République Tunisienne
Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique
Université de Tunis
Ecole Nationale Supérieure
d'Ingénieurs de Tunis

Code : GPE-RP-02-00
Date de création : 16-06-2023

Département Génie Informatique

Ref: PFA2-2024- 10

End of Year Project Report

of

Second year in Computer Engineering

Presented and publicly defended on 09/05/2024

By

Aziz BEN AMOR
Hamdi BACCAR
Mohamed Nadhir NAJJAR

Multimodal Segmentation of Head and Neck Cancer Organs at Risk, using Deep Learning

Composition of the jury

Mrs.	**Ines ELOUEDI**	**(Assistant Professor)**	**President**
Mrs.	**Afef Kacem ECHI**	**(Associate Professor)**	**Supervisor**

Academic Year: 2023-2024

Dedications

To our dear parents

For all their sacrifices, unwavering love, boundless tenderness, steadfast support, and heartfelt prayers that have guided me through our educational journey..

To our dear brothers,

For your constant encouragement and unwavering moral support, helping me overcome challenges and reach for the stars.

To our dear friends,

For your enduring friendship, support, and motivation that have been our pillars of strength, inspiring me to push our boundaries and achieve our goals.

To all our family,

For standing by our side, offering endless support, and believing in our dreams throughout our university career. Your collective presence has been our greatest source of strength and inspiration.

Acknowledgments

We would like to warmly thank ***Pr. Afef Kacem Echi*** who allowed us to benefit from their guidance. The advice they provided, and the confidence they showed, were crucial to the realization of our project.

We would also like to sincerely thank ***Dr. Takwa Ben Aïcha Gader*** for the time she devoted to answering all our questions during this period.

Contents

List of Tables

List of Figures

General Introduction

Medical Context and Objective

Cancer in the head and neck (HaN) region poses a serious health threat, with its risks of rapid spread and potential complications that can alter vital functions such as breathing, swallowing, and speech, primarily treated by radiotherapy. This method aims to deliver a precise dose of radiation to cancer cells while preserving surrounding healthy organs, known as organs at risk (OAR). This approach is crucial for maximizing treatment effectiveness while minimizing side effects on neighboring tissues. Consequently, radiotherapy is widely used in the fight against head and neck cancer, often yielding significant results.

- To ensure optimal distribution of the radiation dose, it is essential to accurately segment the target volumes and OARs in three dimensions, primarily using computed tomography (CT) images. However, some OARs in the HaN region are not well visible in CT but are more clearly visible in magnetic resonance imaging (MRI) images. Although there have been attempts to segment OARs from MRI images, the impact of combined analysis of CT and MRI images on OAR segmentation in HaN has not yet been evaluated [14].

- In this context, the challenge of multimodal segmentation of organs at risk in the head and neck aims to promote the development of fully automated techniques for segmenting OARs in the HNC region, leveraging information from multiple imaging modalities to improve the accuracy of segmentation results. Currently underway, this challenge, called HaN-Seg or "Head and Neck-Segmentation," involves automatically segmenting 30 OARs in the HaN region from CT and MRI images, and our goal is to achieve this multimodal segmentation and thereby identify the OARs.

Proposed Approach

This work presents a multi-modal segmentation approach for Head and Neck (HN) OARs, aiming to leverage the complementary information from both CT and MRI images for improved segmentation accuracy. Accurate OAR segmentation is crucial for treatment planning in HN cancer, ultimately leading to better patient outcomes. We employ deep learning techniques, specifically utilizing U-Net architectures, to automate the segmentation process, significantly reducing the time and effort required for OAR identification. To achieve this, we explore two separate paths:

1. **Basic Approach**: We initially utilize a 2D U-Net architecture that operates on individual slices extracted from the 3D images. This approach provides a simpler initial implementation for understanding the feasibility of the multi-modal approach.

2. **3D-UNet for Enhanced Accuracy**: We subsequently employ a 3D-UNet architecture that operates directly on 3D patches extracted from the original volumes. This approach offers several advantages for multi-modal segmentation. Its ability to handle volumetric data allows for in-depth analysis of three-dimensional structures, while its encoding and decoding mechanism captures contextual information at different spatial scales, enhancing segmentation accuracy.

By using both 2D and 3D U-Net architectures, we achieve a balance between simplicity and accuracy, facilitating the integration of multiple imaging modalities like MRI and CT. Furthermore, these deep learning approaches significantly reduce dependence on manual supervision, accelerating the segmentation process while adapting better to individual patient anatomical variations through training on diverse datasets.

Adopted Methodology

We chose the CRISP-DM methodology, a popular approach for managing projects in data mining and data analysis. Figure 1 below explains its stages, which we have tried to follow.

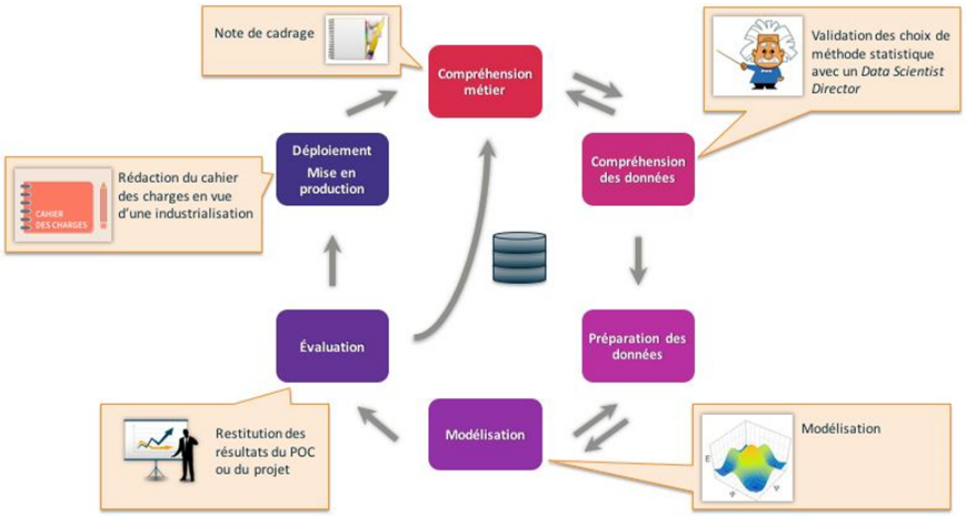


Figure 1: Stages of the CRISP-DM Methodology [6].

- **Business Understanding:** Identifying business objectives, defining success criteria, establishing initial problem understanding.
- **Data Understanding:** Collecting available data, exploring data structure and quality, identifying potential issues.
- **Data Preparation:** Cleaning data by removing missing or outlier values, transforming data for analysis, and integrating different data sources.
- **Modeling:** Selecting appropriate analysis techniques, building prediction or analysis models, evaluating models to find the most performant.
- **Evaluation**: Evaluating models on independent data, verifying achievement of business objectives, revisiting previous steps if necessary.
- **Deployment:** Integrating results into business processes, operationalizing solutions, and continuous monitoring to ensure relevance.

We opted for this methodology due to its numerous advantages, including:

- **Organized Structure:** CRISP-DM provides a clear and structured framework for managing a data analysis project, facilitating planning and execution.
- **Flexibility:** Although structured, CRISP-DM also allows flexibility to adapt to the specific needs of each project.
- **Iterative Approach:** The methodology encourages an iterative approach, meaning steps can be revisited and refined as the project progresses and new information becomes available.

- **Business Orientation:** CRISP-DM emphasizes understanding business objectives from the outset, ensuring analysis results are relevant and useful for the business.
- **Continuous Evaluation:** By regularly evaluating progress and results, CRISP-DM enables quick identification of potential issues and adjustments to the strategy accordingly.

In summary, the CRISP-DM methodology offers a methodical and practical approach to managing data analysis projects, focusing on understanding business needs, data quality, and continuous iteration to achieve meaningful results.

Report Plan

Our report will follow a structure with an introduction, three chapters, and a general conclusion. Firstly, in the first chapter "state of the art," we will examine existing research in deep learning-based medical imaging analysis. Next, we move to the second chapter on the "proposed system," describing the different stages from data acquisition, preprocessing, modeling, evaluating the selected model, and its deployment. The final chapter, We conclude the report with a general conclusion and perspectives for improvement.

Chapter 1

State of the Art

1.1 Introduction

In the field of segmentation of organs at risk HaN, the use of Deep Learning and medical image processing has progressed considerably. This chapter focuses on the field of radiotherapy in this subject by highlighting the exploration of these medical advances and focusing on the different automated segmentation approaches. We will also review existing work, identifying the strengths and weaknesses of each method, to direct our research towards more efficient and precise solutions.

1.2 Radiotherapy in the problem of head and neck cancer

1.2.1 An overview of the disease

Head and neck cancer encompasses various types of cancers that develop in the upper aerodigestive tract, such as the lips, tongue, mouth, throat, and larynx, as well as in the salivary glands, nasopharynx (the part of the nose connected to the upper part of the throat), or the sinuses and the nasal cavity (see Figure 1.1). Most cancers affecting this area are squamous cell carcinomas. Other rare types of cancer, such as those developing in the salivary glands, nasopharynx, paranasal sinuses, and nasal cavity, as well as cancers with a histological type different from squamous cell carcinomas, are subject to specific recommendations that are not covered in this study.

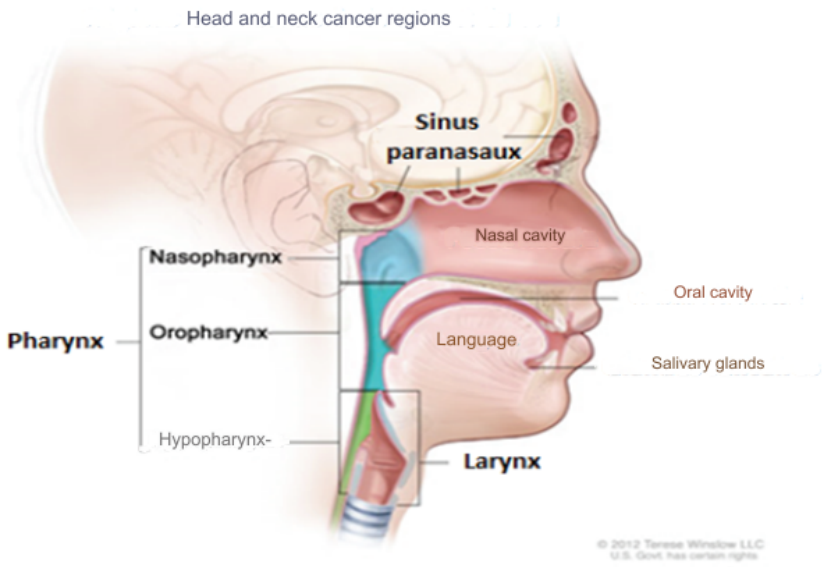


Figure 1.1: Overview of the different regions of the HaN cancer [7].

Head and neck cancers represent 4% of all cancers worldwide. The oral cavity is the most commonly affected location, constituting 41% of all head and neck cancers, followed by pharyngeal and larynx cancers, which account for 22% and 24% of these cancers, respectively.

1.2.2 Radiotherapy and Multimodal Segmentation

Radiotherapy is a treatment method used for head and neck cancers to target and destroy cancer cells locally using high-energy radiation generated by a specific radiotherapy machine. The quality of delineation of organs at risk is an essential factor affecting the effectiveness and side effects of radiotherapy. Clinically, radiologists must spend several hours manually delineating organs. This is usually very time-consuming and requires a high level of professionalism on the part of the radiologist. In some underdeveloped regions, qualified radiologists are very scarce resources. Thus, the design of an efficient and robust OAR segmentation algorithm can effectively alleviate this dilemma. For the treatment of head and neck cancer (HaN) with radiotherapy (RT), segmentation of organs at risk (OAR) is a crucial step in planning. However, current approaches mainly focus on using either computed tomography (CT) or magnetic resonance images (MRI), without fully exploring multimodal segmentation [14].

In recent years, advances in RT, alongside improvements in less invasive organ-sparing surgery as well as more intensive multimodal treatments, have contributed to the preservation of function and reduced mortality from head cancer and neck. Among these advances, considerable progress has been made in the field of artificial intelligence (AI), in particular in the field of deep learning (DL), which has a wide range of applications in radiotherapeutic oncology. For example, segmentation, the

process of partitioning a medical image into multiple anatomical structures, is one of the key steps in RT planning because it provides accurate three-dimensional spatial descriptions of target volumes as well as the OARs needed for an optimal calculation of the radiation dose distribution.

Although segmentation is mainly performed on computed tomography (CT) images because they contain electron density information used for calculating radiation beam energy absorption, one of the main limitations of the images CT is insufficient contrast for soft tissue. Therefore, the integration of complementary imaging modalities, such as magnetic resonance (MR), is highly recommended for the HaN region to improve the segmentation of multiple soft tissue OARs. Although attempts have been made for the segmentation of OARs from MR images, until now there has been no objective assessment of the impact of combined analysis of CT and MR images on the segmentation of OARs in the HaN region. Most existing approaches focus on segmentation from either CT image modality or MR image modality, while multimodal segmentation has not yet been fully explored, probably because CT data and MR of the same patients are not always available or, when they are, they are not systematically included in the RT planning pipeline.

1.3 Deep Learning-based Multimodal Segmentation Methods

A literature review by Zhang et al. [20] divides deep learning-based (DL) multimodal segmentation methods into three groups of fusion strategies: early fusion, late fusion, and hybrid fusion (also known as layer fusion). The first two groups of methods are the most commonly applied; early fusion involves a simple concatenation of modalities along the channel dimension before feeding them into the deep neural network. Additionally, concatenating feature maps (FMs) from separate modality encoders can also be considered as early fusion (see Figure 1.2(a)). In contrast, late fusion uses separate branches for each input modality and then merges the output features either by simple concatenation or by weighting the contributions of separate branches at the decision level (see Figure 1.2(b)). For instance, authors in [2] proposed an attention mechanism to merge FMs from two separate U-Nets that accepted contrast-enhanced arterial and venous phase CT images (see Figure 1.3). The third group, hybrid fusion, aims to combine the strengths of early and late fusion by using two or more separate encoders (i.e., one for each modality) and a single decoder, where features from different resolution levels of the encoder are fused and fed into the decoder that produces the final segmentation at full resolution (see Figure 1.2(c)).

