

Organ Specific Deep Learning Networks for Anatomical Structure Segmentation

A PROJECT REPORT
SUBMITTED IN PARTIAL FULFILMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
Master of Technology
IN
Faculty of Engineering

BY
Nirmalya Gayen



Computer Science and Automation
Indian Institute of Science
Bangalore – 560 012 (INDIA)

July, 2023

Declaration of Originality

I, **Nirmalya Gayen**, with SR No. **04-04-00-10-51-21-1-19464** hereby declare that the material presented in the thesis titled

Organ Specific Deep Learning Networks for Anatomical Structure Segmentation

represents original work carried out by me in the **Department of Computer Science and Automation** at **Indian Institute of Science** during the years **2022-2023**.

With my signature, I certify that:

- I have not manipulated any of the data or results.
- I have not committed any plagiarism of intellectual property. I have clearly indicated and referenced the contributions of others.
- I have explicitly acknowledged all collaborative research and discussions.
- I have understood that any false claim will result in severe disciplinary action.
- I have understood that the work may be screened for any form of academic misconduct.

Date:

Student Signature

In my capacity as supervisor of the above-mentioned work, I certify that the above statements are true to the best of my knowledge, and I have carried out due diligence to ensure the originality of the report.

Advisor Name:

Advisor Signature

© Nirmalya Gayen
July, 2023
All rights reserved

DEDICATED TO

The Student Community

who can use and reuse this template to glory

Acknowledgements

I would like to express my deepest gratitude to my advisor, Professor Vijay Natarajan, as my first and foremost acknowledgment. His unwavering guidance and continuous support throughout the project have been invaluable.

I am immensely grateful for the assistance and support extended by Nithin Shivashankar from Mimyk. Nithin played a pivotal role as a secondary advisor, offering invaluable insights, constructive feedback, and guidance throughout my work. His suggestions for potential areas of focus have been instrumental in shaping the direction of my project. I am deeply appreciative of his contributions to my research.

I extend my heartfelt appreciation to Dr. Kirthi Koushik from the HCG Ramaiah Cancer Center for his invaluable inputs. His guidance has played a significant role in enhancing my understanding of the Brachytherapy procedure at Ramaiah, as well as the current technologies employed in their practices and the limitations they face. Moreover, he has greatly contributed to my comprehension of medical terminology and the anatomical structure of organs. I am truly grateful for his assistance, which has been crucial to my research endeavors.

I would like to express my appreciation for the remarkable work carried out by Aditya, my senior colleague, in the project. His diligent efforts in data collection and exploring multiple approaches have been truly commendable. The dedication and expertise Aditya brought to the project greatly contributed to its success. I am grateful for the valuable contributions made by Aditya in advancing our research.

Abstract

The accurate identification of specific structures within volumetric data derived from diverse scanning devices holds considerable significance in various biomedical applications. However, existing solutions face several challenges. Primarily, the integration of results obtained from the segmented outputs of multiple organs poses difficulty. Additionally, the detection of tubular topological organs, such as the rectum and sigmoid, which have distinct nomenclatures based on their positions, presents challenges. Furthermore, gender-specific organs may exhibit sub-optimal performance in current methodologies. Lastly, the detection of small organs within the context of the entire body remains a challenging task.

The primary objective of this study is to achieve accurate segmentation of three specific organs from volumetric CT images. Once the segmentation process is completed, the subsequent aim is to determine an optimal needle insertion path and perform a simulation in the identified regions.

Contents

Acknowledgements	i
Abstract	ii
Contents	iii
List of Figures	v
List of Tables	vi
1 Introduction	1
1.1 Motivation	1
1.1.1 Surgical Guidance	2
1.1.2 Personalized Patient Care	2
1.1.3 Medical Image Automation	2
1.2 Objectives	2
1.3 Contributions	3
1.3.1 Data-collection	3
1.3.2 Region of Interest	3
1.3.3 Topographic Organs	3
1.3.4 Organ Specific Models	3
1.3.5 Recto-Sigmoid Junction	3
1.3.6 Visualization	4
2 Dataset	5
2.1 Label Issues	5
2.2 Organ Ranges	6
2.3 Region of Interest	6

CONTENTS

2.3.1	Close to Organ	6
2.3.2	Removing Unlabelled Slice	7
3	Literature Review	8
3.1	U-Net	8
3.1.1	U-Net (2D)	9
3.1.2	U-Net (3D)	9
3.1.3	U-Net (ConvLSTM2D)	9
3.2	LOSS FUNCTIONS	9
4	Methodology	11
4.1	Pre-Process	11
4.1.1	Reading Input file	12
4.1.2	Getting Bounding Box (Geometric)	12
4.1.3	Getting Bounding Box (ML)	14
4.2	Training of Organ Specific Models	16
4.3	Prediction of the Organs	20
4.4	Post Process	20
4.4.1	Removing disconnected components	20
4.4.2	Prediction of Recto-Sigmoid junction	20
5	Evaluation	24
6	Implementation and Visualization	27
6.1	Import CT images	27
6.2	Visualize CT Image	27
6.3	Region of Interest	28
6.4	Prediction	28
6.5	Recto-Sigmoid junction	28
6.6	Connected Components	28
6.7	Surface Visualization	28
6.8	Export Segmentation	29
7	Conclusions and Future Work	30
	Bibliography	32

List of Figures

2.1	Region of Interest	6
2.2	Uncertain Junction	7
3.1	U-Net model	8
4.1	Segmentation Framework	11
4.2	Bounding Box	13
4.3	Boundary Points and Selected Boundary	15
4.4	Prediction vs. Ground Truth	18
4.5	Prediction vs. Ground Truth	18
4.6	Training Accuracy for Bladder	19
4.7	Methods to find recto-sigmoid junction	21
4.8	Predicted Boundary Points and Selected Boundary	23
5.1	Predicted vs Ground Truth 2D	26
5.2	Predicted vs Ground Truth 3D	26
6.1	Predicted vs Ground Truth 2D	29

List of Tables

4.1	mean absolute error for the ML approach	16
4.2	mean absolute error for the Geometric & ML approach	16
4.3	Frequency of the approach being selected	23
5.1	Compare different training approaches	25
5.2	Compare different training stages	25

Chapter 1

Introduction

Organ segmentation is a crucial task in medical imaging and computer vision that involves identifying and delineating specific organs or anatomical structures within medical images, such as MRI or CT scans. It plays a vital role in various clinical applications, including disease diagnosis, treatment planning, surgical guidance, and monitoring disease progression.

Accurate organ segmentation enables healthcare professionals to extract quantitative measurements, assess tissue characteristics, and identify abnormalities or plan surgical operations more precisely, allowing for a deeper understanding of the underlying pathology.

Organ segmentation is a challenging problem due to the complex and variable appearance of organs, variations in imaging modalities, noise, artefacts, and anatomical differences across individuals. However, recent advancements in machine learning and deep learning networks, like convolutional neural networks (CNNs), have revolutionized organ segmentation by achieving remarkable accuracy and efficiency. Post-processing algorithms further enhance the segmentation results, improving their clinical utility by increasing the segmentation speed, as deep learning networks, for example, CNN, can run faster than conventional algorithms.

1.1 Motivation

Many different types of radio-logical scanners generate a variety of datasets. Finding important information from them and correctly segmenting organs or geometrical structures is sometimes hard. This creates a need for an expert to find and segment the data. This takes time which can sometimes be fatal for patients who might need immediate surgery, as planning a surgery requires knowing the placement of important organs in the patient's body.

1.1.1 Surgical Guidance

Specific applications like Brachytherapy needs prior knowledge of organ positions which can help plan the process [11]. Precise segmentation with 3D visualization can help surgeons plan, simulate and automate the process to improve patient outcomes.

1.1.2 Personalized Patient Care

Patients can have very different organ structures and placements, so having segmented organs can give an idea for the specific patient and plan accordingly. Sometimes it can be very helpful as some patients have very different anatomical structures.

1.1.3 Medical Image Automation

With the increase in imaging and the use of those devices, we can not rely on humans to do the entire labelling of organs. We need automation and faster methods to help surgeons to work faster and confidently.

1.2 Objectives

Many works were done for some individual organs, and different models of CNN networks have also been discussed. Like: [2] Bladder segmentation based on deep learning approaches (2021) using FCN, Semi-automatic sigmoid colon segmentation in CT for radiation therapy treatment planning via an iterative 2.5-D deep learning approach (2021) using U-Net, etc. And also, some papers discuss multi-organ segmentation like TotalSegmentator: robust segmentation of 104 anatomical (2021), which is close to perfect. Still, it fails in gender-based organs for some datasets. No good segmentation algorithms can currently segment small, tubular organs and their separation points.

