Neural-Network-Based Detection of Radiopharmaceutical Extravasation In PET/CT Data

A Thesis Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Elijah D Berberette

August 2024

ACKNOWLEDGEMENTS

Thank you to Dr. Jens Gregor for being my primary advisor. Thank you to Dr. Dustin Osborne and The UT Medical Center for providing both data and medical expertise in the field of radiology. Thank you to Dr. Hector Santos for feedback on how to improve the neural network's architecture. Thank you to external, domain-specific advice provided by Jack Hutchins and Dray Gentry to ensure a well-written thesis.

ABSTRACT

The immediate identification of PET/CT radiopharmaceutical extravasation can eliminate many adverse effects such as misdiagnosis and improper therapy. Radiopharmaceutical extravasation is the leakage of an injected radiotracer from the patient's intended vein into surrounding tissues. The detection of this phenomenon often requires the use of an external monitoring device; due to a lack of robust visual features that can provide indication that it has occurred. In this thesis, the feasibility of using neural networks trained on PET/CT data to identify extravasation is explored. This approach begins with a novel preprocessing methodology that automatically extracts body weight normalized standard uptake values (SUVbw) from specific regions of interest (ROIs). These measurements are then classified using fully connected artificial neural networks (ANNs) to determine if the patient was extravasated. An alternative approach is examined, where convolutional neural networks (CNNs) are leveraged to explore previously undiscovered features through automatic feature extraction. These two approaches were able to differentiate extravasated from non-extravasated patients with high performance. With additional data and architectural improvements, this work serves as a direction for future neural-network-based PET/CT extravasation detection.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION 1
1.1 Problem Overview 1
1.1.1 Positron Emission Tomography 1
1.1.2 Computed Tomography 2
1.1.3 PET/CT 3
1.1.4 Radiopharmaceutical Extravasation 5
1.1.5 Difficulties in Detection 5
1.1.6 Current Detection Strategy 7
1.2 An Alternative Approach 7
1.2.1 DICOM 9
1.2.2 NIfTI 10
1.2.3 PET/CT Segmentation Pipeline 10
1.2.4 Image Segmentation 12
1.2.5 TotalSegmentator 12
1.2.6 Classification of PET/CT Data 13
1.3 Outline of Thesis 13
CHAPTER 2: A PREPROCESSING METHODOLOGY FOR AUTOMATIC FEATURE EXTRACTION 15
2.1 Overview 15
2.1.1 Data Assumptions 15
2.2 Data Preparation 16
2.3 Data Segmentation & SUVbw Extraction 21
2.3.2 Liver ROI 21
2.3.3 Aorta ROI 24
2.3.4 Cerebellum ROI 24
2.3.5 Calculating SUVbw Metrics 24
2.3.6 Extendibility 26
2.4 Results 29
CHAPTER 3: APPLICATION OF A MULTI-LAYER ARTIFICIAL NEURAL NETWORK 34
3.1 Overview 34
3.2 Neural Network Fundamentals 34
3.2.1 Fully Connected Artificial Neural Networks 34
3.2.2 Loss Functions 38
3.2.3 Backpropagation and Gradient Descent 40
3.2.4 Hyperparameters 40
3.2.5 Performance Metrics 40
3.3 Methods 42
3.3.1 Constructing a Dataset 42
3.3.2 Sequential Selection Classification 42
3.3.3 Classifying Highest Extravasation Cases 44
3.3.4 K-Fold Cross-Validation 44

3.3.5 Feature Selection 44
3.3.6 Model Architecture 47
3.4 Results 47
3.4.1 Sequential Selection Classification Results 49
3.4.2 Extreme Extravasation Classification Results 49
3.4.3 K-Fold Validation Results 49
CHAPTER 4: APPLICATION OF A CONVOLUTIONAL NEURAL NETWORK 55
4.1 Overview 55
4.1.1 Convolutional Neural Networks 55
4.2 Methods 56
4.2.1 Entire Body Single Slice Classification 56
4.2.2 Individual Area Single Slice Classification 57
4.2.3 Slice Clustering Classification 57
4.2.4 Data Augmentation 60
4.2.5 Increasing Training Data 60
4.2.6 Model Architecture 62
4.3 Results 62
4.3.1 Slice-by-Slice Classification Results 62
4.3.2 Slice Clustering Classification Results 64
4.3.3 The Effect of Increasing Training Data 64
CHAPTER 5: CONCLUSIONS 70
5.1 Shortcomings/Limitations 71
5.1.1 Lack of Data 71
5.1.2 Addressing the Reproducibility of the CNN Results 71
5.2 Future Work 72
5.2.1 3D Convolutional Neural Networks 72
5.2.2 Improvements in the SUVbw Extraction Pipeline 72
5.2.3 Model Explainability 73
LIST OF REFERENCES 74
VITA 78

LIST OF TABLES

Table 1.1 Lucerno intensity scale for extravasation [22,23] 8
Table 2.1 Percent differences between the measurements the pipeline produced for the reduced liver ROI and the measurements extracted by a trained radiologist 30
Table 2.2 Percent differences between the measurements the pipeline produced for the reduced cerebellum ROI and the measurements extracted by a trained radiologist . 31
Table 2.3 Percent differences between the measurements the pipeline produced for the reduced aorta ROI and the measurements extracted by a trained radiologist 32
Table 3.1 ANN results for the sequential classification approach. 50
Table 3.2 ANN summarization table for the extreme extravasation cases validation results. 52
Table 3.3 ANN results for K-Fold validation. 53
Table 4.1 Slice-by-slice classification results for the entire body, aorta, liver, brain, and ROI within the liver. 66
Table 4.2 Slice clustering classification results for the entire body, aorta, liver, brain, and ROI within the liver. 66
Table 4.3 Results from applying data augmentation and an increase in the validation split. 67
Table 4.4 Results from applying data augmentation, slice clustering, and using only two patients for validation data. 68

LIST OF FIGURES

Figure 1.1 An axial view of a slice that each medical imaging technique produces: a) CT slice, b) PET slice, c) PET/CT Fused Slice 4
Figure 1.2 A coronal view of a slice that each medical imaging technique produces: (a) CT slice, (b) PET slice, (c) PET/CT Fused Slice 4
Figure 1.3 A sagittal view of a slice that each medical imaging technique produces: (a) CT slice, (b) PET slice, (c) PET/CT Fused Slice 4
Figure 1.4 Extravasation is difficult to determine visually: (a) patient is not extravasated; (b) patient is extravasated. 6
Figure 1.5 A snippet of DICOM information extracted from one of the patients in the dataset 11
Figure 1.6 DICOM information showing anonymized patient information that is used as features in neural network training exercises 11
Figure 1.7 DICOM information for a particular slice highlighting key attributes used for resizing 11
Figure 1.8 A compilation of the workflow containing an overview of each layer followed by a red/green light that indicates whether a patient is extravasated (red) or not extravasated (green) 14
Figure 2.1 A workflow chart capturing the various decisions made when preparing the DICOM data for segmentation 17
Figure 2.2 A CT slice being scaled back to the size of the original scan field using the pixel spacing provided in the DICOM header 19
Figure 2.3 A PET slice being scaled back to the size of the original scan field using the pixel spacing provided in the DICOM header. 19
Figure 2.4 A PET slice cropped to have equal dimensions as the corresponding downsampled CT slice. 20
Figure 2.5 Initial masks produced by TotalSegmentator: (a-c) brain, (d-f) liver, (g-i) aorta 22
Figure 2.6 An illustration of how a cube is created to create the ROI within the liver. The faint line represents the initial cross-section and the expansion represents the additional slices. 23
Figure 2.7. An ROI produced by the pipeline shown from different perspectives: (a) axial, (b) coronal, (c) sagittal 23
Figure 2.8 The two boundaries of the aorta ROI: (a) Superior boundary with the carina highlighted, (b) Inferior boundary 25
Figure 2.9 The two boundaries of the cerebellum ROI: (a) Superior, (b) Inferior 25
Figure 2.10 An overview of how the TotalSegmentator masks are used to create ROIs and calculate each of the features produced by the pipeline 28
Figure 3.1 CT data fed into the neural network where each pixel is an individual neuron 36
Figure 3.2 Final layers of a fully connected neural network where the final neuron is a binary prediction 36

Figure 3.3 Outputs layers for specific tasks: (a) binary output, (b) continuous output, (c) categorical output 37
Figure 3.4 Commonly used activation functions in neural networks 37
Figure 3.5 A depiction of the process used to create a balanced dataset using a Lucerno score of 700 as the recommended threshold. 43
Figure 3.6 A graph depicting how 26 principal components can be used to capture 99% of the cumulative variance of the original data. 46
Figure 3.7 A row of features produced by the pipeline being plugged into the chosen fully connected architecture. 48
Figure 3.8 Correct and incorrect validation guesses produced by the model when trained on MMM, patient weight, and patient age features. 50
Figure 3.9 Correct and incorrect validation guesses produced by the model when trained on the alpha-trimmed mean (10%), variance, skewness, kurtosis, patient weight, and patient age features. 51
Figure 3.10 Correct and incorrect validation guesses produced by the model when trained on MMM, MMM ratios, patient weight, and patient age features. 51
Figure 3.11 Correct and incorrect validation guesses produced by the model when trained on all features except the patient's age. 51
Figure 3.12 Correct and incorrect validation guesses produced by the model when trained on all of the features using a dropout layer in architecture. 52
Figure 3.13 Correct and incorrect validation guesses produced by the model when trained on all of the features. 52
Figure 3.14 Loss plots from the K-Fold Validation exercises: (a) MMM, Weight, and Age, (b) ATM, Skewness, Variance, Kurtosis, Weight, Age (c) MMM, Ratios, Weight, Age, (d) All data excluding age, (e) All data with dropout layer, (f) All data without dropout layer 54
Figure 4.1 Slices corresponding to the organs of interest used for training data 58
Figure 4.2 Slice clustering illustrated to show how the collective confidences for each slice produce a prediction for a single patient 59
Figure 4.3 Data augmentation techniques used to generate 20 separate data points for a single PET slice 61
Figure 4.4 Convolutional neural network architecture designed to ingest a CT slice and output extravasated or not extravasated. 63
Figure 4.5 Accuracy plots for the 96% validation split training exercises: (a) No data augmentation, (b) Augmented data 67
Figure 4.6 Loss plots for the 96% validation split training exercises: (a) No data augmentation, (b) Augmented data 68
Figure 4.7 Loss plots for 98% validation split training: (a) Not Augmented, (b) Augmented 69

CHAPTER 1:
INTRODUCTION

1.1 Problem Overview

Advancements in medical imaging have revolutionized the ability to identify, diagnose, and eliminate underlying health impediments in an unprecedented fashion.
At the forefront of these advancements lies the combination of positron emission tomography (PET) and computed tomography (CT) into a single diagnostic imaging process called PET/CT. The mixture of these two medical imaging techniques provides medical institutions and researchers with the unparalleled capability to directly address the overarching challenge of cancer, tumor assessment, and other obscure health conditions.

This procedure includes administering a radiopharmaceutical agent into the patient's arm and feeding the patient into a PET/CT scanner. PET is used to produce an image of the patient's internal metabolic activity. CT produces an anatomical image which is used to correct the captured metabolic activity, providing a better understanding of the patient's condition. During the administration process of the radiopharmaceutical tracer used for PET arises a notable challenge and the key focus of this thesis, radiopharmaceutical extravasation.

Before one can fully understand radiopharmaceutical extravasation, it is important to provide a general overview and the process for employing these particular medical imaging methodologies. The following content captures this background, providing the reader with additional information about PET and CT.

1.1.1 Positron Emission Tomography

PET is a medical imaging technique used to visualize and assess metabolic activity in the human body [1,2]. Metabolic, in this context, refers to the biochemical reactions that

arise inside the human body that provide the body with essential energy [3,4]. A PET scan begins with the administration of a radiopharmaceutical tracer. A radiopharmaceutical tracer is a chemical compound in which a radioisotope is used to replace one or more of the underlying atoms [5]. One of these radioisotopes is fluorine-18 [6], which is used to produce the radiotracer 18F-fluorodeoxyglucose (18F-FDG). 18F-FDG is a heavily used radiotracer in PET/CT due to its unique, multipurpose capability, such as cancer detection and brain metabolism [1].

After injecting the patient with this radiopharmaceutical, the patient is placed in a PET scanner which detects pairs of coincident photons produced by the annihilation of an emitted positron encountering an electron within the body [1]. A three-dimensional image is reconstructed that quantifies where these annihilations and thus the metabolic activity took place within the patient [7]. This three-dimensional image is composed of voxels, measured in Bq/ml, which define the volume of the radiotracer in each particular area. Attenuation correction is used to compensate for the signal loss that occurred due to photons being absorbed by dense matter, such as bone [8]. An example of a PET slice from an axial, coronal, and sagittal perspective can be seen in Figure 1.1 (a), Figure 1.2 (a), and Figure 1.3 (a), respectively.

1.1.2 Computed Tomography

CT is used to visualize anatomical structures by reconstructing the collected data into cross-sectional slices [9]. These anatomical structures include, but are not limited to, bones, organs, and the vascular system. An example of a cross-section can be seen in Figure 1.1 (b), Figure 1.2 (b), and Figure 1.3 (b).

Seeram et al [10] break the CT scan process into three essential steps: data acquisition, image reconstruction, and image interpretation. Data acquisition involves placing the patient on a table where an X-ray tube and a colinear set of detectors collect X-ray

projection images as the assembly rotates around the patient. The X-ray tube emits photons that are transmitted through the patient and interact with the detectors [11].

Earlier models of CT scanners captured a single slice at a time and then incrementally moved the patient axially. The successor to this model was the helical or spiral CT scanner which allowed for the patient to move continuously through the scanner. This model came with two major advantages: the ability to capture volumes of data and the ability to capture anatomical structures in a single breath [12]. Following helical CT scanners came multi-slice CT scanners (MSCT), which are actively used today [13]. These models are able to capture 64 slices in a single rotation of the X-ray tube and detector.

Each rotation of the gantry and detectors captures a collection of electrical signals that must be converted to digital data [10]. This is done through the use of image reconstruction algorithms. There are two common families of algorithms used for CT reconstruction, filtered backpropagation and iterative techniques [14]. These algorithms convert the measured signals to Hounsfield units. Hounsfield units (HU) measure the radio density of the volume in each voxel of the slice [15]. Radiologists use these reconstructed images for diagnostic interpretation to identify various underlying issues such as tumors in the brain [16].

1.1.3 PET/CT

PET/CT corrects the metabolic information produced by PET using the anatomical information captured by CT through attenuation and scatter correction. CT also provides anatomical correlation to the functional imaging. For example, if a “hot spot” is identified on PET that has an anatomical correlate, that is a strong sign of a malignant disease. The process of combining these two imaging techniques is achieved through overlaying or “fusing” the PET and CT slices [1]. An example of this fusion can be seen in Figure 1.1 (c), Figure 1.2 (c), and Figure 1.3 (c) below.

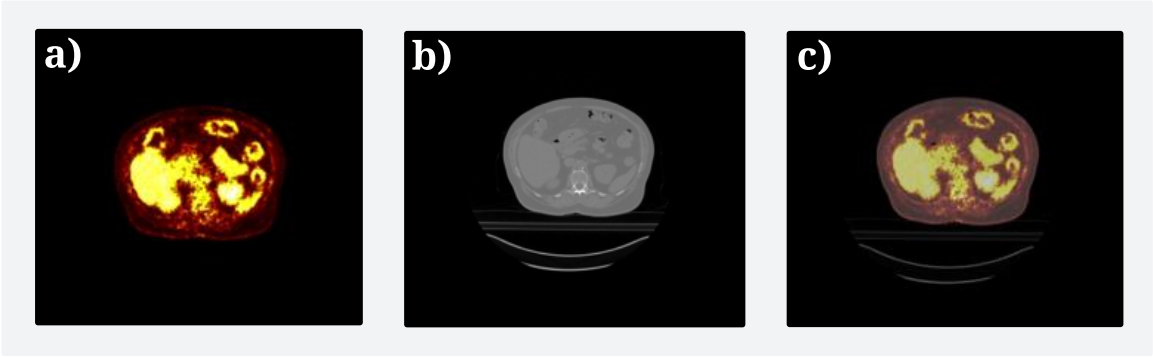


Figure 1.1 An axial view of a slice that each medical imaging technique produces: a) CT slice, b) PET slice, c) PET/CT Fused Slice

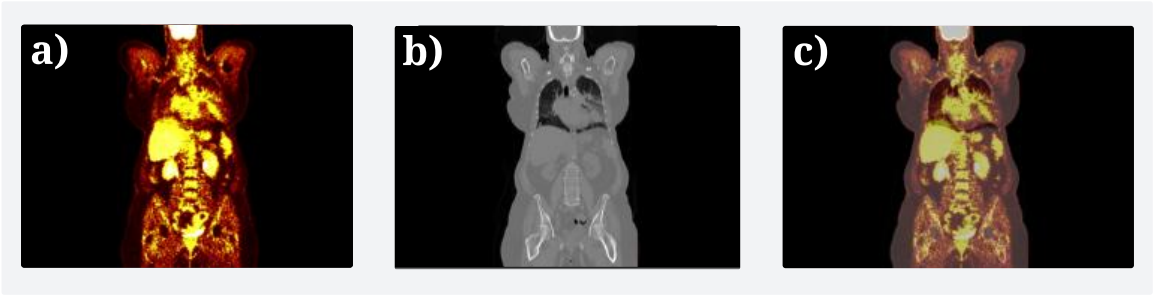


Figure 1.2 A coronal view of a slice that each medical imaging technique produces: (a) CT slice, (b) PET slice, (c) PET/CT Fused Slice

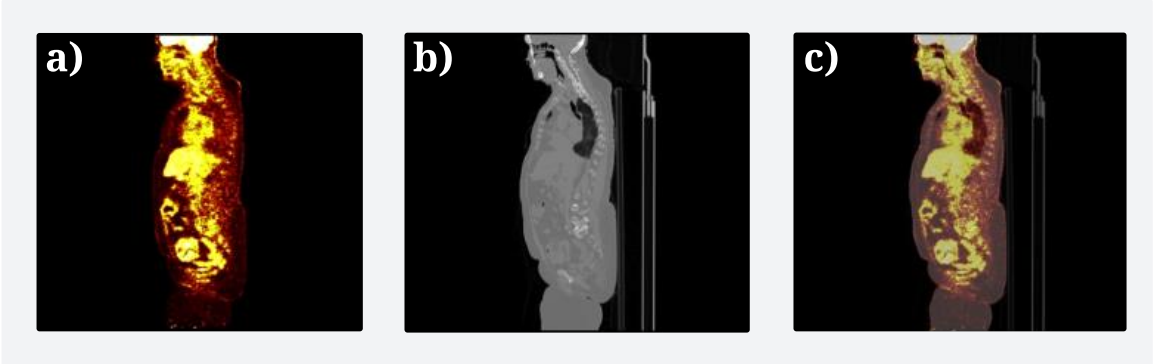


Figure 1.3 A sagittal view of a slice that each medical imaging technique produces: (a) CT slice, (b) PET slice, (c) PET/CT Fused Slice

1.1.4 Radiopharmaceutical Extravasation

During the PET scan, the injection of the radiopharmaceutical occasionally becomes extravasated. Radiopharmaceutical extravasation is defined as the leakage of a radiopharmaceutical agent, such as 18F-FDG, from the intended vein into surrounding tissues [17]. High levels of extravasation can potentially result in the irradiation of patient tissue and inaccurate diagnostic images [18]. This is unfavorable because it can lead to inaccurate diagnosis, resulting in unnecessary therapy. With the possibility of extravasation occurring in over 18% of examinations [18], it is crucial that medical institutions pursue detection efforts for this exacerbation. Although the complete prevention of medical radiopharmaceutical extravasation is an ambitious, potentially unattainable goal [19], consistent detection and reduction of extravasation is essential for ensuring patient safety for the near future.

1.1.5 Difficulties in Detection

It is possible to see extravasation arise in the patient's injection arm, which appears as a noticeable high-activity region. One issue with this detection method is that the patient's arm is not always in the field of view [20]. This poses a problem for radiologists because it removes a key identifier for visual detection of extravasation. Major extravasations can often be seen in other areas slices of the PET scan; however, it is not necessarily the case that all extravasation events will compromise the underlying imaging. Figure 1.4 below shows that even in the most extreme cases, extravasated patients appear similar to non-extravasated patients in the PET scan.