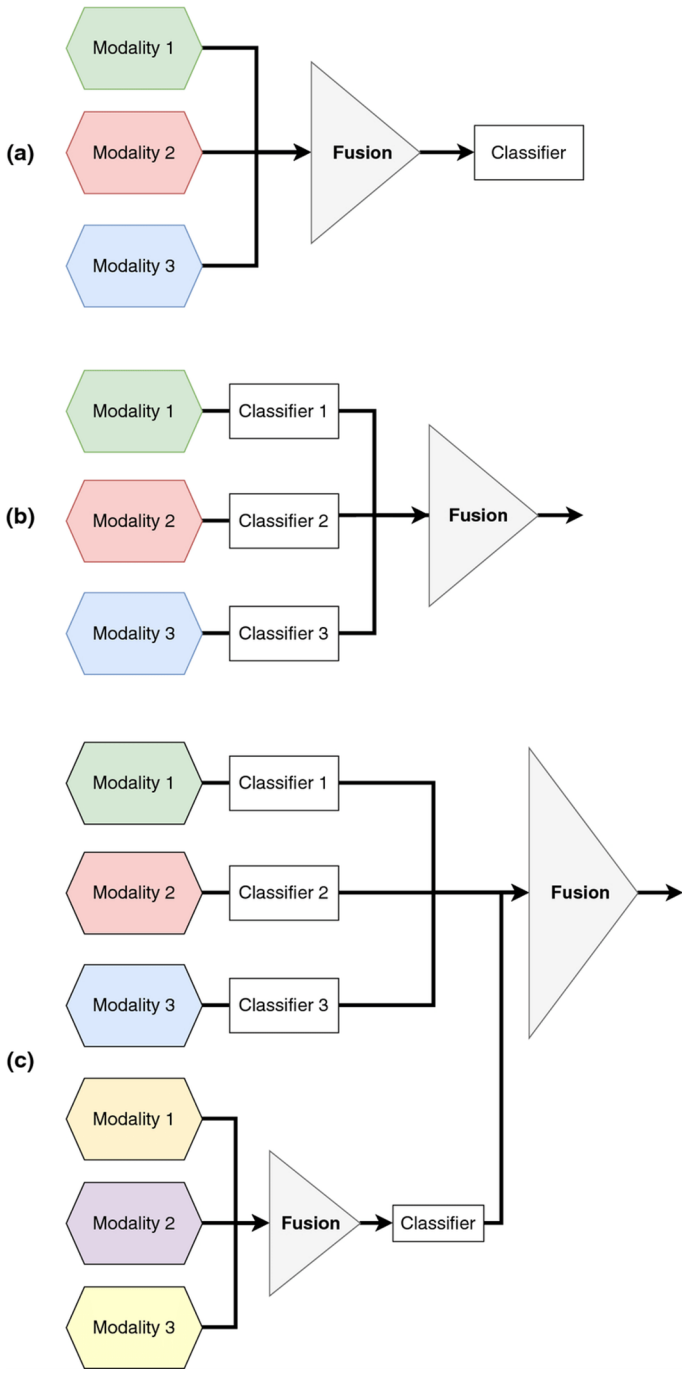


Figure 1.2: Approaches to Multimodal Segmentation based on Deep Learning

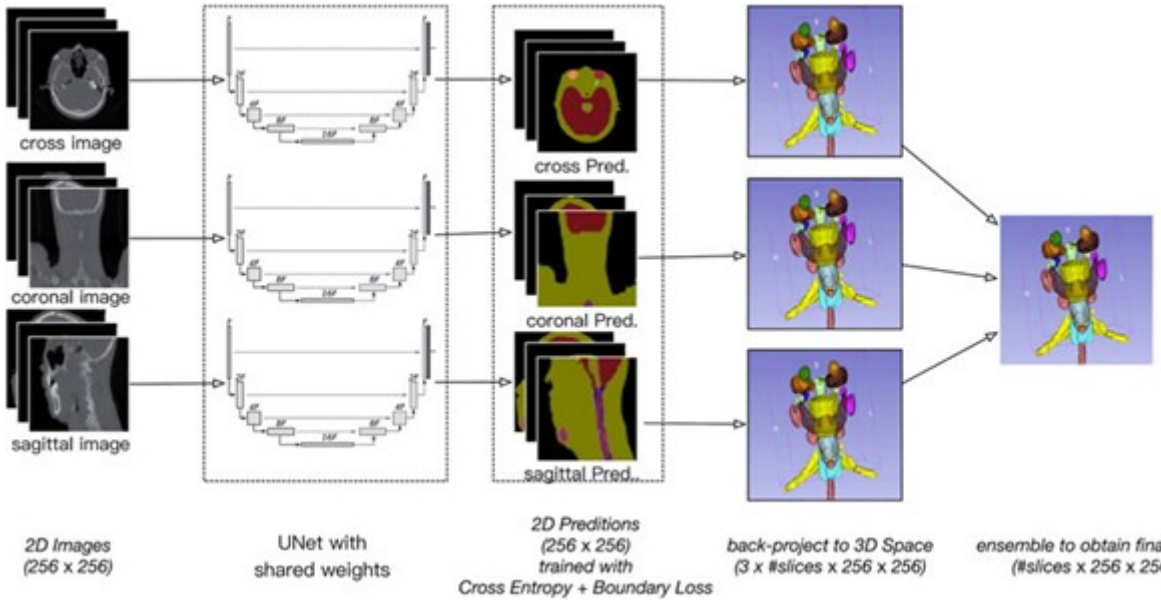


Figure 1.3: Late modality fusion Segmentation.

Another crucial consideration is the absence of certain modalities. This means that the multimodal model must still perform well even if only one type of input is present. However, determining the best fusion approach remains a topic requiring further investigation. Most methods generally use either early fusion or late fusion. However, layer fusion has been recognized as more effective. By favoring dense connections between layers, it can better exploit complex and complementary information, thus improving the training process. A notable example is HyperDenseNet, a 3D network developed by Dolz et al. [5]. It uses dense connections between two pathways and outperforms other fusion methods. However, different studies reveal that the most effective fusion method varies depending on the problem. For instance, Yan et al. [19] found that late fusion works better for detecting diabetic retinopathy over time. In this work, we propose a system for the segmentation of normal anatomical regions of the head and neck (HaN OAR) where CT and MRI image modalities from the same patient are often available. Therefore, our main focus is on the paired multimodal segmentation problem, including the missing modality scenario.

Indeed, when segmenting organs at risk (OARs) in the head and neck region for radiotherapy planning, a multimodal segmentation model that can use information from both CT and MRI images of the same patient could be beneficial compared to separate unimodal models. Firstly, such a model would rely on CT images for bony structures and MRI images for soft tissues, improving the overall segmentation quality by leveraging complementary information from both modalities. Secondly, a multimodal model would facilitate inter-modality learning by extracting knowledge from one modality and applying that knowledge to the other, which could enhance segmentation accuracy. Finally, from a deep learning infrastructure maintenance perspective, it is easier to maintain a single model

capable of handling both modalities than two separate models for each modality. However, clinical practice differs significantly from theory, meaning several considerations must be taken into account. Firstly, while obtaining MRI images is recommended, it is not always feasible due to time constraints, scanner occupancy, and financial aspects. Therefore, automated multimodal OAR segmentation is needed to handle the missing modality scenario and provide segmentation quality similar to a unimodal system. Secondly, because CT and MRI images are not acquired simultaneously and with the same acquisition parameters, there is inherent misalignment between the two modalities. This can be mitigated with image registration but not completely, mainly due to different patient positions that particularly affect soft tissue deformation, and various modality-specific artifacts.

1.4 CNN for Medical Image Segmentation

Recently, convolutional neural networks have significantly advanced the field of medical image analysis due to their ability to learn more representative features from data. CNNs demonstrate state-of-the-art performance in many challenging tasks, such as image classification, segmentation, detection, registration, super-resolution, etc. Long and colleagues' study published in 2015 [12] first proposed a fully convolutional network (FCN), which uses convolutions with filters of size 1x1 to replace the fully connected layer, and allows prediction of multiple pixels at the same time. Ronneberger and his colleagues in 2015 [12] then constructed a "U"-shaped network (called U-Net) with a contracting path and a symmetrical expansion path. Skip connections are also used to propagate features from early layers to later layers. A large number of works based on variants of FCN and U-Net are applied in the field of 2D medical image segmentation. For 3D images like CT or MRI, 2D CNNs can be used in a slice-by-slice manner, however, the contextual information encoded in the volumetric data is ignored. Some 2.5D methods have attempted to incorporate 3D spatial information using three orthogonal slices or adjacent slices. But their representation capabilities are still limited by 2D convolution kernels. To overcome this weakness, algorithms based on 3D CNNs are proposed. For example,the 3D version of U-Net is proposed by Cicek and colleagues in 2016 , Milletari and colleagues in 2016 also proposed V-Net, which introduces the residual connection between building blocks to mitigate the problem of the disappearance of gradients. Several 3D networks have also been proposed for different applications, such as Merkow in 2016; Dou in 2016; and Kamnitsas in 2017. Although 3D CNN-based methods can better exploit spatial context to learn better feature representations, the sample imbalance problem is amplified in 3D tasks because training errors are mainly dominated by voxels belonging to large organs. Ronneberger and colleagues in 2015 proposed using the weighted cross-entropy loss function, while Milletari and colleagues in 2016 proposed the Dice

coefficient, they can only alleviate the challenge of imbalanced data but are far from solving.

1.4.1 FocusNetV2: a proposal for OAR segmentation for CT images

The general framework of this approach is illustrated in the figure below:

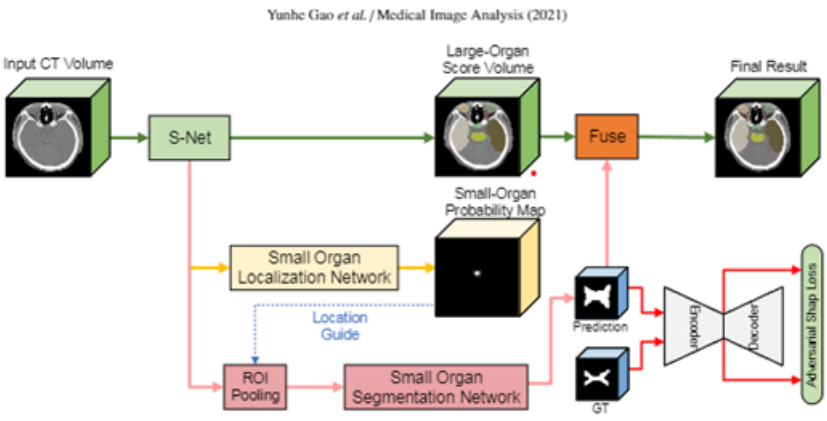


Figure 1.4: Overview of the FocusNetV2 architecture.

In particular, this network has two main components: the segmentation network and the adversarial autoencoder (AAE) for organ shape constraint. The segmentation network resolves extremely imbalanced data in a two-stage framework, which consists of three parts: the main segmentation network (S-Net), the small organ localization branch (SOL-Net), and the branch of segmentation of small organs (SOS-Net). It mimics the process of delineating medical images by doctors. The segmentation network first segments all organs with the main segmentation network (S-Net) and locates the central locations of a series of predefined small organs with the small organ localization branch (SOL-Net). Multi-scale features and high-resolution images are grouped into regions of interest (ROI) by the Small Organ Segmentation Branch (SOS-Net) to generate small organ label maps. After adding additional shape constraints with the proposed adversarial autoencoder (AAE), it encourages the segmentation network's predictions to be consistent with the prior shapes of different organs, even if there are no boundaries clear in CT images. To our knowledge, this is the first segmentation method that takes advantage of both autoencoder and adversarial learning for shape regularization.

1.4.2 Challenges of the FocusNetV2 architecture

Despite the robust performance demonstrated by this method, and its potential to accelerate the radiotherapy planning process in clinical settings, it remains far from perfect. Currently, shape constraints only apply to small organs, due to the sampling imbalance encountered when training the shape autoencoder. Indeed, direct training of an autoencoder for all organs would favor large organs, to the detriment of small ones, thus making it difficult to regulate the segmentation results for the latter. Considering that the S-Net offers good performance for large organs, while a significant variance is observed between patients for small organs, which therefore restricts the application of the shape constraint to small organs with fuzzy boundaries, to encourage the network to generate predictions consistent with previous forms. Additionally, this current training process, consisting of multiple steps, complicates the learning process, which is an important consideration from a performance perspective. Additionally, the network did not achieve optimal performance during end-to-end training.

1.4.3 A Two-Stage Segmentation Framework Based on 3D U-Net

A throwback to the Two-Stage Segmentation Framework study back in 2017 [18], Yueyue Wang and al. performed a 3DUnet segmentation for nine anatomical structures OAR using two 3D UNets while addressing memory limitations and the challenges associated with OAR segmentation tasks. The framework, as shown in Figure 1.5 decomposes the segmentation process into two simpler subtasks: Locating a bounding box containing the target OAR using a dedicated 3D U-Net (LocNet). Segmenting the target OAR within the localized bounding box using another 3D U-Net (SegNet).

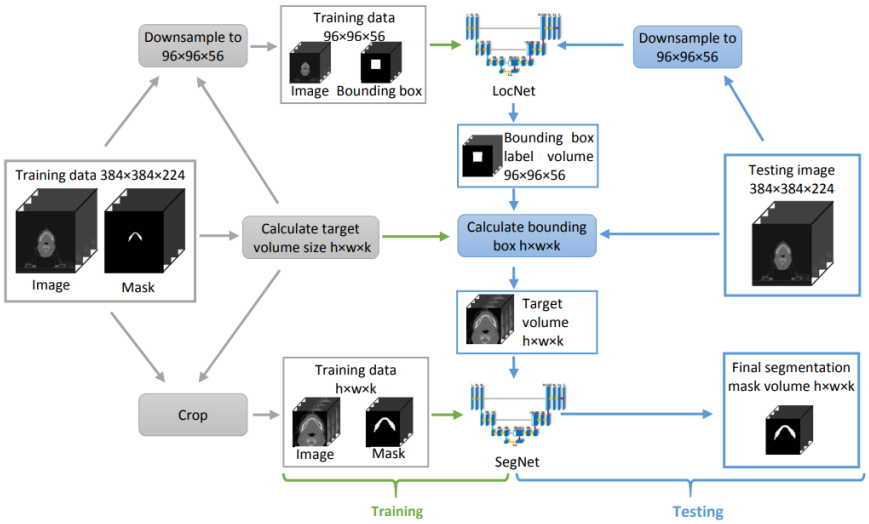


Figure 1.5: Flowchart of the Framework.

1.4.4 Advancements in OAR Segmentation

Several methods have been explored for OAR segmentation in HaN CT images, but their performance often lagged behind other medical image segmentation tasks due to inherent difficulties:

Variability in OAR Characteristics: OARs exhibit significant variability in shape, size, and contrast across different structures, posing challenges for segmentation algorithms. While deep learning networks have emerged as powerful tools in medical image segmentation, their application in OAR segmentation was hampered by memory limitations when handling large, high-resolution images. Previous attempts to address these challenges included:

- **Downsampling:** Reducing image resolution to a manageable size, but sacrificing the ability to distinguish fine details of small structures.
- **Sliding-Window Strategy:** Segmenting the image block-by-block, which is computationally expensive and requires careful parameter tuning to achieve optimal results. Multi-Atlas Based Segmentation: Roughly locating the region of interest with an atlas before segmentation, but potentially limiting accuracy due to potential atlas registration errors. The proposed two-stage framework surpasses these limitations by:
- **Effective Task Decomposition:** Breaking down the segmentation into two specialized tasks simplifies the learning process for each network, leading to better performance.
- **Segmentation within Bounding Box:** Focusing on a smaller, relevant region around the localized OAR allows SegNet to excel in accurate segmentation. These strategies used in state-of-the-art studies have led to significant improvements in segmentation accuracy. We present some study results using 2.7 the Dice score coefficient (%) and 95HD score (mm) metrics. (see Table 2.2)

Table 1.1: Studies Results.

Studies	**[1] LocNet|SegNet Framework (2017)**		**[1] FocusNetv2(2019)**	
	Dice score coefficient (%)	**95HD score (mm)**	**Dice score coefficient (%)**	**95HD score (mm)**
Brain Stem	87.5±2.2	2.01±0.33	88.2±2.5	2.32±0.70
Mandible	93.0±1.9	1.26±0.50	94.7±1.1	2.25±0.85
Optic Chiasm	45.1±17.2	2.83±1.42	71.3±17.0	1.08±0.45
Optic Nerve L	73.7±7.6	2.53±2.34	79.0±7.5	1.92±0.80
Optic Nerve R	73.6±8.8	2.13±2.45	81.7±7.3	2.17±0.74
Parotid L	86.4±2.6	2.41±0.54	89.8±1.6	1.81±0.43
parotid R	84.8±7.0	2.93±1.48	88.1±4.2	2.43±2.00
SMG L	75.8±14.7	2.86±1.60	84.0±4.6	2.84±1.20
SMG R	73.3±9.7	3.44±1.55	83.8±4.1	2.74±1.25

This table summarizes the segmentation results for various OARs using the LocNet|SegNet Framework and FocusNetv2 methods, showcasing the effectiveness of these approaches in terms of Dice score coefficient and 95HD score.

1.5 Conclusion

In this chapter, we have discussed the field of radiotherapy and the various multimodal approaches used for segmenting head and neck regions. We have also reviewed the existing research and analyzed the strengths and weaknesses of each method to guide our research towards developing more efficient and precise solutions. In the upcoming chapter, we will present our system which is based on the UNet architecture.

Chapter 2

Proposed System

2.1 Introduction

In this chapter, we will explain our proposed multimodal segmentation system that uses CT and MRI images to identify the 30 organs at risk in the head and neck region. To achieve this, we followed the CRISP-DM methodology steps, which we will explain below. For the modeling phase, we utilized UNet architecture models.

2.2 System Overview

As shown in Figure 2.1, our system proceeds by early modality fusion by combining data from computed tomography (CT) and magnetic resonance imaging (MRI), before feeding it into the head and neck region segmentation network. The use of early fusion typically involves combining the complementary information provided by CT and MRI modalities to improve the accuracy and robustness of the segmentation process. CT scans are excellent for visualizing bones and dense tissues, while MRI provides superior soft tissue contrast.