Application Oriented: We are driven by the goal of revolutionizing the segmentation process for the specific application of Brachytherapy in cervical cancer treatment [8]. To achieve this, our primary focus lies on delivering precise and accurate segmentation of the organs or specific organ regions that are critical to this application. By dedicating our efforts to these key organs, we aim to develop tailored solutions that address Brachytherapy's unique challenges and requirements [3]. Our ultimate aim is to provide healthcare professionals with enhanced tools and technologies that empower them to make informed decisions, optimize treatment planning, and improve outcomes for cervical cancer patients.

1.3 Contributions

The project aims to create an organ segmentation tool to accurately segment the Bladder, Rectum and Sigmoid for Brachytherapy and use the output for treatment planning and simulation. The contribution we made for that are as follows:

1.3.1 Data-collection

Data collection was one of the most important works done in the project's previous iteration. We have collected 150 CT data of different patients from one hospital.

Then we anonymized the data and randomized them based on the scan date, and separated 100 data from training, 20 for validation and 30 for testing.

1.3.2 Region of Interest

We needed to focus on three specific organs to pre-process the image to find the region where our organs are in the image. This helped us reduce unnecessary data and focus on the important region for the specific application. We developed two methods to solve the issue one based on a geometric algorithm and another based on ML.

1.3.3 Topographic Organs

The continuous organs but named differently based on their position, are hard to distinguish, especially with 2D slices of 3D images. For example, the rectum and sigmoid are similar if we see a 2D slice without knowing the position of the slice, as tissue and structure can be similar, so either we could use two DNNs to train the structures with slices from individual organs only, or we could train one DNN for the entire structure and later find the recto-sigmoid junction. We chose the latter as it fastens the prediction time and produces continuous organ.

1.3.4 Organ Specific Models

We have trained organ-specific 2D U-Nets, which have proven to be highly effective in accurately segmenting the organs from the slices within the region of interest. This achievement showcases the robustness and precision of our approach in achieving reliable organ segmentation results.

1.3.5 Recto-Sigmoid Junction

We have segmented the rectum and sigmoid as the same organ in the previous step, and finding the junction point is important as we need to know the position of the sigmoid for radiotherapy. So we developed geometrical algorithms to find the point.

1.3.6 Visualization

In order to utilize the algorithms, we have developed a comprehensive software tool that facilitates visualization of each step involved. This software not only enables user interaction, if required, but also provides the resulting outputs for further processing. This integrated solution empowers users to effectively utilize the algorithms, offering a seamless workflow for visualizing, interacting with, and leveraging the results obtained.

Chapter 2

Dataset

The study utilizes volumetric images acquired from CT scanners at a single hospital. The dataset comprises 150 samples of CT scans obtained from various individuals. Each image has a resolution of 512x512 pixels for every frame (or slice) in the transverse plane, with the number of frames ranging from 85 to 134. Additionally, the provided dataset includes annotated labels for the organs.

The organs are marked as follows for training:

- 1 for Bladder
- 2 for Rectum
- 3 for Sigmoid colon

The dataset is available in (.dcm) format.

We have also used open-source data for the evaluation of our model which had 1204 CT images. from which we could use 107 for our application and all of them were male images so, we found some inconsistency for recto-sigmoid.

2.1 Label Issues

The label is only available for a small region where we are interested. Even for the same organ the label is not available outside the interested region. For example, the sigmoid is a large organ but a label is available for a small portion of it in the Z direction. So, our training data can not go outside the range the label is available (as we are training using 2D CNN). We will further discuss the issue later.

2.2 Organ Ranges

All three organs fit in the range (128×128) , and the number of slices the organs occupy is in the range $[18, 60]$, but there is one data which takes 195 slices. So, we can say the number of slices can vary a lot, and we need to take care of this in our training strategies. In the case of 2D CNN we can take images of size $(128 \times 128 \times 1)$ and for 3D CNN we can up-scale or down-scale the z-axis by over-sampling or under-sampling the image

2.3 Region of Interest

The first step was to split the data in train, test, and validation. We used 100 CT images to train, 20 images for validation and 30 for testing.

The data which is available to us has a very large amount of background. We can get very close to the required organs by looking at the bone's structure.

2.3.1 Close to Organ

We need to get the Bladder, Rectum and Sigmoid colon all three are inside the left and right Ilium and Pubis to the end of the sacrum Figure 2.1, We are using bone area and size to track the bones sequentially from Femur, This geometrical approach works fine and detects the desired area accurately, the doctors are also able to change the boundary if they require (this part will be discussed in UI part)

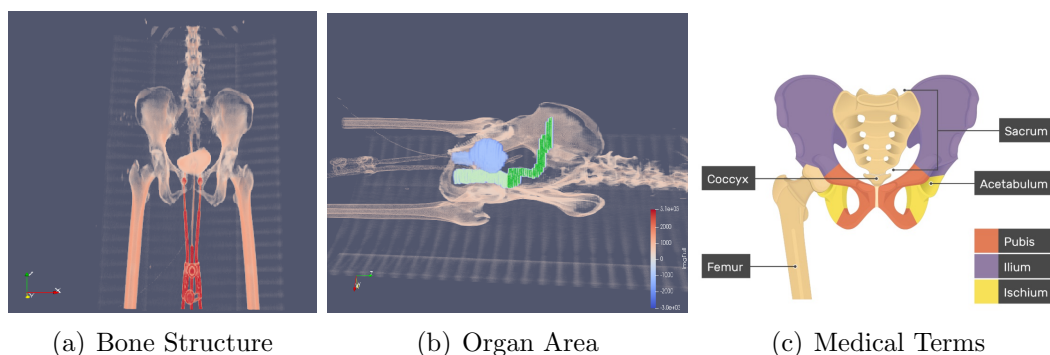


Figure 2.1: Region of Interest

Going closer to the organ helps in many ways:

- Reduce background size and intern reduces the complexity of the image.
- Reduced image size reduces the time to train and predict.
- Reduces noise in prediction.

2.3.2 Removing Unlabelled Slice

The required organs that we need to segment Rectum and Sigmoid are connected organs and very hard to distinguish as they share the internal structure. The difference in name is mainly based on position and geometrical structure. This causes a big issue in two ways:

- If we use a single network with Categorical Cross entropy loss, the network would not be able to distinguish any of the two and the joining place of both will have no category. This can be seen in the following Figure 2.2.
- Again, if we use a single network and don't remove the unlabelled portion of the same organ, the model will get the wrong supervision. For a 2D slice, as it would not have positional information, the model would not be able to train properly to segment the image as it will be confused about when to mark the organ as truth value or when not to. If the number of unlabelled images is the same as the number of labelled images, the model may mark them all as background.

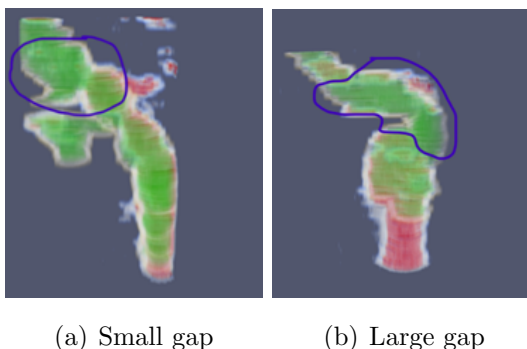


Figure 2.2: Uncertain Junction

To generate the training data, we are using the following ways to remove unlabelled slices for particular organs:

- 1 **Bladder** is not marked fully below or $-Z$ direction, but it is properly marked in the upper direction or $+Z$ direction. So, we started taking slices when the marking started but took the full boundary region or Sacrum length.
- 2 **Rectum** is not marked fully in both direction. i.e. $+Z$ and $-Z$ direction. So, we started taking slices when the marking started and stopped when the marking ended.
- 3 **Sigmoid** also doesn't have marking in both direction. So, we followed the previous strategy.

Chapter 3

Literature Review

Some deep neural network-based models have been used to segment images using different loss functions to suit better the need based on the application. U-Net is among the most popular and works fine for multi-dimensional data.

3.1 U-Net

Currently, we are using U-Net [9] for the segmentation of the images. U-Net is a CNN-based architecture that down-samples the image in a hierarchy using max-pool, which helps it to extract small details or features in the image and then it merges the output from a previous CNN layer and up-sample of the below layer to rebuild the segmented image sequentially Figure 3.1.

