CATÓLICA
ESCOLA SUPERIOR DE BIOTECNOLOGIA
PORTO

SEGMENTATION OF MALIGNANT LESIONS ON [^{18}F]FDG PET/CT IMAGES USING DEEP LEARNING TECHNIQUES

by

Mafalda Moreira Oliveira

January 2023

CATÓLICA
ESCOLA SUPERIOR DE BIOTECNOLOGIA
PORTO

SEGMENTATION OF MALIGNANT LESIONS ON [^{18}F]FDG PET/CT IMAGES USING DEEP LEARNING TECHNIQUES

Training Placement Report presented to *Escola Superior de Biotecnologia* of the *Universidade Católica Portuguesa* to fulfill the requirements of Master of Science degree in Biomedical Engineering

by

Mafalda Moreira Oliveira

Supervisor: PhD Francisco P. M. Oliveira

Co-Supervisor: MSc Cláudia Santos Constantino

Tutor (University): Professor Doutor Pedro Rodrigues

December 2022

Abstract

Cancer is one of the leading causes of death worldwide, and that can be prevented through early detection and effective treatment of malignancies. Hence, tumour identification and segmentation take a vital part in the early detection of lesions, as well as in radiotherapy procedures, and surgical planning. At present, manual segmentation is the gold standard, which remains a daunting challenge as it is time-consuming, labour intensive, tedious, and highly subjective, since introduces inter- and intra-operator variability. Thereby, the paramount purpose of this internship was to apply, optimise and assess the feasibility of deep learning-based techniques for the automatic identification and segmentation of malignant lesions in whole-body [^{18}F]FDG PET/CT images. Hence, three different datasets were used to train different networks: CT images for spleen segmentation; [^{18}F]FDG PET/CT with suspected malignant lesions; and lesions suggestive of lymphoma on whole-body PET images. Subsequently, a 3D U-net architecture was developed and optimised for automatic identification and segmentation of the objects of interest. Due to GPU computational capacity limitations, several approaches needed to be implemented for the 3D U-net training process and inference testing. The Dice coefficient (DC) was used as an overlap measure between the ground truths and the resulting segmentations. The first dataset achieved the highest median DC of 0.57 using the network with the transfer learning method and a CT intensity normalisation of [-250; 250]. Regarding the second dataset, the median DC obtained was 0.28, when the 3D U-net was trained with patches of 48×48×48. Finally, the third dataset achieved a median DC of 0.41, whereas the patch size was 64×64×64 voxels on a U-net configuration with one less layer. In conclusion, fully automatic segmentation methods based on deep learning techniques for lesion identification and segmentation need clinical supervision for verification and adjustments. For now, this method is unacceptable to use in clinical practice alone, since it is not robust enough.

Keywords: [^{18}F]FDG PET/CT; Malignant Lesions; Deep Learning; Fully Automatic Segmentation

Resumo

O cancro é uma das primordiais causas de morte mundialmente, podendo ser prevenida através de uma deteção precoce e do tratamento eficaz das lesões malignas. Deste modo, a identificação e segmentação de tumores têm um papel fundamental na deteção precoce de lesões, bem como na radioterapia e no planeamento cirúrgico. Atualmente, embora a segmentação manual seja o padrão-ouro, este é um método demorado, trabalhoso, tedioso e altamente subjetivo, pois introduz variabilidade inter- e intra-operador. Assim sendo, o objetivo primário deste estágio foi aplicar, otimizar e avaliar a viabilidade de técnicas baseadas em *deep learning* para a identificação e segmentação automática de lesões malignas em imagens de corpo inteiro [^{18}F]FDG PET/TC. Por conseguinte, três conjuntos diferentes de dados foram utilizados para treinar redes neuronais distintas: imagens de TC para a segmentação do baço; imagens [^{18}F]FDG PET/TC com suspeitas de lesões malignas; e lesões sugestivas de linfoma em imagens PET de corpo inteiro. Posteriormente, uma arquitetura U-net em 3D foi desenvolvida e otimizada para a identificação e segmentação automática dos objetos de interesse. Devido às limitações da capacidade computacional da GPU, várias abordagens foram implementadas para o processo de treino da rede e para testes de inferência. O coeficiente de Dice (CD) foi utilizado como medida de sobreposição entre as segmentações. Deste modo, o primeiro conjunto de dados alcançou a mediana de CD mais elevada, 0.57, utilizando a rede treinada através do método de aprendizagem por transferência e com uma normalização de intensidade TC de [-250; 250]. O CD obtido no segundo conjunto de dados alcançou uma mediana de 0.28, quando a 3D U-net foi treinada com *patches* de 48×48×48 vóxeis. Finalmente, o terceiro conjunto de dados obteve uma mediana de CD de 0.41, onde o tamanho do *patch* foi de 64×64×64 vóxeis numa configuração da U-net com uma camada a menos. Concluindo, os métodos de segmentação totalmente automáticos baseados em técnicas de *deep learning* para a identificação e segmentação de lesões malignas continuam a necessitar de supervisão clínica para verificação e correção. Por enquanto, é inaceitável utilizar este método na prática clínica de uma forma independente, uma vez que não é suficientemente robusto.

Palavras-chave: [^{18}F]FDG PET/TC; Lesões Malignas; *Deep Learning*; Segmentação Totalmente Automática

Agradecimentos

E com este relatório finalizo os cinco anos da minha bela e ilustre jornada académica. Para tal, não posso deixar de homenagear todos aqueles que tanto contribuíram para o desfecho deste meu percurso.

Primeiramente, um honrado agradecimento à Fundação Champalimaud e ao Professor Doutor Durval C. Costa pela esplêndida oportunidade de elaborar um projeto que tanto me cativou e fascinou no grupo de Medicina Nuclear – Radiofarmacologia. É uma experiência que levarei para a vida.

Aos meus orientadores, Doutor Francisco Oliveira e Cláudia Constantino, um sincero obrigada. A vossa motivação, entusiasmo e disponibilidade constante esteve presente desde o primeiro dia. O vosso perseverante apoio, orientação e sábios conhecimentos transmitidos ao longo deste estágio tiveram um amplo impacto no sucesso deste projeto.

Agradeço aos meus amigos, antigos e novos, pelo desanuviar, diversão e incessante apoio. Nunca foi tão agradável escrever um relatório científico.

Um agradecimento especial ao meu companheiro de todas as horas, Tomás, pela sua eterna paciência, pelo seu esforço em tornar os meus dias melhores e pelas repetidas viagens Porto-Lisboa, de forma a poder acompanhar-me durante a elaboração deste projeto. Obrigada pelo amor, carinho e amizade. Foram, sem dúvida, essenciais durante esta aventura.

Por último, mas claro, não menos importante, um infindável obrigada à minha família, Cláudia, Manuel e Marta, pelo encorajamento, confiança e conselhos prudentes. É através dos vossos constantes incentivos e motivação que me tornaram a pessoa que sou hoje. Estarei eternamente grata pelo amor e companheirismo que sempre me proporcionaram.

A todos, um genuíno obrigada.

Contents

List of Figures

List of Tables

Abbreviations

[^{18}F]FDG	Fluorine-18-Fluorodeoxyglucose
Adam	Adaptive moment estimation
CAD	Computer Aided Diagnosis
CCC	Champalimaud Clinical Centre
CNN	Convolutional Neural Network
CT	Computed Tomography
CUDA	Compute Unified Device Architecture
DICOM	Digital Imaging and Communication in Medicine
DLBCL	Diffuse Large B Cell Lymphoma
DC	Dice Coefficient
FCN	Fully Convolutional Network
FN	False Negative
FP	False Positive
FWHM	Full Width at Half Maximum
GD	Gradient Descent
GPU	Graphics Processing Units
IQR	Interquartile Range
MICCAI	Medical Image Computing and Computer Assisted Intervention
MRI	Magnetic Resonance Imaging
MSD	Medical Segmentation Decathlon
NIFTI	Neuroimaging Informatics Technology Initiative
PET	Positron Emission Tomography
PPV	Positive Predictive Value
ROI	Region of Interest
SPECT	Single-Photon Emission Computed Tomography
STD	Standard Deviation
SUV	Standardized Uptake Value
TN	True Negative
TP	True Positive
VGG	Visual Geometry Group

Chapter 1 Introduction

1.1 Background and motivation

Cancer is the uncontrolled cell growth in certain regions of the body, being one of the prime causes of death worldwide, with a growing incidence in the population. According to the World Health Organization, in 2020, it was estimated 19.3 million cancer cases, causing approximately 10 million deaths, where metastases spread across the body were the paramount cause of cancer deaths. Over 40% of cancers could be prevented and mortality reduced through early detection and effective treatment. Hence, for an effective and less intensive treatment, it is vital to have a well-thought treatment plan via cancer diagnosis, staging, prognosis, and therapy response [1], [2].

In clinical practice, positron emission tomography/computed tomography (PET/CT) with Fluorine-18-Fluorodeoxyglucose ([^{18}F]FDG) has become a great multi-modal medical imaging modality for cancer early diagnosis, prognosis, and surveillance, since malignant lesions, and early tumours across the body are usually asymptomatic. As the name suggests, PET/CT is the combination of PET and CT medical imaging technologies, designed to resolve the issues associated with PET images, such as attenuation correction and lack of anatomical information [3]. This hybrid technique integrates PET functional and CT structural information, providing quantitative images and improving the accuracy of the anatomical localisation of the regions of interest (ROI) in the medical images, increasing the reliability of lesion detection and segmentation [4], [5]. In parallel, there is going to be a better understanding of the [^{18}F]FDG uptake, along with an attenuation correction based on the CT scan [6]–[8]. PET/CT is widely used in clinical oncology, especially in tumour diagnosis, staging, radiotherapy planning, and treatment follow-up [9].

Tumour segmentation plays a vital role in cancer, especially in radiotherapy and surgical planning, providing a precise location and an accurate boundary of the entire malignant lesion, to diminish the radiation exposure to the surrounding healthy tissues, and reduce the surgical resection of the tumour, respectively [4], [10]. Manual segmentation is the gold standard method. However, this method has major limitations, namely being time-consuming, labour intensive, tedious, and highly subjective, which introduces inter- and intra-operator variability, resulting in less precise and reproducible segmentation results. In addition, PET scans usually have low resolution and high noise, resulting in low-quality images, which affects the

segmentation process. Note that small lesions are difficult to segment due to the partial volume effect, which blurs the boundaries of the object of interest [4], [11].

As a consequence of the massive value of medical image segmentation in cancer, the need to automate this process is a critical aspect to consider. Hence, to overcome manual segmentation-related issues, deep learning has been taking control over the current semiautomatic segmentation methods' research, due to its promising results. In addition, it has been reported that simple traditional segmentation methods have been surpassed by the new deep learning-based methods [12], [13]. For example, thresholding is one of the most implemented methods in today's clinical practice, due to its simplicity and reduced computational costs. However, this technique displays a low performance on small lesions and is highly sensitive to the noise of PET images [4], [14].

1.2 Positron emission tomography using [^{18}F]FDG

PET is a non-invasive medical imaging technique, in the field of nuclear medicine, that provides functional information about the cellular metabolism of different organs, tissues, and abnormal structures. PET imaging uses radiopharmaceuticals, which are radiolabelled molecules, containing a radioisotope linked to a chemical vector. The most used radiopharmaceutical in PET is [^{18}F]FDG, which is a glucose analogue bonded to an ^{18}F radioisotope. [^{18}F]FDG is transported to the interior of the cells by the glucose transporters presented in the cellular membrane of the living organisms, and subsequently is phosphorylated by hexokinases, becoming retained inside the cells [4], [15], [16].

The ^{18}F radioisotope has great properties for nuclear imaging considering that comprises a reasonable short half-life (∼109 minutes) and decays by emitting a positron. This position is then annihilated by a surrounding electron, giving rise to two gamma photons of 511 keV each. ^{18}F has a high positron decay ratio, and low positron emission energy, resulting in PET images of good quality [4], [15].

Carcinogenic cells usually manifest an increased glucose consumption, due to the over-expression of facilitative glucose transporters across the cellular membrane, via concentration gradient and high hexokinase and other glycolytic enzyme activities. Growing tumours suffer metabolic changes and require a high amount of glucose as an energy source (glycolysis), absorbing and accumulating high levels of [^{18}F]FDG inside the cells, demonstrating a higher [^{18}F]FDG avidity compared to normal tissues of the organisms [17], [18]. [^{18}F]FDG PET

quantifies the [^{18}F]FDG concentration in the tissues, which is directly correlated to the glucose consumption or elimination. In addition, [^{18}F]FDG is not completely metabolised, and the ^{18}F radioisotope accumulates inside the cells until its radioactive decay [3], [4].

[^{18}F]FDG is not specifically accumulated in cancerous cells, since inflamed tissues also exhibit high [^{18}F]FDG avidity, as a result of the glycolytic activity of the macrophages, neutrophils, and lymphocytes [19]. Also, it will accumulate in healthy tissues with high glucose metabolism, particularly the active skeletal muscles, brain, myocardium, thyroid, and salivary glands. Despite that, [^{18}F]FDG uptake varies from patient to patient, where in some cases the uptake is negligible and in other cases moderate [20], [21]. In the urinary tract, the [^{18}F]FDG is not reabsorbed in the proximal tubules of the kidneys, accumulating this radiopharmaceutical in the urine, which is then eliminated. Thus, the urinary bladder, kidneys, and sometimes the ureters show [^{18}F]FDG accumulation, which makes the [^{18}F]FDG PET scan interpretation complicated, causing a problem in pelvic malignancies since some tumours are adjacent to these regions [22].

It is crucial to understand that not all malignant lesions have [^{18}F]FDG avidity. For instance, in prostate cancer, prostate-specific membrane antigen (PSMA) based radiopharmaceuticals are more sensitive and specific [23]. It is known that most lymphoma lesions are [^{18}F]FDG avid [24]. For this dissertation, only [^{18}F]FDG avid malignant lesions were taken into account.

1.3 State-of-the-art – Application of neural networks on medical image segmentation

Over the years, several researchers have attempted to tackle image segmentation problems. However, only recently it became possible to use deep learning as an image segmentation tool, due to the enhanced computational efficiency of graphics processing units (GPU) in line with the increased access to open-source software frameworks and the appearance of new optimisation methods. In addition, the growth of big data contributed to the access of larger and more variable datasets, leading to a better data representation, and hence, better algorithm performances [25].

Deep learning has been in high demand, outperforming current traditional methods in some image processing tasks, especially in image classification. This technique started to gain

attention at the medical image segmentation level, where convolutional neural networks (CNN) based architectures play an essential role in the process [14], [25].

The need to develop an automatic segmentation method for medical images is being severely discussed in the literature. Each year, hundreds of publications about deep neural networks applied to medical image segmentation are made [26].

The primordial concepts of deep learning started back in the 60s. In 1998, a network called LeNet5, developed by Lécun et al. [27], was introduced, which expanded the development of CNN. However, it was not until 2012 that, through the appearance of the AlexNet network proposed by Krizhevsky et al. [28], CNN have begun to revolutionise deep learning applied to computer vision. Due to computational restrictions and a low amount of labelled data, these were shallow networks, which means that these networks had a reduced number of layers [14].

In 2014, Simonyan and Zisserman [29] proposed a new network configuration designated as a visual geometry group (VGG) to improve the network performance by adding a pooling layer after the convolutional layer, increasing the network depth. However, problems of vanishing gradient and overfitting may arise with the increase of the network depth. To address these issues, in 2015, GoogLeNet was presented by Szegedy et al. [30]. As the network depth is increased, network performance may potentially improve. However, when the neural network reaches a certain number of layers, the network starts to slowly converge, since the network cannot improve its performance. Hence, a year later, ResNet was designed by He et al. [31], which allows the network to support more layers, increasing the network depth by adding residual blocks.

In 2015, a new deep neural network architecture more suitable for semantic segmentation was proposed, significantly improving the performance, as this configuration can assign a category to each pixel or voxel, instead of only the entire image. Hence, Long et al. [32] proposed the fully convolutional networks (FCN), that have been used in several different variants over the years.

Ronnerberger et al. [33] proposed U-net with an encoder-decoder structure, that allows the combination of semantic and spatial information by adding long skip connections. This network architecture completely revolutionised the medical image segmentation field since it outperformed the previous segmentation methods, being the most used in medical image segmentation. In addition, U-net allows training the network with a smaller dataset, which is opportune, because of the lack of segmented medical imaging datasets. This network is versatile, as it allows the development of new variants, always maintaining the U-net structure

core. Hence, Çiçek et al. [34] introduced the 3D U-net for volumetric images, making a great contribution to medical image segmentation, and increasing the spatial information provided in the network. Milletari et al. [35] modified the original U-net into a 3D V-net configuration by introducing residual blocks and replacing the loss functions with Dice loss.

Currently, multiple different network configurations and variants exist. However, it is difficult to establish a benchmark medical image segmentation method, due to the enormous variability intrinsic to the medical dataset, such as image acquisition and reconstruction protocols, equipment, imaging modalities, intensity heterogeneity, class imbalance, masks segmentation process, pathologies, and their location, morphology, and other biological and physiological aspects heterogeneity. Not to mention, the variability introduced in the deep neural network configuration, network optimisation process, and deep learning frameworks to implement the algorithms [36].

Hence, different deep learning network configurations have been proposed in the literature for malignant lesions detection and segmentation on [^{18}F]FDG PET/CT images in oncology. The implementation of 3D CNN-based methods demonstrated a higher performance at the medical image segmentation [12][13]. Furthermore, 2D CNN-based did not achieve good results in [^{18}F]FDG PET tumour segmentation [12].

In 2021, Isensee et al. [37] developed the nnU-net for biomedical image segmentation, validated on several different datasets of different organs. The particularity of this network is that it adjusts itself for any medical dataset automatically, including the image pre-processing, training process, and post-processing, outperforming the current networks. In the same year, Blanc-Durant et al. [38] applied the nnU-net to segment lymphoma lesions on [^{18}F]FDG PET/CT images, achieving a Dice coefficient (DC) of 0.73 ± 0.20 with a median of 0.79 on the internal validation set. In 2022, Jiang et al. [39] used the nnU-net to segment lymphoma lesions in [^{18}F]FDG PET images, obtaining a mean DC of 0.78 ± 0.25 and a median of 0.88 on the internal validation set. In addition, Siemens Healthineers developed an automatic software to detect both benign and malignant tumours in [^{18}F]FDG PET/CT images using a CNN-based architecture. This prototype is called PET assisted reporting system (PARS). This software obtained a median DC of 0.65 in an external validation dataset of patients with lymphoma [40]. However, in the literature, for clinical and research purposes, automatic segmentation algorithms for [^{18}F]FDG PET images predominate only for head and neck, and lung tumours [12] with promising results. However, there are still various obstacles to a standard, automatic and reproducible tumour segmentation method in [^{18}F]FDG PET/CT images using deep learning.

In the literature, there is a lack of whole-body malignant lesions segmentation deep learning algorithms trained in [^{18}F]FDG PET/CT images. Due to malignancies' heterogeneity in appearance and localisations (lesions may be spread across the entire body), their identification and segmentation are complex tasks, even for experienced clinicians. Thus, there is an urgent need to develop tools to aid nuclear medicine clinicians.

Nowadays, there is no clinical application of fully automatic segmentation methods. The methods are applied mainly in experimental terms and for research purposes, in which a few publications appear in the literature. Thus, current segmentation methods still need an experienced physician to verify and correct what the algorithm has segmented, making its supervision indispensable. It is important to make clear that automated segmentation methods are tools whose purpose is to minimise the time spent by the nuclear medicine physician during the identification and segmentation of the lesions.

1.4 Champalimaud Foundation

Champalimaud Foundation is a Portuguese private foundation mainly aimed at cancer and neurosciences research. An interesting particularity of this foundation is the direct connection between the biomedical research groups of Champalimaud Research and the Champalimaud Clinical Centre (CCC), ensuring that all knowledge and cutting-edge technologies are being applied, providing personalised medical practice of excellence. With the investigations being conducted, Champalimaud Foundation aims to prevent, early detect and provide effective treatments, to achieve leadership in scientific and technological innovation worldwide.

This internship was carried out at the Nuclear Medicine - Radiopharmacology Lab and clinical service, which is dedicated to developing new radiopharmaceuticals for oncology and neuropsychiatry, improving radiation dosimetry, and developing new algorithms to identify, classify and quantify pathologies in PET and single-photon emission computed tomography (SPECT) images, enhancing the process and analysis by their implementation into clinical practice.

Within the Nuclear Medicine – Radiopharmacology clinical service, every week, a high number of whole-body [^{18}F]FDG PET/CT exams are performed in patients with malignant or suspected malignant lesions. This creates the need to develop and improve automatic malignant lesions identification and segmentation methodologies.

1.5 Aim

The primary aim of the developed project of the internship at Champalimaud Foundation is to review, apply, and optimise deep learning-based techniques for the automatic segmentation of malignant lesions in whole-body [^{18}F]FDG PET/CT images.

1.6 Structure of the internship report

This dissertation contains seven chapters, whereas this first chapter comprises an overall survey about the background and motivation that led to this project, and the state-of-art regarding the applications of neural networks on medical image segmentation, particularly, on PET/CT images using [^{18}F]FDG. Chapter 2 covers the essential theory needed for the development of the fully automatic algorithm for malignant lesions segmentations, namely the comprehension of medical image segmentation, deep learning, the CNN architecture, and training mechanism. Chapter 3 describes the materials and methods used during this internship, which includes the datasets, the implemented medical image pre-processing and network optimisation strategies, and the optimised network evaluation methods, by overlapping the ground truth masks and the resulting segmentations. In chapter 4, all the results are presented. Afterward, the following chapters present the discussion, conclusions, and future work perspectives.

Chapter 2 Deep Learning on Medical Image Segmentation

2.1 Medical image segmentation

Medical imaging is a crucial tool in healthcare as it plays a massive role in this area. Medical imaging aims to assist radiologists and nuclear medicine physicians in early detection, diagnosis, and treatment planning, resulting in safer, more efficient, and reliable clinical analyses, while also playing a substantial role in the biomedical and cancer research areas [41], [42]. There are several imaging modalities, namely magnetic resonance imaging (MRI), ultrasound imaging, radiography, X-ray CT, electrocardiography, electroencephalography, and nuclear medicine imaging, particularly PET and SPECT [43]

Digital images are digital depictions of an image represented by a matrix composed of pixels, where each pixel is a discrete numeric representation of its grey intensity that ranges between zero and 255. Note that coloured images have three channels RGB (Red, Green, Blue), whereas grayscale images have only one channel. Digital images can be two-dimensional (2D) matrices where each unit is a pixel or can be three-dimensional (3D) in which those units are known as voxels. Voxel is the elementary unit of a volume in a digital image, usually with a parallelepiped shape, corresponding to scalar values. However, these volumes can be vector corresponding to the image channels.

What we usually call a medical image is not an image, it is a matrix of values in the space where each entry of the matrix represents a real value or vector. These values may have a physical meaning, as in CT or PET. However, it is important to point out that there is a high degree of variability between medical images, for instance, differences in image dimension, dimensionality, spatial resolution, intensity/range scale, noise, data representation, and acquisition modality [44].

One of the most important and challenging steps in medical image processing is image segmentation, which helps physicians analyse and understand medical images. The segmentation process consists of grouping similar pixels/voxels according to a criterion, for example, texture, colour, intensity, gradient, etc. Medical image segmentation divides an image into classes, through the detection of ROI and the precise delineation of a boundary in the objects of interest. Image segmentation can be split into semantic segmentation, wherein a label is assigned to each pixel/voxel in the image, or in instance segmentation, in which it identifies and delineates every object of interest as distinct. Medical image segmentation is considered

semantic segmentation. Furthermore, medical image segmentation allows the delineation of tumours, organs, lesions, and other abnormalities; and aids in radiation therapy by precisely estimating the measurements of the targeted anatomic regions (size, shape and location) and in treatment planning, by predicting and monitoring responses to therapy. Overall, this technique is extremely useful for computer-aided diagnosis (CAD) [4], [14], [45], [46].