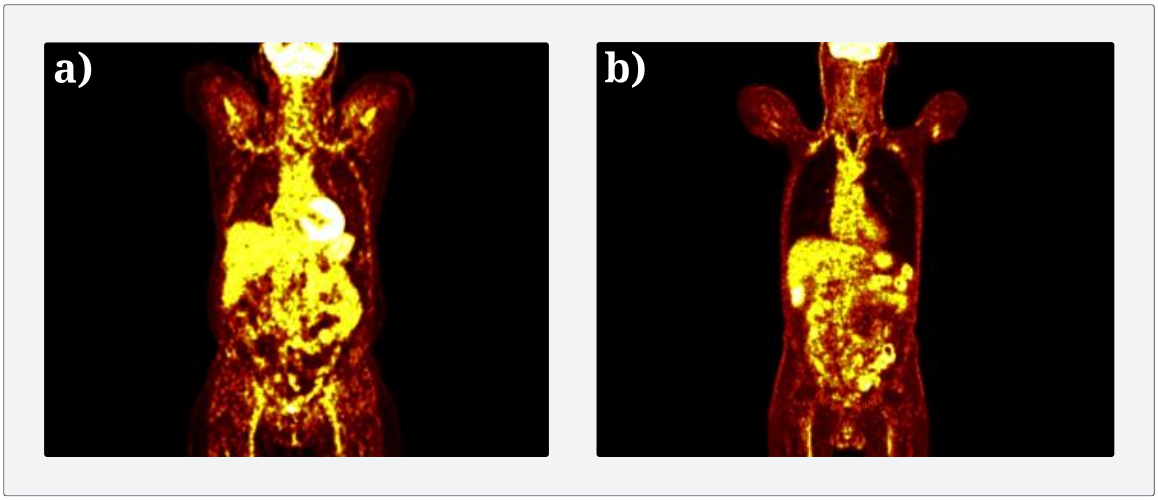


Figure 1.4 Extravasation is difficult to determine visually: (a) patient is not extravasated; (b) patient is extravasated.

1.1.6 Current Detection Strategy

The best existing method for detecting radiopharmaceutical extravasation is the use of an external monitoring device, such as the LARA System (Lucerno Dynamics, LLC) [22]. This approach uses two radiation sensors to monitor the injection of the radiopharmaceutical agent.

When delivering the radiotracer into the patient's arm, the medical professional selects one arm to act as a reference point and the other arm to measure the activity of the radiopharmaceutical being injected into the patient. As the radiopharmaceutical is injected, these sensors produce a time-activity curve that the medical professional can monitor in real time to identify if and when extravasation has occurred. The time-activity curve produces a score that can quantify the significance of the extravasation that arises.

Table 1.1 captures the four tags that Lucerno uses to label each of the extravasations. The score can be converted into the amount of radiotracer that was leaked into the patient's tissue (e.g. a score of 1,000 represents 1% of radiotracer was extravasated). The colors represent the extremity of each extravasation. These colors are subjectively produced by an internal Lucerno algorithm. This table also presents the number of patients' data that are used in this thesis and their corresponding extravasation scores.

1.2 An Alternative Approach

One issue with external monitoring devices, such as the LARA system, is cost. Although Lucerno claims that this product is "cost-effective" [22], their device is somewhat expensive, which is often a deterrent for imaging institutions considering extravasation detection. This work presents a novel streamlined workflow that can be introduced into the PET/CT imaging process with limited to no overhead cost that can detect extravasation in new patient scans with high validation accuracy. This workflow combines the popular TotalSegmentator tool with neural networks' unique prowess to classify data to provide medical institutions with a true cost-effective, extensible

Table 1.1 Lucerno intensity scale for extravasation [22,23]

Number of Patients	Score	Information
48	0- 700	0 – 0.7% EXTRAVASATED
50	700-5,000	0 .7 – 5% EXTRAVASATED
3	5,000-20,000	5 – 20% EXTRAVASATED
2	20,000-100,000	20 – 100% EXTRAVASATED

approach for detecting PET/CT extravasation. This proposed workflow can be broken into two layers: a pipeline for preprocessing the PET/CT data and two classification methodologies for classifying the metric report produced by the pipeline. This approach begins with the examination of two common formats that are used to store and interpret PET/CT slices, DICOM and NIfTI.

1.2.1 DICOM

The Digital Imaging and Communications in Medicine (DICOM) Standard is used for the storage, management, sharing, viewing, and transmission of medical information [24]. This standard eases the process of handling medical images such as MRIs, CT scans, and PET scans. DICOM encompasses a variety of application domains, including network image management, network image interpretation management, network print management, imaging procedure management, and off-line storage media management [25]. These allow researchers to interpret and use the data for building a streamlined process for detecting extravasation.

The patient information and scan details are stored in each slice's DICOM file header. This can be read into a Python environment using the package PyDicom [26]. Each DICOM header uses a tag-based system to store information in a hierarchical structure. Tags are composed of a group number and an element number, represented in hexadecimal format. These tags uniquely identify each data element within the DICOM file. The group number represents a broad category of information, while the element number specifies a particular attribute or value associated with that category. For example, the group number 0x0010 refers to patient information and the element number 0x0010 refers to the patient's name. This means that by accessing the DICOM header at (0x0010, 0x0010), the patient's name can be retrieved.

Snippets of relevant DICOM header information are captured below. Figure 1.5 depicts information about the type of image, when the slice was captured, and other relevant

information related to the DICOM data set as a whole. Figure 1.6 presents patient information, including the patient's age, weight, and name. This information has been anonymized for patient protection. It is important to note that the weight is in kilograms (kg), and the slice thickness is in millimeters (mm).

Figure 1.7 captures information about the individual slice. The instance number is a unique identifier assigned to each slice within a scan series. For example, an instance number of '1' means that this particular slice is the first in the series. The image orientation allows researchers to identify the axial, coronal, and sagittal planes. The pixel spacing is used to rescale PET or CT slices to their original scan field size. This is essential for resizing the CT mask to the size of the PET image. The other information is used for calculating the standard uptake values for each slice.

1.2.2 NIfTI

The next file format that was used throughout this project is the Neuroimaging Informatics Technology Initiative (NIfTI) file format [27]. This file format provides a similar, simpler version of DICOM's header information. This includes information such as crucial information for working with the voxel values. The primary reason using for this format is that it is a requirement for interacting with TotalSegmentator.

1.2.3 PET/CT Segmentation Pipeline

Traditionally, radiologists have analyzed DICOM data manually through contouring and manual SUV examination. This process is both time-consuming and requires a profound knowledge of human anatomy and physiology. The pipeline established in this work aims to alleviate this process in an automated fashion with two overarching objectives:

The first objective is the automatic creation of regions of interest (ROIs) in key organs that have been found to be statistically significant in detecting extravasation [28]. This is

(0008, 0008) Image Type	CS: ['DERIVED', 'AXIAL', 'CT_SOM5 SPI']
(0008, 0020) Study Date	DA: '20171107'
(0008, 0021) Series Date	DA: '20171107'
(0008, 0022) Acquisition Date	DA: '20171107'
(0008, 0023) Content Date	DA: '20171107'
(0008, 002a) Acquisition DateTime	DT: '20171107155411.527000'
(0008, 0030) Study Time	TM: '154610.176000'
(0008, 0031) Series Time	TM: '155511.271000'
(0008, 0032) Acquisition Time	TM: '155411.527000'
(0008, 0033) Content Time	TM: '155411.527000'
(0008, 0060) Modality	CS: 'CT'

Figure 1.5 A snippet of DICOM information extracted from one of the patients in the dataset

(0010, 0010) Patient's Name	PN: 'Doe_10^John'
(0010, 0030) Patient's Birth Date	DA: '19010101'
(0010, 0040) Patient's Sex	CS: 'F'
(0010, 1010) Patient's Age	AS: '099Y'
(0010, 1030) Patient's Weight	DS: '123.6990534625'
(0018, 0050) Slice Thickness	DS: '5.0'

Figure 1.6 DICOM information showing anonymized patient information that is used as features in neural network training exercises

(0020, 0011) Series Number	IS: '2'
(0020, 0012) Acquisition Number	IS: '2'
(0020, 0013) Instance Number	IS: '1'
(0020, 0032) Image Position (Patient)	DS: [-389.23828125, -581.23828125, 1858.5]
(0020, 0037) Image Orientation (Patient)	DS: [1, 0, 0, 0, 1, 0]
(0020, 1041) Slice Location	DS: '-1858.5'
(0028, 0030) Pixel Spacing	DS: [1.5234375, 1.5234375]
(0028, 1052) Rescale Intercept	DS: '-1024.0'
(0028, 1053) Rescale Slope	DS: '1.0'

Figure 1.7 DICOM information for a particular slice highlighting key attributes used for resizing

accomplished using various cross-sections through segmentation masks produced by TotalSegmentator, which creates cubes of voxels that are similar in size and location to the ones drawn by trained radiologists. This process extends the rather simplistic TotalSegmentator masks to specific areas where metrics can be extracted for extravasation detection. The second objective was to generate a set of features that includes summary statistics for the distribution of uptake values extracted from these selected ROIs. The following sections capture a high-level overview of image segmentation and Totalsegmentator to better understand this approach.

1.2.4 Image Segmentation

Image segmentation is a technique from image processing and computer vision that involves partitioning images or video frames into multiple segments or objects [29]. Image segmentation can be broken into two broad categories: traditional methods and deep learning-based approaches. Traditional methods range from thresholding and edge detection to more sophisticated algorithms emulating morphological watersheds [30]. Deep learning approaches include deep convolutional neural networks [31], fully convolutional neural networks [32], and U-Nets [33].

1.2.5 TotalSegmentator

Designing a universal image segmentation method that can carve out certain anatomical structures is a difficult task because anatomy varies from patient to patient. One could potentially use traditional image segmentation; however, a missing feature could cause the analysis pipeline to fail. TotalSegmentator [34] handles this problem using a U-Net [35] trained to identify 117 different anatomical structures in over 1,200 CT examinations.

1.2.6 Classification of PET/CT Data

The classification of DICOM data, such as PET/CT scans, can be accomplished using traditional machine learning and neural networks. This work shows how a fully connected neural network can be used for classifying the tabular data produced by the pipeline. An alternative approach is also shown, employing convolution neural networks to automatically extract features and classify each patient's scan on a slice-by-slice basis. These approaches produce a "red light or green light" for the corresponding patient scan. A green light indicates that it is unlikely the patient was extravasated and the radiologist can confidently proceed with diagnosis and therapy. A red light indicates extravasation was detected by the model.

1.3 Outline of Thesis

Chapter 2 describes the process used to create ROIs and export data that radiologists can examine to infer extravasation arose during the injection process. Chapter 3 shows how fully connected neural networks can be leveraged to reduce these features down to a single number that predicts the level of extravasation. Chapter 4 provides an alternative neural network approach, showing how convolution neural networks can extract these features automatically from the individual PET slices to detect extravasation. Chapter 5 concludes by revisiting the previous chapters, discussing the limitations of each approach and future work that will improve each technique.

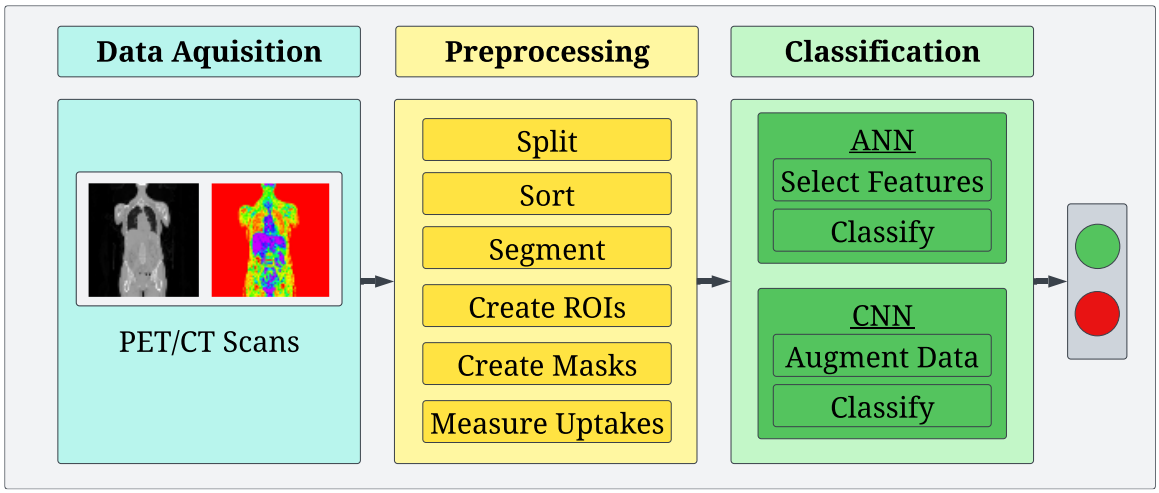


Figure 1.8 A compilation of the workflow containing an overview of each layer followed by a red/green light that indicates whether a patient is extravasated (red) or not extravasated (green)

CHAPTER 2:
A PREPROCESSING METHODOLOGY FOR AUTOMATIC FEATURE EXTRACTION

2.1 Overview

This chapter introduces an extensible pipeline designed for the automatic extraction of body weight-normalized standard uptake values (SUVbw) located in very specific regions within the patient [28]. This pipeline aims to alleviate the process of manually drawing regions of interest (ROIs), providing radiologists with a report of SUVbw values. This report can be quickly examined by these radiologists or combined with the methodologies in Chapters 3 and 4 to determine if a patient was potentially extravasated. Although this methodology is confined to detecting PET/CT radiopharmaceutical extravasation, this approach can be extended for other objectives in the future.

2.1.1 Data Assumptions

Assumptions are made about the data to ensure accurate results are extracted. These assumptions are presented below:

- Each patient's PET and CT scan must have an equal number of slices and slice thickness, allowing for a one-to-one mapping between each segmentation CT slice and its respective PET Data.
- The radiopharmaceutical used during PET imaging must be 18F-FDG.
- Corresponding PET and CT slices must be in the same location on the axial plane to further guarantee the mapping is in the correct anatomical location.
- Each slice must be in the industry standard DICOM format. This format is essential for all manipulation done to the data.
- Each slice's header file must include patient information shown in the DICOM section of Chapter 1.
- The patient's PET/CT scan must contain all of the anatomical structures of interest.

2.1.2 Error Checking

Error logging is used to increase the user's ability to troubleshoot issues. It is important to note that this is not an all-inclusive list; however, the following bullet points show the issues encountered during the lifetime of this thesis:

- slice increment inconsistency
- unequal number of PET and CT slices
- errors thrown from external packages
- failure to convert the DICOM data to a NiFTi .zip file

Upon each error, a comprehensive message is logged to a file specified by the user. For example, if a patient's directory has more PET slices than CT slices, the pipeline logs the message "Could not extract information from <Path to patient directory>. Reason: There was an uneven number of PET and CT slices". This does not stop the pipeline; instead, the pipeline continues to the next patient in the parent directory to ensure a hands-off continuous process for running patients in bulk.

2.2 Data Preparation

The following content captures the methods used to prepare each patient's DICOM data for segmentation and SUVbw extraction. This section covers the separation of PET and CT data as well as the resizing of the CT data to the size of the PET data. An overview of this process can be found in Figure 2.1.

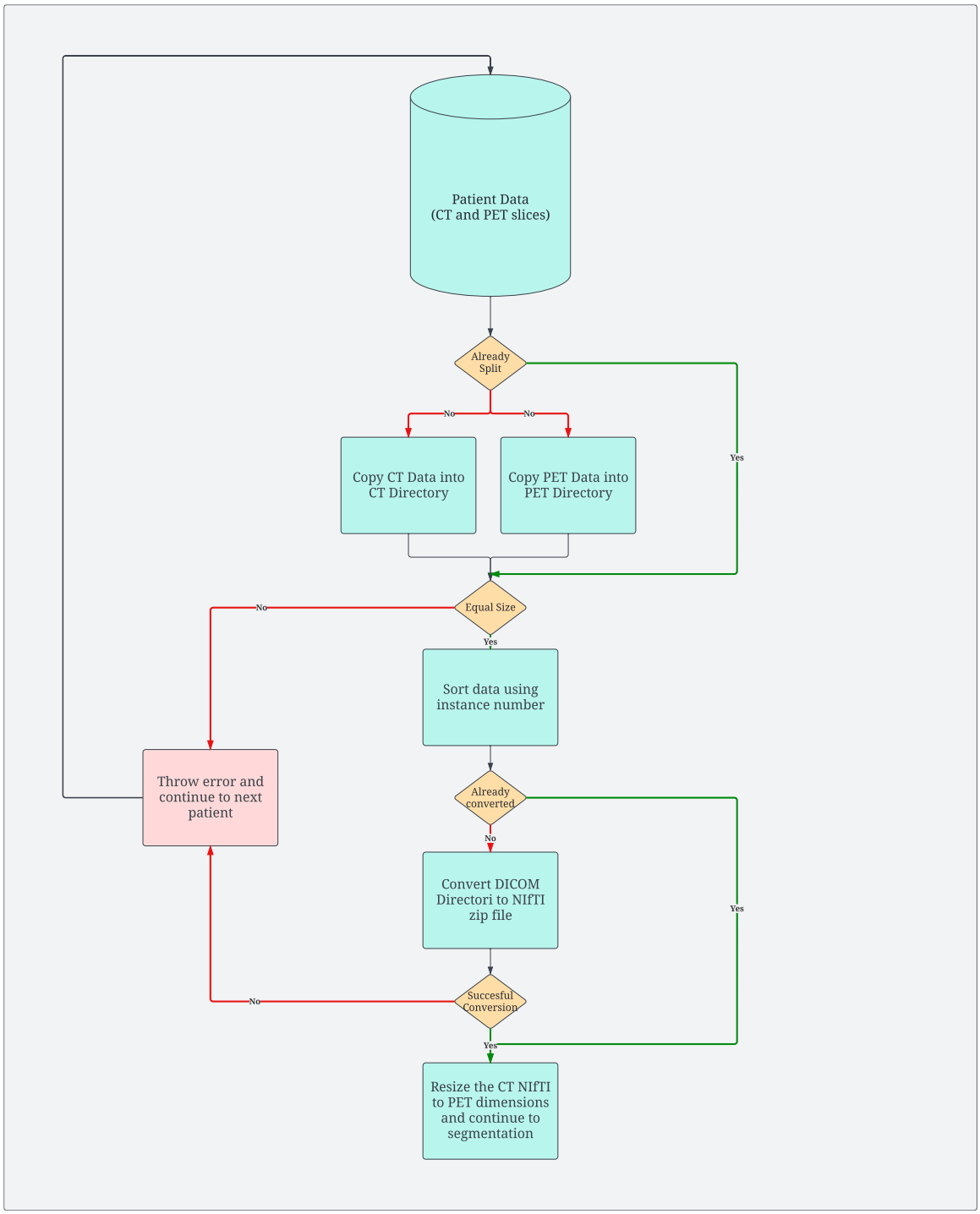


Figure 2.1 A workflow chart capturing the various decisions made when preparing the DICOM data for segmentation

2.2.1 Separating and Resizing the Data

The data preparation process begins by separating the CT and PET slices into two distinct directories using the image type defined in the DICOM header. Afterward, the size of both directories is checked to confirm they have an equal number of slices. The directories are then sorted using each slice's instance number. This process keeps each CT slice aligned with its corresponding PET slice on the axial plane. All errors during this process are logged accordingly.

The data is resized using affine transforms. This is essential because the underlying patient data in both the PET and CT slices must be the same size to ensure that the CT masks overlay correctly on the anatomical locations of the corresponding PET slices. This process is done using the following steps:

1. **The dimensions of the pixel array for both the CT and PET data are recorded.** The z-axis (coronal plane) is ignored because it is guaranteed that there is an equal number of CT and PET slices, and each slice has the same thickness.
2. **The original scan field sizes for the CT and the PET are calculated using the defined pixel spacing.** This process is illustrated in Figure 2.2 and Figure 2.3.
3. **A ratio is established between the dimensions of the CT's original scan field and the PET's original scan field.**
4. **The CT data is embedded into a canvas of equal geometry to the PET data and downsampled using the ratio such that the patient's body is the same size in both the PET and CT.** It is important to note that the CT data is resized instead of modifying the PET data to avoid compromising the physiological data within the PET data. Gaussian filtering is applied to avoid aliasing artifacts [36].
5. **Cropping is performed on the larger data to ensure equal dimensions**. This is shown in Figure 2.4.