For segmentation, we used the U-Net architecture which is a popular choice for medical image segmentation tasks due to its ability to effectively capture spatial information through a combination of downsampling and upsampling paths. When employing U-Net for head and neck segmentation, the network is trained on multi-modal data, where CT and MRI images are concatenated or merged at the input layer before being processed by the network.

During training, the early fusion U-Net learns to exploit the complementary information from CT and MRI modalities to delineate the boundaries of various structures within the head and neck region more accurately. This integration of information helps the network to better distinguish between

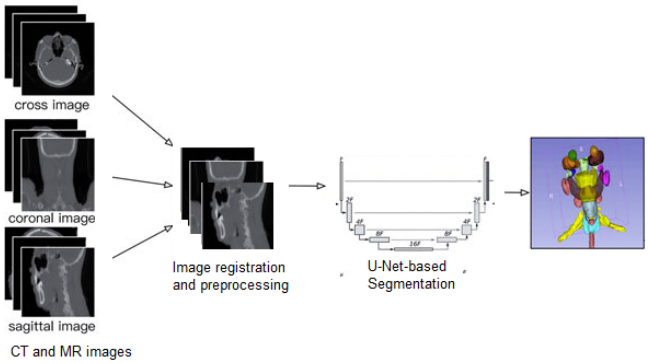


Figure 2.1: Proposed system general architecture.

different tissues and organs, leading to more precise segmentation results.

In summary, early fusion modality for head and neck region segmentation based on U-Net involves combining CT and MRI data at the input layer of the neural network to leverage the complementary information provided by these modalities, thereby improving the segmentation accuracy and robustness.

Next, we will describe the following steps for the system implementation.

2.3 Data Collection

This involves gathering a large dataset consisting of CT and MRI images of the head and neck region with precise annotations of the organs at risk. Next, it is necessary to normalize and pre-process the images to ensure consistency and optimal quality.

- The studied dataset [11] includes CT and T1-weighted MRI images from 56 patients who underwent image-guided radiotherapy. For each patient, reference segmentation of up to 30 OARs was obtained by experts who manually annotated pixel-by-pixel images. Maintaining the age and gender distributions of the patients, as well as the annotation type, the patients were randomly divided into a training set Set 1 (42 cases, 75%) and a test set Set 2 (14 cases, 25%), as shown in Figure 2.2. In fact, images from 60 patients, aged 34 to 79 years, treated with image-guided radiotherapy in the HaN region at Ljubljana, Slovenia, were obtained from the Picture Archiving and Communication System (PACS). CT images were acquired using Philips or Siemens scanners, following standard clinical protocols. MRI images were captured using a GE Medical Systems Optima MR450w scanner, employing various sequences. After exclusions for metallic artifacts or insufficient image quality, the final group comprises 56 cases with one CT image and one MRI image per patient. This group was divided into two while preserving age and gender distribution, with 42 cases in the public set (Set 1) and 14 cases in the

private set (Set 2).

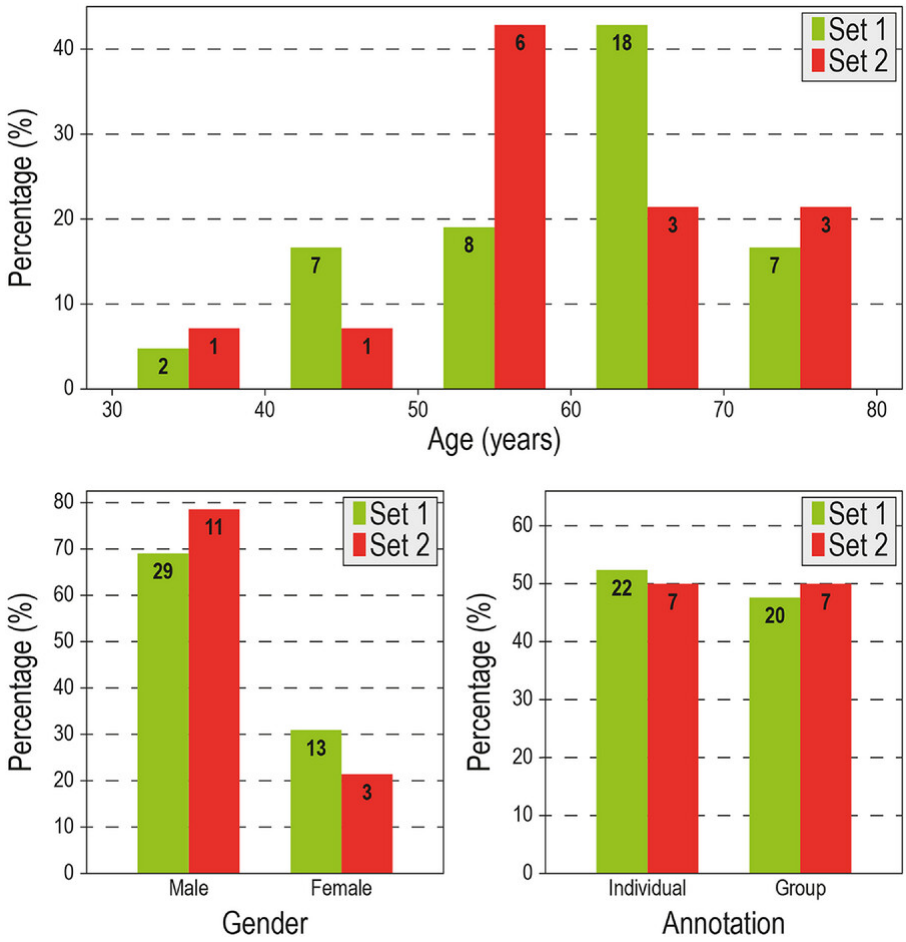


Figure 2.2: Dataset partition.

- Segmentation masks for the organs at risk (OAR) in the CT images within the dataset were created by experts from the Institute of Oncology in Ljubljana, Slovenia, using pixel-by-pixel manual annotations. Approximately half of the cases were annotated by an experienced RT technologist, and the other half by a group of specialized oncologists. MRI images were used to guide annotations in some cases. Annotations were then manually corrected to ensure accuracy and consistency. Certain regions, such as the constrictor muscles of the pharynx, were deemed unreliable and were not included in the final dataset. The final segmentation masks cover up to 30 OARs per case and are named according to the recommendations of the American Association of Physicists in Medicine (AAPM), as shown in Figure 2.3.

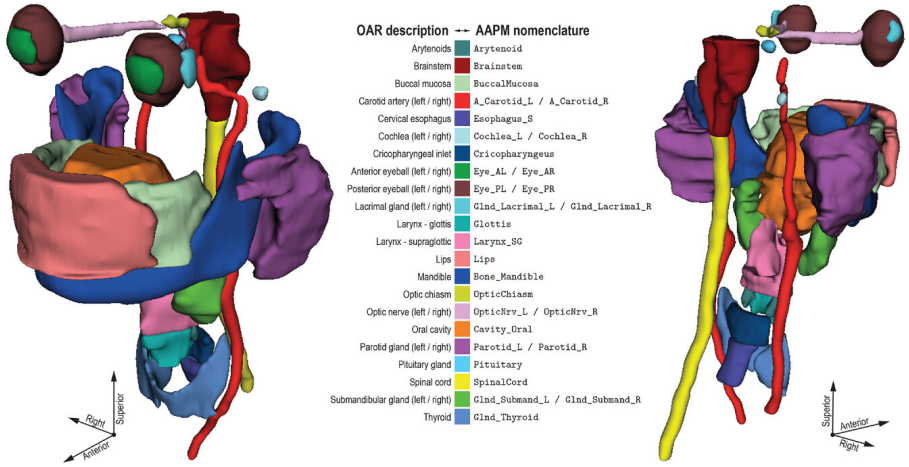


Figure 2.3: Example of reference organ-at-risk (OAR) segmentation, displayed as color-coded 3D binary masks.

- The Set 1 created is publicly accessible and can be downloaded from the open-access repository Zenodo [1] under the collection HaN-Seg: The CT and MRI segmentation dataset of organs at risk in the head and neck, with the condition that any research using this dataset cites this article. The images and reference segmentation masks are provided in the NRRD (Nearly Raw Raster Data) file format, a common format for representing and processing multidimensional raster data. Each case includes an NRRD file containing the 3D CT image and an NRRD file containing the 3D T1-weighted MRI image of the same patient, along with up to 30 NRRD files containing the 3D binary segmentation masks of the OARs for the CT image. All necessary information for image analysis is included in the corresponding NRRD files, while contextual demographic information (gender, age) is provided in a CSV (comma-separated value) file. Another CSV file indicates the availability of reference segmentation for each OAR and each case. The publicly available data is fully anonymized and does not contain protected health information. Figure 2.4 displays the files partition.

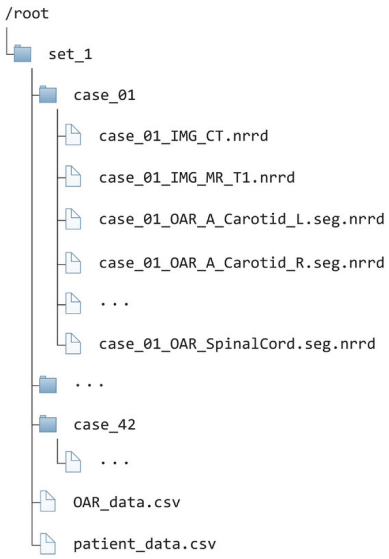
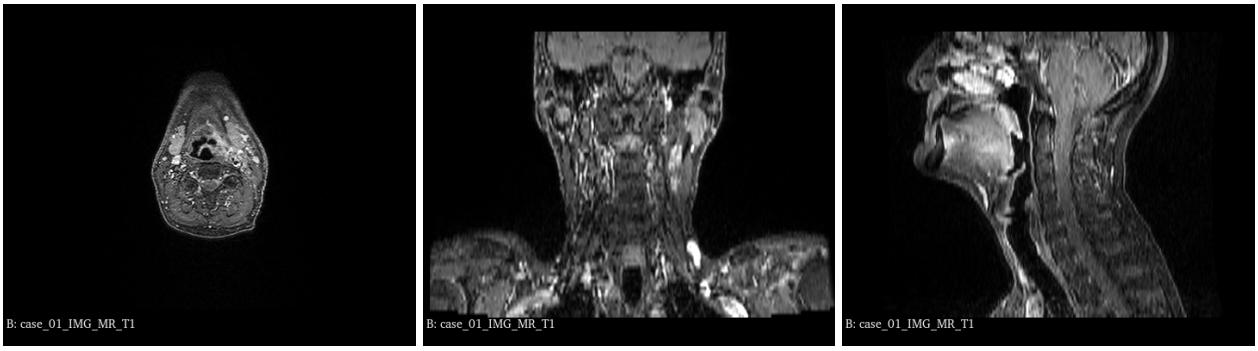


Figure 2.4: Files partition.

2.4 Data Understanding

Visualization : We utilized 3D Slicer [16] as a primary tool for visualizing and pre-processing our dataset. This open-source software provided valuable insights into the data characteristics and facilitated initial experimental preparations.

Inputs : We will now proceed with an in-depth analysis of our dataset using Case number 1 as an example. The figures below show the MRI (Figura 2.20) and CT (Figure 2.6) modalities for all orientations: sagittal, coronal, and axial. The 3D CT modality images have overall dimensions of (width, height, depth) = (1024, 1024, 202), while the MRI modality images have dimensions of (width, height, depth) = (512, 512, 83).



(a) (b) (c)

Figure 2.5: Case 1 MRI image.

(a) (b) (c)

Figure 2.6: Case 1 CT image.

Masks : Next, we present representations of specific masks for the first case, as shown in Figure 2.7, for all three orientations (sagittal, coronal, and axial). These masks accurately represent the ground truth segmentation of various organs at risk (OARs) within the head and neck region.

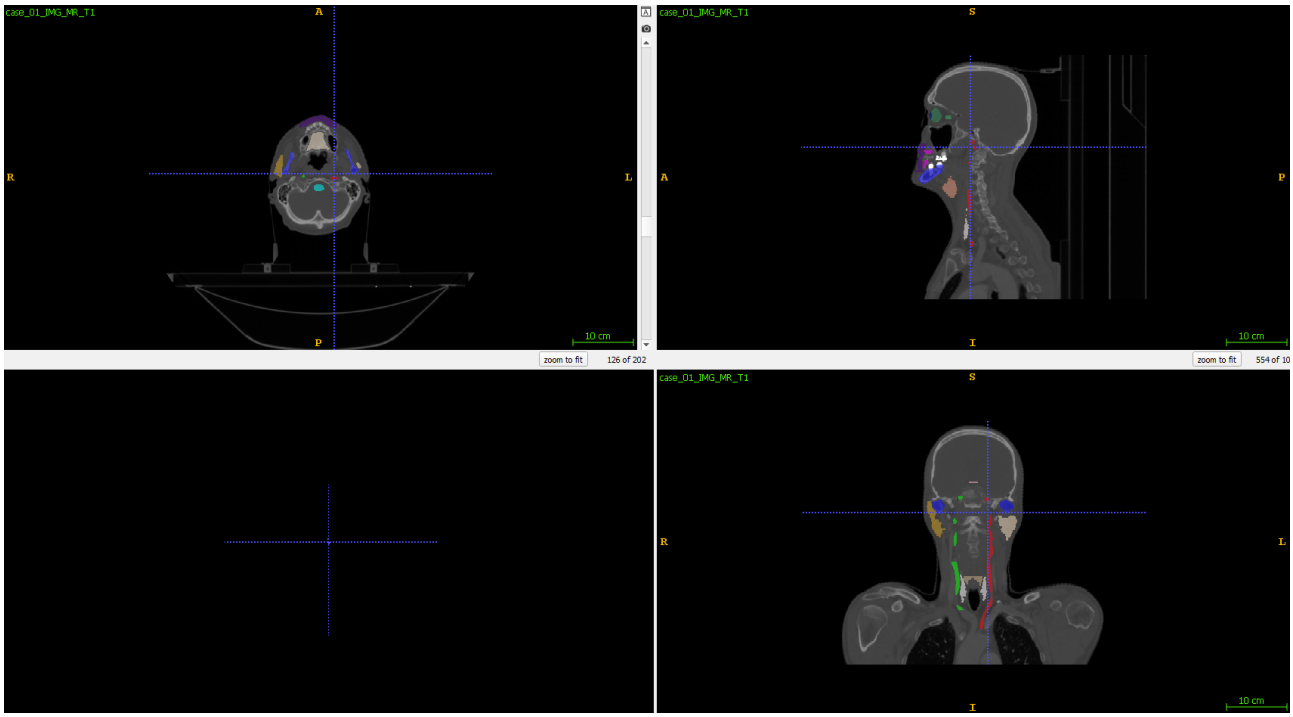


Figure 2.7: Visualisation of case 1 masks.

In the figure above 2.7, we have consolidated the 30 masks into a single mask for the first case. This step is crucial as it allows for the simultaneous visualization of segmentations for all organs at risk (OARs) in the three orientations (sagittal, coronal, and axial). By merging masks from all classes into one image, each class is assigned a specific color, aiding in the understanding and interpretation of the results. This approach provides a clear overview of the segmented anatomical structures in both CT and MRI images, which is essential for assessing segmentation quality and understanding spatial

relationships between different OARs in the head and neck region. We will also introduce a new color map dedicated to the HaN OAR for better distinction and clarity.

Early stage challenges: During initial data exploration, we encountered several challenges that required specific preprocessing steps to ensure data integrity and improve model performance:

- **Missing Masks:** To ensure the integrity of our dataset and avoid potential segmentation errors during training or testing, we performed a meticulous analysis of the provided dataset, particularly focusing on identifying and excluding cases with missing segments.
  - We utilized a CSV file associated with the dataset to identify instances where masks were missing or corrupted.
  - The value 0 indicates the non-availability of the mask for a particular anatomical structure.
  - The value 0.5 indicates a corrupted mask, often due to issues such as phase shift or misplaced segment within the image.