There are several image segmentation techniques for [^{18}F]FDG PET/CT that can be divided into manual segmentation, semiautomatic, and fully automatic methods. Manual segmentation is the gold standard method for obtaining the ground truth, where a boundary is drawn by an expert physician around the object of interest. The ground truth represents the labels of the real boundary of the object of interest that should be identified. Still, as mentioned above, this method has several drawbacks, such as being highly subjective and having high inter- and intra- operator variability, resulting in decreased reproducibility. Semiautomatic methods can be implemented through a variety of algorithms, including threshold-, region-, gradient- and boundary-based. Although less human intervention is required, these methods still demand the user-mediated selection of certain parameters at any given point of the algorithm, such as an initialisation seed or the ROI definition, resulting in a degree of variability in the results obtained [4], [12]. Fully automatic segmentation methods do not need any operator involvement. Deep learning has been strongly requested in this process, as a result of recent developments in the computational and artificial intelligence fields. Besides, in literature, it has been reported that deep learning techniques, especially CNN-based, have achieved good results [12]–[14], [47].

Medical image segmentation faces some difficulties inherent to the type of images used. For instance, PET image segmentation can be affected by the low resolution and high noise inherent to the PET scans. On the other side, CT scans may have low contrast between tissue types and may suffer from artifacts. Image segmentation can also be affected by partial volume effects, motion artifacts and due to a wide variability in pathologies texture, shape, size, and location [4], [45]. Additionally, in the literature, there is no consensus about which segmentation method is the best for clinical practices, due to the absence of benchmark databases to compare the different algorithms [12].

2.2 Deep Learning

Machine learning is a subcategory of artificial intelligence dedicated to learning data and pattern recognition to make predictions in classification and analysis and help in decision-making and computer vision without nearly any human intervention. Machine learning can be divided into two classes: (1) shallow machine algorithms, including support vector machine, decision tree, *k*-nearest neighbour, logistic regression, linear classifier, naïve Bayes and artificial neural networks and (2) deep neural networks that resort to deep learning [48].

In many applications, machine learning is unable to process raw data and needs to have access to good feature representations. The manual design and extraction of the representational features may require experts in the field. Machine learning algorithms may have low interest when handcrafted features are needed to build complex models [49]–[51]. In many problems, features may be automatically extracted to feed the machine learning algorithm [52], [53]. Deep learning is capable of automatically learn representational features from raw data, extracting discriminative feature representations.

Deep learning is a branch of machine learning that mimics the mechanisms of the human nervous system, making it capable of learning and extracting complex features from larger datasets. Deep learning uses deep neural networks which are composed of multiple layers, with several processing units (also called neurons) in each layer, and they are connected to each other [14]. The features are extracted in a hierarchy, that is, from lower-level features to higher-level features, as the network depth (number of layers) increases.

Deep learning has a wide range of applications because of its multilayer layout which allows the execution of complex tasks, namely computer vision, for example, image classification, object detection and semantic segmentation; audio and speech processing; self-driving cars; biomedical research, and healthcare [54]. In the healthcare field, deep learning may have multiple applications, including CAD, accurate predictions of medical conditions, personalised treatment recommendations, support for clinical decision-making, disease monitoring, and, especially, medical imaging analysis [55].

Yet, training a deep neural network is computationally intensive, time-consuming and requires sophisticated equipment. The lack of benchmark, extensive and reproducible datasets for medical imaging is one of the greatest issues in deep learning. The under-represented medical conditions in the datasets result in unbalanced data, leading to a decreased optimised network accuracy [54], [55].

As for other methods, deep learning-based methods can be classified as supervised or unsupervised learning. In the case of supervised learning, both datasets need to have labels paired with the medical images, that is their corresponding ground truths that were previously obtained. This process tends to be more accurate than unsupervised learning and it is used for image classification and object detection, to predict an output label. For image segmentation, since the labels of the image pixels/voxels are known, the error rate is reduced in the network. Unsupervised learning enables the learning process to be implemented without available labelled data. There is also a process called semi-supervised learning, whereby it is provided semi-labelled input datasets, to reduce the amount of labelled data [56], [57].

Various types of deep learning network architectures can be implemented, among which stand out CNN, recurrent neural networks, recursive neural networks, and deep generative networks, such as deep belief network, deep Boltzmann machine, generative adversarial network, and variational autoencoder. It is important to note that these architectures have different variants, depending on the application, the medical image modality, or to overcome certain challenges that might arise [54].

In virtue of deep learning's success, this technique has begun to be applied in medical imaging, particularly in image segmentation. For medical image segmentation, it has been reported that CNN-based is the most popular architecture, due to its excellent results and outstanding feature extraction capability [25], [54].

2.3 Convolutional neural networks

CNN, also called ConvNet, is a neural network architecture that is widely used in computer vision, especially for image processing, due to its outstanding representative feature extraction and pattern recognition capabilities as well as its impressive performance and high accuracy in several competitions. In healthcare, these networks are used in medical image classification, detection, localisation, and semantic segmentation (pixel-level classification). CNN are neural networks where one layer is convolutional, at least. The most popular CNN-based architectures are AlexNet, LeNet, ResNet, VGG, GoogLeNet, SegNet and FCNs [14], [58].

The dimensionality of the input images should be taken into account in the moment of choice of the CNN architecture. Thus, there are 2D CNN that receives 2D images as input; 3D CNN which receives volumetric images; and, lastly, 2.5D CNN which receives 2D slices from

volumetric images. 2.5D CNN has a better spatial representation than 2D CNN, because it uses image slices from different spatial orientations, and requires less computational costs than three-dimensional ones. However, 3D CNN extracts more powerful volumetric representation along the three spatial orientations, giving more spatial information and inter-slice information, that are essential in 3D imaging modalities, such as MRI, CT, and PET [51].

2.3.1 Convolutional neural network architecture

CNN architecture consists of an input layer, hidden layers, and an output layer. The hidden layers contain one or more convolution layers. Frequently they also contain activation function layers, pooling layers, normalisation layers, and fully connected Layers. Neural network architectures can vary in terms of the number of layers and neurons and even in the convolution and pooling operations, and activation functions. By increasing the number of hidden layers of the network, its depth increases as well, which may improve the network's performance if adequate data exist. For a clearer understanding of a general CNN architecture, it was elaborated a scheme presented in Figure 2.1.

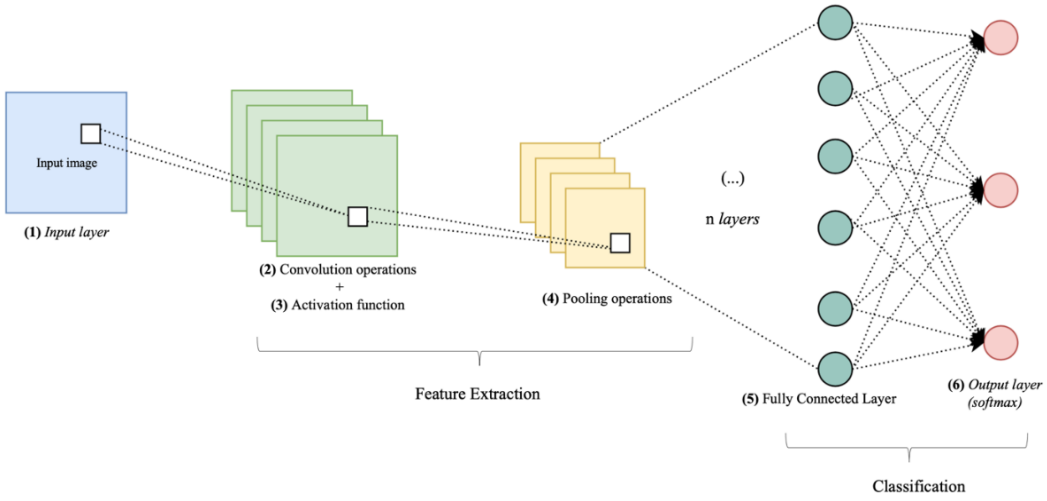

Figure 2.1: General 2D CNN architecture.

(1) Input Layer

The input layer is the place at which the input images enter the neural network, and it is represented by a matrix of pixels with the same dimensions as the input image. The input image may have one channel or more (three for colour images, for instance). The channels may also be image normalisations or even images acquired in a different modality, such as PET and CT images [26], [57], [59].

(2) Convolutional Layer

At the convolution layers are applied convolutional filters, also denominated as kernels, to extract features from the input image. Kernels are usually square matrices, that can be applied in different manners, and they are smaller than the input image. Each value in the matrix is a kernel weight that is going to be learned during the training process, which will be explained further on. Depending on the network architecture, kernels can be 2D or 3D. Mathematically, convolutions calculate the dot product between the corresponding elements of the image and the kernels, and then sum up all values, resulting in a scalar value. A convolution operation extract features, obtaining a single feature or activation map. Each kernel originates a channel in the feature space [26], [57], [59].

Figure 2.2 portrays how convolution operations work in image processing. In this process, a kernel slides over the entire image with a pre-defined stride (step size along the axes), both horizontally and vertically, computing the pixel values, and resulting in a feature map. The operation ends when the kernel can no longer slide over the image. Two phenomena occur in convolutional operations that help to reduce computational costs and memory usage: (a) sparse connectivity, where only a percentage of all connections are made; (b) shared parameters, where each input, learns the same weights, decreasing the number of parameters being learned [26], [57], [59].

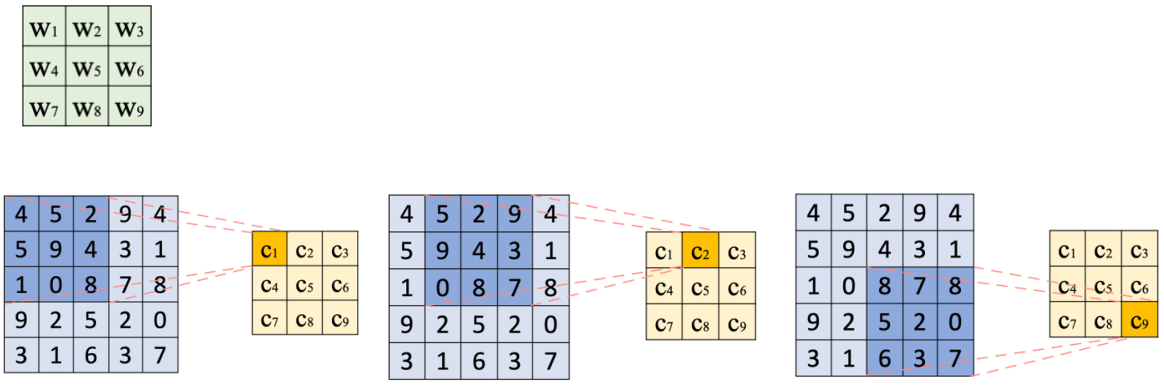

Figure 2.2: Convolution operation steps with a 5×5 input feature map, a 3×3 kernel sliding with a stride of one on 2D grids, resulting in a 3×3 output feature map.

(3) Activation Layer

It is common practice to apply an activation function after each convolution, to introduce non-linearity to the system, allowing the CNN to learn more complex features. The activation functions can be a sigmoid function, a rectified linear unit (ReLU), or a hyperbolic tangent

(Tanh). However, ReLU (equation 2.3.1.1) is the most used, because gets a faster convergence and requires less computation [26], [57], [59].

$$f(x) = \max(0, x) \tag{2.3.1.1}$$

(4) Pooling Layer

Normally, pooling layers come after convolution layers to reduce the dimension of the feature maps and preserve the most relevant information in the next layers, consequently reducing the computational complexity. The most common pooling operations are max-pooling and average-pooling. Both activation and pooling layers do not learn during the training process [26], [57], [59]. Figure 2.3 demonstrates a 2×2 max-pooling operation example and its resulting 2×2 output feature map.

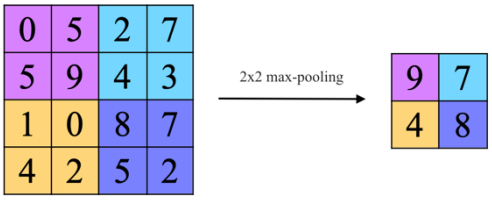

Figure 2.3: 2×2 max-pooling operation of a 4×4 input feature map with a stride of 2, resulting in a 2×2 output feature map.

(5) *Fully Connected Layer*

A fully connected layer is found at the end of the network, where the neurons of the previous layer are fully connected. Feature maps from the preceding layers are reorganised into a vector that is fed to this layer, which acts as a classifier [26], [57], [59].

(6) Output Layer

In the Output layer, a *SoftMax* function (equation 2.3.1.2) is applied for each class (background and foreground for binary segmentation), obtaining a probability distribution, which provides classification scores, representing the probability to belong to each class. The *SoftMax* function transforms the feature map vectors into a probability (values between [0,1]). Then, classification (or, in the case of segmentation, pixel-level classification) occurs, where each probability value is associated with each class, resulting in a predicted output [26], [57], [59].

$$y(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{k} e^{x_j}} \qquad (2.3.1.2)$$

Where x is the input vector with k number of classes in the classifier and x_i is the elements of the input to the *SoftMax* function. The bottom term of the equation is the normalisation, obtaining a probability distribution.

2.3.2 Optimisation and network evaluation process

To obtain an optimised CNN, it is necessary to go through an iterative training process that consists of learning weights, which helps the trained network to make accurate predictions. In the first stage, it is essential to define the hyperparameters, which include the kernel's size and type, the number of kernels being applied in each convolution, the stride, the size and type of kernel in the pooling layers, the optimiser, the loss function, the number of epochs and batches, the learning rate, and the training, internal validation and test sets. For a better understanding of the concepts, an epoch refers to the number of times the algorithm runs through the whole training set; the batch size denotes the number of samples from the training set used in each iteration by the optimiser; and the learning rate provides the step size of the weights and biases updates [60].

In supervised learning, the datasets must consist of the images and their corresponding ground truth labels. The training set constitutes the data that is going to be used to train the network, making it possible to make accurate predictions. The internal validation set is an independent set and simulates how the optimised network behaves to unseen data, making it an excellent criterion for selecting the best trained network. If the values between the train and internal validation sets are approximate, that means the trained network is stable and has converged. This set should not be used to fine-tune the network, only to check whether it shows signs of overfitting or not. However, the internal validation set is essential for the optimisation of the hyperparameters, improving the network indirectly. The test set is an external unseen data (never used in the optimisation process) that is used for an independent trained network's performance evaluation, that should only be used once [60]. In Figure 2.4, the dataset division is summarised.

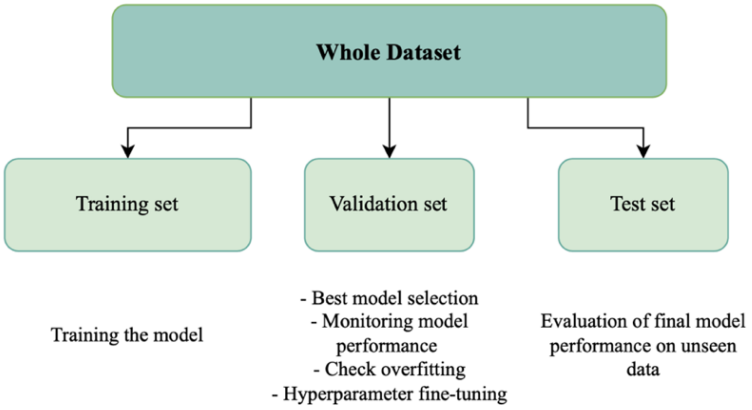

Figure 2.4: Dataset division on training set, internal validation set and testing set (adapted from [60]).

After the definition of the hyperparameters, the optimisation starts. The images enter the network via an input layer; and the learnable parameters (weights and biases) are randomly initialised. Parameters initialisation determines how fast is the network going to converge. During the training process, these intern parameters are going to be adjusted by the optimiser, until a stop criterion is reached. The selection of the hyperparameters is an important step since it will affect the performance of the learnable parameters' optimisation.

The loss function calculates the network error that is related to the difference between the predicted results and the ground truth labels of the training set. Thus, this loss function is characteristic of supervised learning. The aim is to minimise the loss function and, consequently, minimise the network error. Simply, loss functions measure how well the network is learning from the training data. This method will help choose the parameters to improve the automatic algorithm. The loss function affects the learning process when building a deep neural network, including network speed convergence to optimal values and network accuracy. For medical image segmentation, the most used loss functions are cross-entropy and Dice loss [60], [61].

The adjustment of the learnable parameters is accomplished via the backpropagation algorithm allied to the optimisation function. In CNN, the backpropagation algorithm is a method integrated into gradient-based optimisation methods, that allows the intern parameters to be continuously fine-tuned, to enhance the neural network performance by calculating the partial derivatives of the loss function relative to each weight. The term backpropagation comes from the fact that the gradient is calculated, and weights adjusted in reverse order of the network

configuration, from the output layer to the input layer. This method helps accelerate weight optimisation [60], [62], [63].

The optimisation process updates the weights and biases iteratively, in the direction that minimises the loss function based on the gradient calculated in the backpropagation algorithm. This process is repeated until the overall error is minimised, which means in the best scenario the trained network can make accurate predictions [64], [65].

For CNN, the gradient descent (GD) is the most common optimiser and has different configurations depending on the approach, which include the stochastic GD, adaptive GD, adaptive delta, root mean square propagation, adaptive moment estimation (Adam) and many others. The main difference among these gradient-based optimisation methods is the learning rate adjustment [62].

Optimisers need to provide a fast convergence rate of the network weights since there are many parameters to learn and optimise during the training process. The optimiser is sensitive to the learning rate because it affects the convergence of the network. If the learning rate is too high, divergence might occur, or the network is not able to reach the minimum value, because the step will jump over the optimum value. Whereas a low learning rate may lead to slow convergence, or it might get stuck to a local minimum. It is desirable to start with a higher learning rate and, throughout the optimisation method, gradually lower this hyperparameter. Therefore, the selection of the learning rate is a crucial step of the training process [62], [63].

After several epochs, ideally, the optimised network reaches a good performance and is evaluated through different performance evaluation metrics. For an independent evaluation, a set of unseen data, called the test set, is used to make predictions and final optimised network evaluation [55].

It is critical to notice that overfitting phenomena can occur during the training process. At a certain point, during the learning process, as the number of epochs increases, and the training loss decreases. This may happen when the network learns too many details from the training set, including the noise present in the training data. Here, it is said that the network became overfitted to the training data and it is difficult to perform well on internal validation data. Underfitting may also occur, where the networks do not learn enough from the training data. To evaluate the network overfitting, the learning curve that plots the training and internal validation loss over the number of epochs should be analysed [57], [60].

Usually, the learning curve of a good fit network initially demonstrates the decrease of the training and internal validation losses, followed by the convergence of the curves to the minimum loss value. In addition, the training and internal validation losses must be proximate

to each other. However, regarding overfitting, internal validation loss is much higher than training loss [60]. Figure 2.5 shows an example of overfitting during the learning process.

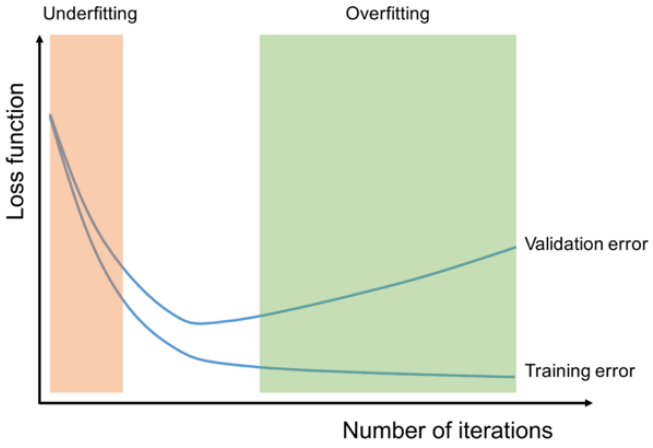

Figure 2.5: Learning curve with training and internal validation loss plots over the number of epochs with overfitting (adapted from [60]).

There are various techniques to overcome network overfitting, including adding more samples to the training set, data augmentation strategies, batch normalisation, early stopping, weight decay, and dropout techniques [57], [60].

Improving the accuracy and reducing the training time of the CNN is desired. CNN performance can be improved, in theory, by increasing the network depth (number of layers), which makes it possible to extract more complex features from more complex data; by adjusting the hyperparameters for the network optimisation process; increasing training time; and using a representational and vast dataset. Additionally, there are other variants of the ReLU activation function and different loss functions that can be applied, to enhance the network performance [51]. However, adding more layers to the neural network can result in an increased training time, and, sometimes, since the optimisation process becomes unstable as the number of learnable parameters increase, the final result can be worse. Therefore, the network depth should be handled with caution.

Nevertheless, the optimisation process of the network requires high computation power, high memory GPU, large amounts of data, and a long training time. CNN are usually described as black boxes, since for the user it is only known exactly what happens at the beginning and at the end, unknowing what happens in the hidden layers.

2.3.3 U-net

U-net was proposed by Ronneberger et al. [33] and, according to the literature, it is one of the most suitable CNN configurations for medical image segmentation. U-net has been widely used for this purpose, due to its outstanding performance. A U-net configuration example is illustrated in Figure 2.6 [14], [66]:

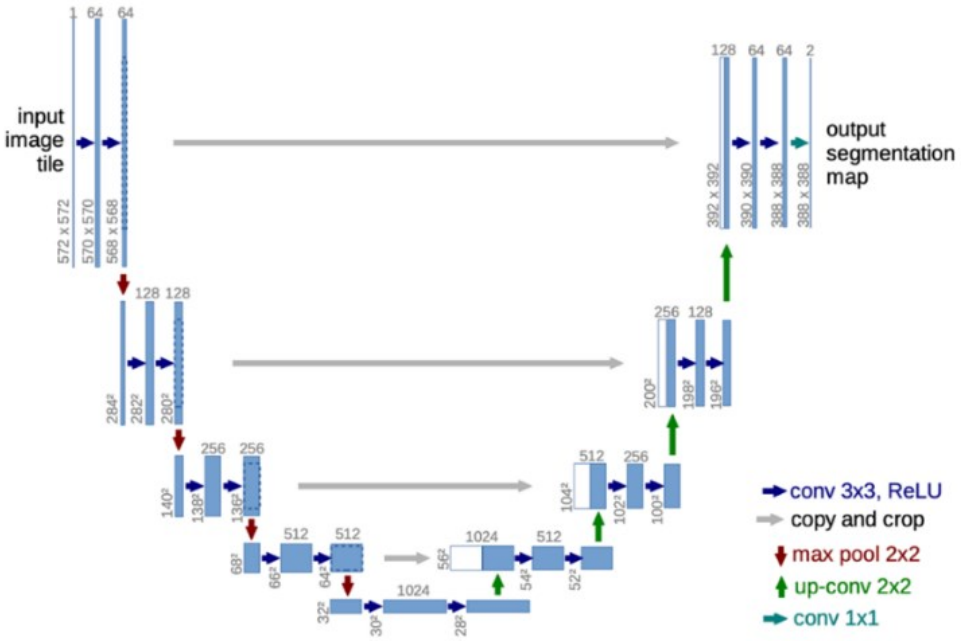

Figure 2.6: U-net configuration proposed by Ronneberger et al. [33]. The contracting path is on the left side and the expansion path is on the right side.

U-net is constituted by a contracting or downsampling path and an expansion or upsampling path. Both paths are symmetric, making a U-shaped architecture.

The contracting path has a similar structure to a standard CNN, and it is divided into several encoder blocks, where the first block has an input layer that receives an image. Each encoder block has two 3×3 convolutions, followed by their corresponding ReLU layers and a 2×2 max pooling operation, to downsample the feature maps. In Figure 2.6, it is noticeable that the size of the image decreases after each convolution, as they are unpadded convolutions.

In this path, the number of convolution filters is doubled as well as the number of feature channels, while the spatial dimensions (height and width) of the feature maps are cut in half in each layer. The increased number of filters allows the network to learn more complex features.

The expansion path is constituted by multiple decoder blocks, where the feature maps suffer upsampling by a 2×2 convolution layer, also known as transposed convolution, that halves the number of convolution kernels and feature channels, while double the spatial dimensions of the feature maps in each layer.