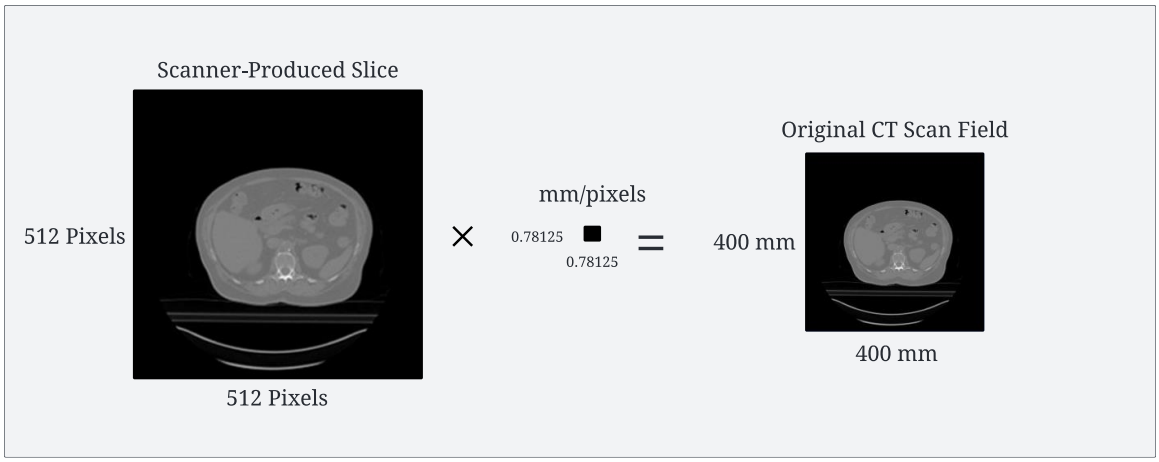


Figure 2.2 A CT slice being scaled back to the size of the original scan field using the pixel spacing provided in the DICOM header

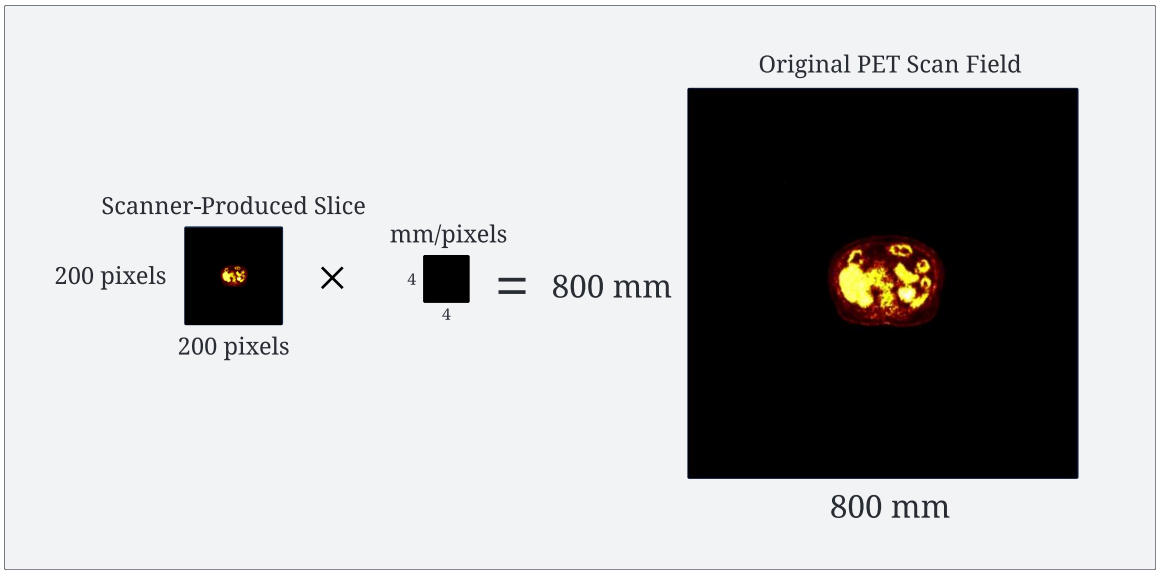


Figure 2.3 A PET slice being scaled back to the size of the original scan field using the pixel spacing provided in the DICOM header.

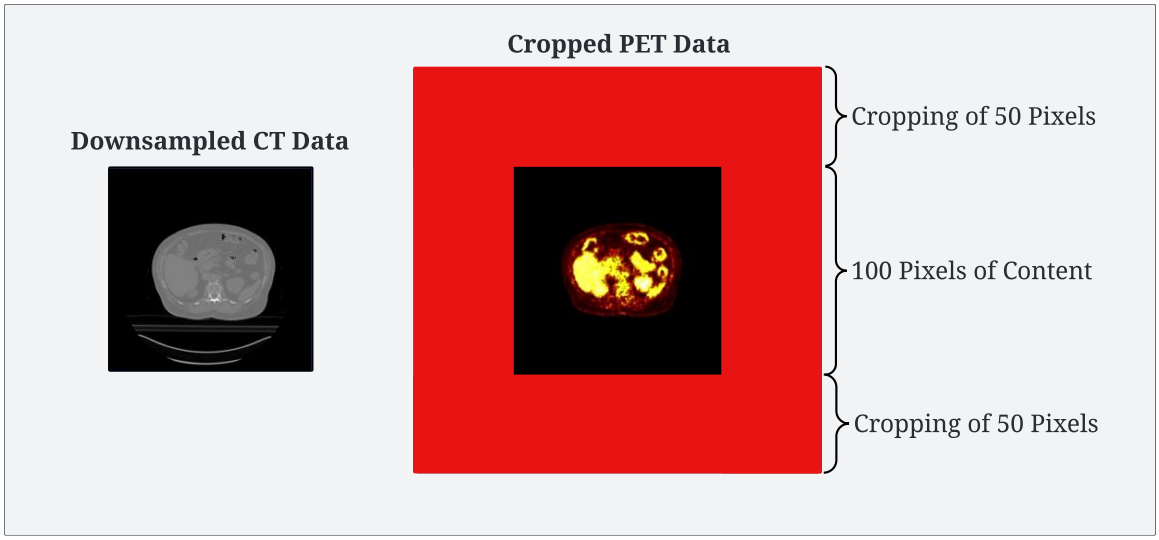


Figure 2.4 A PET slice cropped to have equal dimensions as the corresponding downsampled CT slice.

2.3 Data Segmentation & SUVbw Extraction

2.3.1 Segmenting the Data

Each CT scan is segmented using TotalSegmentator to create binary masks of the aorta, brain, liver, L1 vertebrae, T12 vertebrae, and trachea. A subset of these masks is shown in Figure 2.5. From these segmentations, three ROIs are created using these anatomical locations as guidance. These include an ROI within the upper-right lobe of the liver, a portion of the aorta, and the cerebellum. These masks are created by passing a list of the ROIs to the TotalSegmentator function:

```
totalsegmentator(<Output Directory>, fast=False, roi_subset=<List of anatomical areas>)
```

2.3.2 Liver ROI

After collecting the segmentations produced by TotalSegmentator, a boundary is created on the CT scan at the superior edge of the T12 vertebrae and the inferior edge of the L1 vertebrae. From here, a cross-section is created approximately 3/4 of the way from the superior boundary to the inferior boundary. For example, if the superior boundary is the 112^{th} axial slice and the inferior boundary is the 100^{th} axial slice, then the cross-section is created at the 109^{th} axial slice. The cross-section is expanded 5 slices in the superior and inferior direction creating an ROI of 10 axial slices. These axial slices are limited to 10 coronal slices and 10 sagittal slices, creating a cube. A detailed image of the boundaries drawn on the CT data can be found in Figure 2.6. An example of an ROI produced using this strategy is shown in Figure 2.7.

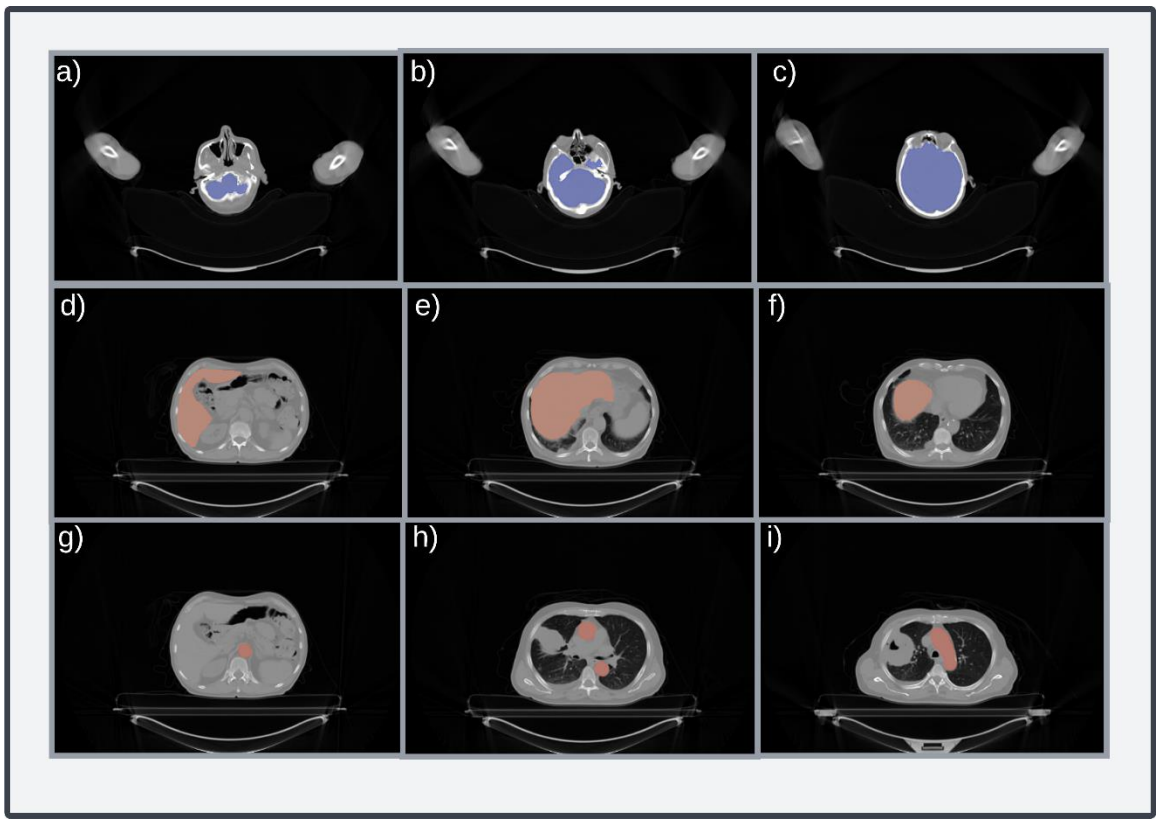


Figure 2.5 Initial masks produced by TotalSegmentator: (a-c) brain, (d-f) liver, (g-i) aorta.

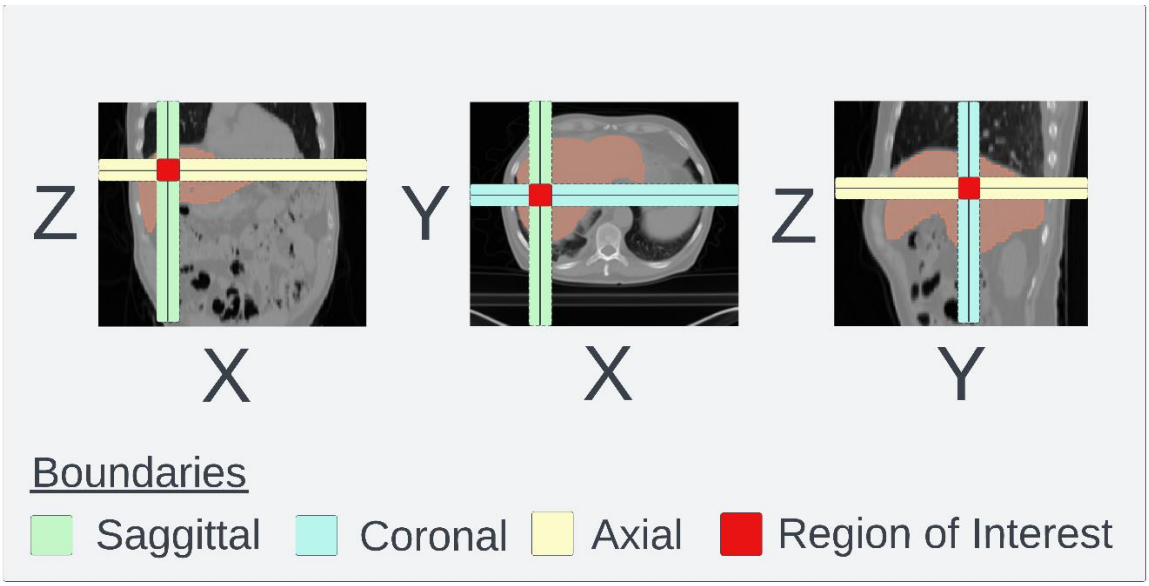


Figure 2.6 An illustration of how a cube is created to create the ROI within the liver. The faint line represents the initial cross-section and the expansion represents the additional slices.

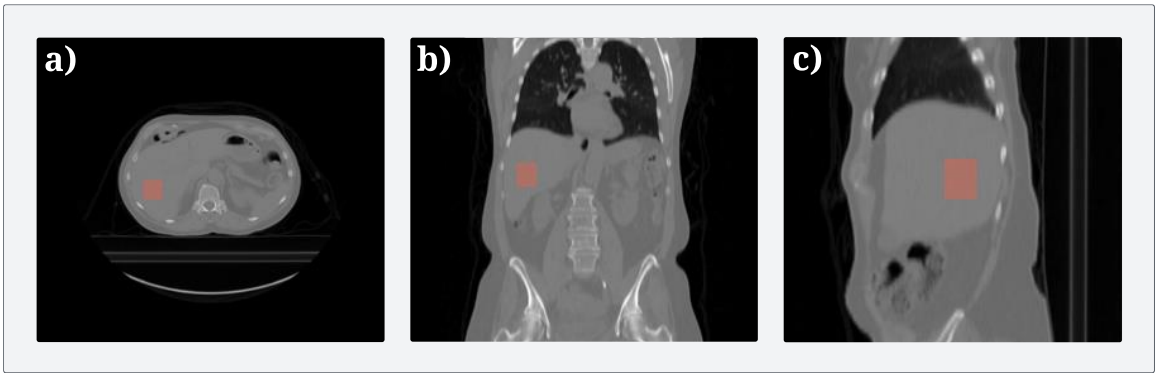


Figure 2.7. An ROI produced by the pipeline shown from different perspectives: (a) axial, (b) coronal, (c) sagittal

2.3.3 Aorta ROI

The aorta was reduced to a small section distal and posterior to the carina. This ROI was created by using TotalSegmentator to segment out the trachea mask, which ended 5 slices distal to the carina. This was an optimal location for the inferior boundary of the aorta ROI. The superior boundary was created 5 slices above this inferior boundary. This segmentation is shown in Figure 2.8.

2.3.4 Cerebellum ROI

The cerebellum posed the greatest segmentation challenge. It is common for radiologists to draw ROIs within the 4th ventricle when manually extracting SUVbw values from patients. Due to the lack of anatomical regions that can guide image processing algorithms to this limited space, a more practical approach was taken. First, the inferior coronal boundary of the brain was used as the cerebellum's inferior boundary. This cross-section was created 35% of the way between the brain mask's inferior and superior coronal boundaries, which defined the cerebellum's superior coronal boundary. The cerebellum's inferior axial boundary was defined at the brain's inferior boundary and the cerebellum's superior axial boundary at 10 slices superior to the inferior boundary. These values were selected through trial and error, which produced the ROI shown in Figure 2.9.

2.3.5 Calculating SUVbw Metrics

The header information from each slice's DICOM file was used to convert each slice's voxel values from Bq/ml to SUVbw. This was done at the voxel level using the following formula:

$$SUV_{bw} = A_c/D \cdot W \cdot 2^{-(t_s - t_m)/T_{\frac{1}{2}}} \quad (1)$$

where,

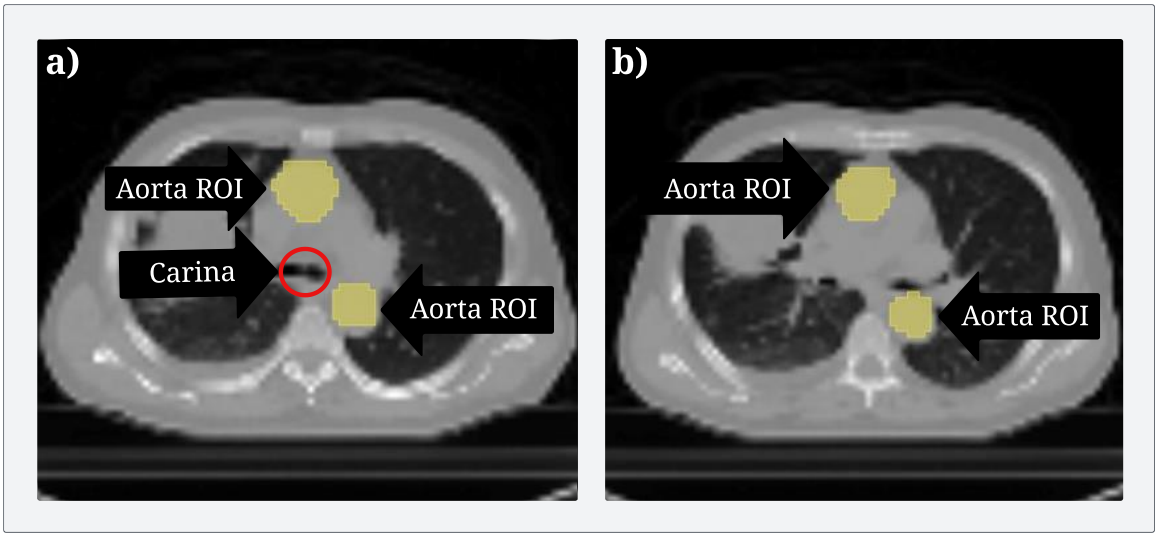


Figure 2.8 The two boundaries of the aorta ROI: (a) Superior boundary with the carina highlighted, (b) Inferior boundary

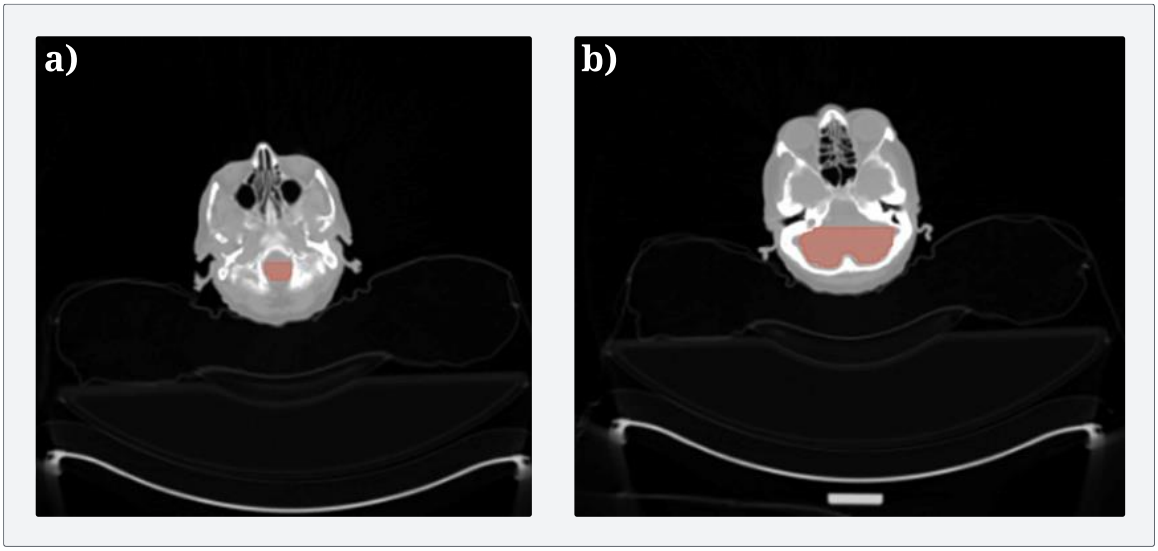


Figure 2.9 The two boundaries of the cerebellum ROI: (a) Superior, (b) Inferior

A_c = Activity Concentration (Bq/ml)

D = Injected Dose (Bq)

W = Weight of patient (g)

t_s = The time when the patient was injected (s)

t_m = The time when an individual slice was captured (s)

$T_{\frac{1}{2}}$ = Half-life of radionuclide (s)

After the voxels are successfully converted, key metrics are extracted from each of the areas of interest, namely the mean, median, and max SUVbw for each area and the ratio between these metrics. The reciprocals were eliminated resulting in 171 features. The alpha-trimmed mean, variance, skewness, and kurtosis were also calculated for each area, generating 24 additional features. Lastly, the patient's weight and age were added to the feature list, for a total of 197 features. Algorithm 1 below provides pseudocode for how this process was done programmatically.