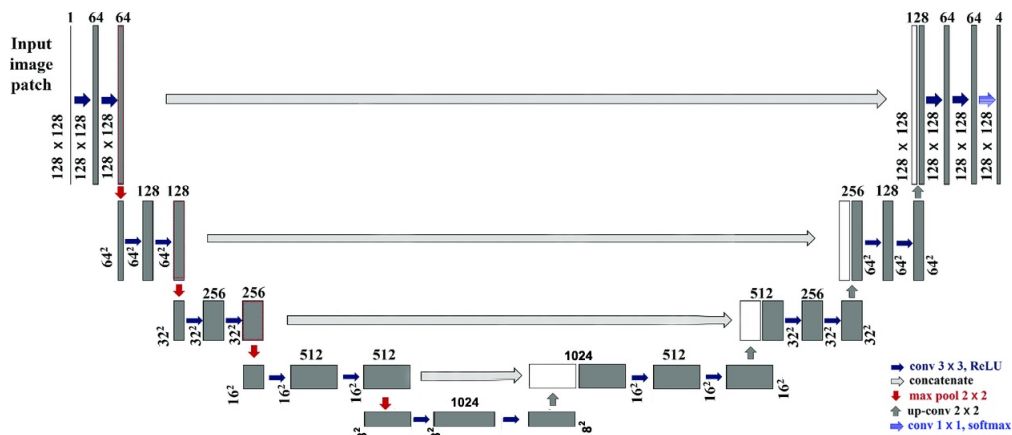


Figure 3.1: U-Net model

U-Net takes $(n \times n \times c)$ as input and gives $(n \times n \times l)$ mask as output representing the segmented layers.

3.1.1 U-Net (2D)

2D U-Net acts as a sliding window. It takes one slice of the CT image and produces segmentation for it. It does not have any information on the previous layer, so it does not have any positional information on the z-axis. This will cause one organ, like the rectum, to continue producing masks even after the sigmoid starts. We will discuss this problem later. The U-Net we used takes $(128 \times 128 \times 1)$ images as input and produces $(128 \times 128 \times 1)$ as output, with total number parameters 34,512,193 We used multiple loss functions to finetune the result, which took in a total of 160 epochs with a batch size of 30

3.1.2 U-Net (3D)

3D U-Net [15] takes the entire organ region at once. So, it contains positional information as well as neighbouring slice information. The U-Net we used takes $(128 \times 128 \times 128 \times 1)$ images as input and produces $(128 \times 128 \times 128 \times 1)$ as output, as we also need at least 10-20 images in a batch to train the GPU memory requirement for this kind of network is so high that currently not feasible (We will discuss this later)

3.1.3 U-Net (ConvLSTM2D)

the ConvLSTM2D [14] can have some neighbouring slice information as well as positional information. So, it can replace 3DCNN. But in practice, this also needs very high GPU memory.)

3.2 LOSS FUNCTIONS

Choosing a good loss function is very important because it greatly affects the output. We initially used tree loss functions to stabilize the weights and then used others to fine-grain results.

BinaryCrossentropy: Binary cross entropy [4] is used initially to stabilize the updates.

$$L_{BCE}(y) = -\frac{1}{N} \sum_{n=1}^N y_i \log(p(y_i)) + (1 - y_i) \log(1 - p(y_i))$$

There are two major problems with this loss function:

- The gradient of this loss function is stable, so the update is also stable, but as it does not include the size of the true value to calculate the loss, if the background is very big, the segmentation for the truth value will not affect the loss that much. In contrast, the loss should be higher if the segmentation is not good for the truth value. Sometimes, it can only produce background or 0 for very small truth values.

- Another problem of this loss function is that the border will have a gradient that is not desired, as we can be confused about the actual organ.

BinaryFocalCrossentropy: [7] This is used to make the segmentation closer to the ground truth. As it uses information like how much is the size of the truth value compared to the background. It achieves this by down-weighting. It adds a modulating factor to cross-entropy loss.

$$L_{BFC}(y) = -\frac{1}{N} \sum_{n=1}^N (1 - p(y_i))^\gamma \log(p(y_i))$$

We trained for 50 epochs with this loss on the pre-trained model we discussed previously.

Dice Loss: [12] This loss function is used for fine-tuning the model. The main idea of this loss is to increase the ratio of True Positive by (Ground truth + Predicted truth).

$$L_{DL}(y) = -\sum_{n=1}^N \frac{2 * TP(y_i)}{TP(y_i) + FP(y_i) + FN(y_i)}$$

The gradient of this loss is not stable. So, if the ground truth is very small in number, the error can be very high for a small number of miss classifications, and also, if the ground truth is very high in number, the change in error will be very less.

We can add constants in the numerator or denominator to stabilise the function. After this, the new loss function will be as follows.

$$L_{DL}(y) = -\sum_{n=1}^N \frac{2 * TP(y_i) + 1}{TP(y_i) + FP(y_i) + FN(y_i) + 1}$$

We can not use this relaxed loss function in the beginning to avoid this issue. We used this to make the border region accurate.

Chapter 4

Methodology

The entire procedure encompasses the reception of an input CT image, preprocessing of the data, organ prediction, post-processing of the predicted organs to generate the final image, and generating an output format compatible with the current workflow Figure 4.1. The step-by-step description of the process is discussed in detail below.

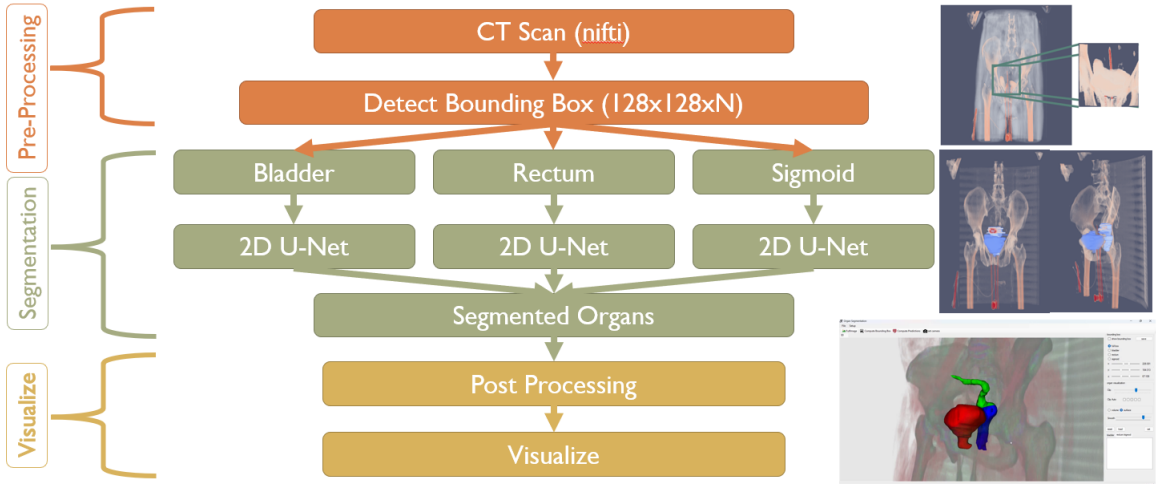


Figure 4.1: Segmentation Framework

4.1 Pre-Process

The pre-processing of the input data takes place in the following sequences:

- Reading Input file (.vtk or .nii.gz or .dcm)
- Getting Bounding Box (Geometric)

- Getting Bounding Box (ML)
- Rectifying the box (by user)

4.1.1 Reading Input file

In order to visualize the image, it is necessary to convert the given image .dcm into the .vtk format [1]. For this purpose, a constant scale of 3 is applied along the z-axis or transverse plane, while the remaining axes have a scale of 1.

4.1.2 Getting Bounding Box (Geometric)

As previously discussed, the fundamental concept of our approach involves utilizing a bounding box technique to approach the organ of interest closely. Furthermore, the geometric approach leverages bone structures as reference points to identify and locate the desired area accurately.

Bone using Isovalue: To get the image of bone from the input image, we are using the isovalue (bone iso-value is > 150 and < 2000). Within the algorithm, we assign a constant value of 1 to the desired object, while considering the background as 0. The isovalue encompassing the object also incorporates any needles or other devices present within the image. However, it is important to note that the algorithm is not designed to handle large objects other than bones. The presence of such objects may introduce confusion, as their placement is not fixed like that of bones.

Components: For every slice along the z-axis, we identify all sizable disconnected components that exhibit sufficient density. The criteria employed to determine the inclusion of a component is based on the following formula:

$$h > \min H \text{ and } w > \min W \text{ and } h < \max H \text{ and } w < \max W \text{ and } \frac{\text{depth}}{h*w} > 0.15$$

where :

$$\min H = 20, \min W = 20, \max H = 140, \max W = 140$$

Starting Slice / Bottom: When the number of bones in a slice is more than 4, we can consider taking slices (transverse plane) as we will have six disconnected bone segments of the femur and os-coxa. We can see the section in the below Figure 4.2 (a).

