

UNIVERSITY OF OKLAHOMA
GRADUATE COLLEGE

AUTOMATIC VERTEBRAL BODY SUV EXTRACTION OF LOW DOSE
FLT-PET/CT SCANS FOR HSCT PATIENTS

A THESIS
SUBMITTED TO THE GRADUATE FACULTY
in partial fulfillment of the requirements for the
Degree of
MASTER OF SCIENCE

By
LUCAS J. POWERS
Norman, Oklahoma
2025

AUTOMATIC VERTEBRAL BODY SUV EXTRACTION OF LOW DOSE
FLT-PET/CT SCANS FOR HSCT PATIENTS

A THESIS APPROVED FOR THE
SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY THE COMMITTEE CONSISTING OF

Dr. Joseph P. Havlicek, Chair

Dr. Choon Yik Tang

Dr. Samuel Cheng

Acknowledgements

My first and foremost thanks go to Dr. Havlicek for his patience and invaluable guidance teaching me about the world of research and academia. The ‘story’ presented in this thesis would not have been possible without him. Thank you, committee members, Dr. Tang and Dr. Cheng for your time willing to review my thesis.

I would like to extend my sincerest thanks to Dr. Jennifer Holter-Chakrabarty, MD of the Stephenson Cancer Center, University of Oklahoma Health Sciences Center (OU HSC), Dr. Kirsten Williams, MD of Children’s Healthcare Atlanta, and Emory University Winship Cancer Institute for the acquisition of the CT/PET images used in this project. I am particularly grateful to Dr. Holter for providing information regarding the medical aspects of this project.

I would also like to thank Chuong Nguyen for his pioneering work in creating the initial annotations that inspired me to make my own labels. Thank you, Brandon Carson, for taking the time to meet with me and get me up to speed in the early stages of my research. Thank you, Favio Hurtado, for providing a baseline in MONAI that I built my training code on top of. My initial steps into training networks would have been more difficult without this tool. Special thanks go to my parents, John and Wende, who never doubted in me for a moment. I love you both. Finally, I would like to extend my gratitude to Justus Sommer for his invaluable support and thoughtful review of this thesis.

The National Institute of Health (NIH) funded this project under contract number HHSN261200800001E.

Abstract

In this thesis, I present a novel, fully automated framework for multi-class vertebral body segmentation and standardized uptake value (SUV) extraction in low-dose FLT-PET/CT scans of hematopoietic stem cell transplantation (HSCT) patients. The approach combines a custom-made, attention-gated 3D U-Net with label augmentation strategies derived from publicly available datasets, specifically designed to address the challenges of coarse axial CT resolution and limited annotated clinical data. Quantitative evaluations using the Dice similarity coefficient and SUV error analysis, along with qualitative assessments via 3D renderings and difference maps, demonstrate high concordance between automated SUV measurements and physician-derived ground truths—even in anatomically variable regions and under modest segmentation accuracy. The model achieved an average Dice score of 0.83 across all validation cases and a mean SUV correlation exceeding 0.95. Unlike previous methods, this pipeline reliably segments spines exhibiting anomalies such as a lumbarized S1 vertebra, which tend to ‘confuse’ existing networks and result in underestimation of vertebral count. The pipeline offers a clinically actionable approach for identifying and monitoring metabolic changes in vertebral marrow cavities, facilitating earlier detection of engraftment success or potential relapse. In contrast to existing multi-step or partially automated workflows, this end-to-end solution demonstrates stable performance across diverse anatomical conditions. To the best of my knowledge, this is the first work to directly segment all 24 individual vertebral bodies in CT images with 5 mm axial slice spacing. The results underscore the feasibility of extracting accurate SUV data from under-sampled FLT-PET/CT scans and provide a foundation for future research into robust, patient-specific relapse prediction models.

Table of Contents

chapterAcknowledgementsiv

List Of Tables **viii**

List Of Figures **ix**

1 Introduction **1**

2 Background **5**

2.1	Anatomical and Medical Terminology	5
2.1.1	Anatomy of the Human Spine	5
2.1.2	Common Spinal Anomalies	6
2.1.3	Hematopoietic Stem Cell Transplantation	7
2.2	Medical Imaging	8
2.2.1	Computed Tomography	8
2.2.2	Positron Emission Tomography	9
2.2.3	Orientations and Coordinate Systems	11
2.2.4	Image Quality	13
2.2.5	File Formats	14
2.3	Machine Learning and Neural Networks	16
2.3.1	Artificial Neural Networks	16
2.3.2	Convolutional Neural Networks	18
2.3.3	Training Neural Networks	20
2.3.4	Transformations, Patch Based Training	22
2.3.5	Regularization	24
2.4	Image Segmentation	25
2.4.1	Traditional Image Segmentation	26
2.4.2	Performance Metrics	27
2.4.3	CNNs for Images	30
2.4.4	CNNs for Medical Image Segmentation	31
2.4.5	Attention in Deep Learning	33
2.4.6	Vertebral Body Segmentation of HSCT Patients	35
2.4.7	VerSe Dataset	37
2.4.8	Tools, Libraries, and Frameworks for Machine Learning	39
2.4.9	Related Work	40

3	Methodology	44
3.1	HSCT dataset	45
3.2	VerSe Augmentation	48
3.3	Attention-Gated UNet	57
3.4	Model Training	59
3.5	SUV extraction	63
4	Results	69
4.1	Model Performance	71
4.1.1	Quantitative Results	71
4.1.2	Qualitative Results	75
4.1.3	Total Segmentator	80
4.2	SUV Extraction	86
4.2.1	Limitations	89
5	Conclusion	93
5.1	Original Contributions	94
5.2	Recommendations for Future Work	96
	Appendix	108
1	Appendix A	108
2	Appendix B	114

List Of Tables

2.1	The Confusion Matrix	28
3.1	Estimated parameter count for several notable segmentation models. . .	58
4.1	Average Dice scores grouped by anatomical region, supplemented by estimated standard deviation from the mean.	71
4.2	Abbreviated results of TS on the same validation set.	81
4.3	Error metrics after FLT-PET SUV extraction of a sample patient. . . .	87
4.4	Average absolute difference between manual ground truth annotations and model predictions for SUV extraction on validation training set. . .	88
A.1	Individual vertebral body Dice scores for my work compared to Total Segmentator. Highest avg scores are in bold.	109
A.2	Set of error metrics after extracting FLT-PET SUV data from ROIs compared to physician measurements. All segmentations derive from images not present during training. Test cases are marked with *, all others are validation.	111

List Of Figures

2.1	An overview of spinal anatomy in humans.	6
2.2	Patients exhibiting 'additional' (a) and 'missing' (b) vertebrae. [3] . . .	7
2.3	Overview of CT operation [13]	9
2.4	ACD illustration. When the ring of detectors (blue) surrounding the patient detect two nearly simultaneous photon collisions, they trace the approximately linear path back to where the initial interaction happened. This is the foundation of PET imaging. [13]	11
2.5	A representation of the three anatomical planes. [30]	12
2.6	World and image coordinates. [31]	13
2.7	The DICOM standard hierarchy. [34]	15
2.8	Illustration of an artificial neuron. The processing information is stored in the numerical weights $w_0 - w_n$, the tuning of which can be considered learning. [44]	17
2.9	The basic structure of an artificial neural network. Each circle is an abstraction of an artificial neuron (see Fig. 2.8) [44]	18
2.10	Trends related to network capacity. As a model becomes more complex, the risk of overfitting increases. [40]	23
2.11	How batch and group norm are affected by batch size. Each point is the same model trained with a different batch size. The error was measured on the same validation set. [56]	25
2.12	Visual representation of applying the Sobel operator for edge detection. [59]	28
2.13	The 2D UNet architecture proposed in [70].	32
2.14	Attention gate mechanism, extracted from [78].	34
2.15	Images from the VerSe dataset showing the high degree of variability. Extracted from [86].	38
3.1	A sagittal slice of a typical patient in the dataset and segmentation labels.	48
3.2	Example slice of connected component algorithm employed on a sample VerSe image. An opening operation on the thin pedicles separate the vertebral body from the rest of the spine.	54
3.3	The same slice as in Fig. 3.2 with the final mask overlaid in white on the CT image.	55
3.4	A VerSe sample in its native spacing with unmodified labels overlaid on top. The red bar indicates axial slice, and yellow indicates sagittal. . .	56
3.5	The same VerSe image downsampled to 5 mm axial spacing, with the vertebral body intact and the rest of the spine annotation removed. . .	56
3.6	Diagram featuring my custom Attention 3D UNet architecture. N_c represents the number of classes.	59

3.7	Example batch of patch-based training, showing various augmentations like flips (3) and Gaussian noise (4) and a foreground to background ratio of roughly 3:1.	63
3.8	Typical correlation coefficient trends. Extracted from [124].	66
3.9	Subsequent FLT-PET scans showing an increase of cell proliferation from left (day -1) to right (day +28). There is also visible metabolism in the sternum and pelvis, which are explored in other works like [83, 85]. . .	68
4.1	Training loss and validation metric curves when training on only 22 HSCT images, then again with VerSe samples included in a separate training run.	74
4.2	Per-class validation dice score averaged over all 4 samples as a heatmap. This shows the validation performance over time during training.	75
4.3	3D view of sample predictions from the validation set.	76
4.4	Concatenated per-class probability map of model output after softmax, depicting spatial confidence from 0 (no confidence) to 1 (pure certainty).	78
4.5	3D volumetric difference map of my model's predictions on P08D1.	79
4.6	Total Segmentator's prediction on P08D3. The 2D slice plane shows the density of the failed segmentation. Red depicts incorrect predictions, gray depicts correct.	84
4.7	3D volumetric difference map of my model's prediction on validation sample P08D3 showing the slice plane.	85
4.8	Box plots depicting the FLT-PET SUV distributions across vertebral body ROIs (C7–L5) on the 28th day post-transplant.	86
4.9	A qualitative comparison of FLT uptake trends of Williams et al. and my proposed model.	89
4.10	A sample co-registered CT (left) and PET (right) scan with the manually annotated label overlaid on both. Basic interpolation causes clear mismatches in the PET image.	91
4.11	FLT-PET visualization of vertebral ROIs illustrating HSC metabolic progression from day -1 (left) to +28 (right) post-transplant. It ranges from 20th percentile (dark blue) to 99th percentile (dark red).	92
A.1	Training my custom UNet over the same 22 HSCT images but with double the iterations. This further demonstrates the model's tendency to underfit without data supplementation.	112
A.2	Sample test Patient 7. SUV distribution on Day -1.	112
A.3	Sample test Patient 7. SUV distribution on Day +5.	113
A.4	Sample test Patient 7. SUV distribution on Day +28.	113
B.1	3D visualization of my model's CT segmentation on test sample 15, which was not included in training or validation. Day -1 (left) to day +28 (right).	114

B.2	3D visualization of my model’s CT segmentation on test case 21, which was also not included in training or validation. Day -1 (left) to day +28 (right).	115
B.3	3D rendering of an original VerSe manual annotation (left) and modified label (right).	116
B.4	A second 3D rendering that demonstrates my augmentation algorithm on the VerSe labels.	117
B.5	FLT-PET visualization of vertebral ROIs illustrating HSC metabolic progression on day -1 (left), day +5 (middle), and day +28 (right) on a validation case. It ranges from 20th percentile (dark blue) to 99th percentile (dark red).	118
B.6	FLT-PET visualization of vertebral ROIs illustrating HSC metabolic progression on day -1 (left), day +9 (middle), and day +28 (right) on a test case. It ranges from 20th percentile (dark blue) to 99th percentile (dark red).	119
B.7	3D volumetric difference map on my model’s predictions of P10D1. . .	120
B.8	3D volumetric difference map on my model’s predictions of P12D2. . .	121
B.9	3D volumetric difference map on my model’s predictions of P02D1 showing the slice plane.	122
B.10	3D volumetric difference map on Total Segmentator’s prediction of P02D1 showing the slice plane	123

Chapter 1

Introduction

Hematopoietic Stem Cell Transplantation (HSCT) is a critical treatment for patients with blood-related malignancies and immune disorders. While this operation can be curative, detecting graft failure and early signs of relapse are crucial challenges faced by clinicians. Current clinical practice for assessing engraftment has relied on invasive single aspirate biopsies, a small sampling limited to one area—usually the iliac crest, located in the pelvis. However, this sampling may miss the emergence of disease elsewhere in the body. Recent advances in nuclear imaging have raised the possibility of evaluating hematopoietic activity in bone marrow cavities across the entire skeleton using Positron Emission Tomography (PET) scans. While [^{18}F]-Fluorodeoxyglucose (FDG) is the most commonly used radiotracer for PET imaging, [^{18}F]-Fluorothymidine (FLT) has been shown to outperform “FDG” at discriminating true hematopoietic metabolism from background glucose uptake. While “FLT” provides better biological contrast and potential for imaging metabolism of marrow cavities, most research protocols still restrict analysis to 5–7 sentinel bones due to the intensive labor cost. This a compromise that leaves large marrow regions unexamined.

These images are typically taken at the same time (co-registered) as Computed Tomography (CT) scans. While CT scans typically make it much easier to highlight different bone structures, the HSCT patients used in my research are required to have a limited radiation dose, which is accompanied by thick axial slice spacing. Typically in co-registered PET/CT scans, the CT image can be used to segment the bones which

can then be propagated to the PET scan for further analysis. However, in the case of these specific patients, the axial spacing is sometimes thicker than the boundaries between vertebrae, which makes it difficult for even physicians to delineate between the bones. In this thesis, I aim to address these challenges by presenting a fully automated, end-to-end framework to assist physicians in detecting and measuring the SUV uptake of the marrow cavities of individual vertebral bodies in low-dose FLT-PET/CT images.

I plan to accomplish this using a custom, lightweight, Attention-gated UNet variant that can robustly segment individual vertebral bodies in the CT images, boot-strapped by an automated label-rewrite tool that adapts a high-resolution dataset to thick-slice imagery. Then, I propagate the labels to the PET space and automatically extract SUV data and compare it to clinically-verified measurements.

By collapsing hours of manual annotation and contouring into a few seconds of algorithmic inference, this pipeline takes a large step towards practicality of full-body FLT-PET measurement for real-time graft-monitoring and relapse prediction. The proposed method consists of the following:

- **Leveraging Data Augmentation through VerSe:** Due to the very small amount of images present in this dataset (36 scans of 22 patients), along with the high complexity of segmenting vertebral boundaries, I propose a method to systematically augment existing manual annotations on publicly available datasets to fit the problem. I selected the Vertebral Segmentation (VerSe) dataset for my experiment. These images and labels are very high quality, often considered the ‘gold standard’ in segmentation. However, the labels contain the posterior processes of the spine. Since these regions are not of clinical relevance to SUV analysis, I created an algorithm to separate the vertebral body from the processes automatically. The method takes advantage of the thin pedicles connecting the two regions, along with a predictable anatomical pattern by using connected

component analysis and expected centroid placement. The final result is labels that closely mimic our images, while also introducing beneficial complexity to help the network learn generalizable features, not just memorizing the training data.

- **Development of a Custom Attention-Gated 3D UNet Architecture:**

Building upon foundational neural networks such as the UNet and Attention mechanism, I created a novel, custom 3D UNet variant that is tailored to our dataset. The originality includes attention gates, dense skip connections, residual units, and group normalization over the traditional batch norm.

- **Anatomical Variability:** Due to a significant number of our patients exhibiting spinal anomalies, I needed to make a model that was robust to these. Lumbarization of the S1 vertebra (referred to as L6 in this work) can make it appear like there is an ‘extra’ vertebra in the spine. Initial attempts with model training revealed it to get ‘confused’ by this anomaly. Consequently, I added cases from the VerSe dataset that include patients with L6 present. That, and labeling such cases in my specific dataset, provided enough material for the model to learn to predict this bone, making it have higher accuracy in the lumbar. This feature can also be very easily extended to sacralized (also called fused) vertebrae.

- **Quantitative and Qualitative Analysis of SUV ROIs:** I created a visualization tool in Python that can realize the SUV distribution in the segmented vertebral bodies. The tool uses translucent isosurfaces, allowing you to get an idea of cell proliferation patterns present in any given FLT-PET image.

This thesis is organized as follows. Chapter 2 provides a broad foundation for the work, covering anatomical definitions, medical imaging fundamentals, machine learning and neural networks as applied to imaging, segmentation techniques, and an overview of

the datasets and tools relevant to this project. Chapter 3 then details the methodology, including the procedures for manual annotation, augmenting VerSe images, designing and training the proposed model, and measuring PET SUV regions of interest (ROIs) based on CT segmentations. It also explains the mathematical metrics used to assess alignment with physician-derived measurements. Chapter 4 presents the results of the proposed method along with extensive qualitative and quantitative analysis. In addition, it compares this method’s performance against a current state-of-the-art public model and explores key limitations in the current approach. Chapter 5 concludes the thesis by outlining the major original contributions and suggesting directions for future research that can further build upon the findings reported here.

Chapter 2

Background

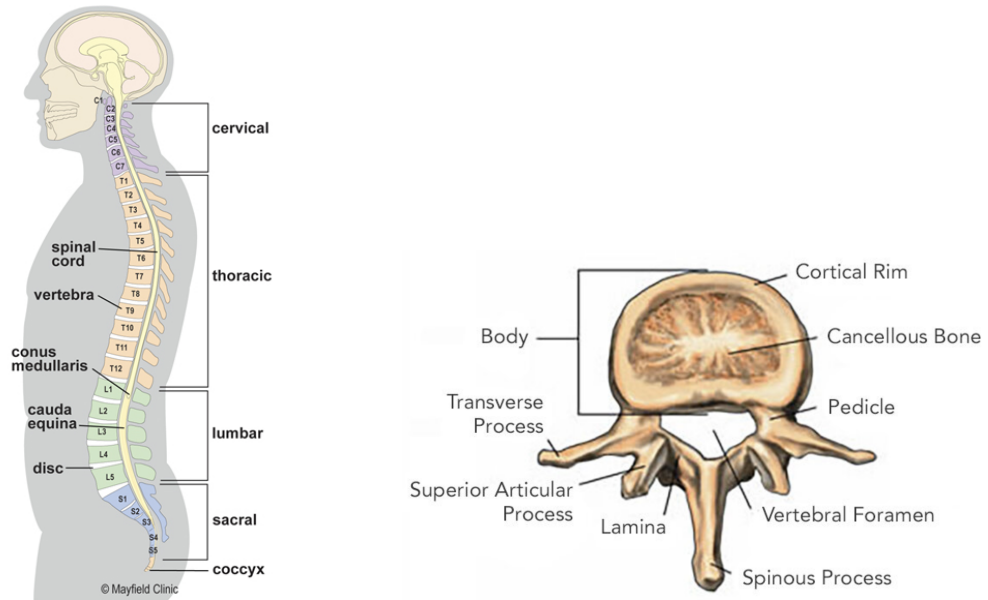
This chapter provides a concise yet detailed overview of the foundational concepts upon which this thesis is built. First, it introduces relevant spinal anatomy and key medical terminology, including an explanation of the procedure our patients underwent and the reasons it is both unique and significant. The discussion then shifts to medical imaging, covering the principal imaging modalities used for these patients and examining how medical images are typically stored and distributed. Finally, the chapter concludes with an exploration of machine learning—its role and context in medical imaging and in our patients’ operation—alongside an examination of recent literature concerning medical image segmentation.

2.1 Anatomical and Medical Terminology

This section will give a brief overview of the relevant anatomy, terminology, and medical operations that will be used throughout this thesis.

2.1.1 Anatomy of the Human Spine

The spine is a central component of human anatomy. It is a set of 33 bones stacked on top of each other with a distinct S-shaped curvature. The spine is divided into four sections: cervical, thoracic, lumbar, and sacral. As you move from the head down to the



(a) Full body view of a human spine. [1] (b) The axial view of a lumbar vertebra.[2]

Figure 2.1: An overview of spinal anatomy in humans.

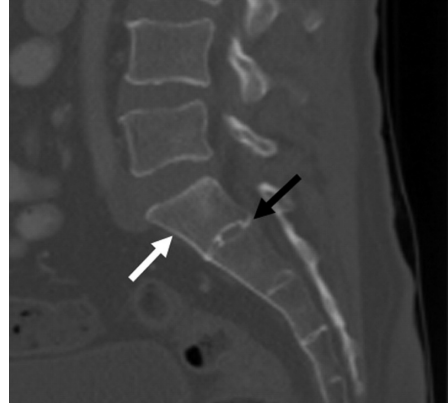
tailbone, individual bones tend to increase in size. The sacrum is made up of 8 bones that are fused together and do not move. Refer to Figure 2.1a. Most vertebrae follow a similar shape. First, a roughly cylindrical area called the vertebral body. Second, a prominent and pointed portion consisting of the transverse process, superior articular process, and spinous process which together will be called the collective term ‘processes’ hereafter. Finally, the former two portions are connected by pedicels on two sides. In between the vertebral body and processes is where the spinal cord passes through [1]. See Figure 2.1b for a detailed drawing of a vertebra.

2.1.2 Common Spinal Anomalies

While the previous section defined the normal shape of the spine, some anomalies can alter the shape and size of it. One such anomaly is lumbosacral transitional vertebrae (LSTV) [3–6]. This can happen when either the bottom-most lumbar vertebra (L5)



(a) Lumbarized S1 vertebra.



(b) Sacralized L5 vertebra.

Figure 2.2: Patients exhibiting 'additional' (a) and 'missing' (b) vertebrae. [3]

fuses to the sacrum or when the uppermost sacral bone (S1) detaches. This could be thought of as a person 'missing' or having an 'additional' vertebra, respectively. This relatively common spinal anomaly can increase the difficulty in counting vertebrae for classification, both by manual means and automated methods. See Figure 2.2 for an example of both cases. The lumbarization of S1 has even been observed in the small dataset of this project and has caused algorithms to fail in those instances. This final point will be further explored later on.

2.1.3 Hematopoietic Stem Cell Transplantation

Hematopoiesis, the process by which blood cells are formed, can be disrupted by any of a multitude of benign or malignant conditions such as leukemia [7]. Hematopoietic Stem Cell Transplantation (HSCT), also known simply as a bone marrow transplant, was the first type of stem cell therapy administered successfully to humans [8]. In traditional chemotherapy, radiation is used to kill cancerous cells [9]. However, in conditions of the blood, the amount of radiation needed to kill tumorous cells can completely destroy the bone marrow. HSCT has been shown to rescue and restore the bone marrow after intense radiation therapy and it aims to do this by injecting the stem cells into the bone

marrow. If the transplant is successful and the patient is able to sustain hematopoiesis for a certain length of time, it is called *engraftment*. If the stem cells do not hold, a *graft failure* happens, and if cancer returns, it is called a *relapse* [10–12].

2.2 Medical Imaging

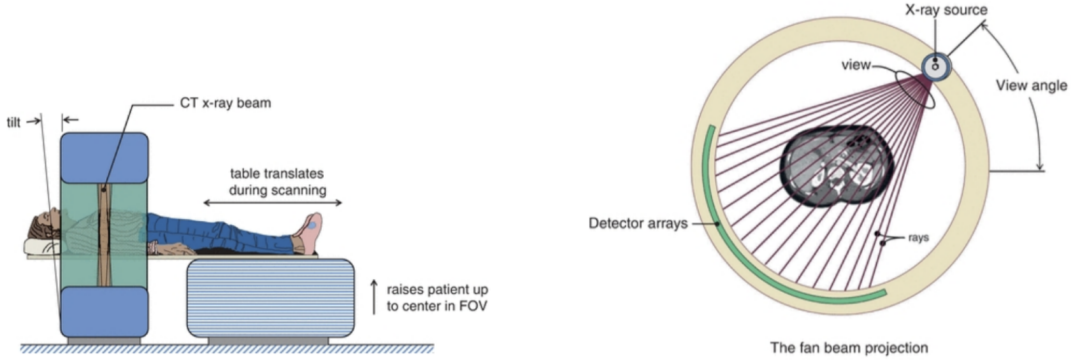
This section will detail the important characteristics of medical imaging relevant to this project. This includes an introduction to CT and PET imaging, how these images are stored, accessed, and displayed, and the characteristics that denote image quality.

2.2.1 Computed Tomography

Computed Tomography (CT) is a non-invasive imaging modality that uses X-rays to create a three-dimensional (3D) image of a patient. Modern CT machines use a helically rotating X-ray emitter that passes through a person. The radiation is then collected again on the other side of the patient, and the attenuation of different body parts is what creates the image [13]. We can measure the sum of X-rays that pass through various substances, and these measurements are assigned an attenuation coefficient, μ . These coefficients are then rescaled to a linear unit suitable for mathematical reconstruction techniques and display on electronic screens. Hounsfield units (HU) are given by

$$HU = 1000 \frac{\mu - \mu_{water}}{\mu_{water}}, \quad (2.1)$$

where μ_{water} is the linear attenuation of water that is measured initially during each scan calibration [14]. One of the biggest strengths of CT scans is their ability to detect and display bones in a detailed manner, which is a major focus of this work. Refer to Fig. 2.2 for an example CT image. More will be shown later in this thesis.



(a) The setup of a patient in a CT scanner. (b) X-ray fan beam projector and detector.

Figure 2.3: Overview of CT operation [13]

2.2.2 Positron Emission Tomography

Positron Emission Tomography (PET) is another non-invasive imaging modality, but is quite different than CT. It is mainly used to detect both typical and atypical metabolic activity of various tissues [15]. Atypical metabolic cellular activity can be from various diseases, such as cancer [16, 17]. It is a nuclear-based imaging technique, which means it relies on atomic activity. Specifically, PET scans detect the energy emitted by the annihilation of a positron and electron. Annihilation Coincidence Detection (ACD) is the process of detecting the interaction of an electron and positron. Electrons are found naturally in the body, but positrons are not and in this context are emitted by a radioactive substance injected into the body called a radiotracer/radionuclide [18]. Similar to CT, there is a normalized value used for quantitative analysis. The standard uptake value (SUV) is given by

$$SUV(t) = \frac{c_{img}(t)}{ID/BW}, \quad (2.2)$$

where c_{img} is the radioactivity measured at time t in units of MBq/mL , ID is the amount of injected dose in MBq , and BW is the patient's body weight converted into body volume, assuming an average mass density of $1g/mL$ [19].

The most common and prolific radiotracer used for PET imaging is $[^{18}F]$ Fluoro-2-deoxy-2-d-glucose (FDG). It is used to assess glucose metabolism in the brain, cancer, central nervous system disorders, etc [20]. Another, more uncommon, radiotracer is 3'-Deoxy-3'- $[^{18}F]$ fluorothymidine (FLT) which is mainly used for the monitoring of cell proliferation of tumors [21]. Importantly for this work, it has been suggested that FLT may have higher contrast for cellular metabolism in bone marrow [22, 23].

It is crucial to closely monitor patients undergoing HSCT as detecting a graft failure or relapse early is key to provide intervening care to the patients [24]. Traditionally, bone marrow transplants are monitored using a *single aspirate biopsy*, typically on the iliac crest, the top-most portion of the pelvic bone. This is because it has been shown to have the highest cell proliferation following the transplant [25]. However, this is just one relatively small area and monitoring multiple locations via biopsy is impossible due to its invasiveness and the high risk of infection.

Recent advances in medicine show that it may be possible to monitor bone marrow metabolism in the entire body during the engraftment process via FLT-PET imaging. This research has several notable consequences: First, physicians may be able to use imaging to understand the exact process by which hematopoiesis and cellular engraftment take place in humans. Second, imaging is non-invasive and therefore more frequent evaluations can happen. This could lead to a more accurate and quick detection of engraftment and potential relapse to enhance patient outcomes [26–29].

Williams et al. performed a pilot study of 23 recipients of HSCT and used dual CT/FLT-PET imaging. Images were taken one day before the bone marrow transplant,

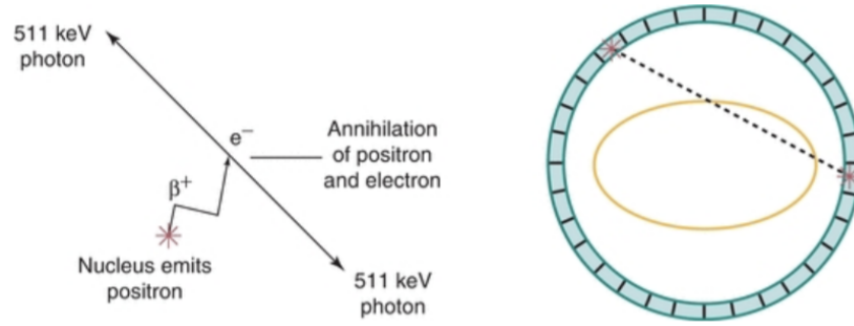


Figure 2.4: ACD illustration. When the ring of detectors (blue) surrounding the patient detect two nearly simultaneous photon collisions, they trace the approximately linear path back to where the initial interaction happened. This is the foundation of PET imaging. [13]

5-9 days after, and again 28 days post transplant. They were able to monitor cell proliferation in not just the pelvis, but also the sternum, spine, femur, and humerus. The spine is of particular importance for this thesis [26–29]. The major motivation of my work is to create a tool that can automatically segment individual vertebral bodies, extract the FLT-PET SUV data per cancellous bone (see Fig. 2.1b) and use those SUV measurements to monitor the recovery of HSCT patients, predicting engraftment, graft failure, or relapse.

2.2.3 Orientations and Coordinate Systems

In medical imaging, there are typically three sets of coordinate systems that need to be tracked. The first set consists of the anatomical planes. In 3 dimensions, these three planes are all orthogonal to each other. Refer to Fig. 2.5 for a visualization.

- The *axial* or *transverse* plane slices from head (superior) to feet (inferior).
- *Coronal* or *frontal* divides from the front of the body (anterior) to the back (posterior).
- *Sagittal* sections the body into left and right.

In radiology and neuroimaging, it is common practice to define the image axes based off of these terms. One common orientation is LPS, defined as L: starting from right and going left, P: from anterior towards posterior, S: from inferior towards superior. Another example is RAS. If you flip the first two axes in what was previously defined, you would get R: left-right, A: posterior-anterior, S: inferior-superior. Both are equally valid and can be interchanged, along with a multitude of other orientations.

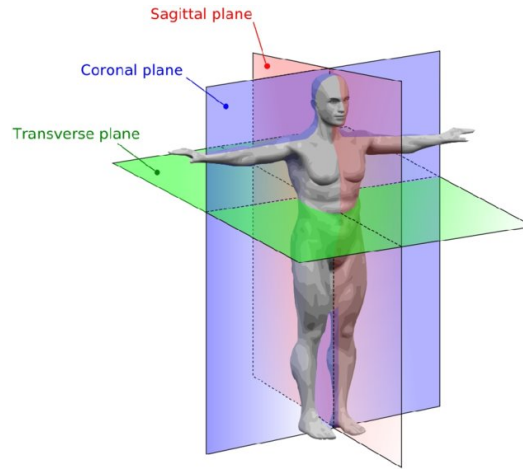
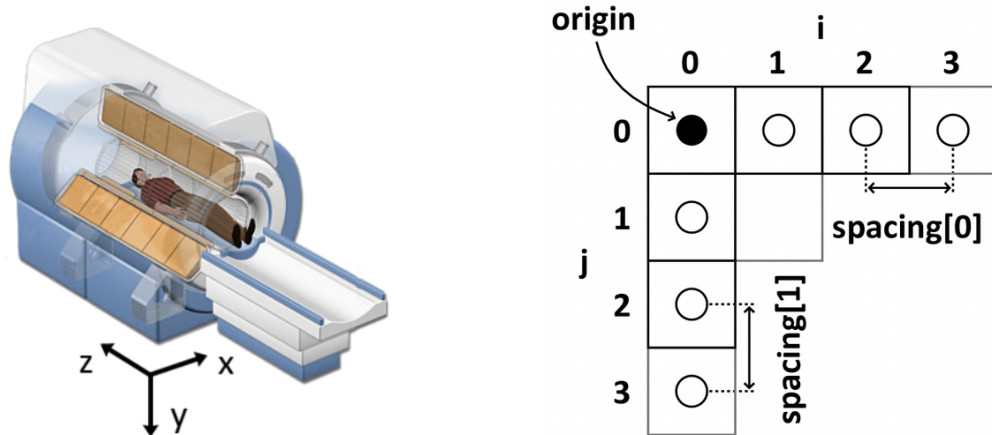


Figure 2.5: A representation of the three anatomical planes. [30]

The world coordinate system is typically defined as a Cartesian system in which the imaging machine physically resides with units of millimeters (mm) and variables of x , y , z .

Finally, the image coordinate system. In two dimensions, pixel locations are defined by (i, j) whereas in 3 dimensions voxel indexing is (i, j, k) . These represent the locations, and the values are the intensities that make up the image. For example, voxel intensities in CT scans are usually HU, defined in Eq. (2.1).



(a) A graphic depicting world coordinates. (b) Image pixels in reference to the origin.

Figure 2.6: World and image coordinates. [31]

2.2.4 Image Quality

In medical imaging, the most important characteristics are spatial resolution, contrast, artifacts, and noise. Below, these factors will be explored in regards to CT images [13, 32].

1. Image Contrast: In radiology, CT contrast is defined as the difference in x-ray attenuation in different types of tissue. In bones, absorption is high and are ‘brighter’ as they are very dense materials. On the other hand, soft tissue like organs have much lower attenuation and therefore are ‘dimmer’.
2. Spatial Resolution: This is the ability to distinguish small and closely arranged items in an image. There are several factors that determine this such as spatial frequency, focal-spot size, and most importantly with regards to this thesis, sampling. Sampling is made up of aperture size (the width of the x-ray detectors) and the physical spacing in between sequential detector measurements. See Fig. 2.6b for a basic illustration of spacing between pixels. In short, to resolve a highly detailed image, you must take many x-ray measurements consequently

introduce the subject to a larger amount of radiation. However, in this project, the CT images have a very thick axial resolution. This will be an important consideration further on.

3. Image Noise: This is related to the number of x-ray photons contributing to each detector measurement. This is directly affected by the peak kilovoltage, kVp , current (amperage), mA , and scan time taken in seconds (considered with amperage as $mA \times s$).
4. Other Quality Factors: Patient movement can introduce blurring. Different reconstruction algorithms can change the outcome of the image. Most importantly, the filter chosen during backprojection can drastically affect quality. Like most things, the filter sharpness is usually chosen as a compromise between noise level and resolution. There are other things, such as artifacts, that pertain to detector geometry and is not the focus of this thesis.