The pipeline outputs each of the 197 features generated by Algorithm 1 to a CSV file. A header is appended to the top of the CSV file to help identify each column and additional spacing is introduced between groups of columns for readability. Each mask is saved for the areas of interest. A summarization of this process and the final output is shown in Figure 2.10.

2.3.6 Extendibility

The selected tools and design patterns were used to make this process agnostic and extendible. This means that future researchers can add additional areas that might be useful for detecting extravasation or other use cases simply by adding areas to the array of anatomical locations that is passed to TotalSegmentator. The created NIfTI mask(s) can then be fed into the same SUVbw calculation method used for each of the masks shown above. Another extension of this pipeline could be the addition of new metrics to Algorithm 1, which would generate more features for identifying extravasation.

ALGORITHM 1: CALCULATE SUV METRICS NORMALIZED TO BODY WEIGHT

Input: A binary mask for the area of interest (e.g. liver), PET Data
Output: SUVbw mean, median, max, skewness, variance, kurtosis, alpha-trimmed mean

```
first_slice, last_slice ← initialize variables to first and last slice of mask
suv_bw_list ← initialize variable to empty list
for slice = first_slice to last_slice do
    dicom_file ← read DICOM header file for the given slice
    patient_weight ← initialize patient weight
    measured_time ← initialize patient measured time
    series_time ← initial series time
    injected_dose ← initialized injected dose
    half_life ← initialized half life
    time_difference = series time - measured time in seconds
    exponentiationOperation = 2^((−(time_difference) / half_life))
    for pixel = first_pixel to last_pixel do
        rescaled_pixel_value = (pixel · rescale_slope) + rescale_intercept
        act_tracer_activity = injected_dose · exponentiationOperation
        suv_bw = (pixel_value · patient_weight) / act_tracer_activity
        append(suv_bw_list, suv_bw)

return mean, median, alpha_trimmed_mean, variance, skewness, kurtosis, max
```

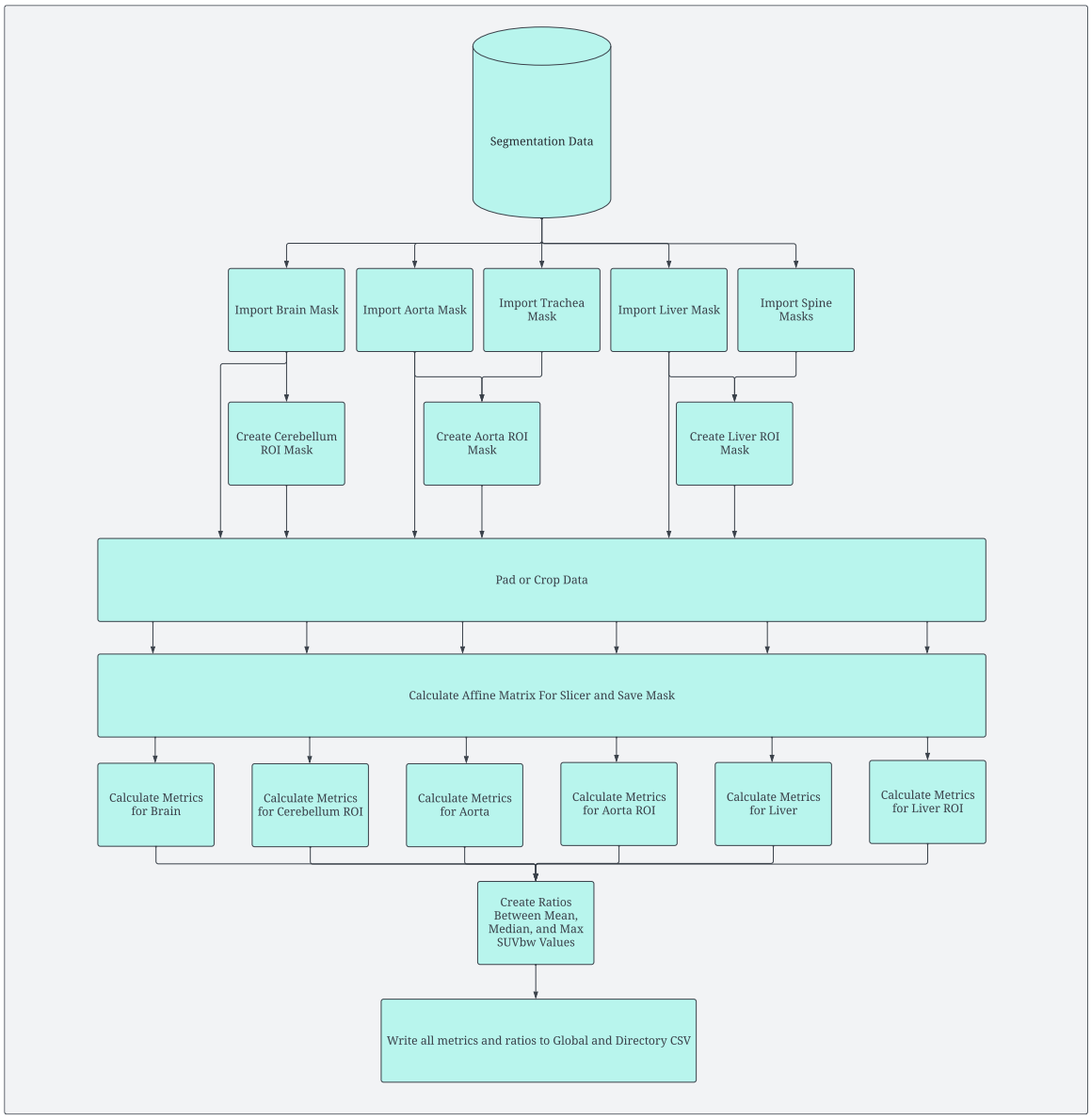


Figure 2.10 An overview of how the TotalSegmentator masks are used to create ROIs and calculate each of the features produced by the pipeline

2.4 Results

The performance of this pipeline was measured by calculating the percent difference between a selection of the outputs from the pipeline and SUVbw measurements extracted by trained radiologists. The percent difference formula is written as follows:

$$\text{Percent Difference} = \frac{|M_1 - M_2|}{\frac{M_1 + M_2}{2}} \cdot 100 \qquad (2)$$

where,

M_1= Pipeline-extracted value

M_2 = Radiologist-extracted value

This showed that the max reduced liver measurements were about **13.81%** different on average, the max cerebellum measurements were about **11.14%** different on average, and the max reduced aorta measurements were about **19.59%** different on average. These comparisons are captured in Table 2.1-2.3 below. The comparisons are summarized using Q-Q plots in Figure 2.12.

A portion of the disparities in measurements lies in the pipelines' inability to circumvent blood pools and other anomalies that radiologists typically avoid when drawing ROIs within areas of interest. Another probable reason for these disparities is the size differences between ROIs produced by the pipeline and radiologist-drawn ROIs.

Table 2.1 Percent differences between the measurements the pipeline produced for the reduced liver ROI and the measurements extracted by a trained radiologist

Patient	Radiologist	Pipeline	Percent Difference (%)
P1	3.4	3.5	2.90
P2	5.2	4.4	16.67
P3	3.3	3.4	2.99
P4	3.0	3.5	15.38
P5	4.0	4	0.00
P6	3.5	3.4	2.90
P7	3.9	4.2	7.41
P8	3.8	3.5	8.22
P9	2.2	2.5	12.77
P10	2.3	2.1	9.09
P11	3.2	3.4	6.06
P12	3.7	3.5	5.56
P13	4.0	3.1	25.35
P14	4.5	2.4	60.87
P15	4.7	4.9	4.17
P16	5.4	4.8	11.76
P17	2.5	2.5	0.00
P18	2.9	3.3	12.90
P19	3.7	3.6	2.74
P20	3.6	3	18.18
P21	4.8	4.2	13.33
P22	5.0	4.5	10.53
P23	3.4	5.3	43.68
P24	3.1	2.3	29.63
P25	3.5	2.8	22.22
Total			**13.81%**

Table 2.2 Percent differences between the measurements the pipeline produced for the reduced cerebellum ROI and the measurements extracted by a trained radiologist

Patient	Radiologist	Pipeline	Percent Difference (%)
P1	11.8	13.5	13.44%
P2	13.1	12.1	7.94%
P3	11.6	11.9	2.55%
P4	13.3	15.2	13.33%
P5	10.3	11.7	12.73%
P6	12.1	11.6	4.22%
P7	11.0	11.3	2.69%
P8	7.9	7.9	0.00%
P9	9.4	10.4	10.10%
P10	12.4	13.4	7.75%
P11	10.2	9.9	2.99%
P12	9.0	7.9	13.02%
P13	13.7	11.7	15.75%
P14	11.1	11.7	5.26%
P15	7.9	10.1	24.44%
P16	10.8	11.8	8.85%
P17	10.8	11.3	4.52%
P18	9.4	9.8	4.17%
P19	12.0	16.7	32.75%
P20	9.1	12.5	31.48%
P21	15.8	14.4	9.27%
P22	18.2	16.7	8.60%
P23	8.4	7.8	7.41%
P24	9.5	7.6	22.22%
P25	11.5	10.1	12.96%
Total			**11.14%**

Table 2.3 Percent differences between the measurements the pipeline produced for the reduced aorta ROI and the measurements extracted by a trained radiologist

Patient	Radiologist	Pipeline	Percent Difference (%)
P1	2.2	2.9	27.45%
P2	3.2	4.9	41.98%
P3	2.2	2.9	27.45%
P4	2.5	2.9	14.81%
P5	3.2	3.3	3.08%
P6	2.6	3.3	23.73%
P7	3.4	3.7	8.45%
P8	3.6	3.9	22.86%
P9	1.8	2.6	36.36%
P10	1.6	2.4	40%
P11	2.7	2.8	3.64%
P12	2.6	3.3	23.73%
P13	3.0	3.9	26.09%
P14	3.3	3.5	5.88%
P15	3.3	3.8	14.08%
P16	3.9	4	2.53%
P17	2.3	2.6	12.24%
P18	2.4	2.9	18.87%
P19	3.3	3.1	6.25%
P20	2.8	3.1	10.17%
P21	3.6	3.8	5.41%
P22	4.7	4.5	4.35%
P23	2.6	3.6	32.26%
P24	1.7	2.5	38.10%
P25	2.2	3.3	40.00%
Total			**19.59%**

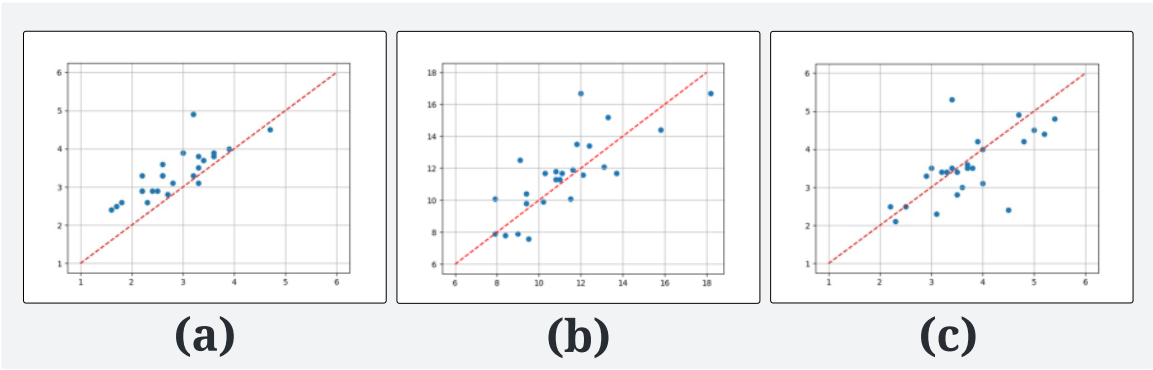


Figure 2.12 Q-Q plots showing the difference between the pipeline-extracted and radiologist-extracted SUVbw measurements: (a) reduced aorta ROI, (b) cerebellum ROI, (c) reduced liver ROI

CHAPTER 3:
APPLICATION OF A MULTI-LAYER ARTIFICIAL NEURAL NETWORK

3.1 Overview

Observing SUVbw metrics and other patient-specific features allows researchers to articulate whether a patient has been exposed to extravasation during the PET/CT injection process. For example, certain ratios of uptake values in the identified ROIs, coupled with low patient body weight, significantly enhance the likelihood of extravasation. Although it is possible to configure decision trees and employ other traditional methods to perform classification, this chapter shows how this process can be offloaded to fully connected neural networks.

3.2 Neural Network Fundamentals

3.2.1 Fully Connected Artificial Neural Networks

A fully connected artificial neural network, often referred to as a fully connected feedforward neural network or multilayered perceptron, is designed to mimic the human brain. The architecture scales the idea of firing neurons to large-scale networks that can classify data. Each neuron produces a value or "fires" with a magnitude based on the input activity, weight of each connection, bias, and activation function. The output for each neuron can be computed using the following formula:

$$\text{Output} = f(b + \sum_{i=1}^{n} x_i w_i) \quad (3)$$

where,

f = Activation function

b = bias

x_i = input activity

w_i = weight

A fully connected neural network can be broken down into three categories: the input layer, the hidden layer(s), and the output layer. The input layer is comprised of neurons tailored to ingest the data where each element of a sample corresponds to an individual neuron. For example, when using a vector of features produced by the pipeline, each feature is fed into one of the neurons in the input layer. A depiction of a row from the feature CSV file being fed into the input layer of an ANN can be seen in Figure 3.1.

The hidden layers of an ANN are user-configured [37]. The number of layers and layer size must be chosen to create an architecture that classifies the training data well without overfitting. Some authors suggest trial and error [38], while others have attempted to develop optimal strategies [39-41]; however, there is no existing, concrete methodology for selecting these layer hyperparameters.

An activation function is used to determine the output of each neuron. Its main purpose is to introduce non-linearity to model features that do not have a linear relationship to their inputs. For the ANN architecture, ReLU is used for the hidden layers [42]. ReLU applies a non-saturating nonlinearity to the output of each neuron which has been proven to increase training times by 25% when compared to a saturating nonlinear function like tanh(x) [43].

The output layer is task-specific. In tasks such as predicting the Lucerno score for a patient or classifying a patient's features as either extravasated or not extravasated, the output layer is a single neuron that uses a linear activation function or sigmoid respectively. If the task is to classify the features as one of the four classes that Lucerno uses to label the extravasation extremity, shown in Table 1.1 of Chapter 1, the output layer will be four neurons. For this approach, each neuron produces a confidence value for its underlying class. The three output layer configurations are shown in Figure 3.3 and each of the discussed activation functions is shown in Figure 3.4

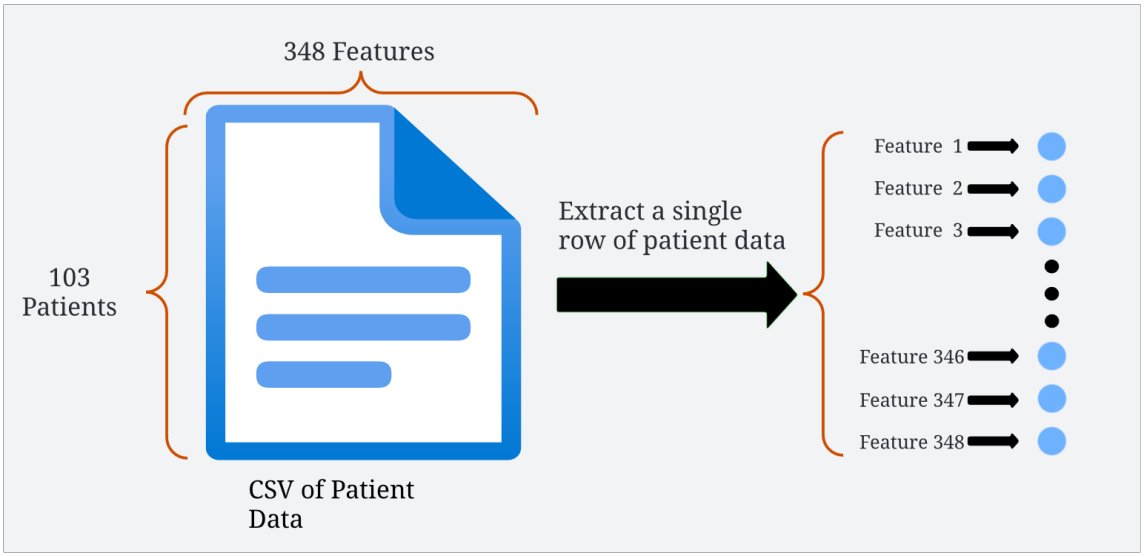


Figure 3.1 CT data fed into the neural network where each pixel is an individual neuron

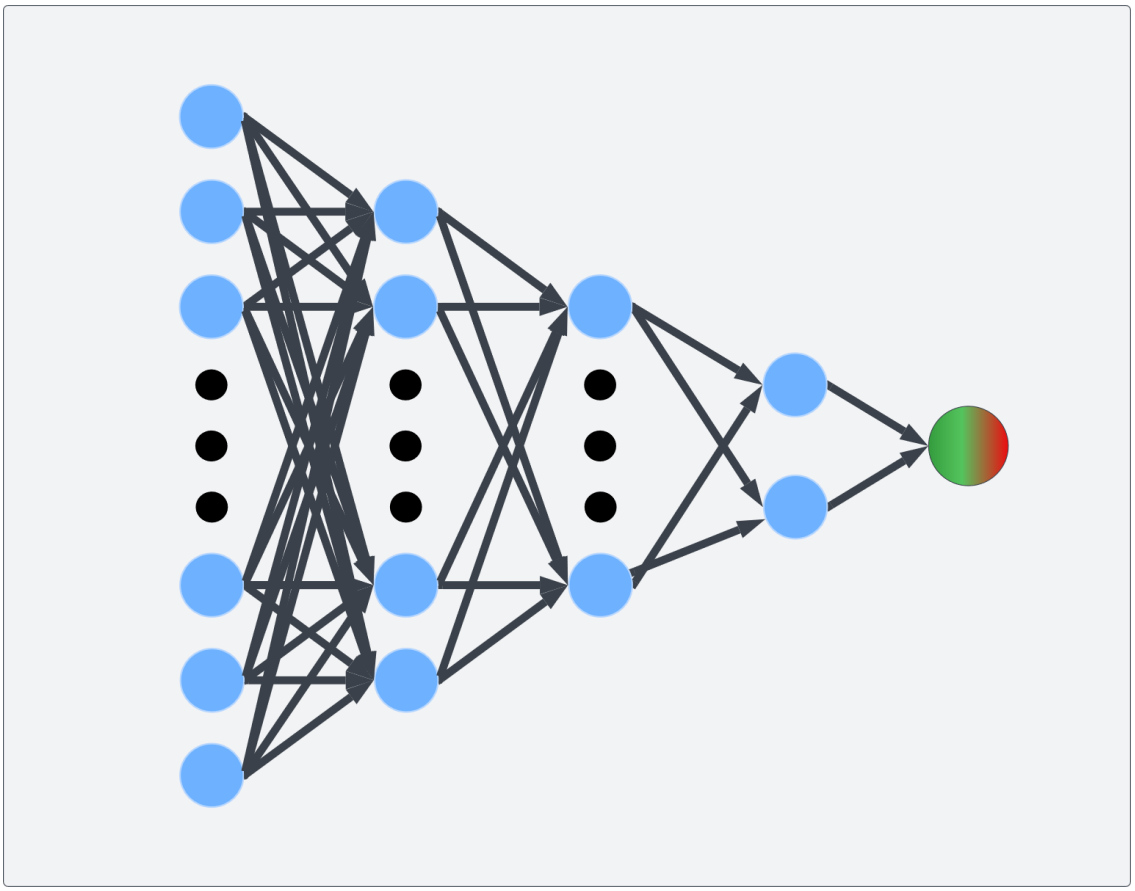


Figure 3.2 Final layers of a fully connected neural network where the final neuron is a binary prediction



Figure 3.3 Outputs layers for specific tasks: (a) binary output, (b) continuous output, (c) categorical output

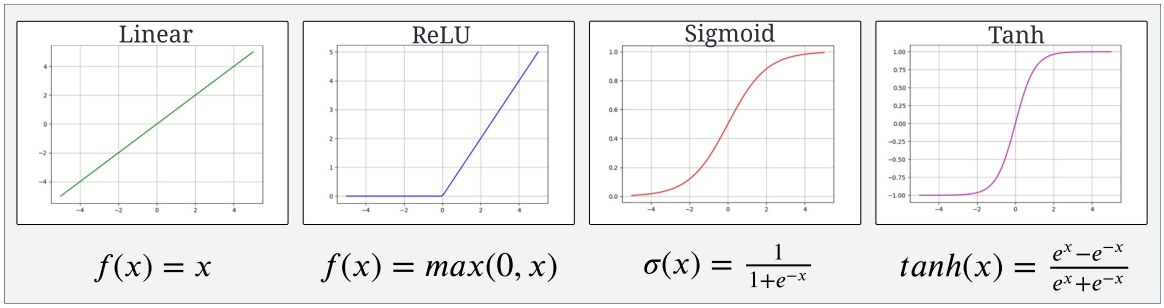


Figure 3.4 Commonly used activation functions in neural networks

3.2.2 Loss Functions

When training neural networks, the goal is to minimize or maximize the difference between the produced output and the expected output. This difference is measured using an objective function. When minimizing the objective function, this term has many interchangeable names including error function, cost function, and loss function [44]. For the purposes of this research, the term "loss function" is used to define the employed objective functions because the goal is to minimize the loss during training.