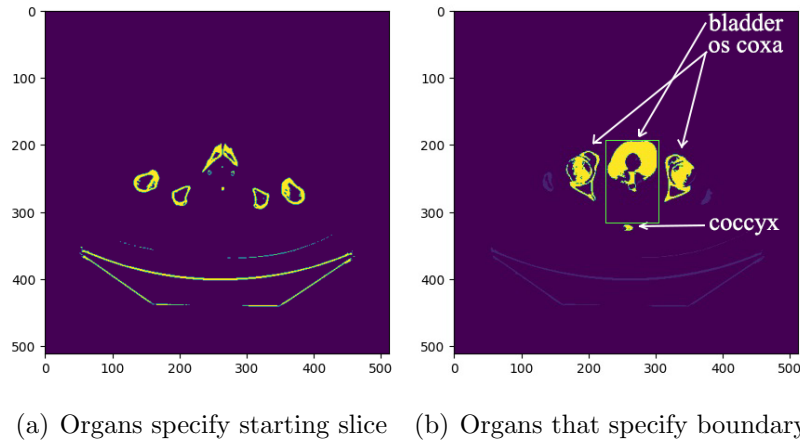


Figure 4.2: Bounding Box

Organs: Subsequently, it becomes necessary to determine the ranges within the sagittal and coronal planes. To achieve this, we can utilize the presence of the os-coxa, bladder, and coccyx as reference points Figure 4.2 (b). It should be noted that the bladder, despite not being a bone, exhibits a notable thickness in its musculature and appears within the previously described isovalue range.

- In the sagittal plane, the range is determined based on the extent of the bladder within that plane. Conversely, for the coronal plane, the range is established from the top of the bladder to the top of the coccyx.
- To locate the bladder, our method initially identifies the two os-coxa bones, which serve as the rightmost and leftmost markers. Subsequently, we identify the component in which the top region lies above the midpoint between the two os-coxa bones, excluding the os-coxa themselves, which represents the bladder. Among all slices of the bladder, we select the slice with the highest top boundary, denoting it as LB (the slice representing the largest bladder).
- To find the coccyx, we use the same LB layer and find any bone at the right of the left os-coxa, left of the left os-coxa and below the bladder. And we also changed the minH and minW to 3 for finding the disconnected components, as the coccyx can be small.

Ending Slice / Top: When we already got the bottom and the number of bones in a slice is less than two, we can consider we have reached the end of our region of interest in the transverse plane as coxa will end and only lumbar disc will be there.

Algorithm 1: BoundingBox(Geometric)

```
input : Image,  $I_{[xn,yn,zn]} = \{I_1, I_2, \dots, I_z\}$ .  
output:  $\{x_{min}, x_{max}, y_{min}, y_{max}, z_{min}, z_{max}\}$   
1 //Initialization  
2      $z_{min} \leftarrow -1$   
3      $z_{max} \leftarrow zn$   
4      $largeOrgans \leftarrow \phi$   
5      $bladderVolume \leftarrow -1$   
6 for each slice,  $I_z \in I$  do  
7      $c \leftarrow Components(I_z)$   
8     if  $z_{min} == -1$  and  $len(c) > 4$  then  
9          $z_{min} \leftarrow z$   
10    end  
11    if  $z_{min} > -1$  then  
12        if  $z_{max} == zn$  and  $len(c) < 2$  then  
13             $z_{min} \leftarrow z$   
14            break  
15        end  
16         $o \leftarrow Organs(I_z)$   
17        if  $o.bladder.size > bladderSize$  then  
18             $bladderSize \leftarrow o.bladder.size$   
19             $large_z \leftarrow z$   
20        end  
21    end  
22 end  
23  $x_{min} \leftarrow largeOrgans.bladder.left$   
24  $x_{max} \leftarrow largeOrgans.bladder.right$   
25  $y_{min} \leftarrow largeOrgans.bladder.top$   
26  $y_{max} \leftarrow largeOrgans.coccyx.top$   
27 return  $\{x_{min}, x_{max}, y_{min}, y_{max}, z_{min}, z_{max}\}$ 
```

4.1.3 Getting Bounding Box (ML)

We create our data set using the labelled data for the ML approach. In the sagittal plane, we mark rectangles bounding the organs, and we get the ranges from the labels provided and

convert the layers to 128x128 image

We used U-Net to predict the boundary from layers and resized the image to its original size. The predicted boundary is not always perfect. To get the rectangle we used *approxPolyDP* using *Ramer–Douglas–Peucker algorithm* [13] from *OpenCV* which uses *Douglas – Peucker* algorithm.

Now we get the coordinates of the Y-axis and Z-axis from the rectangles as we are performing the process on X-axis or (Sagittal plane). We are considering a plane with a bounding box if we get four points from the *Douglas – Peucker* algorithm. We can get y-min, y-max, z-min and z-max from the four points for the particular slice. If we store all valid Y-Z ranges, we will have many outliers, we can remove the outliers using the median absolute distance to the median. After removing outliers we do the average of the remaining points. Below is an example in Figure 4.8.

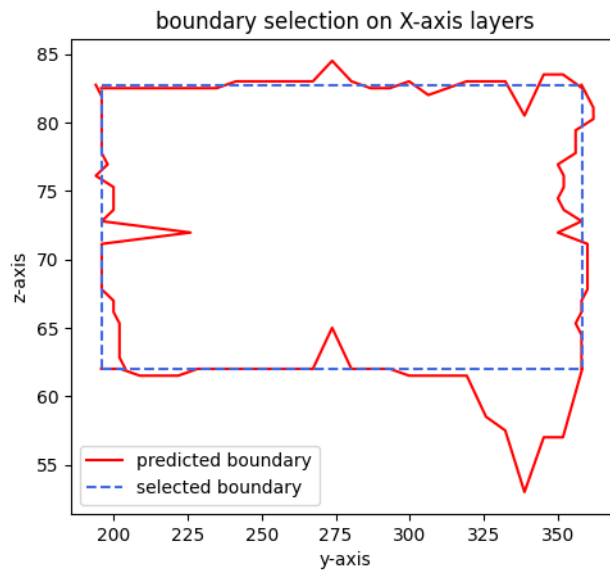


Figure 4.3: Boundary Points and Selected Boundary

We use the same approach for predicting X-axis or sagittal plane range by taking slices in the coronal plane or Y-axis.

The mean absolute error for the ML approach is:

x-min	x-max	y-min	y-max	z-min	z-max
7.32	4.97	4.23	2.83	2.36	1.55

Table 4.1: mean absolute error for the ML approach

The mean absolute error for the Combined result from the *Geometric and ML* approach is:

x-min	x-max	y-min	y-max	z-min	z-max
3.61	3.17	4.23	2.83	2.36	1.55

Table 4.2: mean absolute error for the Geometric & ML approach

4.2 Training of Organ Specific Models

The prediction is the most important part of the entire project, and for this reason, we need to train the model in an organ-specific manner. As we discussed previously, we need to train the models in their range only in the transverse plane for the rectum and sigmoid, and we need to go as close as possible to the organs to make better predictions. This can be done using both geometric and ML algorithms. The process is described in detail in pre-processing phases (4.1.2 and 4.1.3) For training, we calculated the bounding box from the given label ranges. The algorithm for getting the bounding box is in (originalRange.ipynb) and then created training data from the CT images and labels in the ranges from specific organs.

The Models we train are:

1. **Bladder (Transverse Plane or Z axis):** To train our model specifically for the bladder, we extract the entire bounding box encompassing the bladder region. This allows us to focus on the relevant content within that region. The extracted image and corresponding label slices are then resized to a dimension of (128x128). This standardized size ensures consistency during training, as both the images and labels are processed at the same resolution. By utilizing this approach, we are able to effectively train our model for bladder segmentation.

2. **Rectum (Transverse Plane or Z axis):** In the case of the rectum, our approach differs from the bladder. Instead of capturing the entire bounding box, we selectively choose slices that span from the beginning to the end of the rectum in the transverse plane. These selected slices are utilized for training, focusing on the relevant content within them. Similar to the bladder, we resize both the image and label slices to a standardized dimension of (128x128) before training. This consistent resizing ensures uniformity in the training process and facilitates accurate rectum segmentation.
3. **Sigmoid (Transverse Plane or Z axis):** Similarly, when it comes to the sigmoid, our approach involves selecting specific slices from the start to the end of the sigmoid in the transverse plane, rather than considering the entire bounding box. These chosen slices contain the relevant content necessary for training. Like the previous cases, we resize both the image and label slices to a standardized dimension of (128x128) before proceeding with the training process. This consistent resizing allows for uniformity and facilitates accurate segmentation of the sigmoid during training.
4. **Rectum and Sigmoid (Sagittal Plane or X axis):** The visibility and continuity of the entire organ structure are more pronounced when viewed from the sagittal plane. In this plane, we can observe the organ's continuity, which may not be as apparent in the transverse plane. However, in the case of the sigmoid, the available number of slices may be limited, making it impractical to stretch them to a size of 128x128 for training. Consequently, training the rectum and sigmoid as separate organs becomes challenging. To address this, we adopt a different approach. We capture the entire organ bounding box by slicing in the sagittal plane and subsequently resize both the input image and label slices to a dimension of (128x128) for training. This resizing process ensures consistency and enables us to train the organ segments effectively within the limitations of available data.