2.2.5 File Formats

Digital Imaging and Communications in Medicine (DICOM) is the international standard, non-proprietary, data interchange and medical imaging format used to unify different imaging systems [33]. It originated in 1985 and was formed by a joint committee of the American College of Radiology (ACR) and National Electrical Manufacturers Association (NEMA) to create a standard not only for imaging modalities, but also different manufacturers, vendors, as well as medical communication [34, 35].

In this standard, there is an information hierarchy where the information is divided into four main sections: *Patient* is each person being studied. *Study* defines separate scan sessions, including what type of scan being acquired. *Series* is individual scans of the patient. *Instance* these are the sequential image slices. Each instance is stored

as a separate file that together make up the 3D image. Also, the individual instances contain a header, or metadata, that contain relevant parameters such as image spacing, acquisition date, subject identifiers, etc.



Figure 2.7: The DICOM standard hierarchy. [34]

Neuroimaging Informatics Technology Initiative (NIfTI) is an imaging format that was also created to solve a problem in medical imaging. At the time of its envisioning, the most common 3D imaging format for engineering-related tasks, ANALYZE 7.5, could not resolve spatial orientation unambiguously and therefore needed to be shipped with an additional file (usually .mat) that acted as the header. In NIfTI, unlike DICOM, the volumetric information is stored in one file along with a single header that contains all relevant information. The size of this header has been increased overtime to make it more extensive [36, 37]. It initially was developed for fMRI but has been expanded to many modalities, provided they have the correct information available.

In the context of this project, all CT and PET images were initially acquired and stored as DICOM files but were then converted to NIfTI for ease of use and compatibility with various software engineering libraries like NiBabel [38].

2.3 Machine Learning and Neural Networks

Ever since humans created computers, we have dreamed of them being able to think for themselves. Since the invention of the graphics processing unit (GPU) by NVIDIA in 1999, research has thrived for artificial intelligence (AI) and machine learning (ML) due to its ability to process data quickly and efficiently in a parallelized fashion [39]. Machine learning is the process by which a computer automatically creates an algorithm that can learn patterns by statistical estimation and probability. The most prolific areas of ML currently include, but are not limited to, the understanding and processing of images and natural language (including speech and text), automation, finance, control systems (autopilot, robotics, etc), and medical diagnosis [40, 41].

2.3.1 Artificial Neural Networks

In their simplest form, artificial neural networks (ANNs), or simply neural networks, are an imitation of the physical neurons found in human brains. In 1943, the first idea for a mathematical representation of the brain’s processing unit (the neuron) was put forth by McCulloch and Pitts [42]. Neurons, both real and artificial, receive inputs (stimulus), process them, and then after reaching a certain threshold, output their own information to the next neuron(s) connected to it [43, 44].

A group of interconnected artificial neurons, or nodes, are what make up the basic ANN. The signal, or input, that each node receives is a real number x_n , the strength of the signal is called the weight w_n , and the output, $F(x)$, is the sum of the weights passed through some non-linear expression called an activation function. The activation

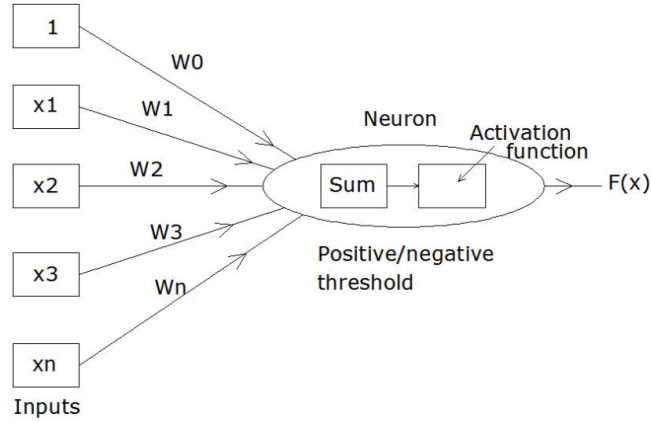


Figure 2.8: Illustration of an artificial neuron. The processing information is stored in the numerical weights $w_0 - w_n$, the tuning of which can be considered learning. [44]

function serves to turn those real numbers into a probability. Using the tanh function as an example, Eq. (2.3) describes this process below:

$$F(x) = \frac{2}{1 + e^{-2s}} - 1, s = \sum_{i=0}^n x_i w_i. \quad (2.3)$$

The deep feedforward network, or multilayer perceptron (MLP), is the cornerstone of modern machine learning. In it, there are multiple connected intermediate layers between the input and output layer called *hidden layers* as shown in Fig. 2.9. They are called feedforward networks because each layer is connected sequentially and the width (or depth) of the network is determined by the amount of hidden layers [40].

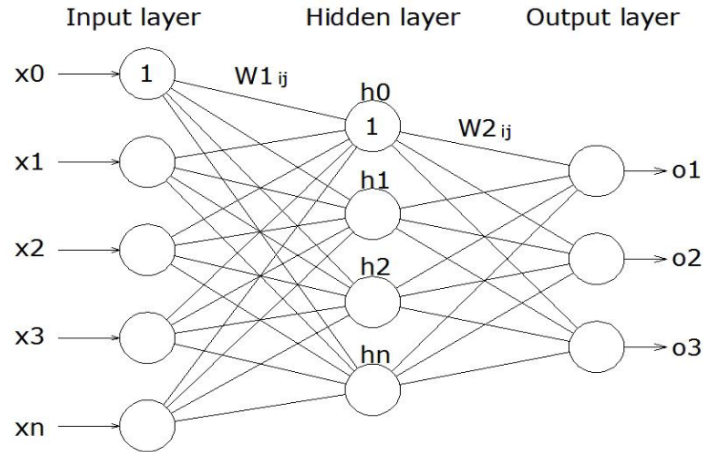


Figure 2.9: The basic structure of an artificial neural network. Each circle is an abstraction of an artificial neuron (see Fig. 2.8) [44]

2.3.2 Convolutional Neural Networks

Convolutional Neural Networks (CNNs), also called simply Conv Nets, are a special type of deep feedforward neural network that learns by optimizing kernel (filter) operations instead of adjusting each individual weight applied to a neuron. This makes CNNs a powerful tool for areas such as computer vision, audio processing, and medical image analysis [45–47].

The fundamental change from a standard feedforward network to a CNN is the inclusion of *convolution* operations in the hidden layers. The motivation behind this is that in traditional ANNs matrix multiplication can become quite unwieldy due to the large amount of data contained in images as well as the loss of nearby information. Convolution, in the context of image processing, is the process of taking an input image and moving a smaller matrix, the kernel (also called a filter), across it. This transforms the image into something called a *feature map* that the algorithm can use to extract information. Usually, the kernel is applied using the dot product, but not always and it depends on the architecture. Interestingly, commonly available CNN implementations

do not actually do convolution, but do a near identical function called *cross-correlation*, where the kernel is not time reversed relative to the input image. According to [40],

$$S(i, j) = (K * I)(i, j) = \sum_m \sum_n I(i + m, j + n) K(m, n), \quad (2.4)$$

where $S(i, j)$ is the extracted feature map, $I(i, j)$ is a 2-dimensional image and $K(m, n)$ is a smaller 2-dimensional kernel. The reason for Eq. (2.4) being considered near-identical to convolution is due to the fact that the kernel contents are learned by the algorithm and are symmetric. For conciseness in this thesis, both operations will be referred to as convolution. The most unique and attractive part of the convolution layer is the fact that the weights in each kernel are actually *shared* between neurons. This means that far fewer weights are needed per neuron [48]. In CNNs, the basic architecture is given by the following adjustable parameters:

- Input channels: This decides the number of kernels applied to the input. In multicolor (RGB for example) images, you would need three kernels to process each color channel.
- Kernel: The size of the kernel determines how many weights and neurons are at the output. In essence, the smaller the kernel, the greater amount of detail is extracted [40].
- Stride: This modifies the amount that the kernel moves when being applied to the input image. A number greater than 1 results in some pixels being skipped and the output being smaller.
- Pooling: Pooling performs an operation on groups of neurons like average or max to reduce the dimensionality of the data.

- **Padding:** Padding appends extra data (usually zeros) to the edges of images in order for the kernel to handle them more effectively and preserve matrix size.
- **Fully connected layers:** This is usually the final layer(s) of the network, and it combines the feature maps extracted from the convolution layers into a final prediction.
- **Activation:** Transforms the output into a non-linearity to provide a probability for each output channel.
- **Output channels:** Determines the number of kernels to be applied to the output of each layer.

2.3.3 Training Neural Networks

The end goal of creating these complex models is for them to be able to learn useful information and patterns automatically. Thus I must discuss the most common aspects of training, including supervision, loss functions, gradient flow, optimization, and backpropagation. I will also discuss the art of *hyperparameter tuning* and how it affects training.

In machine learning, *unsupervised learning* refers to the process of a network receiving data that contains a large amount of information and learning abstract, yet useful features about the data without any human action. This can range from denoising images to clustering similar patterns [40]. *Supervised learning* is the opposite in the sense that the network is provided human created manual annotations, also called ground truth labels or targets which denote the desired outcome. During training, the algorithm is tasked with making a ‘guess’ about relevant features in the data and then is penalized based on how incorrect it was [40]. The success of this type of learning is largely dependent on the consistency of the labels as too much inconsistency can

‘confuse’ the network about finding meaningful patterns. A mix between both of these methods is *semi-supervised learning*, where the data distribution has both labeled and unlabeled data. It can be seen in two views, depending on the task: unsupervised learning guided by constraints or as supervised learning with additional context and distribution about the data [49]. While all three of these methods have their merit, supervised learning is generally preferred for segmentation [40, 46].

The ability for a network to find not only discernible patterns in data, but to capture relevant features again in completely new data that was not present during training (referred to as validation or test data) is called *generalization*. This is both the most attractive and most difficult aspect of neural networks. The basic training algorithm looks like this: the network is provided a sample of the data. Then, it makes a *forward pass* through the input layer and all subsequent layers (refer to Fig. 2.9). The non-linearity at the end converts the output into a probabilistic prediction. Then, that prediction is passed into a *loss function* along with the ground truth, which measures how correct or incorrect the prediction was. There are many different loss functions that are suited for different tasks (like cross entropy). Finally, an *optimizer* attempts to minimize the loss by going backward through the network and adjusting the weights. This is called *backpropagation* [40]. What separates machine learning from a ‘simple’ optimization problem is generalization. The most basic way to assess a networks performance is two fold: it must have a small training error and a small difference between training and test error. Two related terms are *underfitting* and *overfitting*. Underfitting occurs when the network is unable to learn anything meaningful and does not have a sufficiently low training loss. Overfitting is when the model *does* learn features in the data but learns them too well and cannot generalize well (the difference

in loss between train and test is too high). An example loss function, mean squared error (MSE), also called squared L2 norm, is given by

$$L = \ell(x, y) = \{l_1, l_2, \dots, l_N\}^T, l_n = (x_n - y_n)^2, \quad (2.5)$$

where x_n is an element in the input, y_n is an element in the output, and N denotes the total element in each, or the batch size. It is common to use *reduction* in loss functions in order to aggregate the entire sample's loss into one scalar. Two common modes of reduction are averaging and summation. In the case of MSE, it looks like this:

$$\ell(x, y) = \begin{cases} \text{mean}(L) & \text{reduction} = \text{avg} \\ \text{sum}(L) & \text{reduction} = \text{sum} \end{cases} \quad (2.6)$$

Finally, it is important to at least mention the trends of model complexity and their ability to generalize. As ANNs and CNNs become deeper, or have more layers, the number of learnable parameters increases [50]. This is called the model's *capacity*, where more complex networks can theoretically learn more complex features. However, it is possible that too many parameters can lead the network to learn very finely grained details, such as noise, in the training data and fit to that. Fig. 2.10 shows a U-like curvature relating error to model capacity. It is apparent that there is an optimal complexity, which is highly dependent on the problem and amount of data present [40].

2.3.4 Transformations, Patch Based Training

I have already discussed one of the most common pitfalls associated with network training: overfitting. Another extremely common issue is insufficient training data for the model to learn or generalize [51]. To address this, researchers frequently employ data

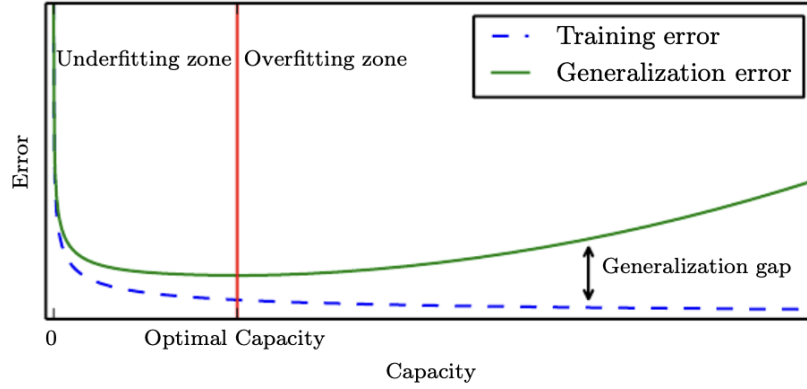


Figure 2.10: Trends related to network capacity. As a model becomes more complex, the risk of overfitting increases. [40]

augmentation to both create ‘new’ data and to increase the variability of samples [52]. The main constraint of this method is that the augmentations need to be realistic. That is, the augmentations should be confined so that they mimic real variations in samples. In the case of images, common transformations include rotations, axis flipping, color space and contrast shifts, scaling, noise addition, and even elastic deformations. Usually, these are applied randomly during each iteration so the model has less chance of ‘memorizing’ the modifications.

A final tactic that will be employed in this thesis is *patch based training*. This is a method of randomly cropping a subsection of the input data and feeding that smaller sample into the network. Usually, the algorithm is set up to still ensure adequate coverage of the image by doing this random cropping multiple times on the same image. This method allows very computationally heavy 3-dimensional volumes like medical images to be broken up into parts and allow for more efficient training, as well as additional regularization [53].

2.3.5 Regularization

As we have seen, networks benefit greatly from having diverse and variable data to prevent the model overfitting. It has been seen that networks can rely on the predictability of a small subset of neurons. In order to force robustness, a common tactic in CNNs is *dropout*. During training, dropout randomly zeros the value of neurons. This idea can also be extended to CNNs by zeroing out entire kernels [52, 54].

In machine learning tasks concerning medical imaging, a popular addition to CNN-style architectures is *batch normalization*. Batch normalization was introduced by [55] to normalize the layer inputs and improve training efficiency. It works by taking the average of the batch’s activations and dividing by the standard deviation. However, 3-dimensional models can be quite large and consume a lot of memory during training meaning it can become infeasible to have a large batch size. It was shown in [56] that as batch size is reduced, training performance is significantly harmed. They proposed a new, more practical method that can achieve stable training not reliant on batch statistics. If a two dimensional image has the shape (N, C, H, W) where N is the batch size, C is the channel dimension, and H and W are the spatial height and width, *group normalization* partitions the C channels into G groups. Then, it applies normalization by computing statistics (mean and std deviation) at the group level, which is across its channel and spatial dimensions and independent of the batch.

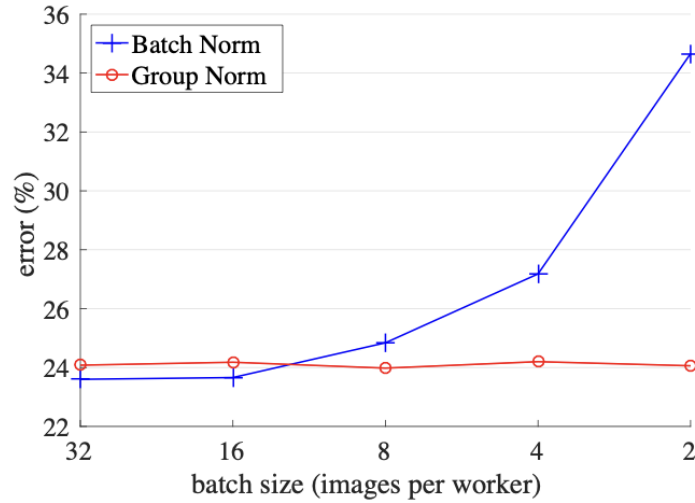


Figure 2.11: How batch and group norm are affected by batch size. Each point is the same model trained with a different batch size. The error was measured on the same validation set. [56]

2.4 Image Segmentation

Image segmentation is the process of dividing the pixels of an image into different objects. The extent, or detail, to which the pixels are divided depends on the specific type of segmentation. *Semantic segmentation*, the simpler of the two methods, is the act of categorizing each pixel or voxel of an image into a general class like ‘dog’ or ‘car’. *Instance segmentation*, the more complex operation, is the act of differentiating between individual objects of the same semantic segmentation class, such as ‘car 1’ and ‘car 2’ [57, 58]. In this section, there will be a brief overview of segmentation techniques that employ signal and image processing as well as a more in depth look at how machine learning has been applied to both 2D images and 3D volumes to solve this problem. One of the main focuses of this thesis is a machine learning approach to semantic segmentation of vertebral bodies in CT images. This will be discussed in further detail in a later subsection.

2.4.1 Traditional Image Segmentation

In this section, I will introduce image segmentation in terms of grayscale (where one end of pixel intensity is white and the other is black). This is for two reasons: first, grayscale operations are typically much simpler than colored ones. Second, the primary function of this thesis is to segment CT images, which are based on a single-channel intensity range of the Hounsfield unit, which is often displayed in grayscale (see Fig. 2.2).

Edges, in the context of grayscale images, can be defined by as a boundary, contour, or outline that is the result of an abrupt change (or discontinuity) in local pixel intensity values. It can also be thought of as a difference between two distinct regions in the image. Edge detection, therefore, is classifying which pixels are edge pixels and which are not. When these edge pixels are then put into an image of their own, it's usually called either an edge map or a *mask* [58–60]. One common and relatively simple way to detect abrupt changes is using the derivative, or the gradient. While the gradient is a continuous operator, it can be approximated to be used in the discrete nature of digital images. There are many different ways to approximate it, but one of the most simple is using the difference between adjacent pixels. We can actually use *convolution*, as defined earlier, to do this difference approximation on 2D images as well as more complex operations. This means we can define *filters* (or kernels as they are used earlier), convolve them with the input, and then do some calculus to get the desired edges. Below is a minimal overview of this operation:

$$S_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}, S_y = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}, \quad (2.7)$$

$$|\nabla f| = \sqrt{G_x^2 + G_y^2}, \theta = \tan^{-1} \frac{G_y}{G_x}, \quad (2.8)$$

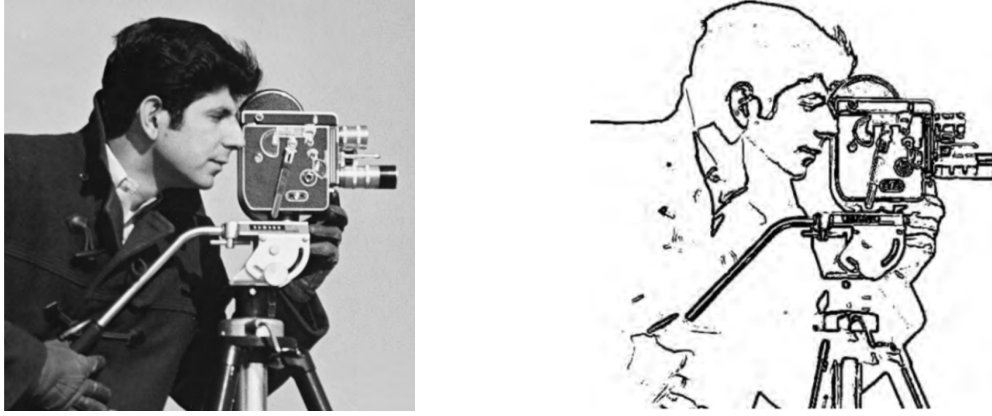
where S_x is the Sobel operator in the vertical direction and S_y in the horizontal direction (G_x and G_y after convolution). Then, we can find the magnitude $|\nabla f|$ to get the strength of the edge and direction θ to find the edge's orientation. After doing all of this followed by a simple thresholding operation, you have an estimate of the edges in a grayscale image. The value of the threshold decides how many edges will be detected (a larger threshold will be less noisy) [59]. See Fig. 2.12 below to see how this applies to a real image. Of course, there is no one correct filter and there are many variations. A few named examples include the Roberts, Prewitt, and Frei-Chen kernels.

Other common ways people have employed traditional image processing techniques to perform segmentation include thresholding, K-mean clustering, region growing, normal/graph cutting, and the more complex Canny algorithm [58, 59]. Often times, these methods are combined to take advantage of their individual strengths.

2.4.2 Performance Metrics

When performing segmentation, like any task, it is important to quantitatively measure how well the algorithm works. There are many ways to evaluate segmentation effectiveness depending on the specific task, and it is important to define several of them in order to have a robust and complete picture. In this section, I will define some of the most common metric for image segmentation: the confusion matrix, accuracy, sensitivity, specificity, Jaccard index, Sørensen–Dice coefficient, and Hausdorff distance [61, 62].

- *Confusion Matrix*: Table 2.1 shows the basic values used for the following metrics. TP is the pixels where both the prediction and ground truth (GT) agree that the pixel is in a given class. TN is the converse of this, pixels that are predicted to



(a) Original 'cameraman' image in grayscale. (b) Edge map of the 'cameraman' image.

Figure 2.12: Visual representation of applying the Sobel operator for edge detection. [59]

Table 2.1: The Confusion Matrix

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

not be in a given class and the GT agrees. FP is when a pixel is predicted to be in a class but the GT disagrees. Finally, FN is the opposite of this.

- *Accuracy*: This measures the overall agreement between the prediction and the GT by ratioing correct predictions over the total number of pixels. However, this metric runs into issues when the regions of interest (ROIs) are proportionally small compared to the background. This is referred to as *class imbalance* and is a common problem in many segmentation problems, such as the work in this thesis. Assume there are 1,000 pixels in an image. If the ROI is only 100 pixels and the other 900 are background, then that means 10% of pixels are in the class.

Therefore, even if the algorithm predicts every single pixel as a background, the accuracy would still show 90% success and is therefore considered a biased metric.

$$A = \frac{TP + TN}{TP + FP + FN + TN}. \quad (2.9)$$

- *Sensitivity*: Sensitivity is also referred to as the true positive rate (TPR) and is an algorithm's ability to recognize positivity, shown as

$$TPR = \frac{TP}{TP + FN}. \quad (2.10)$$

- *Specificity*: Similarly, this is referred to as true negative rate (TNR) to show the ability of detecting negative outcomes accurately, given by

$$TNR = \frac{TN}{TN + FP}. \quad (2.11)$$

- *Jaccard index*: The Jaccard index is also called the *intersection-over-union* (IoU) metric. The bias of accuracy is eliminated by not considering true negative pixels in the calculation. Additional notation is given to illustrate the metric in terms of sets. In the IoU metric, which is represented as,

$$IoU = \frac{TP}{TP + FP + FN} = \frac{|f \cap f^*|}{|f \cup f^*|}, \quad (2.12)$$

f is the GT mask and f^* is the prediction.

- *Sørensen–Dice coefficient*: More commonly called the Dice score, this is one of the most widely used metrics for medical image segmentation [63]. It functions

very similarly to IoU by its ability to overcome class imbalance. The Dice score is given by

$$Dice = \frac{2TP}{2TP + FP + FN} = \frac{2|f \cap f^*|}{|f| + |f^*|}. \quad (2.13)$$

- *Hausdorff distance*: The definition of Hausdorff distance (HD) is the maximum distance of a set to the nearest point in the other set. In imaging, it can be described as a way to quantify how far a point is from the nearest edge. This is less commonly used in medical segmentation, but is still present often enough to be included in this review [64]. It is given as

$$d_H(A, B) = \max_{a \in A} \left\{ \min_{b \in B} \{d(a, b)\} \right\}, \quad (2.14)$$

where a and b are points in the sets A and B and $d(a, b)$ is any distance metric, such as the simple Euclidean distance. Eq. (2.14) was derived from [65].

2.4.3 CNNs for Images

While CNNs in the context of medical image segmentation are the main topic of this thesis, it is important to acknowledge the original problems they were invented to face and the architectures that solved them. CNNs rose to prominence originally in the *image classification* space. Classification refers to assigning one or more labels to an entire image, answering the question of ‘what does this image contain overall?’. In 2012, AlexNet [66] achieved record-breaking performance on ImageNet [67] showing that convolutional networks can indeed learn hierarchical features directly from raw images. Later in 2015, the first fully convolutional network (FCN) for image segmentation was introduced [68]. In this paper, the authors took common networks for time, like AlexNet, and replaced the fully connected layers with 1x1 convolution blocks which made

the network entirely convolutional. In addition, it introduced learnable upsampling (sometimes called transposed convolution) in what can be viewed as a decoder stage, as well as skip connections which are widely used in most modern segmentation CNNs today. The FCN paper showed that *any* classification CNN can be adapted to produce dense outputs of any resolution. While this work was a milestone in automatic segmentation algorithms, it had its drawbacks. It relied on a backbone of relatively shallow (by today's standards), pre-built networks. When people attempted to create deeper FCN networks, they ran into issues like vanishing and exploding gradients, which made training unstable. This is what the ResNet [69] paper addressed and created a solution for. They proposed *residual blocks* that would allow gradients to skip, or flow around, certain convolutional layers. The direct consequence of this was very deep networks that could much more easily extract richer and more detailed features from the input. The ideas in FCN and ResNet are employed in the custom architecture made for this thesis.

2.4.4 CNNs for Medical Image Segmentation

One of the most popular, foundational networks for medical imaging segmentation is the UNet [70]. The UNet is a deep fully convolutional network made for 2-dimensional segmentation of medical images. As seen in Fig. 2.13, the UNet is called so due to its symmetrical *encoder-decoder* design with skip connections at every resolution level. The skip connections carry local spatial information from one encoder stage directly into its corresponding decoder layer, which greatly improves segmentation accuracy of small details that might otherwise get lost in deep layers. The UNet is heavily inspired by the FCN paper, which used skip connections only in the decoder pathway. This architecture is famous for its ability to perform well on small datasets due to both the

skip connections allowing the preservation of fine details and the use of heavy data augmentation during training.

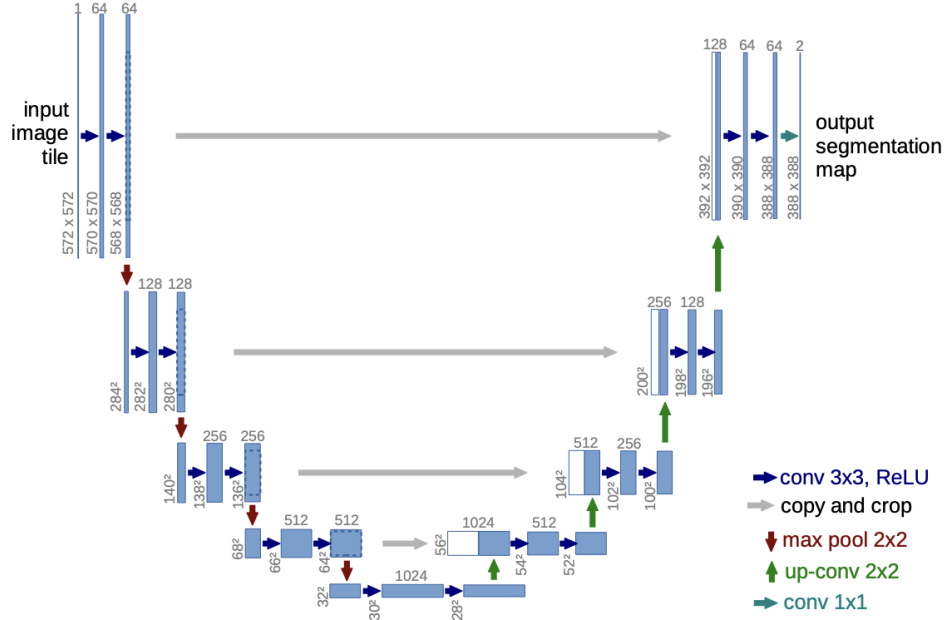


Figure 2.13: The 2D UNet architecture proposed in [70].

Many medical imaging modalities like CT, PET, and MRI are inherently 3 dimensional. This fact led to a direct extension of the UNet to be used in 3D space [71]. In this, operations such as convolution, pooling, and upsampling are replaced by their 3D counterparts allowing for the model to learn volumetric features directly. This is important because many anatomical structures span multiple slices and 3D operations can help the network capture context around any given slice. The authors also talk about switching from a cross-entropy based loss to a Dice based loss, which is a consideration used in this thesis. Additionally, they mitigate the increased cost of 3D operations by employing patch-based training, which crops each volume into a smaller region before being fed into the network. The VNet [72] is another 3D adaptation of the UNet that uses Dice-based loss for training. Similar to how the UNet is based off of the FCN, the VNet employs residual blocks in order to prevent vanishing gradients in deep 3D

networks, much like the ResNet did for deep 2D models. The addition of residual units, along with using the PReLU activation function, are of particular importance for this thesis.

2.4.5 Attention in Deep Learning

Attention, in the context of machine learning, is a mechanism that enables a model to dynamically focus on the most relevant parts of an input. Instead of giving all input features the same weight, attention can calculate a set of weights that are applied to features in order to determine where in the input is the most useful information. This was initially used to solve the shortcomings of recurrent neural networks (RNNs) in sequential tasks. Below is an overview of the evolution of attention in machine learning and its relevance to this thesis.

RNNs were a promising answer to natural language processing (NLP) and machine translation tasks. In [73], the authors introduced a sequence-to sequence (seq2seq) framework that consists of an encoder RNN that compresses an input sequence to a fixed hidden state, and a decoder RNN that unfolds from the hidden state to generate an output sequence. This method achieved record performance on translating English sentences to French. However, a major shortcoming is that the network's performance starts to suffer the longer the sentence become due to a single hidden state representing the entire sequence. In 2015, Bahdanau et al. [74] introduced an *alignment model*, also called *attention*, on top of the seq2seq network. Using this mechanism, at each decoding step, a learned weight highlights which encoder states are most relevant. Rather than relying on that one hidden state's vector to represent the whole input, the model is able to 'look back' at different parts of the input when trying to predict the next word. This greatly improved performance on translation tasks. A similar paper, [75], was able to extend the attention mechanism from text to the speech domain. A radical

approach to attention was given in [76], where they were able to completely remove the dependence on RNNs or even CNNs for NLP tasks. This model is driven purely by multi-head self-attention, which they call the *transformer*. Pure attention-based models are simpler, require less training time than previous networks, and even achieve better performance. The authors of [77] were able to expand the transformer to a more complex task than speech or audio recognition: speech separation, which showed how flexible attention can be in more complex NLP tasks.

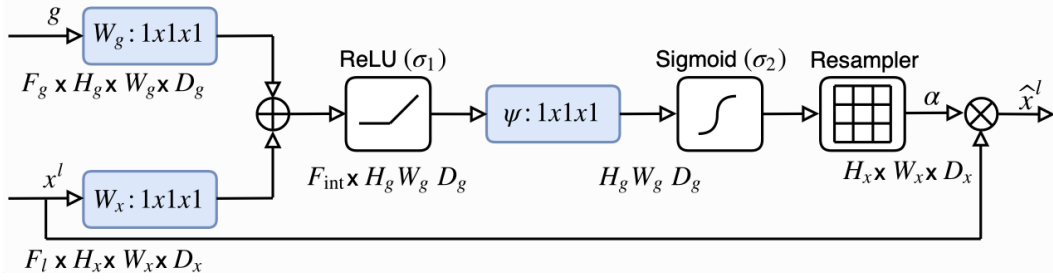


Figure 2.14: Attention gate mechanism, extracted from [78].

With seq2seq and transformer based approaches to natural language processing, it's natural to want to explore what attention could do in the imaging space as well. Where attention can help a network ‘focus’ on the most relevant timestamps of audio in a translation task, attention might be able to help highlight the spatial regions of interest important to segmentation tasks. This is what led to the development of the Attention UNet [78], where learned alignment, or attention, is integrated directly into the encoder and decoder blocks of the UNet architecture. Instead of passing every feature from the encoder directly to the decoder (utilizing global skip connections), *attention gates* learn to filter out irrelevant or noisy background signals, which makes the decoder's job more precise. In other words, the skip connections connecting each encoder and decoder layer are ‘gated’ by attention to filter relevance. One of the more appealing claims of this paper is a very minimal addition to computational cost or overhead when

compared to the original 3D UNet while having an improved performance. Fig. 2.14 above shows an overview of the attention block that is attached to each decoder layer.

2.4.6 Vertebral Body Segmentation of HSCT Patients

The dataset used in this thesis is the same set of images used by Holter et al. in their pilot study of imaging the marrow cavities of HSCT patients [26–28]. From hereafter, I will refer to these images as the HSCT dataset. This dataset consists of co-registered FLT-PET and CT images for patients of sequential scan times: one day before the transplant (referred to in this work as day 1), 5-9 days post transplant (called day 2 in the files), and 28 days post transplant (day 3). *Co-registration* refers to the two imaging modalities being taken on the same machine, and both images having the same spatial reference, or origin. This means that a physical point in one modality refers to the same point in the other. This is especially useful, as this work is about performing image segmentation in the CT image, and then propagating the predicted label to the PET space. This is because [26] proposes that the bone marrow cavity of various bones (including the spine) can be used to monitor hematopoietic recovery. Each CT image has the (x, y, z) spacing of 1.17 mm x 1.17 mm x 5 mm, where 5 mm refers to the z , or axial resolution in an LPS orientation. This is an important note as 5 mm is an exceptionally thick spacing, to the point where even physicians have trouble discerning certain anatomical boundaries [29]. In the context of this thesis, the average anterior height of cervical discs is 2.6 mm [79] and for thoracic discs is 4 mm [80]. These are both less than the spacing thickness of our CT images, which means the boundaries between the vertebrae are not resolved consistently. Our PET images have an isotropic spacing of 4 mm³. While slightly better than the CT images, there is not enough metabolic activity in day 1 or day 2 to resolve many structures in the PET space [29].