Similar to the activation function, the selection of the loss function is task-specific. For regression tasks, such as trying to reduce the difference between the Lucerno score predicted by the model and the ground truth Lucerno score, a common loss function is mean squared error (MSE). MSE measures the average square differences between the predicted value and the actual values. This loss function is calculated as:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \qquad (4)$$

where,

MSE = Mean Squared Error

Y_i = Ground Truth Value

$\hat{Y}_i$ = Predicted Value

n = Number of samples

When dealing with datasets containing outliers, it is advantageous to use mean absolute error (MAE). This loss function removes the exponential component from MSE, therefore removing the additional weight added to larger errors. MAE is defined using the following formula:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|\hat{Y}_i - Y_i| \qquad (5)$$

where,

MAE = Mean Absolute Error

Y_i = Ground Truth Value

$\hat{Y}_i$ = Predicted Value

n = Number of data points

For classification, a loss function based on cross-entropy is often used. Binary cross-entropy, which is common for two-class problems, is defined as:

$$\text{Categorical Cross-Entropy} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{C} Y_{ij} \cdot \log\left(\hat{Y}_{ij}\right) \quad (6)$$

where,

Y_{ij} = Ground Truth Value for $Class_i$ in $Sample_j$

$\hat{Y}_{ij}$ = Predicted Value for $Class_i$ in $Sample_j$

N = Number of training samples

C = Number of classes

Categorial cross-entropy, which is used for a multi-class problem, is defined as:

$$\text{Binary Cross-Entropy} = -\frac{1}{N}\sum_{i=1}^{N}\left(Y_i \cdot \log\left(\hat{Y}_i\right) + (1 - Y_i) \cdot \log\left(1 - \hat{Y}_i\right)\right) \quad (7)$$

where,

Y_i = Ground Truth Value

$\hat{Y}_i$ = Predicted Value

N = Number of training samples

This research used binary cross-entropy, but it is important to note that one could alter this approach to predict the severity of the extravasation using Lucerno's defined labels shown in Table 1.1. Lastly, it should be noted that a decrease in the training loss does not affect the accuracy of the model [41]. This means that even if the loss function is minimized over several epochs, this might not increase accuracy.

3.2.3 Backpropagation and Gradient Descent

Another key element of neural networks is the well-known algorithm backpropagation [45]. This algorithm received its name because it is used to "back propagate" the error produced by the objective function to repeatedly update each of the weights based on their individual contributions to the error. This process is done using the chain rule from Calculus, which allows the error to be expressed in terms of gradients that represent each previous neuron's contribution. These gradients are then used to update a neuron's weight and bias in an optimization algorithm called gradient descent [46].

3.2.4 Hyperparameters

Designing performant neural networks that can accurately predict extravasation requires a meticulous selection of hyperparameters. Hyperparameters are values set by the user before beginning the training process. As previously mentioned, layer sizes and the number of layers are examples of hyperparameters. Other examples include learning rate, dropout rate, number of nodes per hidden layer, etc. These hyperparameters act as knobs that the user can tune to achieve the best results possible. For example, if one is attempting to eliminate overfitting, it might be practical to increase the dropout [47]. This parameter is especially useful for limited datasets because this process of randomly dropping nodes and their connections significantly reduces the neural network from overfitting on noisy samples.

3.2.5 Performance Metrics

There are several ways to measure the performance of the models. The first and most common is accuracy. Accuracy is the measurement of how many samples the model predicted correctly over the total number of predictions. For example, if the model is trained to predict whether a patient is extravasated or not extravasated, the accuracy of a model would be written as:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

where,

TP = Model predicts extravasation correctly.

TN = Model predicts no extravasation correctly.

FP = Model predicts extravasation incorrectly.

FN = Model predicts no extravasation incorrectly.

Identifying whether the model was good at identifying the extravasated vs non-extravasated patients can also provide valuable insight. If the radiologist wants the model to correctly identify true negatives, it is important to use the model that produces the highest specificity. A higher specificity indicates that the model has fewer false positives and is better at distinguishing non-extravasation cases. Specificity can be written as:

$$\text{Specificity} = \frac{TN}{TN + FP} \quad (9)$$

If the radiologists' goal is to identify the true positives, a useful metric to use is recall or sensitivity. This formula is shown in Equation 10. In this context, recall measures how well the model predicted that a person exposed to extravasation was extravasated. Another metric, shown in Equation 11, is precision. This metric measures how well the model predicts true positives with respect to the total number of extravasated patients in the dataset. A high precision value means that the model is making accurate extravasation predictions.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (10)$$

$$\text{Precision} = \frac{TP}{TP + FP} \quad (11)$$

Lastly, F1 score weights precision and recall equally into the following formula:

$$\text{F1 Score} = \frac{\text{TP}}{\text{TP} + \frac{1}{2}(\text{FP} + \text{FN})} \quad (12)$$

3.3 Methods

This section provides readers with an overview of the various approaches used for detecting radiopharmaceutical extravasation. This includes different data partitions tailored to different use cases, the architecture of the chosen model, and a K-Fold validation approach for addressing coincidental data selection.

3.3.1 Constructing a Dataset

103 patients' features were collected using the pipeline in Chapter 2. Each patient's features were labeled using their respective Lucerno scores. For calculating accuracy and other performance metrics, a modest threshold of 700 was used, where any patient with a Lucerno score above this threshold was labeled as extravasated and any patient below this threshold was labeled non-extravasated. This resulted in a dataset with more extravasated patients than non-extravasated patients. This imbalance was fixed by reducing the dataset to 96 patients, 48 extravasated and 48 not extravasated.

3.3.2 Sequential Selection Classification

For the first approach, the data was split using an ~83% validation split. This means that the training data was composed of the first 40 patients with a Lucerno score below the established threshold and the first 40 patients with a score above the threshold. This resulted in a total of 80 patients' features for training. The next 16 patients with a score meeting this criterion were selected as validation data. Although the actual data was not as uniform, a simplified version of this selection process is depicted above in Figure 3.5.

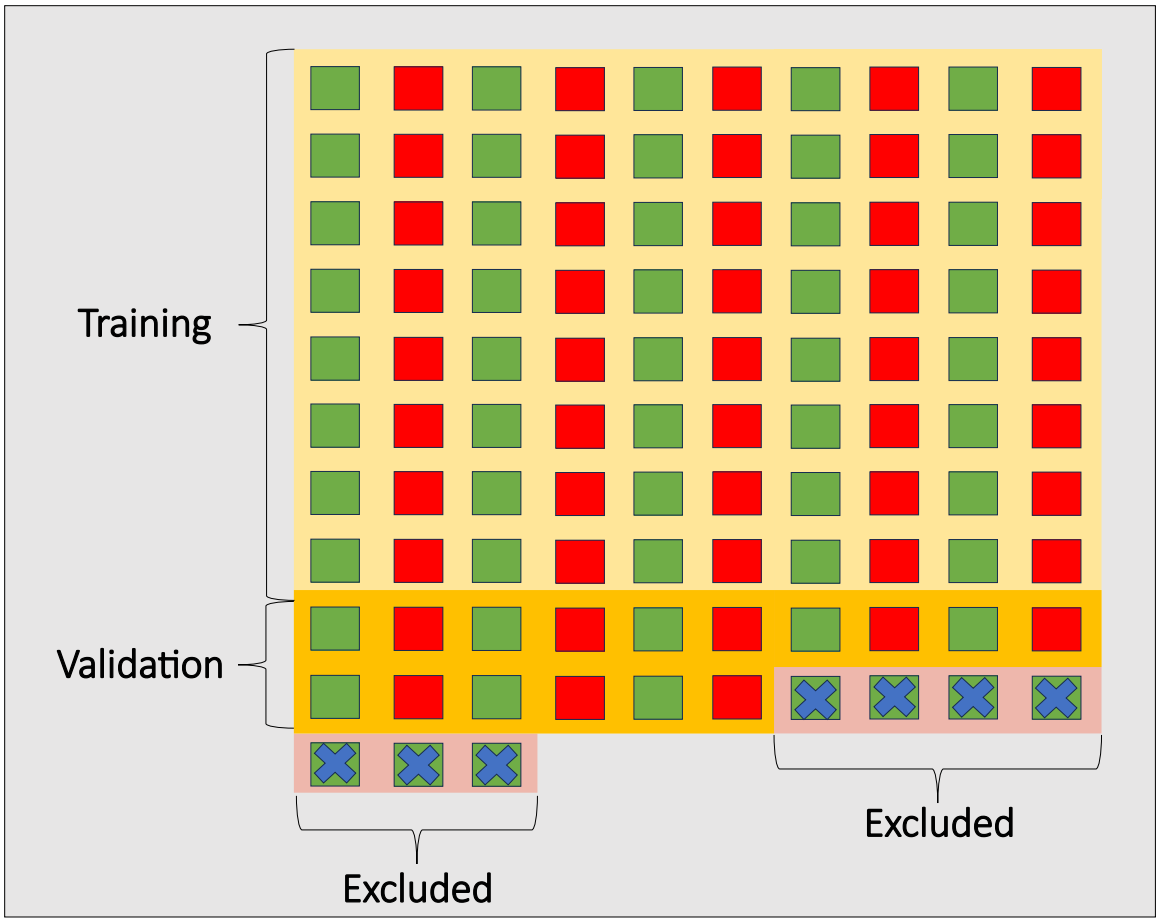


Figure 3.5 A depiction of the process used to create a balanced dataset using a Lucerno score of 700 as the recommended threshold.

3.3.3 Classifying Highest Extravasation Cases

For the next approach, the model's performance was measured based on its ability to classify each of the highest extravasation cases in the data while being trained solely on the remaining extravasation cases. This approach stemmed from the idea that a leakage of only 0.7% of the radiopharmaceutical (a Lucerno score of 700) is not a significant enough extravasation to warrant any post-imaging actions or concern. This process is also more applicable for medical institutions that choose to train their models from scratch, considering they will likely have a relatively low number of extreme extravasation cases.

Similar to the sequential selection, the first 43 patients with a score Lucerno score over and under the threshold were captured for training. The validation dataset included the 4 highest extravasated patients, 1 additional extravasated, and 5 non-extravasated patients for validation purposes. This resulted in a validation split of around 90% for training purposes.

3.3.4 K-Fold Cross-Validation

The possibility of selection bias was combatted by further validating the results using K-Fold cross-validation. This process generated results that were independent of the validation data selected. In this exercise 'K' was 10, meaning the dataset was traversed 10 times from different starting points, selecting a unique set of 16 validation patients each time. This approach also shuffled through previously excluded patients and included them in both the validation data and training data for a portion of the folds performed.

3.3.5 Feature Selection

Various combinations of features were selected for training and testing. These features are shown below:

1. The mean, median, and max (MMM) SUVbw values for the liver, brain, aorta, ROI within the liver, cerebellum, and the reduced aorta segmentation

2. The ratio between the mean, median, and max SUVbw values
3. The alpha-trimmed mean (ATM) with an alpha of 10%, variance, skewness, and kurtosis for the distribution of SUVbw values in each area of interest
4. The weight and age of the patient

Initially, the most straightforward approach was taken, using all 197 features generated by the pipeline, discussed in Chapter 2, into the neural network. It is important to note that too many features can often lead to a lack of generalization capability [53]. With this consideration, principal component analysis was performed to assess the level of correlation between the features. With this analysis, it was found that the entire feature set could be reduced to 26 principal components while still maintaining 99% of the cumulative variance. A plot depicting the relationship between the cumulative variance and the number of principal components is shown in Figure 3.6. This suggested that the data was highly correlated and sets of fewer features could be used with minimal loss of feature information.

In order to address the high correlation present in the data, smaller subsets of features were strategically selected through the logical consideration of which features would have the highest relationship to the output. First, only the mean, median, and maximum (MMM) SUVbw values, weight, and age (1 and 4) were used. Next, only the ATM, kurtosis, variance, skewness, kurtosis, weight, and age (3 and 4) were used. Then, the MMM statistics combined with the ratios of these statistics (1, 2, and 4) were used. Lastly, a test was performed to see if age was a contributing factor to the model's performance, it was excluded from the feature list (1, 2, 3, and 4 excluding age). Each of these feature sets were used in training with the same architecture, number of epochs, and hyperparameters to ensure consistency. The use of smaller feature sets and the architecture decisions discussed in 3.3.6 resulted in an improvement in training performance, supporting the idea that the model might be falling victim to "the curse of dimensionality." Future work should explore pair-wise correlation analysis to select an optimal feature set for training.

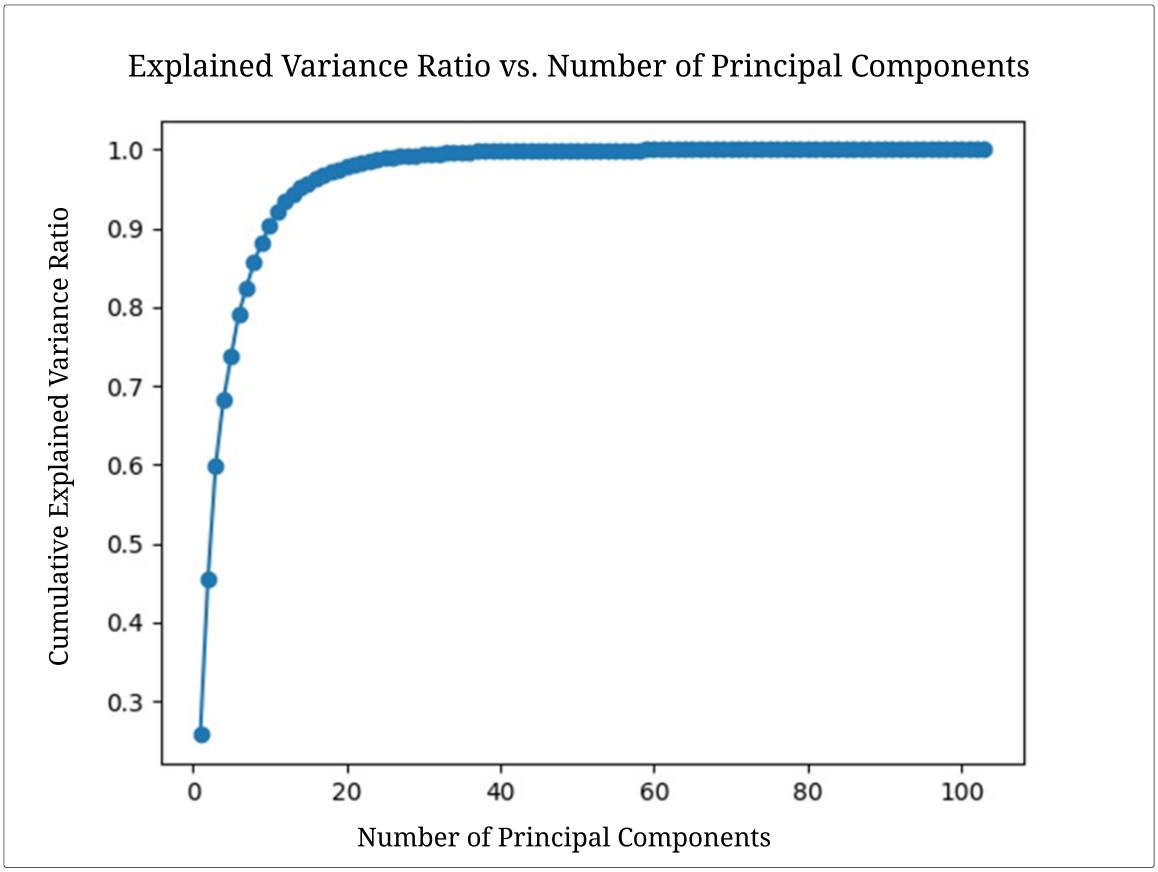


Figure 3.6 A graph depicting how 26 principal components can be used to capture 99% of the cumulative variance of the original data.

3.3.6 Model Architecture

The chosen model architecture is shown in Figure 3.7. This architecture uses an input layer that is dynamically configured to ingest each group of the selected features. Following the input layer are four hidden layers that decrease in size. Additionally, a dropout layer was introduced in training exercises that contained all 197 features with the intention of eliminating highly correlated features. This dropout layer was placed after the first hidden layer with its hyperparameter set to eliminate 50% of the previous layer's neurons' outputs. The output layer is a single neuron that predicts the Lucerno score for the patient. It is important to note that this output layer can be substituted with a binary output layer, if desired. A continuous output layer was selected due to the lack of consensus on how extreme an extravasation needs to be before the need for intervention. With this approach, it is recommended that a radiologist reviews the output of the final layer, using the recommended threshold of 700, to determine if the patient can continue to diagnosis and therapy, or if the patient should be rescanned to ensure accurate PET/CT analysis.

3.4 Results

This section captures the results from each of the classification methods discussed. As previously mentioned, each metric used the recommended threshold of 700 for determining if a patient is extravasated or not extravasated. The fully connected neural network was trained using a Google Colab instance. This instance provides access to an A100 GPU; however, it is important to note that these results can be replicated using a T4 GPU. Each training exercise consisted of 50,000 epochs with a learning rate of 1E-4. Early stoppage with a patience of 5,000 epochs was used to restore the weights that achieved the lowest validation loss. Each of the training exercises were performed more than once to ensure reproducibility.