The training needs to be done in a sequence to fine-tune our model. We are using 128x128 U-Net.

- loss function: BinaryCrossentropy, Samples: 4000, Epochs: 37
- loss function: BinaryCrossentropy, Samples: 1000, Epochs: 100
- loss function: BinaryFocalCrossentropy, Samples: 2000, Epochs: 50
- loss function: Dice Loss, Samples: 400, Epochs: 10

Prediction after BCE has blurred boundaries. It becomes confusing which part is actually inside the region of interest Figure 4.4.

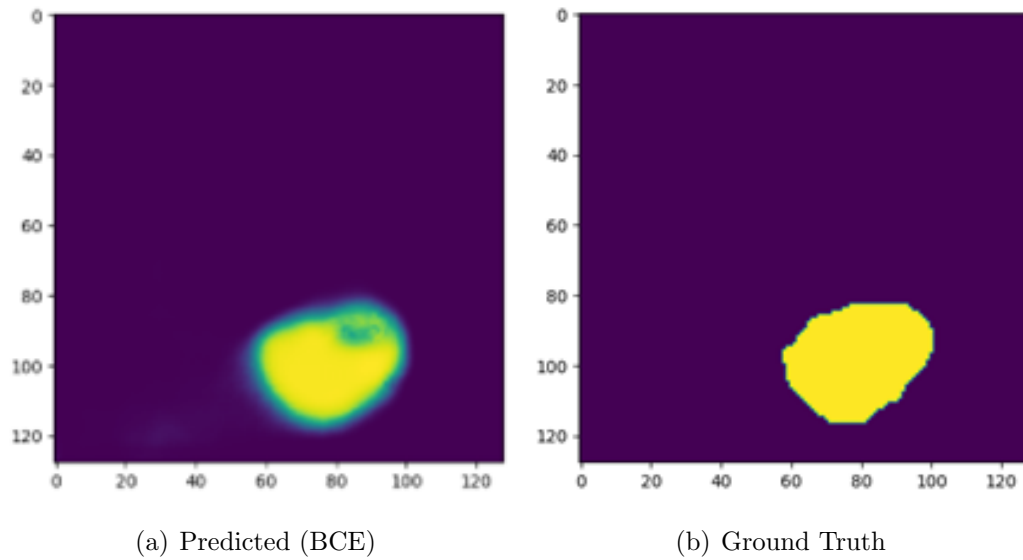


Figure 4.4: Prediction vs. Ground Truth

For final stage, we used Relaxed Dice Loss. The accuracy of border is very high Figure 4.5.

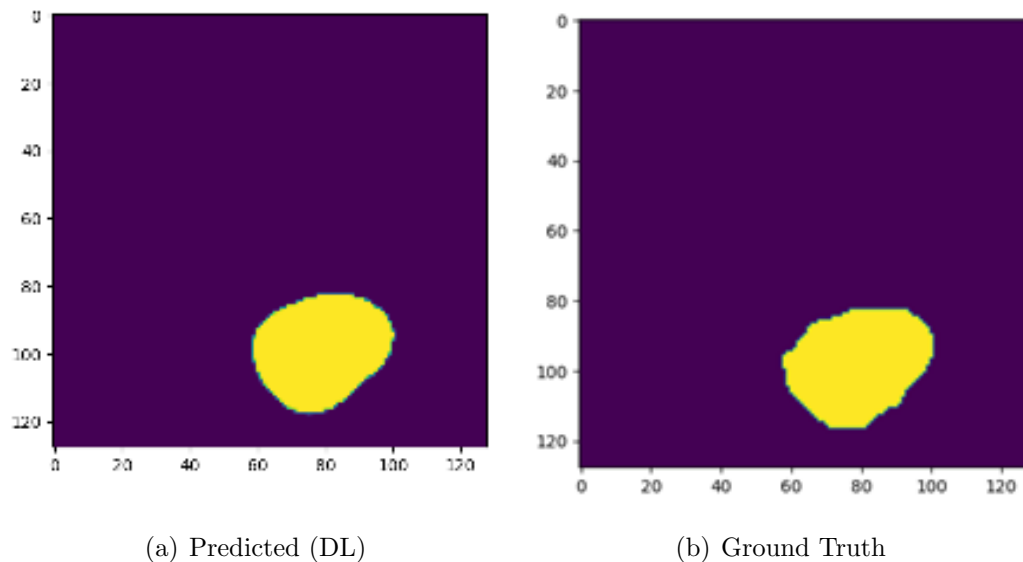


Figure 4.5: Prediction vs. Ground Truth

The training accuracy for bladder in all four stages are following Figure 4.6.

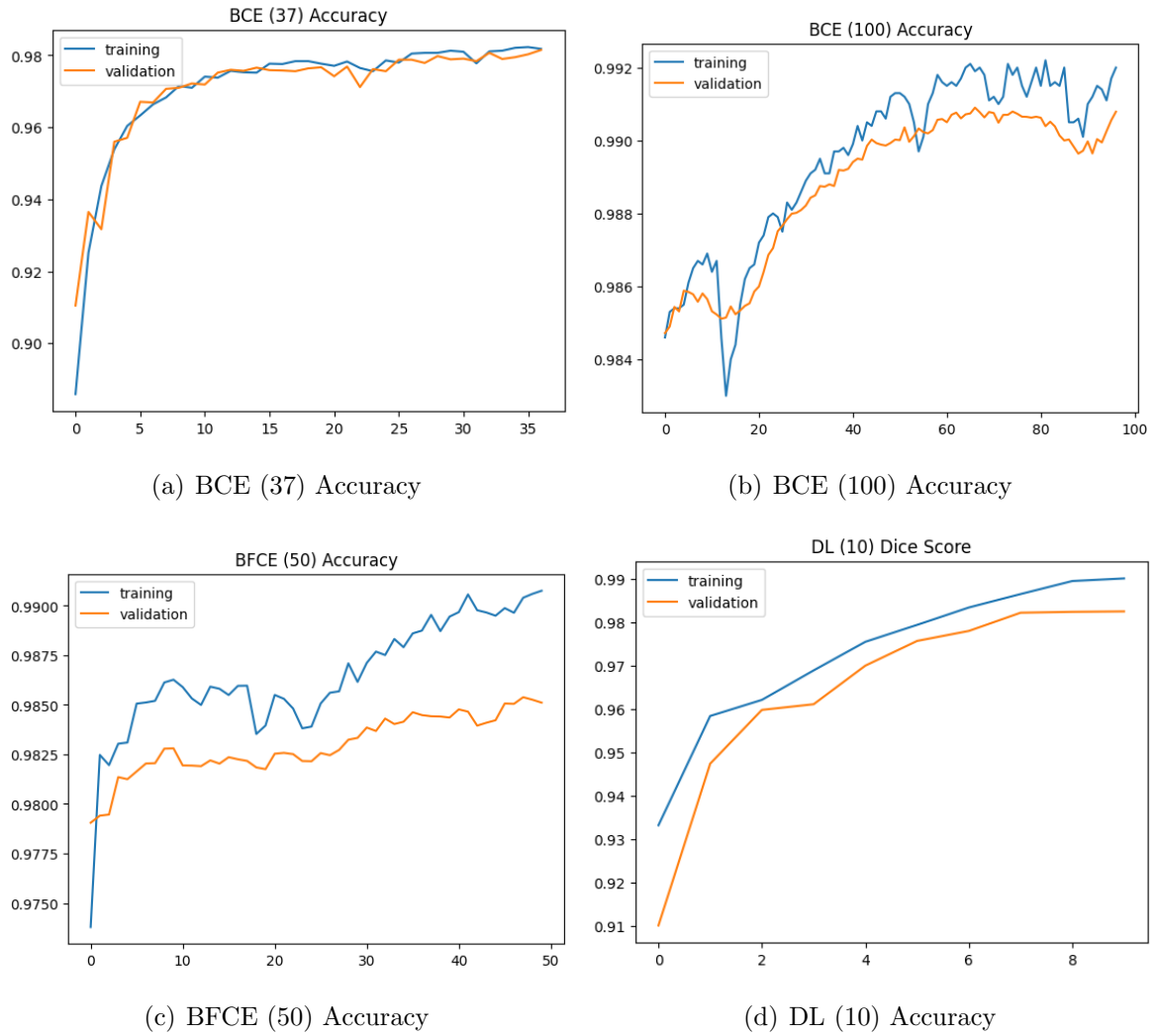


Figure 4.6: Training Accuracy for Bladder

4.3 Prediction of the Organs

In order to predict the organs accurately, it is necessary to obtain their respective bounding boxes and fine-tune them if required. All three organs share the same dimensions (height and width) in the transverse plane (XY plane). However, we can utilize distinct starting and ending slice numbers in the transverse plane to focus specifically on each organ. This approach enables a more targeted analysis of the organs.