An interesting particularity of the U-net architecture is that it extracts both intensity and spatial information from the images, allowing a finer segmentation. This property comes from the concatenation process. At each level, the feature map from the contracting path is cropped to match the same spatial dimensions as the feature map from the expansion path. So, the high-resolution feature maps from the encoder are concatenated with the feature maps from the decoder via skip connections, receiving semantic information. Concatenation aims to propagate contextual information, preserve loss from the previous layers, and allow the network to learn better representation and improve segmentation accuracy [67].

Subsequently, it is applied two 3×3 convolutions followed by ReLU layers and a 2×2 transpose convolution. The expansion path allows the neural network to learn localised information and increase the resolution of the output.

At the final layer, a 1×1 convolution is applied, to reduce the number of channels to two (for two-classes segmentation). The spatial dimensions are restored, making a prediction for each pixel of the input image, and mapping the channels to the desired number of classes (foreground and background). The result is a segmented image.

U-net has several advantages compared to simple CNN, in particular, the ability to extract detailed features with smaller training data, which is very important for medical images, since the datasets and their labels are limited; localisation and contextual information are combined, due to the network structure; preserving the structural integrity of the input image; reduced information loss across the network; and the high architecture mutability [11], [66].

Nonetheless, the U-net architecture has a few drawbacks, among the vanishing gradient problem, the difficulty of training highly deep networks, the skip connections duplicate low-resolution contents, and high GPU memory consumption. Thus, over the years, U-net has evolved and undergone several improvements, resulting in distinct variants to overcome the problems and improve accuracy, namely inception U-net, residual U-net, dense U-net, U-net++, nnU-net [11], [66].

U-net architecture can also be 3D, being capable of processing volumetric images, where all the previous operations and filters are now volumetric. This method is useful for medical image segmentation since a great amount of medical data is acquired in CT, PET, and MRI

tomographs. For volumetric input images, the 3D U-net has performed better than using a 2D U-net with 2D medical image slices [34], [66].

For this project, 3D U-Net architecture was chosen because of its state-of-the-art success performance in medical image segmentation. Blanc-Durand et al. [38] demonstrated the huge potential of this architecture on [^{18}F]FDG PET/CT image for lesion segmentation.

Chapter 3 Materials and Methods

3.1 Dataset

Three separate datasets were used for this study. The first dataset includes 3D spleen CT scans, the second consists of whole-body [^{18}F]FDG PET/CT scans of patients with malignant lesions, and the third dataset comprises whole-body [^{18}F]FDG PET scans of patients with lymphomas.

3.1.1 Dataset 1 – Spleen CT scans

One of the prime objectives of this project was to learn and acquire technical skills and sensitivity on the deep neural network's domain. For that reason, a proof of concept was conducted with a first dataset consisting of 3D spleen CT scans.

The segmentation of malignant lesions in [^{18}F]FDG PET scans is a complex task. Hence, to improve technical skills on this matter, the segmentation of an organ located in the abdominal area of the body in high resolution CT medical images was first studied. This is a simpler network train task since it has a low degree of heterogeneity. In addition, due to the growing access to public datasets, especially for CT medical images, with decent amounts of good quality medical data, it is possible to obtain acceptable results. Thus, through this first approach, it was possible to learn and put in practice the concepts and techniques related to the pre- and post-processing of tomographic medical images. In parallel, it was possible to develop and test neural network architectures along with testing different network's configurations to achieve a better trained network performance.

As a future application, the segmentation of organs in low energy CT scans is of great importance for the Radiopharmacology research group of Champalimaud Foundation, since that for every PET acquisition at the CCC, a low energy CT acquisition is also performed for PET attenuation correction and anatomical referencing. This method can assist the identification and segmentation of lesions within PET images.

The spleen volumes were downloaded from an open-source medical image data base, the medical segmentation decathlon (MSD). The downloaded dataset has 41 CT scans of oncological patients undergoing chemotherapy for treatment of liver metastases provided by

the Memorial Sloan Kettering Cancer Center of New York. The CT scans were acquired in the portal venous phase protocol with an intravenous injection of iodinated contrast agent. The patient's characteristics were not provided. The images were publicly de-identified and into the neuroimaging informatics technology initiative (NIFTI) format. Every volume had a corresponding ground truth label of the spleen that was acquired through a semi-automatic segmentation method. However, the segmentations were manually adjusted by an experienced radiologist [68], [69].

3.1.2 Dataset 2 – Whole-Body [^{18}F]FDG PET/CT malignant lesions

As mentioned above, there is a pressing need of develop and optimise identification and segmentation of malignant lesions in [^{18}F]FDG PET/CT whole-body scans in literature. Therefore, a second dataset was used, and the knowledge acquired within the previous dataset applied into a more complex task with [^{18}F]FDG PET images. The feasibility of automatic detection and segmentation of malignant lesions in whole-body [^{18}F]FDG PET scans was assessed. As [^{18}F]FDG PET images have a lower spatial resolution and anatomical information along with the high heterogeneity level of the malignancies, it was imperative to gain sensitivity with these sorts of images.

The second dataset used comprised 66 oncological patients, who underwent whole-body [^{18}F]FDG PET/CT acquisitions, that were obtained with the Philips Gemini TF scanner, at the Nuclear Medicine department of the CCC. All patients gave written informed consent for the use of their anonymised data for research and pedagogical purposes [70], [71]. Patient's characteristics are summarized in Table 3.1.

Table 3.1: Patient's characteristics from the final dataset.

Patient's characteristics	
Sex (M/F)	30/36
Age (years)*	66 ± 12
Height (cm)*	166 ± 10
Weight (kg)*	72 ± 16

*Data presented as mean ± standard deviation (STD).

This dataset contains 340 lesions in the whole body. Every patient is assigned to a suspected primary tumour location, which is why the patient performed the PET scan. Thus, this dataset includes several pathologies, which are mainly lesions from patients with breast, lung and lymphoma primary cancer. Note that not all segmented lesions are tumours, some are only lesions of suspect malignancy. Table 3.2 resumes the number of patients for each possible pathology.

Table 3.2: Number of patients of each suspected primary tumour.

Suspected primary tumour	**Number of patients**
Lung	14
Breast	13
Non-Hodgkin's lymphoma	9
Hodgkin's lymphoma	8
Colon	5
Pancreas	4
Endometrium	3
Other*	3
Oesophagus	1
Rectum	2
Tongue	1
Kidneys	1
Urothelium (urothelial cells)	1
Gallbladder	1

*Suspect tumour of unknown primary origin

For all patients, it was administrated an intravenous injection of 244 ± 54 MBq of [^{18}F]FDG. The standard clinical reconstruction protocol was followed to obtain the images. The parameters used for image reconstruction is described in Table 3.3. All data were de-identified, to guarantee the patient's confidentiality. The DICOM (digital imaging and communications in medicine) files of the PET and CT images were converted to NIFTI format. The PET images were normalised, by converting the values into standardised uptake value (SUV) [70], [71].

Table 3.3: Reconstruction parameters for PET images acquired in the PET/CT scanner [70].

Reconstruction parameter	Standard clinical protocol
PET	
Voxel size	$4 \times 4 \times 4$ mm^3
3D algorithm	OSEM*
Number of iterations × subsets	3 × 33
Relaxation factor	0.7
Post-reconstruction filter	None
CT	
Attenuation correction	Yes

*Ordered subset expectation maximization (OSEM)

To obtain the ground truth labels, all lesions were identified and annotated by experienced nuclear medicine physicians, that delineated a 3D ROI around the lesions and its surrounding background tissue. This task was performed in 3D Slicer 4.11 software (https://www.slicer.org). Then, a segmentation algorithm based on a Bayesian classifier was applied to the ROI to segment each lesion accurately [71]. Figure 3.1 illustrates an example of a lesion identification and its semi-automatic segmentation in the lungs.

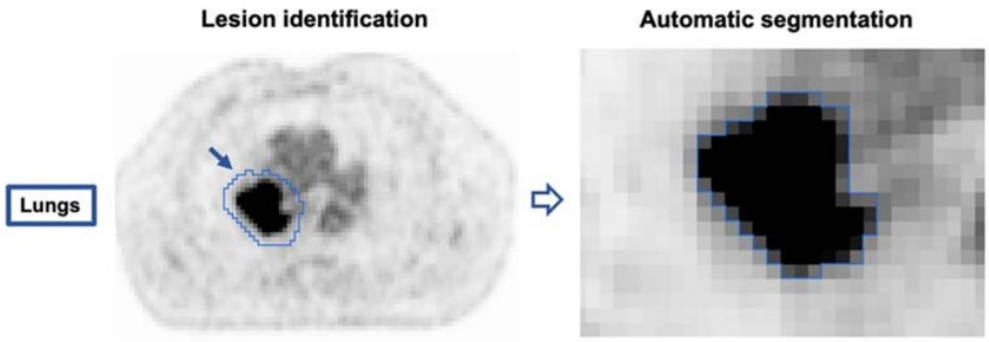

Figure 3.1: Example of a lesion identified in the lungs (left) and the corresponding ground truth label (right) achieved with semi-automatic segmentation (adapted from [70]).

For each [^{18}F]FDG PET scan, it was created a mask that contained every segmented lesion, saved in NIFTI format. In Figure 3.2 it is illustrated an example of a [^{18}F]FDG PET scan

viewed in 3D Slicer 4.11 software (https://www.slicer.org) that was included in this dataset [70].

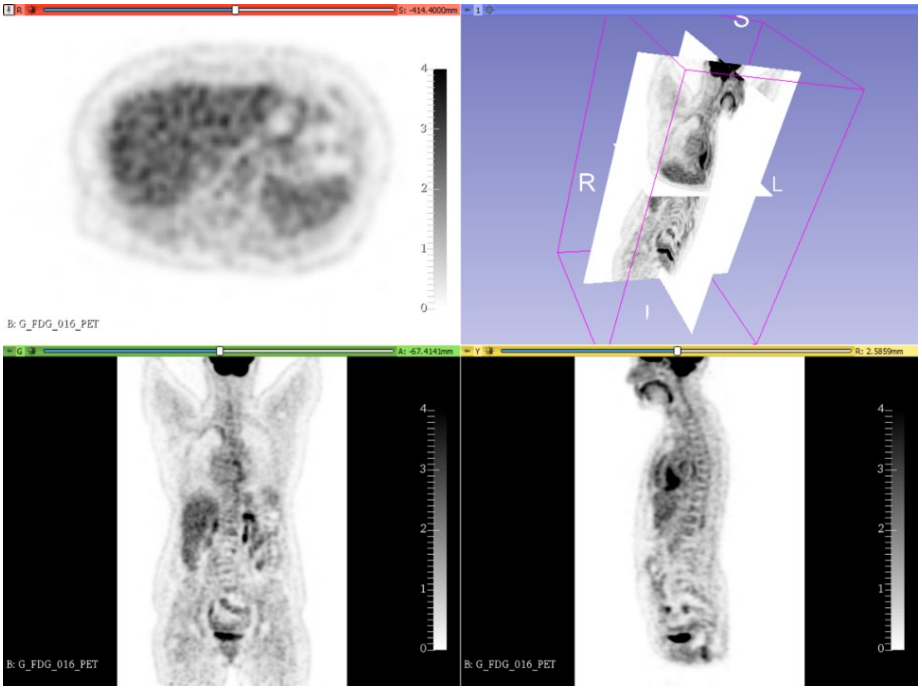

Figure 3.2: Representation of axial (top left), coronal (bottom left), and sagittal (bottom right) slices, and a 3D representation of all three slices from a [^{18}F]FDG PET. Data are from a patient included in this dataset. The 3D Slicer 4.11 software (https://www.slicer.org) was used, and it was selected the colour map InvertedGrey with an intensity range of [0 - 4] SUV.

3.1.3 Dataset 3 – Whole-body [^{18}F]FDG PET/CT lymphoma

The final aim was to develop and apply a U-net architecture on lesion segmentation, particularly on [^{18}F]FDG PET images from patients diagnosed with lymphoma. Due to the high variability in radiotracer uptake and lesions distribution across the whole body depending on the pathology in question, it is required that each deep neural network must be optimised for each type of tumour in cause. In the case of lymphomas, they have a large heterogeneity amongst patients, since lymphoma lesions can be spread across the entire body with different localisations, shapes and sizes, significantly affecting the segmentation process. In addition, in clinical practice, the images are obtained in different equipment, being essential that the trained network has a good identification and segmentation performance regardless the image source.

Therefore, the knowledge and skills acquired from the dataset two were applied in this final dataset.

The third dataset was taken from the "autoPET MICCAI 2022 competition" for automated tumour lesion segmentation on whole-body [^{18}F]FDG PET/CT. The dataset provided by the medical image computing and computer assisted intervention (MICCAI) was select since it had a reasonable amount of annotated medical data. This dataset consists of 900 patients with half being malignant melanoma, lymphoma and cell lung cancer and the other half being negative control patients, that do not have any signs of malignant lesions. The data was collected at the University Hospital Tübingen with a Siemens Biograph mCT PET/CT scanner, following their standard clinical protocol to obtain the volumetric images. It was administrated an intravenous injection of 300-350 MBq of [^{18}F]FDG. The reconstruction parameters used are described in Table 3.4. All data was de-identified, the DICOM files were converted to NIFTI files, and the PET images were converted to SUV. To obtain the ground truth labels, an experienced radiologist identified the [^{18}F]FDG-avid lesions and then manually segmented the previously identified lesions slice-per-slice. All ground truth segmentations were binarized [72].

Table 3.4: Reconstruction parameters for [^{18}F]FDG PET/CT images obtained with the Siemens Biograph mCT scanner [72].

Reconstruction parameters	**University Hospital Tübingen**
PET	
[^{18}F]FDG intravenous injection	300 - 350 MBq
3D algorithm	OSEM
Number of iterations × subsets	2 × 21
Post-reconstruction filter (FWHM)*	Yes
CT	
Slice thickness on contrast-enhanced	2 - 3 mm

*Full width at half maximum (FWHM)

For this project, onlzy the lymphoma patients were used, meaning that only 144 patients were used. The Table 3.5 describes the patient's characteristics.

Table 3.5: Patient's characteristics of the final train and internal validation sets.

Patient's characteristics	
Sex (M/F)	76/68
Age (years)*	46 ± 19
Diagnosis	Lymphoma

*Data presented as mean ± STD.

Plus, a CCC lymphoma patients' dataset was also used. A dataset comprising 65 oncological patients (28 men and 37 women) diagnosed with lymphoma, with a mean age of 64 years old, was used. The different lymphoma types are described in Table 3.6. The [^{18}F]FDG PET scans were obtained with the Philips Gemini TF scanner, at the Nuclear Medicine department of the CCC. Once again, all patients gave written informed consent for research and pedagogical purposes. It was administrated an intravenous injection of 240 ± 47 MBq of [^{18}F]FDG and the image reconstruction was performed in accordance with the standard clinical protocol parameters of the Philips Gemini PET/CT scanner, described in Table 3.3. The [^{18}F]FDG PET images were normalised into SUV.

Table 3.6: Lymphoma types of patient's population of the test set.

Lymphoma type	
Hodgkin	16
Non-Hodgkin	29
DLBCL*	19
Low-grade non-Hodgkin	1

*Diffuse large B cell lymphoma (DLBCL)

All lymphoma lesions were identified by experienced nuclear medicine physicians and manually segmented by one observer in 3D Slicer 4.11 software (https://www.slicer.org). Figure 3.3 displays the lesions spread throughout the entire body contained in this dataset [73]. So far, this dataset included patients with lesions located in head and neck, thorax, abdomen and pelvis.

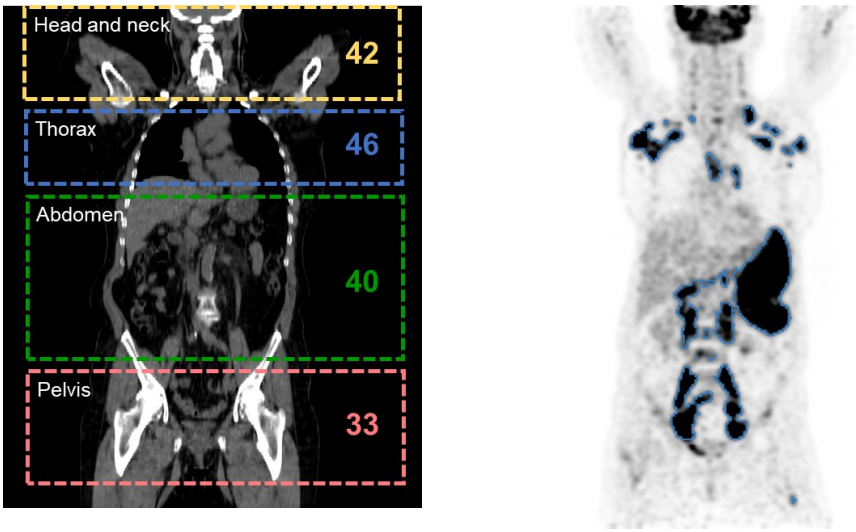

Figure 3.3: Layout of the number of patients with lesions localised in head and neck, thorax, abdomen, and/or pelvis represented in a CT (left) and representation of a coronal slice from a [^{18}F]FDG PET of patient included in this dataset with lesions spread across the body (right). The 3D Slicer 4.11 software (https://www.slicer.org) was used, and it was selected the colour map InvertedGrey (adapted from [73]).

3.2 Implemented strategies to create optimised networks

For each dataset, different network configurations were implemented, as well as the selection of hyperparameters and medical image pre-processing techniques, to study and ascertain the best method to use to obtain an optimised neural network, capable of performing well in the segmentation process. Hence, various step-by-step strategies were implemented, whereas for each strategy, the neural network training process was repeated five times, for reproducibility. For the third dataset, the training process was only carried out once since the training time was extremely long. Note that each strategy has considered the limited hardware resources and the small datasets with available ground truth masks.

3.2.1 Dataset 1 – Spleen CT scans

Network architecture and implementation

As mentioned above, for this project, U-net was chosen due to its popularity amongst the medical image segmentation neural networks available in literature [51], [66]. In particular, the 3D U-net is being widely used for semantic segmentation [34], [74]. The U-net configuration

proposed for this project was based on the original U-net architectures of Ronneberger et al. [33] and Çiçek et al. [34].

The 3D U-net has a total of 56 layers and is illustrated in Figure 3.4. The encoder part of the network consists of two 3D convolutional layers with volumetrics kernels with sizes of 3×3×3 moving in strides of one, followed by two ReLU layers.

Each kernel has a set of weights, where the number of weights is the product between the height, width, depth and number of channels of the filter. For this dataset, for spleen segmentation, the maximum number of weights being learned was 1415680. The total number of parameters being learned throughout the network's layers can be consulted on appendix A (Table A1). Along the training process, after each convolution, feature maps are generated, where the number of feature maps is the same as the number of filters. Simultaneously, after each convolution, the output has the same size as the input, due to addition zero-padding. In the kernels, the weights are randomly initialised from a normal distribution with zero mean and 0.01 STD, while the bias are initialised with zeros.

A batch normalisation layer is added between the convolutional and ReLU layers, to normalise the input values, so they have normal distribution, zero mean and unit variance, helping in the training process. Then a max pooling layer with a 2×2×2 filter, reduces the feature map to half its size. The network starts with 64 filters, doubling the number of filters in each dowsampling process.

The decoder contains two 3D convolutional layers with 3×3×3 kernels, followed by two ReLU layers and a final 3D transposed convolutional layer with a 2×2×2 filter to upsample the feature maps. At the final decoder block, there is a 1×1×1 convolution with two filters, *SoftMax* layer and a dicePixelClassification layer for classification.

The *SoftMax* layer applies a *SoftMax* function, that normalises the feature map values into a probability distribution between zero and one [75]. This process will assign a probability to each voxel to, consequently, classify each voxel to one class at the classification layer. Note that when using this function, the sum of all probabilities will be one.

The classification layer used was the dicePixelClassification layer that is used to classify each voxel into a categorical label (background or foreground) in semantic segmentation using the generalised Dice loss. Class imbalance is a problem amongst real world medical datasets, derived from the unequal distribution of the voxel classes, where there is a high percentage of voxels belonging to the background class (entire body tissue) compared to the foreground class voxels (organ or tumour). This issue leads to the misclassification of the under-represented class, typically the foreground class, whereas the most frequent class (background class) is

correctly classified. In oncology, class imbalance has a greater impact in the segmentation process since some tumours are composed by only a few voxels [76], [77]. The generalised Dice loss function softens this issue by monitoring the contribution that each class provides to the loss [78].

The Dice Loss function is based on the DC that measures the overlap between the predicted result and the ground truth, being commonly used in semantic segmentation and easier to analyse. More details about the network configuration can be consulted on appendix A and B (Table A1 and Figure B1).

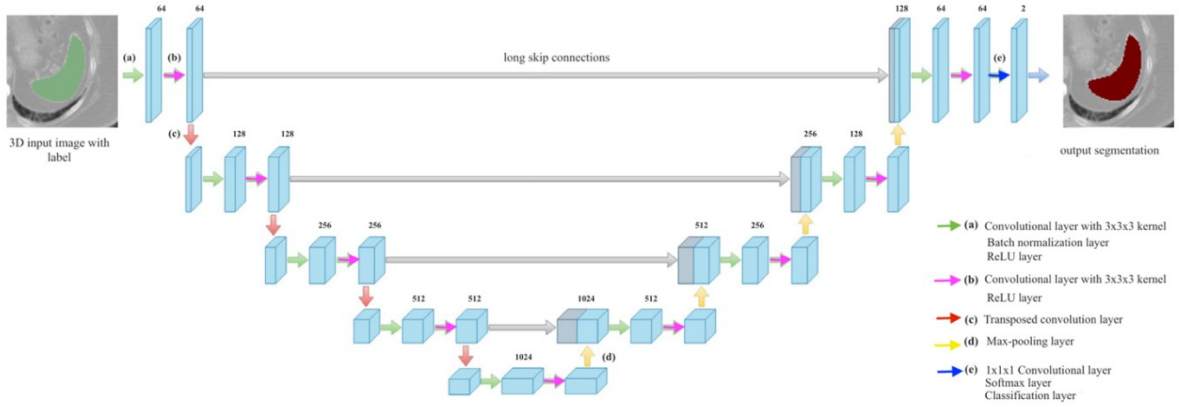

Figure 3.4: 3D U-net architecture designed for the spleen segmentation.

The hyperparameters were pre-defined to optimise the parameters using the "trainingOptions" function in MATLAB. The options selected were adapted to the segmentation of 3D medical images. The training process begins with the network starting from zero. Initially, since it is a non-deterministic process, the parameters are random and, during the network training, will be optimised. The mini-batch size is a subset of the training set used to update the weights and to compute the optimisation function and was set to three, because it was the maximum value that the GPU could support. Additionally, the training and internal validation data was set to shuffle before each epoch and network internal validation, respectively. The internal validation frequency was set to 50, which means that the network was validated every 50 iterations. The optimiser chosen was the Adam optimiser, with an initial learning rate of 5×10^{-4}, remaining constant throughout the training process [79].

The loss functions used was the generalised Dice loss. For the CT spleen dataset used, number of epochs was 25. Table 3.7 summarises all hyperparameters selected for the neural network optimisation.

Table 3.7: Hyperparameters selected for the network optimisation.

Hyperparameters	
Optimiser	Adam
Number of epochs	25
Mini-batch size	3
Initial learn rate	5×10^{-4}
Shuffle	every-epoch
Validation frequency	50

The training progress was supervised by plotting the training and internal validation accuracy and loss. At every iteration, the learnable parameters are being updated and the loss is being calculated. The plots display these metrics at every iteration, making it possible to check for network overfitting, convergence and speed.

Dataset division

The MSD dataset consisted of 41 volumes with a corresponding label. Additionally, the volume "spleen_26.nii" was eliminated, given that the spleen was cropped in the CT scan, affecting the quality of the sample. Subsequently, a randomised partition of the spleen dataset was made, consisting of 30 volumes for training and five volumes for internal validation, that was given to the network optimisation process.

Image pre-processing techniques

Based on the division made between the training set and the internal validation set, various combinations have been made at the image pre-processing level in order to create an optimised network. All operations were implemented on MATLAB, except the resampling operation, that was implemented in Python programming language and in 3D Slicer 4.11 software (https://www.slicer.org).