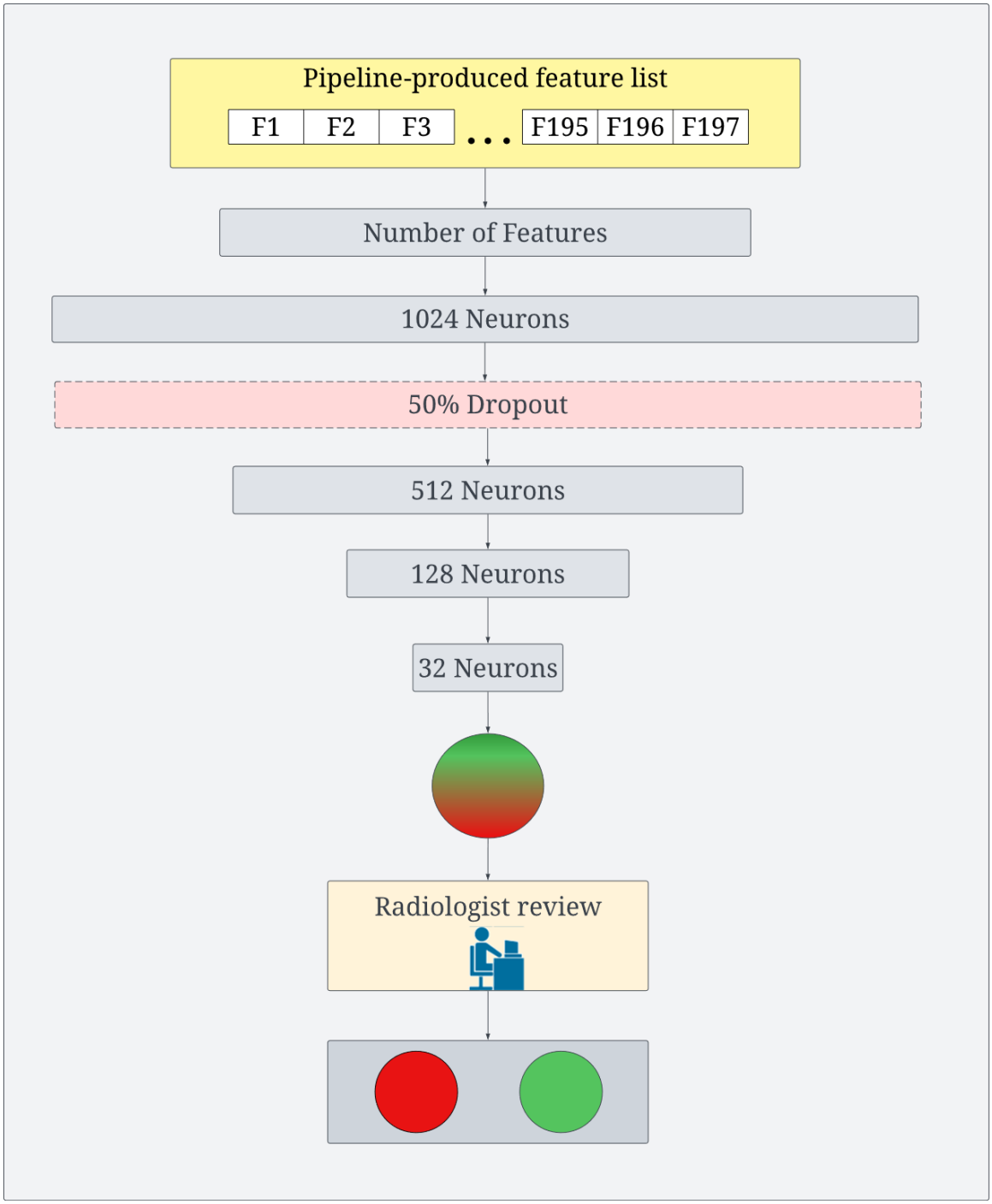


Figure 3.7 A row of features produced by the pipeline being plugged into the chosen fully connected architecture. The final layer is a single neuron that produces a score that the radiologist can observe if the patient was extravasated and should be admitted for a rescan.

3.4.1 Sequential Selection Classification Results

The highest validation accuracy of the sequential data selection approach was **87.50%**. This was achieved using dropout in the architecture and the entirety of the feature list for training. The highest specificity and precision values were **100.00%** and **100.00%** respectively. These were achieved in three distinct categories. Finally, the highest recall and F1-score were **87.50%** and **85.71%** respectively. These results are summarized in Table 3.1 below.

3.4.2 Extreme Extravasation Classification Results

For the extreme extravasation classification approach, Figures 3.7-3.12 illustrate which patients the model predicted correctly and incorrectly to provide a more comprehensive understanding of the results. Each figure includes the predicted and ground truth Lucerno scores for each patient. The correct guesses are identified using a black arrow and the incorrect guesses have a slash through the prediction. A summarization of these results is shown in Table 3.2. Perfect performance was achieved using only the mean, median, and max SUVbws for each of the six areas of interest.

3.4.3 K-Fold Validation Results

Lastly, Table 3.3 shows the K-fold validation results. With this approach, the best validation accuracy of **75.63%** was achieved when using all 197 features. The highest specificity and precision were **90.00%** and **85.71%** respectively and was achieved using only the mean, median, and max, weight, and age. The highest recall was **67.50%**. The highest F1-Score was **73.47%** and was achieved using all of the features. Loss plots from a single fold of this validation is depicted in Figure 3.14. These plots summarize the decrease in loss found for all training exercises.

Table 3.1 ANN results for the sequential classification approach. The best results for accuracy, specificity, precision, recall, and F1 Score are bolded for emphasis.

	Accuracy	Specificity	Precision	Recall	F1 Score
MMM, Weight, Age	81.25%	87.50%	85.71%	75.00%	80.00%
ATM, Skewness, Variance, Kurtosis, Weight, Age	75.00%	87.50%	83.33%	**87.50%**	71.43%
MMM & Ratios, Weight, Age	81.25%	**100.00%**	**100.00%**	62.50%	76.92%
All Data (Excluding Age)	75.00%	**100.00%**	**100.00%**	50.00%	66.67%
All Data (Dropout Layer)	**87.50%**	**100.00%**	**100.00%**	75.00%	**85.71%**
All Data	81.25%	100.00%	100.00%	62.50%	76.92%

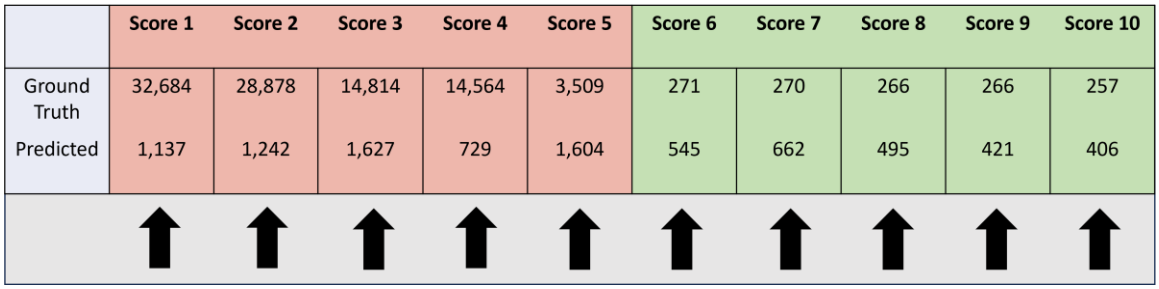


Figure 3.8 Correct and incorrect validation guesses produced by the model when trained on MMM, patient weight, and patient age features.

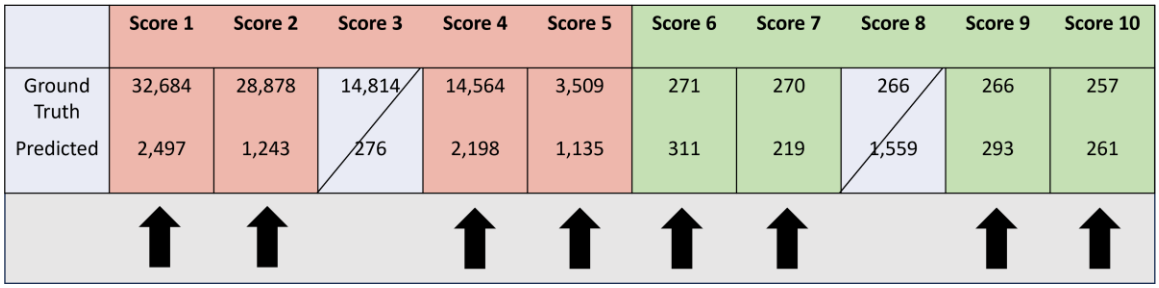


Figure 3.9 Correct and incorrect validation guesses produced by the model when trained on the alpha-trimmed mean (10%), variance, skewness, kurtosis, patient weight, and patient age features.

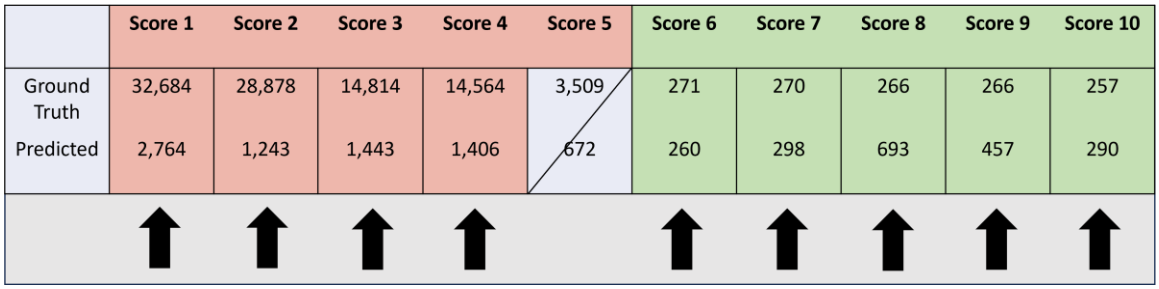


Figure 3.10 Correct and incorrect validation guesses produced by the model when trained on MMM, MMM ratios, patient weight, and patient age features.

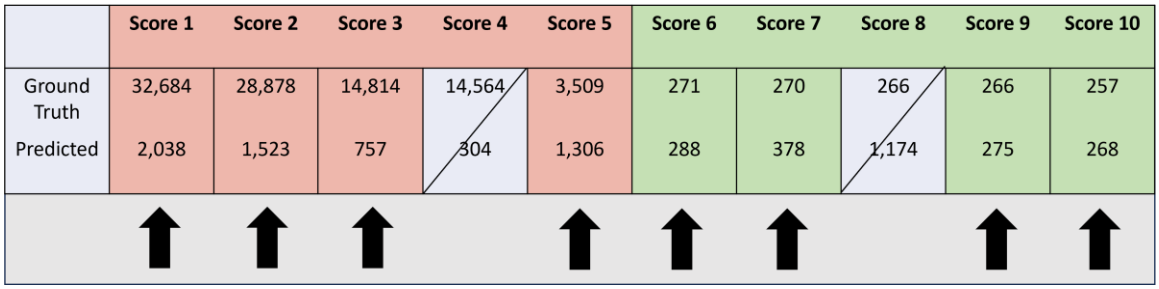


Figure 3.11 Correct and incorrect validation guesses produced by the model when trained on all features except the patient's age.

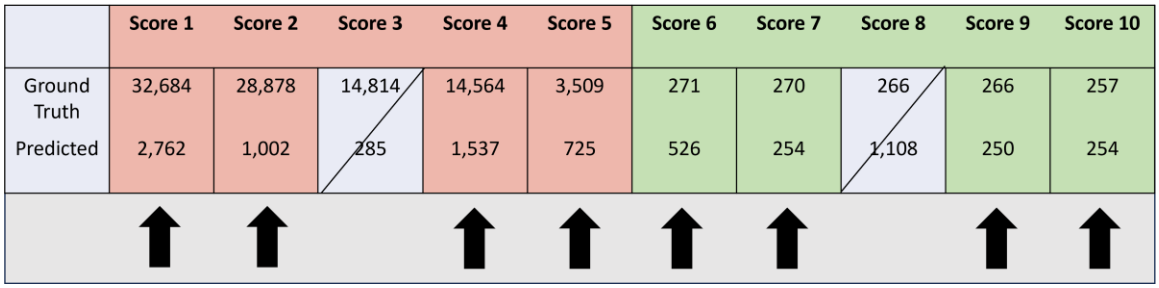


Figure 3.12 Correct and incorrect validation guesses produced by the model when trained on all of the features using a dropout layer in architecture.

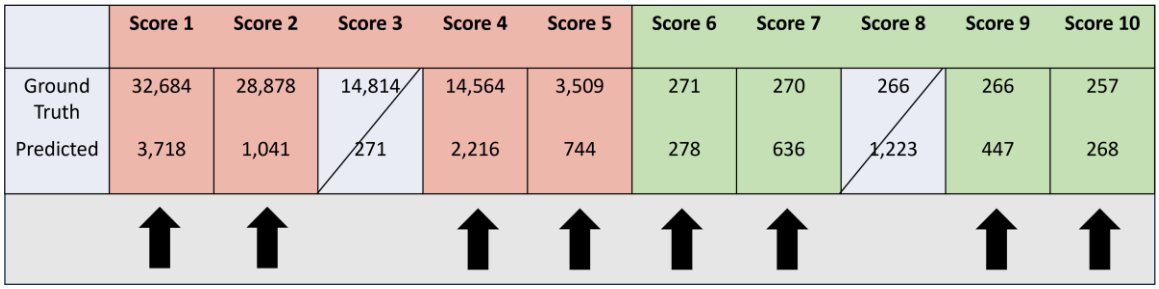


Figure 3.13 Correct and incorrect validation guesses produced by the model when trained on all of the features.

Table 3.2 ANN summarization table for the extreme extravasation cases validation results. The best results for accuracy, specificity, precision, recall, and F1 Score are bolded for emphasis.

	Accuracy	Specificity	Precision	Recall	F1 Score
MMM, Weight, Age	**100.00%**	**100.00%**	**100.00%**	**100.00%**	**100.00%**
ATM, Skewness, Variance, Kurtosis, Weight, Age	80.00%	80.00%	80.00%	80.00%	80.00%
MMM & Ratios, Weight, Age	90.00%	**100.00%**	**100.00%**	80.00%	88.89%
All Data (Excluding Age)	80.00%	80.00%	80.00%	80.00%	80.00%
All Data (Dropout Layer)	80.00%	80.00%	80.00%	80.00%	80.00%
All Data	80.00%	80.00%	80.00%	80.00%	80.00%

Table 3.3 ANN results for K-Fold validation. The best results for accuracy, specificity, precision, recall, and F1 Score are bolded for emphasis.

	Accuracy	Specificity	Precision	Recall	F1 Score
MMM, Weight, Age	75.00%	**90.00%**	**85.71%**	60.00%	70.59%
ATM, Skewness, Variance, Kurtosis, Weight, Age	75.00%	82.50%	79.41%	**67.50%**	72.97%
MMM & Ratios, Weight, Age	74.38%	87.50%	83.05%	61.25%	70.50%
All Data (Excluding Age)	73.75%	86.25%	81.67%	61.25%	70.00%
All Data (Dropout Layer)	75.00%	85.00%	81.25%	65.00%	72.22%
All Data	**75.63%**	83.75%	80.60%	**67.50%**	**73.47%**

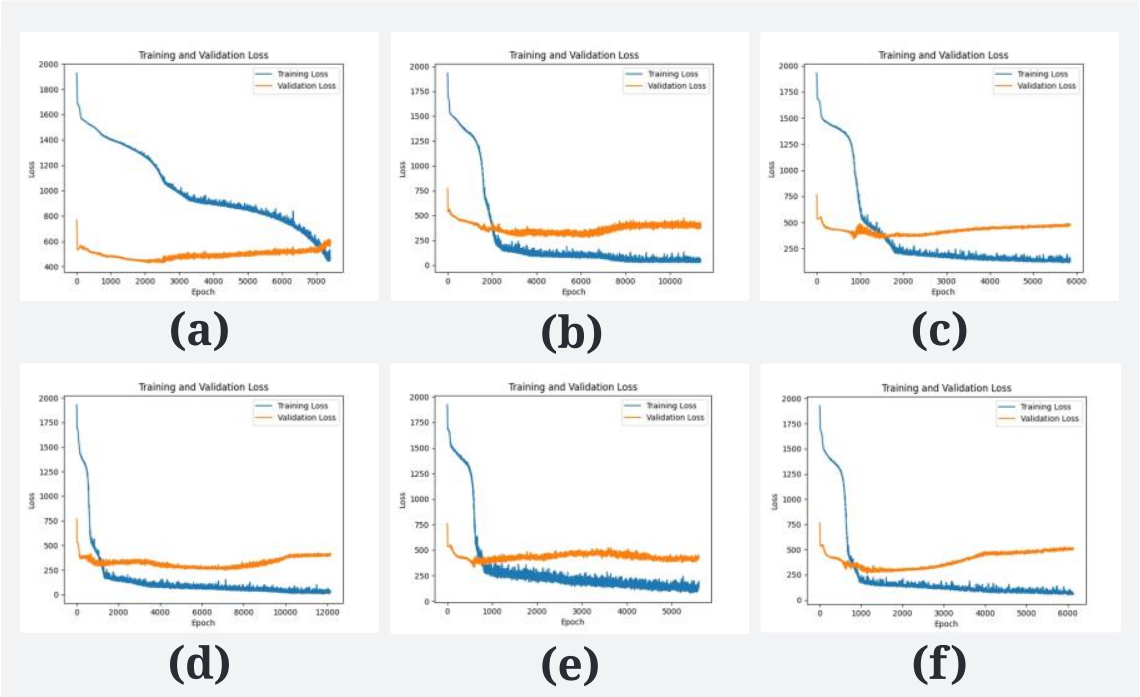


Figure 3.14 Loss plots from the K-Fold Validation exercises: (a) MMM, Weight, and Age, (b) ATM, Skewness, Variance, Kurtosis, Weight, Age (c) MMM, Ratios, Weight, Age, (d) All data excluding age, (e) All data with dropout layer, (f) All data without dropout layer

CHAPTER 4:
APPLICATION OF A CONVOLUTIONAL NEURAL NETWORK

4.1 Overview

The success of fully connected neural networks in Chapter 3 suggested that there is a relationship between radiopharmaceutical extravasation and a patient's summary statistics in the identified regions of interest. This gives rise to the idea that some alternative voxel subsets might provide better performance. One approach for finding this combination is exhaustive combinatorial optimization; however, given that each PET/CT slice is just a two-dimensional vector of voxel values, it is logical to leverage convolutional neural networks, which are well suited for this automatic feature discovery.

This process began with the most straightforward approach, inserting every slice within a patient's scan into the model. Next, this approach is narrowed to only the axial slices within the ROIs that were identified in Chapter 2. This chapter concludes with a demonstration of how an increase in validation data and data augmentation can improve CNNs' ability to classify patients as extravasated. This approach aims to provide an additional option for identifying extravasation in PET/CT data with a methodology that is less coupled to the pipeline presented in Chapter 2.

4.1.1 Convolutional Neural Networks

Similar to ANNs, a CNN is a feed-forward architecture composed of an input layer, a hidden layer(s), and an output layer. The difference between an ANN and a CNN is the introduction of convolution kernels and pooling layers used to replace fully connected layers. These convolution layers use fewer parameters to find hidden features that might indicate extravasation. It is likely that these features are not robust enough to be identified by radiologists but have some patterns that can be learned through the automated feature extraction that CNNs employ.

Convolution kernels, often referred to as a window or filter, are composed of updatable neurons that “convolve” the input image. This convolution is performed by overlaying the kernel on top of a neighborhood of pixels and executing element-wise multiplication. This process produces feature maps, which can be thought of as a less noisy interpretation of the input image.

After this convolution process, pooling layers capture a summary statistic (e.g. max, average, etc.) for the inner neighborhoods of pixels within the feature map [39]. This in turn reduces dimensionality. Following the pooling layer, each neuron is fed through an activation function and used as input for the next layer. The output of the final convolution layer is flattened and fed through a fully connected neural network to determine whether the input slice was extravasated or not extravasated.

In recent years, researchers have increased the number of convolution layers creating deep neural networks that achieve better performance [48-50]. Due to data and resource constraints, these state-of-the-art models, such as ResNet-34, were avoided because they underperform on smaller datasets when compared to smaller CNNs [51].

4.2 Methods

4.2.1 Entire Body Single Slice Classification

The initial approach classified the patient’s entire PET scan on a slice-by-slice basis. For example, if the first patient had 400 axial slices, this would be 400 training samples, where each slice contributes individually to the metrics calculations. Similar to the classification of the tabular data, this process for splitting the data began with selecting the first 40 patients with a score above a Lucerno score of 700 and below a Lucerno score of 700 for training and the next 16 patients that fit this criterion for validation. The data was split at a patient level to avoid data contamination from the same patient’s slices appearing in both the training and validation data. Lastly, the data was shuffled and normalized to the highest pixel value.

Each slice of slices varied between two different dimensions. Reducing the larger PET slices would result in the loss of physiological artifacts and enlarging the smaller slices would have extrapolated data inaccurately, so all of the slices were cropped to an equivalent scale. Although CNNs are not inherently scale-invariant, the method for splitting the data ensured the training and validation data had both types of dimensions. This created a smooth training environment.

4.2.2 Individual Area Single Slice Classification

After testing the entirety of a patient's PET scan, the data was limited to only the axial slices that contained the liver, brain, aorta, and liver ROI. These areas were selected to reflect the areas found by previous work that are highly correlated to extravasation [28]. Some of these regions are shown in Figure 4.1. The general locations of these boundaries are similar for each of the patients within the dataset. The liver slices contained the aorta; however, any axial slice that contained both the liver and aorta was removed from the aorta dataset.