Case	Arytenoid	Brainstem	OpticChiasm	$\text{OpticNrv}_{\text{L}}$	$\text{OpticNrv}_{\text{R}}$	Cavity Oral	Pituitary
Case 01	1	1	1	1	1	1	1
Case 19	1	0.5	0	0.5	0.5	1	0.5

Table 2.1: A simple of missing and corrupted masks

- **Modalities-Related Issues:**
  - **Uncentered Masks:** Figure 2.8, we addressed discrepancies in mask alignment between MRI and CT voxels. visually illustrates the discrepancies in mask alignment encountered between MRI and CT voxel spaces. The misalignment of masks between these modalities can introduce significant challenges and potentially lead to inaccurate segmentation results, impacting the reliability of subsequent analyses. The specific issue highlighted in the figure pertains to the alignment of CT and organ-at-risk (OAR) masks within the voxel spaces. Notably, while the CT and OAR masks demonstrate alignment, the MRI volume appears noticeably displaced, situated far above or away from the aligned CT and OAR masks.

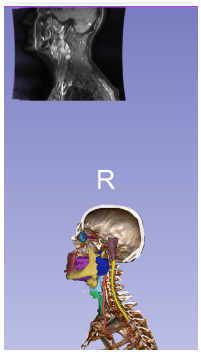


Figure 2.8: Uncentered Modalities.

- **Image Intensity Disparities:** Upon assessment, it was determined that CT image intensities met acceptable standards for our segmentation tasks. However, MR images exhibited significant variations in intensity showing very high intensities. To address this, a crucial preprocessing step involving standardizing the image intensities across MRI volumes is needed.
- **Image Size Discrepancy:** Another challenge encountered was the disparity in image sizes across the dataset. As shown in the figure 2.9, MR and CT images often have different volume shapes.

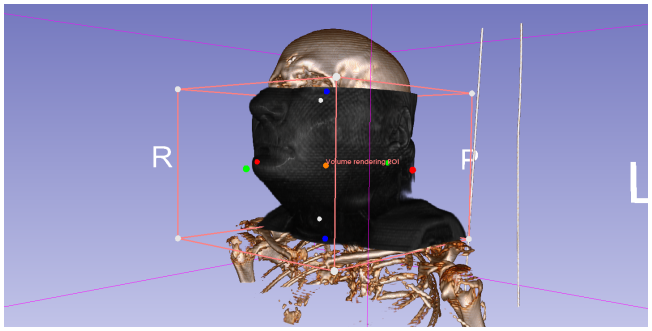


Figure 2.9: Image Size Discrepancy.

- **Class Imbalances:** We acknowledged the inherent class imbalance between large and small organs within the dataset (see Figure 2.10). We plan to address this issue through potential data augmentation techniques or employing loss functions specifically designed to handle imbalanced datasets.

By addressing these data preprocessing challenges, we aim to enhance data quality and integrity and mitigate potential biases introduced by data inconsistencies.

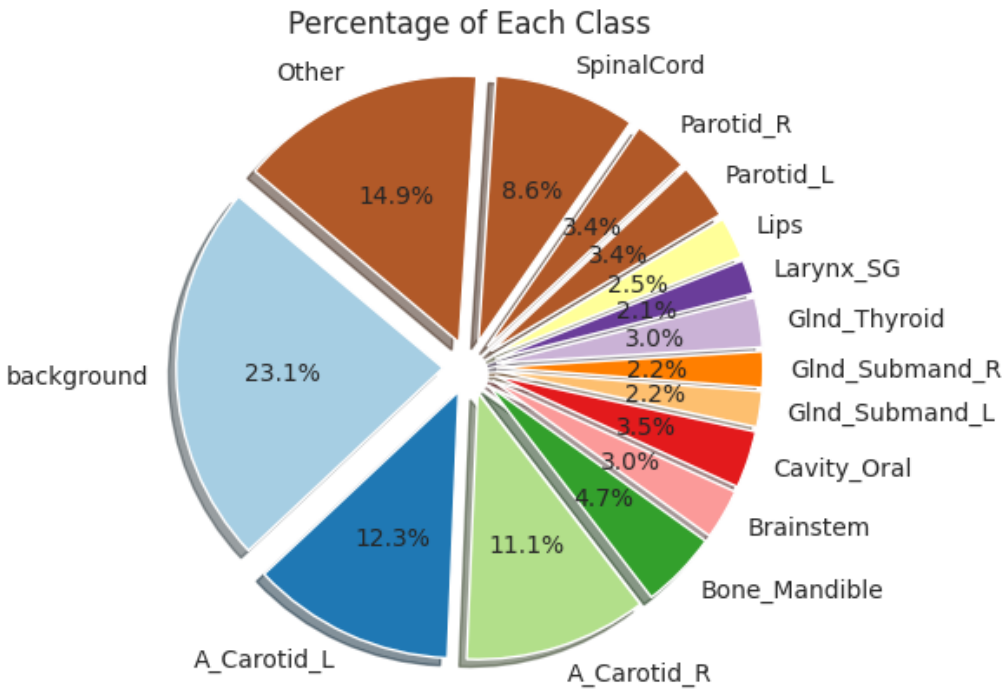


Figure 2.10: Imbalanced Classes.

2.5 Data Preparation

Data preparation forms the foundation upon which the performance of any machine or deep learning model relies, and this is particularly true in the field of medical imaging, where accuracy is vital. To provide our models with different modalities, we need to address the issues listed above. Therefore we opted to use a 3D slicer extension "Jupiter Kernel" to automate the Data preparation process. We prepared a Python script that achieves the following:

Image Standardization

The first phase of our process involves ensuring that the input data is of the highest quality and uniformity possible. This entails a series of checks and transformations to ensure that images from different modalities are compatible and ready for analysis. For instance, differences in size between CT and MR images could introduce biases into the model. To address this, we carefully cropped the CT images, ensuring not to remove areas containing crucial information. This standardization of dimensions is a prerequisite for the comparability and subsequent fusion of data, admitting a default size of $x, 512, 512$ for both modalities.

Data Optimization for the Model

This section focuses on the direct preparation of images for model training, ensuring that the data is presented in a way that maximizes learning and segmentation accuracy.

- **Image Centering and Fusion:** We centered the images from each modality before merging them into a single composite image for each case (see Figure 2.11). This approach allows the model to learn from a comprehensive and integrated representation of diagnostic features, thereby increasing segmentation accuracy.

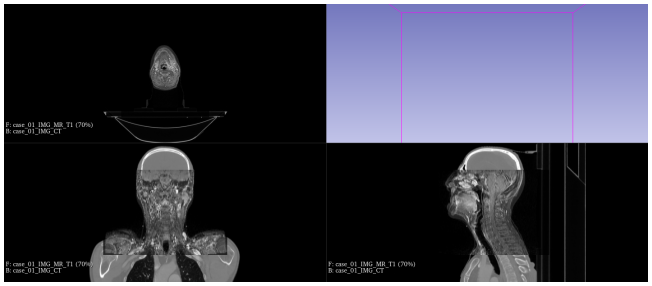


Figure 2.11: Image Centering.

- **Stacking Masks:** Stacking all segments into a single mask file is a powerful technique that simplifies the learning process. By presenting all regions of interest in a single image, we streamline the model's task and improve segmentation efficiency (see Figure 2.12).

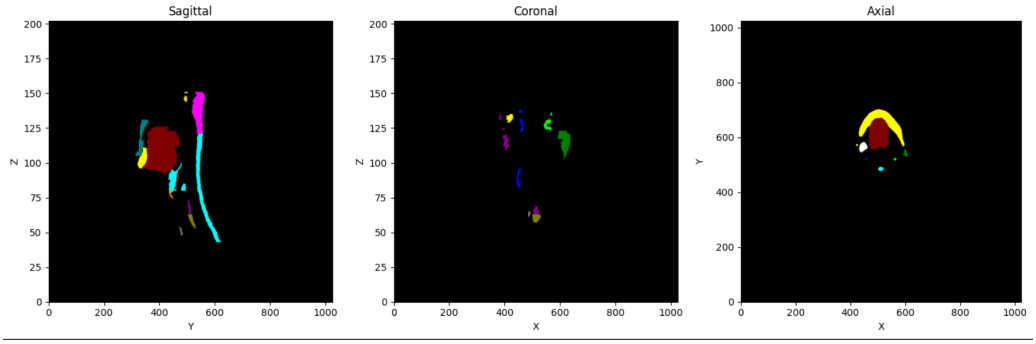


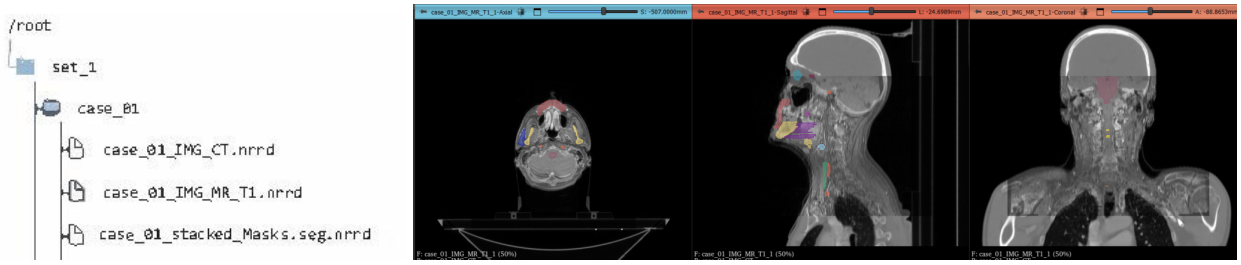
Figure 2.12: Mask Stacking.

- **Intensity Normalization:** Normalizing the intensity of MR images to match that of CT images is a critical step. It ensures that the model does not learn to favor one modality over the other, which could introduce biases and reduce model generalization.

Automation using Jupyter Kernel Extension: To streamline the data preparation process, we developed a Python script pipeline leveraging the capabilities of the 3D Slicer **Jupyter Kernel** extension.

This automated pipeline facilitated seamless execution of the preprocessing tasks discussed earlier, leading to a streamlined and reproducible data preparation process.

The culmination of these automation efforts yielded a new, clean dataset ready for subsequent segmentation tasks (see Figure 2.13). This automated approach not only saved substantial time and effort but also enhanced the reproducibility and reliability of our data preprocessing pipeline.



(a) (b)

Figure 2.13: Automation Pipeline Results

Data Preprocessing and Augmentation

The final phase of data preparation involves preprocessing images for segmentation and employing augmentation techniques to enhance model robustness. We devided the work according to two different scenarios to preprocess the data for the **2D-Unet** and **3D-Unet** .

First the following figure 2.14 demonstrates the differences between **volume slicing** and **volume patching**.

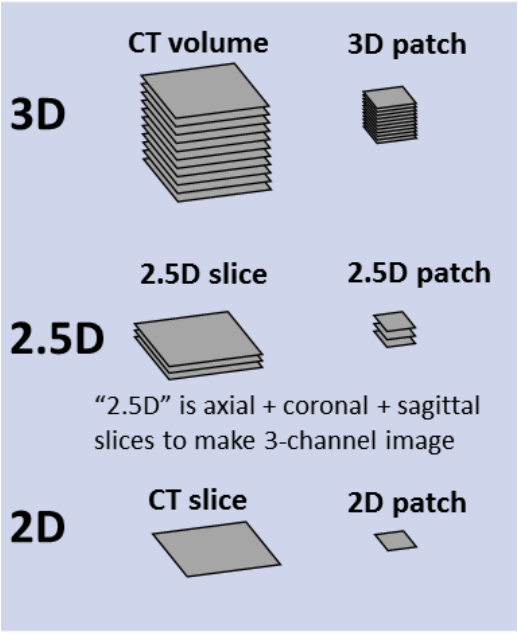


Figure 2.14: Different ways to represent medical data.[2]

We technically performed the **2.5D slicing** for the first scenario, as we make 3 channel image using the axial, coronal and sagittal slices. for the second scenario we performed **3D patching**. And we double the image channels in the end by fusing the MR and CT modalities . The following table cites the different steps of preprocessing for each modality.

- **Image and volume Cropping:** Cropping images to extract relevant regions of interest focuses computational resources on crucial parts of slices, improving processing efficiency.
- **Alignment of MR and CT Images:** Aligning MR and CT images is crucial for meaningful fusion and accurate segmentation. We developed a function to resample MR images to match CT dimensions, ensuring alignment and comparability between modalities.
- **Modalities Fusion:** Creating a new image by concatenating the MR and CT slices along the channel dimension. This results in a single image with two channels, one for each modality. Figure 2.15 displays the result of modalities overlay of the case taken as example.

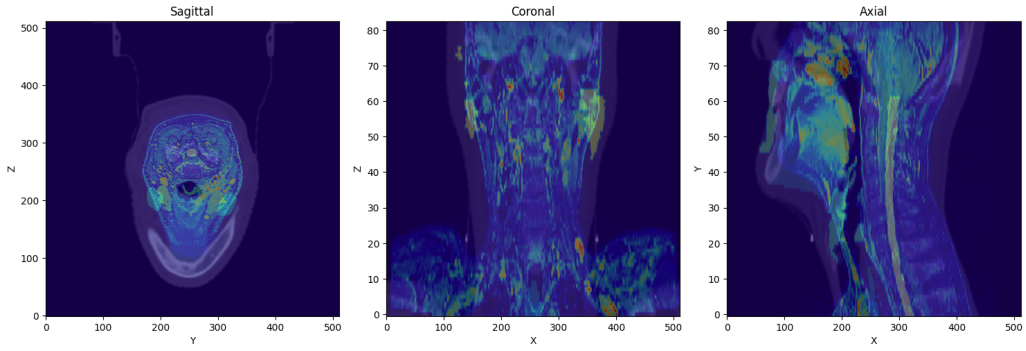


Figure 2.15: Modalities overlay.

- **Data Set Splitting:** We divided the dataset into a training set (80%), a test and validation set (20%) as it is a standard practice used to evaluate the model's generalization ability to new data. This approach assesses model effectiveness by training it on one data portion (training set) and testing it on a separate portion (test set), simulating its behavior with new observations.
- **Class Balancing:** Balancing class distribution using undersampling techniques ensures each class is represented more equally, enhancing model performance and reducing bias. This help us reduce the domination of background and big organs against minority classes of small organs.
- **Data Normalization:** For data normalization, we implemented a function that adjusts image brightness levels to ensure consistency across images, essential for subsequent processing and analysis.

- **Data Augmentation:** Data augmentation is dynamically applied during batch loading in the training phase. This strategy enriches the dataset without additional resource consumption. Techniques such as rotation, zooming, and lighting variations prepare the model to recognize organs at risk (OaRs) in various conditions, enhancing model robustness and reliability (see figure 2.16).

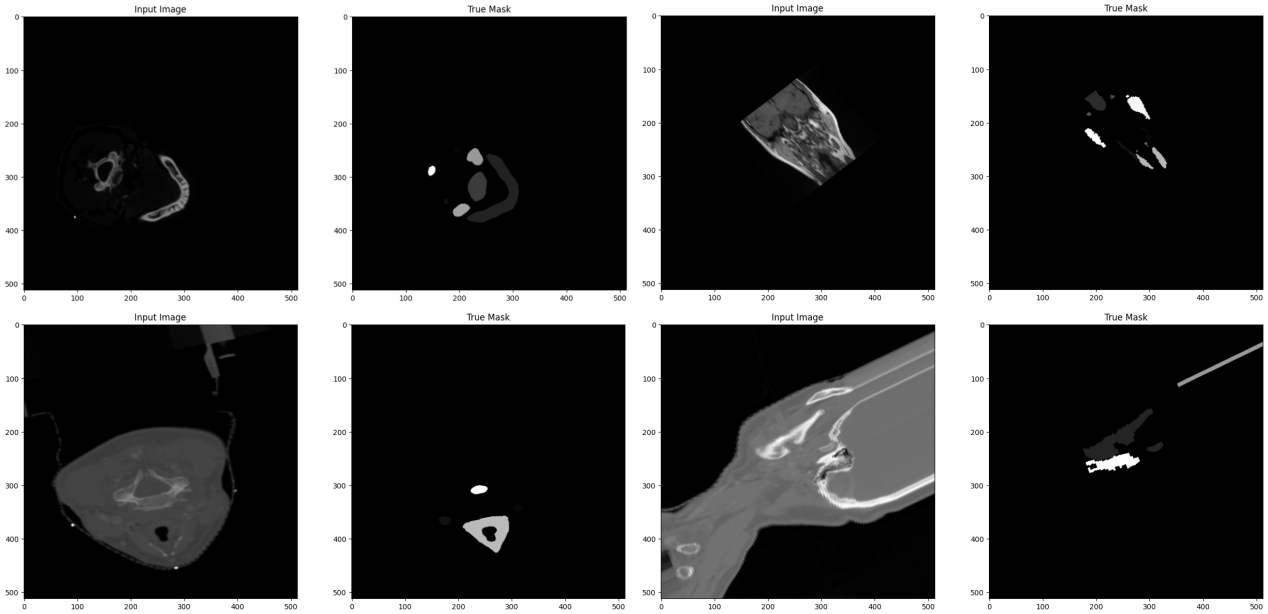


Figure 2.16: Data Augmentation

Preprocessing Results: We resulted with different train and test sets for each scenario:

Table 2.2: Preprocessing results.