All spleen images had heterogeneous voxel sizes, being necessary to resample the dataset for processing deep neural networks. Consequently, the images were resampled to isotropic (identical length along the three axes) voxel sizes of 2×2×2 mm^3 using linear resampling.

The intensity values of the CT scans range approximately from -1000 to 1000 (or more) Hounsfield units corresponding the air and bone densities, respectively. Thus, the images were normalised with different ranges of intensity: [-500; 500], [-250; 250], [-100; 250], [-50; 250]. This technique helped differentiate anatomical structures of intensities similar to the spleen, such as liver and kidneys, and to understand which values were the most suitable to help the trained network segmenting the spleen.

Ideally, the input image should be introduced into the deep neural network with its original dimensions. However, this is impossible, as the entire 3D image requires an extensive amount of GPU memory during the training process. Due to this issue, there is a need for a patch-wise segmentation, where the medical images are cropped into volumetric patches. Preferably, the patches should be as large as possible, in order to cover a decent amount of contextual information, which will be used by the neural network to learn, allowing it to learn all pertinent features.

Furthermore, it was imperative to crop the training and internal validation sets into patches. Therefore, the spleen CT scans and its corresponding labels, were cropped into cubic patches of 96×96×96 voxels according to the centre of mass of the spleen, so that the patch necessarily contained the entire spleen in the centre. Although the image dimensions have been reduced, due to the resampling mechanism, its size still required a lot of GPU memory to process. Thus, the patch size was selected by a compromise between the GPU capacity and the image size, to encompass various anatomical structures, to introduce variability to the training process (the neural network is able to learn more details) and to give spatial context.

Next, to have patches with better contextual information and improve the network's robustness, variability was introduced in the network training process, where the spleen was shifted with random values between -20 and 20 voxels in each direction in each patch. This results in a better representation of the reality because each patch contains more structures and background. Also, the shift value was incremented to ± 45 voxels, in order to add even more variability to the data, where the patches still contained almost the entire spleen, yet exhibiting more anatomical structures, making the learning process more complex and diverse. With this technique, it is expected to obtain improved segmentation results.

Transfer learning

As a last approach to enhance the network performance, the transfer learning method was applied, where the training process begins with a previously trained network, that was trained on a different set. In this case, instead of starting from zero with the random initial weights, the new network is going to be initialised with the weights of the last layers of the pre-trained network.

The network configuration was not altered, maintaining the hyperparameters previously selected, expect for the initial learn rate that decreased to 5×10^{-8}, to reduce the overfitting.

For each experiment, the network with higher internal validation DC was selected as the pre-trained network, initialising the weights of the next training process. Thus, the experiments consisted of using the pre-trained network of the patches centred in the spleen set and then retrain this network with a new training set, that consisted of the patches of the shifted spleen of ± 20 voxels. Subsequently, this resulting network was retrained again with the dataset of the patches of the shifted spleen of ± 45 voxels.

In literature, transfer learning has proven to improve the performance of the segmentation trained network [51].

All strategies performed for the spleen dataset in CT images are summarised in Table 3.8 as well as the corresponding codification that will be used throughout the text.

Table 3.8: Strategies applied for the spleen CT dataset images and its corresponding codification.

S_1	Spleen centred in patches
S_2	Spleen centred in patches with intensity normalisation [-500; 500]
S_3	Spleen centred in patches with intensity normalisation [-250; 250]
S_4	Spleen centred in patches with intensity normalisation [-100; 250]
S_5	Spleen centred in patches with intensity normalisation [-50; 250]
S_6	Patches with spleen translation of [-20; 20] voxels
S_7	Patches with spleen translation of [-20; 20] voxels with intensity normalisation of [-500; 500]
S_8	Patches with spleen translation of [-20; 20] voxels with intensity normalisation of [-250; 250]
S_9	Patches with spleen translation of [-20; 20] voxels with intensity normalisation of [-100; 250]
S_{10}	Patches with spleen translation of [-20; 20] voxels with intensity normalisation of [-50; 250]
S_{11}	Patches with spleen translation of [-45; 45] voxels
S_{12}	Patches with spleen translation of [-45; 45] voxels with intensity normalisation of [-500; 500]

S_{13}	Patches with spleen translation of [-45; 45] voxels with intensity normalisation of [-250; 250]
S_{14}	Patches with spleen translation of [-45; 45] voxels with intensity normalisation of [-100; 250]
S_{15}	Patches with spleen translation of [-45; 45] voxels with intensity normalisation of [-50; 250]
S_{16}	Transfer Learning - using pre-trained network of spleen centred patches in patches with spleen translation of [-20; 20] voxels
S_{17}	Transfer Learning - using pre-trained network of spleen centred patches in patches with spleen translation of [-20; 20] voxels IN [-500; 500]
S_{18}	Transfer Learning - using pre-trained network of spleen centred patches in patches with spleen translation of [-20; 20] voxels IN [-250; 250]
S_{19}	Transfer Learning - using pre-trained network of spleen centred patches in patches with spleen translation of [-20; 20] voxels IN [-100; 250]
S_{20}	Transfer Learning - using pre-trained network of spleen centred patches in patches with spleen translation of [-20; 20] voxels IN [-50; 250]

3.2.2 Dataset 2 – Whole-body [^{18}F]FDG PET/CT malignant lesions

Network architecture and implementation

The U-net architecture used had the same configurations and hyperparameters selected as the previous network, except for the number of epochs. For the training process that used the dataset that contained only the whole-body [^{18}F]FDG PET dataset, the number of epochs was 40. When the CT channel was added to the dataset, the number of epochs increased to 100.

As mentioned above, the limiting GPU capacity is a huge problem, which restricts the number of patches being given to the network during the training process. Therefore, it is important to test different approaches to overcome this obstacle. To ascertain whether the number of filters applied in each convolution significantly affected the results, they were reduced to half. Figure 3.5 illustrates the U-net architecture with half of the number of filters used for the whole-body [^{18}F]FDG PET/CT malignant lesions dataset.

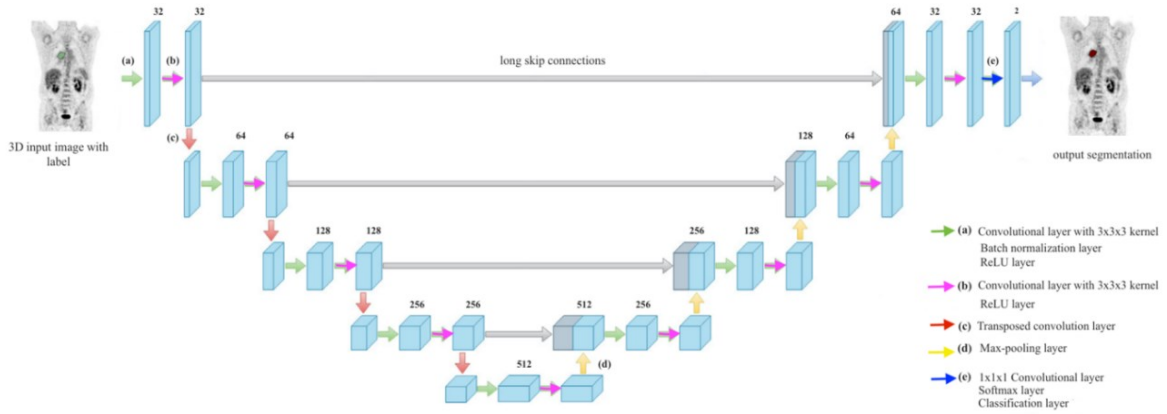

Figure 3.5: 3D U-net architecture with half of the number of filters designed for the whole-body [^{18}F]FDG PET/CT malignant lesion segmentation.

Dataset division

Once again, for the training process, the dataset must be split into training and internal validation sets. Considering that the number of patients varies for each possible pathology, a strategy had to be adopted to determine the criterion of division of the dataset, to ensure that the data is representative in all sets. Thus, it was defined that the training set must have at least one patient per each possible pathology. This is due to the importance of training the network with as much data as possible and have a high representativeness of the data. If the training data does not contain a pathology, the algorithm may not learn to recognise lesions from that pathology. Therefore, if there is only one patient for a pathology, it should go to the training set. In the case where there are two patients for a pathology, one of them should go to the training set and the other to the testing set, that will be explained in the next chapter. In the event that there are three patients for one pathology, a patient should go to each set. If there are four or more patients per pathology, the data must be distributed evenly, ensuring that there is an equal proportion between men and women, since the body anatomy varies depending on the patient's sex. It is important to note that patients were distributed randomly. The data distribution according to the suspicion of primary tumour and sex for each set is described in appendix C (Table C1). Table 3.9 summarises the data division of the dataset into training and internal validation sets.

Table 3.9: Data division into training and internal validation sets.

Sex	Training set	Internal validation set
M	19	5
F	21	7
Total	40	12

Image pre-processing techniques

Once again, due to the minimal GPU capacity, images were cropped to patches of 96×96×96 voxels centred in each lesion, meaning that if the image had five lesions, it would result in five patches. However, since this dataset had a higher number of volumetric images, it was required to reduce the patch size even more due to memory constraints. Thus, the experiments were conducted in patch sizes of 48×48×48 and 32×32×32 voxels centred in each lesion according to its centroid. In a first approach, the original network architecture was trained with the patches of size 48×48×48 voxels. Afterwards, the following networks were trained with the same network architecture, albeit with half the number of filters.

As previously mentioned, PET/CT integrates both metabolic/functional and anatomical information. For PET imaging, there is a discrepancy between the malignant lesions and healthy tissues, albeit the edge of the lesions is generally blurred, due to the low spatial resolution of PET images. In CT scans, the images may have higher resolution, although the difference in intensity between lesions and surrounding soft tissues may not be significant, and therefore difficult/impossible to differentiate. Studies have shown that combining both PET and CT images may improve the performance of malignant lesion detection and segmentation, through the complementary information provided by other modalities [80], [81].

Therefore, to obtain a better performance, a second channel containing the correspondent CT scans was added to the [^{18}F]FDG PET scans. Thus, this channel will ideally contribute to the elimination of the organs of high [^{18}F]FDG avidity, such as the brain, heart, liver, parotid glands, gastrointestinal tract, bladder and kidneys. In addition, it provides more detail to the network. In order to accomplish that, a series of pre-processing methods were applied to the CT images. In a first stage, the CT images were resampled to the corresponding original PET sizes. Subsequently, the CT images were cropped into patches with the identical size of the PET

48×48×48 patches. At a final stage, both PET and CT patches were combined into a two-channel image.

All strategies performed for the whole-body [^{18}F]FDG PET/CT malignant lesions dataset are summarised in Table 3.10 as well as the corresponding codifications that will be used throughout the text.

Table 3.10: Strategies applied for the whole-body [^{18}F]FDG PET/CT malignant lesions dataset and its corresponding codifications.

S_1	48×48×48 Patches centred in each lesion
S_2	48×48×48 Patches centred in each lesion with half of filters number
S_3	32×32×32 Patches centred in each lesion with half of filters number
S_4	48×48×48 Patches centred in each lesion with two channels (PET/CT) with half of filters number

3.2.3 Dataset 3 – Whole-body [^{18}F]FDG PET lymphoma

Network architecture and implementation

For the third dataset, the network architecture was similar to the U-net, with the number of filters reduced to half. Furthermore, since this is the largest dataset being experimented so far, a layer from the U-net structure was removed, in order to increase the size of the patches being fed to the network. Thus, it was expected that with the removal of a network layer, the number of weights being learned during the training process will significantly decrease, which may accelerate the training process and allow to feed the network with larger patches. The hyperparameters selected were equal to the previous networks, except for the number of epochs that increased to 50.

Figure 3.6 illustrates the U-net architecture with one less layer used for the whole-body [^{18}F]FDG PET lymphoma lesions dataset.

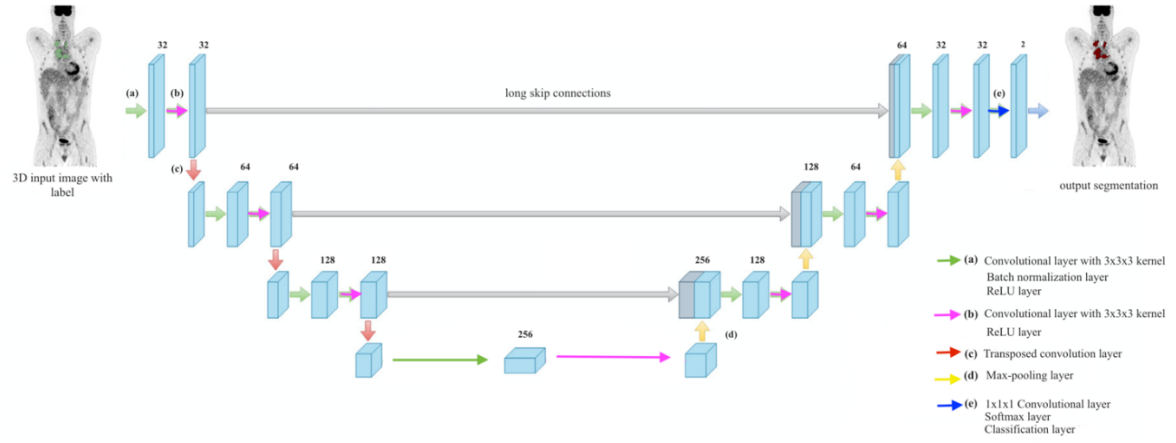

Figure 3.6: 3D U-net architecture with one less layer designed for the whole-body [^{18}F]FDG PET lymphoma lesion segmentation.

Dataset division

For the training process, only the MICCAI dataset was used, being split: 80% for training and 20% for internal validation. The former set consists of 115 PET images, while the latter set consists of 29 PET images.

Image pre-processing techniques

The size of each lesion is highly heterogeneous amongst the volumetric images, with lesions with only a few voxels and larger ones with hundreds of voxels, being necessary to ensure that the whole lesion appears in the patch, or at least most of it. Hence, both training and internal validation sets were cropped into patches centred in each lesion, according to its centroid, with a patch size of 48×48×48 voxels, due to memory constraints. However, due to the small patch size, the patches did not include many anatomical structures, making it difficult to have contextual information.

Consequently, 10 random patches of each patient with a size of 48×48×48 voxels were created, in order to introduce variability and anatomical regions that did not appear in the previous patches. This would provide further information to the trained network, which would make it more robust. Afterwards, more 10 random patches of the same size were added, further increasing the information given to the network.

After removing the last layer from the U-net structure, it was possible to increase the patch dimensions. Therefore, the network was fed with patches centred in each lesion of size 64×64×64 voxels.

Afterwards, to increase even more the patch size to 96×96×96 voxels, the number of patches were reduced by reducing the number of disconnected lesions by applying two morphological operations. Initially, a dilate morphological operation was applied to the labels with a circular kernel with a radius of three, merging the adjacent lesions, resulting in less lesions per patient. This means that lesions that were at a distance inferior of three voxels were merged using the morphological operation of the dilate on the label masks. Thus, the number of independent lesions diminish and consequently the number of patches that are going to be fed into the network. This also allowed to increase the patch size to 96×96×96 voxels. Nonetheless, it is essential to note that the network is going to be provided with the original lesions rather than the lesions that underwent the morphological operation. Moreover, the open operation was applied, eroding and then dilating the labels, whereas small lesions (less than three voxels diameter) were not used to define the patch centres. This procedure also reduced the number of patches for the training process.

The final image pre-processing technique used was the delineation of a boundary box to crop the excess background of the image (outer body image content), only leaving the image that contains the human body. After cropping according to the boundary box, patches centred in each lesion were cropped with a patch size of 96×96×96 voxels.

All strategies performed for the whole-body [^{18}F]FDG PET lymphoma dataset are summarised in Table 3.11 as well as the corresponding codifications that will be used throughout the text.

Table 3.11: Strategies applied for the whole-body [^{18}F]FDG PET lymphoma dataset and its corresponding codifications.

S_1	48×48×48 Patches centred in each lesion
S_2	48×48×48 Patches centred in each lesion with more 10 48×48×48 random patches per patient
S_3	48×48×48 Patches centred in each lesion with more 20 48×48×48 random patches per patient
S_4	64×64×64 Patches centred in each lesion with U-net with less one layer
S_5	96×96×96 Patches centred in each lesion after dilate operation with U-net with less one layer
S_6	96×96×96 Patches centred in each lesion after open operation with U-net with less one layer
S_7	96×96×96 Patches centred in each lesion after creation of a boundary box with U-net with less one layer

3.3 Inference – Optimised network evaluation

To evaluate the performance and efficiency of a segmentation network in unseen data, different evaluation metrics must be used. More than one measure should be calculated to ensure a fairer assessment [4].

The DC is the most widely used quantitative measure of spatial overlap between the ground truth and the predicted segmentation. DC values range of [0,1], where zero means no overlap and one equal to total overlap. Therefore, the higher the DC value, the higher the overlap, which indicates a better performance. DC is calculated according to the following expression (equation 3.3.1):

$$DC\ (V_1,V_2) = \frac{2 \times |V_1 \cap V_2|}{|V_1| + |V_2|} \qquad (3.3.1)$$

Where V_1 is volume of the ground truth, V_2 is the predicted volume and $V_1 \cap V_2$ is the overlap between both volumes [4], [57], [82].

The true positives (TP) represent the intersection between the ground truth and the segmentation, false positives (FP) represent the segmentation output that do not overlap with the ground truth, false negatives (FN) correspond to the portions of the ground truth that was not segmented, and true negatives (TN) represents the portion that is outside of both the segmentation output and the ground truth.

Sensitivity measures the percentage of TP, i.e., the capacity to classify each positive voxel (organ or tumour) correctly [4], [57]. The sensitivity can be calculated as follows (equation 3.3.2):

$$Sensitivity = \frac{TP}{TP + FN} \qquad (3.3.2)$$

Specificity measures the percentage of TN, i.e., the ability to classify each negative voxel (background tissue) correctly [4], [57].Specificity can be computed accordingly by (equation 3.3.3):

$$Specificity = \frac{TN}{FP+TN} \qquad (3.3.3)$$

Positive predictive value (PPV) (equation 3.3.4) or also known as precision, gives the proportion of the predicted positive voxels that are correctly predicted [4], [57].

$$PPV = \frac{TP}{TP + FP} \tag{3.3.4}$$

Accuracy refers to the ratio between correctly predicted voxels and the total number of voxels being evaluated. Accuracy values range of [0,1], where one signifies that all positive and negative voxels are classified correctly and zero means that neither of the positive nor negative voxels have been properly classified [4], [57]. Accuracy can be calculated as the following expression (equation 3.3.5):

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.3.5}$$

However, it is important to refer that it is expected that accuracy will always have high values, due to the high percentage negative voxels in the datasets used. So, it is important to note that a high accuracy, does not translate to an accurate segmentation of the object of interest.

3.3.1 Inference strategies

Although the training process was performed with the patches, the test set images were tested on the entire image, with the intention of represent the reality, because this is what is intended in clinical practice. Hence, a sliding window approach was adopted, by setting the window size as the same as the patch size used for the training process.

In parallel, since the images used for the training process were provided in patches of dimensions smaller than the dimensions of the original images, it was necessary to divide the images that will be tested into patches with equal dimensions. However, for the images used, it is not possible to divide the original images into adjacent patches with equal sizes, in which their union gives the original image. Thus, there are always the need for an overlap between patches. Therefore, the sliding window moved by a step size of approximately 70% of the patch size, resulting in overlapped patches, where the smaller the step, the higher the overlap percentage (different overlap sizes affects the DC).

Consequently, some of the voxels may enter several times in different patches to be classified. It is required to define that in all images, each voxel belongs only to a single class (non-lesion/spleen or lesion/spleen tissue). Hence, a voting strategy by majority has been defined.

For the spleen CT images dataset, the test set consisted of five CT scans. It is important to recall that the test set is an external set of unseen volumetric images used for an independent network's performance evaluation, that should only be used once. Afterwards, the strategies applied were the sliding window technique of a step proportion of 70% of the patch size as well as the voting by majority strategy.

Regarding the second dataset, the test set is constituted of 14 independent PET images and both aforementioned approaches were implemented. In addition, border voxels were eliminated (zero, three and six border voxels for all three axes in each inference approach), to increase the results, since it reduces the noise associated with the borders. Also, different step sizes (20 and 30 voxels) for the sliding window mechanism were analysed.

Concerning the lymphoma lesions dataset, the test set comprises 65 PET images from an external dataset obtained at the Nuclear Medicine department of the CCC. The strategies applied were the sliding window technique of a step proportion of 70% of the patch size, border voxels removal (zero and five border voxels in all three axis), and the voting by majority strategy.

The optimised network selection criterion was through the internal validation DC, which is expected that the higher the value, the higher the network's performance. Hence, statistical metrics were calculated for the DCs in the IBM SPSS (https://www.ibm.com), namely mean, STD, median (2^{nd} quartile), minimum, maximum, 1^{st} quartile and 3^{rd} quartile.

3.4 Hardware and software

The hardware used for this project was a personal computer equipped with a NVIDIA GeForce RTX 2070 SUPER GPU with 8.0 GB of dedicated memory. The selection of the hardware is of vital importance since it influences the image pre-processing and deep neural network performance, accelerating the training process. Due to the usage of volumetric images and deep learning algorithms computational sophistication, it is necessary to have a high computational capacity and high dedicated memory on the GPU [11], [25]. In addition, the

computer used is a Windows 10 with an Intel® Core™ i7-9700K central processing unit (CPU) and 16 GB of RAM (random-access memory).

Several software programs were used, i.e., MATLAB, Compute Unified Device Architecture (CUDA) toolkit (*https://developer.nvidia.com/*), Python, 3D Slicer 4.11 software (*https://www.slicer.org/*) and ITK-SNAP (*http://www.itksnap.org/*).

Starting with MATLAB, this programming framework served as the backbone of this project, since it was used to develop and implement all the algorithms to pre-process the medical images, train the network and test the best trained network with unseen images. In order to accomplish these tasks, it was necessary to use MATLAB toolboxes and even require to the MathWorks File Exchange platform, allowing the download of functions created by different contributors. The Table 3.12 summarises the toolboxes used in this project and their functionalities.

The parallel computing toolbox speed up the training process since it allows the algorithm to be executed on NVIDIA GPUs and multicore processors. Therefore, this toolbox enables functions from the deep learning and computer vision toolboxes, allowing to perform CUDA-enabled NVIDIA GPU directly from MATLAB, without the need to have existence knowledge about CUDA programming [83], [84].

CUDA is a parallel computing platform developed by NVIDIA, which exploits the GPU capacity, increasing the computational performance. It is important to mention that to use CUDA, it is mandatory to have a NVIDIA GPU, download and install the CUDA toolkit, that provides all the necessary tools and libraries to develop GPU-accelerated applications. For this study, CUDA 11 was installed, because it was supported by the MATLAB 2021a version. MATLAB has a collaboration with CUDA, allowing the usage of NVIDIA GPU directly on MATLAB and that the GPU is automatically selected for computations. To note that MATLAB only supports NVIDIA GPU [85]–[87].

The computer vision toolbox offers several features and functions to operate in computer and 3D vision, as well as in video processing systems. Thus, this toolbox allows semantic image segmentation using deep learning networks, their design and to evaluate the segmentation results [88].

The deep learning toolbox allows the design and implementation of deep neural networks, in this case CNN. With this toolbox, it is possible to train neural networks, design the deep neural network with the layers available, use pretrained networks and transfer learning method, and import PyTorch and TensorFlow networks into MATLAB by using ONNX (open neural network exchange) [89].

The image processing toolbox focuses on image processing, as well as image visualisation and analysis, being capable to support both 2D and 3D images. Then, this toolbox can perform various operations, such as contrast adjustment, image filtering and enhancement, morphological operations, cropping and binary conversion. Thus, for this study it was necessary to use this platform, so that 3D volumetric image processing could be carried out. Moreover, in alliance with the deep learning toolbox, it allows the creation of datastores, their combination and random extraction of patches from images [90].

Table 3.12: MATLAB Toolboxes and their functionalities in the development of the algorithms.

