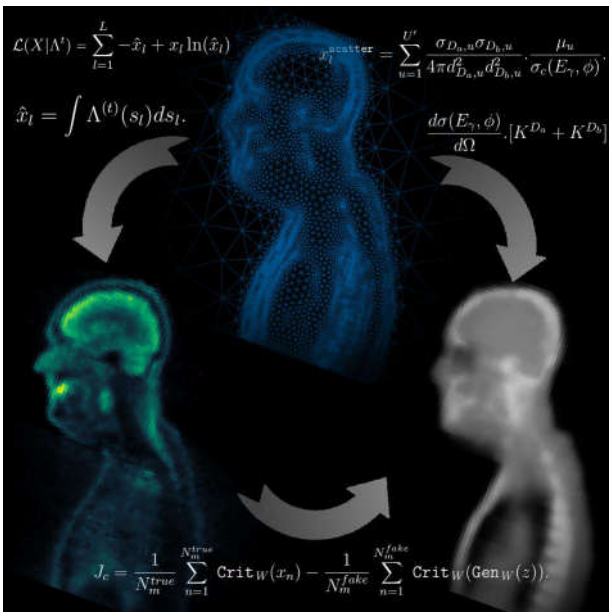
Towards Reduced Dose Positron Emission Tomography Imaging Using Sparse Sampling and Machine Learning

Ramiro German Rodriguez Colmeiro

Thèse
de doctorat
de l'UTT

Ramiro Germán RODRÍGUEZ COLMEIRO

Towards Reduced Dose Positron Emission Tomography Imaging Using Sparse Sampling and Machine Learning



Champ disciplinaire :
Sciences pour l'Ingénieur

2021TROY0015 | **Année 2021**

Thèse en cotutelle avec l'Universidad Tecnológica Nacional - CABA - Argentine

UTN.BA
UNIVERSIDAD TECNOLÓGICA NACIONAL
FACULTAD REGIONAL BUENOS AIRES

utt
UNIVERSITÉ DE TECHNOLOGIE
TROYES

THESE

pour l'obtention du grade de

DOCTEUR
de l'UNIVERSITE DE TECHNOLOGIE DE TROYES

en SCIENCES POUR L'INGENIEUR

Spécialité : OPTIMISATION ET SURETE DES SYSTEMES

présentée et soutenue par

Ramiro Germán RODRÍGUEZ COLMEIRO

le 18 mai 2021

Towards Reduced Dose Positron Emission Tomography Imaging Using Sparse Sampling and Machine Learning

JURY

Mme M. FOULADIRAD	PROFESSEURE DES UNIVERSITES	Présidente
M. C. A. DELRIEUX	PROFESOR TITULAR	Rapporteur
M. F. MORAIN-NICOLIER	PROFESSEUR DES UNIVERSITES	Rapporteur
M. T. GHARBI	PROFESSEUR DES UNIVERSITES	Examinateur
Mme S. RUAN	PROFESSEURE DES UNIVERSITES	Examinatrice

Personnalités invitées

M. T. GROSGES	PROFESSEUR DES UNIVERSITES	Directeur de thèse
M. D. MINSKY	PROFESOR TITULAR	Directeur de thèse
M. C. VERRASTRO	PROFESOR ASOCIADO	Directeur de thèse

To my beloved Ruth...

Quisiera empezar agradeciendo a los integrantes del grupo AR-PET de CNEA y la división de medicina nuclear del Hospital de Clínicas "José de San Martín", sin los cuales nada de lo que se presenta aquí podría haber sido posible. A Lucio Martínez Garbino, que me ayudo a dar mis primeros pasos en la instrumentación nuclear. A Damián Pirlo, que se tomó el trabajo de escuchar y ayudar en todas mis ideas poco convencionales. A Ariel Hernández, cuyo entusiasmo por la ciencia me animó a explorar el mundo del *machine learning*. A Matias Scremin y Federico De La Cruz Arbizu, que me ayudaron a traer a la tierra muchas de las ideas descritas en esta tesis, sin tenerle (mucho) miedo a las locuras que pensábamos y posibilitaron la aplicación de muchas de las ideas propuestas. A Flavio Gino Bertolini, por hacer que todas nuestras simulaciones de movimientos y mecanizaciones se convirtieran en una realidad y por su arrolladora alegría, que siempre levantaba al grupo. A Juan Alarcón por su conocimiento y las numerosas charlas sobre los temas más aleatorios de la ciencia (y la vida). A Liliana Questa, que sin su ayuda no podríamos haber llevado a cabo ninguna de las mediciones del AR-PET. A ellos, y al resto de los integrantes, que sin su apoyo jamás hubiera terminado la tesis antes de fin de año.

También debo nombrar a los integrantes del grupo GIAR, los culpables de mi camino por la ciencia y amigos personales, como Sebastián Jaremczuk, Joaquín "Ingelosofo" Toranzo Calderón, Emiliano Statello, Alejandro Álvarez, y muchos más. Entre ellos se destacan Juan Carlos Gómez, que me mostró el mundo de la investigación durante mi cursada de la carrera de grado y fue mi primer tutor dentro de un grupo de investigación. También a Claudio Verrastro, que me incluyó dentro del proyecto AR-PET, podría decirse que mi tesis (y el proyecto AR-PET en general) son consecuencias directas de su visión y genialidad. También debo nombrar a Pablo De Cesare, que dirigió uno de mis primeros proyectos en el mundo de las cosas poco posibles y hoy en día es un amigo más.

Finalmente y no menos importante, a mi familia y amigos de la vida. A Ruth, por soportar cuatro viajes y casi dos años enteros de relaciones a larga distancia. Sé que sin su humor y sagacidad no hubiera podido mantener la salud mental durante mi doctorado. A Ale y Giova, sin ustedes claramente estaría perdido (¿o no estaría perdido?). A Gabo, por años de vicio constante, sin ellos probablemente hubiera terminado antes la tesis. A mi mamá y papá, por su ayuda durante mis interminables años de estudio, sin su amparo nunca hubiera llegado hasta acá. A mis abuelos Quiquí, Neli, Lilu y en especial a mi abuelo Hernando, culpable sembrar la ingeniería en mi desde chico, sé que hoy estarían orgullosos de ver a donde llegué (excepto Neli, que si me va ver, y ayer cumplió sus 90 añitos ¡Feliz Cumple!).

Abstract

In this thesis the reduction of the patient radiation dose in screening Positron Emission Tomography (PET) studies is explored. Due to its inherent radiation risk and cost, PET imaging is normally used only on patients with diagnosticated or suspected cancer. However much of the radiation dose absorbed by patient can be reduced if the acquisition and image reconstruction methods are improved. This thesis analyses three aspects of PET imaging, which can reduce the patient dose: the data acquisition, the image reconstruction and the attenuation map generation.