First Scenario	**Second Scenario**
For this scenario, we have the advantage of loading the whole dataset, as this method does not imply any high memory resources. In fact, we will proceed by generating augmented data on the fly during training. The TensorFlow data input pipelines will help load and preprocess data in batches.	This method will require high and intense resources, as loading volumes during the preprocessing phase and crafting the new set of patches will make it very challenging to maintain the training phase with an equivalent amount of cases as in the first scenario. Thus, it is very demanding to divide the dataset into subsets due to memory insufficiencies.

After meticulously preprocessing and augmenting the data, we proceed to the modeling stage. This crucial step involves selecting and configuring appropriate deep learning architectures to tackle the segmentation task. By leveraging the prepared data, we aim to train robust and generalizable models capable of accurately segmenting organs at risk in medical images.

2.6 Modelling

2.6.1 Model selection

Famous Segmentation models architectures :

When considering segmentation tasks in deep learning, several established architectures stand out due to their effectiveness. Some of the most renowned segmentation models include U-Net, SegNet, DeepLab, and Mask R-CNN. Each of these architectures has unique characteristics and advantages, making them suitable for different segmentation tasks depending on the specific requirements of the application.

Unet overview :

U-Net is a widely used convolutional neural network (CNN) architecture designed specifically for semantic segmentation tasks, particularly in the field of medical imaging. Its architecture consists of a contracting path, which captures context through convolutional and pooling layers, and an expansive path, which enables precise localization using transposed convolutions. U-Net's distinguishing feature is the incorporation of skip connections between corresponding layers in the contracting and expansive paths, facilitating the flow of information at multiple scales and improving segmentation accuracy.

2.6.2 Crafting a case-specific Unet architecture

the first scenario will provide the Unet model with 2d slices. This technique is simply used due to its reduced computing resources. Data is prepared by batches from disk during the training phase. contrary to the second scenario with the 3D-UNet model provided with 3D patch input. training data will be allocated threw memory passed from the preprocessing phase, and this will result in memory shortage issues and lead to limits in model training.

While U-Net provides a solid foundation for segmentation tasks, it's often beneficial to tailor the architecture to the specific characteristics of the dataset and task at hand. This customization can

involve adjusting parameters such as the number of levels, initial features, kernel sizes, and number of blocks to optimize the model's performance for the given task and data distribution.

- **Scenario A : 2D model architecture** For the initial exploration of the multi-modal segmentation approach, we employed a 2D U-Net architecture. This popular network design is well-suited for tasks like medical image segmentation due to its ability to effectively capture spatial information and context.

  The 2D U-Net architecture consists of two main components: an encoder and a decoder pathway.

  - **Encoder Pathway:** The encoder pathway is responsible for extracting features from the input image. It comprises a series of **2d convolutional blocks**, each containing two convolutional layers followed by batch normalization and a ReLU activation function. Each convolutional block progressively increases the number of filters, allowing the network to learn increasingly complex features. Additionally, **max pooling** layers are employed between convolutional blocks to downsample the feature maps, reducing the spatial dimensions while maintaining relevant information.
  - **Decoder Pathway:** The decoder pathway aims to reconstruct the segmented mask based on the extracted features. It utilizes transposed convolution layers to upsample the feature maps, gradually increasing the spatial resolution. Skip connections are established between the corresponding encoder and decoder blocks. These connections concatenate the upsampled feature maps with the feature maps from the encoder pathway at the same scale. This strategy allows the decoder to incorporate detailed spatial information from the earlier stages, leading to more accurate segmentation results. Finally, a convolutional layer with a softmax activation function is applied at the output to generate the final segmentation mask with the predicted class probabilities for each pixel.

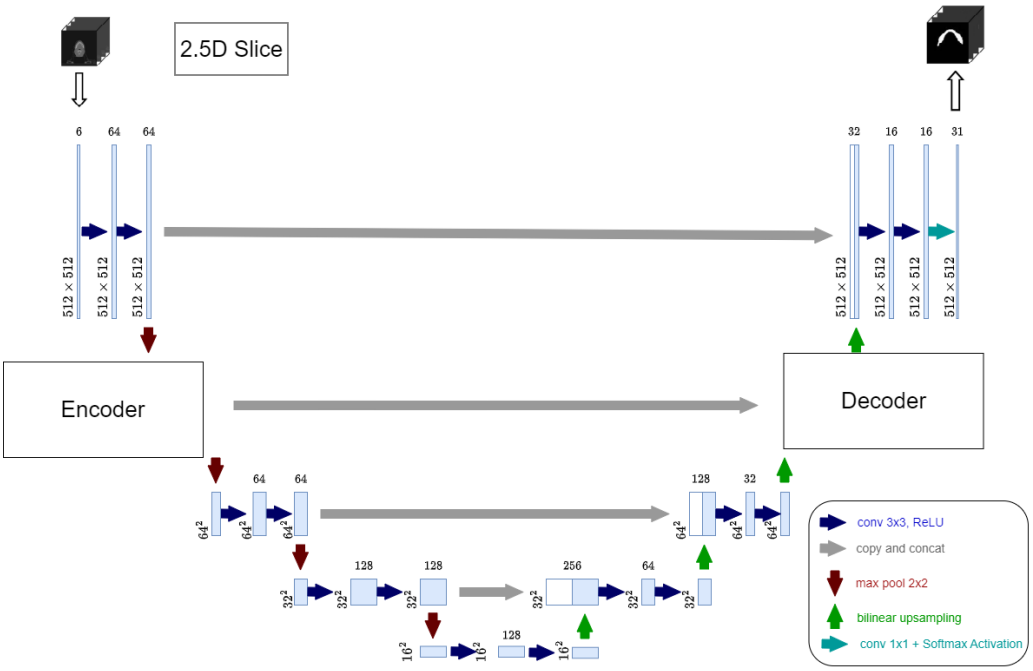


Figure 2.17: 2D Unet Architecture.

This 2D U-Net architecture serves as the foundation for our initial exploration of the multi-modal segmentation approach. Its ability to extract features and reconstruct detailed segmentation masks makes it a suitable choice for tackling the task of Head and Neck OAR segmentation. Input Layer: The model starts with an input layer with dimensions $(512, 512, 3, 2)$ representing the image size (512×512) and the number of channels (RGB). The model comprises a total of 31 classes, including background and various organs of interest within the head and neck region.

Row	Layer Name	Channels
1	inputLayer	(None, 512, 512, 3, 2)
2	2x Conv2D 3x3, MaxPool2D 2x2	(None, 512, 512, 64)
3	2x Conv2D 3x3, MaxPool2D 2x2	(None, 256, 256, 64)
10	2x Conv2D 3x3, MaxPool2D 2x2	(None, 32, 32, 1024)
11	Conv2DTranspose 2x2, Concatenate	(None, 64, 64, 512)
17	Conv2D 3x3	(None, 512, 512, 64)
18	Conv2D, Output Layer (softmax activation)	(None, 512, 512, 31)

Table 2.3: Summary of the 2D segmentation network's structure

- **Scenario B : 3D model architecture**

  Building upon the success of the 2D U-Net architecture, we opted for a 3D U-Net to handle the inherent three-dimensional nature of our Head and Neck OAR segmentation task. This choice allows us to leverage the spatial information present in volumetric medical images for more accurate segmentation results.

  The 3D U-Net architecture shares similar principles with its 2D counterpart, but with modifications tailored for 3D data using pre-trained models like VGG16 and ResNet18 in the Backbone serving as feature extractors in the encoder:

  – **Encoder Pathway:**

    * **Input Layer:**Accepts 3D volumetric data of size (64, 64, 64, 3), representing a 64x64x64 voxel with three channels (e.g., RGB or medical image channels).
    * **Convolutional Blocks:** Each block consists of convolutional layers (Conv3D) followed by batch normalization (BatchNormalization) and rectified linear unit (ReLU) activation functions (Activation). These blocks progressively reduce the spatial dimensions (length, width, and height) while increasing the number of feature channels, and capturing hierarchical features from the 3D data.
    * **Max Pooling:** max-pooling layers are employed to reduce the spatial dimensions by half along each axis (length, width, and height). This aids in capturing context while reducing computational load.
    * **Center Block:**This section consists of convolutional layers without pooling operations. These layers further refine the feature representation while maintaining the spatial information crucial for accurate segmentation.

  – **Expanding Path (Decoder):**

    * **Upsampling:**Unlike the 2D U-Net, which utilizes upsampling layers with a factor of 2, the 3D U-Net might employ specialized 3D upsampling techniques like transposed convolutions to increase the spatial dimensions and recover the original resolution.
    * **Concatenation:** Feature maps from the corresponding contracting path blocks are concatenated with the upsampled outputs. This process preserves context during upsampling and allows the decoder to incorporate detailed spatial information from the earlier stages, leading to more accurate segmentation results.
    * **Convolutional Blocks:**Similar to the contracting path, but in reverse order, these blocks gradually reduce the number of feature channels while increasing the spatial

dimensions, reconstructing the segmentation mask.

– **Output Layer:**

  * **Final Convolution:** A final convolutional layer reduces the channel dimension to the desired number of output classes (in this case, 31 classes for the Head and Neck OARs).

  * **Softmax Activation:** The final layer applies a softmax activation function (softmax) to generate class probabilities across all output channels, providing a probabilistic segmentation output for each voxel within the 3D volume.

This 3D U-Net architecture effectively captures multi-scale contextual information through its contracting and expanding paths, making it well-suited for volumetric data segmentation tasks like Head and Neck OAR segmentation in medical images.

Row	Layer Name	Channels
1	inputLayer	(None, 64, 64, 64, 3, 2)
2	2x Conv3D Block, MaxPool3D	(None, 64, 64, 64, 64)
3	2x Conv3D 3x3, MaxPool2D 2x2	(None, 32, 32, 32, 64)
4	3x Conv3D 3x3, MaxPool2D 2x2	(None, 16, 16, 16, 128)
6	2x center blocks	(None, 2, 2, 2, 512)
7	Decoder Stage 1 (Upsampling3D + Concatenate)	(None, 4, 4, 4, 1024)
15	Decoder stage 4b (Conv3D + BN + Activation relu)	(None, 64, 64, 64, 16)
16	Conv3D, Output Layer (softmax activation)	(None, 64, 64, 64 , 31)

Table 2.4: Summary of the 3D segmentation network's structure

2.6.3 Compiling the model

- **Loss function: categorical focal cross-entropy** The choice of loss function is crucial for guiding the training process and optimizing the model's performance. In this case, the Categorical Focal Cross entropy loss function is employed. This loss function is particularly effective for addressing **class imbalance** issues commonly encountered in segmentation tasks, as it assigns higher weights to misclassified examples, focusing the model's attention on challenging instances and improving overall performance.

- **Optimizer: Adam** The optimizer plays a key role in updating the model's parameters during training to minimize the chosen loss function. Adam, an adaptive learning rate optimization algorithm, is utilized in this scenario. Adam dynamically adjusts the learning rates for each

parameter individually, facilitating efficient optimization and convergence while reducing the likelihood of getting stuck in local minima.

2.6.4 Training and callbacks

- **Checkpoints** Model checkpoints are essential for preserving the model's progress during training. They allow for the saving of model weights at specified intervals, ensuring that training can be resumed from the last saved point in case of interruptions or failures. Checkpoints enable model evaluation and deployment by providing access to the best-performing model parameters.

- **early stoping** Early stopping is a technique used to prevent overfitting by monitoring the model's performance on a validation dataset during training. If the validation performance fails to improve for a specified number of epochs (patience), training is halted early. This prevents the model from memorizing noise in the training data and improves its generalization to unseen data, ultimately leading to better performance on unseen data.

- **learning rate reduction** Learning rate reduction techniques dynamically adjust the learning rate during training based on certain criteria. In this case, learning rate reduction on plateau is employed, where the learning rate is decreased if the validation loss fails to decrease for a certain number of epochs (patience). This helps fine-tune the model's performance and improve convergence by adapting the learning rate to the current training dynamics.

- **CSV logging** CSV logging involves recording training metrics such as loss, accuracy, and learning rates to a CSV file during training. This facilitates monitoring and analysis of the training process, enabling the identification of trends, anomalies, and areas for improvement. CSV logging provides valuable insights into the model's behavior and performance throughout the training process, aiding in model refinement and optimization.

- **Scenario A :**

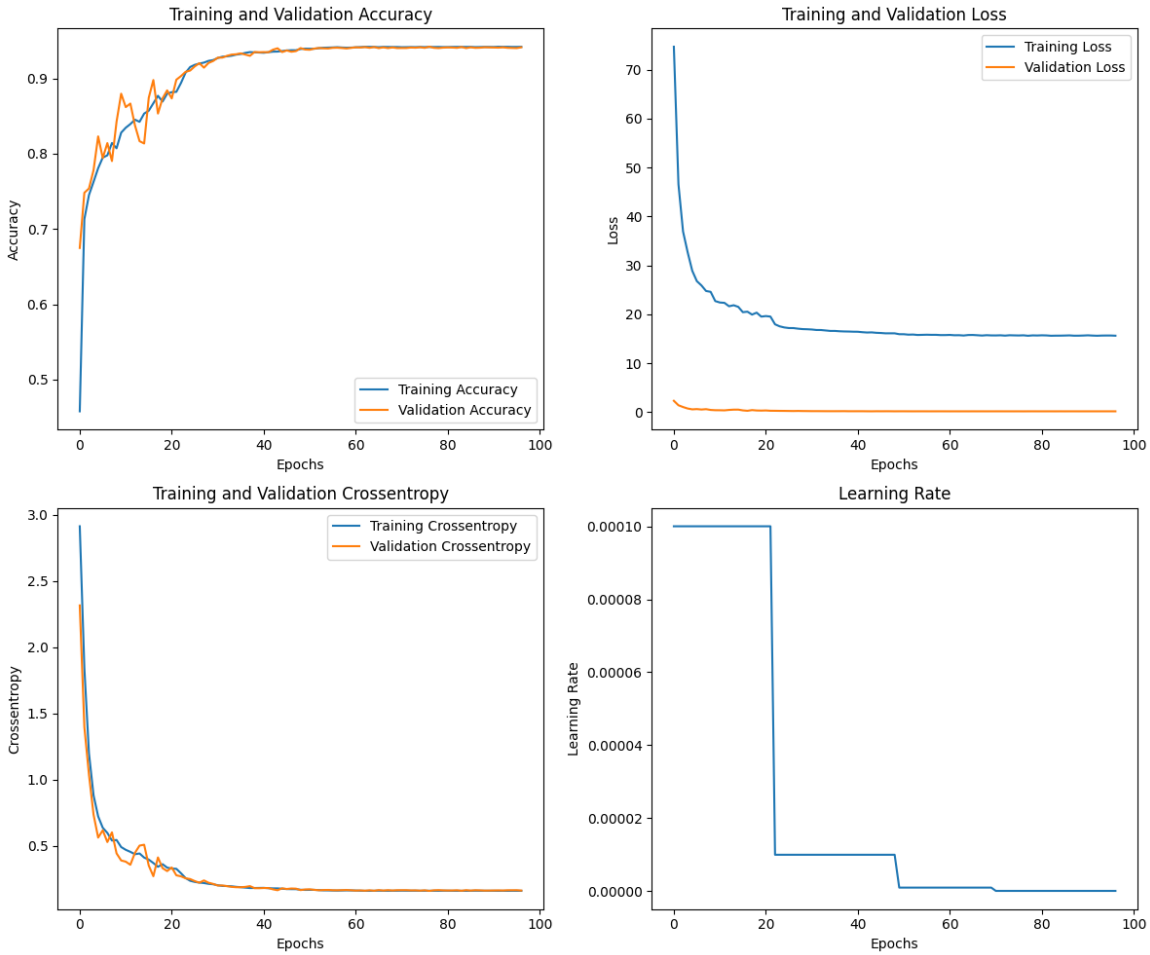


Figure 2.18: 2D-Unet Model Training History Using categorical cross-entropy

The provided graphs depict the training and validation performance metrics, including accuracy, cross-entropy, loss, and learning rate, across various epochs. These metrics offer valuable insights into the model's learning behavior and generalization ability.