MATLAB Toolbox	**Functionality**
Parallel computing toolbox	Execute the algorithms on NVIDIA GPU
Computer vision toolbox	Semantic segmentation operations
Deep learning toolbox	Design and implement deep neural networks
Image processing toolbox	Image process operations

The functions utilised from File Exchange were the "Tools for NIfTI and ANALYSE image" to load, save, make and view NIFTI data; the "Natural-Order Filename Sort" that has functions that sort the names of files and folders in an alphanumeric order; and, the "Image Segmentation Quality Scores" that have a function to evaluate the segmentation performance, that includes the following evaluation metrics, the DC, Jaccard index, precision, sensitivity, accuracy, and specificity [91]–[93].

The 3D Slicer 4.11 software (https://www.slicer.org) and ITK-SNAP were used for the purpose of image visualization and analysis, resample the voxel sizes and crop images. However, these software only handle one image at once.

To automate the resampling process, due to the large quantity of images in the dataset, Python language was used. The Python script was developed in PyCharm 2021.3.3 (https://www.jetbrains.com), that is an IDE (integrated development environment) for Python programming language. For the resampling script the NumPy and Nibabel libraries were used, along with the os.path and glob modules.

Chapter 4 Results

As previously mentioned, the selection criterion of the best optimised network was through the analysis of the internal validation DC. Therefore, the following results comprise the internal validation DCs of every applied strategy of each dataset, the selected best trained network and the test DC results.

4.1 Spleen CT scans

For the spleen CT scans dataset, only the trained networks that converged to an optimal solution were used, where each network was trained five times. In the case of network divergence, the training process would be ceased, and a percentage of divergence of 36% was obtained when the networks were initialised from zero, that is, with random weights. Regarding networks initiated by a pretrained network, the percentage of divergence decreased to 11%, which would be expected, since the network was not initialised with random weights.

The number of training patches fed to the networks was 30 patches, while the number of internal validation patches was five. For the network architecture, hyperparameters and strategies chosen, the training time reached approximately 40 minutes.

The DC was calculated between the ground truths and the segmentation results obtained during the training process and the inference test for the internal validation and test sets, respectively. The DC evaluates the overlap between both segmentations. In Table 4.1, the mean ± STD, median, minimum, maximum, and interquartile range (IQR) of the internal validation DC of all trained networks throughout the different strategies are presented. The statistical parameters were used to assess the optimised network performance, allowing the selection of the best trained network by analysing the median internal validation DC. The strategies codifications correspondences to the applied strategies can be accessed on Table 3.8.

Analysing Table 4.1, it is possible to verify that the strategy S_{18} that obtained the highest median DC was the transfer learning using the pretrained network of spleen centred in patches with a different training set that contained the patches with a spleen shifted ± 20 voxels with intensity normalisation of [-250; 250] approach, obtaining a median DC of 0.71 with an IQR of 0.12 and a mean DC of 0.69 ± 0.065.

Table 4.1: Internal validation DC of the trained networks for each applied strategy.

Applied strategy	Mean DC ± STD	Median DC (IQR)	Minimum DC	Maximum DC
S_1	0.43 ± 0.16	0.47 (0.24)	0.15	0.57
S_2	0.41 ± 0.14	0.39 (0.25)	0.22	0.59
S_3	0.58 ± 0.13	0.58 (0.24)	0.38	0.71
S_4	0.48 ± 0.093	0.48 (0.18)	0.36	0.58
S_5	0.37 ± 0.12	0.31 (0.24)	0.24	0.50
S_6	0.40 ± 0.10	0.34 (0.18)	0.33	0.53
S_7	0.41 ± 0.090	0.45 (0.17)	0.28	0.51
S_8	0.41 ± 0.17	0.33 (0.33)	0.24	0.63
S_9	0.43 ± 0.16	0.34 (0.30)	0.27	0.62
S_{10}	0.50 ± 0.15	0.54 (0.27)	0.30	0.69
S_{11}	0.52 ± 0.13	0.55 (0.19)	0.31	0.65
S_{12}	0.56 ± 0.12	0.50 (0.22)	0.47	0.70
S_{13}	0.56 ± 0.081	0.53 (0.15)	0.47	0.67
S_{14}	0.51 ± 0.21	0.60 (0.34)	0.15	0.67
S_{15}	0.63 ± 0.13	0.62 (0.21)	0.42	0.78
S_{16}	0.43 ± 0.17	0.41 (0.32)	0.21	0.61
S_{17}	0.62 ± 0.055	0.64 (0.10)	0.54	0.67
S_{18}	0.69 ± 0.065	0.71 (0.12)	0.60	0.76
S_{19}	0.52 ± 0.096	0.52 (0.17)	0.42	0.66
S_{20}	0.36 ± 0.081	0.32 (0.15)	0.29	0.47

For these experiments, the overfitting phenomena did not occur, which is a good indicator of the network robustness against unseen data. Thus, among the five trained networks, it was selected the network with the higher internal validation DC. Consequently, this network was selected to be applied on the inference test. The Table 4.2 displays the DCs obtained during the spleen segmentation in the test images.

Observing Table 4.2, it is possible to identify that the median DC decreased to 0.57 with an IQR of 0.37. All the results obtained for each CT image from the external test set obtained through strategy S_{18} are presented on appendix D (Table D1).

Table 4.2: DC obtained on the segmentation of the external test set.

Best trained network	Mean DC ± STD	Median DC (IQR)	Minimum DC	Maximum DC
S_{18}	0.64 ± 0.20	0.57 (0.37)	0.46	0.91

The following table displays the mean and median sensitivity, specificity, PPV and accuracy for the inference test.

Table 4.3: Sensitivity, specificity, PPV and accuracy obtained on the external test set.

	Sensitivity	Specificity	PPV	Accuracy
Mean ± STD	0.83 ± 0.11	0.99 ± 0.00	0.55 ± 0.26	0.99 ± 0.00
Median (IQR)	0.88 (0.17)	0.99 (0.010)	0.44 (0.47)	0.99 (0.010)

Analysing Table 4.3, it is verified that the trained network has a high median sensitivity of 0.88 with an IQR of 0.17, which indicates that the algorithm correctly classifies 88% spleen voxels (TP) and 12% classifies spleen voxels as background. Regarding specificity, the trained network obtained a median of 0.99 with an IQR of 0.010, demonstrating that 99% of background voxels were correctly classified and only 1% of the background voxels were incorrectly classified as spleen voxels. The high specificity is due to class imbalance issue. The median PPV obtained was 0.44 with an IQR of 0.47, which suggests that the probability of the fully automatic algorithm to classify a spleen voxel truly as a spleen voxel is 44%. Lastly, the median accuracy obtained was 0.99 with an IQR of 0.010, which is not an accurate value, also because of the class imbalance problem.

Figure 4.1 presents the delineations of the ground truth mask and the resulting spleen segmentations in a coronal view slice. For the first two examples, in the resulting segmentations, it is possible to observe that the liver was incorrectly segmented when using the fully automatic algorithm. In addition, in some CT images, the heart is also incorrectly segmented. In appendix D, a detailed description of the resulting spleen segmentations is presented.

It is good practice to choose only an optimised network and test with the unseen images solely once, to not bias the results. As mentioned above, the internal validation DC simulates the reality of the optimised network behaviour, meaning that a larger DC denote a better performance of the trained network. However, it is not guaranteed that a trained network that has obtained a higher internal validation DC, means it has better results when applied in the test

images. For research purposes, all networks were tested, and the results of the mean ± STD, median, and IQR DCs are present on appendix D (Table D2).

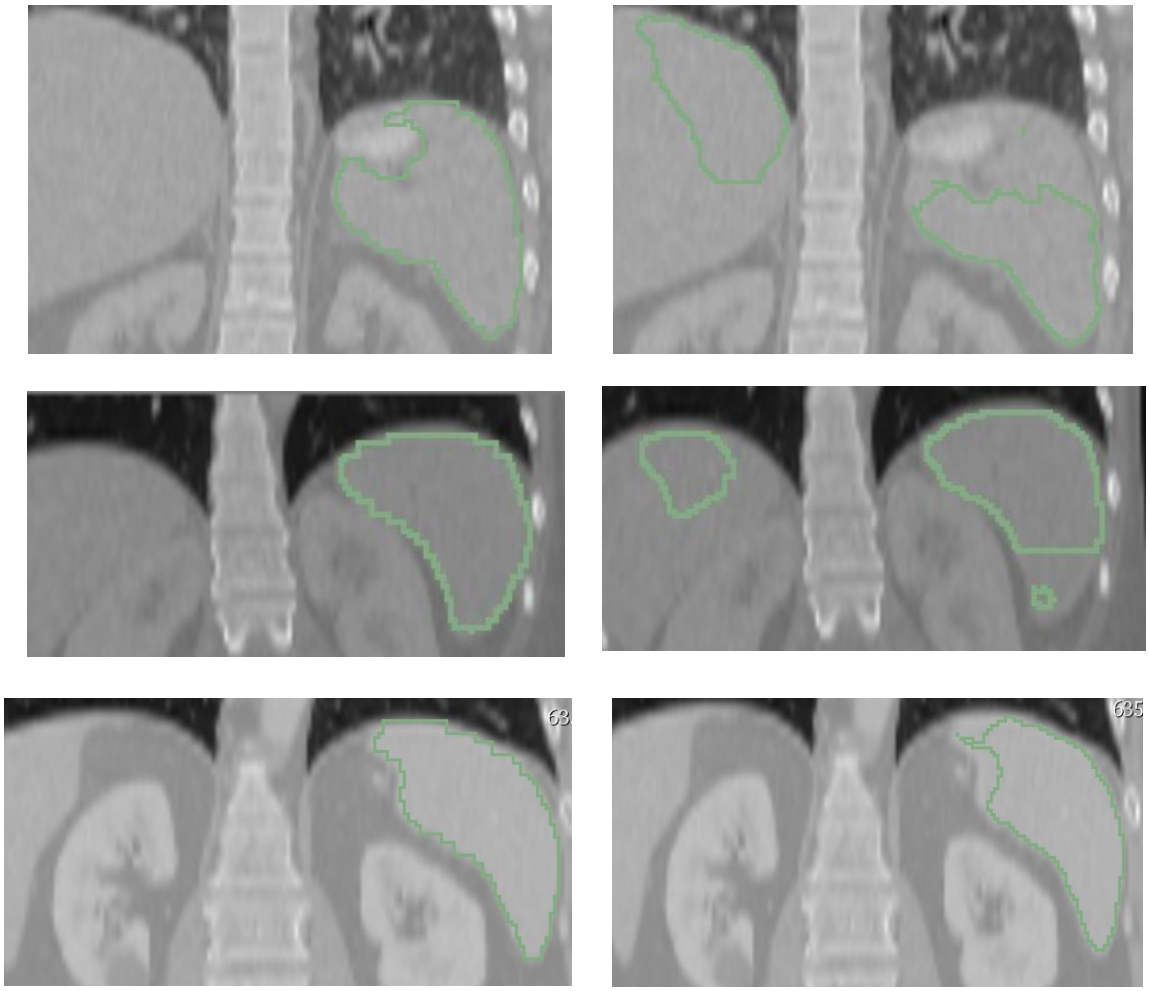

Figure 4.1: External test set examples of ground truth (upper row) and its corresponding spleen resulting segmentations (bottom row) for three different DC: 0.46, 0.57, 0.91, respectively.

4.2 Whole-body [^{18}F]FDG PET/CT malignant lesions

Regarding the whole-body [^{18}F]FDG PET/CT malignant lesion dataset, once more, each network was trained five times, and all networks converged to an optimal solution. For the training process, the network was provided with 170 patches for training, and 69 patches for internal validation.

The Table 4.4 presents the mean ± STD, median, minimum, maximum, and IQR of DC, and the training time in minutes of all trained networks across the several approaches. Since all strategies varied in patch size and architecture, including the last strategy having a second

channel, the training time differed between training processes. The strategies codifications correspondences to the applied strategies can be accessed on Table 3.10.

Table 4.4: DC of all the trained networks for each applied strategy.

Applied strategy	**Mean DC ± STD**	**Median DC (IQR)**	**Minimum DC**	**Maximum DC**	**Training time**
S_1	0.26 ± 0.038	0.26 (0.067)	0.22	0.31	~ 23
S_2	0.27 ± 0.019	0.27 (0.025)	0.26	0.30	~ 11
S_3	0.15 ± 0.038	0.13 (0.068)	0.10	0.19	~ 6
S_4	0.21 ± 0.067	0.21 (0.12)	0.12	0.30	~ 11

Analysing Table 4.4, it is possible to verify that strategy S_2 corresponding to the network trained with patches of size 48×48×48 centred in each lesion with the U-net architecture with half of filters number achieved a superior median DC of 0.27 with an IQR of 0.025 and a mean DC of 0.27 ± 0.019.

In these training processes, it is observed some overfitting during the network training. Besides that, the chosen network was the one with the highest internal validation DC. Therefore, this was the trained network selected to be implemented on the test set.

The Table 4.5 demonstrates the DCs obtained during the malignant lesion segmentation in the test images for different approaches, by varying the step and border values.

Table 4.5: DC obtained on the segmentation of the external test set.

Best trained network	**Step**	**Border**	**Mean DC ± STD**	**Median DC (IQR)**	**Minimum DC**	**Maximum DC**
S_2	20	0	0.29 ± 0.27	0.23 (0.50)	0.00	0.84
		3	0.31 ± 0.28	0.29 (0.45)	0.00	0.89
		6	0.31 ± 0.27	0.31 (0.42)	0.00	0.88
	30	0	0.30 ± 0.26	0.28 (0.45)	0.00	0.81
		3	0.31 ± 0.27	0.29 (0.43)	0.00	0.87
		6	0.31 ± 0.27	0.28 (0.42)	0.00	0.86

Analysing Table 4.5, for the step of 20 voxels, it is possible to verify that eliminating six border voxels in each direction, the median DC increases from 0.23 with an IQR of 0.50 to 0.31 with an IQR of 0.43, and the mean DC increases from 0.29 ± 0.27 to 0.31 ± 0.27. As for the step of 30 voxels, it is possible to observe that there are not any considerable DC changes, obtaining a median DC of 0.28 with an IQR of 0.45, and a mean DC of 0.31 ± 0.27, by eliminating six border voxels in all axis directions. All of the DCs obtained for each inference approach are demonstrated on appendix D (Table D3 and Table D4). The following table displays the mean and median sensitivity, specificity, PPV and accuracy.

Table 4.6: Evaluation metrics obtained on the test set for different steps and borders.

	Step	Border	Sensitivity	Specificity	PPV	Accuracy
Mean ± STD	20	0	0.31 ± 0.31	1.00 ± 0.00	0.44 ± 0.37	1.00 ± 0.00
Median (IQR)			0.17 (0.58)	1.00 (0.00)	0.48 (0.87)	1.00 (0.00)
Mean ± STD		3	0.35 ± 0.34	1.00 ± 0.00	0.41 ± 0.35	1.00 ± 0.00
Median (IQR)			0.21 (0.66)	1.00 (0.00)	0.44 (0.82)	1.00 (0.00)
Mean ± STD		6	0.39 ± 0.33	1.00 ± 0.00	0.36 ± 0.32	1.00 ± 0.00
Median (IQR)			0.25 (0.57)	1.00 (0.00)	0.39 (0.59)	1.00 (0.00)
Mean ± STD	30	0	0.34 ± 0.30	1.00 ± 0.00	0.40 ± 0.34	1.00 ± 0.00
Median (IQR)			0.22 (0.52)	1.00 (0.00)	0.46 (0.75)	1.00 (0.00)
Mean ± STD		3	0.38 ± 0.32	1.00 ± 0.00	0.36 ± 0.32	1.00 ± 0.00
Median (IQR)			0.25 (0.56)	1.00 (0.00)	0.37 (0.65)	1.00 (0.009
Mean ± STD		6	0.41 ± 0.32	1.00 ± 0.00	0.34 ± 0.29	1.00 ± 0.00
Median (IQR)			0.33 (0.59)	1.00 (0.00)	0.36 (0.52)	1.00 (0.00)

Analysing Table 4.6, it is possible to verify that the optimised network has median sensitivity ranging from 0.17 with an IQR of 0.58 to 0.33 with an IQR of 0.59, pointing out that the fully automatic algorithm correctly classifies 17% (when the step value is 20 and zero border voxels were removed) and 33% (when the step value is 30 and six border voxels were removed) malignant lesion voxels (TP). In terms of specificity, the optimised network obtained a median of 1.00 with an IQR of 0.00, indicating that all background voxels were correctly classified, again due to the class imbalance problem. Regarding this issue, accuracy demonstrates also a median of 1.00 with an IQR of 0.00. In relation to PPV, the obtained median ranges from 0.36 with an IQR of 0.52 to 0.48 with an IQR of 0.87, when the step value is 30 with six border voxels removed and the step value is 20 with zero border voxels removed, respectively,

demonstrating the probability of the algorithm classifying a malignant lesion voxel actually as a malignant lesion voxel.

Figures 4.2 and 4.3 illustrate boxplots that represent the distribution of the 14 patient's DC values between the manual segmentation and the predicted segmentations using this trained network.

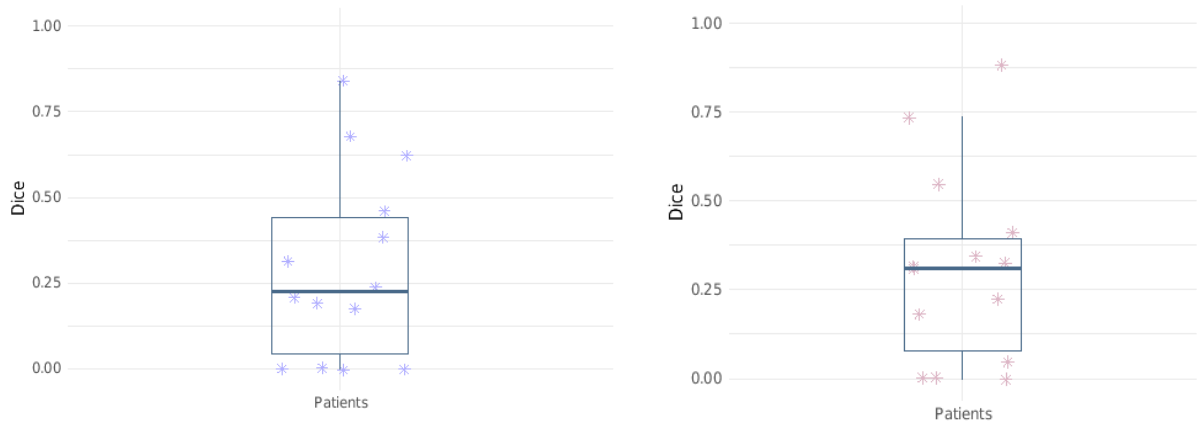

Figure 4.2: Boxplots representing the DC distribution of the 14 patients using a step of 20 voxels and a removal of zero (left) and six (right) border voxels.

Figure 4.3: Boxplots representing the DC distribution of the 14 patients using a step of 30 voxels and a removal of zero (left) and six (right) border voxels.

By analysing the boxplots, it is possible to observe that there were some patients with a DC of zero, which means that the fully automatic algorithm did not perform well at all. In a few cases, the algorithm performed reasonable segmentations, obtaining DCs above 0.60.

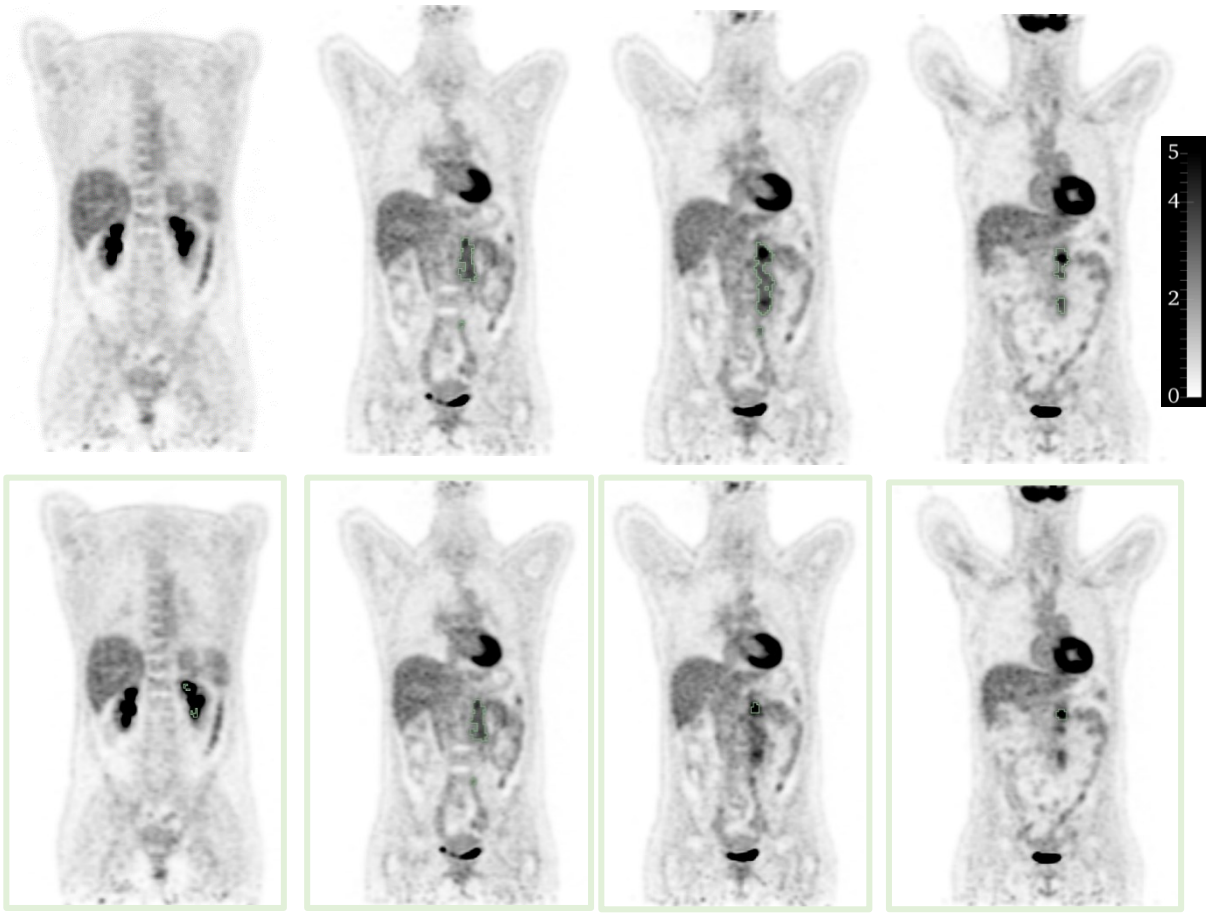

Figure 4.4: Several coronal slices of the ground truths (upper row) and the resulting segmentations (bottom row), with a DC of 0.21, when applied the fully automatic algorithm, with strategy S_2 with a step value of 30 voxels and zero border voxels removed, of a patient included in the test set from the whole-body [^{18}F]FDG PET/CT malignant lesions dataset (dataset two). The ground truth was manually identified and segmented by an experienced nuclear medicine physician from Champalimaud Foundation. The 3D Slicer 4.11 software (https://www.slicer.org) was used, and it was selected the colour map InvertedGrey with an intensity range of [0 - 5] SUV.

In Figure 4.4, it is observable that the resulting segmentation has almost no overlap with its respective ground truth, obtaining a DC of 0.21, which is below the average. In the ground truths, it is possible to view some lesions located in the abdominal region and the normal high uptake organs, such as the brain, heart, urinary bladder, and kidneys. In the resulting segmentation, only a portion of the abdominal malignant lesion identified by the nuclear medicine physician was identified and segmented. In addition, the kidneys were identified and segmented, meaning that the trained network did not distinguish this region from lesions.