Nguyen et al. performed the first automated vertebral body bone marrow cavity segmentation on this dataset in [81, 82]. In this method, they perform a 3D graph-cut to perform a full-body segmentation of the spine as a whole (meaning as one class) followed by connected component analysis and iterative thresholding to segment the binary spinal column. Subsequently, they introduce a Kalman filter to isolate the individual vertebral bodies. The authors use the aforementioned true positive rate as their performance metric criterion and report an average $TPR = 0.917$, but admit mixed performance on the test samples. Additionally, TPR only measures the ratio of positive values and does not account for false positives. This means that a model could achieve a high TPR, while also having a lot of pixels incorrectly labeled as positive. This method is more in line with the ‘traditional’ image segmentation methods that was mentioned briefly above. Carson et al. proposed the first use of machine learning for this project [83, 84]. He developed a multi-class, multi-view ensemble 2D UNet that predicted binary vertebral body column as well as bilateral pelvis and sternum classes. He reported an average dice score of 0.92. After this, he takes advantage of the the metabolic activity in day +28 of the PET space to localize individual vertebral bodies. However, a drawback of this method is that it cannot segment the individual vertebrae on initial scans before any hematopoiesis is readily captured by the PET scan. Also, the axial line of separation is always perfectly horizontal (one slice thick), leading to incorrect shapes for upper thoracic and cervical regions. Eguez [85] trains a fully 3D UNet to again segment binary vertebral column as well as the pelvis and sternum and reported a mean validation dice score of 0.901. After this, he used the unique anatomical characteristics of the spine to perform instance segmentation and localize individual vertebral bodies. He used the fact that to an observer at any given point, there is a unique angle at the same height of the vertebrae due to its distinct ‘S’ shape. Using this plus a padding and thresholding algorithm, he was able to roughly predict

the separations between vertebral bodies, improving on the boundaries generated in Carson’s work [83]. Due to a lack of annotated data, there was no way for him to report a per-vertebrae dice score on our dataset, but reported an average of 0.87 on the VerSe dataset [86] after resampling. One of the main drawbacks of this approach is anatomical variability. When the spine exhibits minimal curvature, a lack of well-defined pedicles in the axial plane, the presence or absence of transitional/sacrilized vertebrae, or any other number of conditions that are outside the ‘norm’, the algorithm will be unable to generalize. This is something Eguez mentions in his work, stating a ”proper geometric model” would be needed for patients with pre-existing conditions [85].

2.4.7 VerSe Dataset

One of the key challenges identified in the previous chapter is anatomical variability, which is part of what this thesis attempts to solve. Since our dataset is extremely small, consisting of only 36 volumes, there is a serious lack of representation for pre-existing conditions. To help with this, this work incorporates the Vertebral Segmentation (VerSe) dataset [86–88].

The VerSe dataset was introduced as part of the “Large Scale Vertebrae Segmentation Challenge” at MICCAI 2019 and 2020 and includes nearly 400 spinal CT scans with highly detailed manual annotations. Each vertebra — C1-C7, T1-T12, L1-L5, and transitional vertebrae L6 and T13 — is individually labeled. The dataset captures a broad range of scanner types, acquisition protocols, noise levels, anatomical variations, and slice thicknesses, making it a valuable resource for training robust segmentation models. CNNs dominated the challenge’s submissions, achieving Dice scores between 0.88 and 0.92, providing a strong reference for high-quality segmentation performance on high-quality data.

Figure 2.15 illustrates the VerSe dataset, where different colors overlaying the CT images represent distinct vertebral classes. One of the most important differences between the HSCT dataset and VerSe is the ROIs that are labeled. The VerSe dataset includes labels for both the vertebral bodies and their processes (the entire spine), whereas this project focuses exclusively on the marrow cavity of the vertebral bodies. This discrepancy presents a challenge in adapting the dataset for our purposes. Additionally, while our dataset maintains a fixed slice thickness of 5mm, the VerSe dataset shows wild variation, with spacings ranging from 0.3mm to 2mm—none as coarse as ours. The majority, if not all, of the models developed for the VerSe challenge rely on this higher resolution, making direct application to our dataset non-trivial. The adaptation of VerSe images and labels to match our data will be addressed in Chapter 3.

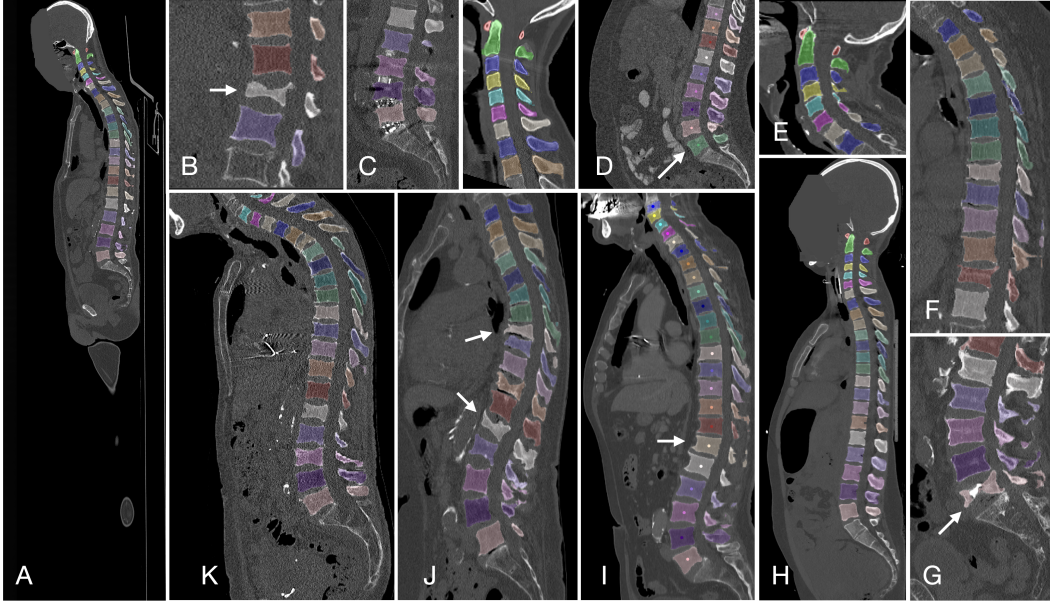


Figure 2.15: Images from the VerSe dataset showing the high degree of variability. Extracted from [86].

2.4.8 Tools, Libraries, and Frameworks for Machine Learning

Over the past decade, deep learning frameworks have rapidly expanded across several programming languages. Numerous libraries have been developed to streamline the training of ANNs, making the process more efficient and simplifying complex computations. A key feature of these frameworks is their integration with hardware, particularly support for GPU acceleration and parallelization/hyperthreading, all of which significantly enhance computation speed. In particular, the invention of CUDA cores for NVIDIA GPUs has dominated the field as the top hardware for AI/ML training [89].

- *PyTorch*: Developed by Meta, PyTorch is one of the most popular and widely used machine learning frameworks. Its dynamic computation graph allows for intuitive and flexible model construction, making it easy to create custom networks, like the one used as the backbone of this thesis. PyTorch is optimized for CUDA-enabled hardware, enabling efficient parallel processing on both CPUs and GPUs. It uses tensors as its primary data structure, which are able to efficiently represent higher dimensional data like images and gradients. Additionally, PyTorch increasingly supports specialized research areas, such as medical imaging [90].
- *MONAI*: is an open-source library built on top of PyTorch that is tailored specifically for deep learning applications in medical imaging. It is particularly well suited to tasks like medical image preprocessing, data augmentation, and model deployment for non-researcher use. All of these strengths make it a natural tool to use for incorporating deep learning in CT, MRI, Ultrasound, PET, or other modalities [91].
- *CuPy*: another open source library, CuPy provides the ability to run numerical computations on CUDA-enabled GPUs. Typically, as is the case in this thesis, it is used to accelerate more complex operations that PyTorch does not natively

support. It provides a nearly drop in replacement for NumPy and SciPy operations like linear algebra and signal processing routines [92–94].

- *Tensorflow*: Developed by Google, its simpler API, along with its tools for visualization and debugging, makes it quite popular for rapid prototyping and deployment [95].
- *MATLAB*: The Deep Learning Toolbox in MATLAB provides accelerated computing capabilities, enabling efficient training and deployment of neural networks. With its robust visualization API and support for engineering applications like control systems, signal processing, and robotics, it is popular among scientists [96].

2.4.9 Related Work

In recent years, there has been a tide of organ and spinal segmentation models that can achieve near expert level accuracy. Classic, but cleverly used convolutional-based architectures as well as highly complex transformer based models are taking up the field. Many consider VerSe to be the ‘gold standard’ in vertebrae segmentation, so many of the following papers will report their performance on that dataset.

CNN based: One method that is of particular importance to this work is [97]. The CT images are low-dose, so they suffer from similar streaking artifacts and noise levels. The authors employed residual blocks in their UNet variant, which showed considerable improvements in liver segmentation. Other methods implemented novel variations of the attention gating mechanism for specific segmentation tasks. The authors of [98] used attention gates with deep supervision for prostate cancer segmentation of CT images. They used 3D convolutions, which was an early example of performance boosts by using 3D operations. [99] is another example of using 3D attention blocks in a UNet, but noteworthy for this project is their use of multi-modal FDG-PET/CT images. While

the previous methods were early examples of using attention with UNets, [100] showed their true strength by having them be able to capture relevant regions in multiple medical imaging tasks ranging from pancreas to fetal spinal segmentation.

There have been plenty of UNet architectures recently devoted to spine segmentation. The dual densely connected UNet architecture in [101] has high accuracy in finding inconspicuous edges between vertebral bodies and intervertebral discs. The OP-convNet [102] presents a framework in which they take multiple 2D patched slices from CT scans and overlap them to preserve some spatial context but remove the computational costs associated with 3D operations. The MRNet presented in [103] is an example of multitask segmentation using a residual UNet performing well in X-ray segmentation of the lumbosacral spine and femoral heads. A more advanced UNet variant, [104], uses cascaded spatial pyramid pooling layers, residual blocks, and attention modules. Used as stage 1 in a two-step pipeline, it focuses on spine regions via attention and atrous convolutions. This model achieved Dice scores 0.94-0.95 on the VerSe dataset, making it have one of the highest performances of CNN-based models on that dataset. One of the most notable CNN UNet variants out there is nnUNet (no new UNet) [105]. It is a 2D *or* 3D self configuring framework that adapts both its architecture and training protocol to the specific dataset provided. It immediately became a top-performer on many medical datasets, including ~ 0.94 on VerSe and beat many hand crafted and fine tuned architectures. Finally, Total Segmentator (TS) is a comprehensive model trained to segment numerous structures in both CT and MRI images - covering anything from bones, organs, to other tissues like veins [106]. Its backbone is a UNet variant trained by nnUNet. Other than its amount of segment-able structures, its main claim to fame is its ease of deployment and use on any dataset out of the box. Relevant to this work, TS asserts that it can predict intervertebral discs, vertebral bodies, and even individual vertebrae. However, it resamples all images to

an isotropic 1.5mm spacing. Its results on the HSCT dataset will be compared to the method in this thesis, which keeps the images at their native spacing. An evaluation of TS in ultra low dose CT scans is given in [107]. In this paper, the authors share the challenges of working with CT images that have a low radiation dose and show that, for organs, TS can still find the boundaries. However, they do not mention bones and especially not vertebrae. In this work, I hypothesize that the intervertebral boundaries in low-dose CT images are much more difficult to segment and that even tools like TS, who claim to be able to segment any CT image, will struggle with it.

Transformer based: In recent years, transformer-based models have gained significant traction, driven by the widespread availability of powerful, cost-effective computing resources. These architectures capture long-range dependencies in spine imaging that conventional CNNs often overlook, and Vision Transformers (ViT) as well as hybrid CNN-Transformer frameworks now define the state of the art in vertebra segmentation. TransUNet [108] is the first medical segmentation model that combines a ViT encoder with a UNet decoder. The transformer showed that it can provide global context, which helped the model be able to distinguish vertebrae from adjacent similar structures. It was able to outperform pure CNN models on spinal CT datasets. The UNETR model [109] was the first network to use a pure vision transformer in the encoder, and it followed the classic U shape of the UNet. On the VerSe dataset, it achieved ~ 0.94 Dice segmentation performance, leveraging global self attention to improve segmentation continuity across slices. Built on UNETR with Swin transformer blocks, [110] and its second version [111] leveraged multi-scale feature representation to have higher efficiency and even slightly higher performance compared to the regular UNETR for MRI brain tumor segmentation. Tang et al in [112] proposed a self-supervised training strategy on the Swin UNETR that learns abstract features from abundant (over 5000 images) unlabeled CT abdominal images. This yields a pre-trained

model that can be then used as a strong foundation for more specific downstream tasks like segmentation. Verteformer [113] is a single-stage transformer based UNet that could handle arbitrary field of view inputs, meaning it could process an entire CT volumes in one go. It introduces an edge detection block (a convolutional self-attention module) to sharpen boundaries between adjacent vertebrae, and a global information extraction block to assist in vertebra labeling. It underperformed compared to other methods in its class, but it was still a promising step forward with its ability to directly output labeled vertebrae in one step. Finally, VerFormer [114] adds a vertebrae-aware global query (VGQ) module to a ViT backbone, which explicitly highlights tokens corresponding to vertebrae regions during self-attention. It is a 2D model, but was able to outperform every 3D model at the time showing how global attention can work in slice-based approaches as well.

The choice between a CNN-based model and a transformer may come down to available data and computational resources: CNNs can be efficient and easier to train on limited data, whereas transformers can unlock ultimate accuracy given sufficient data and computational resources. The method presented in this thesis is a lightweight (by today’s standards) CNN-based architecture, modified Attention UNet for segmenting vertebral bodies in low-dose CT images. The motivation for a shallow/lightweight model is due to limited computational resources.

Chapter 3

Methodology

In supervised learning tasks, particularly in medical imaging, a persistent and common limitation is the scarcity of annotated data. This issue becomes increasingly pronounced in complex segmentation tasks, such as segmenting closely situated and anatomically similar structures, like vertebrae. This thesis proposes a solution to a challenging problem: segmenting 25 distinct vertebral body classes within CT images. Due to their similarity in shape and close proximity, accurate segmentation of these structures typically requires a large and diverse dataset to effectively train sophisticated neural network models. Furthermore, harder segmentation problems often demand models with greater representational capacity, resulting in increased computational resources needed.

Our particular challenge is exacerbated by the severely limited nature of our dataset, only having 36 low-dose CT images with very thick axial spacing. These unique dataset characteristics further restrict our ability to achieve accurate segmentation results. In response to these constraints, this chapter proposes a methodology designed to overcome data scarcity by leveraging the publicly available, high-quality VerSe dataset. I systematically modify the VerSe annotations [86–88] using connected component analysis and expected centroid enforcement to isolate and retain only the vertebral bodies, explicitly removing all structures posterior to the pedicles (see Fig. 2.1b). This is due to Dr. Holter et al. stating the vertebral body marrow cavities exuberate high

levels of stem cell proliferation post HSCT, where the other parts of the spine have low enough activity to not need to be considered [26–29].

To further bridge the domain gap and simulate conditions closer to our original dataset, extensive data augmentation techniques—including the downsampling of voxel spacing, Gaussian smoothing, the addition of noise, and elastic deformation—are applied during model training. Additionally, due to practical computational constraints (utilizing an RTX 4070 GPU with 12GB VRAM), I employ a custom, lightweight yet robust, Attention-Gated 3D UNet architecture specifically modified to manage resource limitations while effectively capturing the complexities of the segmentation task. While previous papers have attempted segmentation of individual vertebral bodies in this low-quality dataset [81–85], this work represents the first fully end-to-end neural network approach to predict these labels and compare the results to physician measurements. This is due to Nguyen [81], Carson [83] and Hurtado [85] proposing a multi-step approach, where an initial model segments the binary vertebral column as a whole, then a subsequent image processing technique delineates the vertebral boundaries.

Finally, the resulting CT segmentation predictions are interpolated onto co-registered PET images, which are obtained immediately after the CT scan, using the same machine. The predicted segmentations facilitate the extraction of standardized uptake value (SUV) statistics, which are subsequently utilized to predict potential relapse in HSCT patients.

3.1 HSCT dataset

In prior work using the HSCT dataset, the manual annotations—provided by a non-physician postdoctoral engineer—were limited to binary masks for the sternum, bilateral pelvis, and the entire vertebral column as a single unit. Consequently, each voxel was

classified only as “vertebral body” or “not vertebral body,” with no distinction between individual vertebrae. These studies therefore relied on separate algorithms to segment each vertebra from the unified vertebral column mask.

A key contribution of this work is the creation of individually annotated vertebral body labels for 28 CT volumes spanning 17 patients. Using 3D Slicer [31, 115], I manually assigned separate classes to each vertebral body (C1 through L5, plus transitional vertebra L6 if present). This process was extremely labor-intensive, due to the need for precise boundaries in thick axial slices. Initially, I attempted to exclude L6 from the labeling schema, but models trained without it failed to segment properly in patients who had that extra vertebra (this will be further evaluated in my comparison to TS in Ch. 4). Although the physicians in the pilot study did not examine these anomalous vertebrae [29], I found that including labels for them was critical for consistent network convergence.

Additionally, I took several steps to ensure consistency among all images. Since the original DICOM format can be cumbersome to handle in coding libraries, I first converted the scans into the Python-friendly NIfTI format as suggested by Hurtado [85]. I also discovered that a subset of images were acquired in an oblique orientation and therefore resampled all of them to the standard LPS orientation. Further normalization techniques specific to network training are described in the Model Training subsection.

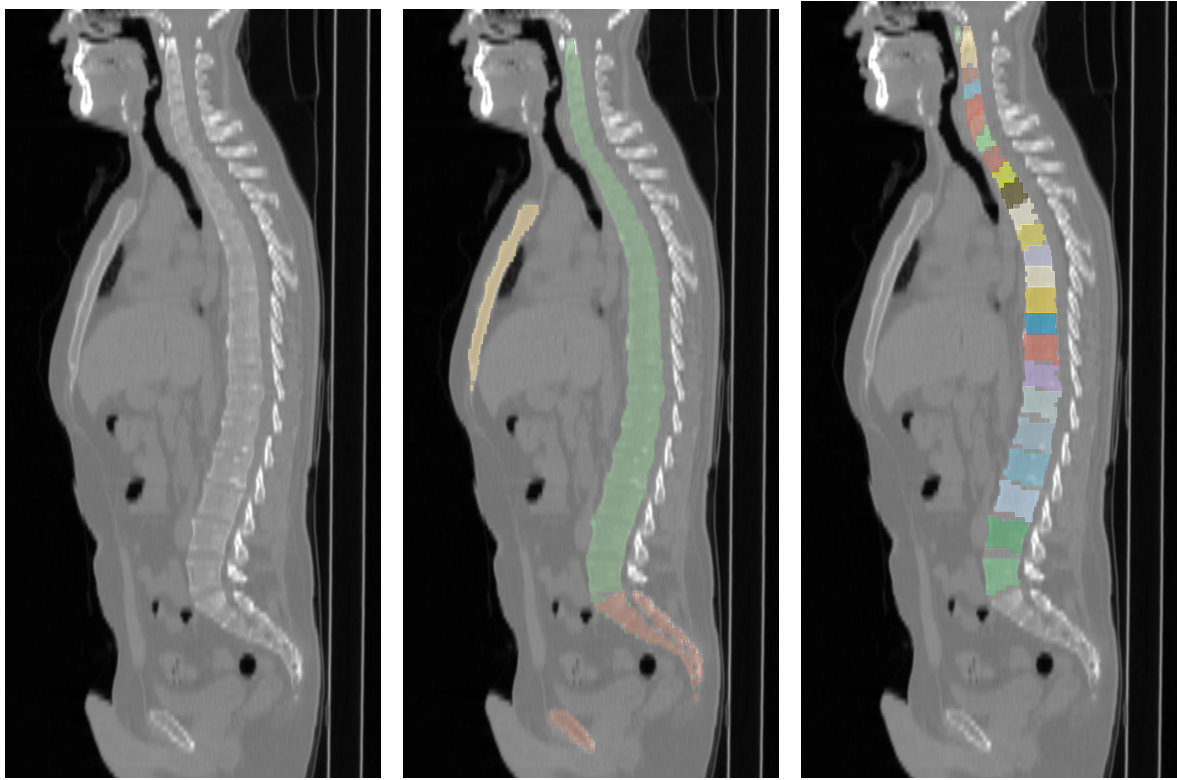
Figure 3.1a below illustrates how difficult it is to distinguish intervertebral boundaries above the lumbar region, even for trained physicians. Figure 3.1b shows the original three-class labeling (sternum in yellow, bilateral pelvis in red, and spinal column in green), which lacks detail regarding individual vertebrae or intervertebral disc space. By contrast, Figure 3.1c highlights the new multi-class labels, where each vertebra is annotated separately (C2 is the yellow label at the top, and L5 is the green label at the bottom), enabling more precise segmentation. While it is apparent that previous works

could segment other bones in the images, this work is primarily focused on segmenting individual vertebral bodies, which present a particular challenge under the given CT dosage constraints.

A recurring challenge in medical image segmentation tasks, including this dataset, is class imbalance. In the case of vertebral body segmentation, this imbalance arises in three main ways:

- **Vertebral volume vs. background:** In this dataset, the labels pertaining to all vertebral body voxels make up 1-3% of the total volume, meaning the vast majority is background.
- **Size variation among vertebrae:** The lumbar vertebrae are physically larger than cervical vertebrae, enabling them to contribute a dominant signal in supervised learning tasks.
- **Differences across slice planes:** In full-body scans like the ones in this dataset, nearly every axial slice contains vertebra voxels. However, sagittal and coronal views typically include few to none proportionally.

Having described the labeling methodology, it is also important to note the specific imaging parameters of the HSCT dataset. Each CT volume was acquired axially at a resolution of 512×512 pixels, with anisotropic voxel spacing of $1.17 \text{ mm} \times 1.17 \text{ mm} \times 5 \text{ mm}$. In contrast, the FLT-PET volumes were acquired axially at 144×144 pixels and an isotropic $4 \text{ mm} \times 4 \text{ mm} \times 4 \text{ mm}$ spacing. Nominally, this gives the PET scans a slightly finer spacing in the axial plane, but the low metabolic activity on days 1 and 2 after HSCT substantially diminishes the visible detail in the PET images. Consequently, the CT volumes remain the primary modality for accurate segmentation of individual vertebral bodies.



(a) CT sagittal view.

(b) Old binary labels.

(c) New multiclass labels.

Figure 3.1: A sagittal slice of a typical patient in the dataset and segmentation labels.

There is often an inevitable need for more data in relatively complex tasks such as semantic segmentation of 25 classes in order not only to ensure model convergence, but also to mitigate class imbalance. The next section describes data augmentation process I developed to make the VerSe dataset homogeneous with our dataset, allowing it to be feasible for training data supplementation.

3.2 VerSe Augmentation

The VerSe dataset is especially valuable for its individually annotated vertebrae, which provide clear boundary distinctions. In addition, it offers nearly 400 image-label pairs, making it a significant resource for spinal semantic segmentation. Although the diversity

in VerSe can increase the difficulty of the task, it can also yield more robust models if handled properly. In this thesis, I propose that supplementing our niche, small HSCT dataset with VerSe data can improve both training convergence and final segmentation performance/robustness on low-dose CT vertebral bodies. The main challenge lies in the mismatch of anatomical labels. While my HSCT annotations focus exclusively on vertebral bodies, VerSe annotates the entire vertebra, including processes, pedicles, and laminae. Blindly including these structures without modification would effectively confuse the model about the segmentation task. To address this, I present a systematic method to remove the processes from the VerSe labels while preserving the original vertebral body annotations.

In spinal anatomy, the pedicles are thin, narrow bones that connect the front of a vertebra (vertebral body) to the back (lamina) (see Fig. 2.1b). Since these structures are much thinner than the surrounding bones, I hypothesized that the simple, yet effective approach of connected component analysis could erode the pedicles, separating the anterior and posterior parts of the spine labels.

Morphological operations are image processing techniques that probe and transform binary (or grayscale) images using the predefined shape of a structuring element. Two of the fundamental morphological operations used in this section are binary erosion and dilation. *Erosion* is an operation that removes pixels on object boundaries if they do not fit entirely in the structuring element. It is defined in [116] as

$$A \ominus B = \{z | (B)_z \subseteq A\}, \quad (3.1)$$

where A is a binary image,, B is the structuring element, z is a vector, and B_z is the translation of the structuring element B so that its origin is at point z . Formally, z looks like this:

$$B_z = \{b + z | b \in B\}, \forall z \in E, \quad (3.2)$$

where E is some space (often $\mathbb{Z}^2$, or a 2D integer space). On the other side, *dilation* is the duality: it adds pixels to the boundaries if they intersect the element, or inflates the image. It can be represented by:

$$A \oplus B = \{z | (B^S)_z \cap A \neq \emptyset\}, \quad (3.3)$$

where B^S is the reflection of B . In simpler terms, this equation says for every point z in E 's image grid, it shifts the structuring element B so that its origin is at z 's new location. Then, it looks to see if B touches any foreground pixel of A . If so, z becomes a foreground pixel in the new dilated image. *Binary opening* is the process of performing erosion followed by dilation in succession using the same structuring element. In essence, it removes narrow connections in the foreground and ‘salt’ noise which is ideal for separating a vertebral body from the protruding, posterior processes.

A *structuring element* is a user-defined probe with a specific shape, employed in morphological operations to systematically traverse the image. Its choice can significantly affect the outcome. I selected a *disk*-shaped structuring element because it more closely matches the rounded contour of vertebral bodies and avoids the artificial corners that a square element would introduce. Furthermore, I dynamically adjusted the size of this structuring element on a per-slice basis, guided by the pixel area of the binary label. This approach was twofold in rationale: (1) vertebrae vary widely in volume along the spine—lumbar vertebrae are much larger than cervical ones—and (2) an element that is too large will erode too many details during opening, while one

that is too small will have little effect on the image. It is worth noting that during operation, I oriented every label volume to the RAS+ orientation so that the direction the vertebrae faced match the shape of the structuring element closely.

A notable limitation is that the radius, although updated dynamically on a slice-by-slice basis, was hard-coded using empirically determined values stored in a lookup table. Future work should investigate more principled or adaptive approaches to improve the robustness of this parameter.

Next, since the opening operation on a typical slice will separate the vertebral body from the processes into ≥ 1 components, I need to perform *connected component analysis* on it. This is so that I can evaluate which component is the body or not. Using the *scipy.ndimage* (ndi) python library, I scan the resulting ‘opened’ image and group adjacent 1-pixels into *components*. Adjacency is typically defined by an 8-neighborhood where each connected set of 1-pixels is assigned a unique integer label i . Then, ndi’s `sum_labels` computes for each i the sum of pixel values. Since the manually annotated labels I am modifying are binary for each class, the sum is simply the area in pixels of each connected component in each slice. This can be summarized by:

$$\text{size}(i) = \sum_{(x,y):\text{connected}(x,y)=i} \text{opening}(x,y). \quad (3.4)$$

Typically, the vertebral body is the largest part of the spine in any given slice so a naïve approach would be to always select the largest component and return that for every slice. However, due to outliers, partial volume effects, and the general variability found in anatomy, this is not always the case. Especially near the top and bottom of the vertebra, the processes can be larger than the body. So, a different way to pick the correct component more reliably was needed. I originally attempted to analyze the contours and other metrics of ‘roundness’ to pick the component closest to a perfect

circle, using tools such as Fourier analysis. This was an ineffective approach for several reasons: again, natural variability means that some vertebral bodies (especially the cervical ones) will be irregular and will not assume an oval shape as closely as the lumbar region might. Moreover, partial volume effects like changing slices or the morphological operation not completely removing the process as well as noise can introduce extra ‘sharpness’ in a given slice. This being said, I decided to use a selection procedure that is more robust to shape variance.

The *centroid*, or center of mass, is the average location of all pixels belonging to a particular object in a binary image. In mathematical terms, for a 2D binary image I , the centroid $(\bar{x}, \bar{y})$ of a connected component C is defined in the Scipy library [94] as:

$$\bar{x} = \frac{1}{|C|} \sum_{(x,y) \in C} x, \quad (3.5)$$

$$\bar{y} = \frac{1}{|C|} \sum_{(x,y) \in C} y, \quad (3.6)$$

where $|C|$ is the total number of pixels in a given component. Since the image is binary, each foreground pixel gives an equal weight of 1. The result is a single point in 2D space that corresponds to the center of the label. I leveraged the fact that vertebrae do not move much in between neighboring slices, so I took a moving average of several previous slices to get a rough *expected* centroid location for the current slice. This can be given by:

$$\bar{x}_{avg} = \frac{1}{k} \sum_{i=1}^k \bar{x}_i, \quad (3.7)$$

where the last k non-empty slices have centroids $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_k$. Then, I compute the centroid of each labeled connected component. Finally, I compute the absolute difference between each candidate centroid and the ‘expected’ location along with a thresholding to find the closest match. In the event that none meet the threshold, the largest

component is taken as a default. The selected component's centroid is then added to the running statistics, which creates a *rolling average* to make the expected centroid calculation dynamic. This approach is typically more robust than any contour detection because the vertebrae move in a predictable way, especially in the x direction (assuming RAS), for the vast majority of patients.

Even though morphological operations are good in theory at preserving most of a target's shape, they are not perfect. To preserve as much detail from the original annotation as possible, I find the most positive x pixel in the selected component and multiply everything to the left of that by the original label. This ensures that the erosion portion of the opening operation does not remove too much of the boundaries. Finally, all slices are stacked in an empty volume with the same dimensions as the original label and saved in the LPS orientation to ensure consistency in libraries that force the LPS standard, such as Simple ITK and 3D Slicer.

After discussing the modifications done to the labels, the way I chose a subset of VerSe should be addressed. I did not employ my algorithm on every available image. Instead, I picked a subset of the images that most accurately represent my dataset. A large percentage of the VerSe dataset contains images of *only* the lumbar vertebrae. As will be shown later, the lumbar are the easiest classes for the network to segment, due to their distinct, predictable shape along with clearly defined boundaries even in low-quality CT images. Therefore I hypothesized that adding every lumbar case into training could actually hamper performance by making the class imbalance larger for the thoracic and cervical vertebrae. Moreover, I selected every image that contained the transitional vertebra L6, but did *not* include any samples with fused or sacralized L5 (effectively meaning that the L5 class would be absent from the labels). My reasoning for the latter statement is that I could not discern any patients in the HSCT dataset

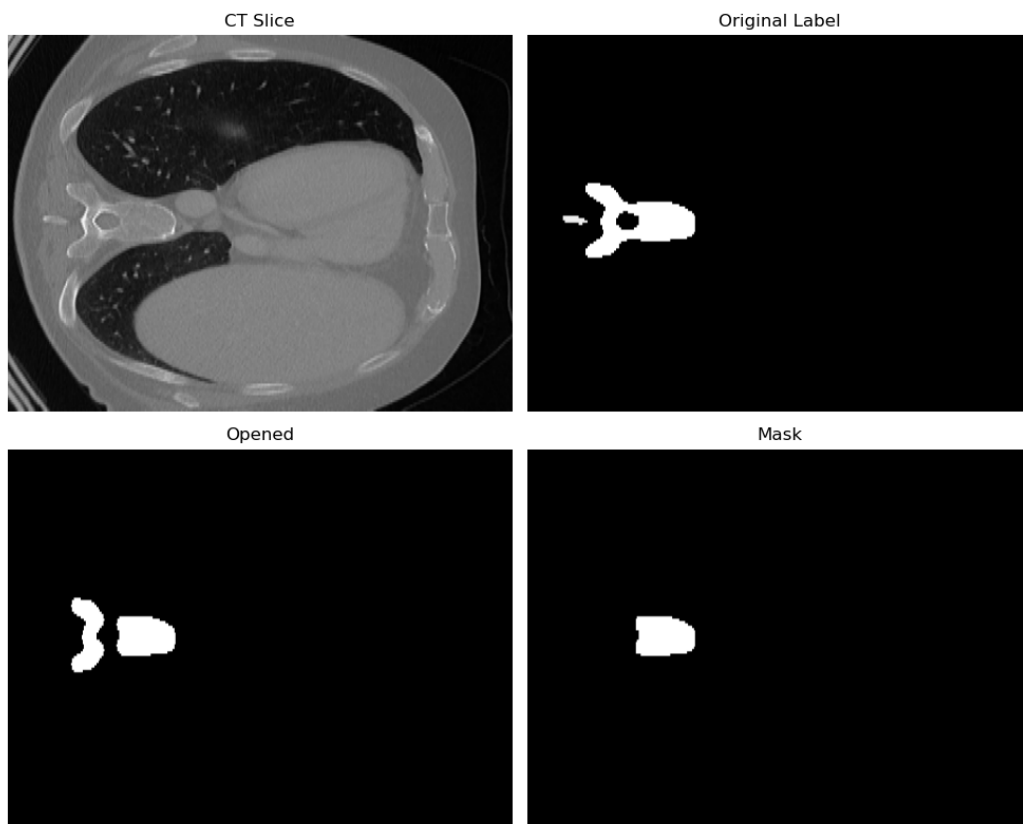


Figure 3.2: Example slice of connected component algorithm employed on a sample VerSe image. An opening operation on the thin pedicles separate the vertebral body from the rest of the spine.

that had a fused vertebra. Therefore, it did not make sense to include them as I wanted the network to do as well as possible on our narrower dataset.

The final thing to note about my modification of the VerSe dataset is sampling. The images in the VerSe dataset have a wide range of voxel spacings, from 0.3 mm-2.5 mm. However, it still remains that none of these have nearly the thick spacing as the HSCT dataset. To employ a network that is trained on our dataset's native resolution, I downsampled every VerSe image to 1.17 mm x 1.17 mm x 5 mm. This was done using MONAI's Spacingd transformation with trilinear interpolation. The results can be seen below, in Figures 3.4-3.5. Additional 2D figures as well as 3D representations



Figure 3.3: The same slice as in Fig. 3.2 with the final mask overlaid in white on the CT image.

of the modified VerSe labels can be viewed in Appendix B, where I provide many supplementary visualizations.