Training and Validation Accuracy:

The training accuracy curve generally exhibits an upward trend, reaching a peak of approximately 94.2% after 95 epochs. This indicates that the model effectively learns to segment the organs of risk (OaRs) within the training data.

The validation accuracy curve also demonstrates a positive trend. This is a common observation, as the validation set serves as an independent measure of the model's ability to generalize to unseen data. The gap between training and validation accuracy highlights the potential for overfitting or underfitting, where the model memorizes the training data patterns but may not generalize well to novel data. However, advancing threw epochs, both training and validation stabilises with the same curve, resulting to a perfect fit.

Training and Validation Cross-Entropy:

The training cross-entropy curve shows a steady decline throughout the training process, reaching a value close to 0.16 after 95 epochs. This signifies a reduction in the model's classification error as it learns to distinguish between OaRs and background regions more effectively. The validation cross-entropy curve also exhibits a decreasing trend, with the same curve as the training cross-entropy.

Training and Validation Loss:

The training loss curve follows a similar pattern to the cross-entropy curve, decreasing steadily over the epochs. However, the values are generally higher than the cross-entropy values, which is expected as the loss function often incorporates additional terms beyond the classification error. It's important to note that the relatively high values of training loss can be partially attributed to the use of class weights. Class weights are a strategy employed with TensorFlow's Categorical Cross-entropy loss function to address class imbalances within the dataset. When classes are not equally represented, assigning higher weights to minority classes helps the model focus on learning those classes more effectively. This can lead to higher overall loss values, even if the model is successfully distinguishing between classes. This also explains the gap between the training and validation Loss.

Learning Rate:

The learning rate graph depicts a steady decrease in the learning rate from 1e-04 to 1e-07 over the course of 95 epochs. This indicates the implementation of a learning rate reduction technique, which is a common strategy to optimize the training process. By gradually reducing the learning rate, the model can make smaller adjustments to its parameters as it progresses through training. This can help to mitigate overfitting and potentially improve the model's generalizability. 2.6.4.

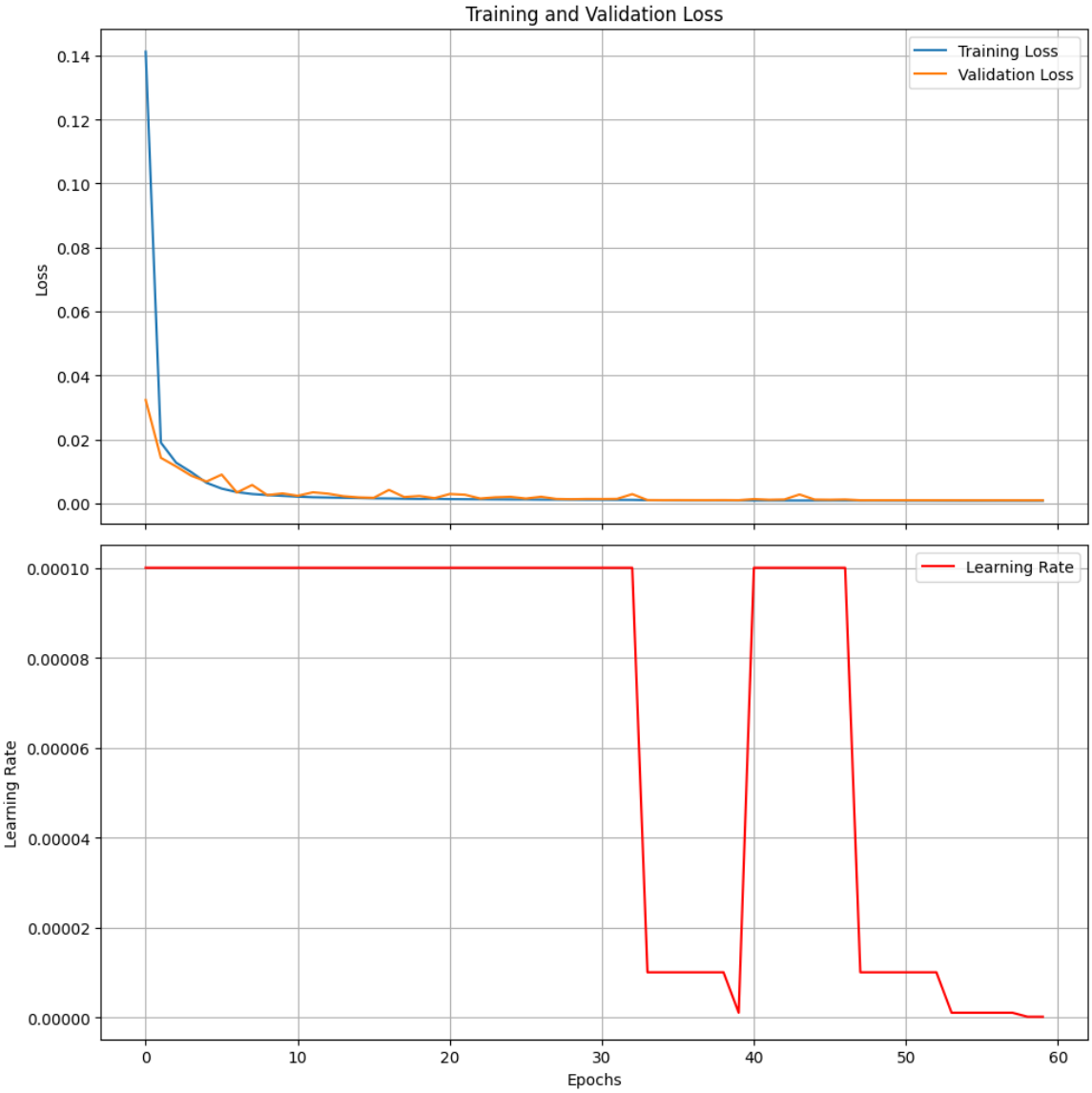


Figure 2.19: 2D-Unet Model Training History Using categorical focal cross-entropy

As observed in the figure 2.18, the initial training loss values were relatively high. To mitigate this issue and improve the model's performance, we employed the categorical focal loss function instead of the traditional categorical cross-entropy loss.

Categorical focal loss offers several advantages over cross-entropy, particularly when dealing with class imbalance:

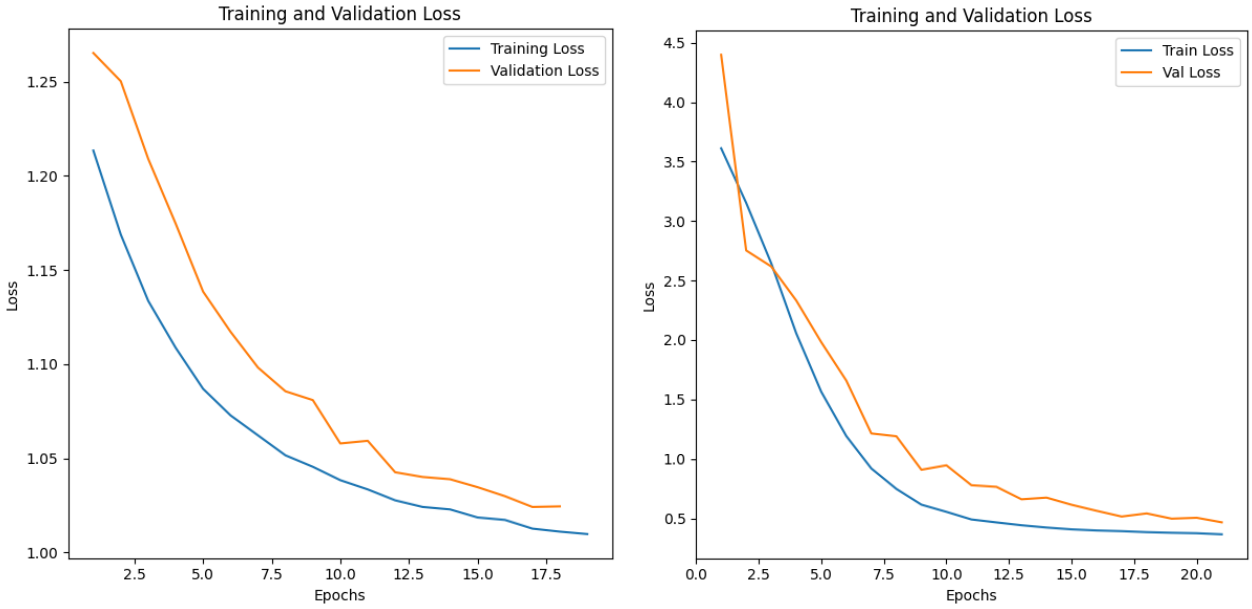
Focuses on Hard-to-Classify Examples: Focal loss downplays the impact of easily classified samples, giving more weight to misclassified or hard-to-learn examples. This encourages the model to prioritize learning from the more challenging instances, potentially leading to better overall performance. The provided figure 2.19 demonstrates the effectiveness of this approach. We can observe a significant reduction in both training and validation loss compared to the initial values. This improvement suggests that the categorical focal loss function successfully

addressed the high training loss issue and potentially contributed to a better fit on the training data.

Furthermore, the gap between training and validation loss has narrowed, indicating a potential reduction in overfitting. This suggests that the model is generalizing better to unseen data, which is a crucial aspect of robust model performance.

In conclusion, utilizing categorical focal loss instead of categorical cross-entropy proved beneficial in this scenario. It effectively reduced training loss, potentially improved model learning, and contributed to a more balanced training process, as evidenced by the reduced gap between training and validation loss.

- **Scenario B :**



(a) (b)

Figure 2.20: 3D-Unet Training and Validation Loss over epochs.

The provided graphs (a) and (b) depict the training and validation loss of the 3D-Unet model for two different patch sizes: 64x64x64 and 128x128x128. While both graphs exhibit a decreasing trend in training loss and some degree of generalization based on the validation loss, a key observation is the reduced gap between training and validation loss in graph (b) using the larger patch size.

This suggests that using 128x128x128 patches potentially leads to better generalization and reduced overfitting compared to the 64x64x64 patch size. This can be attributed to the larger patches providing the model with a wider context of the 3D image, allowing it to capture more

spatial information and potentially learn relationships between neighboring voxels more effectively.

However, it's crucial to acknowledge the trade-offs associated with larger patch sizes:

Increased Computational Cost: Training with larger patches requires significantly more computational resources due to the increased number of parameters and memory demands.

Reduced Training Speed: Processing larger patches takes longer, potentially slowing down the training process.

In our case, these resource limitations hindered the implementation of a model with a larger input size. Therefore, we opted for the first scenario as a practical compromise between capturing sufficient context and computational feasibility.

2.7 Model Evaluation

We evaluated the model's performance on the validation and test sets by calculating metrics such as the Dice Similarity Coefficient (DSC) and Intersection over Union (IoU) based on overlapping regions (overlap-based segmentation metrics), as well as HD95 (Hausdorff Distance at 95%) based on boundaries (boundary-based segmentation metric).

The Dice Similarity Coefficient (DSC) and Intersection over Union (IoU) are the most popular and conceptually easy-to-understand evaluation methods for segmentation. For two segmentations A and B, they are calculated as follows:

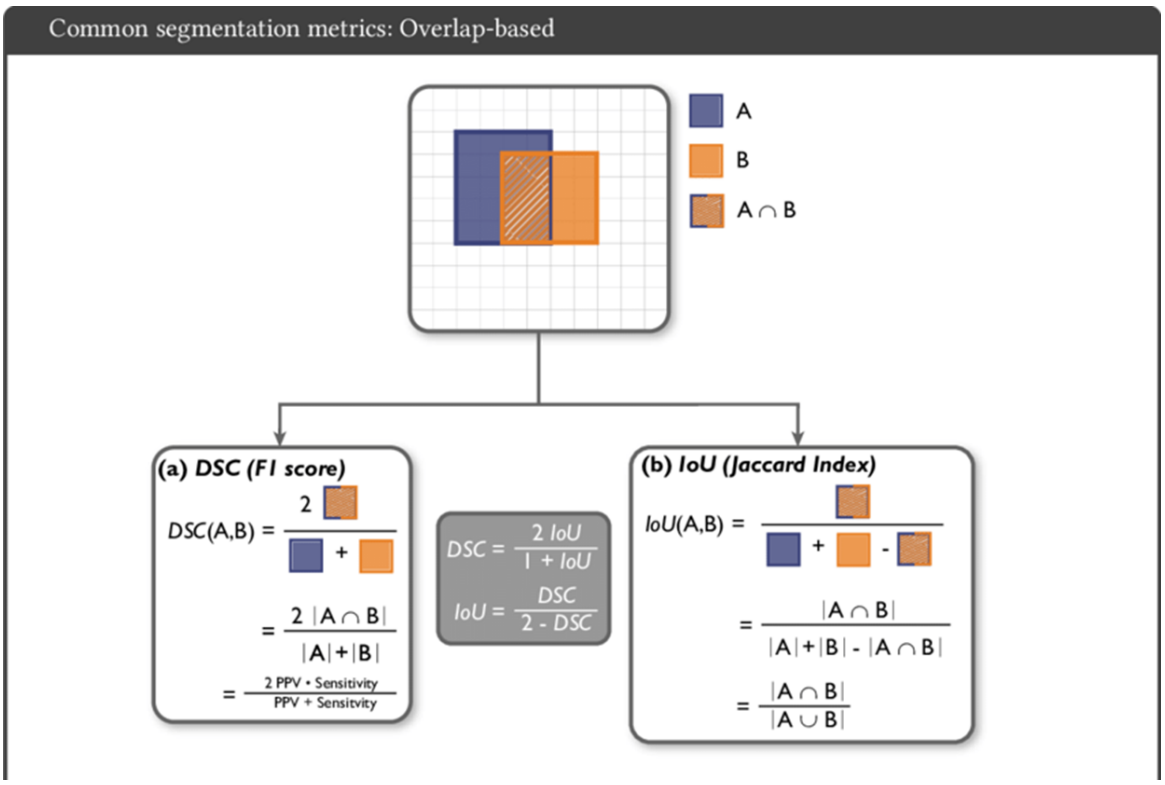


Figure 2.21: Calculation Formulas for DSC and IoU.

In geometry, the Hausdorff distance is a topological tool that measures dissimilarities between two subsets of an underlying metric space. The figure below explains the calculation of this distance.