The first part of the thesis is dedicated to the PET scanner technology. Two optimization techniques are developed for a novel low-cost and low-dose scanner, the AR-PET scanner. First a photomultiplier selection and placement strategy is created, improving the energy resolution by 5%. The second work focuses on the localization of gamma events on solid scintillation crystals. The method, based on neural networks and a single flood acquisition, results in an increase of 23% of the AR-PET detector's sensitivity area and enabling depth estimation.

In the second part, the PET image reconstruction on mesh support is studied. Mesh methods can reduce the amount of sample points required to describe an image or volume, which is translated in the reduction of the dimensionality of the reconstruction problem. A mesh-based reconstruction algorithm is proposed which uses a series of 2D meshes to describe the 3D radiotracer distribution. It is shown that with this reconstruction strategy the number of sample points can be reduced by one order of magnitude without loosing accuracy and enabling parallel mesh optimization.

Finally the attenuation map generation using deep neural networks is explored. A neural network is trained to learn the mapping from non attenuation corrected $^{18}F-FDG$ PET images to a synthetic Computerized Tomography. The attenuation map generation is tested on 133 samples, achieving a mean absolute error of 103 ± 18 HU. The attenuation correction is measured by means of attenuation sinograms, obtaining a mean absolute error of 1% with a standard deviation below 8%. Moreover, these values are obtained in the presence of data form multiple sources and modalities.

With these approaches, this thesis lays a base for a low-cost and low-dose PET screening system that can reduce the absorbed patient dose by up to 40%, dispensing the need of a computed tomography image in exchange of an artificial attenuation map.

Resumen

En esta tesis se explora la reducción de la dosis de radiación en estudios de Tomografía por Emisión de Positrones (PET). Debido a su costo y riesgo de radiación inherente, normalmente, el estudio PET es solamente utilizado en pacientes con cáncer diagnosticado o sospechado. Sin embargo gran parte de la dosis de radiación absorbida por el paciente puede ser reducida si los métodos de adquisición y reconstrucción de imágenes son optimizados. Esta tesis analiza tres aspectos del estudio PET que pueden reducir la dosis absorbida por el paciente: la adquisición de datos, la reconstrucción de imágenes y la generación de mapas de atenuación.

La primera parte de la tesis es dedicada a la tecnología del escáner PET. Dos técnicas son desarrolladas para un prototipo de escáner de bajo costo y baja dosis, el escáner AR-PET. Primero se crea una estrategia de selección y posicionamiento de fotomultiplicadores, incrementando la resolución energética en un 5%. El segundo trabajo se concentra en la localización de eventos gamma en cristales de centelleo sólidos. Dicho método, basado en redes neuronales y una única adquisición de campo inundado, resulta en un incremento del 23% del área sensible de los detectores del AR-PET y posibilita la detección de profundidad de interacción.

En la segunda parte se estudian los métodos de reconstrucción de imagen PET sobre mallas. Los métodos de mallas permiten reducir la cantidad de puntos de muestreo requeridos para describir una imagen o volumen, lo cual se traduce en una reducción de la dimensionalidad del problema de reconstrucción. Un algoritmo de reconstrucción basado en mallas es propuesto, el cual usa una serie de mallas 2D que describen la distribución 3D del radiotrazador. Se demuestra que con dicha reconstrucción el numero de puntos de muestreo se puede reducir en un orden de magnitud, sin perder exactitud y posibilitando la optimización de las mallas de manera paralela.

Finalmente se explora la generación de imágenes de atenuación usando redes neuronales profundas. Se entrena una red neuronal para aprender el mapeo desde una imagen PET $^{18}F - FDG$, sin corrección de atenuación, a una imagen de Tomografía Computada sintética. El mapa de atenuación generado es probado sobre 133 muestras, obteniendo un error medio absoluto de 103 ± 18 HU. La corrección de atenuación es medida por medio de sinogramas de atenuación, alcanzando un error medio absoluto del 1% con un desvío estándar de 8%. Estos valores son logrados con datos de múltiples fuentes y modalidades.

Con estos enfoques, esta tesis pone una base para un sistema PET de bajo costo y baja dosis que puede reducir la dosis de radiación sobre el paciente hasta un 40%, reemplazando la necesidad de una tomografía computada por un mapa de atenuación artificial.

Résumé

Cette thèse étudie le problème de la dose de radiation dans les études de tomographie par émission de positons (PET). La PET scan est normalement utilisée uniquement pour des patients avec un cancer diagnostiqué ou soupçonné, en raison de son coût et risque de radiation. Toutefois beaucoup de la radiation absorbée par le patient peut être réduite si l'acquisition et la reconstruction d'images sont améliorées. Cette thèse analyse trois aspects de la PET scan: l'acquisition de données, la reconstruction d'images et la génération d'images d'atténuation.

La première partie de cette thèse est dédiée à la technologie de la scan PET. Deux techniques d'optimisation sont développées pour une petite dose PET scanner à bas coût, l'AR-PET scanner. D'abord une stratégie de sélection et placement de photo-multiplicateurs est proposé, augmentant la résolution énergétique de 5%. De plus un travail a été effectué concernant la localisation d'impacts des photons gamma dans les cristaux solides de scintillation. Cette méthode est basée sur des réseaux de neurones artificiels et une acquisition unique de champ. L'application résulte en une augmentation de 23% de la superficie sensible et permet l'estimation de la profondeur d'impact.

Dans la deuxième partie de cette thèse, la reconstruction de l'image PET avec l'aide de maillages est étudiée. Les méthodes de maillage peuvent réduire le nombre de points d'échantillonnage requis pour la description d'une image ou d'un volume. Cette réduction est traduite en un réduction de la dimensionnalité du problème de reconstruction d'image PET. Un algorithme de reconstruction basé sur un maillage est proposé. L'algorithme utilise une série de maillages 2D pour décrire la distribution 3D du radiotraceur.Les résultats montrent que le nombre de points d'échantillonnage est réduit d'un ordre de grandeur, sans perte de précision et rendent possible l'optimisation et la parallélisation des maillages.

Enfin, la génération de l'image d'atténuation au moyen de réseaux de neurones artificiels profondes est explorée. Le réseau de neurones est entrainé à apprendre d'une transformation d'images PET $^{18}F - FDG$ sans correction d'atténuation en une image de tomodensitométrie synthétique. La génération d'images d'atténuation est analysée avec 133 échantillons, réalisant une erreur absolue moyenne de 103 ± 18 HU. La correction d'atténuation est testée avec des sinogrammes d'atténuation, produisant une erreur moyenne de 1% et un écart type de 8%. De plus, ces valeurs sont obtenus en présence de données de plusieurs sources et modalités.

Avec ces approches, cette thèse pose une base pour une petite dose PET scanner à bas coût, qui peut réduire la radiation absorbée par le patient jusqu'à 40%, dispensant le besoin d'images de tomodensitométrie et en échangeant par une image d'atténuation artificielle.