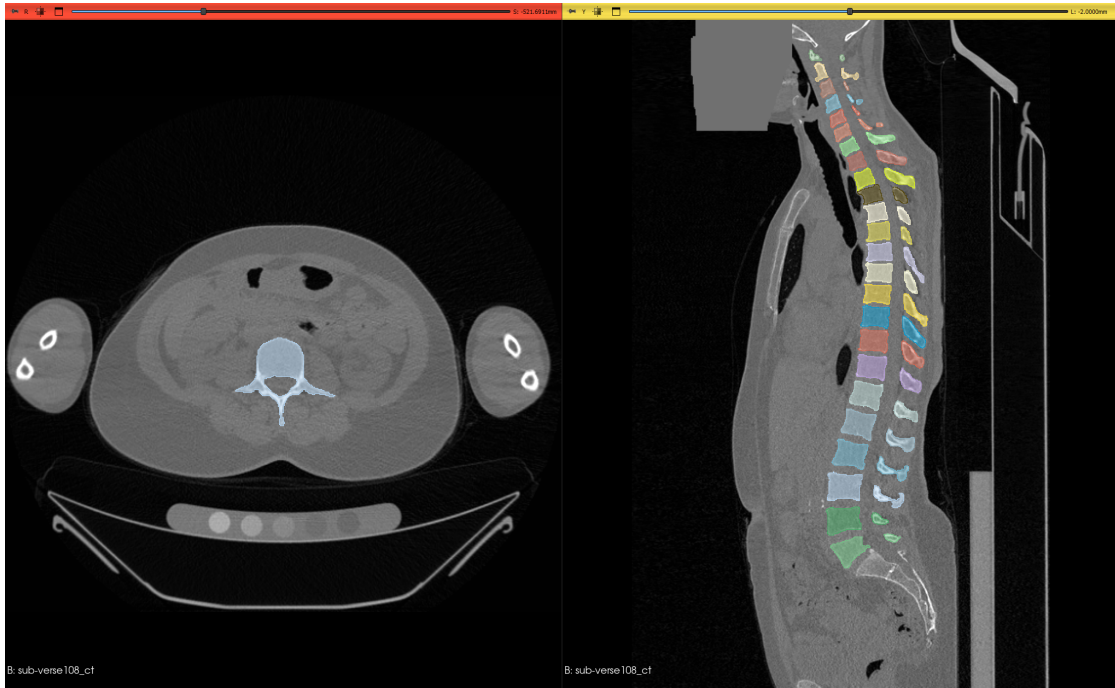


Figure 3.4: A VerSe sample in its native spacing with unmodified labels overlaid on top. The red bar indicates axial slice, and yellow indicates sagittal.

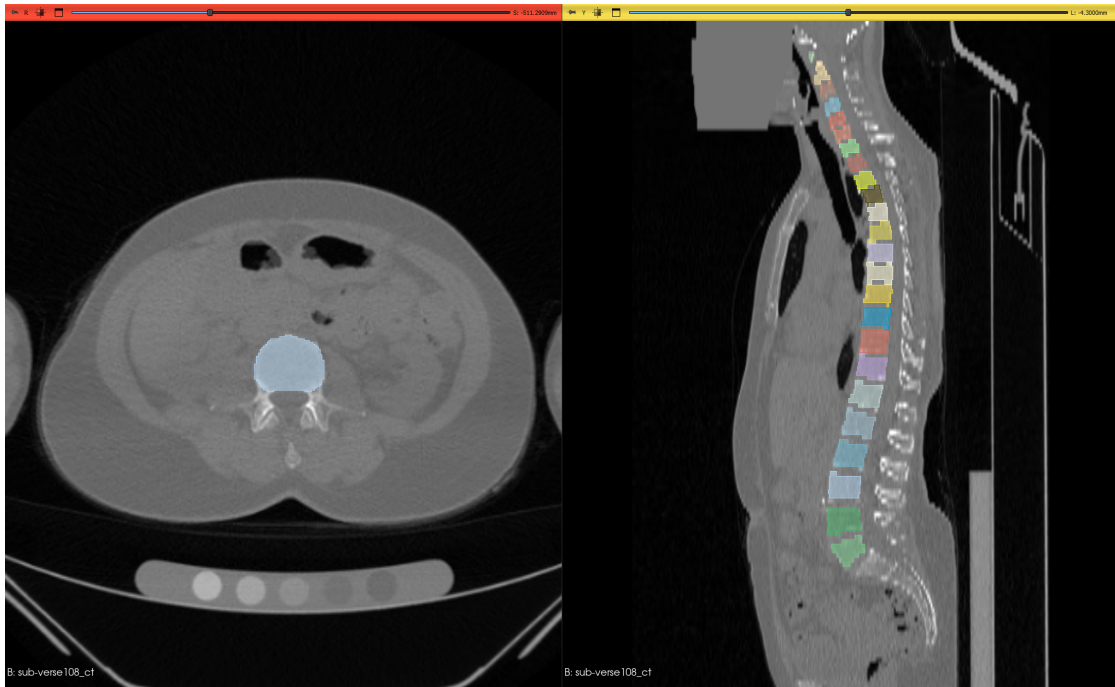


Figure 3.5: The same VerSe image downsampled to 5 mm axial spacing, with the vertebral body intact and the rest of the spine annotation removed.

3.3 Attention-Gated UNet

For this project, I developed a custom, modified UNet architecture tailored to volumetric segmentation tasks. The backbone of this network remains fundamentally a classical UNet architecture, utilizing a symmetric encoder-decoder structure interconnected by skip connections, as proposed in [70, 71]. To further enhance the network’s capability, I incorporated attention gates into the skip connections, similar to the methodology in [78]. These attention gates refine feature maps and skip connections by emphasizing clinically relevant spatial features, which I found to be helpful given the task of distinguishing between a large number of visually similar classes.

However, unlike the conventional Attention UNet in [78], I replaced the standard convolutional layers with residual blocks. These residual units improve gradient flow, mitigate the vanishing gradient problem commonly encountered in deeper networks, and enhance convergence stability during training. Additionally, another modification involved the integration of group normalization after each convolution. As previously explained, my training scheme had memory constraints, limiting the batch size to a single sample per iteration. Group norm was specifically chosen due to its statistics and normalization parameters being invariant of batch size, helping with stability.

Furthermore, I introduced a deep skip connection inspired by recent developments in recent works like [117, 118]. This deep skip connection merges feature maps from multiple encoder layers, enriching the decoder input with multi-scale context and giving richer feature representation. The selected activation function is Leaky ReLU, which is thought to improve performance compared to ReLU [119] by allowing negative gradients to flow. Finally, I used relatively substantial dropout in most of the encoder layers to help prevent overfitting.

While this network employs many clever ideas such as residual units, skip connections, attention gates, and 3 dimensional operations, it is still a intentionally lightweight design given the computational restraints imposed on my research. There have been many studies conducted attempting to reduce parameter count for medical image segmentation tasks such as [120, 121]. To place the size of my model in context with relevant architectures, refer to Table 3.1 below. It should be noted it is difficult to find an exact parameter count of the 3D UNet, since papers will implement different variations for their comparisons. For example, in [113], they report the UNet to have 28.27 M parameters while [120] reports only 15.56 M. One of the simplest and most likely explanations for the difference is the depth of the UNet used. My proposed model only goes to 256 layers, while others may go to 512 or even 1024 layers.

Table 3.1: Estimated parameter count for several notable segmentation models.

Method	Param count (M)
This work	4.84
3D-UNet [71]	~15-30
Swin-UNet [110]	82.3
Verteformer [113]	146.68

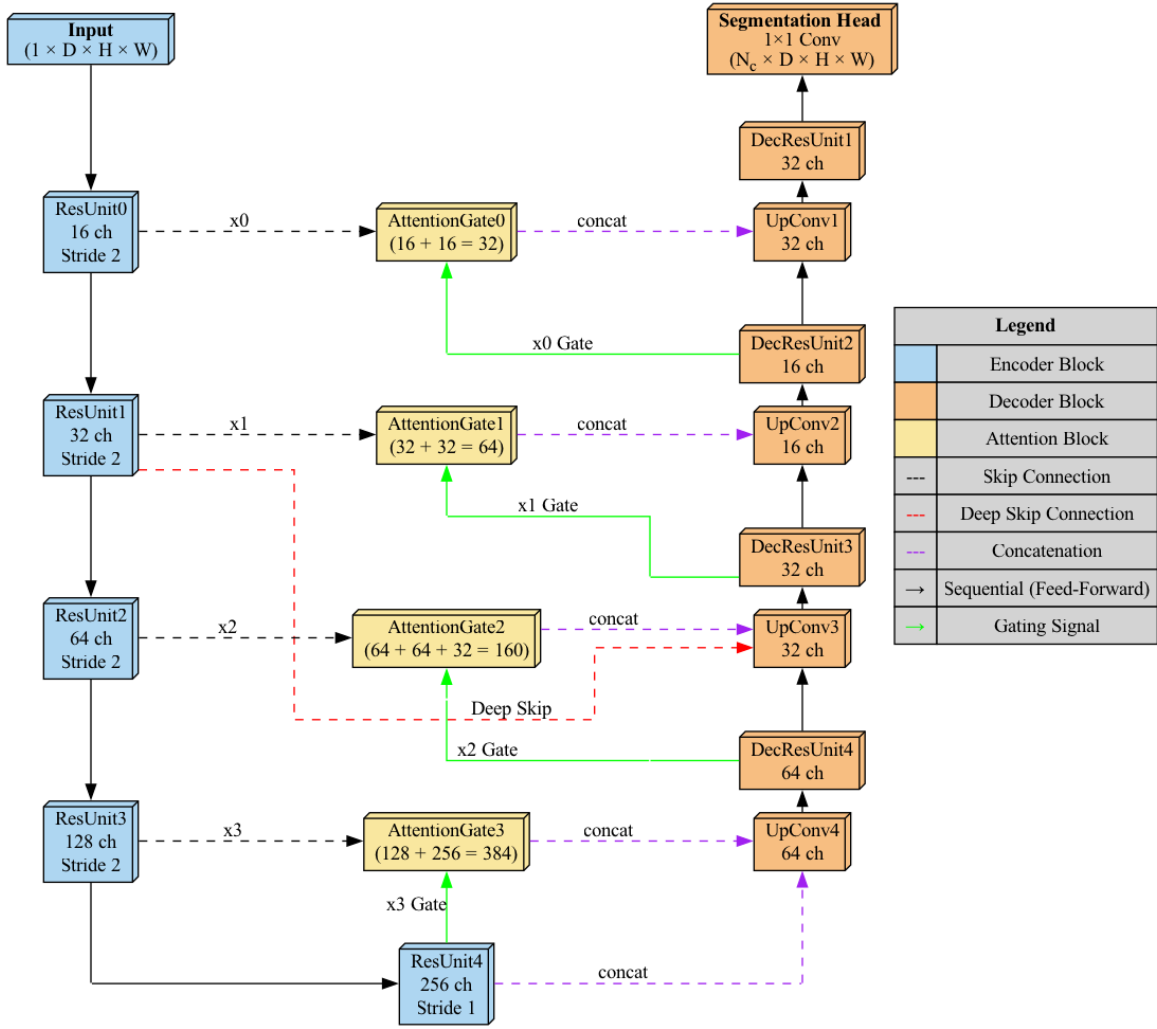


Figure 3.6: Diagram featuring my custom Attention 3D UNet architecture. N_c represents the number of classes.

3.4 Model Training

This section will give an overview of the techniques I employed to train my architecture, ensuring I get the best performance I can given the constraints mentioned already.

For my loss function, I explored various options including dice loss, focal loss [122], and cross entropy (CE). I settled on a combination of dice and CE, dubbed by the MONAI [91] library as DiceCELoss. It is given by:

$$\mathcal{L}_{DiceCE} = \lambda_{Dice}\mathcal{L}_{Dice} + \lambda_{CE}\mathcal{L}_{CE}, \quad (3.8)$$

where the Dice loss in terms of multiclass segmentation is given by:

$$\mathcal{L}_{Dice} = 1 - \frac{1}{C} \sum_{c=1}^C \frac{2 \sum_i p_{i,c} y_{i,c}}{\sum_i p_{i,c} + \sum_i y_{i,c} + \epsilon}, \quad (3.9)$$

where C is the number of classes, $p_{i,c}$ is the predicted probability of voxel i belonging class c , $y_{i,c}$ is the one-hot encoded binary ground truth for the class c that voxel i belongs to, and ϵ is a small term added to prevent division by zero. The cross entropy term is given by:

$$\mathcal{L}_{CE} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{i,c} \log(p_{i,c}), \quad (3.10)$$

where N is the total number of voxels in a given volume. This normalizes the term so that the CE weight is consistent across different image sizes.

I found that carefully selecting the weighting coefficients λ allowed me to leverage the complementary strengths of the Dice and CE loss functions, achieving better performance than using either alone. Specifically, Dice loss excels at handling class imbalance, provides global context awareness, and directly optimizes the Dice validation metric utilized in this project. However, Dice loss can sometimes result in instability, especially in early training stages. In contrast, the CE loss is effective at promoting voxel-level accuracy and helping to refine local boundaries, both of which are important for accurately delineating individual vertebral bodies.

To rigorously evaluate the performance of my segmentation model while preventing data leakage, I employed Sklearn’s GroupShuffleSplit with a fixed random seed, ensuring that multiple scans from the same patient remained mutually exclusive between training and validation subsets and that the HSCT train/validation split remained identical across runs. My dataset comprises 130 modified VerSe images (with posterior processes removed, as detailed in a previous section) and 28 HSCT images, of which 5 HSCT scans were explicitly reserved for validation purposes. Initially, I trained the model solely on the remaining 23 HSCT images. Subsequently, I reran the training including the 130 VerSe images, maintaining the exact same hyperparameters and the same HSCT train/validation split. Given that the primary goal of this study is to enhance segmentation performance on the custom HSCT dataset rather than the VerSe images, the latter were purposefully excluded from the validation subset.

For preprocessing and data augmentation, I utilized MONAI’s Compose function, which allowed me to build a structured yet highly customizable pipeline. This pipeline begins with normalization transformations, including adding channel dimensions, standardizing orientation to LPS coordinates, downsampling images to a consistent voxel spacing of $1.17\text{ mm} \times 1.17\text{ mm} \times 5\text{ mm}$, and scaling intensity values within a clinically relevant range (-100 to 250 Hounsfield Units). Additionally, foreground cropping and symmetric padding ensured consistent input dimensions, especially crucial for regions of interest.

Beyond these baseline normalizations, the pipeline incorporates an aggressive set of randomized augmentations applied on-the-fly during each training epoch. These augmentations include:

- Patch-based cropping (via RandCropByPosNegLabel) to enrich positive sample representation. I set the patch size to a non-uniform $96 \times 96 \times 64$ pixels since

the thicker z spacing means that it takes up a larger proportion of each image compared to x and y.

- Random zooms (80–120% of original size) to simulate slight variations in field-of-view and scale.
- Random flips and rotations in the axial plane to account for anatomical variations.
- Elastic deformations, which simulate realistic anatomical variations and help the model generalize to patient-specific anatomy.
- Intensity-based transformations such as random intensity shifting, Gaussian noise addition, and contrast adjustments, further enhancing the robustness of the model against variability in imaging conditions.

To optimize model training, I chose the AdamW optimizer due to its effectiveness in achieving faster convergence and better generalization compared to traditional Adam [123], particularly in deep convolutional architectures like UNet. I coupled AdamW with a CosineAnnealingLR scheduler, configuring it with an initial learning rate of $1e-4$, an `eta_min` value of $1e-5$, and a `T_max` parameter set to one-quarter of the total number of epochs, allowing it to cycle twice. This scheduler progressively decreases the learning rate following a cosine decay pattern, enabling the model to initially learn rapidly and subsequently refine its convergence. Although the model performance typically peaks before the designated 1000 epochs, I intentionally trained for this extended period to demonstrate the model’s convergence stability and robustness against overfitting. Furthermore, I employed mixed-precision training techniques—specifically PyTorch’s GradScaler and autocast functionalities — to optimize GPU memory usage and accelerate training without sacrificing accuracy.

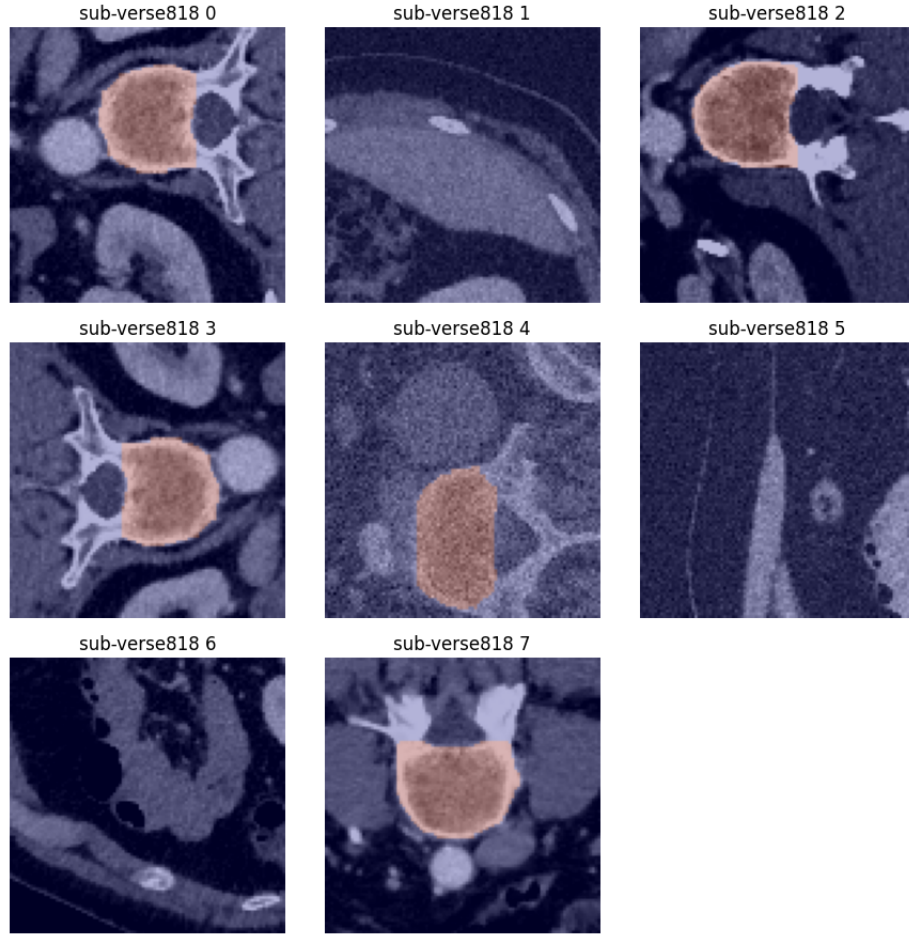


Figure 3.7: Example batch of patch-based training, showing various augmentations like flips (3) and Gaussian noise (4) and a foreground to background ratio of roughly 3:1.

Finally, I ran this training scheme over just the 22 HSCT images and then again with the 152 total images. Ch. 4 will demonstrate that the inclusion of VerSe enables faster convergence, significantly higher validation performance, better stability, and increased robustness.

3.5 SUV extraction

Everything in the methodology thus far has been describing the process I created to segment individual vertebral bodies in the CT scans of HSCT patients. The final aim

of this whole work is to predict vertebrae in a CT scan, propagate that segmentation to a co-registered PET scan, perform statistical analysis on the SUV ROIs of each vertebral body, and use that information to predict possible relapse in these bone marrow transplant patients. This section details how I transferred the segmentations predicted by my custom Attention-Gated UNet model to the PET space, what data was extracted from which anatomical regions, and how this information may be used to predict possible relapse.

In our HSCT dataset, the PET and CT images are *co-registered*. This means that they are taken nearly simultaneously using the same physical machine, and that the images are aligned spatially so that anatomical and physical points match up in both volumes. For example, in a CT scan, if the center of L5 is located 20 mm away from the origin, then it will roughly be 20 mm away from the origin of the corresponding PET scan as well. Note, however, that I said ‘roughly’ as this process is not perfect. If there is slight patient movement or other small issues, they can still be misaligned by a little. Or, as in this case of our dataset, the images might have different voxel spacings. The (x, y) spacings have a difference of 3 mm (1.17 mm in CT to 4 mm in PET), while the (z) direction is actually 1 mm better in the PET (5 mm to 4 mm). This means that in order to propagate any labels that a model segments on the CT image, I will need to perform interpolation. Nearest neighbor as chosen, as it does not introduce extra, fractional label values and keeps them discrete (compared to methods like linear, cubic, or B-spline). Nearest neighbor interpolation, as shown in the Nibabel library [38], can be expressed as follows:

$$(i_{NN}, j_{NN}, k_{NN}) = \arg \min_{(i,j,k) \in \mathbb{Z}^3} \sqrt{(x-i)^2 + (y-j)^2 + (z-k)^2}, \quad (3.11)$$

where the $\arg \min$ term describes the Euclidean distance between continuous coordinate (x, y, z) and discrete, integer label coordinate (i, j, k) , and (i_{NN}, j_{NN}, k_{NN}) is the nearest voxel index. Then, after finding the nearest voxel,

$$\text{Label}_{\text{NN}}(x, y, z) = \text{Label}(i_{NN}, j_{NN}, k_{NN}). \quad (3.12)$$

Eq. (3.12) says that you simply assign that voxel to the new image space.

Interpolation introduces smoothing, loss of detail, and the partial volume effect. These issues are outside the scope of this thesis, but can be addressed in future works to help make the SUV extraction more accurate. Because of the inherent limitations with interpolation, I will provide error measurements between the physician data and the ground truth, error between physician measurements and predicted labels, and the difference in error between the ground truth and predicted segmentations. This will be helpful in establishing a baseline error between the ground truth and physician measurements.

Since the SUV data contained within the PET images is the most important, it may be natural to ask why we do not attempt to directly segment the vertebral bodies in the PET space. The most important reason for this is that cell proliferation is significantly reduced during initial days of imaging (in our dataset, days 1 and 2). This is due to the chemotherapy killing most cancerous and healthy cells and there hasn't been enough time for the stem cells to take effect. Since there is a vastly reduced amount of detail, it is nearly impossible to distinguish anatomical structures. This can clearly be seen in Fig. 3.9 below. Additionally, the x and y directions are significantly coarser in the PET image, leading to a loss of detail that was captured in the CT scan.

While my method can segment all vertebral bodies - from C1 down to L5 -, Dr. Holter only provided physician-extracted SUV data for C7-L5. This was done due to the

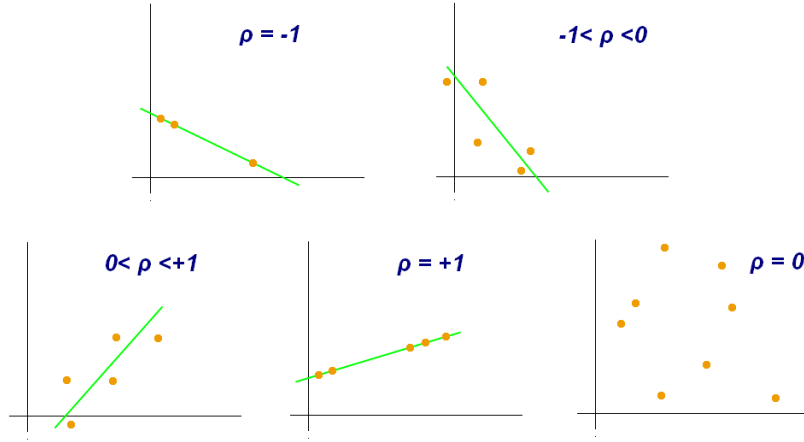


Figure 3.8: Typical correlation coefficient trends. Extracted from [124].

small size and irregular structure of the other cervical vertebrae [29]. Therefore, my method will also only extract the SUV data for these bones. I will find the voxel ROIs using nearest neighbor interpolation and compute the statistics: mean, median, max, min, and standard deviation (STDV). I will then compute error metrics to assess the final performance of my model compared to physician data. Mean absolute error (MAE) is good for telling the average magnitude of errors without direction. Root mean squared error (RMSE) is similar, but it is more sensitive to outliers, which is good for emphasizing substantial deviations. Ideally, both of these metrics should be as close to zero as possible. Finally, I will use the Pearson correlation coefficient (PCC) [125] to capture trends and see how well of a linear relationship exists between the ground truth and predictions (see Fig. 3.8). It complements the previous metrics since it can help show the consistency of the model. The closer to -1 or 1 correlation is, the more closely a prediction represents trends in ground truth. Correlation is especially useful in my case because as scans progress, there will be a predictable increase in cellular metabolism uptake, which will be able to show if my model can track the same trends as the physician's measurements.

MAE is given by:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i|, \quad (3.13)$$

where N is the number of voxels in the ROI, y_i is the the physician-given SUV value for the i -th voxel, and $\hat{y}_i$ is the predicted SUV value. While this is the general version for voxel-wise comparison, it is similar to comparing statistical values - I did not have access to per-voxel SUV measurements or raw PET data from the physicians.

RMSE is given by:

$$\text{RMSE} = \frac{1}{N} \sqrt{\sum_{i=1}^N (y_i - \hat{y}_i)^2}. \quad (3.14)$$

Pearson correlation for a *sample* is defined as covariance over the standard deviations. In more specific terms, covariance between X , the physician's measurements, and Y , the model's predictions, is given by:

$$\text{Cov}(X, Y) = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y}), \quad (3.15)$$

where $(\bar{x}, \bar{y})$ are the means of X and Y , respectively.

The standard deviation of X is given by:

$$\sigma_X = \frac{1}{N} \sqrt{\sum_{i=1}^N (x_i - \bar{x})^2}, \quad (3.16)$$

and is defined similarly for Y . Putting all these together, Pearson correlation is given as:

$$r = \frac{\text{Cov}(X, Y)}{\sigma_X \sigma_Y}. \quad (3.17)$$

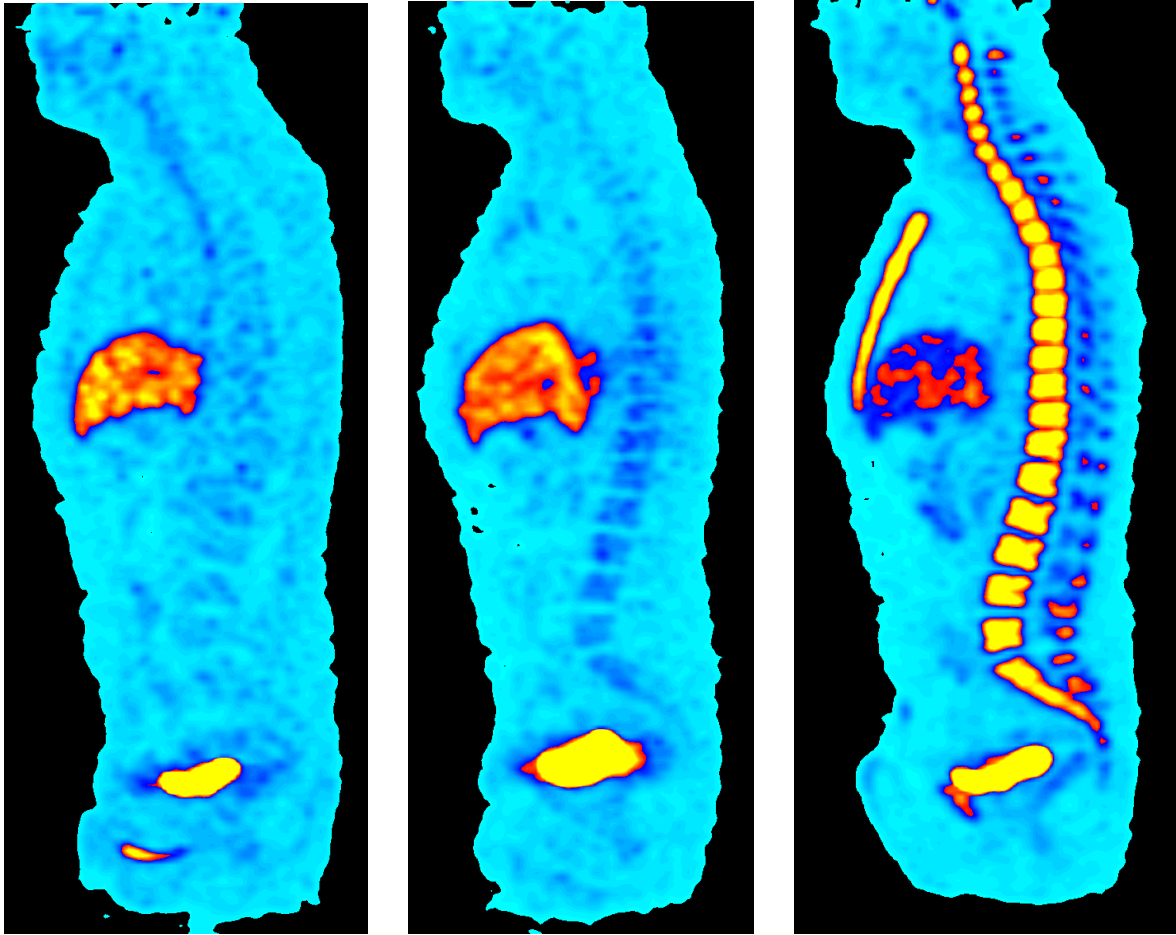


Figure 3.9: Subsequent FLT-PET scans showing an increase of cell proliferation from left (day -1) to right (day +28). There is also visible metabolism in the sternum and pelvis, which are explored in other works like [83, 85].

Chapter 4

Results

In this chapter, the performance of the developed segmentation method is thoroughly evaluated, both quantitatively and qualitatively. This work introduces the first fully automated, end-to-end segmentation algorithm capable of delineating individual vertebral bodies specifically in our low-dose, thick-spacing FLT-PET/CT dataset for HSCT patients. Unlike previous approaches that first segment the vertebral column as a binary region and subsequently delineate boundaries through additional algorithms, the proposed Attention-gated 3D-UNet model directly predicts individual vertebral bodies using machine learning exclusively. A crucial component of this thesis was augmenting labels from the VerSe dataset to supplement the training data. Without this augmentation, training solely on a small set of 22 images, the model exhibited poor convergence and limited accuracy. Incorporating augmented VerSe data significantly improved the validation performance, resulting in a more robust and generalizable model. To validate the model’s performance, segmentation accuracy is assessed using established metrics, including the Dice coefficient, which provides insight into the overlap between predicted segmentations and ground truth annotations.

Additionally, the model is explicitly trained to handle cases involving transitional L6 vertebrae, as these were relatively common in the patient cohort. However, it was not trained to handle fused vertebrae scenarios, such as cases where L5 is absent or only 23 ‘floating’ vertebrae are present. This limitation arose because sacralization was encountered in only one test patient, compared to at least four patients exhibiting an

L6 [29]. While extending the training set to include additional annotated cases from the VerSe dataset that address fusion scenarios could easily overcome this limitation, it was not deemed necessary within the scope of this work.

First, quantitative results are presented, detailing model’s performance via training loss and validation metric curves both with and without supplemental VerSe data. Also, I provide an abbreviated, per-region set of dice scores to show where the model excels, and tends to struggle more. A detailed per-class dice score is listed in Appendix A. Next, qualitative analyses provide visual evidence of the model’s segmentation capabilities, including visualizations of segmented vertebral bodies and custom-made volumetric difference maps highlighting areas of accurate and inaccurate predictions.

To contextualize these findings, the performance of the proposed method is compared against Total Segmentator [106, 107], a widely recognized and publicly available segmentation model. This comparison includes scenarios specifically designed to highlight the proposed model’s robustness in handling lower-quality images and anatomical anomalies, areas in which it notably outperforms existing solutions.

Finally, the results from the SUV extraction process are presented and analyzed. SUV measurements derived from model predictions are compared against physician-measured statistics. Detailed error analysis sheds light on the model’s reliability and discusses implications of these findings, highlighting specific limitations due to interpolation errors and inherent differences in the two imaging modalities.

The collective analysis in this chapter provides comprehensive evidence supporting the utility of the developed segmentation and SUV extraction framework, laying the groundwork for practical clinical applications in monitoring HSCT patients and potentially aiding in early relapse detection.

4.1 Model Performance

This section will be split into three parts: quantitative, qualitative, and comparative results. In the first subsection, I will detail the numerical performance of my model. This will be followed by graphical illustrations and a qualitative analysis of the segmentation performance. Finally, I will compare my model to a state-of-the-art CT segmentation model.

4.1.1 Quantitative Results

The results of my model on its highest performing iteration of the validation set is provided in Table 4.1, where the best metric scores are in bold. The table is grouped into the average Dice score per anatomical region for conciseness. Refer to Appendix A for a complete table denoting individual class Dice scores and comparison to a public model.

Table 4.1: Average Dice scores grouped by anatomical region, supplemented by estimated standard deviation from the mean.

Patient	Cervical	Thoracic	Lumbar	Overall
02 Day 1	0.74 ± 0.07	0.869 ± 0.03	0.914 ± 0.03	0.832
08 Day 1	0.725 ± 0.07	0.872 ± 0.03	0.921 ± 0.01	0.843
08 Day 3	0.726 ± 0.05	0.87 ± 0.04	0.873 ± 0.12	0.83
10 Day 1	0.703 ± 0.09	0.876 ± 0.06	0.926 ± 0.01	0.836
12 Day 2	0.752 ± 0.04	0.839 ± 0.07	0.928 ± 0.01	0.832

My model consistently performs very well on the lumbar region, usually exceeding 0.92. This is due to several factors, such as those vertebrae having the greatest anatomical consistency, the highest voxel space proportionally, the clearest boundaries, and the greatest number of labels present during training. This should not be too

surprising, as the intervertebral discs generally exceed the slice thickness, as reported in Chapter 2. However, it is quite exciting to see the high performance in the thoracic region, as this is where the boundaries begin to appear quite blurred and indistinguishable, even to trained physicians [29] where the model usually exceeds 0.87 with a relatively low deviation. However, and perhaps unsurprisingly, the model starts to have poorer performance on the cervical region, with higher deviation and 13% lower Dice score compared to thoracic. This is due to several factors, similar but opposite of the lumbar’s success. First, these regions by far occupy the smallest region proportionally, (approximately 7% on average compared to 45% of both lumbar and thoracic in our dataset). Additionally, the cervical region exhibits substantial anatomical variability: C1 and C2 lack the typical round body and posterior ‘spiky’ processes, and the cervical vertebrae in general appear distinctly ‘stacked’.

The slight erratic behavior of my model on transitional L6, as seen in P08D1 vs P08D3 in Table 4.1, is due to those cases being vastly outnumbered in training, even with the addition of VerSe (only 25 out of the 152 total training images have that label present). This class imbalance is still severe, even when using the Dice metric. However, the model still needed to learn what to do when an additional vertebra is present, so as not to confuse it. This will be shown quite clearly in the upcoming comparison to Total Segmentator.