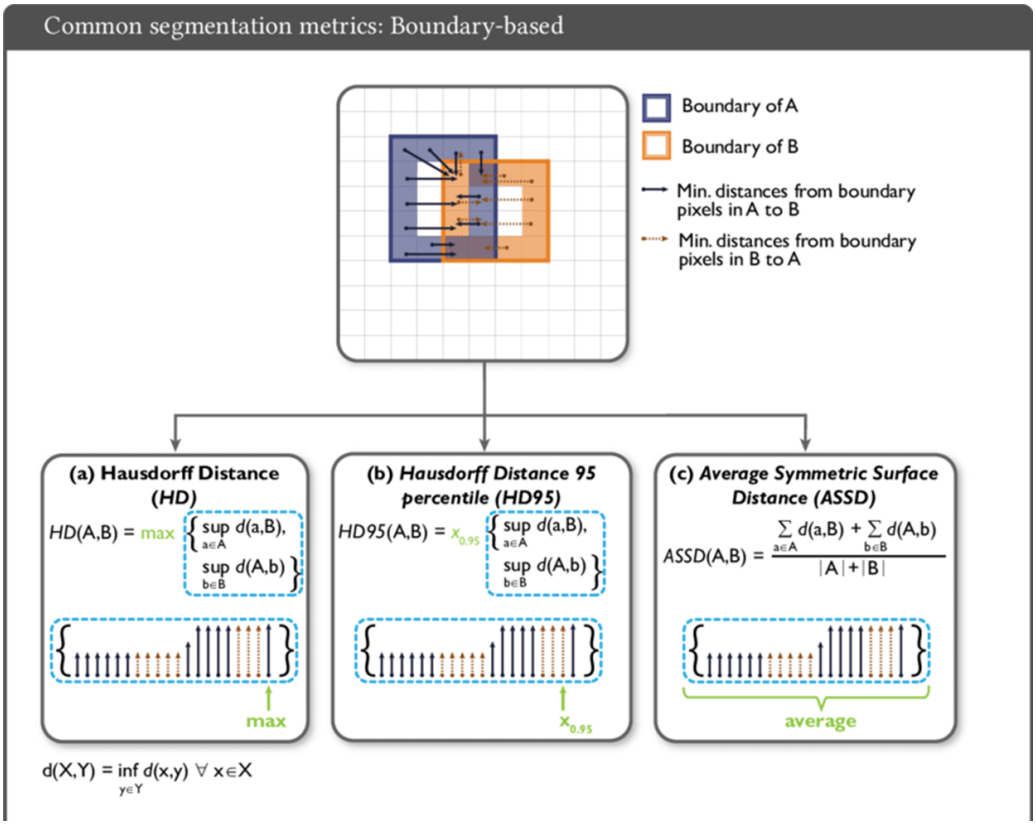


Figure 2.22: Formulas for calculating the Hausdorff distance.

We also visually analyzed the segmentation results to verify the accuracy and precision of organ localization, and to identify weaknesses and potential areas for improvement.

However, the DSC has been identified as not the most appropriate measure for evaluating the clinical adequacy of segmentations, especially when results are close to inter-observer variability. Additionally, it is not suitable for small-volume structures. On the other hand, distance-based measures such as HD95 (Hausdorff Distance at 95%) are preferred as they better measure the shape consistency between reference and predicted segmentations.

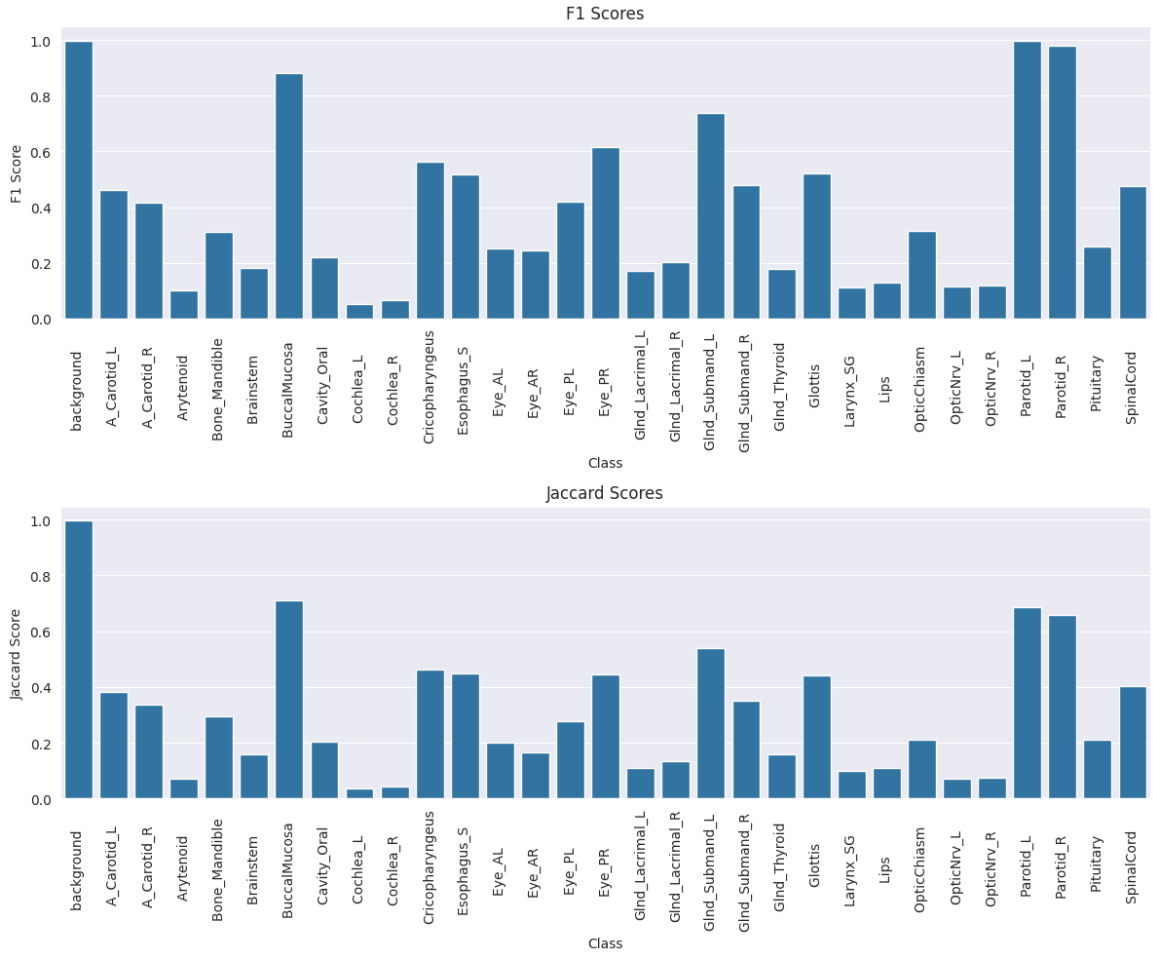


Figure 2.23: F1 score and Jaccard index

The figure shows a wide range of F1 and Jaccard scores across different classes. We can detect:

- **High-Performing Classes**: Classes like Parotid L, Parotid R, BuccalMucosa, and Glnd Submand L have very high F1 scores (>0.9) and Jaccard indices (>0.6), indicating excellent performance in segmentation.
- **Medium-Performing Classes**: Classes like Cricopharyngeus, Esophagus S, Eye PR, and Glottis have moderate F1 scores (0.4 - 0.6) and Jaccard indices (0.3 - 0.5), suggesting acceptable performance.

- **Low-Performing Classes**: Classes like Arytenoid, Bone Mandible, Brainstem, and Cochlea L have low F1 scores (<0.3) and Jaccard indices (<0.2), indicating poor segmentation performance.

Class	Hausdorff distance
background	16.7
$\text{Carotid}_{\text{L}}$	135.8
$\text{Carotid}_{\text{R}}$	148.1
Bone Mandible	123.8
BuccalMucosa	31.6
Cricopharyngeus	183.4
Esophagus S	174.0
Eye PL	218.6
Eye PR	166.8
$\text{Glnd Submand}_{\text{L}}$	51.9
$\text{Glnd Submand}_{\text{R}}$	249.3
Glottis	194.5
$\text{Parotid}_{\text{L}}$	25.6
$\text{Parotid}_{\text{R}}$	27.9
SpinalCord	61.3

Table 2.5: Hausdorff distance Values for Classes

The rest of unrepresented classes had infinite Hausdorff distance values witch means that they had bad placement in the prediction and the model is not capable of segmenting these images. they mainly represent the minority classes. Here is a representation of classes with low values that need to be improved and worked on :

Class	F1	Jaccard	Hausdorff distance
Arytenoid	0.1027	0.0714	∞
Brainstem	0.18106	0.16	∞
Cavity_Oral	0.21891	0.20238	∞
Cochlea_L	0.0522	0.0345	∞
Cochlea_R	0.0645	0.0439	∞
Eye_AL	0.2526	0.1999	∞
Eye_AR	0.2459	0.1662	∞
Glnd_Lacrimal_L	0.1699	0.1087	∞
Glnd_Lacrimal_R	0.2017	0.1341	∞
Glnd_Thyroid	0.17846	0.15992	∞
Larynx_SG	0.11143	0.09843	∞
Lips	0.13036	0.11006	∞
OpticChiasm	0.316	0.21	∞
OpticNrv_L	0.1148	0.0718	∞
OpticNrv_R	0.1173	0.0748	∞
Pituitary	0.26	0.2117	∞

Table 2.6: Classes with infinite Hausdorff distance values

The table 2.5 shows HD values for some classes, while others have infinite values 2.6. Classes with infinite HD indicate the model failing to predict those structures entirely. These are likely minority classes based on the text provided. Among the classes with finite HD values, there's a significant variation, ranging from 16.7 for background to 249.3 for Glnd Submand R.

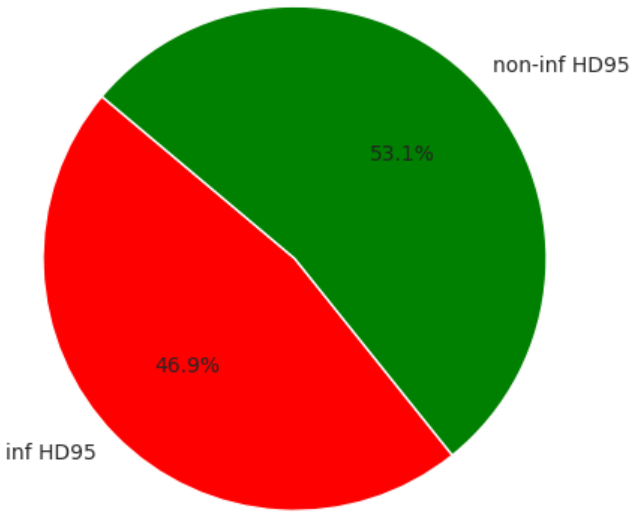


Figure 2.24: Percentage of classes based on Hausdorff distance metric

The results suggest that the model performs well for some classes but not for others.

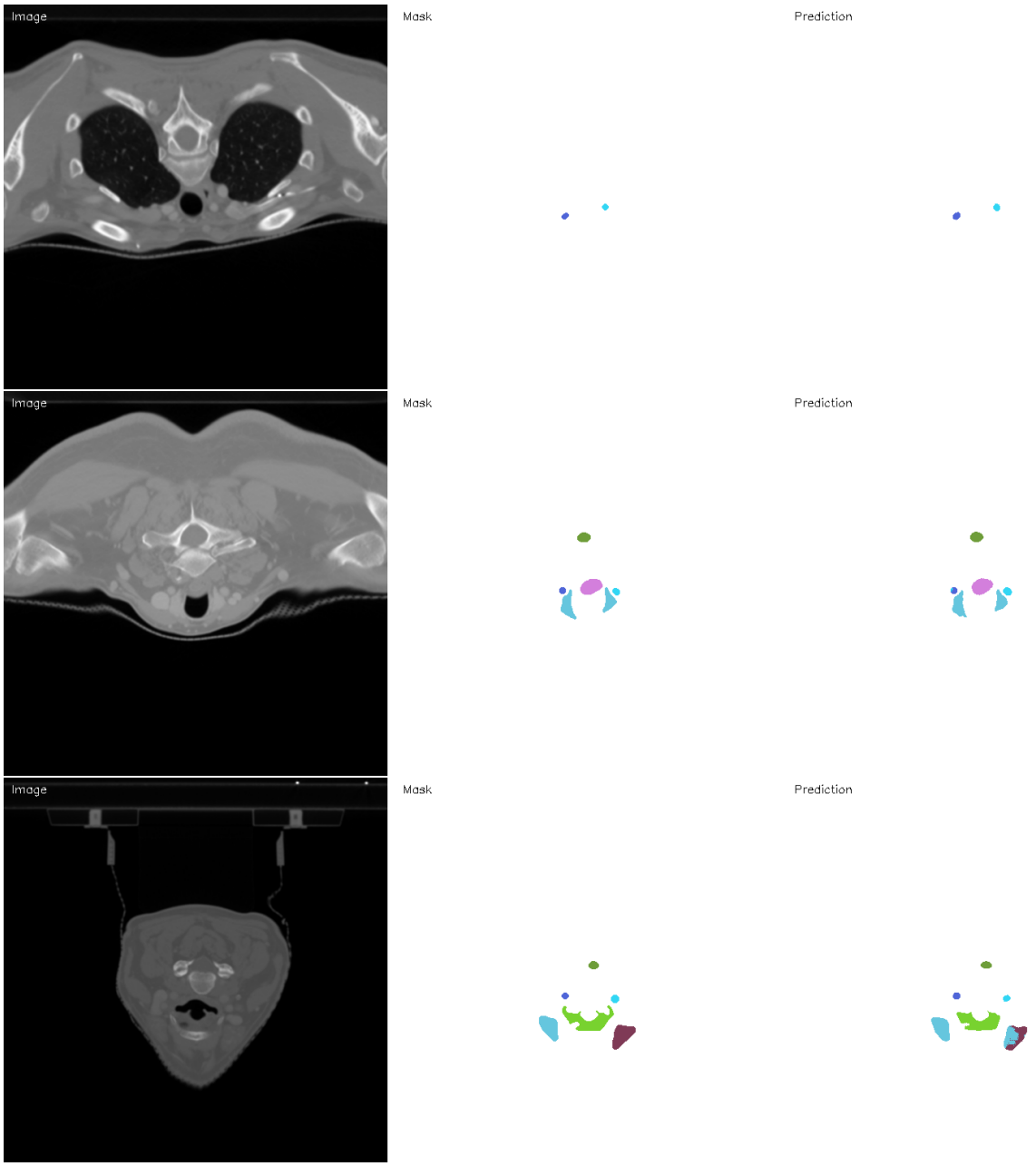


Figure 2.25: Ground Truth and Prediction Comparison

We achieved an average inference time of 18.23 seconds on one fused MR and CT scan on an NVIDIA RTX 4060 GPU. Additionally, the average inference time is 0.024 seconds per slice for a single CT scan, resulting in a 4.488 seconds full volume scan average inference time.

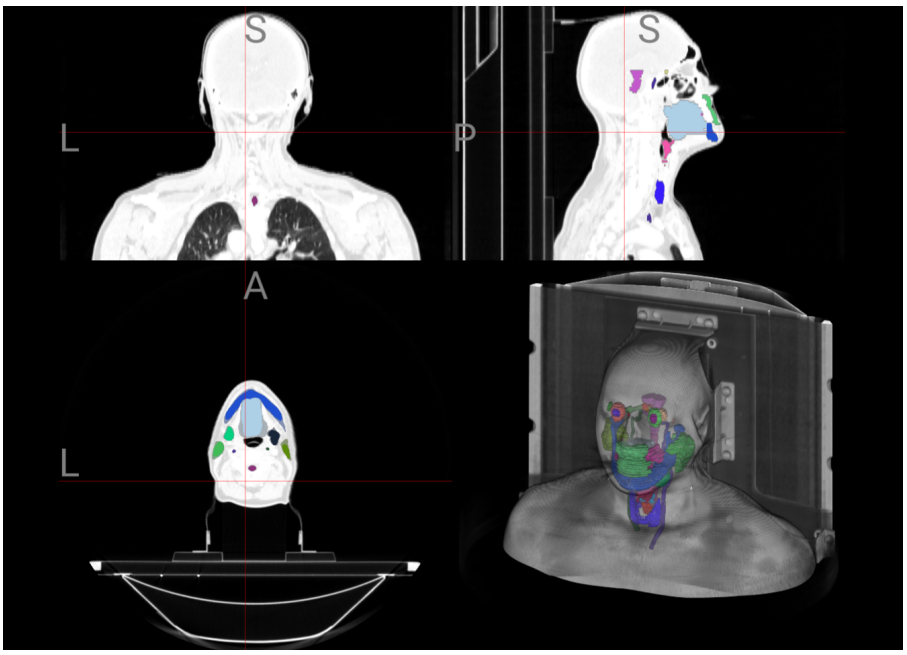


Figure 2.26: Case Prediction

The results presented in Figures 2.25 and 2.26 demonstrate the potential of the proposed model for medical image segmentation. However, it is crucial to acknowledge the mixed performance observed across different classes. While the model achieves satisfactory results for specific classes, further investigation is necessary to address the limitations observed in others.

