, Oct. 2017, doi: 10.1007/s00330-017-4789-9.
[122] S. Lefkovits, L. Lefkovits, and L. Szilágyi, "HGG and LGG Brain Tumor Segmentation in Multi-Modal MRI Using Pretrained Convolutional Neural Networks of Amazon Sagemaker," *Applied Sciences*, vol. 12, no. 7, p. 3620, Apr. 2022, doi: 10.3390/app12073620.
[123] A. BabaAhmadi and Z. FallahPour, "Efficient Deep Learning Algorithms for Lower Grade Gliomas Cancer MRI Image Segmentation: A Case Stud," *AJMC*, no. Online First, Oct. 2023, doi: 10.22060/ajmc.2023.22562.1172.
[124] R. Dattangire, D. Biradar, and A. Joon, "AI-Enhanced U-Net for Accurate Low-Grade Glioma Segmentation in Brain MRI: Transforming Healthcare Imaging," in *2024 Third International Conference on Electrical, Electronics, Information and Communication Technologies (ICEEICT)*, Trichirappalli, India: IEEE, July 2024, pp. 1–6. doi: 10.1109/ICEEICT61591.2024.10718440.
[125] N. Pedano *et al.*, "The Cancer Genome Atlas Low Grade Glioma Collection (TCGA-LGG)." The Cancer Imaging Archive, 2016. doi: 10.7937/K9/TCIA.2016.L4LTD3TK.

[126] M. Buda, A. Saha, and M. A. Mazurowski, "Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm," *Computers in Biology and Medicine*, vol. 109, pp. 218–225, June 2019, doi: 10.1016/j.compbiomed.2019.05.002.
[127] E. J. Schafer *et al.*, "Recent Patterns and Trends in Global Prostate Cancer Incidence and Mortality: An Update," *European Urology*, vol. 87, no. 3, pp. 302–313, Mar. 2025, doi: 10.1016/j.eururo.2024.11.013.
[128] National Cancer Institute, "Cancer Stat Facts: Prostate Cancer," Surveillance, Epidemiology, and End Results Program. Accessed: May 02, 2025. [Online]. Available: https://seer.cancer.gov/statfacts/html/prost.html
[129] American Cancer Society, "Key Statistics for Prostate Cancer," American Cancer Society. Accessed: May 02, 2025. [Online]. Available: https://www.cancer.org/cancer/types/prostate-cancer/about/key-statistics.html
[130] J. Trotter *et al.*, "Positron Emission Tomography (PET)/Computed Tomography (CT) Imaging in Radiation Therapy Treatment Planning: A Review of PET Imaging Tracers and Methods to Incorporate PET/CT," *Advances in Radiation Oncology*, vol. 8, no. 5, p. 101212, Sept. 2023, doi: 10.1016/j.adro.2023.101212.
[131] R. L. J. Qiu, C.-W. Chang, and X. Yang, "Deep learning-assisted lesion segmentation in PET/CT imaging: A feasibility study for salvage radiation therapy in prostate cancer," *Oncoscience*, vol. 11, pp. 49–50, May 2024, doi: 10.18632/oncoscience.603.
[132] S. Jeong, W. Cheon, S. Kim, W. Park, and Y. Han, "Deep-learning-based segmentation using individual patient data on prostate cancer radiation therapy," *PLoS ONE*, vol. 19, no. 7, p. e0308181, July 2024, doi: 10.1371/journal.pone.0308181.
[133] G. Litjens *et al.*, "Evaluation of prostate segmentation algorithms for MRI: The PROMISE12 challenge," *Medical Image Analysis*, vol. 18, no. 2, pp. 359–373, Feb. 2014, doi: 10.1016/j.media.2013.12.002.
[134] S. Bhandary *et al.*, "Investigation and benchmarking of U-Nets on prostate segmentation tasks," *Computerized Medical Imaging and Graphics*, vol. 107, p. 102241, July 2023, doi: 10.1016/j.compmedimag.2023.102241.
[135] Z. Tian, L. Liu, Z. Zhang, and B. Fei, "PSNet: prostate segmentation on MRI based on a convolutional neural network," *J. Med. Imag.*, vol. 5, no. 02, p. 1, Jan. 2018, doi: 10.1117/1.JMI.5.2.021208.
[136] Y. Zhang, J. Wu, W. Chen, Y. Chen, and X. Tang, "Prostate Segmentation Using Z-Net," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, Venice, Italy: IEEE, Apr. 2019, pp. 11–14. doi: 10.1109/ISBI.2019.8759554.
[137] Q. Zhu, B. Du, B. Turkbey, P. L. . Choyke, and P. Yan, "Deeply-Supervised CNN for Prostate Segmentation," 2017, *arXiv*. doi: 10.48550/ARXIV.1703.07523.
[138] Y. Lei *et al.*, "Ultrasound prostate segmentation based on multidirectional deeply supervised V-Net," *Medical Physics*, vol. 46, no. 7, pp. 3194–3206, July 2019, doi: 10.1002/mp.13577.
[139] H. Jiang *et al.*, "MicroSegNet: A deep learning approach for prostate segmentation on micro-ultrasound images," *Computerized Medical Imaging and Graphics*, vol. 112, p. 102326, Mar. 2024, doi: 10.1016/j.compmedimag.2024.102326.
[140] Y. Lei *et al.*, "CT prostate segmentation based on synthetic MRI-aided deep attention fully convolution network," *Medical Physics*, vol. 47, no. 2, pp. 530–540, Feb. 2020, doi: 10.1002/mp.13933.