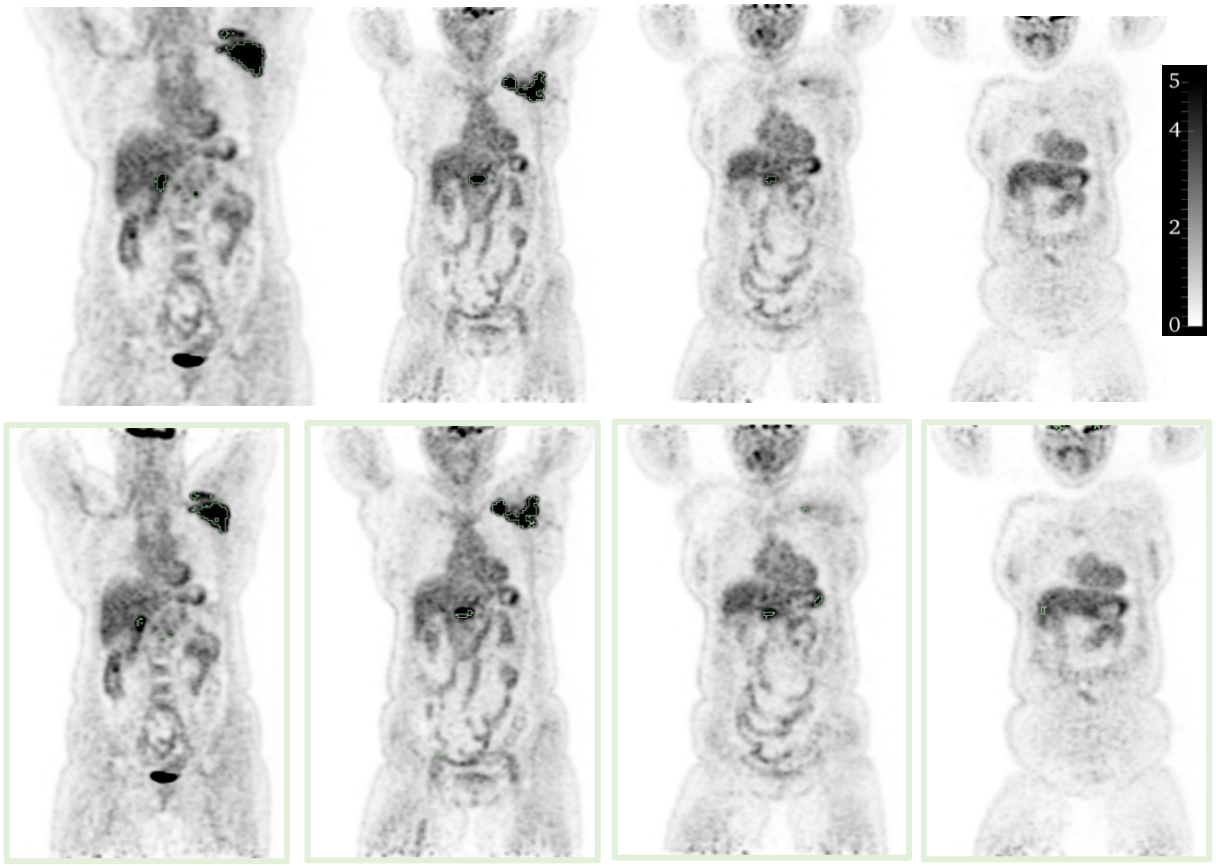

Figure 4.5: Several coronal slices of the ground truths (upper row) and the resulting segmentations (bottom row), with a DC of 0.81, when applied the fully automatic algorithm, with strategy S_2 with a step value of 30 voxels and zero border voxels removed, of a patient included in the test set from the whole-body [^{18}F]FDG PET/CT malignant lesions dataset (dataset two). The ground truth was manually identified and segmented by an experienced nuclear medicine physician from Champalimaud Foundation. The 3D Slicer 4.11 software (https://www.slicer.org) was used, and it was selected the colour map InvertedGrey with an intensity range of [0 - 5] SUV.

In Figure 4.5, results achieved with another patient image from the test set are shown. The DC was much superior, reaching a value of 0.81, which means that the fully automatic algorithm performed adequately in the identification and segmentation of this patient's lesions, having an overlap of 81% between the ground truth and the resulting segmentation. In the ground truth image, at the patient's left, a huge mass suspected of malignancy is located in the axilla and, two smaller lesions in the abdomen. The darker regions correspond to the normal high uptake organs, such as the urinary bladder and the brain, due to having a high glucose uptake, presenting an elevated FDG accumulation. Switching to the resulting segmentation, all

lesions were correctly identified, and most of the lesions were segmented. However, the brain, heart, and a portion of the liver were segmented, which was not supposed to happen.

4.3 Whole-body [^{18}F]FDG PET lymphoma lesions

Relatively to the whole-body [^{18}F]FDG PET lymphoma lesion dataset, each network was trained only once since the training process took a considerable amount of time. Anew, all of the trained networks converged to an optimal solution.

Table 4.7 displays the mean ± STD, median, minimum, maximum, and IQR of DC of all trained networks across the various network optimisation strategies. For each strategy, the number of patches provided to the network during the training process varied, as it required a compromise between the limited GPU capacity and patch size. The number of patches fed to the network for training and internal validation is shown in Table 4.7. Due to the fact that the number of patches is different for each network, as well as changes in the network structure, the training time in minutes differs between training processes, being also present in the following table. The strategies codifications correspondences to the applied strategies can be accessed on Table 3.11.

Table 4.7: DC of all the trained networks for each applied strategy.

Applied strategy	**Mean DC** ± STD	**Median DC (IQR)**	**Minimum DC**	**Maximum DC**	**Number of training patches**	**Number of validation patches**	**Training time**
S_1	0.54 ± 0.29	0.65 (0.50)	0.00	0.85	4165	1252	469
S_2	0.53 ± 0.29	0.62 (0.46)	0.00	0.87	5315	1542	1146
S_3	0.45 ± 0.28	0.44 (0.49)	0.00	0.85	6465	1832	2475
S_4	0.60 ± 0.29	0.72 (0.37)	0.00	0.91	1865	672	692
S_5	0.49 ± 0.32	0.65 (0.65)	0.00	0.83	466	156	237
S_6	0.49 ± 0.30	0.58 (0.56)	0.00	0.85	137	36	56
S_7	0.52 ± 0.31	0.61 (0.64)	0.00	0.86	154	43	65

Examining Table 4.7, it is possible to affirm that strategy S_4 corresponding to the network trained with 64×64×64 patches centred in each lesion suspected of lymphoma with the U-net architecture without the last layer obtained the highest median DC of 0.72 with an IQR of 0.37, and a mean DC of 0.60 ± 0.29. Hence, this network was selected to be conducted on the inference test, since it fulfilled the network selection criterion.

Table 4.8: DC obtained on the segmentation of the external test set.

Best trained network	Step proportion	Border	Mean DC ± STD	Median DC (IQR)	Minimum DC	Maximum DC
S_4	70%	0	0.41 ± 0.29	0.43 (0.57)	0.00	0.89
		5	0.43 ± 0.30	0.46 (0.61)	0.00	0.90

Table 4.8 presents the DCs obtained while segmentation of the lymphoma lesions in the unseen test images with a constant step proportion and variation of border values. When the border voxels of the PET images are not eliminated, the median DC achieved 0.43 with an IQR of 0.57, and a mean DC of 0.41 ± 0.29. When five border voxels are removed, the median DC increases to 0.46 with an IQR of 0.61, and the mean DC to 0.43 ± 0.30. All of the DC results, and other evaluation metrics obtained for each volumetric image for the S_4 approach are demonstrated in appendix D (Table D5 and Table D6). Table 4.9 exhibits the mean and median sensitivity, specificity, PPV and accuracy.

Table 4.9: Evaluation metrics obtained on the test set for different borders.

	Border	Sensitivity	Specificity	PPV	Accuracy
Mean ± STD	0	0.53 ± 0.32	0.99 ± 0.00	0.39 ± 0.31	0.99 ± 0.00
Median (IQR)		0.63 (0.59)	0.99 (0.00)	0.33 (0.62)	0.99 (0.00)
Mean ± STD	5	0.53 ± 0.32	0.99 ± 0.00	0.42 ± 0.31	0.99 ± 0.00
Median (IQR)		0.64 (0.62)	0.99 (0.00)	0.40 (0.60)	0.99 (0.00)

Analysing Table 4.9, it is possible to verify that the trained network median sensitivity is 0.63 with an IQR of 0.59 and 0.64 with an IQR of 0.62, when zero border voxels and five border voxels are removed, respectively suggesting that the algorithm correctly classified 63% and 64% lymphoma lesion voxels (TP). With regards to specificity, the trained network obtained a median of 0.99 with an IQR of 0.00, which indicates that all background voxels were correctly

classified, as well as the accuracy that demonstrates a median of 0.99 with an IQR of 0.00. Regarding PPV, the obtained median is 0.33 with an IQR of 0.62 and 0.40 with an IQR of 0.60, when zero border voxels and five border voxels are removed, respectively, demonstrating the probability of the algorithm classifying a lymphoma lesion voxel truly as a lymphoma lesion voxel.

Figure 4.6 illustrates the boxplots that display the distribution of the 65 patient's DC between the manual segmentation and the resulting segmentations using this trained network. By examining the boxplots, it is possible to check that, once more, there are DCs of zero, which corresponds to a poor performance by the optimised network. Otherwise, in some cases, the fully automatic algorithm obtained DCs above 0.75, indicating a good performance.

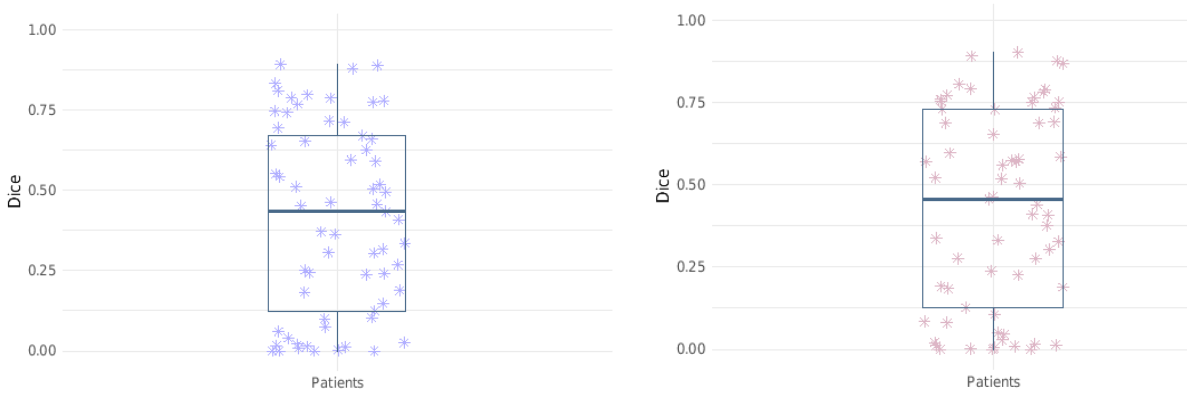

Figure 4.6: Boxplots representing the Dice coefficient distribution of the 65 patients with a removal of zero (left) and five (right) border voxels.

Regarding Figure 4.7, the U-net obtained unsatisfactory results, achieving a 18% of overlap between the ground truth and the resulting segmentation, representing a low DC. In the ground truth, the lesions suggestive of lymphoma are spread across the supradiaphragmatic region and it is visible a normal high uptake on the brain and urinary bladder. In automatic segmentation result both the brain, parotid glands and kidneys were wrongly identified and segmented. In the resulting segmentation, only one of the lesions was identified and segmented and the algorithm could not distinguish normal high uptake organs from lymphoma lesions on further slices.

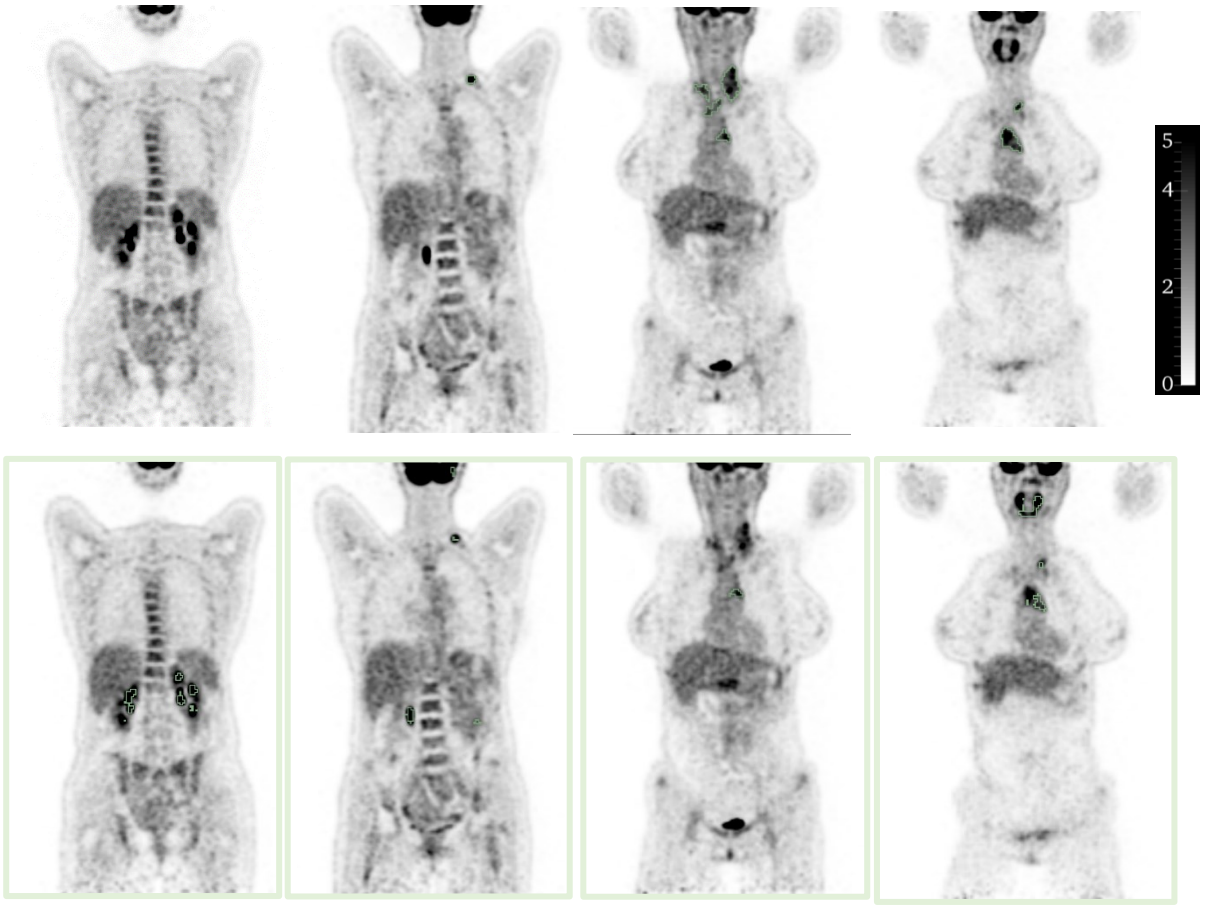

Figure 4.7: Several coronal slices of the ground truths (upper row) and the resulting segmentations (bottom row), with a DC of 0.18, when applied the fully automatic algorithm, with strategy S_4 with zero border voxels removed, of a patient included in the test set from the whole-body [^{18}F]FDG PET lymphoma lesions dataset (dataset three). The ground truth was manually identified and segmented by an experienced nuclear medicine physician from Champalimaud Foundation. The 3D Slicer 4.11 software (https://www.slicer.org) was used, and it was selected the colour map InvertedGrey with an intensity range of [0 - 5] SUV.

With regard to Figure 4.8, it was achieved an overlap of 89% between the ground truth and the resulting segmentation, which means that the fully automatic algorithm behaved successfully. In the ground truth, there are multiple lesions suggestive of lymphoma spread throughout the entire body and darker regions are noticed, corresponding to the brain and heart, which are normal high uptake organs, as previously mentioned. In the resulting segmentation, it is correct to affirm that all lesions were identified and segment. In addition, the U-net was not fully capable of distinguishing the normal high uptake organs from the lesions suggestive of lymphoma in this example, with only a portion of brain, heart, and kidneys identification and segmentation.

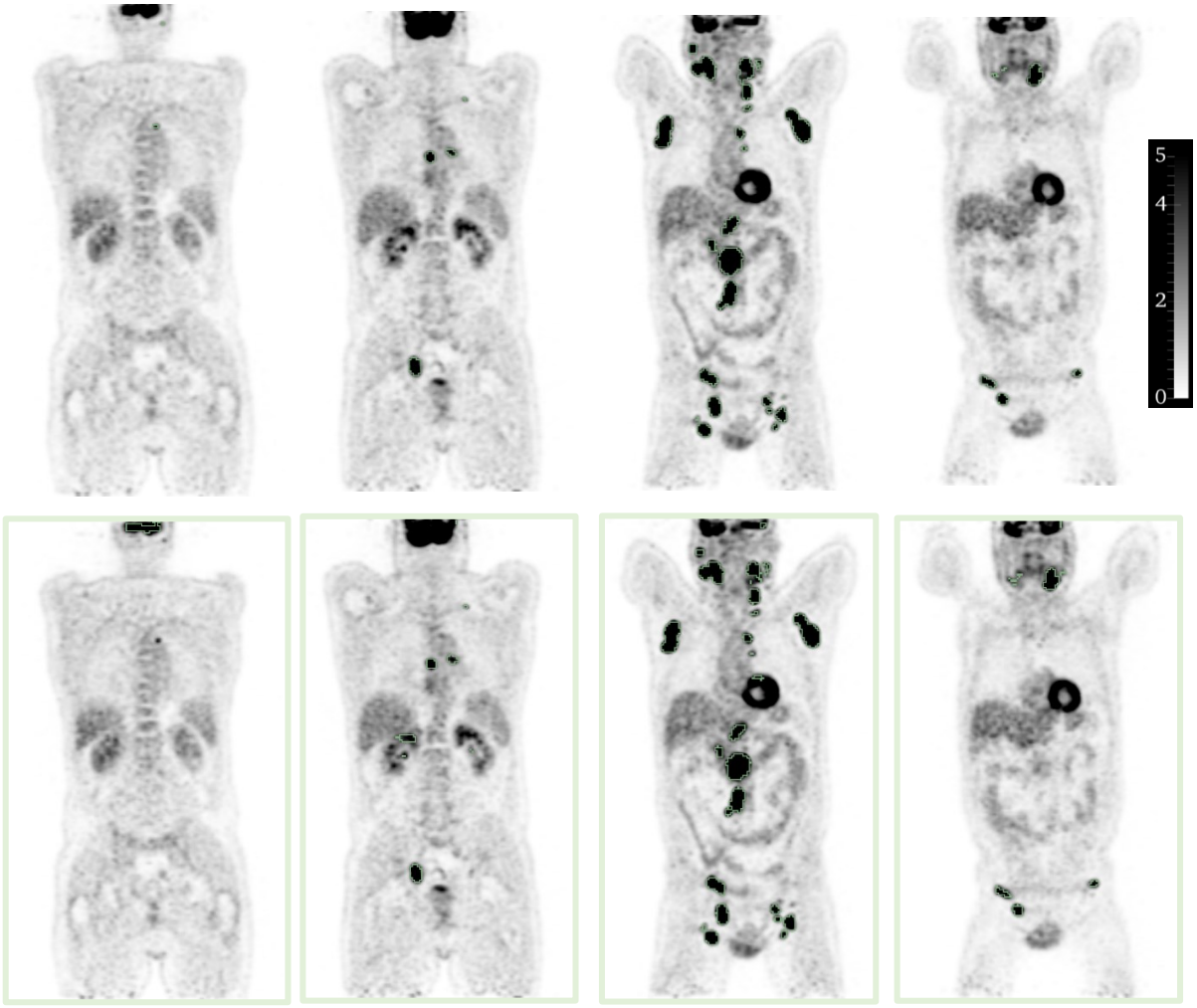

Figure 4.8: Several coronal slices of the ground truths (upper row) and the resulting segmentations (bottom row), with a DC of 0.89, when applied the fully automatic algorithm, with strategy S_4 with zero border voxels removed, of a patient included in the test set from the whole-body [^{18}F]FDG PET lymphoma lesions dataset (dataset three). The ground truth was manually identified and segmented by an experienced nuclear medicine physician from Champalimaud Foundation. The 3D Slicer 4.11 software (https://www.slicer.org) was used, and it was selected the colour map InvertedGrey with an intensity range of [0 - 5] SUV.

Chapter 5 Discussion

For this internship, CNN-based segmentation architectures were designed, and, subsequently, the pre- and post-processing techniques, definition of hyperparameters, and inference testing were developed and optimised for automatic identification and segmentation of malignant lesions in whole-body [^{18}F]FDG PET/CT images.

As previously mentioned, nowadays, the U-net architecture has been a hot topic in medical image segmentation. Although there have been reported multiple U-net variants with excellent results in the literature, only the original U-net was used for this study. It is difficult to generalise a single network configuration as there are many different settings and conditions to consider. Therefore, a U-net-based architecture was built and its hyperparameters were defined, to accomplish the aim of this internship. Besides, due to low GPU memory and computational capacity restrictions, several strategies were implemented to overcome this limitation, which may have affected the results. In the experimental work, three datasets were used: datasets one, two, and three.

Dataset one was used to gain experience and sensitivity on the subject to be developed during the project. Thus, a proof of concept was carried out using this dataset. It contained CT images of the abdominal region and the corresponding masks of the spleen. To build an optimised network, 20 experiments were carried out using different strategies, resulting in 20 networks. The network that originated the best results was tested on five unseen CT images. By analysing the results obtained, it is possible to verify that the transfer learning method improved the internal validation DC, which is in line with the literature [51]. This was expected because the final optimised network merged previously learned features from the pre-trained network. By giving smaller image patches to the network during the training process, the variability introduced to the system by the adjacent anatomical structures and background is reduced, and the network is not able to learn all the necessary information, along with the loss of contextual information. However, to tackle GPU capacity constraints associated with training the network on volumetric images, the patch-wise segmentation method had to be adopted, which reduced the contextual information given to the system, which may have impacted the segmentation performance. Overall, there was a correct identification and segmentation of the spleen on the resulting segmentations when applying the best network. However, in some cases, the liver, and heart were incorrectly segmented, possibly due to the similarity with the spleen in terms of the CT intensity values. The intensity normalisations were intended to attenuate this issue, yet

it was challenging to reach a consensus on the best intensity normalisation scale to be used in the training and optimisation process of the neural network for the CT scans. It is essential to remark that there were only five test images, which makes it difficult to represent a variety of cases. In addition, with these results, it is difficult to generalise the best intensity normalisation range to apply to the CT scans, as well as the best strategy to be implemented.

Furthermore, it was hypothesised that increasing the shift applied to the spleen inside the patches, would originate higher internal validation DC because it introduced a higher variability through the addition of new anatomical structures. Consequently, higher DC on the test segmentation results were expected, since the spleen shift simulates better what happens in reality. Thus, and despite not being a good practice in literature, for research purposes, all optimised networks were tested (Table D2, appendix), corroborating the hypothesis, since the increase of the shift value led to a gradual increase of the DC results.

Simultaneously, it was verified that when using the transfer learning method, the new network being initialised from a previously trained network, the probability for the network to diverge was approximately one-third of the obtained, when being initialised from scratch, because the initial weights are no longer random. The training process of the CNN is not a deterministic method, meaning that the initial weights are random, which can often lead to divergence in the optimisation process, and sometimes, never being able to converge to an optimal value.

Regarding training time, this factor remained constant for all implemented strategies, since the size and number of patches provided to the CNN were not altered.

After acquiring the fundamental competencies from the previous experiment, it was possible to develop and test CNN configurations and implement distinct strategies capable of identifying and segmenting malignant lesions in whole-body [^{18}F]FDG PET/CT images from different cancer types (dataset two). This task is a greater challenge for CNN training, due small size of the lesions compared with the image size, the high level of noise in the PET images, and the inclusion of different cancer types. In addition, there is high heterogeneity amongst lesions in shape, size, metabolic activity and localisation. In parallel, there are regions with high normal uptake, such as the brain and heart, as well as the elimination ways (intestines, kidneys, urinary bladder, and ureters) which can make CNN optimisation difficult.