2.8 Integration into a Clinical System

Following the model training and evaluation, the next crucial step was to make the model accessible and usable for real-world applications.
For the proposed segmentation model to have a tangible impact, integration into a clinical system for practical use by healthcare professionals is crucial.

2.9 Conclusion

In conclusion, we developed and evaluated a multimodal segmentation system for identifying organs at risk in head and neck CT and MRI images. Following the CRISP-DM methodology, we used UNet architecture models for segmentation tasks.

Our system employs early modality fusion of CT and MRI data to enhance segmentation accuracy by leveraging complementary information. Evaluation metrics such as IoU, and Hausdorff distance provided insights into segmentation performance across different classes.

Integrating our model into a clinical system is crucial for real-world applicability, requiring seamless integration and robust performance. While we achieved promising results, addressing class-specific challenges and refining the model remain ongoing objectives.

The next chapter will discuss deploying our model into a web solution application, focusing on practical implementation aspects.

Chapter 3

Experimentation and Results

3.1 Introduction

This chapter delves into the practical aspects of deploying the developed Head and Neck segmentation model as a user-friendly web application.This chapter details the deployment process, the development of a user interface, and the resulting web application [10].

3.2 Work environment

3.2.1 Hardware Environment

During the development of our application, we used the personal computer with the following specifications:

- **Brand:** DELL
- **RAM:** 32.0 GB
- **Processor:** 13th generation Intel Core i7
- **GPU:** NVIDIA RTX 4060
- **Operating System:** Windows 11

3.2.2 Software Environment

During the implementation of our project, we used the following technologies:

- **Version Control System:**

- **Git:** Git is a version control system (VCS) that tracks the history of changes to the source code. It allows us to quickly identify changes made in the project [**1**].

Figure 3.1: Git [9]

- **Containerization:**
  - **Docker:** Docker is a containerization platform that allowed us to create, deploy, and manage software containers containing all the dependencies of our application [**1**]. We used several Docker images, including:
    - **nginx:latest [13]:** An image containing the Nginx web server, used to host our front-end application.
    - **python:3.10.14:** We build the flask prediction service based on a python container.
    - **tensoflow/serving:latest-gpu [17]:** REST API used for hosting multiple model versions.
    - **node:20.11.0:** We used the node image to build and run the visualize container and the main React application container.

    Docker allowed us to ensure a consistent development environment and to deploy our application efficiently.

Figure 3.2: Docker [4]

- **Development Environment:**
  - **PyCharm 2023.3.4:** PyCharm is an integrated development environment (IDE) for developing Python code.

3.3 Model Deployment

To facilitate user interaction and model prediction, the chosen deployment strategy involved a web application accessible through a web browser. This approach offers several advantages:

Figure 3.3: PyCharm [15]

Accessibility: Users can access the model from any device with an internet connection, eliminating the need for local software installations.

Ease of Use: A web interface simplifies user interaction, making the model readily available without requiring extensive technical knowledge.

3.3.1 Deployment process

1. **Initial Deployment:**

   TensorFlow Serving (TFServing) was chosen to initially deploy and test different model versions. This open-source platform allows for deploying and testing different model versions within Docker containers. This containerized approach enables easy testing and scalability through API calls.

2. **Performance Optimization:**

   Server response time was a critical factor for a smooth user experience. To minimize loading times, efficient image processing algorithms were developed.

   Flask played a crucial role in seamlessly integrating the pre-processing steps used during model training into the deployment stage. This significantly reduced image loading times, achieving a remarkable improvement from 15-5 minutes on the CPU to just 20 seconds on the GPU.

3. **Visualization Integration:**

   Following the successful deployment of the prediction service, a user-friendly interface was designed for interacting with the model.

   Extensive research led us to **Niivue** [3], a JavaScript library specifically designed for efficient visualization of 3D medical images, particularly NRRD files.

Figure 3.4: 3D Web Visulization for medical images.

4. **React-based Application:**

   A React application was developed to integrate seamlessly with both the Flask-based prediction service and the Niivue visualization library. This React app leverages a custom-built user interface (UI) based on Niivue to provide a comprehensive user experience.

Figure 3.5: Website UI.

3.3.2 User Scenario:

The web application facilitates a straightforward user experience:

Image Upload:

Users begin by uploading a 3D MR/CT image or even a single 2D image.

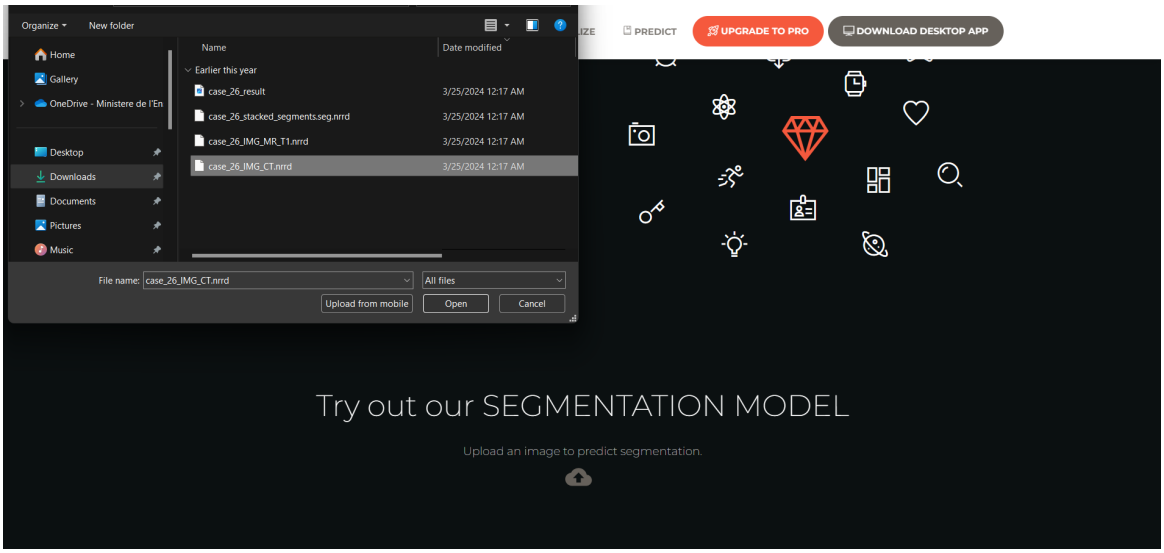


Figure 3.6: Uploading image

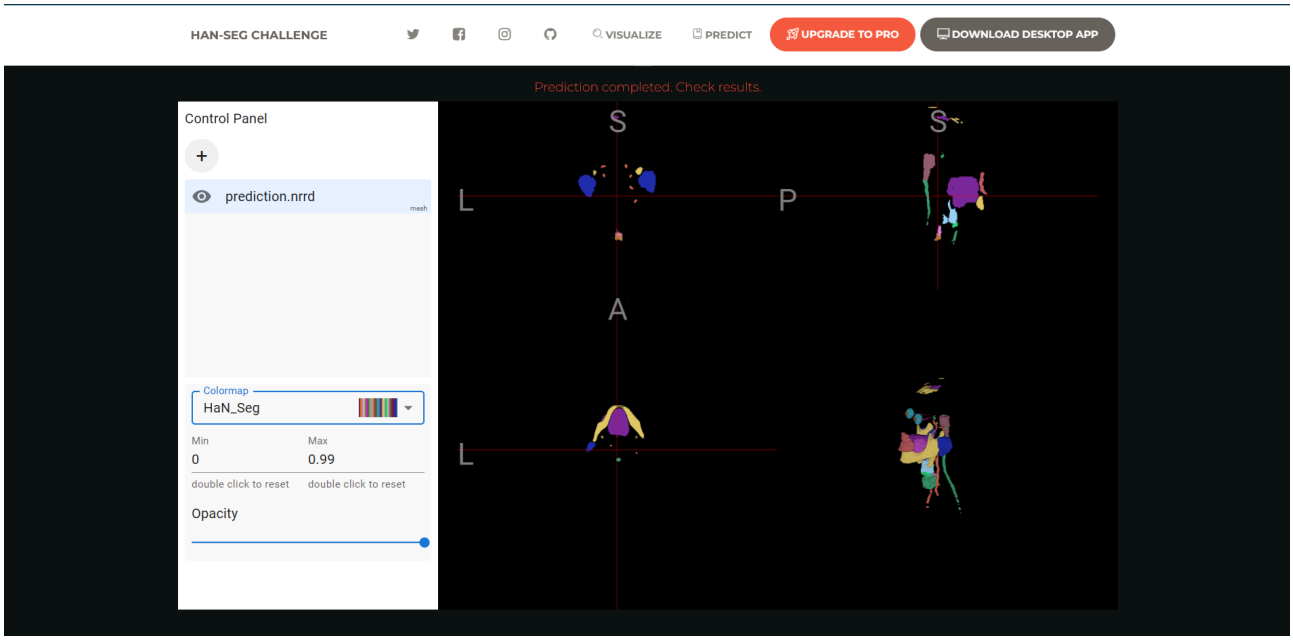


Figure 3.7: Visualize prediction.

Prediction Service Interaction:

The uploaded image is transferred to the Flask service, which performs the necessary pre-processing steps. The pre-processed image is then forwarded to the TFServing instance for model prediction. Finally, the predicted mask, representing the segmentation results, is returned. (see figure 3.6).

Visualization:

The received prediction results are then visualized using the Niivue integration within the React

application. This allows users to interact with and analyze the model's output. (see figure 3.6).

The developed web application serves as a user-friendly platform for predicting segmentations of 31 organs at risk within the Head and Neck region. This interactive platform allows medical professionals to leverage the model's capabilities for analysis and decision-making purposes.

3.4 Conclusion

The successful deployment of the Head and Neck segmentation model as a web application marks a significant step towards its practical application in clinical settings. This user-friendly platform offers medical professionals a readily accessible tool for image analysis, potentially improving patient care and treatment planning.

Conclusion and perspectives

In conclusion, our exploration of multimodal segmentation for head and neck cancer delineation from MRI and CT images led us through two distinct approaches. The 3D approach encountered limitations primarily due to computational resource constraints, hindering progress and causing challenges with patching and computational efficiency. However, recognizing the need for an alternative strategy, we transitioned to a 2D approach.

The 2D approach, involving the segmentation of image slices and utilizing a 2.5D slicing technique, proved to be a more feasible solution. Despite the setbacks encountered with the 3D method, the 2D approach yielded promising results, demonstrating improved efficiency and segmentation accuracy.

Furthermore, to make our model accessible to users, we deployed it into a functional web application. This application, developed with React for the frontend, Flask for the backend, and Niivue for 3D visualization, allows users to upload volumetric images in nrrd format and receive segmented images in return. This deployment underscores our commitment to translating research findings into practical applications, offering medical professionals a user-friendly tool to aid in diagnosis and treatment planning for head and neck cancer patients.

Bibliography

[1] HaN-Seg Dataset. *HaN-Seg: The head and neck organ-at-risk CT MR segmentation dataset*. Accessed: May 2024. 2023. URL: https://zenodo.org/records/7442914#.ZBtfBHbMJaQ.

[2] Marleen De Bruijne et al. *Medical Image Computing and Computer Assisted Intervention–MICCAI 2021: 24th International Conference, Strasbourg, France, September 27–October 1, 2021, Proceedings, Part III*. Vol. 12903. Springer Nature, 2021.

[3] niivue Developers. *niivue: WebGL2 based medical image viewer*. Accessed: May 2024. 2024. URL: https://github.com/niivue/niivue.

[4] *Docker*. Address : https://en.wikipedia.org/wiki/Docker_(software). [Accessed: May 2024].

[5] Dolz et al. "HyperDense Net: a hyper-densely connected CNN for multimodal image segmentation". In: *Trans. Med. Imaging* 38 (2019), pp. 116–1126.

[6] Matthew S Edwards et al. "CRISP method: the key to success in Data Science". In: ().

[7] Anticancer Fund. *Head and Neck Cancers: A Guide for Patients—Based on ESMO Recommendations*. ESMO-ACF, 2021, p. 34.

[8] Yunhe Gao and Yunhe Gao. "FocusNetv2: Imbalanced large and small organ segmentation". In: *Medical image analysis* 67 (2020), p. 101831. URL: https://api.semanticscholar.org/CorpusID:225110279.

[9] *Git*. Address : https://fr.wikipedia.org/wiki/Git. [Accessed: May 2024].

[10] *HaN-Seg deployment*. Address : https://github.com/aziz0220/Hanseg_deployment_PFA/. [Accessed: May 2024].

[11] Head and Neck Segmentation Challenge. *Head and Neck Segmentation Challenge 2023*. Accessed: May 2024. 2023. URL: https://han-seg2023.grand-challenge.org/.

[12] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully convolutional networks for semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3431–3440.

[13] *NginX*. Address : `https://www.nginx.com/resources/glossary/nginx/`. [Accessed: May 2024].

[14] Gasper Podobnik, Primoz Strojan, and Peterlin. “HaN-Seg: The head and neck organ-at-risk CT and MR segmentation dataset”. In: *Medical Physics* (2023). URL: `https://doi.org/10.1002/mp.16197`.

[15] *PyCharm*. Address : `https://www.jetbrains.com/pycharm/`. [Accessed: May 2024].

[16] 3D Slicer. *3D Slicer image computing platform*. Accessed: May 2024. URL: `https://www.slicer.org/`.

[17] *TensorFlow*. Address: `https://www.tensorflow.org/`. [Accessed: May 2024].

[18] Yueyue Wang et al. “Organ at Risk Using a Two-Stage Segmentation Framework Based on 3D U-Net”. In: *Digital Medical Research Center, School of Basic Medical Sciences, Fudan University, Shanghai, China* (2017), pp. 1–10.

[19] Yan et al. “Longitudinal detection of diabetic retinopathy early severity grade changes using deep learning”. In: *OMIA* 12970 (2021), pp. 11–20.

[20] Yifei Zhang et al. “Deep multimodal fusion for semantic image segmentation: A survey”. In: *Image and Vision Computing* 105 (2021), p. 104042.

Résumé

Le cancer de la tête et du cou requiert une segmentation précise des organes à risque (OAR) pour la radiothérapie. Notre système intègre CT et IRM pour une segmentation automatisée des OAR. Après avoir testé les approches 3D et 2D, nous avons opté pour cette dernière, plus réussie malgré les contraintes computationnelles. Notre modèle U-Net 2D assure une segmentation précise avec des précisions compétitives de 94.2% en entraînement et 94.1% en test, promettant d'améliorer la radiothérapie pour ce cancer.

Mots-clés : cancer de la tête et du cou, segmentation des OAR, planification de la radiothérapie, imagerie CT, imagerie par résonance magnétique, modèle U-Net.

Abstract

Head and neck cancer requires precise segmentation of organs at risk (OAR) for radiotherapy. Our system integrates CT and MRI for automated segmentation of OARs. After testing both 3D and 2D approaches, we opted for the latter, which was more successful despite computational constraints. Our 2D U-Net model ensures accurate segmentation with competitive accuracies of 94.2% in training and 94.1% in testing, promising to improve radiotherapy for this cancer.

Keywords: head and neck cancer, OAR segmentation, radiotherapy planning, CT imaging, MR imaging, U-Net model.

ملخص

يتطلب سرطان الرأس والرقبة تفصيل دقيق للأعضاء ذات الخطورة لتخطيط العلاج بالإشعاع بشكل فعّال. يدمج نظامنا المقترح الصور المقطعية المحوسبة والرنين المغناطيسي لتقسيم تلقائيًا. بعد أن اختبرنا النهجين الثلاثي الأبعاد والثنائي الأبعاد ، اخترنا الأخير الناجح رغم التحديات الحسابية. يضمن نموذجنا او ت ٢ض تقديم تقسيم دقيق بدقة تنافسية تبلغ ٩٤٠٢% في التدريب و ٩٤٠١% في الاختبار، مما يعد بتحسين العلاج بالإشعاع لهذا النوع من السرطان.

كلمات مفاتيح : سرطان الرأس والعنق، تحديد الأعضاء ذات المخاطر، تقسيم متعدد الأوضاع، تعلم عميق، اونت، التصوير المقطعي المحوسب، الرنين المغناطيسي، العلاج الإشعاعي.