The prediction output does not inherently differentiate between the rectum and sigmoid, unless separate boundaries are provided in the transverse plane. Therefore, if the values of z_{Min2} and z_{Min3} are the same, and z_{Max2} and z_{Max3} are also the same, the predictions for the rectum and sigmoid will encompass labels throughout the entire region. However, the post-processing phase allows for the separation of the organs based on the obtained predictions.

To facilitate this process, we employ four models, with one dedicated to the bladder and three dedicated to the rectum and sigmoid. In practice, a single model encompassing the rectum and sigmoid sagittal plane can be used, streamlining the implementation.

4.4 Post Process

For further improvement of the result, we provide two geometric algorithms.

- Removing disconnected components.
- Prediction of Recto-Sigmoid junction.

4.4.1 Removing disconnected components

To address the issue of small glitches or disconnected components that may appear in the 3D predictions, we employ a post-processing step known as removing disconnected components. This step involves identifying and eliminating these isolated or fragmented segments from the prediction results. By removing these disconnected components, we enhance the overall accuracy and cohesiveness of the final segmentation output.

4.4.2 Prediction of Recto-Sigmoid junction

As we discussed previously the prediction stage does not distinguish between rectum and sigmoid junction, so we need to use this post-processing step to do the same, this method uses a geometric algorithm to calculate the required output Figure 4.7.

To find the Recto-Sigmoid junction we can use three methods:

Method 1: Distance from the Sacrum bone.

Method 2: Radius of the organ.

Method 3: Angle from X-Y Plane

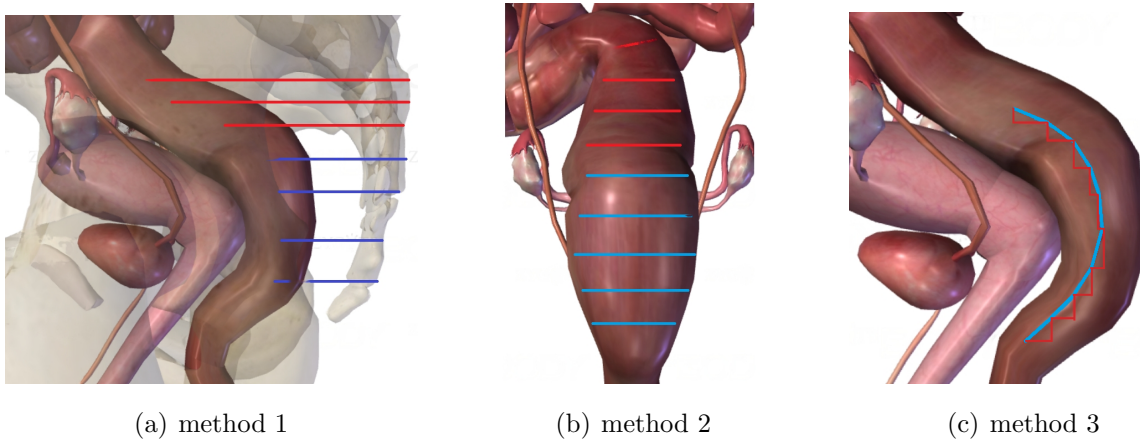


Figure 4.7: Methods to find recto-sigmoid junction

We are using the 3rd method for finding the Recto-Sigmoid junction.

In order to determine the center points of all slices in the transverse plane of the organ, we follow a specific procedure. First, we calculate the average coordinates of all points within each slice. This average serves as the center point for that particular slice. To ensure accurate results, we prioritize selecting the largest disconnected component. By focusing on the largest component, we mitigate the potential issue of a distant disconnected component causing significant displacement of the center point. This approach helps us achieve more precise and reliable center point estimation for each slice.

After determining the center points of two consecutive slices, we proceed to calculate the angle with the XY plane. This angle measurement provides valuable information about the orientation of the slices. When plotting the angles against the corresponding slice numbers, it is possible to observe some jaggedness in the plot, which may be attributed to outliers or irregularities in the data.

Of particular interest is the occurrence of a significant drop in the angle, as it indicates the presence of the Recto-Sigmoid junction. This distinct change in the angle represents the transition point between the rectum and sigmoid regions. By identifying and analyzing these sharp drops in the angle plot, we gain insights into the location and characteristics of the Recto-Sigmoid junction.

In order to enhance the smoothness of the curve, we apply the Savitzky-Golay filter from the SciPy library [10]. This filter helps to reduce noise and irregularities in the angle or curve plot, resulting in a smoother representation of the data.

Additionally, we calculate the rate of change in the angle or curve and store these values. To further refine the data, we apply the Savitzky-Golay filter to the rate of change values as well. By smoothing both the curve and its rate of change, we improve the accuracy and reliability of the analysis.

To determine the junction point, we utilize five different methods. These methods involve examining the processed curve and identifying specific criteria or patterns that indicate the location of the junction point. By employing multiple approaches, we increase the robustness and consistency of our findings regarding the junction point.

Algorithm 2: RectoSigmoidJunction(Geometric)

input : Image, $I_{[xn,yn,zn]}$ of recto-sigmoid
output: $\{z_1, z_2, z_3, z_4, z_5\}$

- 1 $I_c \leftarrow \text{removeSmallComponents}(I)$
- 2 $Slopes \leftarrow \text{slopeOfMidPoints}(I_c)$
- 3 $Slopes_{smooth} \leftarrow \text{savgolFilter}(Slopes)$
- 4 $dSlope \leftarrow \text{discreteSlope}(Slopes_{smooth})$
- 5 $dSlope_{smooth} \leftarrow \text{savgolFilter}(dSlope_{smooth})$
- 6 $z_1 \leftarrow \text{maxDrop}(Slopes)$
- 7 $z_2 \leftarrow \text{maxDrop}(Slopes_{smooth})$
- 8 $z_3 \leftarrow \text{maxDrop}(dSlope_{smooth})$
- 9 $z_4 \leftarrow \text{maxima}(dSlope_{smooth})[0]$
- 10 $z_5 \leftarrow \text{indexOf}(Slopes_{smooth} < 76^\circ)$
- 11 return $\{z_1, z_2, z_3, z_4, z_5\}$

The *mean absolute error* for the closest point from all five algorithms is **1.95**. Below are five samples that use all five approaches, and the closest to the given is selected. The samples show a selection of all five approaches. In the below Figure 4.8 and frequency of selection from the five approaches in Table 4.3.

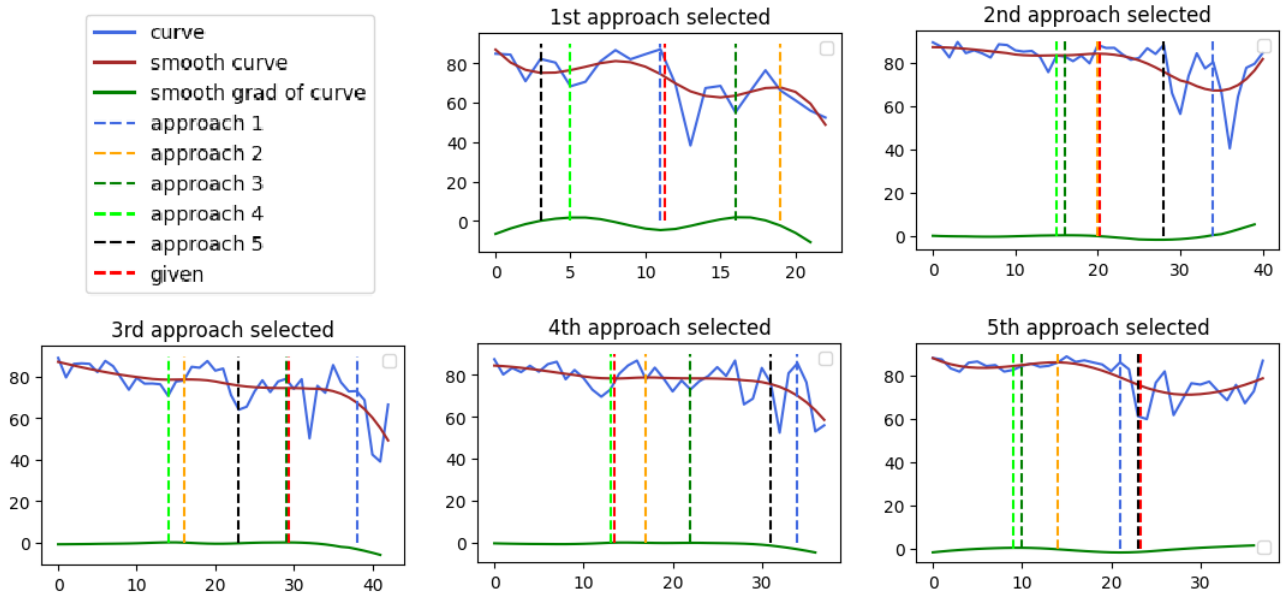


Figure 4.8: Predicted Boundary Points and Selected Boundary

1st approach	2nd approach	3rd approach	4th approach	5th approach
0.16	0.16	0.30	0.11	0.25

Table 4.3: Frequency of the approach being selected

Chapter 5

Evaluation

To assess the performance of the various models, we employed a dataset consisting of 30 CT images. The evaluation was conducted using the DICE Score as the metric. The relaxed DICE Score serves as a quantitative measure to evaluate the accuracy and similarity of the predicted segmentations compared to the ground truth. By utilizing this widely used evaluation metric, we were able to objectively compare and assess the performance of the different models.