Contents

Introduction

The Positron Emission Tomography (PET) remains as one of the most powerful tools in the detection and follow up of cancer, metabolism measurement and drug research. Today, PET scanners are widely used in medicine using the ^{18}F-fluoro-2-deoxy-D-glucose (FDG) tracer, mostly for patient with suspected or diagnosticated cancer. Also the use of PET studies as a screening study in asymptomatic patients for early cancer detection is being studied [1, 2]. However the use of this technique carries a reasonable risk to the patient. This risk comes from the radiation dose absorbed by the patient that can be estimated around 10 `mSv` to 18 `mSv` for a screening study [3, 4]. This dose is considered as a *substantial dose* for the International Commission on Radiation Protection (ICRP) [5], and is one of the points risen against the use of PET studies as a screening study on asymptomatic patients [6]. But the radiation dose involved in a PET scan is not entirely from the FDG, around 1.3 `mSv` to 8.0 `mSv` of the dose comes from the Computed Tomography (CT) scan (if using a low dose CT scan) on an average screening study [3, 4, 7]. The CT scan is mistakenly regarded *light-dose* study by the physicians and avoiding it when possible is recommended [8]. Thus, reduction of the PET dose and CT dose in particular are active fields in PET imaging [9, 10]. Further reduction of dose could be achieved by completely removing the CT scan from the PET study, without loosing accuracy and image quality. In this way, the idea of reconstructing the Attenuation Map (AM), without using the CT scan is being explored using increasingly complex models [11, 12], and more recently using machine learning methods based on Non Attenuation Corrected (NAC) PET data [13–15] or other image modalities [16, 17].

In this scope, the contribution of this thesis is the application of Machine Learning (ML) techniques and sparse sampling to reduce the patient dose on PET scans due to CT imaging. More specifically, the methods to directly achieve AMs from NAC PET data are studied, enabling the use of computationally costly methods. Such is achieved by using both a mesh representation of the image and by generating high quality AM priors using ML methods. For this, a reconstruction scheme is first build on an adaptive sparse representation map using high quality meshes in order to apply higher complexity reconstruction models, reducing the asymptotic computational costs and enabling the generation or refinement of AMs and enhancing PET image quality in general. Second the Deep Convolutional Networks (DCN) and Generative Adversarial Networks (GAN) are studied, to infer AMs obtaining results higher than the state of the art. By generating high quality priors with ML techniques and reducing the computational cost of complex reconstruction models, it is expected to reduce the cost and improve the convergence of image reconstruction and correction methods based on Single Scatter Simulation techniques [18], resolution models [19] and Monte Carlo simulations [20].

Outline of the manuscript

The goal of this thesis paves the way to a low dose PET scan, eliminating the need of a CT or gamma transmission scan. The contributions are based on simulations, public datasets and measurements from a low dose PET scanner prototype. This thesis is divided in five main chapters. In chapter 1 the architecture of a PET scanner and image reconstruction algorithms are introduced. Then in chapter 2, the AR-PET prototype and its construction details are presented, which aim to enhance count rate and accuracy of continuous crystal detectors. Following, in chapter 3, the mesh representation as a sparse sampling technique for PET images is presented. The basis of the reconstruction framework is shown and the ability of meshes to automatically adapt and describe a PET images is demonstrated. The meshes represent the PET image without losing quality and reducing the number of parameters when compared to a voxelized representation. The chapter is finalized by outlining the application of Single Scatter Simulation (SSS) on an optimized representation such as a mesh. Finally the chapter 4 is devoted to the application of DCNs and GANs to the problem of AM generation from a NAC PET image. The network architectures and dataset construction are developed, showing the capabilities and shortcomings of both techniques. The code to reproduce these results on the open datasets is also provided.
In the last chapter, the global conclusions of the thesis are presented and the results and projection of its contributions are discussed.

The results presented in this thesis are supported by the following peer-reviewed papers:

- "Heuristic Method for Photo-detectors Localization over Continuous Crystal Scintillation Cameras" [21]
- "Event Localization in Continuous Crystal Scintillation Cameras using Distribution Matching Neural Networks" [22]
- "Reconstruction of Positron Emission Tomography images using adaptive sliced remeshing strategy" [23]
- "Multimodal Brain Tumor Segmentation Using 3D Convolutional Networks" [24]
- "Towards a Whole Body [$^{1}8F$] FDG Positron Emission Tomography Attenuation Correction Map Synthesizing using Deep Neural Networks"[1] [25]

[1]Code can be found in: https://github.com/RawthiL/PET_DeepAttCorrection

1 Positron Emission Tomography

In this chapter the basis of the PET technique is introduced. The state of the art of PET scanners, their technologies and capabilities are described, followed by the basic algorithms for the PET image reconstruction algorithms and the correction techniques.

1.1 Introduction

The objective of PET imaging devices is to visualize molecular activity in living tissues. The technique is based on the detection of photon pairs created by radioactive tracers injected in the patient. There are many types of tracers that are designed to correlate with different types of biological characteristics. The FDG tracer is a modified glucose with ^{18}F, which decays with positron emission (β^+), and accumulates on living cells. The cells consume the FDG through the glucose transporter protein at the cell membrane and is phosphorylated to FDG-6-phosphate by hexokinase and ATP. Once inside the cell the FDG-6 is not metabolized any further and remains trapped. Since the malignant cancer cells of various types present an increased glucose transport [26] they present an increased uptake compared to healthy cells of the same tissue. Also the relative uptake of the cell indicates the growth-rate of the tumor [27]. The FDG PET scan is useful for differential diagnostic, follow-up and recurrence of different cancers. The whole body imaging technique is a common technique for the detection of metastases in distant regions or lymph nodes in patients with history of cancer affection [28].

1.1.1 Basic concepts of PET scanners

The PET scanner works measuring pairs of 511 `keV` gamma photons in coincidence, the pair of photons originates with the interaction of the $\beta+$ decay from the radionuclide with an electron in the nearby area (normally a $\beta+$ travels less than 1`mm`). The positron-electron interaction generates the pair of gamma photons which travel in an almost opposite directions. Since they are not at rest when they interact, they are emitted in an angle of $180.0° \pm 0.2°$. In figure 1.1 the annihilation and detection of a coincidence in a PET system is presented.