[141] C. Li *et al.*, “FocalUNETR: A Focal Transformer for Boundary-Aware Prostate Segmentation Using CT Images,” in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2023*, vol. 14222, H. Greenspan, A. Madabhushi, P. Mousavi, S. Salcudean, J. Duncan, T. Syeda-Mahmood, and R. Taylor, Eds., in Lecture Notes in Computer Science, vol. 14222. , Cham: Springer Nature Switzerland, 2023, pp. 592–602. doi: 10.1007/978-3-031-43898-1_57.

[142] K. He *et al.*, “HF-UNet: Learning Hierarchically Inter-Task Relevance in Multi-Task U-Net for Accurate Prostate Segmentation in CT Images,” *IEEE Trans. Med. Imaging*, vol. 40, no. 8, pp. 2118–2128, Aug. 2021, doi: 10.1109/TMI.2021.3072956.

[143] J. Zhong, T. Liu, Y. Piao, W. Sun, and H. Lu, “ProSegDiff: Prostate Segmentation Diffusion Network Based on Adaptive Adjustment of Injection Features,” *IEEE Signal Process. Lett.*, vol. 32, pp. 1236–1240, 2025, doi: 10.1109/LSP.2025.3548422.

[144] L. Ma, Q. Fan, Z. Tian, L. Liu, and B. Fei, “A novel Residual and Gated Network for prostate segmentation on MR images,” *Biomedical Signal Processing and Control*, vol. 87, p. 105508, Jan. 2024, doi: 10.1016/j.bspc.2023.105508.

[145] E. Kot, T. Les, Z. Krawczyk-Borysiak, A. Vykhodtsev, and K. Siwek, “Semantic Segmentation of the Prostate Based on Onefold and Joint Multimodal Medical Images Using YOLOv4 and U-Net,” *Applied Sciences*, vol. 14, no. 21, p. 9814, Oct. 2024, doi: 10.3390/app14219814.

[146] Y. Song *et al.*, “Multidisciplinary Consensus Prostate Contours on Magnetic Resonance Imaging: Educational Atlas and Reference Standard for Artificial Intelligence Benchmarking,” *International Journal of Radiation Oncology*Biology*Physics*, vol. 123, no. 1, pp. 183–194, Mar. 2025, doi: 10.1016/j.ijrobp.2025.03.024.

CURRICULUM VITAE

2009 – 2013	B.Sc., Computer Education and Instructional Technologies, Inonu University, Malatya, TÜRKİYE
2017 – 2025	PhD, Electrical and Computer Engineering, Abdullah Gul University, Kayseri, TÜRKİYE

SELECTED PUBLICATIONS AND PRESENTATIONS

J1) B. Ç. Tez, Y. Güzel, B. B. Kızıltan Eliaçık, Z. Aydın, Deep-Learning-Based AI-Model for Predicting Dental Plaque in the Young Permanent Teeth of Children Aged 8–13 Years published in Children (Apr. 2025).

J2) Y. Güzel, Z. Aydın, A Comparative Study of UNet Variants for Low-Grade Glioma Segmentation in Magnetic Resonance Imaging published in Journal of Inonu University Health Services Vocational School (May 2025).

C1) N. Ş. Ersoz, Y. Guzel, B. Bakir-Gungor, In-silico Identification of Papillary Thyroid Carcinoma Molecular Mechanisms in Proc. Of 27th Signal Processing and Communications Applications Conference (Apr. 2019).