In Fig. 4.1 below, I provided training loss per-iteration and validation metrics per-epoch for training the model over 22 HSCT images and then again with those same 22 HSCT images and 130 augmented VerSe images. The results are clear: there is a significant, positive impact upon the addition of VerSe cases. The ‘HSCT Only’ DiceCE loss settles into a value of approximately 0.9, while the validation Dice score barely peaks above 0.62, an indication of underfitting. Finally, the figure has a third green curve titled ‘Vanilla UNet’, which refers to an ablation study where I ran training with

all 152 images (HSCT+VerSe) on MONAI’s 3D UNet with no residual connections or attention gates, but otherwise identical parameters. This run was done to highlight not only the need of supplemental training data, but th

Conversely, the ‘HSCT+VerSe’ training run shows a similar loss but with greater variance, which likely arises due to the added anatomical diversity from VerSe, introducing beneficial complexity into the training process. More importantly, the validation Dice score demonstrates substantial improvement, stabilizing around 0.82. The similarity in final training loss between the two scenarios suggests that incorporating additional augmented data enables the model to better capture generalized and transferable features, rather than simply memorizing the limited training examples from the smaller original dataset. This highlights the value of data augmentation, even when incorporating images with inherently different characteristics such as resolution, spacing, and acquisition protocols.

Finally, the figure has a third green curve titled ‘Vanilla UNet’, which refers to an ablation study where I ran training with all 152 images (HSCT+VerSe) on MONAI’s 3D UNet with no residual connections or attention gates, but otherwise identical parameters (group norm, 256 layer depth, etc). This run was done to highlight that not only were the supplemental images important to proper convergence, but the residual connections and attention gates played a large role as well.

Appendix A shows an additional example of performance when training just the HSCT images, but doubling the iterations. The result of that figure proves that even with additional training iterations, the 22 images alone just cannot provide enough diversity for the model to converge properly.

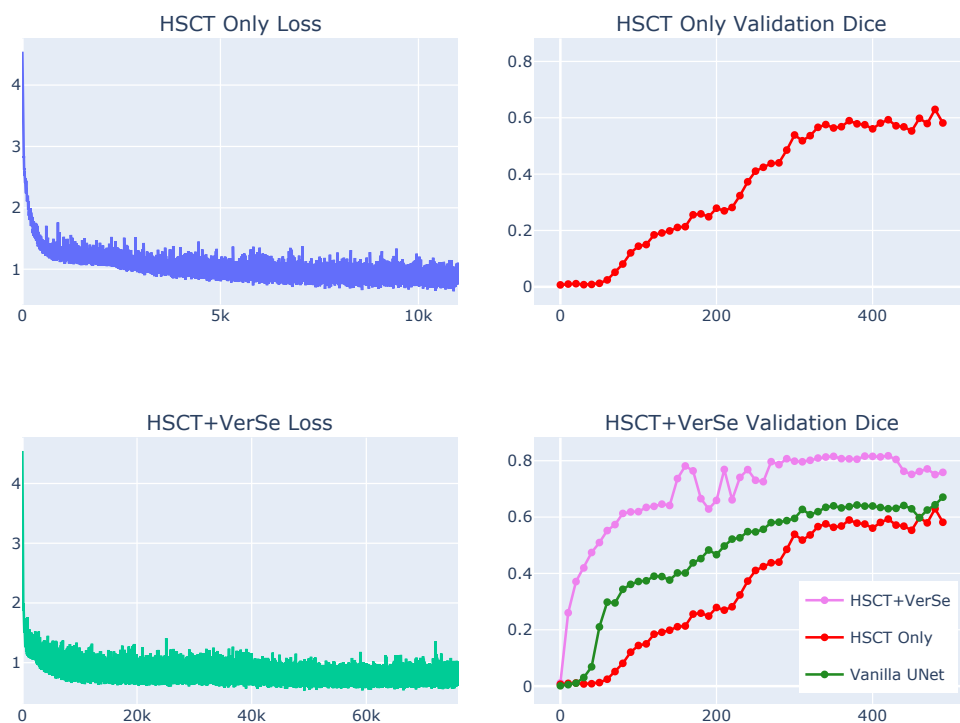


Figure 4.1: Training loss and validation metric curves when training on only 22 HSCT images, then again with VerSe samples included in a separate training run.

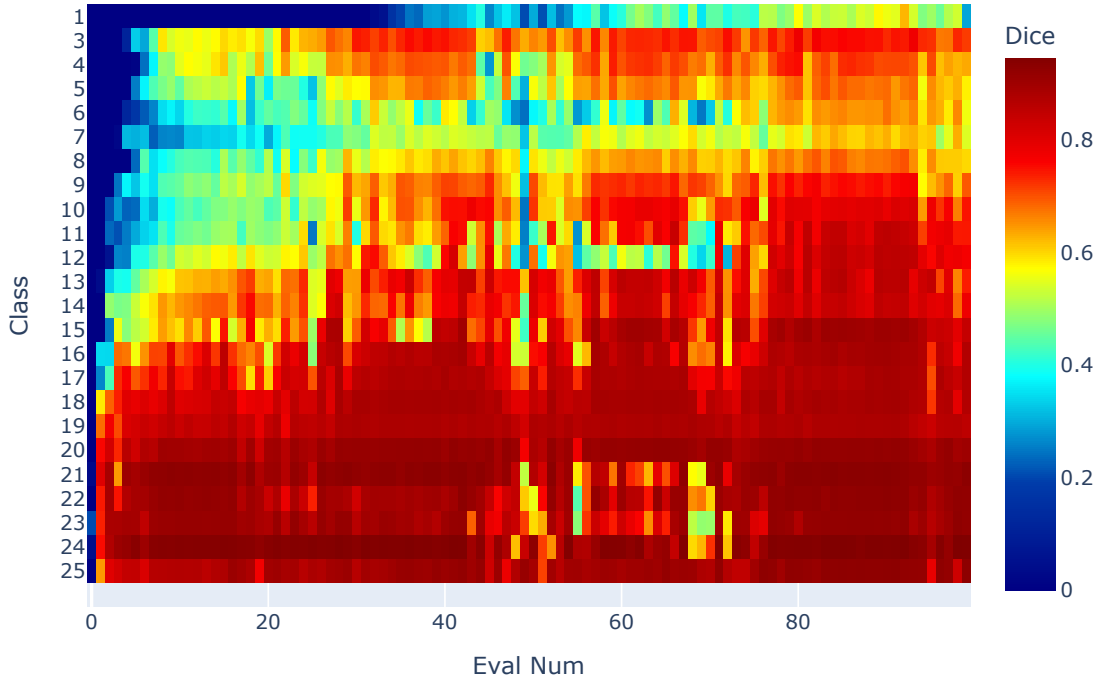


Figure 4.2: Per-class validation dice score averaged over all 4 samples as a heatmap. This shows the validation performance over time during training.

4.1.2 Qualitative Results

Fig. 4.3 presents three 3D visualizations of vertebral body segmentations generated by my model from the validation set. Individual vertebrae are assigned a unique color to distinguish separate classes. The model was able to identify all 24 vertebral bodies in each case, with the primary discrepancies appearing mostly in the shape of cervical vertebrae rather than at intervertebral boundaries. Remarkably, the model effectively preserved vertebrae gaps in regions where the discs were clearly visible, a significant improvement compared to previous methods that typically merged these gaps into

adjacent vertebral bodies. For improved visual appeal, a small Gaussian smoothing was applied. These renderings were produced using the Segmentations tool in Slicer [31].

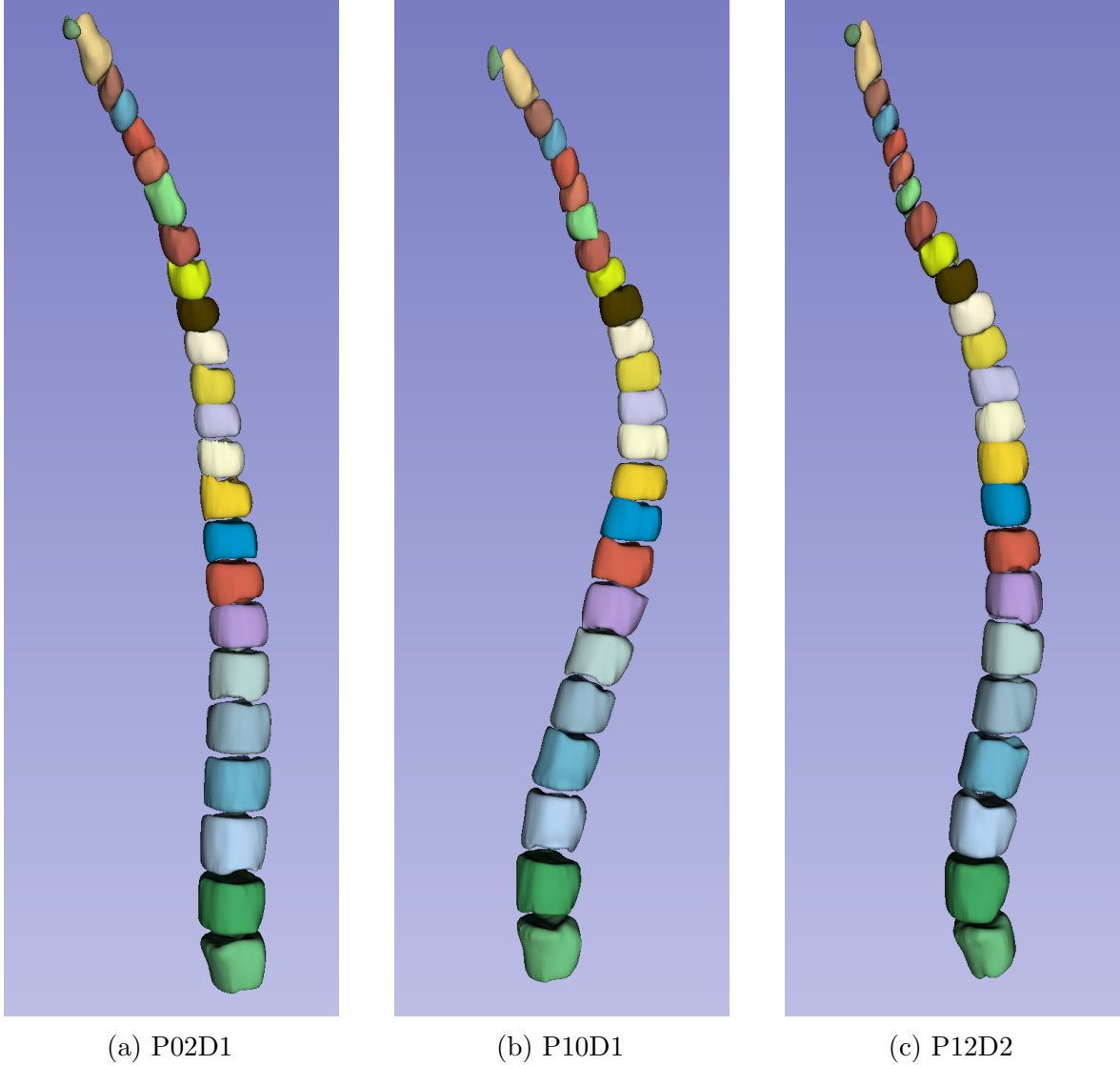


Figure 4.3: 3D view of sample predictions from the validation set.

Fig. 4.4 below is a visualization of the model’s confidence for each class. The model’s raw output logits are in the shape of (B, C, D, H, W) where B is the batch size, C is the number of classes, and (D, H, W) is the spatial volume size in depth, height, and width, respectively. For example, the HSCT dataset almost always has a size of

512 x 512 pixels in the x and y planes, with an arbitrary number of axial slices, typically around 180-200. My training used a batch size of 1 and had 25 classes plus background, so an example resulting shape would be (1, 26, 180, 512, 512). To convert the model's output volume, which has a logit $z_{c,d,h,w}$ for each voxel (d, h, w) , into probabilities, I use the softmax function, which is given in the Pytorch library [90] as:

$$p_{c,d,h,w} = \frac{\exp(z_{c,d,h,w})}{\sum_{c'=1}^C \exp(z_{c',d,h,w})}, c = 1, 2, \dots, C. \quad (4.1)$$

Finally, to get these probabilities into the map shown below, I summed the value of every voxel for each separate class volume into one 3D volume (from (C, D, H, W) to $(1, D, H, W)$) then scaled the values to the appropriate range of softmax: 0-1.

To convert these probabilities into integer values for each class, I opted to use argmax. This function simply chooses which class has the highest probability at each voxel (d, h, w) . Formally, it can be described as:

$$\hat{c}_{d,h,w} = \arg \max_{c \in \{1 \dots C\}} p_{c,d,h,w} \quad (4.2)$$

My reasoning for choosing this function is that previous methods, like [83, 85], found that argmax consistently improved segmentation accuracy via Dice score compared to thresholded values, of which the optimal value can vary between samples.

In Fig. 4.4, the color gradient extends from light pink (low probability) to dark blue (high probability), representing the model's per-voxel confidence. The model exhibits consistently high certainty at the center of each vertebra, with lower certainty along the boundaries. Confidence also decreases moving from the lumbar up into the cervical region, reflecting the lower Dice scores observed there. Additionally, the model has some nonzero, low confidence readings in the sacral region, where transitional vertebrae could be. This sample is of a patient with no transitional vertebrae or anomalies, so

you can see the very low confidence of the fact that L6 is lumbarized. A very small probability threshold has been applied in order to exclude the background and have boundary uncertainties more pronounced.

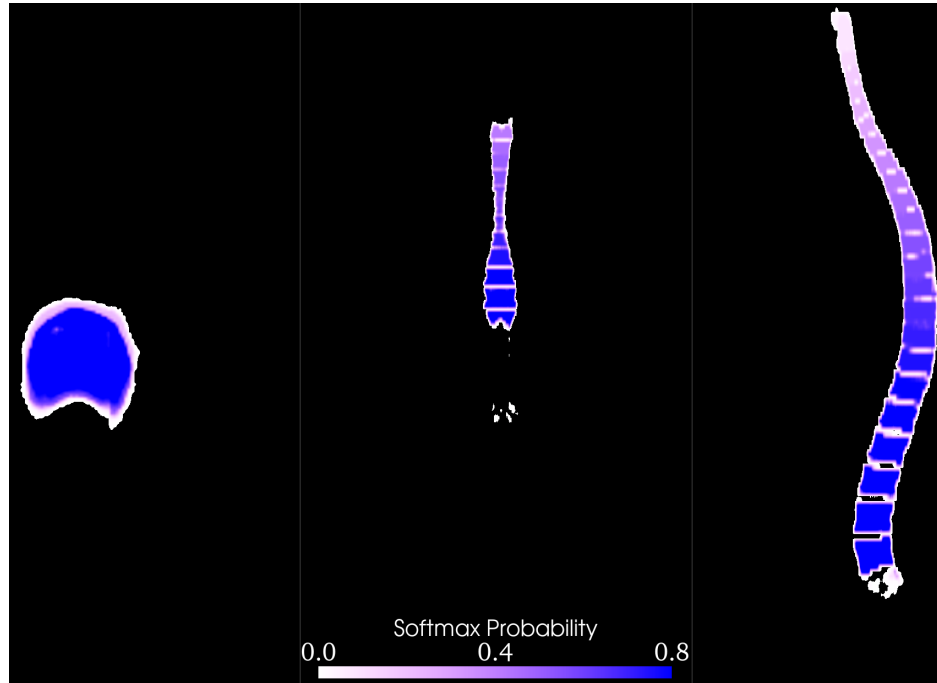


Figure 4.4: Concatenated per-class probability map of model output after softmax, depicting spatial confidence from 0 (no confidence) to 1 (pure certainty).

To evaluate segmentation accuracy, I created an error map by comparing the ground truth and predicted labels on a voxel-by-voxel basis. Any voxel where the prediction differs from the ground truth is marked as an error (red), while correctly classified voxels remain in gray. In other words, there is no distinction between false positives and negatives, whereas true positives are gray and true negatives are white background. I then filter out small, isolated misclassifications to emphasize the more significant errors. Next, the filtered error map is converted into a solid red 3D surface, and the correctly identified vertebrae are visualized as a semi-transparent surface in gray. In this way, regions of disagreement between the ground truth and predicted labels become visually distinct.

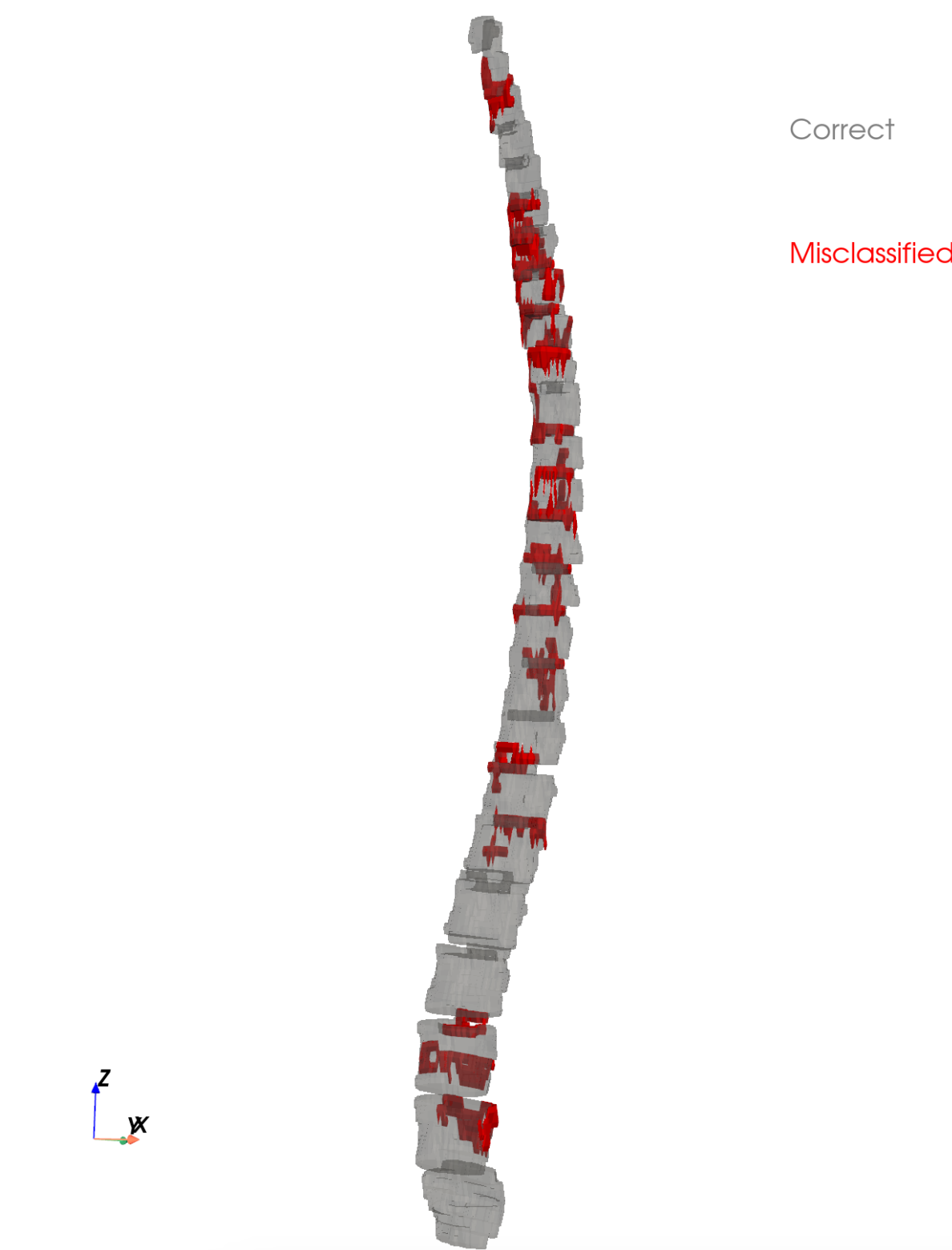


Figure 4.5: 3D volumetric difference map of my model's predictions on P08D1.

By examining the error map, we observe that most misclassifications occur in a surface-level “crust,” particularly at the intervertebral boundaries. This finding aligns with previous probability map results, which showed high confidence in central regions of each vertebra and greater uncertainty near the edges. Moreover, the errors underscore the inherent difficulties posed by the thick axial slice spacing, indicating that while the model performs well overall, it continues to face challenges in capturing fine details along the z-axis. While this prediction did quite well overall, additional figures in Appendix B will help to show the inconsistency in the cervical region.

It should be noted that while the validation accuracy in the graphs only goes up to 0.81, I report an average dice of 0.83 elsewhere. This inconsistency is due to postprocessing techniques. After doing a small component removal and connected component analysis, the segmentation is able to improve slightly. This is a common practice in image segmentation techniques.

4.1.3 Total Segmentator

Total Segmentator (TS) is a publicly available model based on nnU-Net that is able to capture a wide range of anatomical regions - from arteries to livers to, most importantly for this work, individual vertebrae. Since it is based on nnU-Net, it resamples all input images as close to isotropic as it can while still preserving details. For our specific dataset, this is an issue. For context, if you down-sample a high-resolution VerSe image from its native resolution (say, 1 mm isotropic) to our dataset’s spacing (1.17 mm in x and y, 5 mm in z), it actually tends to keep the detail between vertebral boundaries (see Fig. 3.5). Conversely, when a scan is natively acquired at a thicker slice thickness and lower dose, there is an inherent blurring effect that resampling cannot easily reverse. The boundaries between vertebrae are simply not present at sufficient resolution to be enhanced by standard up-sampling or super-resolution methods (see [83] - Carson

reports good results on test VerSe cases for his super resolution model but mixed results for HSCT images). Consequently, my hypothesis is that the built-in resampling logic in nnU-Net-based algorithms will still struggle with our low-dose and especially thick-spaced images. Furthermore, no direct comparison can be made to earlier studies using this dataset because they all relied on different ground-truth annotations. As a result, TS serves as a convenient, publicly available benchmark for assessing how a generalized segmentation system performs on data that deviate from the typical resolution distributions it was designed to handle. Although TS claims effectiveness even for low-dose CT images, the following demonstrates that its performance does not extend very well to our specific dataset.

It is important to note that TS does not provide a direct, end-to-end instance-level segmentation for each vertebral body in the same way as my model. Instead, it outputs a multi-class segmentation for entire vertebrae (aligning with the VerSe labeling scheme) and can optionally produce a separate, binary mask for the spinal column as a whole. To ensure compatibility with both my model and ground-truth annotations, I combined these two outputs—intersecting them to create a single unified volume.

Quantitatively, the results of TS on the same validation set that I used in the previous sections is given below, where best values in each column are in bold.

Table 4.2: Abbreviated results of TS on the same validation set.

Patient	Cervical	Thoracic	Lumbar	Overall
02 Day 1	0.623 $\pm$ 0.20	0.60 $\pm$ 0.32	0.917 $\pm$ 0.03	0.672
08 Day 1	0.62 $\pm$ 0.20	0.744 $\pm$ 0.27	0.63 $\pm$ 0.40	0.684
08 Day 3	0.608 $\pm$ 0.19	0.604 $\pm$ 0.38	0.478 $\pm$ 0.45	0.579
10 Day 1	0.6 $\pm$ 0.18	0.904 $\pm$ 0.04	0.939 $\pm$ 0.02	0.822
12 Day 2	0.61 $\pm$ 0.20	0.575 $\pm$ 0.36	0.934 $\pm$ 0.02	0.660

There is a clear pattern that emerges from this data: while TS does quite well on the thoracic and lumbar of one sample (Patient 10 Day 1), it struggles both in maximum Dice scores and having a much larger deviation. This shows not only a much lower overall accuracy per-region, but also a much higher inconsistency. In the cervical region, TS consistently falls below 0.63, while mine is in the region of 0.70-0.75. TS exhibits wild variability in the thoracic region, from 0.6 all the way to 0.9, while my work reports a steadier 0.86. Both models perform their best in the lumbar region, with TS having a distinct drop in only one patient (8, days 1 and 2). This showcases the fact that the resampling pipeline may not be robust in outlier datasets like ours.

In Fig. 4.6 (along with some in Appendix B), I modified the volumetric error map by adding an axial (z-direction) slice plane to reveal internal segmentation errors beyond surface-level misclassifications. The right side of each figure presents the 3D volume with the intersecting slice plane, while the left side displays the corresponding 2D slice, including a slice index for clarity and consistency.

There are three primary regions where TS consistently struggles: the cervical, mid-thoracic, and occasionally lower lumbar regions. Patient P08D3 was specifically chosen for visualization because it highlights TS’s uncertainty in each of these problematic areas. Firstly, TS generally performed well in the lumbar region except for Patient 8, whose transitional L6 vertebra led to significant errors. Specifically, TS incorrectly labeled the lowest vertebra (L6) as L5, causing a cascading misclassification that continued until it correctly identified vertebrae at L3. In the mid-thoracic region, the reason for the errors are less apparent, but are likely due to unclear boundaries between adjacent vertebrae. Finally, despite visually appearing to delineate cervical intervertebral boundaries clearly, TS consistently included vertebral processes rather than isolating only the vertebral bodies. These extraneous regions are irrelevant to the SUV extraction goal of this

project and introduce unnecessary variability into SUV measurement statistics. Lastly, in Fig. 4.7, I provide an example of my model’s prediction for the same case (Patient 8, Day 3), using the identical axial slice plane for comparison. This visualization highlights that although my model can sometimes produce noticeable individual errors, they are typically confined to surface-level boundaries (like a ‘crust’) and therefore have limited impact on overall segmentation metrics.

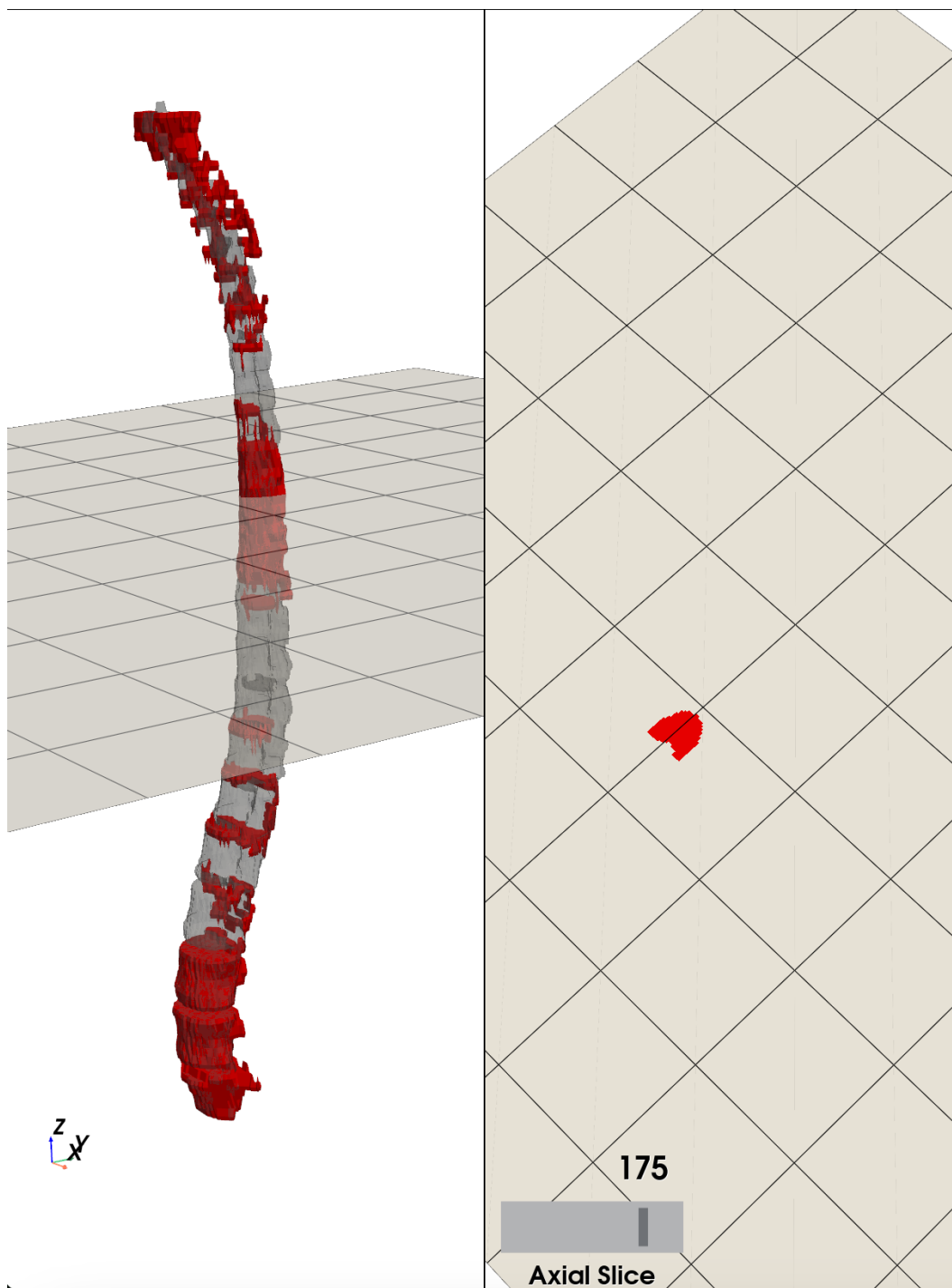


Figure 4.6: Total Segmentator’s prediction on P08D3. The 2D slice plane shows the density of the failed segmentation. Red depicts incorrect predictions, gray depicts correct.

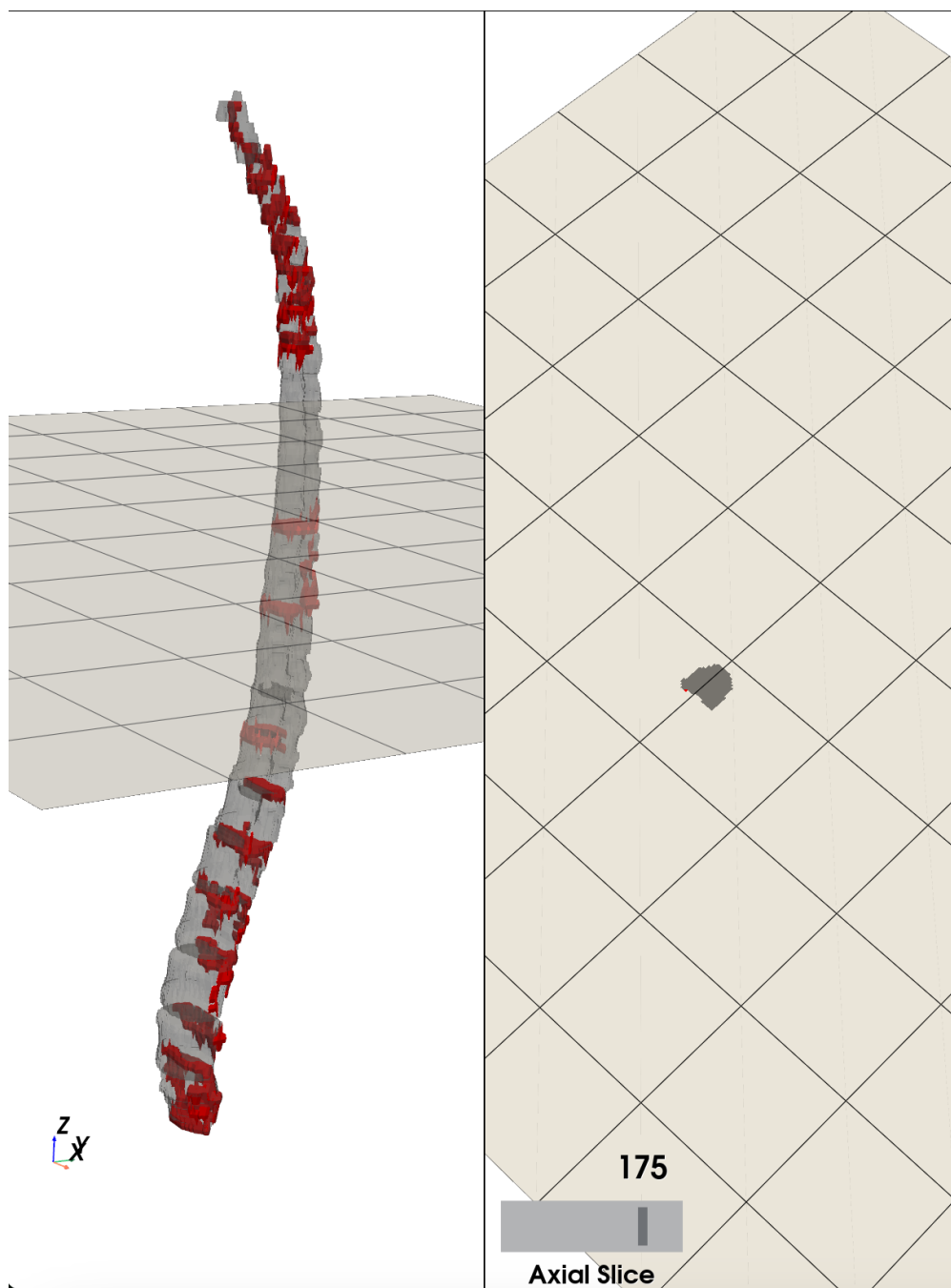


Figure 4.7: 3D volumetric difference map of my model's prediction on validation sample P08D3 showing the slice plane.

4.2 SUV Extraction

I ran my SUV measurement algorithm on every day of each validation set, as two additional test samples that were not included in training or validation, dubbed test cases. For example, my validation set had only Day 1 of Patient 2, but I ran segmentation and SUV measurements on all three days, meaning that days 2 and 3 are considered test cases.