These photons, and other from other disintegrations, are detected by a pair of detectors and tested for coincidence. If a pair of events is detected within a given time window (that depends on the geometry and accuracy of the scanner) they are marked as a valid coincidence. In figure 1.2 several types of coincidences are presented, the expected type of coincidence is the *true* (figure 1.2(a)), both photons come from the same annihilation, *random* (figure 1.2(b)), the events come from different annihilation

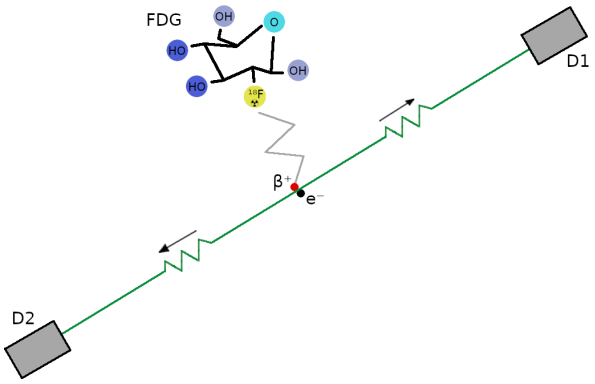


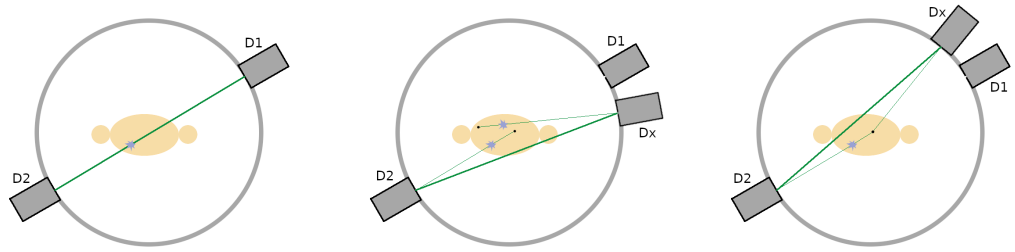
Figure 1.1: Detection of PET coincidence. The ^{18}F in the FDG decays releasing a β^+ positron which travels a short distance until interacting with an electron e^-. The gamma photons emitted by the annihilation are detected by the D1 and D2 detectors.

that happened within the time window, or *scattered* (figure 1.2(c)), one or both events have suffered scattering when interacting with matter along their path to the detector. All coincidences that are not *true* will add noise to the system.

In order to estimate (or reconstruct) the volumetric distribution of the tracer, a large amount of coincidences must be acquired, typically around tenths of millions (10^7). These coincidences describe lines along the *Field of View* (FoV) of the scanner which are called *Lines of Response* (LoRs). Each LoR is described by the spacial position and time of the detection of both gamma photons. If the LoR parameters are described with enough precision, the position of the disintegration can be directly obtained. Nevertheless several factor impede to achieve this direct reconstruction of the position. First the uncertainty in positioning of the events in the detectors is around 2 mm in commercial scanners, which translates into a blurring of the LoR location. Second the time of each event must be known with a precision that is not yet achievable. In a scanner with $\varnothing = 800$ mm the photon takes 1.33×10^{-9} s to reach the detector if emitted from the center. To directly measure the position of the disintegration with a precision of 4 mm[1] a time change of 26.67×10^{-12} s must be measured. Nevertheless the ability to measure the time of interaction (at a given precision) helps to reduce the probable section LoR where the event was emitted. The scanners that are able to measure this time are called Time of Flight (ToF) scanners and have a time resolution of 400×10^{-12} s [29]. Given the current technological limitations, the images must be obtained trough a mathematical reconstruction process which is described in section 1.3.

The Signal to Noise Ratio (SNR) of the reconstructed image is proportional to the square root of the total number of *true* coincidences ($\sqrt{L}$). A high number of detected coincidences improves the scanner ability to detect small lesions. The number of coincidences that are acquired depends on the acquisition time T, the longer the study the more coincidences collected, but the comfort of the patient limits this parameter to

[1] modern systems achieve an image of voxel size around $3 - 5$ mm



(a) (b) (c)

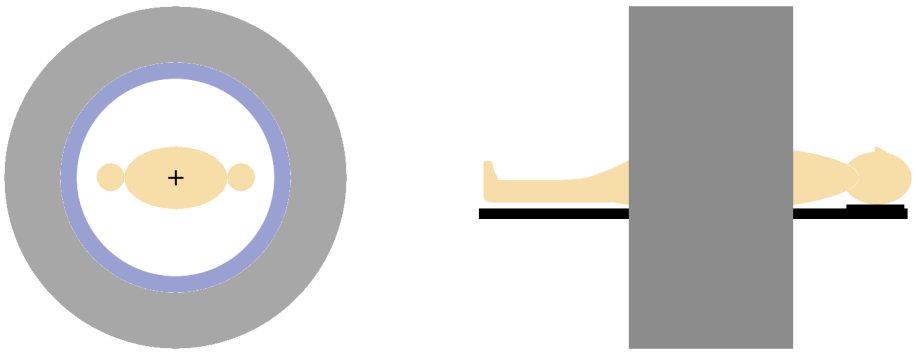
Figure 1.2: Types of coincidences detected in a PET scanner: true coincidence (a), random coincidence (b) and scattered coincidence (c). The thin green line represents the real gamma photon path, the bold green line represents the detected path.

approximately 1200 s. Also the total activity A injected to the patient is proportional to L, this parameter depends on the type of tracer used and the type of study but it is limited by the acceptability of the radiactive dose absorbed by the patient. An other technological limitation of the scanners is the photo-peak efficiency υ of the detectors. This factor represents the probability of interaction of a 511 keV gamma photon with the detector. A higher υ means a higher fraction of the total emissions detected. The total coverage of the scanner also limits L, since the scanner does not fully surround the patient only a fraction G of the total emissions is detected. Finally, the influence of the ToF measurement in L is proportional to $\frac{\varnothing_{\text{pt.}}}{\text{SR}_{\text{ToF}}}$, where $\varnothing_{\text{pt.}}$ is the diameter of the patient and SR_{ToF} is the spacial time resolution [30]. Apart from the scanner construction and technology, the patient influences L. The patient's attenuation and scatter fraction reduces the amount of events detected to a fraction $k_{\text{pt.}}$. The SNR can be then expressed as [31]:

$$\text{SNR} \approx \sqrt{\frac{k_{\text{pt.}}\, A\, G\, \upsilon^2\, T\, \varnothing_{\text{pt.}}}{\text{SR}_{\text{ToF}}}}. \tag{1.1}$$

1.1.2 Current technology of a PET scanner

The PET scanners are sophisticated devices that require high speed signal processing electronics, large data handling capabilities and complex mathematical models. They have been in use in clinics and research for decades and they have gone through a great number of improvements since the first prototypes. The constructional elements of the scanner can be roughly divided into three main groups: geometry and detector material, the detection and amplification electronics and the processing firmware/software.



(a) (b)

Figure 1.3: Front (a) and lateral (b) views of a basic cylindrical PET scanner geometry. In grey the scanner structure, in light blue the detector surface and in black the patient couch.