Analysing the results obtained from the dataset two, it is possible to verify that reducing the number of filters of the U-net architecture did not influence significantly the DC. Therefore, it was opted to start using a U-net configuration with fewer filters than the one used in dataset one. Subsequently, another hypothesis was called into question, assessing whether the patch

size significantly affected the DC. By reducing the patch size to 32×32×32, it is possible to ascertain that the DC reduced from 0.27 (median) to 0.13 (median), which was expected since the patch had less contextual information. It is important to refer that was not possible to train a network with patch sizes bigger than 48×48×48, due to GPU memory constraints.

The final approach was to add a CT channel to the PET images, expecting an improvement in the internal validation DC, by providing supplementary anatomical information to the U-net. However, the internal validation DC (median of 0.21) was not as high as expected. The CT taken with the PET is not intended for diagnosis, using a lower radiation dose, which results in a lower quality CT. In addition, in many lesions, there are no morphological alterations detected by the CT. Hence, these are plausible reasons for the lower DC when adding a supplementary CT channel. Therefore, the approach that was favourable to the training process was U-net which used patches of size 48×48×48 with half the number of filters from the original U-net (dataset one).

For the inference test, it was expected that by removing the border voxels, the noise associated with the border would be reduced. Contradictory results were obtained, thus no conclusion can be taken.

Overall, the DC values obtained in dataset two were unsatisfactory. This was expected due to the small dataset size and the high heterogeneity of the data. Analysing the DC represented in the boxplots (Figures 4.2 and 4.3) and Table D3 (appendix), it is visible that some PET images gave rise to DC values of zero. An in-depth analysis of each corresponding image allowed us to gather a panoply of characteristics, in particular, the suspected lesions having low SUV, and being small compared to the large body volume. In addition, the network segmented the normal high radiopharmaceutical uptake organs, such as the brain, heart, and kidneys. Besides the issues related to the dataset variability and small size, we cannot exclude that with a different training process and the U-net architecture better results may have been obtained.

We hypothesised that a U-net trained for each specific pathology may originate better results. Hence, we decided to optimise a U-net specific for the identification and segmentation of lymphoma lesions using [^{18}F]FDG PET/CT images. Moreover, it was hypothesised that using different PET scanners and having different nuclear medicine physicians identifying and delineating the ground truths labels influences the results. Thus, dataset three contains only [^{18}F]FDG PET/CT images from patients diagnosed with lymphoma. First and foremost, it is critical to understand that lymphoma segmentation is a challenging task since the pathology can be spread across the entire body with different localisations, shapes and sizes. Furthermore, the

lymphoma lesions are complicated to discriminate from normal high-uptake regions near the intestines, for instance.

In dataset three, the results for the internal validation were higher in comparison to the previous dataset, achieving the highest median DC of 0.72 and the lowest of 0.44, from the several experiments performed. Besides the lesser heterogeneity of dataset three compared to dataset two, its size was larger (115 images for training against 40 in dataset two), which obviously may have contributed also to the higher DC. Analysing the internal validation DC of all trained networks for each applied strategy, the best strategy was the one that was fed with patches of dimension 64×64×64 by removing a U-net layer (median of 0.72). When the network was fed with 48×48×48 patches centred in each lesion, the internal validation median DC was 0.65, which was a satisfactory result, according to the literature. However, it was hoped that a better outcome could be achieved. Therefore, since cropping the images in small patches reduced the contextual information, few patches containing normal high uptake organs and other anatomical structures were fed to the neural network. By adding random patches containing parts of the body, it was expected to increase diversity, and, consequently, the DC. However, the DC decreased, possibly due to low contextual information due to the small size of the patches. Due to the low GPU capacity, the only manner to increase the patch size was by removing a layer from the U-net, allowing the increase of the patch size to 64×64×64. Once again, it was expected that by increasing even further the patch size, the DC would increase. Thus, a different criterion was applied to select the patches to be included, decreasing the number of patches being fed to the network, although with a bigger dimension of 96×96×96 (strategy S_5, Table 3.11). The median DC obtained (0.65) was lower compared to the previous strategy result (0.72), possibly due to the accentuated decrease in the number of patches fed to the network, being provided only 466 for training, instead of 1865 patches provided in strategy S_4 (Table 3.11).

Analysing the DC results on the test set provided by Champalimaud Foundation, it was possible to verify that the median DC decreased to 0.43, when compared to the internal validation median DC (0.72). This DC decrease may be due to multiple factors. In a superficial analysis, it is plausible that the decrease of the DC is related to the different sources of the PET acquisition equipment and clinical protocols from Champalimaud Foundation and the difference in the identification and segmentation of the ground truth segmentations performed by different nuclear medicine physicians.

By looking at the results represented in the boxplots in Figure 4.6, it is possible to observe that, once again, some images give rise to DC values of zero. In these volumetric images,

normal high-uptake regions and organs such as the brain, heart, parotid glands, intestine, kidneys, urinary bladder, and ureters were segmented with frequency, since the network was not able to distinguish them from the lymphoma lesions. In the best DC results, the lesions were mostly located in the head and neck, and were large and uniform, making them easily identifiable.

Additionally, one more plausible reason for the low DC on the test set may be justified by the identification and segmentation performed by different observers that were given to the model to train the U-net. Simultaneously, the MICCAI PET data were acquired in different equipment with different clinical protocols, affecting the images' characteristics. These experiments show the importance of having coherent and comparable segmented data for training, internal validation, and testing. Therefore, heterogeneity between train/internal validation sets and the test set may also result in a decrease in DC.

The U-net that achieved the best performance in the identification and segmentation of unseen images was the one that used transfer learning for spleen segmentation on CT images. This was expected since the patches contained more information, the CT scans were smaller, and the spleen was a “static” organ with few variations, facilitating the U-net task. However, for the conditions available, the 3D U-net used for the segmentation of lymphoma lesions achieved satisfactory performance, not being too far from the internal validation DC reported in the literature.

This study has two main limitations already identified in the previous chapters: the limited hardware resources available (GPU memory, and computational capacity) for CNN architecture used; and the lack of large and representative datasets with segmentations. Even though the CNN architecture used is one of the best for image segmentation, it would be of interest to test other CNN architectures and different configurations of the U-net used. This is also a limitation of the current study.

Chapter 6 Conclusions

This internship aimed to apply, optimise and assess the feasibility of deep learning-based techniques for the automatic identification and segmentation of malignant lesions in whole-body [^{18}F]FDG PET/CT images. Hence, the main conclusion is that it was not possible to build an optimised U-net good enough to be used autonomously in clinical practice. All U-net built throughout this project are not robust enough, not being able to replace the current methods, since it was not verified a good agreement between the manual segmentation and the one provided by the U-net. Despite that, these U-net may be used as an aiding tool for clinicians, to minimise the time spent on lesions' identification and segmentation. It was observed that U-net-based segmentation takes less than a second to be performed, whereas manual segmentation may take tens or hundreds of minutes if the patient has several lesions.

In the near future**,** fully automatic segmentation methods, using deep learning may be an excellent tool for clinical practice for lesions' identification and segmentation. However, clinical supervision is expectable to be indispensable for the verification and correction of the segmentation.

Chapter 7 Future Work

Complementary approaches should be considered in future studies, where there are several critical steps to be taken, to improve the identification and segmentation of [^{18}F]FDG PET images, namely the addition of a second channel containing the correspondent CT to the PET images (PET/CT); to increase the dataset of [^{18}F]FDG PET images with accurate segmentations; to develop of a U-net for multiclass segmentation (more than two classes). In this case, regions of normal high uptake, such as the brain, heart, kidneys, and urinary bladder may also be identified and automatically segmented. Finally, to do experiments with the application of post-processing techniques on the resulting segmentations, such as morphological operations, to increase lesions segmentation accuracy.

Bibliographic References

[1] Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A. and Bray, F. (2021) Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: a cancer journal for clinicians*, *71*(3), 209 249.

[2] World Health Organization. (2022) Cancer. Retrieved on 13 August, in 2022, from https://www.who.int/

[3] Basu, S., Hess, S., Braad, P. E. N., Olsen, B. B., Inglev, S. and Høilund-Carlsen, P. F. (2014) The basic principles of FDG-PET/CT imaging. *PET clinics*, *9*(4), 355-370.

[4] Foster, B., Bagci, U., Mansoor, A., Xu, Z. and Mollura, D.J. (2014) A review on segmentation of positron emission tomography images. *Computers in biology and medicine*, *50*, pp.76-96.

[5] Hatt, M., Lee, J. A., Schmidtlein, C. R., Naqa, I. E., Caldwell, C., De Bernardi, E. and Kirov, A. S. (2017) Classification and evaluation strategies of auto-segmentation approaches for PET: Report of AAPM task group No. 211. *Medical physics*, *44*(6), e1- e42.

[6] Boellaard, R., Delgado-Bolton, R., Oyen, W. J., Giammarile, F., Tatsch, K., Eschner, W. and Krause, B. J. (2015) FDG PET/CT: EANM procedure guidelines for tumour imaging: version 2.0. *European journal of nuclear medicine and molecular imaging*, *42*(2), 328-354.

[7] Hess, S., Blomberg, B. A., Zhu, H. J., Høilund-Carlsen, P. F. and Alavi, A. (2014) The pivotal role of FDG-PET/CT in modern medicine. *Academic radiology*, *21*(2), 232-249.

[8] Basu, S., Kwee, T. C., Surti, S., Akin, E. A., Yoo, D. and Alavi, A. (2011) Fundamentals of PET and PET/CT imaging. *Annals of the New York Academy of Sciences*, *1228*(1), 1- 18.

[9] Surasi, D. S., Bhambhvani, P., Baldwin, J. A., Almodovar, S. E. and O'Malley, J. P. (2014) 18F-FDG PET and PET/CT patient preparation: a review of the literature. *Journal of nuclear medicine technology*, *42*(1), 5-13.

[10] Blanc-Durand, P., Van Der Gucht, A., Schaefer, N., Itti, E. and Prior, J. O. (2018) Automatic lesion detection and segmentation of 18F-FET PET in gliomas: a full 3D U- Net convolutional neural network study. *PLoS One*, *13*(4), e0195798.

[11] Malhotra, P., Gupta, S., Koundal, D., Zaguia, A. and Enbeyle, W. (2022) Deep Neural Networks for Medical Image Segmentation. *Journal of Healthcare Engineering*, *2022*.

[12] Hatt, M., Laurent, B., Ouahabi, A., Fayad, H., Tan, S., Li, L. and Visvikis, D. (2018) The first MICCAI challenge on PET tumour segmentation. *Medical image analysis*, *44*, 177-195.

[13] Weisman, A. J., Kieler, M. W., Perlman, S., Hutchings, M., Jeraj, R., Kostakoglu, L. and Bradshaw, T. J. (2020) Comparison of 11 automated PET segmentation methods in lymphoma. *Physics in Medicine & Biology*, *65*(23), 235019.

[14] Liu, X., Song, L., Liu, S. and Zhang, Y. (2021) A review of deep-learning-based medical image segmentation methods. *Sustainability*, *13*(3), 1224.

[15] Jacobson, O., Kiesewetter, D. O. and Chen, X. (2015) Fluorine-18 radiochemistry, labeling strategies and synthetic routes. *Bioconjugate chemistry*, *26*(1), 1-18.

[16] Kawada, K., Iwamoto, M. and Sakai, Y. (2016) Mechanisms underlying 18F-fluorodeoxyglucose accumulation in colorectal cancer. *World Journal of Radiology*, *8*(11), 880.

[17] Macheda, M. L., Rogers, S. and Best, J. D. (2005) Molecular and cellular regulation of glucose transporter (GLUT) proteins in cancer. *Journal of cellular physiology*, *202*(3), 654-662.

[18] Griffeth, L. K. (2005) Use of PET/CT scanning in cancer patients: technical and practical considerations. In *Baylor University Medical Center Proceedings* (Vol. 18, No. 4, pp. 321-330). Taylor & Francis.

[19] Zhuang, H. and Codreanu, I. (2015) Growing applications of FDG PET-CT imaging in non-oncologic conditions. *Journal of biomedical research*, *29*(3), 189.

[20] Mosci, C., Kumar, M., Smolarz, K., Koglin, N., Stephens, A. W., Schwaiger, M., and Mittra, E. S. (2016) Characterization of physiologic 18F FSPG uptake in healthy volunteers. *Radiology*, *279*(3), 898-905.

[21] Abouzied, M. M., Crawford, E. S. and Nabi, H. A. (2005) 18F-FDG imaging: pitfalls and artifacts. *Journal of nuclear medicine technology*, *33*(3), 145-155.

[22] Lin, W. Y., Wang, K. B., Tsai, S. C. and Sun, S. S. (2009) Unexpected accumulation of F-18 FDG in the urinary bladder after bladder irrigation and retrograde filling with sterile saline: a possible pitfall in PET examination. *Clinical nuclear medicine*, *34*(9), 560-563.

[23] Liu, Y. (2014) Diagnostic role of fluorodeoxyglucose positron emission tomography-computed tomography in prostate cancer. *Oncology Letters*, *7*(6), 2013-2018.

[24] Meignan, M., Hutchings, M. and Schwartz, L. H. (2015) Imaging in lymphoma: The key role of fluorodeoxyglucose-positron emission tomography. *The Oncologist*, *20*(8), 890-895.

[25] Zhou, T., Ruan, S. and Canu, S. (2019) A review: Deep learning for medical image segmentation using multi-modality fusion. *Array*, *3*, 100004.

[26] Altaf, F., Islam, S. M., Akhtar, N. and Janjua, N. K. (2019) Going deep in medical image analysis: concepts, methods, challenges, and future directions. *IEEE Access*, *7*, 99540-99572.

[27] LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998) Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.

[28] Krizhevsky, A., Sutskever, I. and Hinton, G. E. (2012) Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, *25*.

[29] Simonyan, K. and Zisserman, A. (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[30] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D. and Rabinovich, A. (2015) Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).

[31] He, K., Zhang, X., Ren, S. and Sun, J. (2016) Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[32] Long, J., Shelhamer, E. and Darrell, T. (2015) Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3431-3440).

[33] Ronneberger, O., Fischer, P. and Brox, T. (2015) U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.

[34] Çiçek, Ö., Abdulkadir, A., Lienkamp, S. S., Brox, T. and Ronneberger, O. (2016) 3D U-Net: learning dense volumetric segmentation from sparse annotation. In *International conference on medical image computing and computer-assisted intervention* (pp. 424-432). Springer, Cham.

[35] Milletari, F., Navab, N. and Ahmadi, S. A. (2016) V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 fourth international conference on 3D vision (3DV)* (pp. 565-571). IEEE.

[36] Renard, F., Guedria, S., Palma, N. D. and Vuillerme, N. (2020) Variability and reproducibility in deep learning for medical image segmentation. *Scientific Reports*, *10*(1), 1-16.

[37] Isensee, F., Jaeger, P. F., Kohl, S. A., Petersen, J. and Maier-Hein, K. H. (2021) nnU-Net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature methods*, *18*(2), 203-211.

[38] Blanc-Durand, P., Jégou, S., Kanoun, S., Berriolo-Riedinger, A., Bodet-Milin, C., Kraeber-Bodéré, F. and Itti, E. (2021) Fully automatic segmentation of diffuse large B cell lymphoma lesions on 3D FDG-PET/CT for total metabolic tumour volume prediction using a convolutional neural network. *European Journal of Nuclear Medicine and Molecular Imaging*, *48*(5), 1362-1370.

[39] Jiang, C., Chen, K., Teng, Y., Ding, C., Zhou, Z., Gao, Y. and Zhang, J. (2022) Deep learning–based tumour segmentation and total metabolic tumour volume prediction in the prognosis of diffuse large B-cell lymphoma patients in 3D FDG-PET images. *European Radiology*, 1-12.

[40] Pinochet, P., Eude, F., Becker, S., Shah, V., Sibille, L., Toledano, M. N. and Decazes, P. (2021) Evaluation of an automatic classification algorithm using convolutional neural networks in oncological positron emission tomography. *Frontiers in Medicine*, *8*, 628179.

[41] Zhou, S. K., Greenspan, H., Davatzikos, C., Duncan, J. S., Van Ginneken, B., Madabhushi, A. and Summers, R. M. (2021) A review of deep learning in medical imaging: Imaging traits, technology trends, case studies with progress highlights, and future promises. *Proceedings of the IEEE*.

[42] Domingues, I., Pereira, G., Martins, P., Duarte, H., Santos, J. and Abreu, P. H. (2020) Using deep learning techniques in medical imaging: a systematic review of applications on CT and PET. *Artificial Intelligence Review*, *53*(6), 4093-4160.

[43] Ahmad, H. A., Yu, H. J. and Miller, C. G. (2014) Medical imaging modalities. In *Medical imaging in clinical trials* (pp. 3-26). Springer, London.

[44] Niyas, S., Pawan, S. J., Kumar, M. A. and Rajan, J. (2021) Medical image segmentation using 3d convolutional neural networks: A review. *arXiv preprint arXiv:2108.08467*.

[45] Sharma, N. and Aggarwal, L. M. (2010) Automated medical image segmentation techniques. *Journal of medical physics/Association of Medical Physicists of India*, *35*(1), 3

[46] Minaee, S., Boykov, Y. Y., Porikli, F., Plaza, A. J., Kehtarnavaz, N. and Terzopoulos, D. (2021) Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*.

[47] Weisman, A. J., Kieler, M. W., Perlman, S. B., Hutchings, M., Jeraj, R., Kostakoglu, L. and Bradshaw, T. J. (2020) Convolutional neural networks for automated PET/CT detection of diseased lymph node burden in patients with lymphoma. *Radiology: Artificial Intelligence*, *2*(5), e200016.

[48] Janiesch, C., Zschech, P. and Heinrich, K. (2021) Machine learning and deep learning. *Electronic Markets*, *31*(3), 685-695.

[49] LeCun, Y., Bengio, Y. and Hinton, G. (2015) Deep learning. *nature*, *521*(7553), 436-444.

[50] Sahiner, B., Pezeshk, A., Hadjiiski, L. M., Wang, X., Drukker, K., Cha, K. H. and Giger, M. L. (2019) Deep learning in medical imaging and radiation therapy. *Medical physics*, *46*(1), e1-e36.

[51] Hesamian, M. H., Jia, W., He, X. and Kennedy, P. (2019) Deep learning techniques for medical image segmentation: achievements and challenges. *Journal of digital imaging*, *32*(4), 582-596.

[52] Oliveira, F. P., Faria, D. B., Costa, D. C., Castelo-Branco, M. and Tavares, J. M. R. (2018) Extraction, selection and comparison of features for an effective automated computer-aided diagnosis of Parkinson's disease based on [123I] FP-CIT SPECT images. *European journal of nuclear medicine and molecular imaging*, *45*(6), 1052-1062.

[53] Oliveira, F., Leuzy, A., Castelhano, J., Chiotis, K., Hasselbalch, S. G., Rinne, J. and Castelo-Branco, M. (2018) Data driven diagnostic classification in Alzheimer's disease based on different reference regions for normalization of PiB-PET images and correlation with CSF concentrations of Aβ species. *NeuroImage: Clinical*, *20*, 603-610.

[54] Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M. P. and Iyengar, S. S. (2018) A survey on deep learning: Algorithms, techniques, and applications. *ACM Computing Surveys (CSUR)*, *51*(5), 1-36.

[55] Razzak, M. I., Naz, S. and Zaib, A. (2018) Deep learning for medical image processing: Overview, challenges and the future. *Classification in BioApps*, 323-350.

[56] Shrestha, A. and Mahmood, A. (2019) Review of deep learning algorithms and architectures. *IEEE access*, *7*, 53040-53065.

[57] Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O. and Farhan, L. (2021) Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. *Journal of big Data*, *8*(1), 1-74.

[58] Sarvamangala, D. R. and Kulkarni, R. V. (2021) Convolutional neural networks in medical image understanding: a survey. *Evolutionary intelligence*, 1-22.

[59] Kang, X., Song, B. and Sun, F. (2019) A deep similarity metric method based on incomplete data for traffic anomaly detection in IoT. *Applied Sciences*, *9*(1), 135.

[60] Yamashita, R., Nishio, M., Do, R. K. G. and Togashi, K. (2018) Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, *9*(4), 611-629.

[61] El Jurdi, R., Petitjean, C., Honeine, P., Cheplygina, V. and Abdallah, F. (2021) High-level prior-based loss functions for medical image segmentation: A survey. *Computer Vision and Image Understanding*, *210*, 103248.

[62] Zhang, J. (2019) Gradient descent based optimisation algorithms for deep learning models training. *arXiv preprint arXiv:1903.03614*.

[63] Yaqub, M., Feng, J., Zia, M. S., Arshid, K., Jia, K., Rehman, Z. U. and Mehmood, A. (2020) State-of-the-art CNN optimiser for brain tumour segmentation in magnetic resonance images. *Brain Sciences*, *10*(7), 427.

[64] Balas, V. E., Kumar, R. and Srivastava, R. (Eds.). (2020) Recent trends and advances in artificial intelligence and internet of things (pp. 389-425). Springer.

[65] Kriegeskorte, N. and Golan, T. (2019) Neural network models and deep learning. *Current Biology*, *29*(7), R231-R236.

[66] Siddique, N., Paheding, S., Elkin, C. P. and Devabhaktuni, V. (2021) U-net and its variants for medical image segmentation: A review of theory and applications. *Ieee Access*, *9*, 82031-82057.

[67] Isensee, F. (2020) *From Manual to Automated Design of Biomedical Semantic Segmentation Methods* (Doctoral dissertation).

[68] Simpson, A. L., Antonelli, M., Bakas, S., Bilello, M., Farahani, K., Van Ginneken, B. and Cardoso, M. J. (2019) A large annotated medical image dataset for the development and evaluation of segmentation algorithms. *arXiv preprint arXiv:1902.09063*.

[69] Antonelli, M., Reinke, A., Bakas, S., Farahani, K., Kopp-Schneider, A., Landman, B. A. and Cardoso, M. J. (2022) The medical segmentation decathlon. *Nature Communications*, *13*(1), 1-13.

[70] Constantino, C. S., Oliveira, F. P., Silva, M., Oliveira, C., Castanheira, J. C., Silva, Â. and Costa, D. C. (2021) Are lesion features reproducible between 18F-FDG PET/CT images when acquired on analog or digital PET/CT scanners?. *European Radiology*, *31*(5), 3071-3079.

[71] Constantino, C. S. (2019) *Reproducibility Study of Tumour Biomarkers Extracted from Positron Emission To-mography Images with 18F-Fluorodeoxyglucose* (Master thesis).

[72] Gatidis S. and Kuestner T. (2022) A whole-body FDG-PET/CT dataset with manually annotated tumour lesions (FDG-PET-CT-Lesions) [Dataset]. The Cancer Imaging Archive.

[73] Constantino, C. S., Leocádio, S., Oliveira, F. P., Silva, M., Oliveira, C., Castanheira, J. C., Silva, Â., Vaz, S., Teixeira, R., Neves, M., Lúcio, P., João, C., and Costa, D. C. (2022) Evaluation of fully and semiautomatic segmentation methods on [18F]FDG PET images from patients with lymphoma: influence on tumor characterization. *Submitted.*

[74] Wang, R., Lei, T., Cui, R., Zhang, B., Meng, H. and Nandi, A. K. (2022) Medical image segmentation using deep learning: A survey. *IET Image Processing*, *16*(5), 1243-1267.

[75] MathWorks. (2022) *softmaxLayer*. Retrieved August 10, 2022, from https://www.mathworks.com/

[76] Yeung, M., Sala, E., Schönlieb, C. B. and Rundo, L. (2022) Unified Focal loss: Generalising Dice and cross entropy-based losses to handle class imbalanced medical image segmentation. *Computerized Medical Imaging and Graphics*, *95*, 102026.

[77] Li, Z., Kamnitsas, K. and Glocker, B. (2020) Analysing overfitting under class imbalance in neural networks for image segmentation. *IEEE transactions on medical imaging*, *40*(3), 1065-1077.

[78] MathWorks. (2022) *dicePixelClassificationLayer*. Retrieved August 10, 2022, from https://www.mathworks.com/

[79] MathWorks. (2022) *trainingOptions*. Retrieved August 10, 2022, from https://www.mathworks.com/

[80] Zhao, X., Li, L., Lu, W. and Tan, S. (2018) Tumor co-segmentation in PET/CT using multi-modality fully convolutional neural network. *Physics in Medicine & Biology*, *64*(1), 015011.