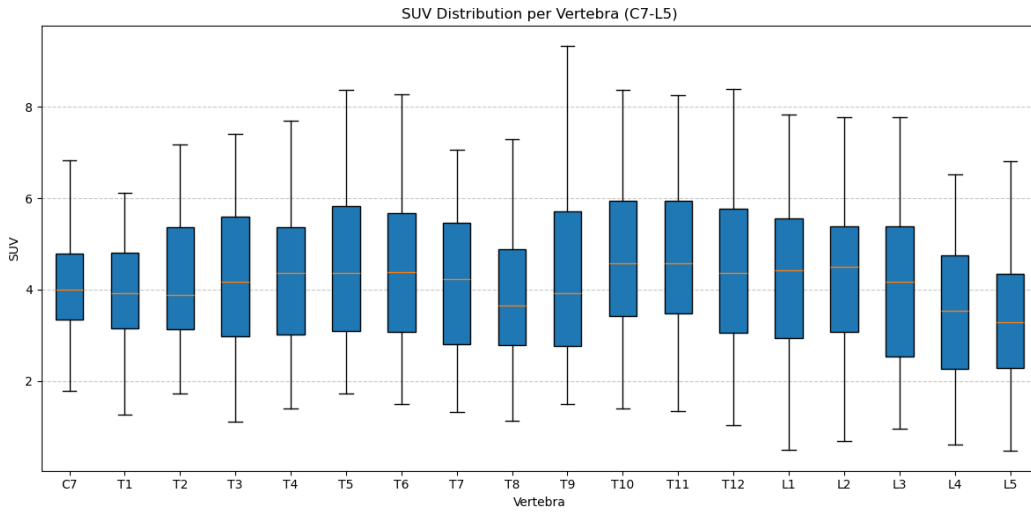


Figure 4.8: Box plots depicting the FLT-PET SUV distributions across vertebral body ROIs (C7–L5) on the 28th day post-transplant.

In Fig. 4.8, I computed the box plot for a sample patient on day +28 post-transplant. Additional box plots denoting the distribution across all three days of a sample are located in Appendix A. Each box spans from the first quartile (Q1) to the third quartile (Q3), with an orange segment marking the median SUV value. Whiskers extend to the minimum and maximum values within 1.5 times the interquartile range (IQR), while individual circles indicate outliers beyond this range.

To rigorously validate the performance of SUV measurements, I computed the error metrics and correlation against physician provided ground truth statistics. These

statistics include mean, median, minimum, maximum, and standard deviation (STDV) SUV distribution. For sample patient 2, I computed every error metric and the correlation coefficient on all the three scans combined and the results are shown in Table 4.3.

Table 4.3: Error metrics after FLT-PET SUV extraction of a sample patient.

Patient	Measurement	MAE	RMSE	Correlation
P02	SUV Mean	0.642	0.982	0.994
P02	SUV Median	0.703	1.096	0.989
P02	SUV Min	1.011	1.648	0.773
P02	SUV Max	0.372	0.505	0.99
P02	STDV	0.26	0.417	0.849

The results indicate excellent performance for SUV mean, median, and maximum with very high correlation values, demonstrating the model’s ability to accurately track SUV trends. SUV maximum also shows strong accuracy which can be seen in its absolute errors. These statistics also show agreement with the segmentation results and previous methods [83, 85], as, like them, I noticed the maximum SUV value of a given vertebral tends to be near the center of mass.

However, SUV minimum exhibits notably higher errors and substantially lower correlation. This aligns with the known difficulties in precisely capturing minimum SUV due to increased sensitivity to partial-volume effects that arises from interpolation as well as image noise at low metabolic activity levels.

Table 4.4 presents the average absolute magnitude error across all classes for each validation sample, comparing the model’s segmentation results to the manual CT annotations. These findings are highly encouraging; despite an average Dice accuracy of approximately 0.83, the SUV measurements demonstrate strong agreement with

Table 4.4: Average absolute difference between manual ground truth annotations and model predictions for SUV extraction on validation training set.

Patient	MAE
P02 Day 1	0.0244
P08 Day 1	0.0267
P08 Day 3	0.0906
P10 Day 1	0.0088
P12 Day 2	0.1717

manual annotations. However, it’s worth noting that the SUV errors may be slightly underestimated. Specifically, SUV measurements in the cervical region were limited exclusively to vertebra C7, whereas the segmentation model primarily struggles with the cervical vertebrae as a whole.

To further analyze the effectiveness of my model’s ability to extract relevant biological signals from the PET images, I have provided two plots in Fig. 4.9 below. The left plot is the measured median FLT uptake trends for the thoracic marrow cavities across the entire patient cohort in the initial pilot study by Williams et al. [26]. The plot on the right is generated from interpolating the CT segmentation from my model to the PET space, unionizing all predicted thoracic ROIs, and computing the box plots per scan day for a single patient. To eliminate any confusion, note that the y-axes are in different units. The left plot’s units are Ki , or the *net influx rate constant* when you fit the raw, dynamic FLT-PET data to a curve. Ont the other hand, the right plot is measured on static, precomputed SUV measurements. Since I only had access to the SUV measurements and not the raw PET data, I cannot rigorously compute or even approximate Ki for my proposed method. However, the figures are still worth comparing as they show the same clear FLT uptake trend that arises from subsequent

scan dates: minimal uptake pre-transplant, a marginal rise 5 days post-transplant, and a significant increase by the 28th day.

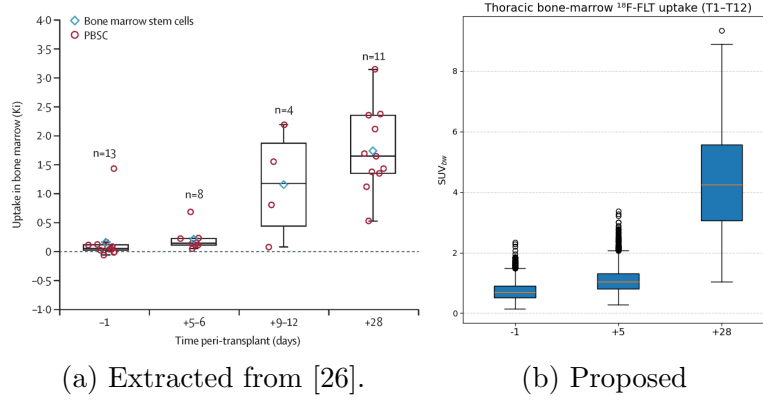


Figure 4.9: A qualitative comparison of FLT uptake trends of Williams et al. and my proposed model.

4.2.1 Limitations

A key limitation arises when labels derived from the CT space are directly propagated to the PET domain via basic voxel resampling and nearest-neighbor interpolation. As shown below in Fig. 4.10, even if the patient has very little movement, there is clear misalignment. I noticed that the amount of misalignment varies quite a bit across individuals. Although grayscale colormaps are typically not used for PET scans, I opted for it in this case to highlight the boundary mismatches. This challenge underscores a broader limitation: any method relying purely on CT-derived labels for PET analysis may inherit systematic registration errors that adversely affect quantitative measurements. More sophisticated rigid or elastic registration techniques could address some of these issues, but I attempted some initial experiments and they revealed excessive label erosion along with other distortions. This suggests more a dedicated pipeline might be needed for robust multi-modal image alignment.

The variability in SUV measurement errors observed in my results—particularly the increased error in SUV minimum despite consistently strong correlations—is consistent with findings in prior literature. For instance, [126] demonstrated substantial differences in clinical interpretation and response assessments depending on the SUV metric used, reporting considerable variability in individual tumor measurements when comparing SUV max, SUV mean, and SUV peak. This shows that there is a chance that the lower performance on SUV minimum may in part be due to metric sensitivity.

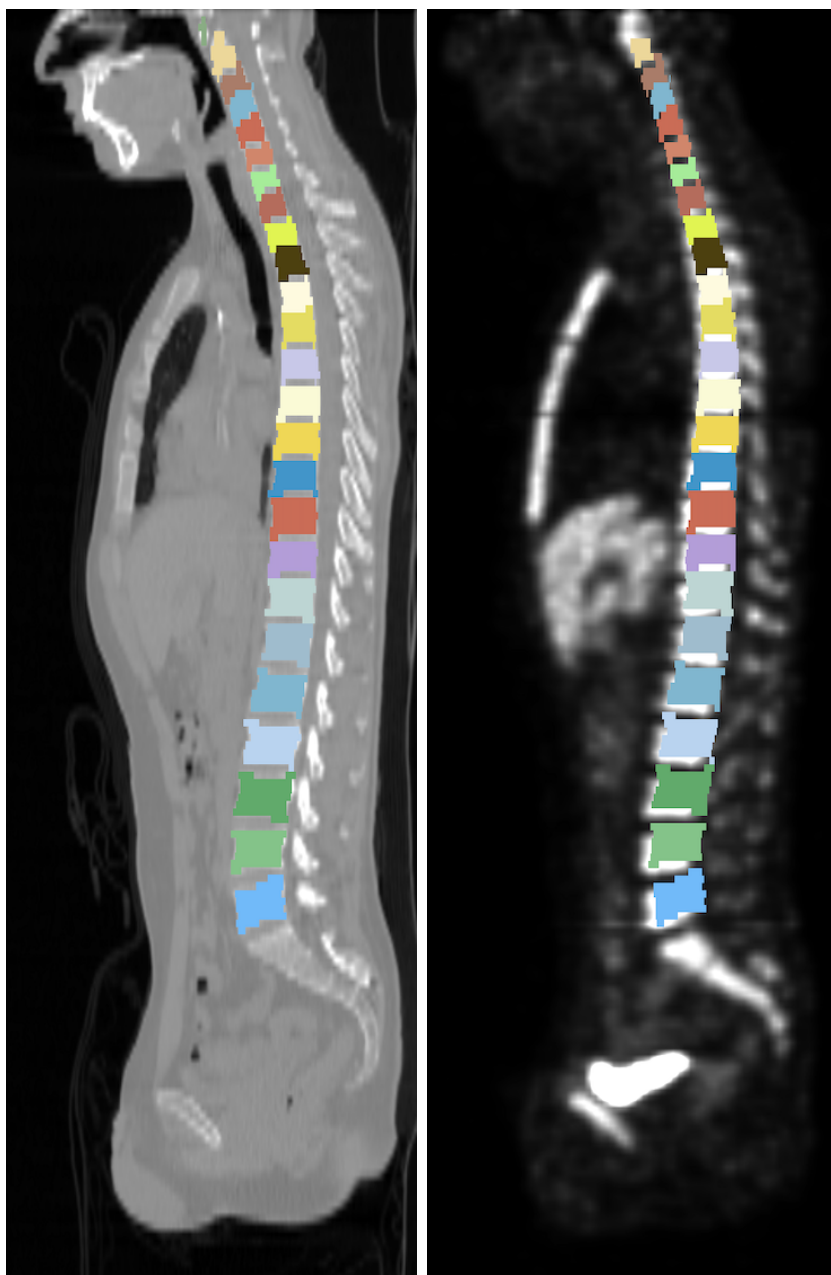


Figure 4.10: A sample co-registered CT (left) and PET (right) scan with the manually annotated label overlaid on both. Basic interpolation causes clear mismatches in the PET image.

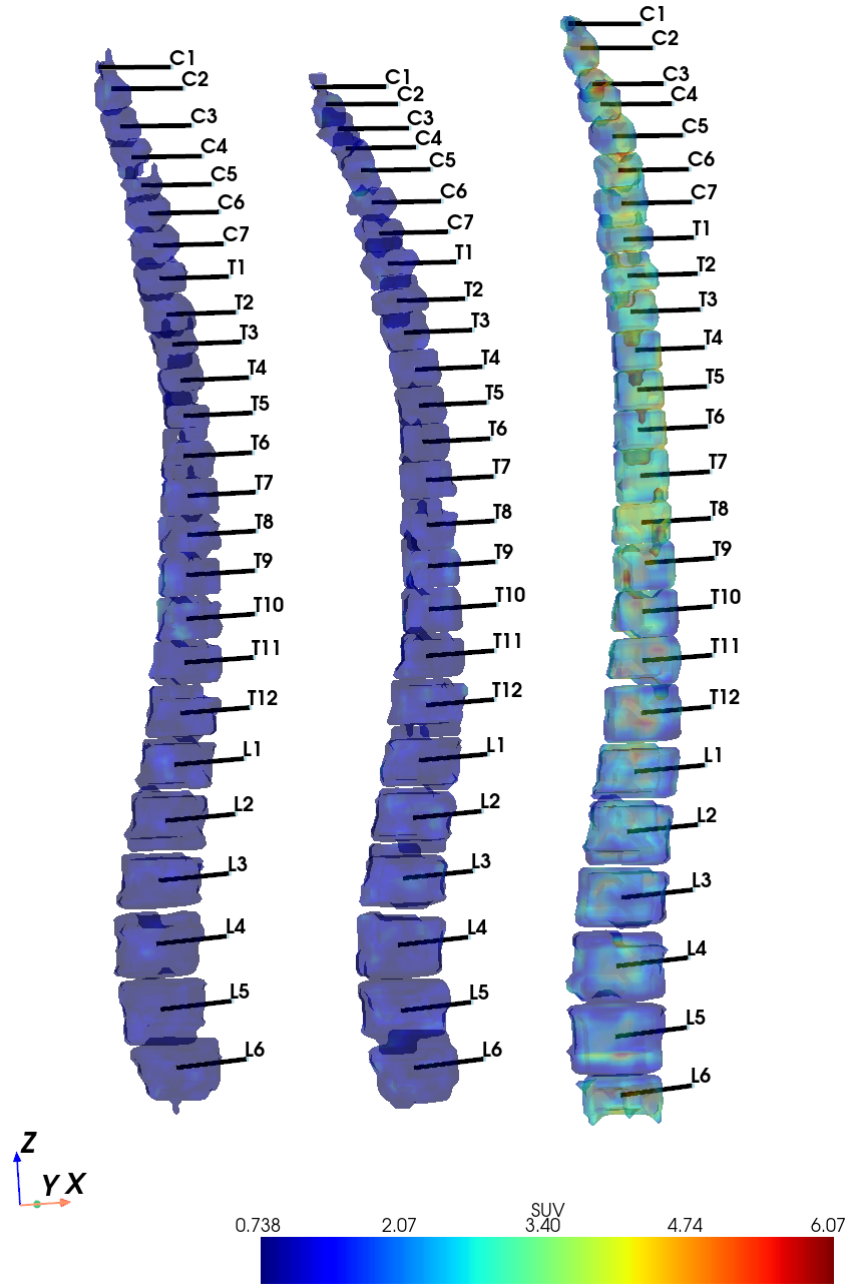


Figure 4.11: FLT-PET visualization of vertebral ROIs illustrating HSC metabolic progression from day -1 (left) to +28 (right) post-transplant. It ranges from 20th percentile (dark blue) to 99th percentile (dark red).

Chapter 5

Conclusion

In this thesis, a fully automated, end-to-end segmentation framework was successfully developed and evaluated for the task of extracting standardized uptake values (SUV) from vertebral bodies in low-dose, thick-spaced FLT-PET/CT scans of hematopoietic stem cell transplant (HSCT) patients. This work significantly advances the state-of-the-art by directly addressing challenges unique to this dataset, including low axial resolution and anatomical variability, through a specialized attention-gated 3D UNet architecture.

The effectiveness of supplementing limited clinical training data with augmented labels derived from the VerSe dataset was clearly demonstrated, significantly enhancing the segmentation accuracy and robustness of the model. Quantitative analysis confirmed high overall segmentation performance, with notably strong results in lumbar and thoracic regions, despite known challenges distinguishing boundaries in that region. Even though the cervical region had limited success, it still consistently ranked higher than other methods tested on this dataset.

A detailed comparative analysis highlighted the proposed model’s superior reliability and accuracy compared to the publicly available Total Segmentator model. Particularly, the presented method demonstrated improved robustness when handling transitional vertebrae and lower-quality imaging scenarios—common in certain clinical settings (like ours) but inadequately addressed by current generalized segmentation models. Additionally, I demonstrated that with the right refinements for low-dose CT data, a

simpler model trained on a smaller dataset (< 150 images) can match or outperform more complex approaches like TS, which was trained on $> 1,200$ images.

Critically, the automated SUV measurements obtained from model-generated segmentations showed excellent correlation with physician-derived SUV values. This rigorous clinical validation marks a significant advancement over previous work, ensuring the practical applicability of the proposed method for clinical monitoring of HSCT patients. The thorough error analysis presented provides transparency regarding the limitations inherent to voxel interpolation and spatial resolution, while clearly illustrating areas for continued methodological refinement.

In conclusion, the methods developed in this thesis provide a powerful and reliable automated solution for vertebral body segmentation and SUV extraction in challenging clinical imaging contexts. These advancements mark an important step towards enhanced clinical monitoring capabilities, offering potential for early relapse detection in HSCT patients.

5.1 Original Contributions

The novel contributions I have made to the HSCT project are as follows:

1. **VerSe Label Augmentation:** I created a tool using connected component analysis to strip the posterior processes from the vertebral bodies, followed by centroid computation to find the ‘expected’ location of the body. This tool made it possible to use an existing public dataset and modify it to fit the specific needs of our project. Additionally, it allowed my model to generalize and converge when it couldn’t without them.

2. **Manual Annotations:** I created 22 manual annotations of our CT images, each capturing every vertebral body of the entire spine. This was a very labor-intensive task, so the supplementation of VerSe helped here too.
3. **First End-to-End Individual Vertebral Body Segmentation:** This work is the first in the project to directly segment individual vertebral bodies from low-dose and thick-axial-spacing CT images.
4. **Custom Attention-gated UNet:** I created a specialized 3D UNet architecture tailored to our unique dataset. Unlike a standard attention-gated UNet, this version combines residual units with the attention gates situated along the skip connections, employs group normalization in place of batch normalization, and incorporates a dense skip connection for preserving fine-grained detail. These modifications improve stability on limited data, enhance the focus on key anatomical structures, and help to provide higher segmentation quality.
5. **Robustness to Anatomical Variability:** This work is the first in the project to be able to handle significant anatomical anomalies, such as lumbarized vertebrae. Additionally, this model can easily scale to handle fused vertebrae and other common anomalies such as mild scoliosis/kyphosis, bone spurs, and degenerative disc disease, as long as enough training data is supplied. VerSe does indeed have data for fused/sacrilized L5 (a ‘missing’ vertebra).
6. **Isosurface and difference map tools:** I created a tool that can visualize 3D ROIs of the SUV data contained within the marrow cavities of FLT-PET images from any day’s scan. Carson [83] created something similar, but his only works on the 28th day post transplant. This could be used by researchers to more easily analyze SUV distribution.

7. **Automated FLT-PET SUV Extraction and Analysis:** this work is the first to directly compare the SUV measurement statistics with physician data of our specific dataset and provide a detailed error analysis as well as the sources of SUV variability.

5.2 Recommendations for Future Work

To further enhance the accuracy and robustness of vertebral body segmentation and SUV extraction, several promising directions can be explored in future research.

1. Annotations, Training, and Models

- Improving the quality and consistency of the ground truth labels could significantly enhance model performance. Personal annotation inconsistencies, which became apparent throughout the project, introduce noise and can reduce the ability of the model to learn precise anatomical boundaries. Implementing rigorous annotation guidelines and/or employing professional annotators to refine the existing dataset could provide substantial performance gains. Additionally, refining the vertebral body separation algorithm introduced in section 3.2 would further improve the quality of the augmented labels, leading to better model generalization.
- Broader dataset scope. While VerSe is considered the ‘gold standard’ in vertebral segmentation, it only contains <400 volumes. An essential next step to refining the segmentation of our HSCT spines is to include additional images from other datasets. The most representative datasets should be similar to VerSe: publicly available, adult, whole 3D spine of individually labeled vertebrae, and having a mix of healthy and pathological cases. One such example is CTSpine1K [127].

- Hyperparameter tuning and model optimization represent another immediate avenue for performance improvement. Systematic hyperparameter optimization, such as Bayesian optimization or grid searches, could lead to noticeable accuracy improvements without altering the fundamental model architecture. Furthermore, exploring minor architectural modifications, such as improved attention mechanisms, additional residual connections, or different normalization techniques, could yield incremental performance enhancements.
- If computational resources allow, it may be beneficial to investigate more complex model architectures such as transformer-based segmentation models or hybrid CNN-transformer approaches. These advanced models could potentially better capture the anatomical variability and subtle boundaries inherent in vertebrae.
- Handling anisotropy in a more sophisticated manner could provide additional improvements. Methods like anisotropic convolutional kernels, multi-view fusion (already explored by [83] but only for binary spinal column), or adaptive resampling techniques may help the model better handle the uncommonly thick axial spacing present in our dataset.

2. SUV Measurement and Evaluation

- This work reports SUV statistics for individual lumbar, thoracic, and C7 vertebral bodies. However, our physician SUV measurements extend to many other bones, such as sternum and bilateral pelvis (these first two are already explored in [83, 85] but they did not have access to physician measurements), bilateral iliac crest, humerus, and femur. Expanding segmentation and subsequent SUV extraction to these bones would greatly enhance the clinical feasibility and credibility of an automated method. Additionally, expanding

physician measurements to the rest of the cervical region could assess my (and any future) work’s accuracy in that region.

- Provided that sufficient physician-curated data is available, a dedicated algorithm should be developed to detect spatial SUV distribution anomalies - early warning signs that may be indicative of graft failure or relapse. This was lightly touched on by Hurtado [85], but improving its scale and robustness would be a significant advancement in this project.

3. Multi-Modality and Co-Registration

- It could be worth exploring multi-modal segmentation models, which directly integrate CT and PET modalities within the segmentation pipeline. Since the ultimate goal of this project is to isolate individual vertebral ROIs in the PET space, it could be worth exploring algorithms that have the potential to learn more ‘anatomy aware’ features than a pipeline for either modality alone could achieve.
- Finally, one of the most significant opportunities lies in improving the alignment between the CT and PET images. Building a dedicated rigid or deformable registration pipeline between the two modalities could lead to more clinically accurate and reliable labels in the PET space.

Exploring these recommendations has the potential to significantly advance the automated segmentation and SUV extraction of bones in low-dose FLT-PET/CT images, furthering the ultimate goal of developing a tool to assist physicians in monitoring the recovery of HSCT patients.

Bibliography

- [1] Tonya Hines. *Anatomy of the Spine*. URL: <https://mayfieldclinic.com/pe-anat spine.htm>. (accessed: 02.12.2025).
- [2] *Spinal Anatomy*. URL: <https://www.acsneuro.com/resources/anatomy-spinal>. (accessed: 02.12.2025).
- [3] G P Konin and D M Walz. “Lumbosacral transitional vertebrae: classification, imaging findings, and clinical relevance”. In: *AJNR Am J Neuroradiol* (2010). DOI: 10.3174/ajnr.A2036.
- [4] *Sports Medicine: Lumbosacral Transitional Vertebra (LSTV)*. URL: <https://www.nationwidechildrens.org/conditions/sports-medicine-lumbosacral-transitional-vertebra>. (accessed: 02.12.2025).
- [5] Ram Doo et al. “The prevalence and clinical significance of transitional vertebrae: a radiologic investigation using whole spine spiral three-dimensional computed tomographic images”. In: *Anesth Pain Med (Seoul)* (2020). DOI: <https://doi.org/10.17085/apm.2020.15.1.103>.
- [6] Luis Becker et al. “Influences of lumbo-sacral transitional vertebrae for anterior lumbar interbody fusion”. In: *Scientific Reports* (2024). DOI: <http://dx.doi.org/10.1038/s41598-024-53179-w>.
- [7] Mauricio Nicolas Ferrao Blanco, Mirjam Belderbos, and Hermann Josef Vormoor. “Leukemia suppressing normal bone marrow: how long does it last?” In: *Haematologica* (2023). DOI: <https://doi.org/10.3324/haematol.2023.282955>.
- [8] E. Donnall Thomas M.D. et al. “Intravenous Infusion of Bone Marrow in Patients Receiving Radiation and Chemotherapy”. In: *N Engl J Med* (1957). DOI: 10.1056/NEJM195709122571102.
- [9] Rajamanickam Baskar et al. “Cancer and Radiation Therapy: Current Advances and Future Directions”. In: *Int J Med Sci* (2012). DOI: 10.7150/ijms.3635.
- [10] Robert R. Jenq and Marcel R. M. van den Brink. “Allogeneic haematopoietic stem cell transplantation: individualized stem cell and immune therapy of cancer”. In: *Nat Rev Cancer* (2010). DOI: <https://doi.org/10.1038/nrc2804>.
- [11] Alexander Birbrair and Paul S Frenette. “Niche heterogeneity in the bone marrow”. In: *Ann N Y Acad Sci* (2016). DOI: <https://doi.org/10.1111/nyas.13016>.
- [12] Bokyoung Park, Keon Hee Yoo, and Changsung Kim. “Hematopoietic stem cell expansion and generation: the ways to make a breakthrough”. In: *Blood Res* (2015). DOI: 10.5045/br.2015.50.4.194.
- [13] Mahadevappa Mahesh. *The Essential Physics of Medical Imaging, Fourth Edition*. American Association of Physicists in Medicine, 2021.
- [14] Tami DenOtter and Johanna Schubert. *Hounsfield Unit*. Stat Pearls, 2023. URL: <https://www.ncbi.nlm.nih.gov/books/NBK547721/>.

- [15] Mayo Clinic Staff. *Positron emission tomography scan*. URL: <https://www.mayoclinic.org/tests-procedures/pet-scan/about/pac-20385078>. (accessed: 02.16.2025).
- [16] Rafael Moreno-Sánchez et al. “Energy metabolism in tumor cells”. In: *Molecular Imaging and Contrast Agent Database (MICAD) [Internet]* (2007). URL: <https://doi.org/10.1111/j.1742-4658.2007.05686.x>.
- [17] Naima Hammoudi et al. “Metabolic alterations in cancer cells and therapeutic implications”. In: *Molecular Imaging and Contrast Agent Database (MICAD) [Internet]* (2011). URL: [10.5732/cjc.011.10267.%20PMID:%2021801600](https://doi.org/10.5732/cjc.011.10267.%20PMID:%2021801600).
- [18] Katherine Lameka, Michael D. Farwell, and Masanori Ichise. “Chapter 11 - Positron Emission Tomography”. In: *Neuroimaging Part I*. Elsevier, 2016. URL: <https://doi.org/10.1016/B978-0-444-53485-9.00011-8>.
- [19] Harvey A. Ziessman, Janis P. O’Malley, and James H. Thrall. “Chapter 11 - Oncology: Positron Emission Tomography”. In: *Nuclear Medicine: The Requisites*. Elsevier, 2014. URL: <https://doi.org/10.1016%2FB978-0-323-08299-0.00011-0>.
- [20] Kam Leung. “[18F]Fluoro-2-deoxy-2-d-glucose”. In: *Molecular Imaging and Contrast Agent Database (MICAD) [Internet]* (2004). URL: <https://www.ncbi.nlm.nih.gov/books/NBK23335/>.
- [21] Kam Leung. “3’-Deoxy-3’-[18F]fluorothymidine”. In: *Molecular Imaging and Contrast Agent Database (MICAD) [Internet]* (2004). URL: <https://www.ncbi.nlm.nih.gov/books/NBK23373/>.
- [22] Ali Agool et al. “F-18 FLT PET: a noninvasive diagnostic tool for visualization of the bone marrow compartment in patients with aplastic anemia: a pilot study”. In: *Clin Nucl Med* (2011). DOI: <https://doi.org/10.1097/rlu.0b013e31820aa1a1>.
- [23] A F Shields et al. “Imaging proliferation in vivo with [F-18]FLT and positron emission tomography”. In: *Nat Med* (1998). DOI: <https://doi.org/10.1038/3337>.
- [24] SN Wolff. “Second hematopoietic stem cell transplantation for the treatment of graft failure, graft rejection or relapse after allogeneic transplantation”. In: *Bone Marrow Transplantation* (2002). DOI: <https://doi.org/10.1038/sj.bmt.1703389>.
- [25] TR Spitzer. “Engraftment syndrome following hematopoietic stem cell transplantation”. In: *Bone Marrow Transplantation* (2001). DOI: [10.1038/sj.bmt.1703015](https://doi.org/10.1038/sj.bmt.1703015).
- [26] Kirsten M Williams et al. “Imaging of subclinical haemopoiesis after stem-cell transplantation in patients with haematological malignancies: a prospective pilot study”. In: *Lancet Haematol* (2018). DOI: [https://doi.org/10.1016/s2352-3026\(17\)30215-6](https://doi.org/10.1016/s2352-3026(17)30215-6).
- [27] Kirsten M Williams and Jennifer Holter-Chakrabarty. “Imaging haemopoietic stem cells and microenvironment dynamics through transplantation”. In: *Lancet Haematol* (2020). DOI: [https://doi.org/10.1016/s2352-3026\(20\)30003-x](https://doi.org/10.1016/s2352-3026(20)30003-x).

- [28] Jennifer Holter-Chakrabarty et al. “18F-FLT PET and Blood-based Biomarkers for Identifying Gastrointestinal Graft versus Host Disease after Allogeneic Cell Transplantation”. In: *Radiology: Imaging Cancer* (2025). DOI: <https://doi.org/10.1148/rycan.240096>.
- [29] Jennifer Holter-Chakrabarty. Personal communication. 2024.
- [30] Kevin Adistambha et al. “Enhancing Multimedia Search Using Human Motion”. In: *Multimedia - A Multidisciplinary Approach to Complex Issues*. InTech, 2012. URL: <http://dx.doi.org/10.5772/36585>.
- [31] *3D Slicer Documentation*. URL: <http://www.slicer.org/>. (accessed: 02.16.2025).
- [32] Lee W. Goldman. “Principles of CT: Radiation Dose and Image Quality”. In: *Nuclear Medicine Technology* (2007). URL: <https://doi.org/10.2967/jnmt.106.037846>.
- [33] W Dean Bidgood Jr et al. “Understanding and Using DICOM, the Data Interchange Standard for Biomedical Imaging”. In: *J Am Med Inform Assoc* (2007). URL: <https://doi.org/10.1136/jamia.1997.0040199>.
- [34] Shivam Sharma. *Introduction to DICOM for Computer Vision Engineers*. URL: <https://blog.redbrickai.com/blog-posts/introduction-to-dicom-for-cv-engineers>. (accessed: 02.17.2025).
- [35] NEMA. *Digital Imaging and Communications in Medicine (DICOM)*. NEMA Publications, 2023. URL: https://dicom.nema.org/dicom/2004/04_01pu.pdf.
- [36] *NIfTI (Neuroimaging Informatics Technology Initiative) Reader/Writer*. URL: <https://docs.safe.com/fme/html/FME-Form-Documentation/FME-ReadersWriters/nifti/nifti.htm>. (accessed: 02.16.2025).
- [37] *The NIFTI file format*. URL: <https://brainder.org/2012/09/23/the-nifti-file-format/>. (accessed: 02.17.2025).
- [38] Zenodo. *NiBabel Release Records*. URL: <https://doi.org/10.5281/zenodo.591597>. (accessed: 02.17.2025).
- [39] Chris McClanahan. “History and Evolution of GPU Architecture”. In: *A Survey Paper* (2010). DOI: <https://picture.iczhiku.com/resource/paper/WyiWoZ0ZoDZWgvxN.pdf>.
- [40] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. URL: <http://www.deeplearningbook.org>.
- [41] Alexander L. Fradkov. “Early History of Machine Learning”. In: *International Federation of Automatic Control* (2020). DOI: <https://doi.org/10.1016/j.ifacol.2020.12.1888>.
- [42] Warren McCulloch and Walter Pitts. “A Logical Calculus of the Ideas Immanent in Nervous Activity”. In: *Bulletin of Mathematical Biology* (1943). DOI: <https://doi.org/10.1007/BF02478259>.
- [43] Frank Rosenblatt. “The perceptron: A probabilistic model for information storage and organization in the brain”. In: *Psychological Review* (1958). DOI: <https://psycnet.apa.org/doi/10.1037/h0042519>.
- [44] Kuldeep Shiruru. “Neural Network Approach for Processing Substation Alarms”. In: *International Journals of Power Electronics Controllers and Converters*

- (2015). DOI: https://www.researchgate.net/publication/319903709_Neural_Network_Approach_for_Processing_Substation_Alarms.
- [45] Y. LeCun et al. “Backpropagation Applied to Handwritten Zip Code Recognition”. In: *Neural Computation* (1989). DOI: 10.1162/neco.1989.1.4.541.
 - [46] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning”. In: *Nature* (2015). DOI: <https://doi.org/10.1038/nature14539>.
 - [47] Ragav Venkatesan and Baoxin Li. *Convolutional Neural Networks in Visual Computing*. CRC Press, 2015. URL: <https://www.taylorfrancis.com/books/mono/10.4324/9781315154282/convolutional-neural-networks-visual-computing-ragav-venkatesan-baoxin-li>.
 - [48] Yann Lecun and Y. Bengio. “Convolutional Networks for Images, Speech, and Time-Series”. In: *The Handbook of Brain Theory and Neural Networks* (1995). DOI: https://www.researchgate.net/publication/216792820_Convolutional_Networks_for_Images_Speech_and_Time-Series.
 - [49] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. *Semi-Supervised Learning*. MIT Press, 2010. URL: <https://mitpress.mit.edu/9780262514125/semi-supervised-learning/>.
 - [50] Misha Denil et al. “Predicting Parameters in Deep Learning”. In: *Advances in Neural Information Processing Systems* (2013).
 - [51] Moritz Laurer et al. “Less Annotating, More Classifying: Addressing the Data Scarcity Issue of Supervised Machine Learning with Deep Transfer Learning and BERT-NLI”. In: *Political Analysis* (2024). DOI: 10.1017/pan.2023.20.
 - [52] Connor Shorten and Taghi M. Khoshgoftaar. “A survey on Image Data Augmentation for Deep Learning”. In: *International Federation of Automatic Control* (2019). DOI: <https://doi.org/10.1186/s40537-019-0197-0>.
 - [53] Jason Hu et al. “Learning Image Priors through Patch-based Diffusion Models for Solving Inverse Problems”. In: *Neural Information Processing Systems* (2024). DOI: <https://doi.org/10.48550/arXiv.2406.02462>.
 - [54] Nitish Srivastava et al. “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”. In: *Journal of Machine Learning Research* (2014). DOI: 10.5555/2627435.2670313.
 - [55] Sergey Ioffe and Christian Szegedy. “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”. In: *International Conference on Machine Learning* (2015). DOI: <https://doi.org/10.48550/arXiv.1502.03167>.
 - [56] Yuxin Wu and Kaiming He. “Group Normalization”. In: *European Conference on Computer Vision* (2018). DOI: <https://doi.org/10.48550/arXiv.1803.08494>.
 - [57] Ayman S. El-Baz et al. *Multi Modality State-of-the-Art Medical Image Segmentation and Registration Methodologies*. Springer New York, NY, 2011. URL: <https://doi-org.ezproxy.lib.ou.edu/10.1007/978-1-4419-8204-9>.