In Table 5.1, we present a comparative analysis of different training approaches. This analysis focuses on the inclusion or exclusion of unlabelled slices, as well as the utilization of various loss functions. Specifically, we examine methods that do not incorporate Dice Loss as the final layer. In these methods, additional refinement is applied by considering values exceeding a threshold of 0.4 as 1, resulting in improved sharpness of the predicted output. The DICE Score, ranging from 0 to 1, serves as an evaluation metric, with higher scores indicating superior results.

Table 5.2 presents a comparison of various training stages for different organs. In stage 1 and stage 2, BinaryCrossentropy loss function is utilized. In stage 3, BinaryFocalCrossentropy loss function is employed, while in stage 4, relaxed Dice Loss is utilized. Additionally, we also showcase the Relaxed Dice Score after applying post-processing techniques. The table provides insights into the performance and effectiveness of each training stage for accurate organ segmentation.

Model Comparison	Binary Cross Entropy	Accuracy	DICE Score
Single U-Net without removing unlabelled slices loss: Categorical Cross Entropy	0.1318	0.9512	0.79
Organ wise Separate U-Net without removing unlabelled slices loss: Binary Cross Entropy	0.1018	0.9724	0.89
Organ wise Separate U-Net after removing unlabelled slices loss: Binary Cross Entropy	0.0197	0.9919	0.9882
Organ wise Separate U-Net after removing unlabelled slices loss: Binary Cross Entropy Binary Focal Cross Entropy Relaxed Dice Loss	0.006	0.9937	0.9898

Table 5.1: Compare different training approaches

Organ Slices	Stage1	Stage2	Stage3	Stage4	Post-Process
Bladder z-axis	0.9826	0.9929	0.9822	0.9813	0.9882
Rectum z-axis	0.9826	0.9919	0.9897	0.9886	0.9891
Sigmoid z-axis	0.9904	0.9968	0.9927	0.9899	0.9886
Rectum + Sigmoid x-axis	0.9882	0.9932	0.9911	0.9897	0.9915

Table 5.2: Compare different training stages

Some Interesting 2D outputs: from bladder, rectum and sigmoid Figure 5.1.

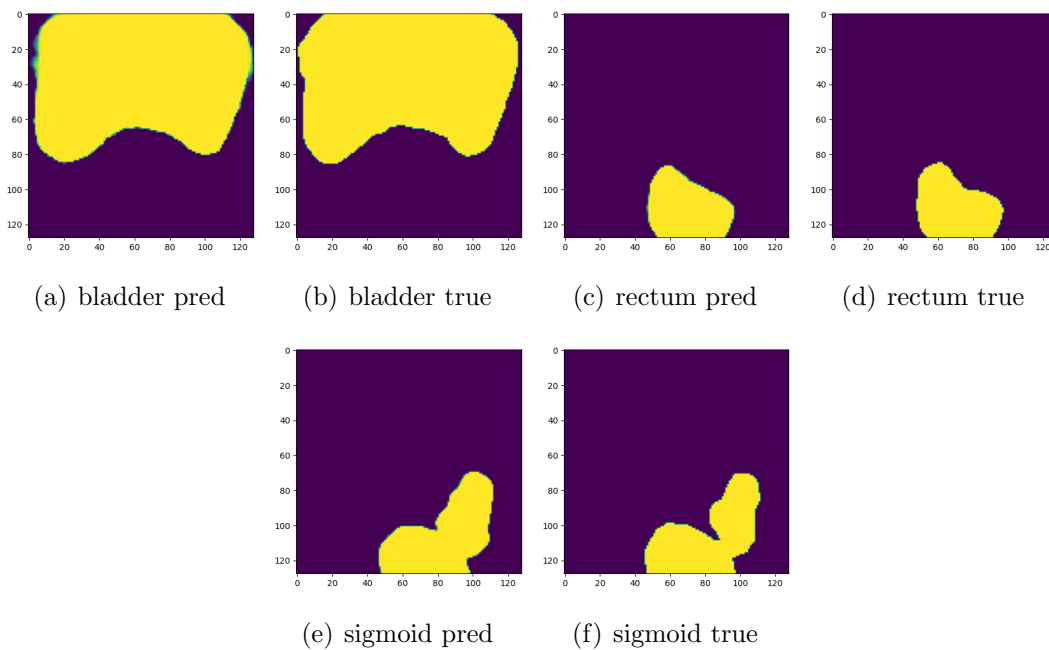


Figure 5.1: Predicted vs Ground Truth 2D

3D Result The blue colour represents ground truth. green: bladder, light red: rectum, dark red: sigmoid Figure 5.2.

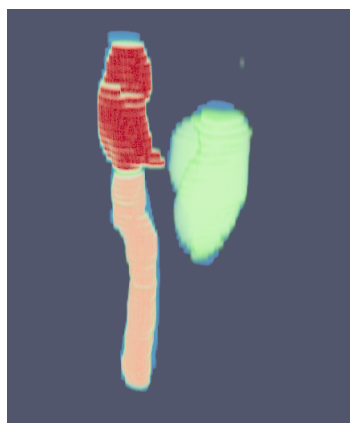


Figure 5.2: Predicted vs Ground Truth 3D

Chapter 6

Implementation and Visualization

We have developed a user-friendly interface that effortlessly helps doctors generate segmentations from exported data obtained through existing software. The segmented data produced by our software can be imported back into the existing software, enabling doctors to perform further processing and analysis easily. The features of the software are below.

6.1 Import CT images

The software allows importing images from three different file formats 1) .dcm, exported from the existing software. 2) .nii.gz, one of the most popular formats in academics and archived data, allows us to import publicly available datasets easily. 3) .vtk or .vti This is one of the most popular formats used in visualization tools. It allows us to import the data faster without any further conversion, as our tool uses this format, making the import faster.

The backend provides two APIs to import .dmm and .nii.gz files.

6.2 Visualize CT Image

After importing the data, our software provides a valuable feature that allows users to visualize the volumetric CT image using VTK (Visualization Toolkit) [6]. This functionality serves as a crucial guideline, providing a comprehensive view of the CT image and assisting in subsequent processes. By leveraging VTK's visualization capabilities, users can gain insights, analyze the data, and make informed decisions for further steps in their workflow.

6.3 Region of Interest

The software predicts two possible regions of interest and shows the predicted region of interest (which can then be modified by the user if needed). This allows the flexibility to choose the bounding box, relaxing this step so that the accuracy of the next prediction step does not depend on the accuracy of this step.

The backend provides an API that predicts two possible bounding boxes, one using the geometric approach and the other using the ML approach.

6.4 Prediction

The prediction step generates segmented organs along with essential metadata, including the number of connected components within the organs and the slope of the tubular structure, specifically in the recto-sigmoid region. This valuable metadata is crucial for subsequent processing, enabling more informed and effective analysis and decision-making.

The backend provides an API that predicts all three organs and saves the files for visualization and post-processing (6.5 and 6.6).

6.5 Recto-Sigmoid junction

The UI helps select one of the five recto-sigmoid junction points produced by the five algorithms. Also, the UI allows the user to move the clipping plane and choose the best junction point freely (NoteTo adjust the slicer plane, it is moved along the z-axis or transverse plane. The centre and slope parameters are then chosen from the auto-generated options derived from the previous step.)

6.6 Connected Components

Through our user interface, we can select the connected components of the predicted organs which we wish to retain. This important step minimises small glitches or inaccuracies, resulting in cleaner and more refined segmentation Figure 6.1 (b).