4.2.3 Slice Clustering Classification

Upon examination of the individual area slice classifications, it was evident that the model was consistently classifying two slices from the same patient as two different labels, extravasated and non-extravasated. This was a side effect of the artificial data generation used to break each patient into a collection of individual data points. This issue was overcome by developing a classification strategy coined slice clustering classification. This slice clustering classification technique performs classification on each individual slice and then takes the average of the slices' predictions to produce a patient-level prediction. This process is illustrated in Figure 4.2.

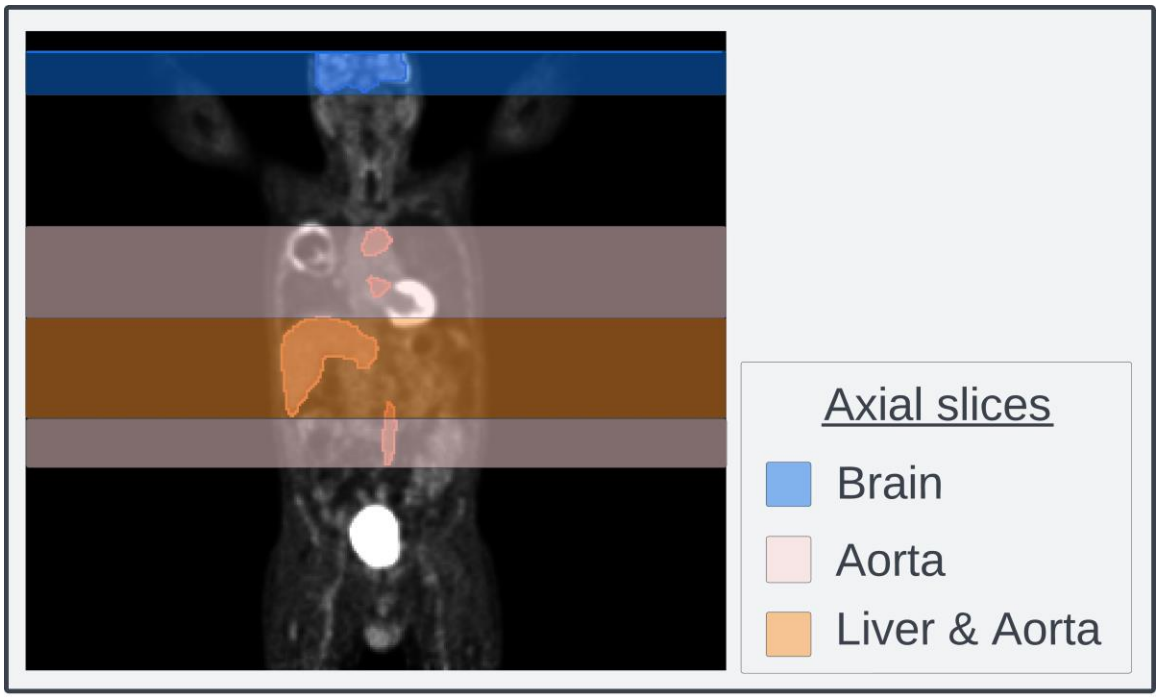


Figure 4.1 Slices corresponding to the organs of interest used for training data

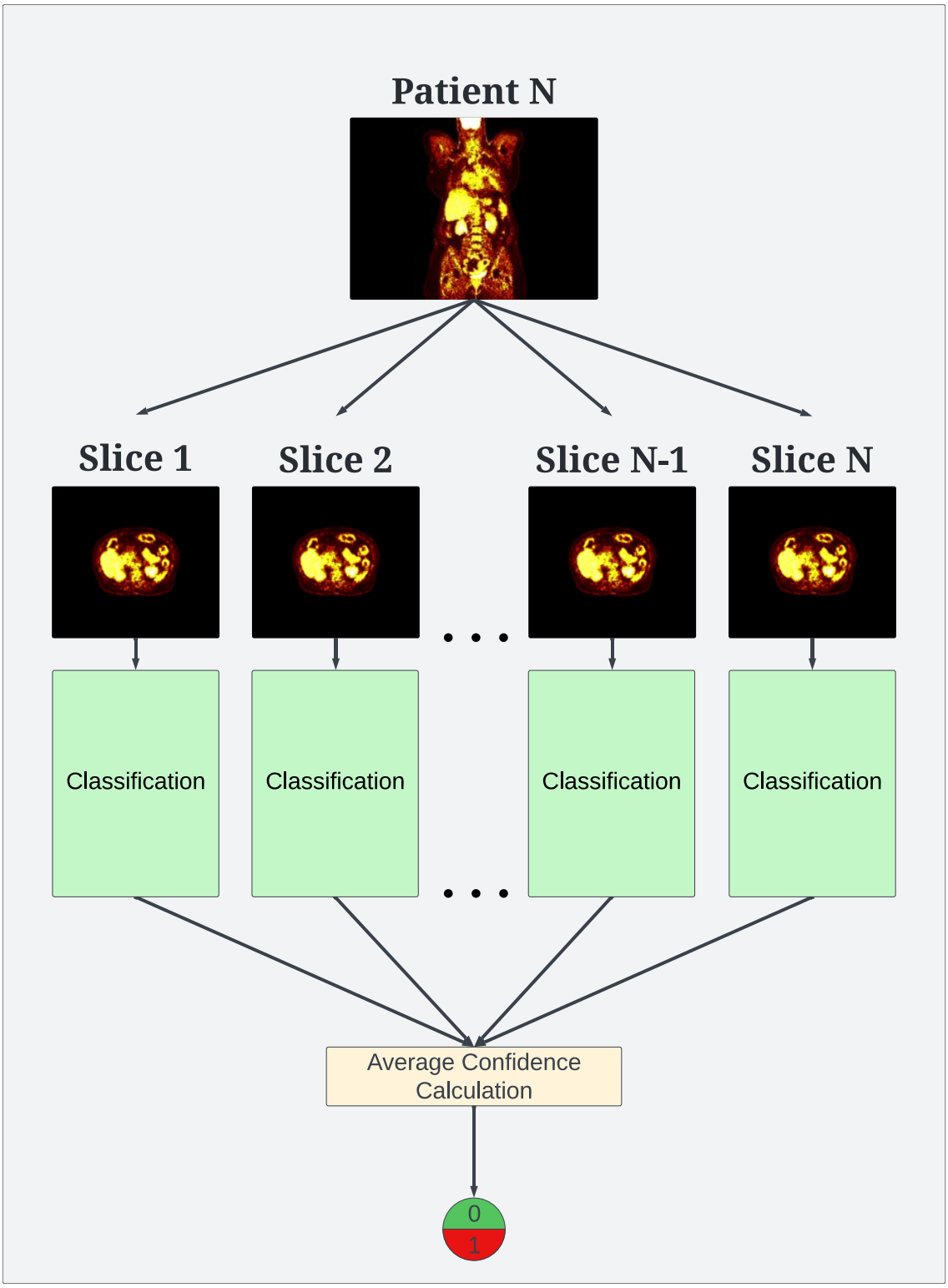


Figure 4.2 Slice clustering illustrated to show how the collective confidences for each slice produce a prediction for a single patient

4.2.4 Data Augmentation

The scale-invariance and inconsistent positioning of anatomical structures within PET slices were addressed using data augmentation. This technique artificially creates additional training samples for neural networks [53], using operations such as cropping, rotating, and shifting of the data. The objective is to force the model to refrain from creating relationships solely based on positioning and scale, thereby enforcing the neural network to learn a more complex latent space. This creates a more general mapping between the various PET slices and the corresponding outputs.

The selected data augmentation operations generated 19 additional training samples for each PET slice. These augmentations are shown in Figure 4.3 below. First, 3 augmented slices were produced by rotating each PET slice 90, 180, and 270 degrees. Next, each PET slice was shifted and rotated in each of the corners using two distinct approaches to create 8 additional slices. Lastly, each PET slice was rotated and then cropped. The coronal plane was cropped to eliminate artifacts, such as the patient's injection arm, which were only present in a limited number of the patients' PET/CT scans.

There are two additional details to note for this strategy. First, data augmentation was only performed on the training data. This prevented any modification to the validation data that might lead to an artificial improvement in performance. Second, the data augmentation techniques were only applied to the reduced liver ROI slices. Although an increase in performance was found for other regions, this approach produced the best results. This implementation decision is also more reproducible for future research efforts because it requires significantly less computational resources for retraining the model.

4.2.5 Increasing Training Data

Adding additional data from the validation dataset to the training dataset simulates the effect that an increase in training data would have on the validation accuracy. This was done by adjusting the validation split from 83% to ~96% and ~98%. For the 96%

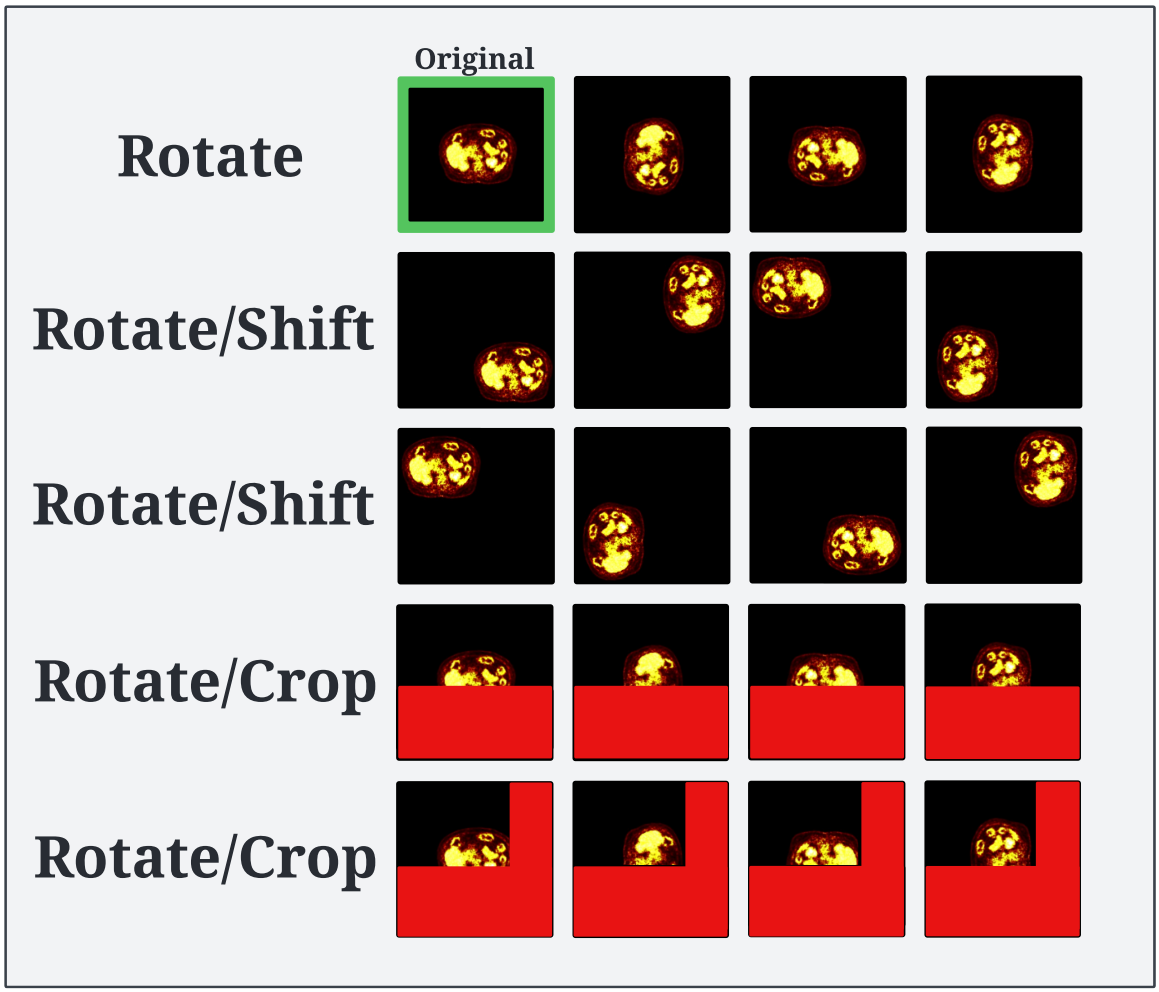


Figure 4.3 Data augmentation techniques used to generate 20 separate data points for a single PET slice.

validation split, a similar process was followed of iterating through the dataset sequentially and creating a balanced training and validation dataset of 92 patients and 4 patients respectively. The 98% validation split used 2 patients as holdout samples.

4.2.6 Model Architecture

The CNN architecture used is shown in Figure 4.4. This configuration begins with five convolution layers, each using a convolution kernel of size 3x3 to create feature maps of the input data. The number of layers was chosen through trial and error. The kernel size for each convolution layer was relatively small to limit any excessive blurring that might eliminate a key physiological artifact that might exist in a limited subset of voxels. A max-pooling layer is employed between each convolution layer to reduce the dimensionality of each feature map. The final convolution layer feeds into an ANN for classification that produced a confidence value which represents the likelihood that the patient is extravasated. This means that the model classifies each slice as either extravasated or not extravasated; however, the same results could be achieved using a score-based output with a threshold of 700.

4.3 Results

The following figures and tables depict each of the results found using the methods discussed above. Each of these training exercises used an A100 GPU on a Google Colab instance. The training exercises were composed of 2,500-10,000 epochs with a learning rate of 1E-6.

4.3.1 Slice-by-Slice Classification Results

Table 4.1 presents the results collected for the entire body and each of the areas of interest discussed above. With this approach, the best validation accuracy, recall, and F1 score found was **71.33%, 70.42%, and 70.92% respectively** using the liver ROI axial

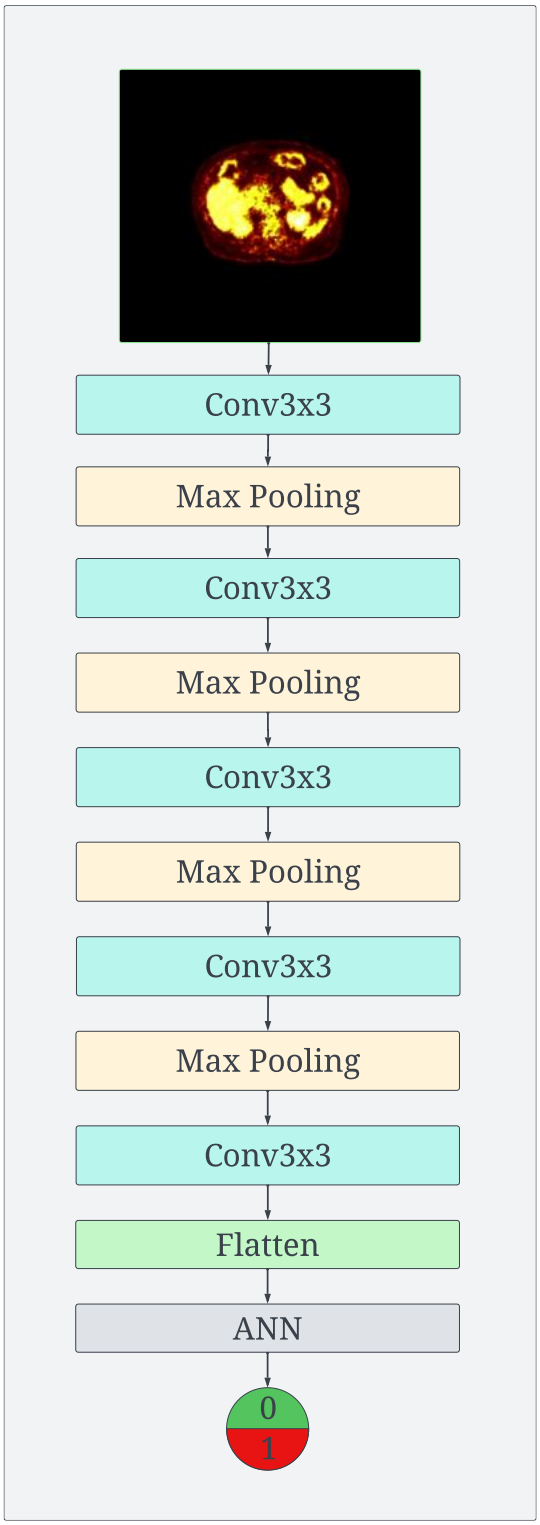


Figure 4.4 Convolutional neural network architecture designed to ingest a CT slice and output extravasated or not extravasated.

slices. The best specificity found was **85.34%** using the brain axial slices. The best precision score was **72.50%** and was captured when using the liver slices.

4.3.2 Slice Clustering Classification Results

The results from the slice clustering process employed to capture metrics at a patient level are displayed in Table 4.2. From this approach, there was a significant improvement in the model's ability to classify each patient. The model achieved the best results when classifying liver ROI slices and all of the slices collectively, achieving a validation accuracy, specificity, precision, recall, and F1-Score of **81.25%, 87.50%, 85.71%, 75.00%, and 80.00%,** respectively.

4.3.3 The Effect of Increasing Training Data

Table 4.3 shows the results found from increasing the training data through data augmentation and validation split adjustments. As mentioned above, these results were all collected using the liver ROI slices within each patient. Each model was trained for 2,500 epochs, and the weights that resulted in the lowest validation loss were restored and used to measure performance. Using this strategy, there was an increase in validation accuracy from 71.33% to 74.83% when augmenting the data. There was an increase from 71.33% to 88.57% when raising the validation split from 83% to 96%. The best validation accuracy, specificity, precision, and recall for this training exercise of **91.43%, 83.33%, 85.00%, 100%, and 91.89%** respectively were found when performing data augmentation and increasing the validation split simultaneously. Perfect recall was also achieved when adjusting the validation split without performing data augmentation. The accuracy plots and loss plots from the 96% validation split training exercises are depicted in Figure 4.5 and Figure 4.6, respectively. These results give weight to the notion that a hurdle for these training exercises was a lack of training data.

One final training exercise was used where all of the discussed methodologies were employed simultaneously. The results from this approach are shown in Table 4.4. This

included increasing the validation split from 83% to 98%, performing data augmentation, and clustering the slice predictions. Early stoppage was introduced to end training and restore the best-performing weights after 300 epochs without a decrease in validation loss. With this approach, the model was able to classify both the randomly selected validation patients, with perfect performance. It is important to note that the model was able to achieve perfect performance without data augmentation; however, Figure 4.7 shows how using data augmentation leads to faster convergence.

Table 4.1 Slice-by-slice classification results for the entire body, aorta, liver, brain, and ROI within the liver. The best results are bolded for emphasis.

	Accuracy	Specificity	Precision	Recall	F1 Score
Aorta	64.17%	65.88%	66.52%	62.58%	64.49%
Brain	69.76%	**85.34**%	71.43%	48.95%	58.09%
Liver	66.28%	75.68%	**72.50**%	57.80%	64.33%
Liver ROI	**71.33**%	72.22%	71.42%	**70.42**%	**70.92**%
All Data	69.27%	68.62%	69.74%	6989%	69.81%

Table 4.2 Slice clustering classification results for the entire body, aorta, liver, brain, and ROI within the liver. The best results are bolded for emphasis.

	Accuracy	Specificity	Precision	Recall	F1 Score
Aorta	68.75%	75.00%	71.43%	62.50%	66.67%
Brain	68.75%	75.00%	71.43%	62.50%	66.67%
Liver	68.75%	75.00%	71.43%	62.50%	66.67%
Liver ROI	**81.25%**	**87.50**%	**85.71**%	**75.00**%	**80.00**%
All Data	**81.25%**	**87.50**%	**85.71**%	**75.00**%	**80.00**%

Table 4.3 Results from applying data augmentation and an increase in the validation split. The best results are bolded for emphasis.