- [58] Nida M. Zaitoun and Musbah J. Aqel. “Survey on Image Segmentation Techniques”. In: *Procedia Computer Science* (2015). DOI: <https://doi.org/10.1016/j.procs.2015.09.027>.
- [59] Al Bovik. *The Essential Guide to Image Processing*. Academic Press, 2009.
- [60] Salem Saleh Al-amri1, N.V. Kalyankar, and Khamitkar S.D. “Image Segmentation By Using Edge Detection”. In: *International Journal on Computer Science and Engineering* (2010). DOI: <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=1f2ac2b9ef743be79a40bcef23bfa3a5bf750b36>.
- [61] A. Fenster and B. Chiu. “Evaluation of Segmentation algorithms for Medical Imaging”. In: *IEEE Engineering in Medicine and Biology* (2006). DOI: 10.1109/IEMBS.2005.1616166.
- [62] Dominik Müller, Iñaki Soto-Rey, and Frank Kramer. “Towards a guideline for evaluation metrics in medical image segmentation”. In: *BMC Res Notes* (2022). DOI: <https://doi.org/10.1186/s13104-022-06096-y>.
- [63] Rongjian Zhao et al. “Rethinking Dice Loss for Medical Image Segmentation”. In: *IEEE International Conference on Data Mining* (2020). DOI: 10.1109/ICDM50108.2020.00094.
- [64] D.P. Huttenlocher, G.A. Klanderman, and W.J. Rucklidge. “Comparing images using the Hausdorff distance”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (1993). DOI: 10.1109/34.232073.
- [65] *Hausdorff distance between convex polygons*. URL: <https://cgm.cs.mcgill.ca/~godfried/teaching/cg-projects/98/normand/main.html>. (accessed: 02.24.2025).
- [66] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Advances in Neural Information Processing Systems* (2012). DOI: https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.
- [67] Jia Deng et al. “ImageNet: A large-scale hierarchical image database”. In: *IEEE Conference on Computer Vision and Pattern Recognition* (2009). DOI: 10.1109/CVPR.2009.5206848.
- [68] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully convolutional networks for semantic segmentation”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2015). DOI: 10.1109/CVPR.2015.7298965.
- [69] Kaiming He et al. “Deep Residual Learning for Image Recognition”. In: *Computer Vision and Pattern Recognition* (2015). DOI: <https://doi.org/10.48550/arXiv.1512.03385>.
- [70] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *International Conference on Medical Image Computing and Computer-Assisted Intervention* (2015). DOI: <https://doi.org/10.48550/arXiv.1505.04597>.
- [71] Özgün Çiçek et al. “3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation”. In: *CoRR* (2016). DOI: <https://doi.org/10.48550/arXiv.1606.06650>.

- [72] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. “V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation”. In: *Fourth International Conference on 3D Vision* (2016). DOI: <http://dx.doi.org/10.1109/3DV.2016.79>.
- [73] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. “Sequence to Sequence Learning with Neural Networks”. In: *Conference on Neural Information Processing Systems* (2014). DOI: <https://doi.org/10.48550/arXiv.1409.3215>.
- [74] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. “Neural Machine Translation by Jointly Learning to Align and Translate”. In: *International Conference on Learning Representations* (2015). DOI: <https://doi.org/10.48550/arXiv.1409.0473>.
- [75] Dzmitry Bahdanau et al. “End-to-end attention-based large vocabulary speech recognition”. In: *International Conference on Acoustics, Speech, and Signal Processing* (2016). DOI: <https://doi.org/10.1109/ICASSP.2016.7472618>.
- [76] Ashish Vaswani et al. “Attention is All you Need”. In: *Advances in Neural Information Processing Systems* (2017). DOI: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.
- [77] Cem Subakan et al. “Attention Is All You Need In Speech Separation”. In: *International Conference on Acoustics, Speech and Signal Processing* (2021). DOI: [10.1109/ICASSP39728.2021.9413901](https://doi.org/10.1109/ICASSP39728.2021.9413901).
- [78] Ozan Oktay et al. “Attention U-Net: Learning Where to Look for the Pancreas”. In: *CoRR* (2018). DOI: <https://doi.org/10.48550/arXiv.1804.03999>.
- [79] David Shin et al. “Morphometric analysis of cervical disc space height and interpedicular distance using computed tomography”. In: *Surg Neurol Int* (2024). DOI: https://doi.org/10.25259/SNI_279_2024.
- [80] Justin G. R. Fletcher et al. “CT morphometry of adult thoracic intervertebral discs”. In: *Eur Spine* (2015). DOI: <https://doi.org/10.1007/s00586-015-3925-y>.
- [81] Chuong Nguyen et al. “An automatic 3D CT/PET segmentation framework for bone marrow proliferation assessment”. In: *IEEE International Conference on Image Processing (ICIP)* (2016). DOI: [10.1109/ICIP.2016.7533136](https://doi.org/10.1109/ICIP.2016.7533136).
- [82] Chuong T. Nguyen et al. “Towards automatic 3D bone marrow segmentation”. In: *IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI)* (2016). DOI: [10.1109/SSIAI.2016.7459162](https://doi.org/10.1109/SSIAI.2016.7459162).
- [83] Brandon Carson. “Automatic Bone Structure Segmentation of Under-Sampled CT/FLT-PET Volumes for HSCT Patients”. MA thesis. University of Oklahoma, 2021.
- [84] Brandon D. Carson et al. “Approximate Vertebral Body Instance Segmentation by PET-CT Fusion for Assessment After Hematopoietic Stem Cell Transplantation”. In: *IEEE International Conference on Bioinformatics and Bioengineering (BIBE)* (2023). DOI: [10.1109/BIBE60311.2023.00018](https://doi.org/10.1109/BIBE60311.2023.00018).

- [85] Christian Favio Hurtado Eguez. “Multiclass Bone Segmentation of PET/CT Scans for Automatic SUV Extraction”. MA thesis. University of Oklahoma, 2021.
- [86] Anjany Sekuboyina et al. “VerSe: A Vertebrae labelling and segmentation benchmark for multi-detector CT images”. In: *Medical Image Analysis* (2021). DOI: <https://doi.org/10.1016/j.media.2021.102166>.
- [87] Maximilian T. Löffler et al. “A Vertebral Segmentation Dataset with Fracture Grading”. In: *Radiology: Artificial Intelligence* (2020). DOI: <https://doi.org/10.1148/ryai.2020190138>.
- [88] Hans Liebl et al. “A computed tomography vertebral segmentation dataset with anatomical variations and multi-vendor scanner data”. In: *Sci Data* (2021). DOI: <https://doi.org/10.1038/s41597-021-01060-0>.
- [89] John Nickolls et al. “Scalable parallel programming with CUDA”. In: *SIGGRAPH* (2008). DOI: 10.1145/1401132.1401152.
- [90] Adam Paszke et al. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Conference on Neural Information Processing Systems* (2019). DOI: <https://doi.org/10.48550/arXiv.1912.01703>.
- [91] *MONAI Homepage*. URL: <https://monai.io/index.html>. (accessed: 03.02.2025).
- [92] *CuPy Homepage*. URL: <https://cupy.dev>. (accessed: 03.02.2025).
- [93] *NumPy homepage*. URL: <https://numpy.org>. (accessed: 03.02.2025).
- [94] *SciPy Homepage*. URL: <https://scipy.org>. (accessed: 03.02.2025).
- [95] Martín Abadi et al. “TensorFlow: A system for large-scale machine learning”. In: *Distributed, Parallel, and Cluster Computing* (2016). DOI: <https://doi.org/10.48550/arXiv.1605.08695>.
- [96] *Deep Learning in MATLAB*. URL: <https://www.mathworks.com/help/deeplearning/ug/deep-learning-in-matlab.html>. (accessed: 03.02.2025).
- [97] Muhammad Nadeem Cheema et al. “Liver Extraction Using Residual Convolution Neural Networks From Low-Dose CT Images”. In: *IEEE Transactions on Biomedical Engineering* (2019). DOI: 10.1109/TBME.2019.2894123.
- [98] Vasant Kearney et al. “Attention-enabled 3D boosted convolutional neural networks for semantic CT segmentation using deep supervision”. In: *Phys Med Biol* (2019). DOI: <https://doi.org/10.1088/1361-6560/ab2818>.
- [99] Daniel R Bourigault Emmanuelle and McGowan, Abolfazl Mehranian, and Bartłomiej W. Papież. “Multimodal PET/CT Tumour Segmentation and Prediction of Progression-Free Survival using a Full-Scale UNet with Attention”. In: *CoRR* (2021). DOI: <https://doi.org/10.48550/arXiv.2111.03848>.
- [100] Jo Schlemper et al. “Attention gated networks: Learning to leverage salient regions in medical images”. In: *Medical Image Analysis* (2019). DOI: <https://doi.org/10.1016/j.media.2019.01.012>.
- [101] He Tang et al. “Automatic Lumbar Spinal CT Image Segmentation With a Dual Densely Connected U-Net”. In: *IEEE Access* (2020). DOI: 10.1109/ACCESS.2020.2993867.

- [102] Syed Furqan Qadri et al. “OP-convNet: A Patch Classification-Based Framework for CT Vertebrae Segmentation”. In: *IEEE Access* (2021). DOI: 10.1109/ACCESS.2021.3131216.
- [103] Van Luan Tran, Huei-Yung Lin, and Hsaio-Wei Liu. “Multitask Deep Learning for Segmentation and Lumbosacral Spine Inspection”. In: *IEEE Transactions on Instrumentation and Measurement* (2022). DOI: 10.1109/TIM.2022.3184341.
- [104] Muhammad Usman Saeed et al. “An Automated Deep Learning Approach for Spine Segmentation and Vertebrae Recognition Using Computed Tomography Images”. In: *Diagnostics* (2023). DOI: <https://doi.org/10.3390/diagnostics13162658>.
- [105] Fabian Isensee et al. “nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation”. In: *Nat Methods* (2021). DOI: <https://doi.org/10.1038/s41592-020-01008-z>.
- [106] Jakob Wasserthal et al. “TotalSegmentator: Robust Segmentation of 104 Anatomic Structures in CT Images”. In: *Radiology: Artificial Intelligence* (2023). DOI: <https://doi.org/10.1148/ryai.230024>.
- [107] Samaneh Mostafapour et al. “Evaluation of Automatic Segmentation Tools on Low-Dose and Ultra-Low-Dose CT Images in PET/CT Scans”. In: *European Workshop on Visual Information Processing (EUVIP)* (2024). DOI: <http://dx.doi.org/10.1109/EUVIP61797.2024.10772868>.
- [108] Jieneng Chen et al. “TransUNet: Transformers Make Strong Encoders for Medical Image Segmentation”. In: *Medical Image Analysis* (2021). DOI: <https://doi.org/10.1016/j.media.2024.103280>.
- [109] Ali Hatamizadeh et al. “UNETR: Transformers for 3D Medical Image Segmentation”. In: *Winter Conference on Applications of Computer Vision* (2022). DOI: 10.1109/WACV51458.2022.00181.
- [110] Ali Hatamizadeh et al. “Swin UNETR: Swin Transformers for Semantic Segmentation of Brain Tumors in MRI Images”. In: *Image and Video Processing* (2022). DOI: <https://doi.org/10.48550/arXiv.2201.01266>.
- [111] Yufan He et al. “SwinUNETR-V2: Stronger Swin Transformers with Stagewise Convolutions for 3D Medical Image Segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention* (2023). DOI: https://doi.org/10.1007/978-3-031-43901-8_40.
- [112] Yucheng Tang et al. “Self-Supervised Pre-Training of Swin Transformers for 3D Medical Image Analysis”. In: *Conference on Computer Vision and Pattern Recognition* (2022). DOI: 10.1109/CVPR52688.2022.02007.
- [113] Xin You et al. “VerteFormer: A single-staged Transformer network for vertebrae segmentation from CT images with arbitrary field of views”. In: *International Journal of Medical Physics Research and Practice* (2023). DOI: <https://doi.org/10.1002/mp.16467>.
- [114] Xinchun Li et al. “VerFormer: Vertebrae-Aware Transformer for Automatic Spine Segmentation from CT Images”. In: *Diagnostics* (2024). DOI: <https://doi.org/10.3390/diagnostics14171859>.

- [115] Fedorov A. et al. “3D Slicer as an Image Computing Platform for the Quantitative Imaging Network”. In: *Magnetic Resonance Imaging* (2012). URL: <https://doi.org/10.1111/j.1742-4658.2007.05686.x>.
- [116] Rafael Gonzalez and Richard Woods. *Digital Image Processing*. Pearson, 2002.
- [117] Zongwei Zhou et al. “UNet++: A Nested U-Net Architecture for Medical Image Segmentation”. In: *Computer Vision and Pattern Recognition* (2018). DOI: <https://arxiv.org/abs/1807.10165>.
- [118] Huimin Huang et al. “UNet 3+: A Full-Scale Connected UNet for Medical Image Segmentation”. In: *Image and Video Processing* (2020). DOI: <https://arxiv.org/abs/2004.08790>.
- [119] Bing Xu et al. “Empirical Evaluation of Rectified Activations in Convolutional Network”. In: *Machine Learning* (2015). DOI: <https://arxiv.org/abs/1505.00853>.
- [120] Syed Javed et al. “Advancing Medical Image Segmentation with Mini-Net: A Lightweight Solution Tailored for Efficient Segmentation of Medical Images”. In: *Image and Video Processing* (2024). DOI: <https://arxiv.org/abs/2405.17520>.
- [121] Ange Lou, Shuyue Guan, and Murray Loew. “CFPNet-M: A Light-Weight Encoder-Decoder Based Network for Multimodal Biomedical Image Real-Time Segmentation”. In: *Computer Vision and Pattern Recognition* (2021). DOI: <https://doi.org/10.48550/arXiv.2105.04075>.
- [122] Tsung-Yi Lin et al. “Focal Loss for Dense Object Detection”. In: *IEEE International Conference on Computer Vision* (2017). DOI: 10.1109/ICCV.2017.324.
- [123] Ilya Loshchilov and Frank Hutter. “Decoupled Weight Decay Regularization”. In: *Machine Learning* (2019). DOI: <https://arxiv.org/abs/1711.05101>.
- [124] Kiatdd. *Correlation Figure*. URL: <https://commons.wikimedia.org/w/index.php?curid=37108966>. (accessed: 03.22.2025).
- [125] Charles J. Kowalski. “On the Effects of Non-Normality on the Distribution of the Sample Product-Moment Correlation Coefficient”. In: *Journal of the Royal Statistical Society* (1972). DOI: <https://doi.org/10.2307/2346598>.
- [126] Matt Vanderhoek, Scott B. Perlman, and Robert Jeraj. “Impact of Different Standardized Uptake Value Measures on PET-Based Quantification of Treatment Response”. In: *J Nucl Med* (2018). DOI: <https://doi.org/10.2967/jnumed.112.113332>.
- [127] Yang Deng et al. “CTSpine1K: A Large-Scale Dataset for Spinal Vertebrae Segmentation in Computed Tomography”. In: *Image and Video Processing* (2024). DOI: <https://arxiv.org/abs/2105.14711>.

1 Appendix A

Additional quantitative results. Included is a detailed table of each individual dice score for all 5 validation images compared against Total Segmentator. I also included the error metrics for every statistic computed on the SUV distribution of validation and test FLT-PET images as well as blox plots show SUV distribution per-vertebral body for each scan of one test patient. Additionally, I included a training curve for the model on purely HSCt images for double the iterations to prove that it indeed suffers from underfitting, rather than just having slower convergence.

Table A.1: Individual vertebral body Dice scores for my work compared to Total Segmentator. Highest avg scores are in bold.

Vertebra	P02D1		P08D1		P08D3		P10D1		P12D2	
	Proposed	TS	Proposed	TS	Proposed	TS	Proposed	TS	Proposed	TS
C1	0.73	0.22	0.61	0.18	0.67	0.19	0.52	0.21	0.73	0.16
C2	0.82	0.69	0.81	0.66	0.84	0.64	0.80	0.61	0.81	0.65
C3	0.79	0.69	0.79	0.74	0.68	0.64	0.77	0.67	0.73	0.68
C4	0.66	0.64	0.83	0.71	0.70	0.71	0.77	0.66	0.70	0.67
C5	0.69	0.57	0.85	0.69	0.72	0.64	0.67	0.58	0.74	0.67
C6	0.84	0.71	0.81	0.65	0.72	0.68	0.68	0.70	0.80	0.70
C7	0.64	0.83	0.77	0.71	0.76	0.77	0.71	0.75	0.76	0.74
Mean Cervical	0.74	0.62	0.72	0.62	0.73	0.61	0.70	0.60	0.75	0.61
T1	0.82	0.89	0.85	0.87	0.75	0.75	0.71	0.87	0.76	0.84
T2	0.88	0.92	0.84	0.87	0.87	0.89	0.78	0.83	0.71	0.81
T3	0.87	0.91	0.87	0.83	0.88	0.89	0.85	0.86	0.79	0.81
T4	0.81	0.66	0.86	0.87	0.91	0.88	0.90	0.91	0.78	0.79
T5	0.86	0.04	0.87	0.81	0.90	0.04	0.88	0.87	0.77	0.77
T6	0.82	0.31	0.80	0.31	0.82	0.01	0.88	0.91	0.83	0.84

Continued on next page

Vertebra	P02D1		P08D1		P08D3		P10D1		P12D2	
	Proposed	TS	Proposed	TS	Proposed	TS	Proposed	TS	Proposed	TS
T7	0.90	0.74	0.89	0.10	0.86	0.00	0.93	0.94	0.90	0.57
T8	0.86	0.63	0.88	0.59	0.91	0.41	0.89	0.92	0.95	0.00
T9	0.84	0.16	0.91	0.90	0.87	0.74	0.91	0.94	0.91	0.00
T10	0.83	0.25	0.92	0.95	0.87	0.87	0.94	0.93	0.88	0.00
T11	0.81	0.79	0.88	0.90	0.92	0.91	0.92	0.92	0.88	0.57
T12	0.91	0.91	0.91	0.93	0.88	0.86	0.92	0.94	0.93	0.87
Mean Thoracic	0.87	0.60	0.87	0.74	0.87	0.60	0.88	0.90	0.84	0.57
L1	0.92	0.92	0.93	0.93	0.94	0.92	0.92	0.93	0.93	0.94
L2	0.90	0.92	0.93	0.94	0.88	0.87	0.93	0.95	0.91	0.94
L3	0.88	0.88	0.94	0.80	0.91	0.59	0.91	0.91	0.94	0.95
L4	0.96	0.96	0.90	0.47	0.93	0.00	0.94	0.96	0.94	0.96
L5	0.90	0.91	0.90	0.00	0.94	0.00	0.94	0.94	0.91	0.89
L6	N/A	N/A	0.93	0.00	0.63	0.00	N/A	N/A	N/A	N/A
Mean Lumbar	0.91	0.92	0.92	0.63	0.87	0.48	0.93	0.94	0.93	0.93
Mean Overall	0.83	0.67	0.84	0.68	0.83	0.58	0.84	0.82	0.83	0.66

Table A.2: Set of error metrics after extracting FLT-PET SUV data from ROIs compared to physician measurements. All segmentations derive from images not present during training. Test cases are marked with *, all others are validation.

Patient	Measurement	MAE	RMSE	Correlation
P02	SUV Mean	0.642	0.982	0.994
P02	SUV Median	0.703	1.096	0.989
P02	SUV Min	1.011	1.648	0.773
P02	SUV Max	0.372	0.505	0.99
P02	STDV	0.26	0.417	0.849
P07*	SUV Mean	1.179	1.877	0.992
P07*	SUV Median	1.168	1.874	0.989
P07*	SUV Min	1.962	3.204	0.886
P07*	SUV Max	0.532	0.695	0.995
P07*	STDV	0.549	0.891	0.789
P08	SUV Mean	0.662	0.939	0.981
P08	SUV Median	0.679	0.988	0.979
P08	SUV Min	0.803	1.194	0.836
P08	SUV Max	0.633	0.756	0.963
P08	STDV	0.19	0.277	0.87
P10	SUV Mean	1.284	2.057	0.993
P10	SUV Median	1.352	2.147	0.991
P10	SUV Min	2.216	3.71	0.867
P10	SUV Max	0.494	0.775	0.994
P10	STDV	0.643	1.075	0.883
P12	SUV Mean	0.906	1.44	0.979
P12	SUV Median	0.867	1.469	0.964
P12	SUV Min	1.816	2.612	0.947
P12	SUV Max	0.628	1.145	0.975
P12	STDV	0.599	0.872	0.877
P21*	SUV Mean	0.949	1.193	0.945
P21*	SUV Median	1.008	1.274	0.941
P21*	SUV Min	1.335	1.631	0.702
P21*	SUV Max	0.915	1.215	0.894
P21*	STDV	0.254	0.317	0.903

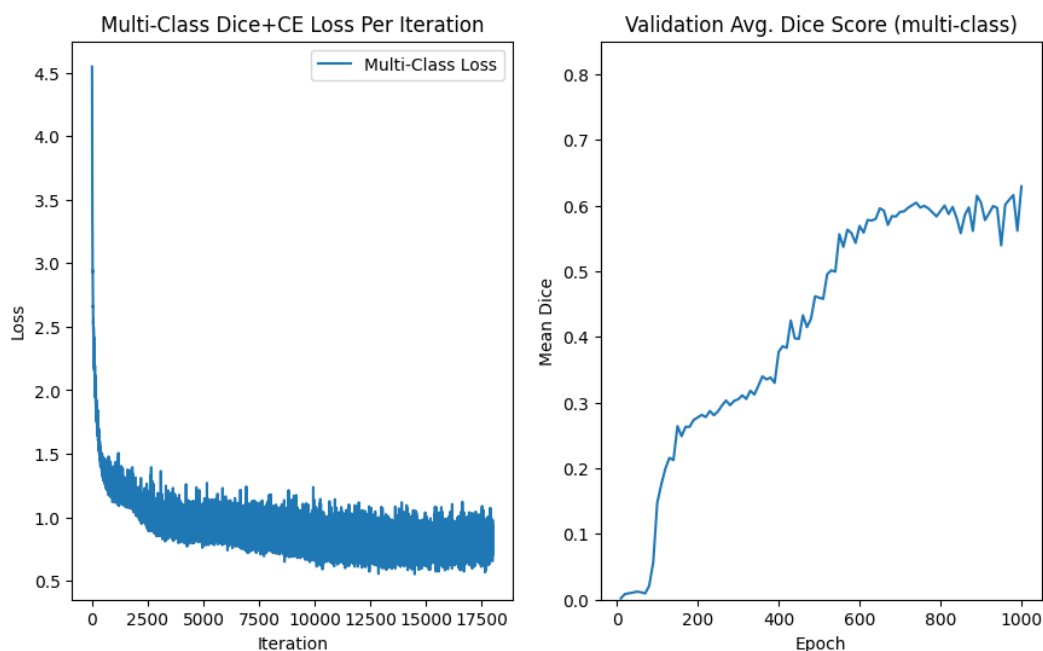


Figure A.1: Training my custom UNet over the same 22 HSCT images but with double the iterations. This further demonstrates the model’s tendency to underfit without data supplementation.

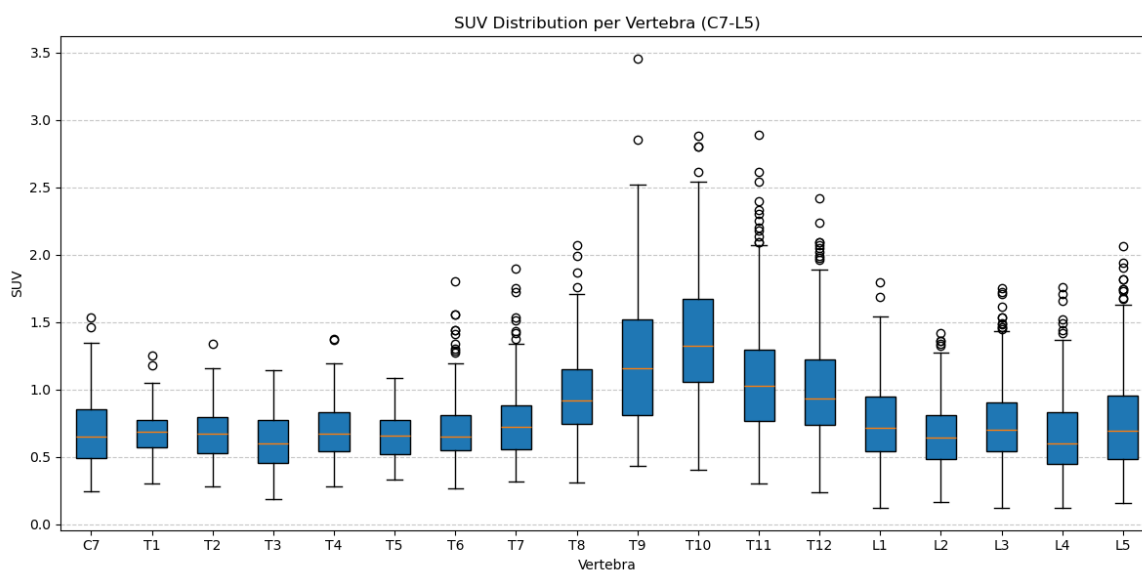


Figure A.2: Sample test Patient 7. SUV distribution on Day -1.

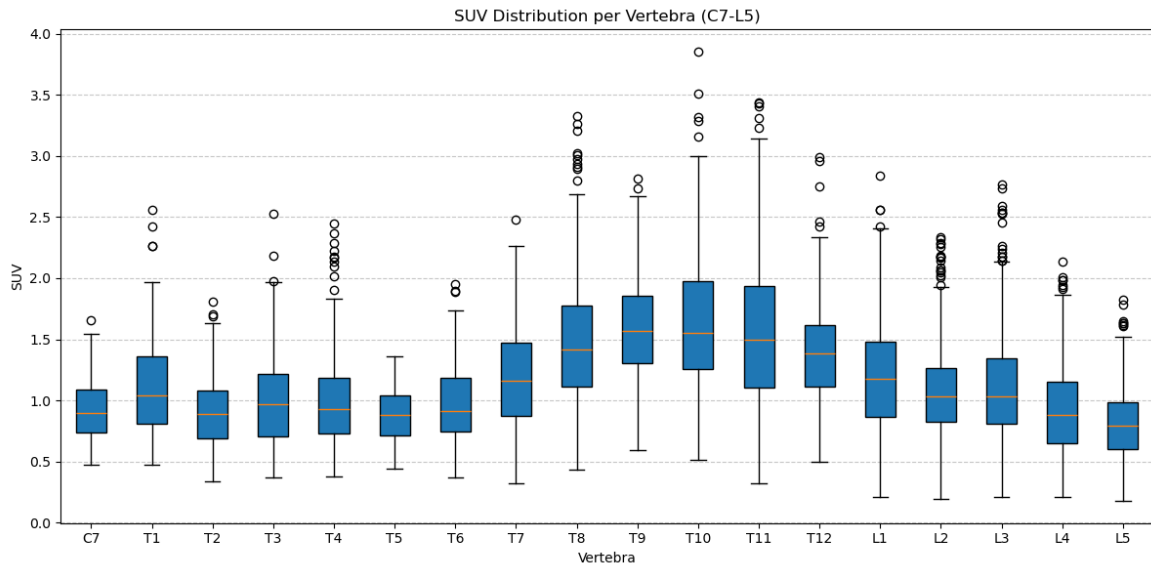


Figure A.3: Sample test Patient 7. SUV distribution on Day +5.

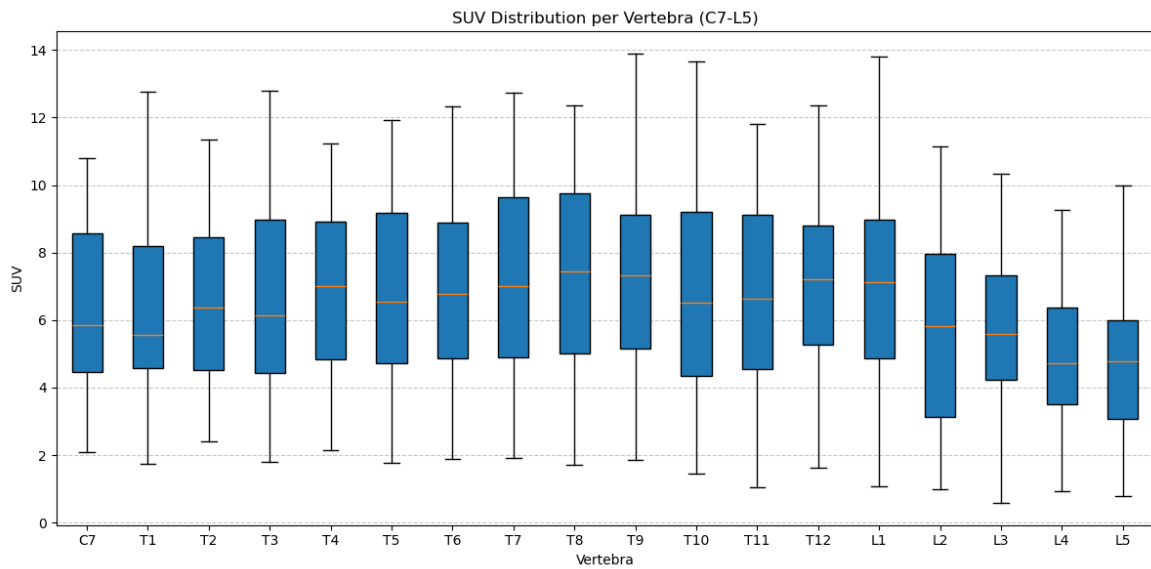


Figure A.4: Sample test Patient 7. SUV distribution on Day +28.

2 Appendix B

This second appendix contains more visualizations of the qualitative analysis. This includes 3D renderings of multiclass, individual vertebral body segmentation using my custom UNet, 3D examples of the VerSe connected component removal algorithm, 2D/3D hybrid difference maps for my model’s predictions on test cases, and isosurface volumes showing SUV proliferation.

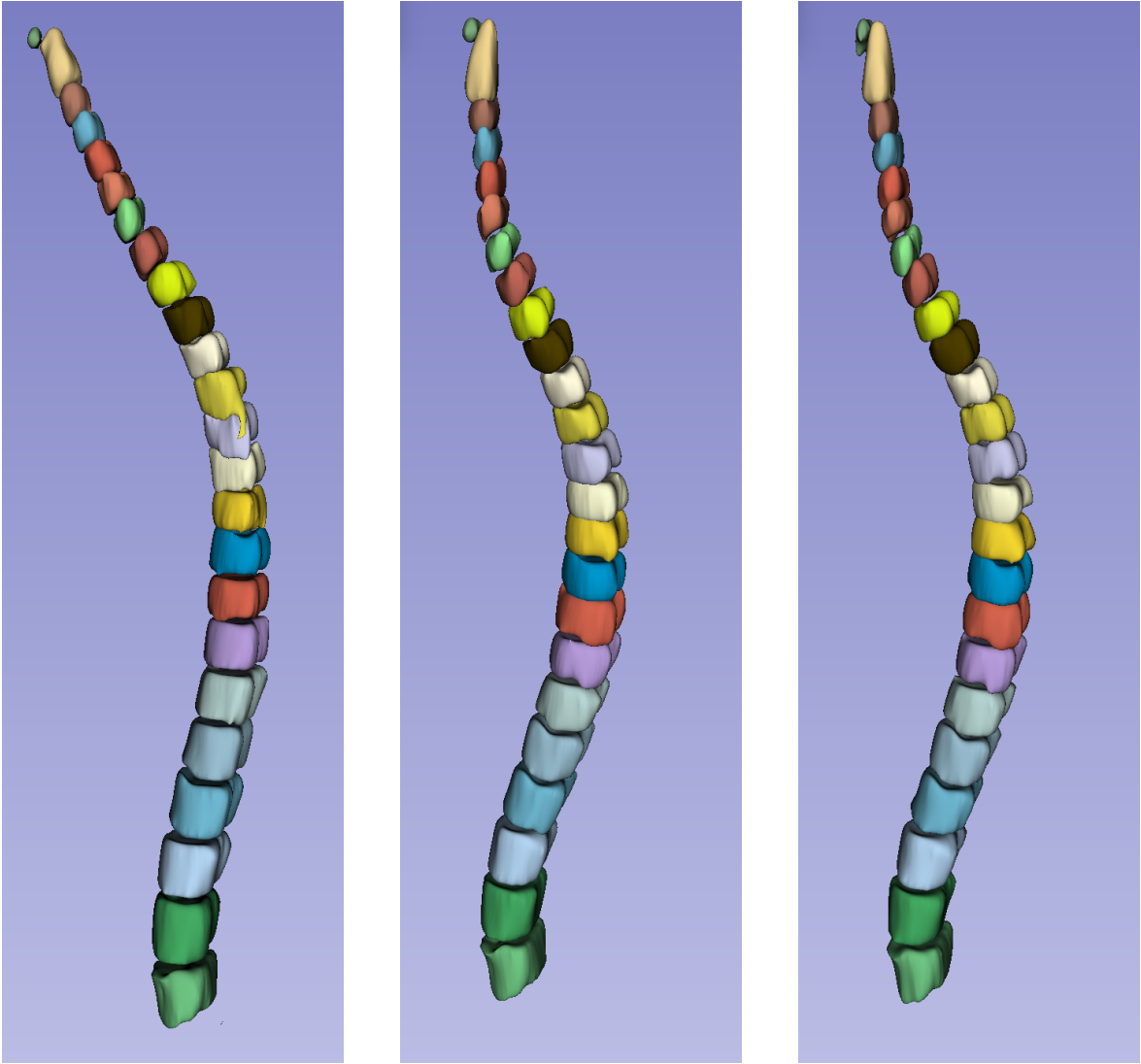


Figure B.1: 3D visualization of my model’s CT segmentation on test sample 15, which was not included in training or validation. Day -1 (left) to day +28 (right).