[81] Li, L., Lu, W., Tan, Y. and Tan, S. (2019) Variational PET/CT tumor co-segmentation integrated with PET restoration. *IEEE transactions on radiation and plasma medical sciences*, *4*(1), 37-49.

[82] Zou, K. H., Warfield, S. K., Bharatha, A., Tempany, C. M., Kaus, M. R., Haker, S. J. and Kikinis, R. (2004) Statistical validation of image segmentation quality based on a spatial overlap index1: scientific reports. *Academic radiology*, *11*(2), 178-189.

[83] MathWorks. (2022) *Parallel Computing Toolbox*. Retrieved August 6, 2022, from https://www.mathworks.com/

[84] MathWorks. (2022) *MATLAB GPU Computing Support for NVIDIA CUDA-Enabled GPUs*. Retrieved August 6, 2022, from https://www.mathworks.com/

[85] NVIDIA Developer. (2022) *CUDA Toolkit Documentation*. Retrieved August 6, 2022, from https://www.developer.nvidia.com/

[86] NVIDIA Developer. (2022) *CUDA Toolkit*. Retrieved August 6, 2022, from https://www.developer.nvidia.com/

[87] NVIDIA Developer. (2022) *About CUDA*. Retrieved August 6, 2022, from https://www.developer.nvidia.com/

[88] MathWorks. (2022) *Computer Vision Toolbox*. Retrieved August 6, 2022, from https://www.mathworks.com/

[89] MathWorks. (2022) *Deep Learning Toolbox*. Retrieved August 6, 2022, from https://www.mathworks.com/

[90] MathWorks. (2022) *Image Processing Toolbox*. Retrieved August 6, 2022, from https://www.mathworks.com/

[91] Jimmy Shen (2022) *Tools for NIfTI and ANALYSE image*. MATLAB Central File Exchange. Retrieved August 6, 2022, from https://www.mathworks.com/

[92] Stephen23 (2022) *Natural-Order Filename Sort*. MATLAB Central File Exchange. Retrieved August 6, 2022, from https://www.mathworks.com/

[93] Dang N. H. Thanh (2021) *Image Segmentation Quality Scores*. MATLAB Central File Exchange. Retrieved August 6, 2022, from https://www.mathworks.com/

Appendix A

Table A 1: 3D U-net configuration detailed description.

ANALYSIS RESULT

#	Name	Type	Activations	Learnables	Total Learnables
1	input 96×96×96×1 images	3-D Image Input	96×96×96×1	-	0
2	en1_conv1 64 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	96×96×96×64	Weights 3×3×3×1×64 Bias 1×1×1×64	1792
3	en1_bn Batch normalization	Batch Normalization	96×96×96×64	Offset 1×1×1×64 Scale 1×1×1×64	128
4	en1_relu1 ReLU	ReLU	96×96×96×64	-	0
5	en1_conv2 64 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	96×96×96×64	Weights 3×3×3×64×64 Bias 1×1×1×64	110656
6	en1_relu2 ReLU	ReLU	96×96×96×64	-	0
7	en1_maxpool 2×2×2 max pooling with stride [2 2 2] and padding 'same'	3-D Max Pooling	48×48×48×64	-	0
8	en2_conv1 128 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	48×48×48×128	Weights 3×3×3×64×128 Bias 1×1×1×128	221312
9	en2_bn Batch normalization	Batch Normalization	48×48×48×128	Offset 1×1×1×128 Scale 1×1×1×128	256
10	en2_relu1 ReLU	ReLU	48×48×48×128	-	0
11	en2_conv2 128 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	48×48×48×128	Weights 3×3×3×128×128 Bias 1×1×1×128	442496
12	en2_relu2 ReLU	ReLU	48×48×48×128	-	0
13	en2_maxpool 2×2×2 max pooling with stride [2 2 2] and padding 'same'	3-D Max Pooling	24×24×24×128	-	0
14	en3_conv1 256 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	24×24×24×256	Weights 3×3×3×128×256 Bias 1×1×1×256	884992
15	en3_bn Batch normalization	Batch Normalization	24×24×24×256	Offset 1×1×1×256 Scale 1×1×1×256	512
16	en3_relu1 ReLU	ReLU	24×24×24×256	-	0
17	en3_conv2 256 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	24×24×24×256	Weights 3×3×3×256×256 Bias 1×1×1×256	1769728
18	en3_relu2 ReLU	ReLU	24×24×24×256	-	0
19	en3_maxpool 2×2×2 max pooling with stride [2 2 2] and padding 'same'	3-D Max Pooling	12×12×12×256	-	0
20	en4_conv1 512 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	12×12×12×512	Weights 3×3×3×256×512 Bias 1×1×1×512	3539456
21	en4_bn Batch normalization	Batch Normalization	12×12×12×512	Offset 1×1×1×512 Scale 1×1×1×512	1024
22	en4_relu1 ReLU	ReLU	12×12×12×512	-	0
23	en4_conv2 512 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	12×12×12×512	Weights 3×3×3×512×512 Bias 1×1×1×512	7078400
24	en4_relu2 ReLU	ReLU	12×12×12×512	-	0
25	en4_maxpool 2×2×2 max pooling with stride [2 2 2] and padding 'same'	3-D Max Pooling	6×6×6×512	-	0
26	de5_conv1 1024 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	6×6×6×1024	Weights 3×3×3×512×1024 Bias 1×1×1×1024	14156800
27	de5_relu1 ReLU	ReLU	6×6×6×1024	-	0
28	de5_conv2 512 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	6×6×6×512	Weights 3×3×3×1024×512 Bias 1×1×1×512	14156288
29	de5_relu2 ReLU	ReLU	6×6×6×512	-	0
30	de5_transconv 512 2×2×2 transposed 3D convolutions with stride [2 2 2] and cropping [0 0 0; 0 0 0]	Transposed Convolutio...	12×12×12×512	Weights 2×2×2×512×512 Bias 1×1×1×512	2097664
31	concat4 Concatenation of 2 inputs along dimension 4	Concatenation	12×12×12×1024	-	0
32	de4_conv1 512 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	12×12×12×512	Weights 3×3×3×1024×512 Bias 1×1×1×512	14156288
33	de4_relu1 ReLU	ReLU	12×12×12×512	-	0
34	de4_conv2 256 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	12×12×12×256	Weights 3×3×3×512×256 Bias 1×1×1×256	3539200
35	de4_relu2 ReLU	ReLU	12×12×12×256	-	0
36	de4_transconv 256 2×2×2 transposed 3D convolutions with stride [2 2 2] and cropping [0 0 0; 0 0 0]	Transposed Convolutio...	24×24×24×256	Weights 2×2×2×256×256 Bias 1×1×1×256	524544
37	concat3 Concatenation of 2 inputs along dimension 4	Concatenation	24×24×24×512	-	0
38	de3_conv1 256 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	24×24×24×256	Weights 3×3×3×512×256 Bias 1×1×1×256	3539200
39	de3_relu1 ReLU	ReLU	24×24×24×256	-	0
40	de3_conv2 128 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	24×24×24×128	Weights 3×3×3×256×128 Bias 1×1×1×128	884864
41	de3_relu2 ReLU	ReLU	24×24×24×128	-	0
42	de3_transconv 128 2×2×2 transposed 3D convolutions with stride [2 2 2] and cropping [0 0 0; 0 0 0]	Transposed Convolutio...	48×48×48×128	Weights 2×2×2×128×128 Bias 1×1×1×128	131200
43	concat2 Concatenation of 2 inputs along dimension 4	Concatenation	48×48×48×256	-	0
44	de2_conv1 128 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	48×48×48×128	Weights 3×3×3×256×128 Bias 1×1×1×128	884864
45	de2_relu1 ReLU	ReLU	48×48×48×128	-	0
46	de2_conv2 64 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	48×48×48×64	Weights 3×3×3×128×64 Bias 1×1×1×64	221248
47	de2_relu2 ReLU	ReLU	48×48×48×64	-	0
48	de2_transconv 64 2×2×2 transposed 3D convolutions with stride [2 2 2] and cropping [0 0 0; 0 0 0]	Transposed Convolutio...	96×96×96×64	Weights 2×2×2×64×64 Bias 1×1×1×64	32832
49	concat1 Concatenation of 2 inputs along dimension 4	Concatenation	96×96×96×128	-	0
50	de1_conv1 64 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	96×96×96×64	Weights 3×3×3×128×64 Bias 1×1×1×64	221248
51	de1_relu1 ReLU	ReLU	96×96×96×64	-	0
52	de1_conv2 64 3×3×3 convolutions with stride [1 1 1] and padding 'same'	Convolution	96×96×96×64	Weights 3×3×3×64×64 Bias 1×1×1×64	110656
53	de1_relu2 ReLU	ReLU	96×96×96×64	-	0
54	convLast 2 1×1×1 convolutions with stride [1 1 1] and padding [0 0 0; 0 0 0]	Convolution	96×96×96×2	Weights 1×1×1×64×2 Bias 1×1×1×2	130
55	softmax softmax	Softmax	96×96×96×2	-	0
56	output Generalized Dice loss	Dice Pixel Classificatio...	96×96×96×2	-	0

Appendix B

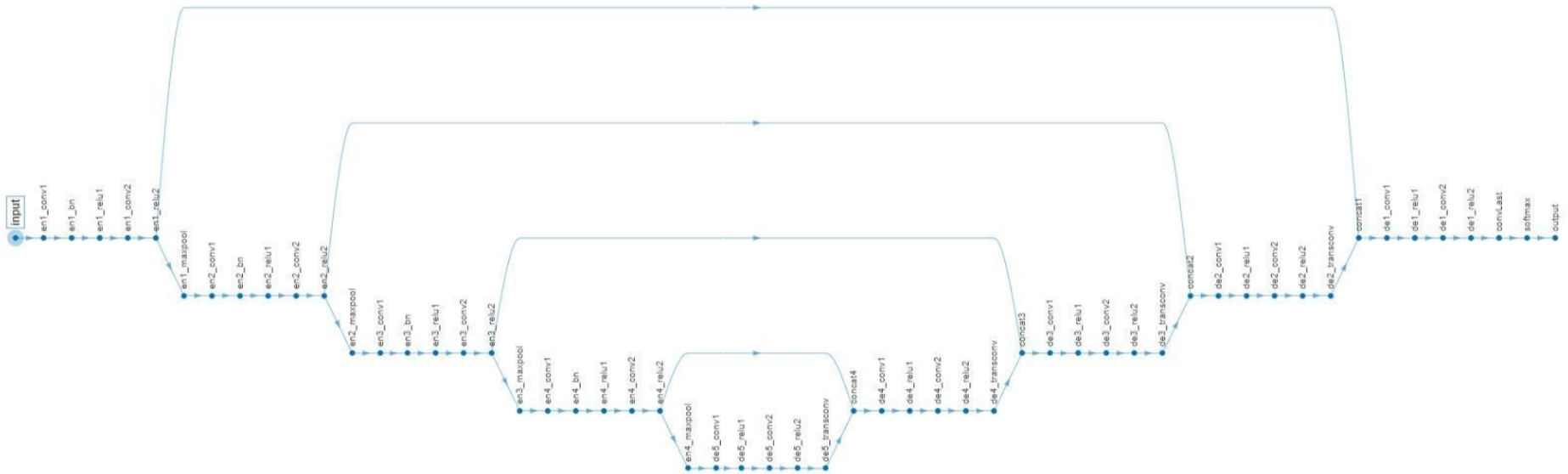

Figure B 1: 3D U-net architecture constituted by 56 layers.

Appendix C

Table C 1: Data distribution according to suspect of primary tumour and sex for training, internal validation, and external test sets.

Suspect of primary tumour	Sex	Training set	Internal validation set	External test set
Colon	M	2	1	1
	F	1	-	-
Endometrium	F	1	1	1
Oesophagus	M	1	-	-
Tongue	F	1	-	-
Breast	F	9	2	2
Other	M	1	-	-
	F	-	-	1
Hidden	F	1	-	-
Pancreas	F	1	1	1
	M	1	-	-
Lung	M	7	2	2
	F	1	1	1
Rectum	M	1	-	1
Kidneys	M	1	-	-
Urothelium	M	1	-	-
Gallbladder	F	1	-	-
Non-Hodgkin lymphoma	M	2	1	1
	F	3	1	1
Hodgkin lymphoma	M	2	1	1
	F	2	1	1

Appendix D

In the following tables (Table D1 to D6), the DC, sensitivity, and PPV obtained in the respective experiments are shown. The specificity and accuracy were not included since in the datasets used their value is almost 1 in all cases. This is due to the very small number of voxels of the ROI to segment compared to all the other voxels of the image.

Dataset 1

Table D 1: DC, sensitivity, and PPV of all CT images from the external test set obtained through strategy S_{18}.

Applied strategy	DC	Sensitivity	PPV
	0.49	0.91	0.33
	0.57	0.81	0.44
S_{18}	0.46	0.65	0.35
	0.78	0.90	0.69
	0.91	0.88	0.94

Spleen test images

- Image 1, 3 and 4 – in the resulting segmentation, the fully automatic algorithm wrongly identified and segmented the liver and heart, possibly due to similarities in tissue intensity. In addition, there were some regions on the spleen that were not identified and segmented.
- Image 2 – the liver was identified and segmented in the resulting segmentation.
- Image 5 – almost all spleen was correctly identified and segmented. To note that a little portion of the wall next to the intestine was identified and segmented.

Table D 2: DC of the external test set for all applied strategies.

Applied strategy	Mean DC ± STD	Median DC (IQR)
S_1	0.57 ± 0.26	0.60 (0.51)
S_2	0.66 ± 0.19	0.71 (0.31)
S_3	0.65 ± 0.21	0.67 (0.42)
S_4	0.56 ± 0.17	0.57 (0.26)
S_5	0.32 ± 0.21	0.27 (0.36)
S_6	0.71 ± 0.10	0.68 (0.19)
S_7	0.76 ± 0.12	0.76 (0.22)
S_8	0.74 ± 0.15	0.81 (0.28)
S_9	0.80 ± 0.079	0.80 (0.14)
S_{10}	0.76 ± 0.21	0.84 (0.38)
S_{11}	0.58 ± 0.14	0.55 (0.26)
S_{12}	0.86 ± 0.066	0.87 (0.12)
S_{13}	0.77 ± 0.20	0.85 (0.34)
S_{14}	0.78 ± 0.11	0.76 (0.19)
S_{15}	0.81 ± 0.14	0.85 (0.26)
S_{16}	0.74 ± 0.15	0.75 (0.29)
S_{17}	0.75 ± 0.18	0.81 (0.35)
S_{18}	0.64 ± 0.20	0.57 (0.37)
S_{19}	0.79 ± 0.12	0.81 (0.21)
S_{20}	0.57 ± 0.19	0.57 (0.37)

Dataset 2

Table D 3: DC, sensitivity, and PPV of all PET images from the external test set obtained through strategy S_2, with a step of 20 and by varying border values between zero and six.

Step					**20**				
Border		**0**			**3**			**6**	
Test Volumes	**DC**	**Sensitivity**	**PPV**	**DC**	**Sensitivity**	**PPV**	**DC**	**Sensitivity**	**PPV**
Patient 1	0.38	0.34	0.44	0.40	0.42	0.38	0.41	0.48	0.36
Patient 2	0.00	0.00	0.00	0.00	0.00	0.00	0.045	0.24	0.025
Patient 3	0.20	0.46	0.12	0.19	0.57	0.12	0.18	0.62	0.11
Patient 4	0.63	0.71	0.56	0.60	0.75	0.49	0.55	0.77	0.43
Patient 5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Patient 6	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Patient 7	0.31	0.21	0.64	0.32	0.23	0.52	0.33	0.26	0.45
Patient 8	0.24	0.14	0.86	0.30	0.19	0.81	0.34	0.22	0.78
Patient 9	0.67	0.54	0.90	0.73	0.63	0.88	0.74	0.65	0.85
Patient 10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Patient 11	0.21	0.12	0.93	0.22	0.13	0.82	0.22	0.14	0.55
Patient 12	0.18	0.11	0.52	0.28	0.19	0.56	0.32	0.22	0.53
Patient 13	0.84	0.78	0.91	0.89	0.90	0.87	0.88	0.93	0.84
Patient 14	0.46	0.87	0.31	0.39	0.89	0.25	0.31	0.90	0.19

Table D 4: DC, sensitivity, and PPV of all PET images from the external test set obtained through strategy S_2, with a step of 30 and by varying border values between zero and six.

Step					**30**				
Border		**0**			**3**			**6**	
Test Volumes	**DC**	**Sensitivity**	**PPV**	**DC**	**Sensitivity**	**PPV**	**DC**	**Sensitivity**	**PPV**
Patient 1	0.43	0.43	0.43	0.42	0.50	0.36	0.44	0.55	0.36
Patient 2	0.039	0.14	0.022	0.038	0.20	0.021	0.057	0.34	0.031
Patient 3	0.21	0.57	0.13	0.19	0.61	0.11	0.16	0.56	0.092
Patient 4	0.63	0.71	0.56	0.58	0.72	0.48	0.57	0.74	0.46
Patient 5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Patient 6	0.00	0.00	0.00	0.015	0.042	0.010	0.019	0.053	0.012
Patient 7	0.32	0.24	0.49	0.32	0.27	0.38	0.34	0.32	0.36
Patient 8	0.26	0.16	0.76	0.32	0.20	0.73	0.34	0.23	0.69
Patient 9	0.69	0.57	0.88	0.74	0.67	0.83	0.75	0.71	0.79
Patient 10	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Patient 11	0.21	0.12	0.78	0.24	0.14	0.65	0.23	0.15	0.49
Patient 12	0.29	0.20	0.54	0.32	0.24	0.49	0.31	0.24	0.47
Patient 13	0.81	0.76	0.86	0.87	0.90	0.84	0.86	0.93	0.80
Patient 14	0.29	0.80	0.18	0.26	0.90	0.15	0.24	0.91	0.14

Dataset 3

Table D 5: DC, sensitivity, and PPV of all PET images from the external test set obtained through strategy S_4 with no removal of border voxels.

		Border = 0		
Best trained network	**Test Volumes**	**DC**	**Sensitivity**	**PPV**
S_4	Patient 1	0.80	0.71	0.91
	Patient 2	0.81	0.76	0.87
	Patient 3	0.60	0.47	0.81
	Patient 4	0.021	0.032	0.016
	Patient 5	0.060	0.11	0.042
	Patient 6	0.25	0.25	0.26
	Patient 7	0.012	0.048	0.010
	Patient 8	0.00	0.00	0.00
	Patient 9	0.78	0.69	0.89
	Patient 10	0.24	0.22	0.26
	Patient 11	0.78	0.83	0.73
	Patient 12	0.77	0.69	0.86
	Patient 13	0.24	0.52	0.16
	Patient 14	0.43	0.51	0.38
	Patient 15	0.00	0.00	0.00
	Patient 16	0.64	0.96	0.48
	Patient 17	0.30	0.68	0.20
	Patient 18	0.71	0.64	0.81
	Patient 19	0.66	0.84	0.54
	Patient 20	0.19	0.18	0.20
	Patient 21	0.010	0.010	0.014
	Patient 22	0.27	0.26	0.27
	Patient 23	0.75	0.71	0.78
	Patient 24	0.88	0.86	0.90
	Patient 25	0.014	0.010	0.036
	Patient 26	0.89	0.88	0.91
	Patient 27	0.45	0.86	0.31
	Patient 28	0.65	0.71	0.60
	Patient 29	0.017	0.010	0.31

Patient 30	0.50	0.39	0.69
Patient 31	0.10	0.22	0.065
Patient 32	0.46	0.44	0.47
Patient 33	0.32	0.80	0.20
Patient 34	0.51	0.88	0.36
Patient 35	0.69	0.66	0.74
Patient 36	0.075	0.24	0.044
Patient 37	0.79	0.88	0.71
Patient 38	0.041	0.38	0.022
Patient 39	0.52	0.77	0.39
Patient 40	0.83	0.91	0.77
Patient 41	0.18	0.96	0.10
Patient 42	0.50	0.43	0.60
Patient 43	0.54	0.63	0.48
Patient 44	0.24	0.32	0.19
Patient 45	0.79	0.87	0.72
Patient 46	0.46	0.42	0.52
Patient 47	0.89	0.93	0.86
Patient 48	0.31	0.78	0.19
Patient 49	0.74	0.96	0.60
Patient 50	0.72	0.72	0.71
Patient 51	0.33	0.46	0.26
Patient 52	0.026	0.39	0.013
Patient 53	0.41	0.54	0.33
Patient 54	0.63	0.95	0.47
Patient 55	0.15	0.71	0.082
Patient 56	0.37	0.52	0.29
Patient 57	0.00	0.00	0.00
Patient 58	0.12	0.76	0.068
Patient 59	0.10	0.069	0.22
Patient 60	0.36	0.90	0.23
Patient 61	0.67	0.86	0.55
Patient 62	0.55	0.99	0.38
Patient 63	0.00	0.071	0.00
Patient 64	0.00	0.00	0.00

	Patient 65	0.59	0.48	0.76

Table D 6: DC, sensitivity, and PPV of all PET images from the external test set obtained through strategy S_4 with the removal of five border voxels.

		Border = 5		
Best trained network	**Test Volumes**	**DC**	**Sensitivity**	**PPV**
	Patient 1	0.81	0.72	0.91
	Patient 2	0.79	0.74	0.85
	Patient 3	0.60	0.47	0.81
	Patient 4	0.021	0.030	0.016
	Patient 5	0.045	0.084	0.031
	Patient 6	0.24	0.22	0.25
	Patient 7	0.00	0.016	0.00
	Patient 8	0.00	0.00	0.00
	Patient 9	0.77	0.69	0.88
	Patient 10	0.23	0.20	0.25
	Patient 11	0.76	0.79	0.73
	Patient 12	0.77	0.69	0.87
	Patient 13	0.28	0.52	0.19
	Patient 14	0.46	0.51	0.42
S_4	Patient 15	0.00	0.00	0.00
	Patient 16	0.75	0.96	0.62
	Patient 17	0.58	0.68	0.51
	Patient 18	0.73	0.64	0.85
	Patient 19	0.65	0.85	0.53
	Patient 20	0.19	0.18	0.20
	Patient 21	0.010	0.010	0.018
	Patient 22	0.30	0.26	0.37
	Patient 23	0.76	0.71	0.81
	Patient 24	0.88	0.86	0.89
	Patient 25	0.013	0.010	0.037
	Patient 26	0.90	0.88	0.92
	Patient 27	0.46	0.85	0.32
	Patient 28	0.73	0.72	0.74
	Patient 29	0.015	0.010	0.33

Patient 30	0.52	0.41	0.71
Patient 31	0.080	0.16	0.053
Patient 32	0.52	0.44	0.63
Patient 33	0.34	0.84	0.21
Patient 34	0.59	0.88	0.44
Patient 35	0.69	0.65	0.74
Patient 36	0.085	0.24	0.051
Patient 37	0.79	0.89	0.72
Patient 38	0.051	0.39	0.027
Patient 39	0.56	0.76	0.44
Patient 40	0.87	0.91	0.83
Patient 41	0.19	0.96	0.10
Patient 42	0.51	0.44	0.60
Patient 43	0.57	0.63	0.52
Patient 44	0.28	0.33	0.24
Patient 45	0.78	0.87	0.71
Patient 46	0.44	0.39	0.51
Patient 47	0.89	0.93	0.86
Patient 48	0.33	0.75	0.21
Patient 49	0.75	0.96	0.62
Patient 50	0.73	0.72	0.75
Patient 51	0.33	0.44	0.27
Patient 52	0.028	0.39	0.015
Patient 53	0.41	0.54	0.33
Patient 54	0.69	0.96	0.54
Patient 55	0.19	0.68	0.11
Patient 56	0.38	0.48	0.31
Patient 57	0.00	0.00	0.00
Patient 58	0.13	0.76	0.069
Patient 59	0.11	0.066	0.26
Patient 60	0.41	0.89	0.27
Patient 61	0.69	0.85	0.58
Patient 62	0.57	0.99	0.40
Patient 63	0.011	0.17	0.010
Patient 64	0.00	0.00	0.00

Patient 65	0.57	0.46	0.76