Geometry and Detector Materials

The geometry and detector materials of a PET scanner define the maximum SNR that can be obtained. They define three main aspects of the equation 1.1. First the geometric form of the scanner fixes the coverage factor G, larger detectors translate into more captured coincidences and the detector material defines the photo-peak efficiency υ and time of flight capabilities, which then define the resolution $\mathrm{SR}_{\mathtt{ToF}}$.
All commercial have different types of detector technologies and capabilities but they share a common geometric structure. Their designs are normally a short cylinder of $\varnothing \approx 800$ mm with a height of 150 mm to 300 mm. They arrange the detectors around the surface of the cylinder following its form. The patient is scanned in horizontal position and in order to acquire a full body image, the couch moves inside the detector. In figure 1.3 a basic PET geometry is displayed.

The detectors arranged in the cylinder are commonly built of small pixels of a scintillation material. The size of these pixels is normally $D_{\mathrm{px}} \approx 4 \times 4$ mm^2 with a thickness of $D_{\mathrm{thkns}} \approx \times 15 - 25$ mm depending mostly on the material. The pixelated design is normally chosen over large continuous crystals for its higher count rate and easier event positioning logic [32]. The material of these pixels is normally a scintillator crystal which produces a short-lived light glow when a gamma photon interacts with it. The amount of light and the time it takes to decay play an important role, they define the energy resolution, count rate and ToF capabilities of the scanner. The light yield is given in [Photons/MeV] units, its value represents the amounts of visible photons created as a function of the energy of the gamma photon. The higher is this value, the better is the energy resolution of the scanner. Also, high light yield

values enable the scanner to discriminate direct photons from scattered photons. The decay time is the time that the crystal remains exited (emitting visible photons) after interacting with a gamma photon. If this time is too long the probability of pile-up with an other incoming gamma photon increases, resulting in a loss of count rate. Also this time value defines the time resolution of the system: short scintillation times means better time definition of the event. Currently the most common materials are the Lutetium-yttrium oxyorthosilicate (LYSO) or Bismuth Germanium Oxide (BGO), in the table 1.1 the characteristic values of some common scintillation materials are presented.

Property	Material		
	NaI(TI)	BGO	LYSO
Linear Attenuation Coefficient (μ) [$\mathtt{cm}^{-1}$]	0.34	0.92	0.86
Light yield (% of NaI(TI))	100	15	80
Decay constant [ns]	230	300	41

Table 1.1: Reference values of principal characteristics of common scintillation crystals used in commercial PET scanners.

Signal Processing Electronics

The main electronic complexity in a PET scanner is the translation of the light glow into an electronic signal, preserving the time information, total optical photon count and position of the event. This information is used to describe the LoR and is essential in the image reconstruction process. The conversion of the light glow into an electronic signal is done using a photomultiplier, a device which receives a small light glow and creates an electrical signal proportional to the number of photons. The photomultipliers are directly coupled to the scintillation crystal using a light guide and are arranged in different forms depending the manufacturer of the scanners. That vary from a single photomultilplier per pixel to an array of photomultipliers covering several pixels. The topology of this arrangement is optimized to improve positioning of the gamma detection and to reduce the loss of visible photons. The task of photomultiplier is then to translate a signal of a few hundreds visible photons into a signal that is above the electronic noise of the input amplifier, a task that involves an amplification factor of hundreds of millions (10^8) times. During this conversion process not all optical photons are translated and electronic noise and dead time is added, degrading even further the energy, time resolution and positioning of the event.
The photomultipliers are normally of two kinds, the photomultiplier tubes (PMTs) and the silicon photomultipliers (SiPMs). The PMTs are vacuum tubes with a photocathode, several dynodes and a cathode at the end. The photocathode is facing the scintillation crystal and releases a small number of electrons when the optical photon hits it. The total amount of released electrons is then proportional to the light intensity. The released electrons are picked up by the first dynode which in turn releases more electrons and so on, this process is called *avalanche* and is where the gain of

the PMT resides. During the avalanche process the PMT adds Poisson noise at each dynode, reducing the amplification accuracy and enlarging the photo-peak. Finally the cathode absorbs the electrons, producing the electrical signal that is read by the input amplifiers. The PMTs works with voltage differences in the order of 1 kV between the photocathode and cathode, using an array of resistances to modulate the voltage difference between each dynode. Their large sizes (round footprint of $\varnothing \approx 25$ mm or square footprint from 10 to 50 mm side), delicate construction of the PMTs and their susceptibility to magnetic fields limits their application to PET-MRI scanners and are gradually disappearing from newer PET scanners.
The SiPM are solid state versions of the PMTs, they own several advantages over PMTs such as their small footprint, lower voltage and relative immunity to magnetic fields which has enabled the creation of PET-MRI systems [33]. They work by converting single photons into an avalanche of electrons by exploiting the impact ionization mechanism of semiconductors in which an electron (or hole) is created by the impact of an optical photon in a region of a large voltage difference (electric field) which triggers an avalanche breakdown [34]. This avalanche is then large enough to be read by the input amplifiers of the PET system. Among the disadvantages of the SiPM are the large dead zones between SiPMs due to manufacturing limitations which lead to sensitivity loses and the dark current and second emission effects which add noise to the signal. Nevertheless their small footprint, around 1×1 mm for several thousands microcells, allow the placement of large number of SiPM in the same area of a single PMT, thus reducing the impact of the noise.
After the acquisition, the generated electrical signal is processed using a wide variety of electronic configurations that vary in each scanner. The electronic noise added to the signal after the signal acquisition is generally low and does not impact the final resolution of the system. After the amplification the signal is generally digitized and processed by field-programmable gate arrays (FPGAs), further processing of the signal is done in digital domain using algorithms tailored for each scanner to obtain the event information. The processed gamma photon with all its corresponding information is called *single*, and contains the energy, position and time stamp of the interaction. Each *single* is, potentially, part of a coincidence.

Processing Firmware and Software

Converting the digital signal of each *single* into LoRs requires the processing of a large amount of information. To accumulate a total of $L = 10^7$ LoRs in a $T = 1200$ s, the system need to process roughly 8.33×10^3 LoRs per second. For each LoR the system must analyze each detected *single* and find coincidences within a time window of $\tau = 2$ ns to $\tau = 10$ ns, depending on the time resolution of the scanner. The probability of finding a true coincidence within the time window depends on the detectors and is proportional to $k_{\text{pt.}} v^2 \approx 0.1$. Due to this small probability the system requires the processing of an even larger number of *singles* per second, this number largely varies with the detectors type and the patient. If one assumes a *single-to-coincidence* ratio of 0.01 the system will require to process and transmit 833×10^3 *singles* per second to be able to finish an acquisition in the expected time. Hence the system must process