6.7 Surface Visualization

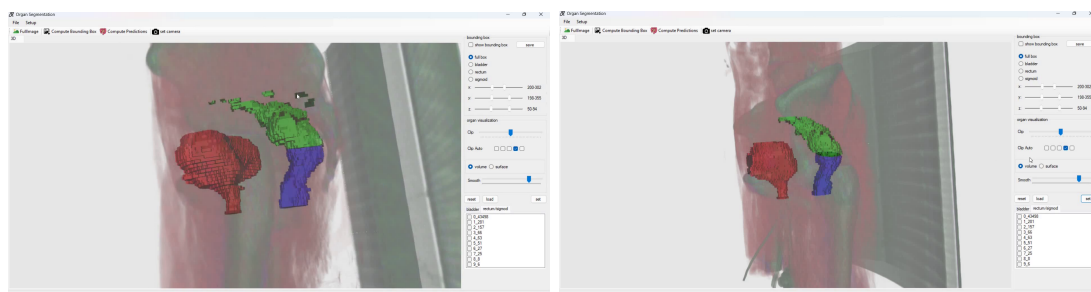
The software displays segmented organs as seamless surfaces, enhancing the ease with which doctors can visualize and comprehend the segmentation. This smooth surface representation

greatly facilitates the interpretation of the segmented organs, enabling medical professionals to analyze and evaluate the results efficiently Figure 6.1 (c) (d).

6.8 Export Segmentation

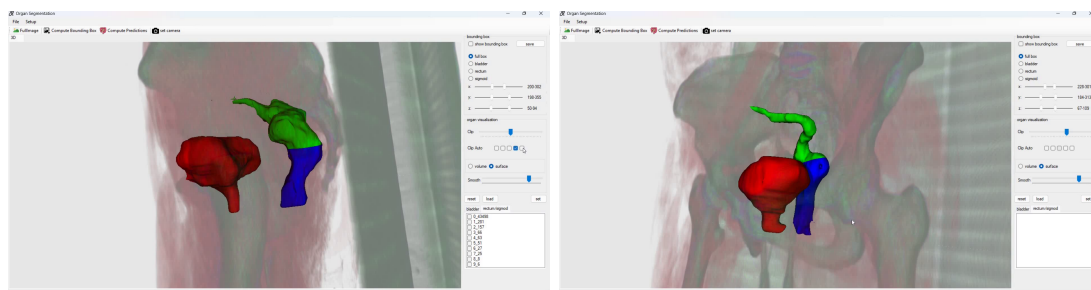
The software allows exporting the result in all three formats, which can help to use the software seamlessly.

The backend provides an API that can export organs in three formats (.vtk or .nii.gz or .dcm)



(a) Prediction Output

(b) Removed Small Components



(c) Surface Visualization 1st

(d) Surface Visualization 2nd

Figure 6.1: Predicted vs Ground Truth 2D

The **detailed description** of the visualization tool can be found in the [repository](#).

Chapter 7

Conclusions and Future Work

The conclusion of this report represents a complete tool that enables doctors or medical professionals to segment organs with minimal interaction for the purpose of Brachytherapy for cervical cancer.

- **Fast Segmentation:** The incorporation of 2D U-Net architecture enables remarkably fast predictions, typically taking only 3-4 seconds to process. This accelerated prediction speed greatly enhances the efficiency and responsiveness of our system, allowing for swift and timely outcomes.
- **Targeted Organ Specific:** Incorporating the identification of the region of interest as a pre-processing step significantly enhances the accuracy of the final prediction.
- **Ignoring Topographic Organs:** The inclusion of the region of interest during pre-processing significantly improves the accuracy of 2D U-Net. However, it is important to note that the CNN output does not inherently possess knowledge of topographical segmentation for organs such as the rectum and sigmoid. Consequently, additional post-processing steps utilizing geometric algorithms were necessary to refine and enhance the output. These supplementary algorithms play a crucial role in achieving more precise and reliable segmentation results, particularly for organs with complex anatomical structures.
- **2D U-Net:** Due to the limited number of available CT images, specifically only 100 3D training images, we opted to utilize 2D U-Net. Although employing 3D U-Net has the potential to significantly improve the results without the need for post-processing steps or organ-specific models. However, due to the scarcity of training data, 2D U-Net proved to be a more viable option. Despite the absence of a three-dimensional perspective, 2D U-Net still offers effective segmentation capabilities within the given region of interest [5].

- **Evaluation:** Our accuracy has been validated through preliminary evaluations conducted by doctors from a specific hospital. This positive feedback from medical professionals confirms the reliability of our system. Furthermore, our accuracy has been further verified by testing it on new CT images, which serves as an additional validation of our system's effectiveness. The consistent validation from medical experts and successful testing on new CT images reinforce the accuracy and robustness of our approach.
- **System Requirement & Integration:** During our testing, we determined that the minimum system requirements for our application were an Intel i3 processor, 4GB of system memory, and Windows 10 operating system. However, for optimal usability and performance, we recommend having at least a 4-core CPU with 8GB of system memory. While a GPU is optional, it is important to note that our predictions can be obtained within 3-4 seconds even without GPU acceleration. However, if you choose to utilize a GPU, we recommend a minimum V-RAM capacity of 8GB. This configuration ensures smoother operation and improved efficiency when working with our application.

In the project's subsequent phase, we focus on developing a path-planning algorithm specifically designed for needle insertion planning and simulation in Brachytherapy. The algorithm will aim to optimize the process of determining the ideal path for needle insertion, enhancing precision and accuracy in Brachytherapy procedures.

Bibliography

- [1] Brian M. Anderson, Kareem A. Wahid, and Kristy K. Brock. Simple python module for conversions between dicom images and radiation therapy structures, masks, and prediction arrays. *Practical Radiation Oncology*, 11:226–229, 5 2021. ISSN 18798500. doi: 10.1016/j.prro.2021.02.003. [12](#)
- [2] Mark G. Bandyk, Dheeraj R Gopireddy, Chandana Lall, K. C. Balaji, and Jose Dolz. Bladder segmentation based on deep learning approaches: current limitations and lessons. 1 2021. [2](#)
- [3] Kris Derks, Jacco L.G. Steenhuijsen, Hetty A. van den Berg, Saskia Houterman, Jeltsje Cnossen, Paul van Haaren, and Katrien De Jaeger. Impact of brachytherapy technique (2d versus 3d) on outcome following radiotherapy of cervical cancer. *Journal of Contemporary Brachytherapy*, 10:17–25, 2018. ISSN 1689-832X. doi: 10.5114/jcb.2018.73955. [2](#)
- [4] .Usha Ruby Dr.A. Binary cross entropy with deep learning technique for image classification. *International Journal of Advanced Trends in Computer Science and Engineering*, 9: 5393–5397, 8 2020. ISSN 22783091. doi: 10.30534/ijatcse/2020/175942020. [9](#)
- [5] Akshay Pai Erik Bjørnager Dam and Christian Igel Mathias Perslev. One network to segment them all, 2019. [30](#)
- [6] W. Schroeder et al. The visualization toolkit, 5th edition. kitware, inc., 2006. [27](#)
- [7] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. 8 2017. [10](#)
- [8] Umesh Mahantshetty, Shivakumar Gudi, Roshni Singh, Ajay Sasidharan, Supriya (Chopra) Sastri, Lavanya Gurram, Dayanand Sharma, Selvaluxmy Ganeshrajah, Janaki MG, Dinesh Badakh, Abhishek Basu, Francis James, Jamema Swamidas, Thayalan Kuppuswamy,

BIBLIOGRAPHY

- and Rajendra Bhalavat. Indian brachytherapy society guidelines for radiotherapeutic management of cervical cancer with special emphasis on high-dose-rate brachytherapy. *Journal of Contemporary Brachytherapy*, 11:293–306, 2019. ISSN 1689-832X. doi: 10.5114/jcb.2019.87406. [2](#)
- [9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. 5 2015. [8](#)
- [10] Abraham. Savitzky and M. J. E. Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, 36:1627–1639, 7 1964. ISSN 0003-2700. doi: 10.1021/ac60214a047. [22](#)
- [11] Janusz Skowronek. Current status of brachytherapy in cancer treatment – short overview. *Journal of Contemporary Brachytherapy*, 9:581–589, 2017. ISSN 1689-832X. doi: 10.5114/jcb.2017.72607. [2](#)
- [12] Carole H Sudre, Wenqi Li, Tom Vercauteren, Sébastien Ourselin, and M. Jorge Cardoso. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. 7 2017. doi: 10.1007/978-3-319-67558-9_28. [10](#)
- [13] M. Visvalingam and J. D. Whyatt. The douglas-peucker algorithm for line simplification: Re-evaluation through visualization. *Computer Graphics Forum*, 9:213–225, 9 1990. ISSN 0167-7055. doi: 10.1111/j.1467-8659.1990.tb00398.x. [15](#)
- [14] Fan Xu, Haoyu Ma, Junxiao Sun, Rui Wu, Xu Liu, and Youyong Kong. Lstm multi-modal unet for brain tumor segmentation. pages 236–240. IEEE, 7 2019. ISBN 978-1-7281-2325-7. doi: 10.1109/ICIVC47709.2019.8981027. [9](#)
- [15] Özgün Çiçek, Ahmed Abdulkadir, Soeren S. Lienkamp, Thomas Brox, and Olaf Ronneberger. 3d u-net: Learning dense volumetric segmentation from sparse annotation. 6 2016. [9](#)