	Accuracy	Specificity	Precision	Recall	F1 Score
Non-Augmented (83%)	71.33%	72.22%	71.42%	70.42%	70.92%
Augmented (83%)	74.83%	68.06%	71.60%	81.69%	76.31%
Non-Augmented (96%)	88.57%	77.78%	80.95%	100.00%	89.47%
Augmented (96%)	**91.43%**	**83.33%**	**85.00%**	**100.00%**	**91.89%**

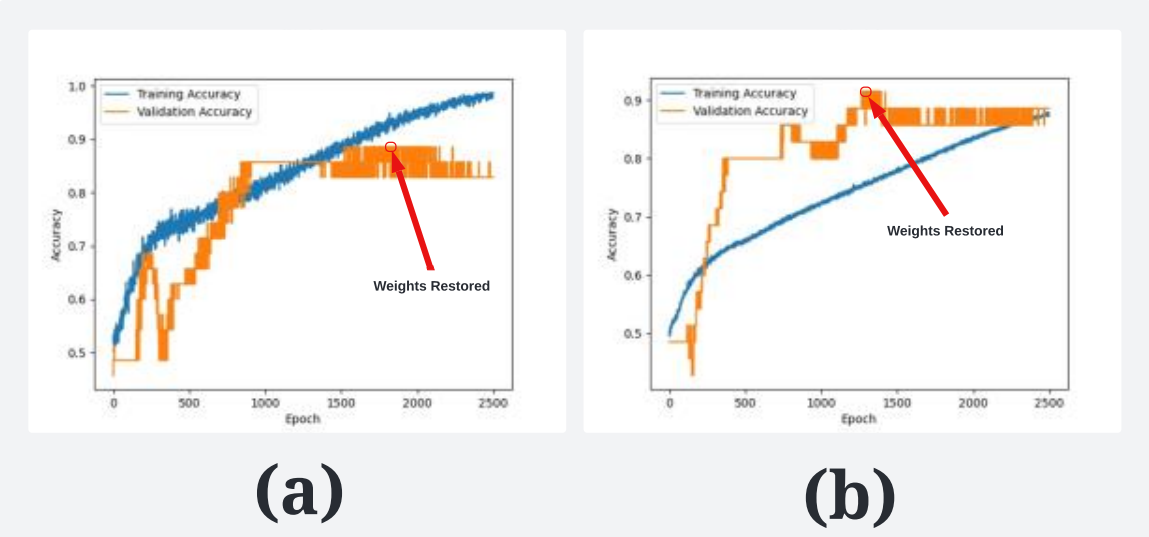


Figure 4.5 Accuracy plots for the 96% validation split training exercises: (a) No data augmentation, (b) Augmented data

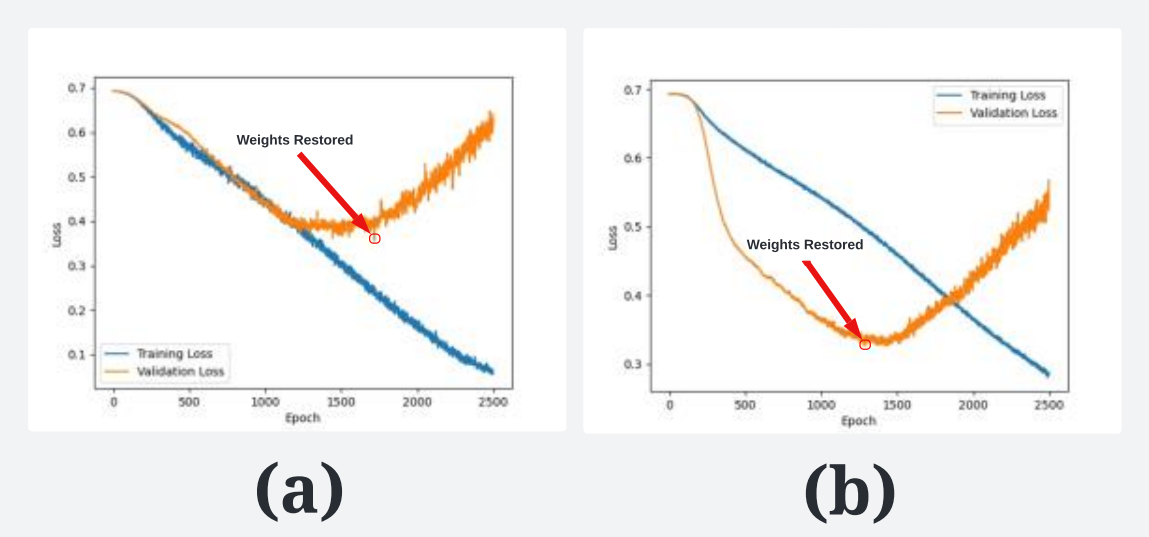


Figure 4.6 Loss plots for the 96% validation split training exercises: (a) No data augmentation, (b) Augmented data

Table 4.4 Results from applying data augmentation, slice clustering, and using only two patients for validation data.

	Accuracy	Specificity	Precision	Recall	F1 Score
Non-Augmented No Slice Clustering (98%)	100.00%	100.00%	100.00%	100.00%	100.00%
Augmented No Slice Clustering (98%)	100.00%	100.00%	100.00%	100.00%	100.00%
Non-Augmented Slice Clustering (98%)	100.00%	100.00%	100.00%	100.00%	100.00%
Augmented Slice Clustering (98%)	100.00%	100.00%	100.00%	100.00%	100.00%

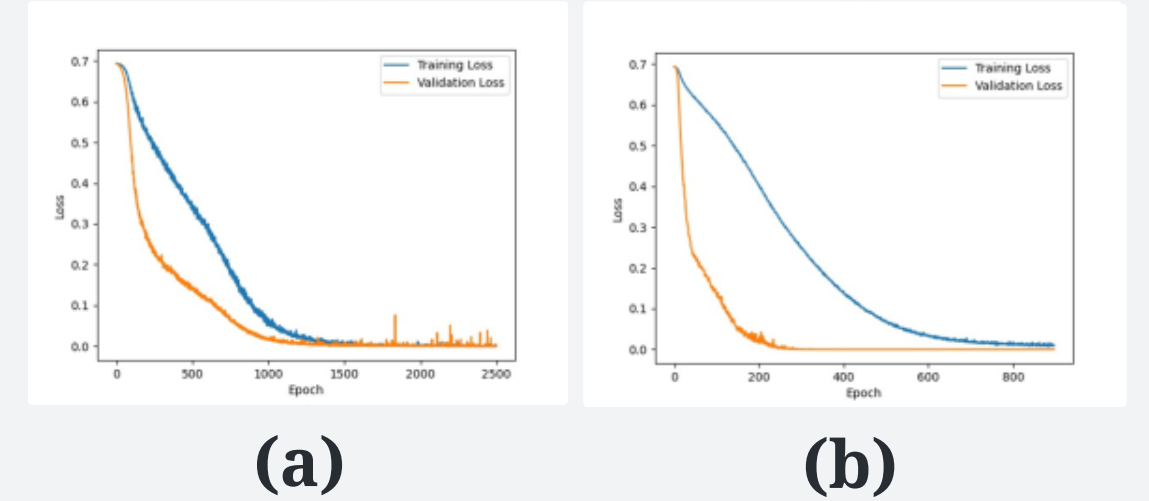


Figure 4.7 Loss plots for 98% validation split training: (a) Not Augmented, (b) Augmented

CHAPTER 5:
CONCLUSIONS

The objective of this work was to develop a streamlined workflow capable of ingesting directories of PET/CT data and determine whether the underlying patients were extravasated. This approach involved the use of a pipeline that automated the process of creating ROIs within key organs, resembling ROIs manually drawn by radiologists. The pipeline also extracted summary statistics from the distribution of body weight normalized standard uptake values in these ROIs. These extracted values were exported to an SUVbw metric report.

From here, the examination of these pipeline-produced features could be examined by radiologists; however, this thesis showed how the analysis can be offloaded to a fully connected neural network that predicts how much of the radiopharmaceutical tracer was extravasated. The success of this approach prompted the idea that other prominent features within the body may have evaded previous works' medically informed feature extraction. This idea led to an alternative classification approach using convolutional neural networks employed to discover these non-robust, unconventional feature sets that have previously gone unexplored. This approach leveraged data augmentation, various validation splits, and slice clustering to develop better models that directly combat the lack of sufficient data.

With the combination of this novel pipeline and distinct classification methodologies, this research aimed to provide a ready-to-deploy workflow that can be integrated into existing PET/CT channels. Like most research, shortcomings and limitations were encountered and future work was considered during the lifetime of this research. The following content captures this material.

5.1 Shortcomings/Limitations

5.1.1 Lack of Data

A significant constraint for the classification methods was the lack of both data quantity and quality that could be used for training and testing. Many neural networks leverage an abundance of data, such as ImageNet [54] which is composed of over 1.2 million training images. This allows neural networks to build a complex latent space that allows for a generalized mapping between inputs and outputs. With medical applications, researchers often run into data allocation issues due to patient protection standards and institutional review board (IRB) overhead.

For this project, the dataset was limited to 103 patients, which is far less than preferred. In terms of data quality, there was only a limited number of extremely extravasated patients. This meant that even though half the dataset was labeled as "extravasated," there might not be enough underlying physiological differences between the extravasated patients' PET scans and non-extravasated patients' PET scans to differentiate the two cases. This lack of extremely extravasated patients stems from it being easier to collect non-extravasated patients and slightly extravasated patients than extremely extravasated patients. It is important to note that efforts are developing to simulate extravasation by administering the radiopharmaceutical tracer in various fashions that mimic the appearance of an extravasation without causing patient harm. These efforts combined with the methodologies discussed in this thesis can create additional data allowing for better classification performance in the future.

5.1.2 Addressing the Reproducibility of the CNN Results

Another limitation was the lack of resources that could be used for further verification of results. For the purposes of transparency and reproducibility, it is important to note that due to these resource constraints, additional validation for the convolutional neural

network results using techniques such as K-Fold validation demonstrated in Chapter 3 could not be performed. Although not illustrated, efforts were made to rotate through a few selections of validation data to ensure consistent results; however, it is possible higher results were achieved due to a fortunate data selection.

5.2 Future Work

5.2.1 3D Convolutional Neural Networks

With the success of convolutional neural networks, it became more evident that there are features that can be found through automatic feature extraction. These features may exist between axial slices or are volumetric features that are not obtainable when working in two dimensions. These features could be discovered through the use of three-dimensional convolutional neural networks. Efforts were made during this research to create a 3D CNN capable of identifying PET/CT radiopharmaceutical extravasation; however, these efforts were discontinued due to resource constraints that forced excessive scaling of the data. This scaling resulted in subpar results when compared to the ANN and CNN approaches. With an increase in resources and data, especially highly extravasated patients, this could be a useful direction for future research.

5.2.2 Improvements in the SUVbw Extraction Pipeline

Many improvements could be made to improve the pipeline's ability to create ROIs. One recommended improvement is retraining TotalSegmentator's underlying U-Net to segment the desired ROIs. Within this approach, one could significantly extend the cross-sectional approach shown in this work. The model could also learn to avoid certain discrepancies in the CT data, such as blood pools in the liver, that produced large inaccuracies in the approach shown in Chapter 2.

5.2.3 Model Explainability

When developing any medical-oriented system, an understanding of what features contributed to certain outcomes can allow researchers and medical professionals to develop strategies for addressing or adjusting their approaches in the future. Although the scope of this thesis was to provide medical professionals with only an indication that extravasation may have arisen, one enhancement to this approach would be the explainability of these models. This explainability would provide insight into which features contribute more to the likelihood of extravasation, so extra precautions can be taken to avoid exposing patients to this phenomenon in the future. This can also assist with modifications to networks that increase their ability to detect medical extravasation.

LIST OF REFERENCES

1 Bailey, D. L., Maisey, M. N., Townsend, D. W., & Valk, P. E. (2005). *Positron emission tomography* (Vol. 2, p. 22). London: Springer.

2 Rouëssé, J., Alberini, J. L., Wartski, M., Gutman, F., Collignon, M. A., Corone, C., ... & Pecking, A. P. (2005). FDG (18 fluorodeoxyglucose) positron emission tomography and breast cancer. *Bulletin de L'academie Nationale de Medecine*, *189*(5), 963-75.

3 de Nava, A. S. L., & Raja, A. (2022). Physiology, Metabolism. In *StatPearls [Internet]*. StatPearls Publishing.

4 *NCI Dictionary of Cancer terms*. National Cancer Institute. (2011). Retrieved May 15, 2024. https://www.cancer.gov/publications/dictionaries/cancer-terms/def/metabolic

5 IAEA. (2016, April 13). Retrieved May 1, 2024. *Radiotracers*. IAEA. https://www.iaea.org/topics/radiotracers#:~:text=A%20radioactive%20tracer%20is%20a,the%20mechanism%20of%20chemical%20reactions%20.

6 Shahhosseini, S. (2011). Retrieved May 1, 2024. *Pet radiopharmaceuticals*. Iranian journal of pharmaceutical research : IJPR. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3869596/#:~:text=Currently%2C%20there%20are%20four%20PET,for%20identifying%20the%20regions%20of

7 Center for Devices and Radiological Health. (2020). Retrieved May 1, 2024. *What is computed tomography?: FDA*. U.S. Food and Drug Administration. https://www.fda.gov/radiation-emitting-products/medical-x-ray-imaging/what-computed-tomography

8 Kinahan, P. E., Townsend, D. W., Beyer, T., & Sashin, D. (1998). Attenuation correction for a combined 3D PET/CT scanner. *Medical physics*, *25*(10), 2046-2053.

9 U.S. Department of Health and Human Services. (2022). Retrieved May 1, 2024. *Computed Tomography (CT)*. National Institute of Biomedical Imaging and Bioengineering. https://www.nibib.nih.gov/science-education/science-topics/computed-tomography-ct

10 Seeram, E. (2022). *Computed Tomography-E-Book: Computed Tomography-E-Book*. Elsevier Health Sciences.

11 Hermena, S., & Young, M. (2021). CT-scan image production procedures.

12 Schöder, H., Erdi, Y. E., Larson, S. M., & Yeung, H. W. (2003). PET/CT: a new imaging technology in nuclear medicine. *European journal of nuclear medicine and molecular imaging*, *30*, 1419-1437.

13 Kohl, G. (2005). The evolution and state-of-the-art principles of multislice computed tomography. *Proceedings of the American thoracic Society*, *2*(6), 470-476.

14 Hsieh, J., Nett, B., Yu, Z., Sauer, K., Thibault, J. B., & Bouman, C. A. (2013). Recent advances in CT image reconstruction. *Current Radiology Reports*, *1*, 39-51.

15 DenOtter, T. D., & Schubert, J. (2023). Hounsfield Unit. In *StatPearls*. StatPearls

Publishing.
16 Hounsfield, G. N. (1973). Computerized transverse axial scanning (tomography): Part 1. Description of system. *The British journal of radiology*, *46*(552), 1016-1022.
17 Harris, S., Crowley, J. R., & Warden, N. (2023), Radiopharmaceutical Administration Practices-Are They Best Practice?. *Frontiers in Nuclear Medicine*, *3*, 1244660.
18 Osborne, D., Lattanze, R., Knowland, J., Bryant, T. E., Barvi, I., Fu, Y., & Kiser, J. W. (2021). The scientific and clinical case for reviewing diagnostic radiopharmaceutical extravasation long-standing assumptions. *Frontiers in Medicine*, *8*, 684157.
19 *NRC leaders prioritize patients over special interests in extravasation vote*. American College of Radiology. (2023). Retrieved May 15, 2024. https://www.acr.org/Advocacy-and-Economics/Advocacy-News/Advocacy-News-Issues/In-the-Jan-7-2023-Issue/NRC-Leaders-Prioritize-Patients-Over-Special-Interests-in-Extravasation-Vote
20 Takada, K., Kondo, M., & Furukawa, M. (2024). A comparative study of the CT effective dose in whole-body 18F-FDG PET/CT for arm-raised position and arm-lowered position. *Radiography*, *30*(1), 416-421.
21 *The Lucerno solution for extravasations*. Lucerno Dynamics. (2024). Retrieved May 1, 2024. https://lucerno.com/the-lucerno-solution-for-extravasations/
22 *Helping clinicians*. Lucerno Dynamics. (2024). Retrieved May 1, 2024. https://lucerno.com/
23 Perrin, S., Kiser, J. W., Knowland, J., & Bowen, S. L. (2022). Development of a classifier for [18F] fluorodeoxyglucose extravasation severity using semi-quantitative readings from topically applied detectors. *EJNMMI physics*, *9*(1), 61.
24 DICOM. (2021). Retrieved May 15, 2024. https://www.dicomstandard.org/
25 Bidgood Jr, W. D., Horii, S. C., Prior, F. W., & Van Syckle, D. E. (1997). Understanding and using DICOM, the data interchange standard for biomedical imaging. *Journal of the American Medical Informatics Association*, *4*(3), 199-212.
26 Pydicom. (2023). May 15, 2024. https://pydicom.github.io/
27 U.S. Department of Health and Human Services. 2013). Retrieved May 15, 2024. *NIFTI: - neuroimaging informatics technology initiative*. National Institute of Mental Health. https://nifti.nimh.nih.gov/
28 Osborne, D., Thakur, S., & Acuff, S. (2019). A Retrospective Study of Physiologic Markers to Assess Radiopharmaceutical Infiltration Compared to External Injection Monitoring.
29 Minaee, S., Boykov, Y., Porikli, F., Plaza, A., Kehtarnavaz, N., & Terzopoulos, D. (2021). Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, *44*(7), 3523-3542.
30 Sonka, M., Hlavac, V., & Boyle, R. (2013). *Image processing, analysis and*

machine vision. Springer.
31 Schwing, A. G., & Urtasun, R. (2015). Fully connected deep structured networks. *arXiv preprint arXiv:1503.02351*.
32 Kang, K., & Wang, X. (2014). Fully convolutional neural networks for crowd segmentation. *arXiv preprint arXiv:1411.4464*.
33 Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., ... & Girshick, R. (2023). Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 4015-4026).
34 Wasserthal, J., Breit, H. C., Meyer, M. T., Pradella, M., Hinck, D., Sauter, A. W., ... & Segeroth, M. (2023). Totalsegmentator: Robust segmentation of 104 anatomic structures in ct images. *Radiology: Artificial Intelligence*, *5*(5).
35 Isensee, F., Jaeger, P. F., Kohl, S. A., Petersen, J., & Maier-Hein, K. H. (2021). nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, *18*(2), 203-211.
36 *Skimage.transform#*. skimage.transform - skimage 0.23.2 documentation. (2023). Retrieved May 1, 2024, https://scikit-image.org/docs/stable/api/skimage.transform.html#skimage.transform.resize
37 Rachmatullah, M. I. C., Santoso, J., & Surendro, K. (2021). Determining the number of hidden layer and hidden neuron of neural network for wind speed prediction. *PeerJ Computer Science*, *7*, e724.
38 Ramadevi, R., Sheela Rani, B., & Prakash, V. (2012). Role of hidden neurons in an elman recurrent neural network in classification of cavitation signals. *Int J Comput Appl*, *37*(7), 9-13.
39 Sun, J. (2012). Learning algorithm and hidden node selection scheme for local coupled feedforward neural network classifier. *Neurocomputing*, *79*, 158-163.
40 Mattioli, F., Caetano, D., Cardoso, A., Naves, E., & Lamounier, E. (2019). An experiment on the use of genetic algorithms for topology selection in deep learning. *Journal of Electrical and Computer Engineering*, *2019*.
41 LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.
42 Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.
43 Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, *25*.
44 Akbari, A., Awais, M., Bashar, M., & Kittler, J. (2021, July). How does loss function affect generalization performance of deep learning? Application to human age estimation. In *International Conference on Machine Learning* (pp. 141-151). PMLR.
45 Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *nature*, *323*(6088), 533-536.
46 Cauchy, A. (1847). Méthode générale pour la résolution des systemes d'équations simultanées. *Comp. Rend. Sci. Paris*, *25*(1847), 536-538.

47 Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), 1929-1958.
48 He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).
49 Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
50 Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).
51 Brigato, L., & Iocchi, L. (2021, January). A close look at deep learning with small data. In *2020 25th International Conference on Pattern Recognition (ICPR)* (pp. 2490-2497). IEEE.
52 Shorten, C., & Khoshgoftaar, T. M. (2019). A survey on image data augmentation for deep learning. *Journal of big data*, *6*(1), 1-48.
53 Petrini, L., Cagnetta, F., Vanden-Eijnden, E., & Wyart, M. (2022). Learning sparse features can lead to overfitting in neural networks. *Advances in Neural Information Processing Systems*, *35*, 9403-9416.
54 Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., & Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition* (pp. 248-255). Ieee.

VITA

Elijah was originally born in Minnesota. At a young age, he moved to a small town named Lobelville in middle Tennessee. Here, he attended Perry County High School. After graduation, Elijah attended The University of Tennessee at Knoxville, where he received his bachelor's degree in computer engineering. In the later years of his college career, Elijah found a particular interest in machine learning, deep learning, and the various applications that this domain provided. This interest led to the pursuit of a master's degree in computer science, where he worked with Dr. Dustin Osborne and Dr. Jens Gregor to develop neural-network-based strategies for identifying and classifying radiopharmaceutical extravasation.