a single event in a few tenths micro-seconds.
The algorithms applied in this stage range from simple gated logic to complex statistical methods and machine learning techniques and they have three main tasks. First they must provide the system with an event time by analyzing the pulse received from the photomultiplier, this task is done using constant fraction discrimination on the pulse signal or more complex methods like maximum likelihood estimation [35] and convolutional neural networks [36]. Second they provide a measurement of the total energy of the photon. This value is obtained by careful calibration of the detectors and integration of the photomultipliers output. This value is used to discriminate valid $E_\gamma \approx 511$ `keV` gamma photons from scattered photons ($E_\gamma \leq (511 - \Delta E_\gamma)$ `keV`) and pile up or random gamma photons with higher energy ($E_\gamma \geq (511 + \Delta E_\gamma)$ `keV`), where ΔE_γ is the size of the energy window which varies with the detector construction. Finally the position of the event is determined using several methods. On pixelated crystals the position is mostly determined by the position of the pixel that captured the gamma photon and the resolution is determined by the size of the pixel and normally no depth information is given. Monolithic crystals need to estimate the position of the interaction by using the spatial information of the light glow which is obtained from multiple photomultipliers. The simplest method is the center of gravity (COG) method, which only provides position on the surface of the scintillatior (x, y) but no *depth of interaction* (DoI) (z). Other methods based on neural networks [37], maximum likelihood methods [38], k-nearest neighbours (kNN) [39] or special cristals with multiple phases [40, 41] achieve resolutions of 1.5 `mm` for X and Y axis and 3 `mm` of DoI (Z-axis), however these special crystals can result in loss of scintillation light and hence energy accuracy [31].
To achieve the required processing speeds, the scanners implement the *single* construction and coincidence sorting logic into FPGAs before transmitting and storing the data.

1.2 Attenuation Map Adquisition

PET scanners, using the FDG tracer, collect information only from molecular processes without providing direct information of the anatomy of the patient. The anatomical information is needed to correctly reconstruct the radiotracer distribution or activity image and it also provides anatomical reference to the physician. This limitation of PET scanners lead to different methods to obtain the anatomical information.

1.2.1 Attenuation scanning with transmission gamma scans

The straight-forward strategy to obtain the attenuation map of the patient is to scan the patient using a gamma source. The gamma source is normally a Caesium-137 ($^{137}_{55}$Cs) with peak energy in $E_\gamma = 661.7$ `keV` and Gallium-68 (^{68}Ga), obtained from Germanium-68 (^{68}Ge) with peak energy $E_\gamma = 511.0$ keV (pair creation). The transmission scan is performed by marching the gamma source around the patient and recollecting the incident gamma photons with a collimated detector. The source can

be open (broad-beam) or collimated (narrow-beam), the last one having longer acquisition times but resulting in better attenuation maps. For PET practice the narrow beam-source is used and can be estimated from the broad-beam measurement [42]. The attenuation map must be reconstructed from the transmission data. This process can be done using the Filtered Back Projection (FBP) algorithm presented in section 1.3. The resulting attenuation map can be directly applied to the correction of PET emission data without further processing.

1.2.2 PET-CT and PET-MRI scanners

The most direct method to acquire anatomical information of the patient is to do a Computed Tomography (CT) or Magnetic Resonance Imaging (MRI) scan along the PET scan. This is the method used by most modern PET scanners, which are called PET-CT or PET-MRI scanners.

PET-CT

The first hybrid scanners were the PET-CT systems which include a coaxial CT scanner with the PET detectors. The system works by performing a X-Ray scan in a series of beams, calculating the attenuation in the beam using the Lambert-Beer law ($E^f_\gamma = E^i_\gamma e^{-\int \mathrm{AM}^{\mathtt{X-Ray}}(l)dl)}$) and reconstructing the attenuation image using filtered back projection or other algorithm [43]. The energies used in the CT range from 40 `keV` to 140 `keV`, where the scintillation detectors have high efficiency, and thus adding negligible noise to the PET measurement. The attenuation map is a volumetric map of attenuation coefficients ($\mu_u^{X-Ray}(x, y, z)$, where x, y and z are the coordinates of the voxel) that must be converted to match the attenuation coefficients at 511 `keV` using a linear transformation or more accurate methods [44]. Nevertheless the remaining attenuation differences can produce errors in osseous lesions [45].
Besides the aditional hardware and time required, the CT scan method is not free from errors. The CT scan adds a significant amount of radiation dose to the patient, up to a 40% in a whole body PET scan with a low-dose CT [3, 4]. Also the method suffers of missaligment from patient movement, patient prosthetic implants or respiratory artifacts [46], inducing errors in the reconstruction of the PET image.

PET-MRI

The MRI scanners exploit the nuclear magnetic resonance effect to obtain anatomical and functional information of the patient. During an MRI scan the patient is subject to a strong magnetic field which polarizes the hydrogen atoms and then captures the radio signal emitted from these nuclei when the magnetic field is removed [47]. The resulting image is a three dimensional map of hydrogen in the body, which is an excellent soft-tissue representation. The resulting image must then be converted to a map of attenuation at 511 `keV`, which is a non linear process based on segmentation of the anatomical image and imputing the attenuation to each element [48]. More recently the translation of MRI to PET attenuation correction images is being explored using

machine learning methods [49, 50]. The PET-MRI scanners were impractical due to the PMT sensibility to magnetic fields but with the development of the SiPM the construction was greatly simplified making PET-MRI devices common in the market. The MRI, like the CT, suffers from misalignment but since it does not use ionizing radiation, it is preferred over CT hybrids in term of patient risk. Nevertheless the intensity of the magnetic field renders the MRI imaging impossible on patients with non-removable metallic implants.

1.2.3 Reconstruction of the attenuation map from PET data

A less common approach to obtain the attenuation map is to use the information already available in the PET study. These methods seek to jointly reconstruct the radiotracer distribution and the attenuation map. These methods generally implement mathematical models of the gamma photon emissions and interaction and try to maximize the likelihood of the proposed images to the acquired data and are called maximum-likelihood reconstruction of attenuation and activity (MLAA) [11, 51, 52]. These methods achieve in general a low quality image due to the severe cross-talk between the estimated activity and attenuation images, where errors in the activity images can be compensated by errors in the attenuation image. These methods can benefice from ToF information, reducing the ambiguous representation in the acquired data and improve the MLAA algorithm [53].
Other methods exploit the information that exists in scattered events and single scatter modeling to improve the convergence and quality of the attenuation map [54, 55]. Including additional information improves even further the accuracy of these methods, such as energy and prior structures from other studies [56].

1.2.4 Image based methods

Retrieving the attenuation map based on the NAC PET image or other image modality have been recently proposed due to the improvement of machine learning methods and enhanced computer capabilities. These methods are based on the application of Deep Convolutional Networks (DCN) or Generative Adversarial Networks (GAN) to generate an attenuation correction image from MRI [49, 57] and more recently from NAC PET [13, 16]. This subject is further developed in chapter 4.

1.3 Image Reconstruction

The information gathered by the PET scanner is a series of LoRs that correspond to the acquired events. To produce an image from this information several methods have been proposed and is still an active research topic. These methods must account for the noise and non completeness of the data due to attenuation, random and scattering effects. In the following sections the traditional reconstruction algorithms and their correction strategies are presented.