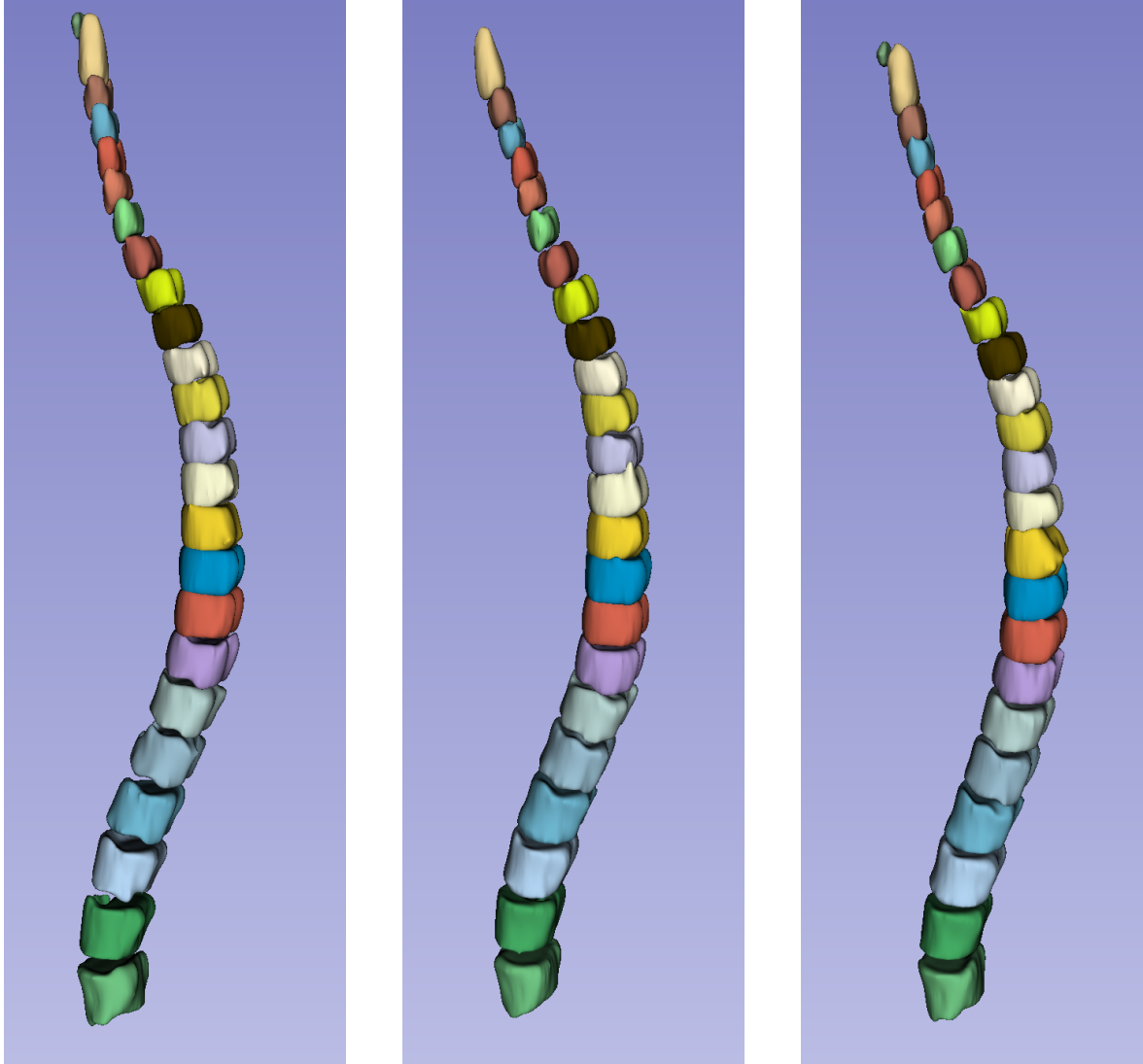


Figure B.2: 3D visualization of my model's CT segmentation on test case 21, which was also not included in training or validation. Day -1 (left) to day +28 (right).

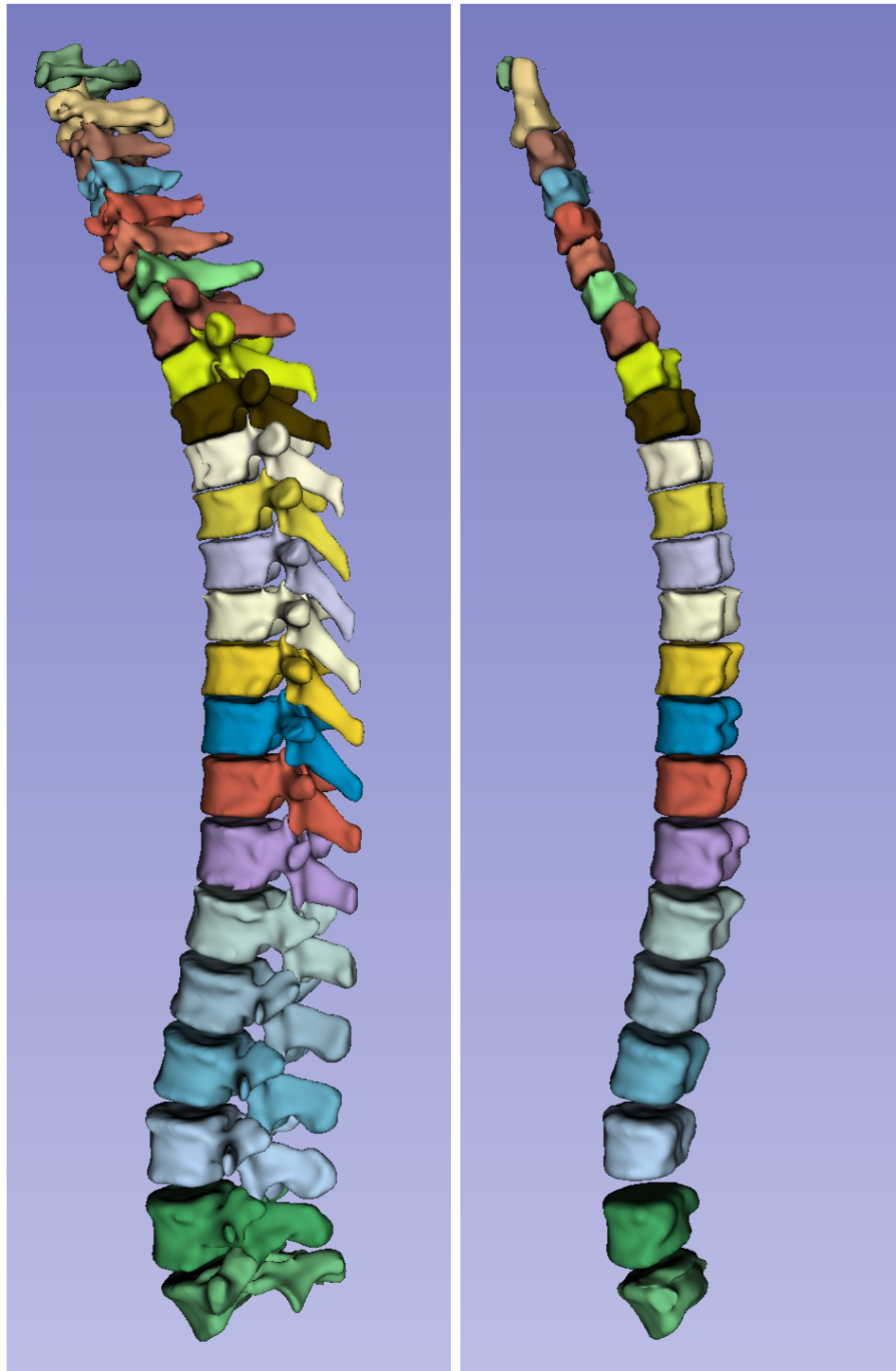


Figure B.3: 3D rendering of an original VerSe manual annotation (left) and modified label (right).

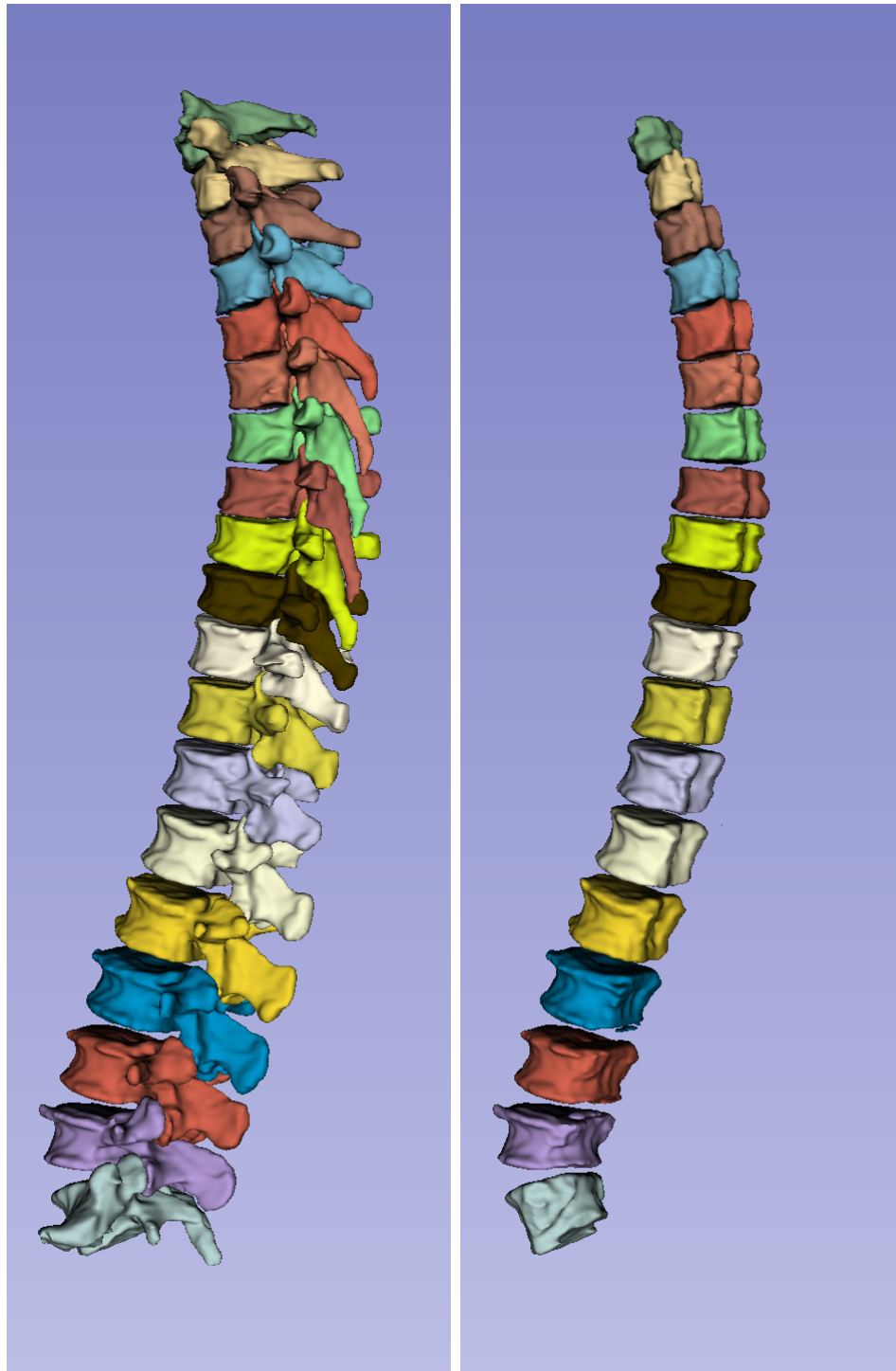


Figure B.4: A second 3D rendering that demonstrates my augmentation algorithm on the VerSe labels.

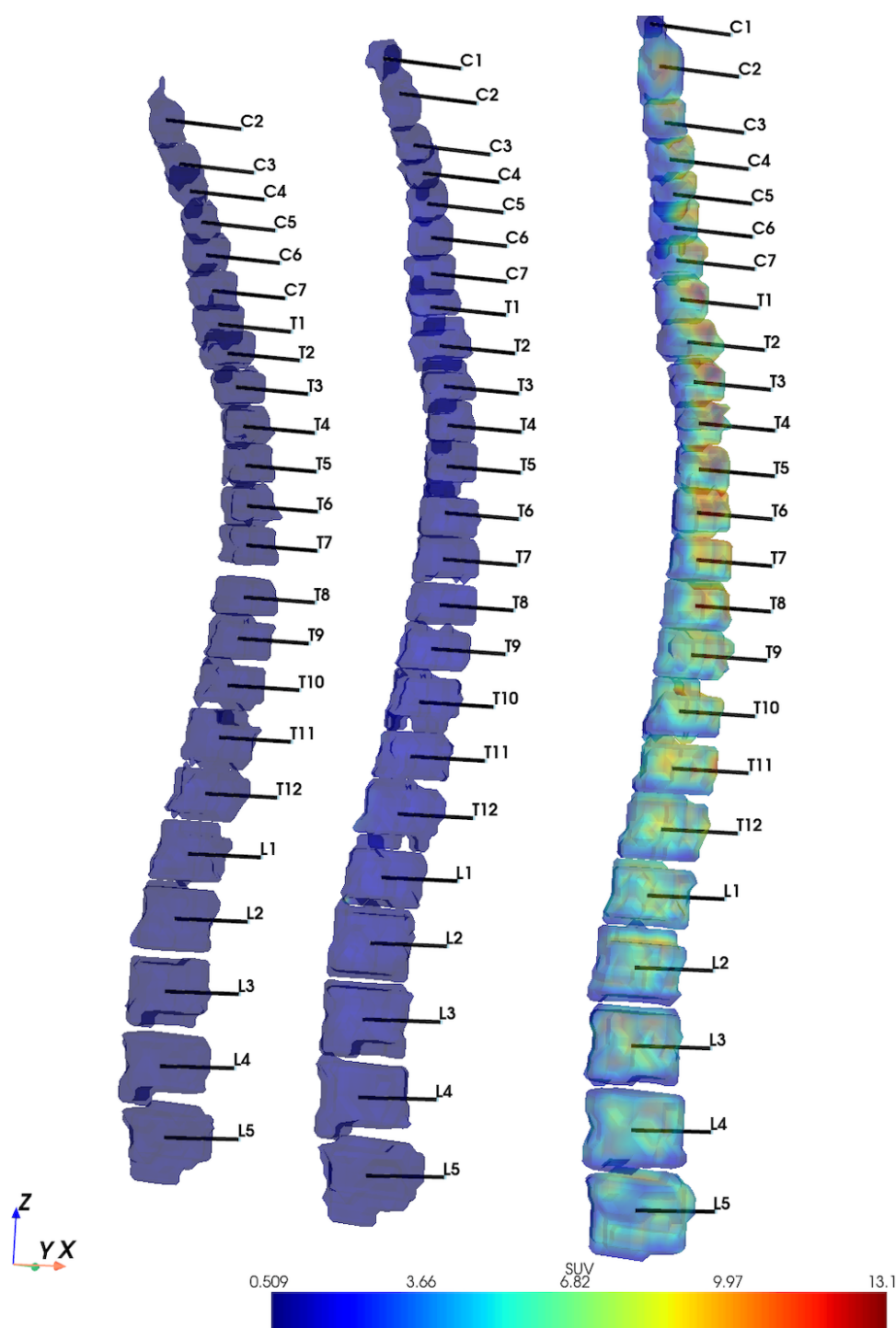


Figure B.5: FLT-PET visualization of vertebral ROIs illustrating HSC metabolic progression on day -1 (left), day +5 (middle), and day +28 (right) on a validation case. It ranges from 20th percentile (dark blue) to 99th percentile (dark red).

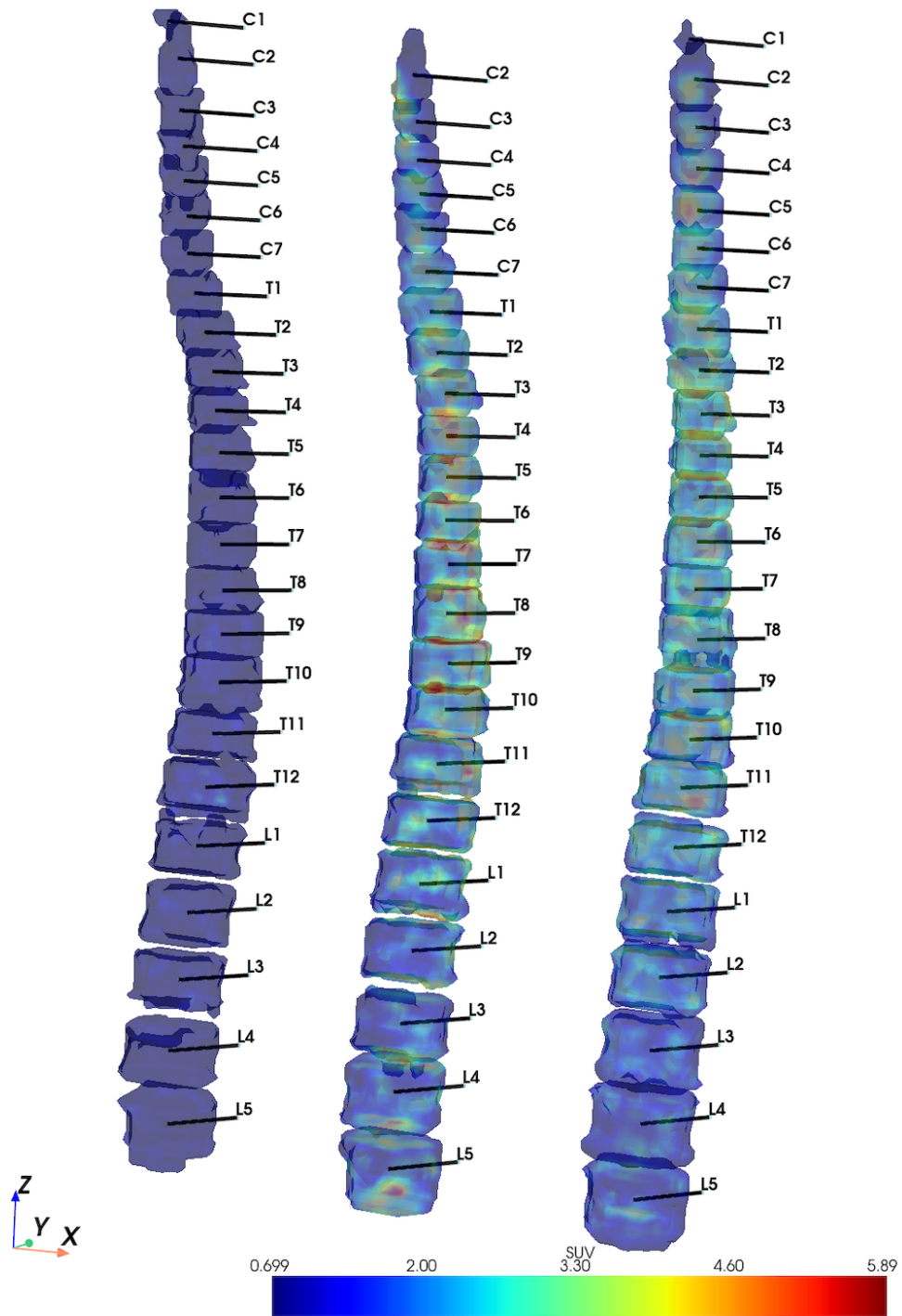


Figure B.6: FLT-PET visualization of vertebral ROIs illustrating HSC metabolic progression on day -1 (left), day +9 (middle), and day +28 (right) on a test case. It ranges from 20th percentile (dark blue) to 99th percentile (dark red).

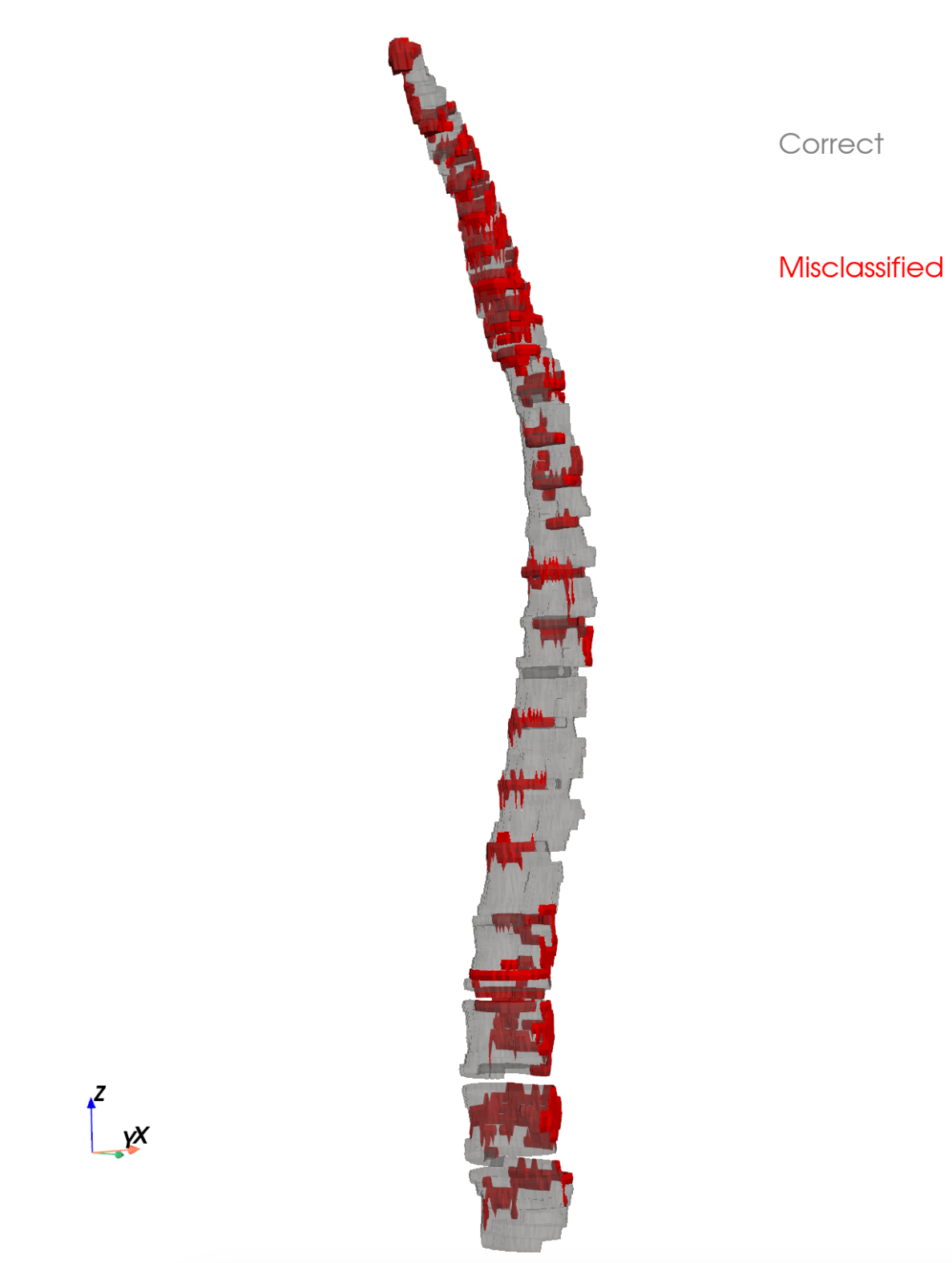


Figure B.7: 3D volumetric difference map on my model's predictions of P10D1.

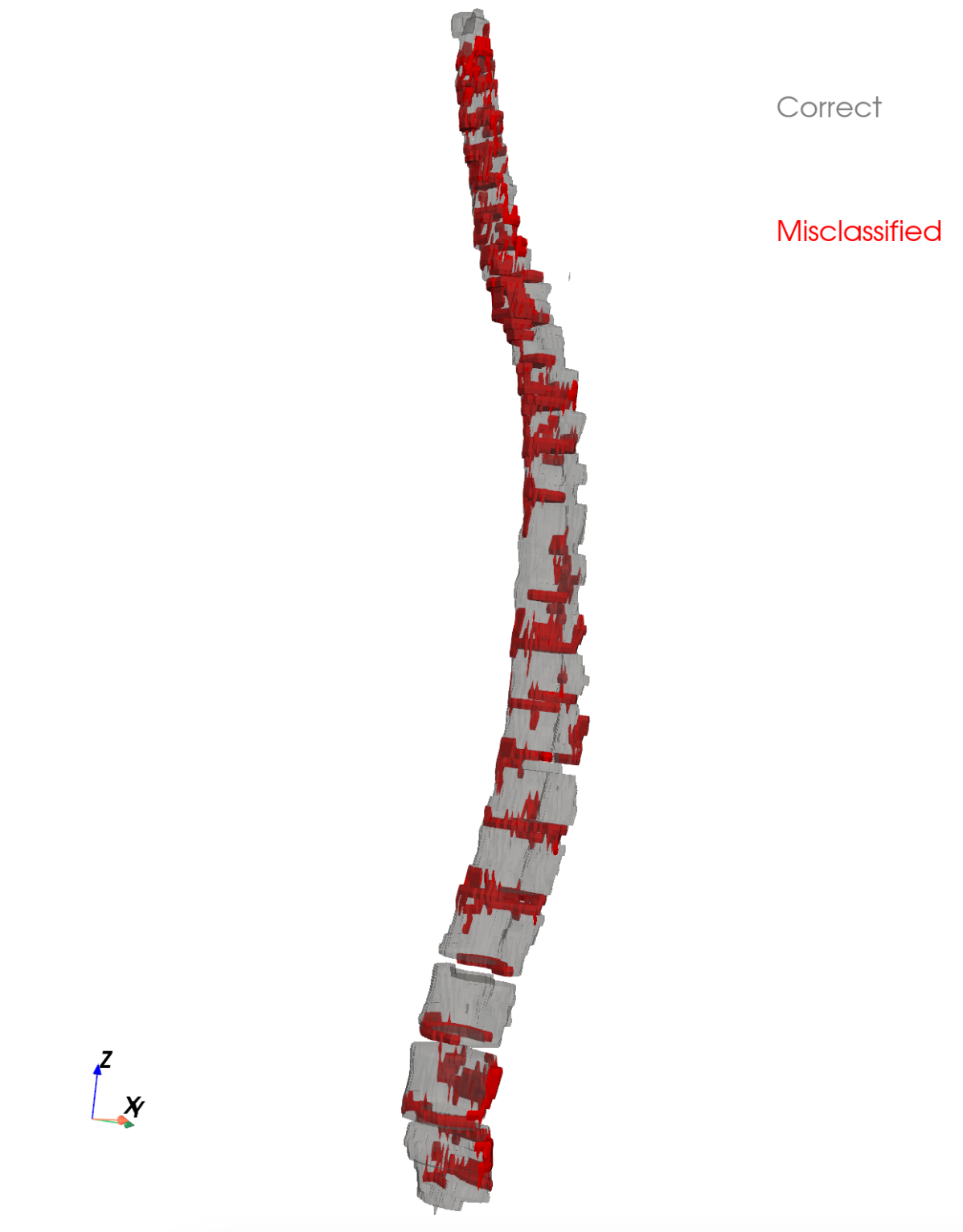


Figure B.8: 3D volumetric difference map on my model's predictions of P12D2.

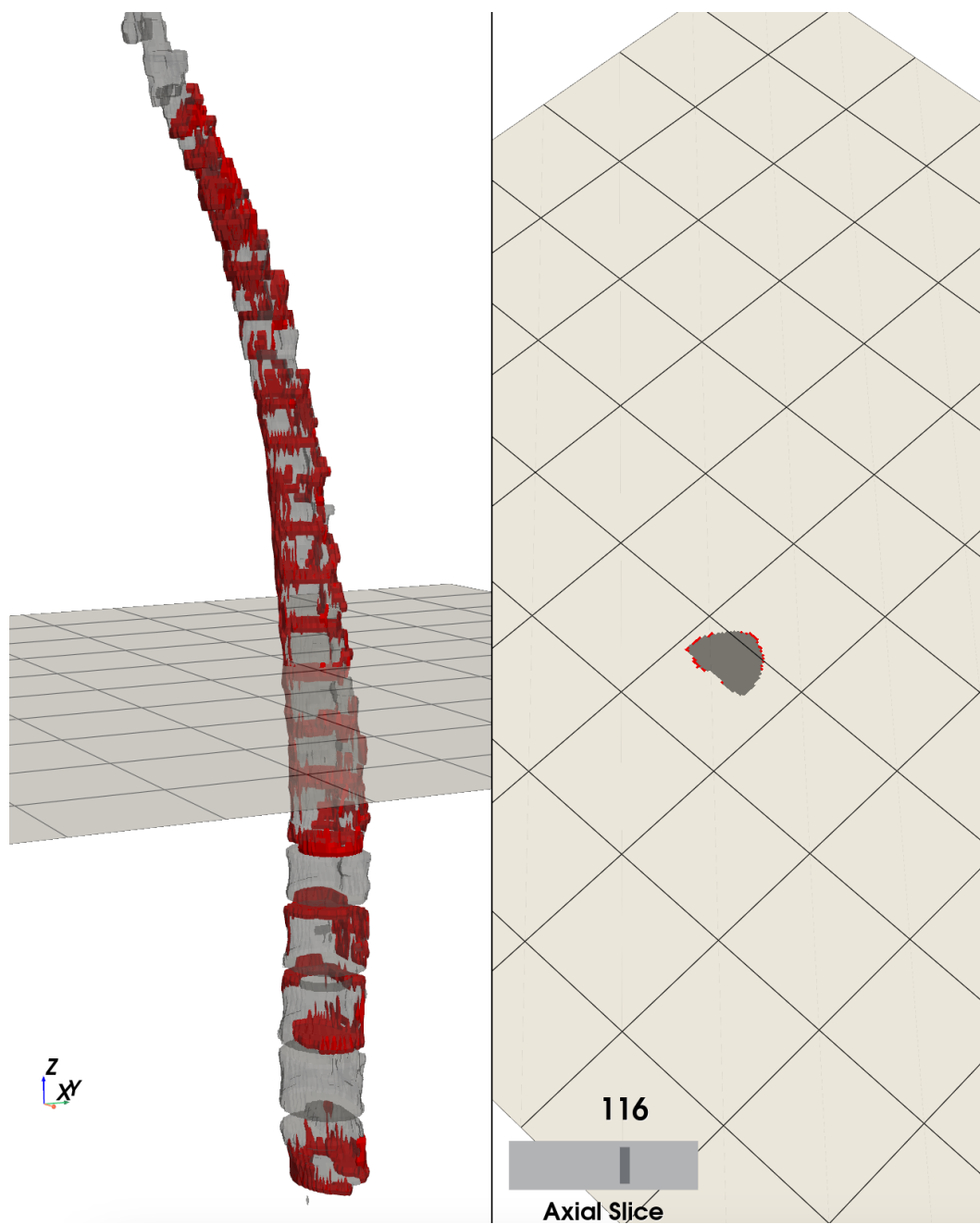


Figure B.9: 3D volumetric difference map on my model's predictions of P02D1 showing the slice plane.

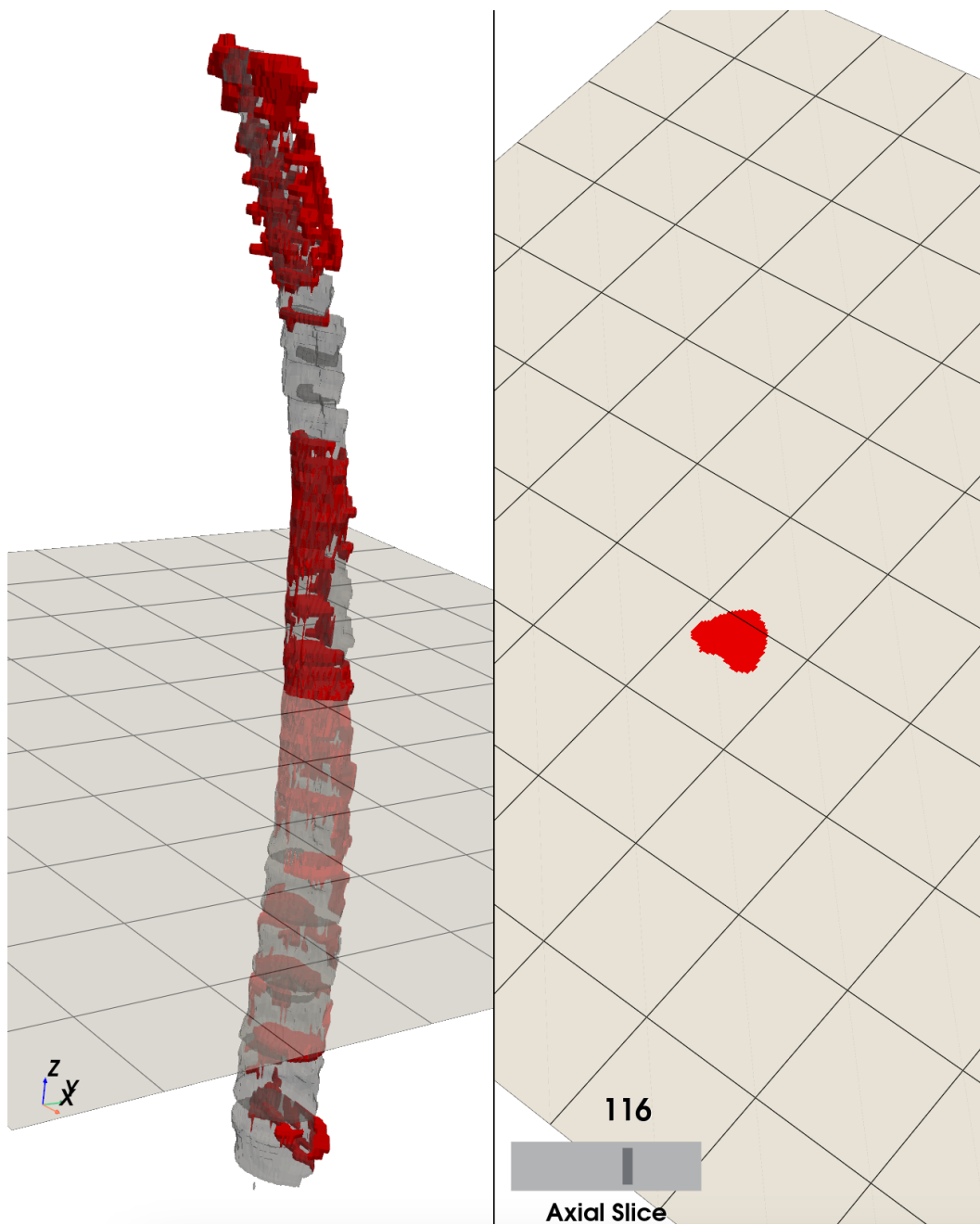


Figure B.10: 3D volumetric difference map on Total Segmentator's prediction of P02D1 showing the slice plane