1.3.1 Reconstruction algorithms overview

The injected activity A is distributed around the anatomy of the patient following different patterns depending on the radiotracer design and patient condition. The distribution inside the FoV is represented by $\Lambda(x, y, z)$ and is proportional to the total injected activity,

$$A \propto \iiint \Lambda(x, y, z) dx\, dy\, dz. \tag{1.2}$$

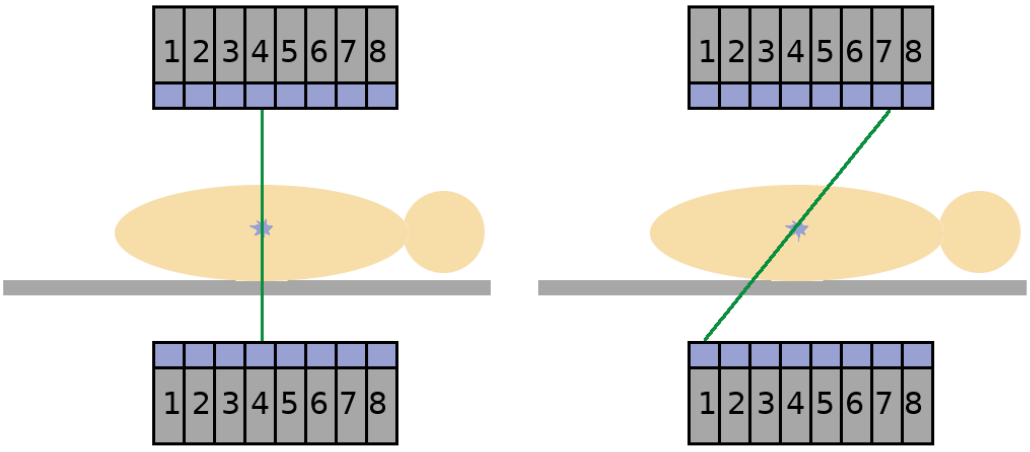
The activity inside the FoV is responsible for the total amount of detected LoRs L, which come from the atomic decay of the radiotracer. The number of expected radioactive decays in a given time can be described using a Poisson distribution, which in turn describes the exponential decay of a very large number of atoms with almost perfect accuracy in the detecting conditions of a PET scanner [58]. This model enables us to calculate the expected number of events detected along the path of each LoR l in a given time. This operation is called *projection* and is presented in equation 1.3. It is the simplest mathematical model of the PET acquisition process.

$$x_l = \int \Lambda(s_l) ds_l. \tag{1.3}$$

The reconstruction algorithms intend to perform the inverse operation to equation 1.3 using the information from a large number of LoRs.

Data Format

The storage format of the acquired LoRs is done commonly in two ways. The traditional way to store the data is the *Sinogram Mode* or *Michelogram Mode*. The sinogram was initially created for 2D data, which is a subset of only the LoRs included in the axial planes of the FoV (see figure 1.4(a)). The sinogram is a histogram representation of the data, where the FoV is defined using three parameters and LoRs are accumulated as counts in each value interval (bin). In figure 1.5 the sinogram parameters θ and ρ are represented. These parameters, $[0 \leq \theta \leq \pi]$, $[(-\varnothing_{FoV}/2) \leq \rho \leq (\varnothing_{FoV}/2)]$ and $[0 \leq q \leq Q]$, are the angular position, the distance of LoR to the center of the FoV with $\varnothing_{FoV}$ being the diameter of the FoV, and the axial plane or ring number (seen in the detectors of figure 1.4), with Q being the total number of rings or axial divisions of the FoV, respectively. The number of rings and the selected number of bins for θ and ρ parameters define the sinogram's accuracy. A large number of bins reduces the quantization error of the LoR positioning but creates large and sparse sinograms. In modern scanners, where the events cross multiple axial planes (see figure 1.4(b)) and are detected in 3D, new sinograms are added in oblique planes. These new sinograms are defined by their inclination expressed as the difference in ring numbers $\Delta_q = q_a - q_b$, where q_a and q_b are the rings of the LoR extremes. This results in a large amount of sinograms (depending on the ring number) which is called *Michelogram*. A michelogram presents the sinograms sorted by their initial and final rings, and allows compression techniques such as *ring mashing* [59] where singograms of similar inclination (similar Δ_q) are summed together to reduce the size and sparseness of the



(a) (b)

Figure 1.4: Representation of the LoR types according to its plane, in (a) a LoR perpendicular to the axial plane of ring 4 and in (b) a LoR along an oblique plane ranging from ring 1 to ring 7.

representation.

The second method is the *List Mode*, where all the LoRs are saved in a list. Each LoR in the list is not standardized and may include whatever information needed in the reconstruction process, such as the LoR timestamp, needed for dynamic or "4D" PET reconstruction.

Back-Projection

One of the simplest and fastest reconstruction methods is the *Back-Projection* method. It is an analytical method derived from the interpretation of 2D PET data as the Radon transformation of the dose distribution. The model proposes that the LoR (or bin in a sinogram) is a projection of the distribution along the LoR path, agreeing with equation 1.3. In this scenario the distribution of the radio tracer can be retrieved by performing a pseudo inverse of the Radon transformation. In the ring q, the LoRs obtained at a given ρ and θ are modeled following the Radon transformation:

$$x_l = Rad(\rho, \theta) = \iint \Lambda(x, y, q)\delta(x\ \cos(\theta) + y\ \sin(\theta) - \rho)\, dx\, dy, \tag{1.4}$$

where $\delta()$ is the delta function. The 2D reconstruction operation seeks to invert the equation 1.4 to obtain $\Lambda(x, y, q)$ at each ring. This inversion can be approximated using the Back-Projection (BP) operation as the sum of all crossing LoRs:

$$\Lambda(x, y, q) \approx \int_0^{2\pi} Rad(x\ \cos(\theta) + y\ \sin(\theta), \theta)d\theta, \tag{1.5}$$

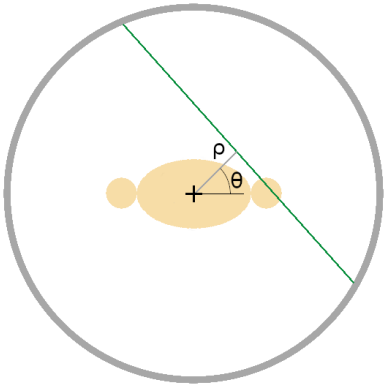


Figure 1.5: The LoR (in green) is represented by the angle θ and the distance to the center of the FoV ρ

which will shield a blurry representation of the real $\Lambda(x, y, q)$. The exact inversion of equation 1.4 must be calculated in the Fourier domain, using the Fourier Slice Theorem also known as the Central Slice Theorem of the Fourier transform [60]. Transforming the projections to the Fourier domain,

$$Rad(\omega_\rho, \theta) = \int_{-\infty}^{\infty} Rad(\rho, \theta) e^{-j2\pi\omega_\rho \rho}\, d\rho, \tag{1.6}$$

and the tracer distribution,

$$\Lambda(\omega_x, \omega_y, q) = \iint_{-\infty}^{\infty} \Lambda(x, y, q) e^{-j2\pi(\omega_x x + \omega_y y)}\, dx\, dy. \tag{1.7}$$

Thus using the Central Slice Theorem,

$$\Lambda(\omega_x, \omega_y, rn) = Rad(\omega_\rho, \theta). \tag{1.8}$$

Then the reconstruction operation is,

$$\Lambda(x, y, q) = \iint_{-\infty}^{\infty} Rad(\omega_\rho, \theta) e^{-j2\pi(\omega_x x + \omega_y y)}\, d\omega_x\, d\omega_y. \tag{1.9}$$

Replacing $\omega_x = \omega_\rho \cos(\theta)$ and $\omega_y = \omega_\rho \sin(\theta)$, and then $d\omega_x\, d\omega_y = \omega_\rho\, d\omega_\rho\, d\theta$, the reconstruction expression of the *Filtered Back-Projection* (FBP) is obtained,

$$\Lambda(x, y, q) = \int_0^{\pi} \int_{-\infty}^{\infty} Rad(\omega_\rho, \theta) e^{-j2\pi\omega_\rho \rho} \omega_\rho\, d\omega_\rho\, d\theta. \tag{1.10}$$

The name of this operation comes from the term ω_ρ multiplying the reconstruction operation which acts as a ramp filter enhancing the higher frequencies of the image.

The method presented in equation 1.10 will recover $\Lambda(x, y, q)$ perfectly if all the angular projections are available, but this is not the case in PET scanners where the projections are discrete and the number of detected events is limited. Under these conditions the ramp filter results in the enhancement of high frequency noise. In practice equation 1.5 is implemented as a summation over all the LoRs or sinogram bins,

$$\Lambda(x, y, q) = \frac{1}{\Theta} \sum_{\theta \in \Theta} Rad(x \cos(\theta) + y \sin(\theta), \theta), \tag{1.11}$$

where Θ is the number of angles in the sinogram. This implementation may leave pixels without data (due to non compatibility of image and sinogram sampling), in that case the image is completed through bi-linear sampling or nearest neighbor. To implement the FBP procedure of equation 1.10 the sinogram can be transformed using 1D Fourier transform of each θ row of the sinogram, multiplied by the response of the filter (ω_ρ in the case of the theoretical ramp filter), converted back using the inverse filter transformation and finally the filtered sinogram is used in the computation of equation 1.11. More information on the calculation of the inverse of the Radon transformation to achieve an analytical reconstruction of the radiotracer distribution can be found in [61].

Iterative Methods

Unlike the back-projection reconstruction, the iterative reconstruction methods do not assume that the PET data is deterministic. Iterative methods include in the reconstruction model the uncertainties associated with the nature of the PET data, such as atomic decay process, scatter and random effects. The formulation of iterative methods has three main parts, an imaging model, an objective function and an optimization algorithm. They use the image model to translate the image data into LoR data, then measure the accuracy of this data against the real data and perform a correction using the optimization technique.
The most common modeling of the acquisition is the linear image model. It can be make explicit by changing the notation of the projection operation in equation 1.3, using a matrix notation:

$$X = H\Lambda, \tag{1.12}$$

where H is the linear imaging model or system matrix, containing the relationship coefficients of each voxel in the distribution Λ to each LoR $x_l \in X$. It is conformed by physical and geometrical factors obtained by modeling of the acquisition process. Now Λ is a discrete representation of the continuous distribution $\Lambda(x, y, z)$, it can be interpreted as a voxelized 3D image, a series of sample points or point cloud. A single

LoR is then projected as:

$$x_l = \sum_{i=1}^{I} h_{l,i}\lambda_i, \tag{1.13}$$

where I is the total number of elements or voxels in Λ, λ_i is the i^{th} element of Λ and $h_{l,i}$ is the influence of λ_i on the counts detected in x_l. Using this notation the back-projection equation in 1.11 can also be defined:

$$\Lambda = H^{-1}X. \tag{1.14}$$

Each element of H^{-1} is the influence factor of each LoR $x_l \in X$ to each voxel or element in the distribution image Λ. These coefficients cannot be obtained through inversion of the matrix H due to its large size, the matrix H has a size of $[I \times L]$, and is potentially non-inversible. This operation is then performed using the transpose of the imaging matrix H^T. The coefficients are normally re-calculated for each LoR using a line tracing algorithm, such as the Siddon [62] tracer in voxelized support. A single element of Λ is back-projected as:

$$\lambda_i = \sum_{l=1}^{L} h_{i,l}^T x_l, \tag{1.15}$$

where $h_{i,l}^T$ is the influence of x_l on the activity in element λ_i.

The objective function defines the metric to judge the correctness of the current tracer distribution Λ^t. The objective function can have different forms. The most common is the *Log-Likelihood*, based on the assumption that PET data responds to a Poisson distribution. The counts observed in a single LoR can be described using this distribution as:

$$P(\hat{x}_l = x_l|\Lambda^t) = e^{-\hat{x}_l}\frac{\hat{x}_l^{x_l}}{x_l!}, \tag{1.16}$$

where $\hat{x}_l$ is the number of counts expected at the x_l LoR using the current proposed distribution Λ^t. The value of $\hat{x}_l$ is calculated using equation 1.3 or 1.13. From equation 1.16 the likelihood is derived:

$$P(X|\Lambda^t) = \prod_{l=0}^{L} e^{-\hat{x}_l}\frac{\hat{x}_l^{x_l}}{x_l!}. \tag{1.17}$$

And finally the log-likelihood is obtained by applying the natural logarithm:

$$\mathcal{L}(X|\Lambda^t) = \ln(P(X|\Lambda^t)) = \sum_{l=1}^{L} -\hat{x}_l + x_l \ln(\hat{x}_l). \tag{1.18}$$

Finally the objective function is maximized (or minimized) using an optimization technique. In the case of equation 1.18 the most popular maximization strategy is the *Expectation Maximization*, which derives in the *Maximum Likelihood-Expectation Max-*

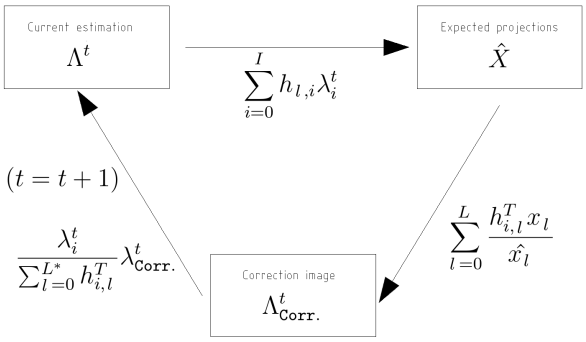


Figure 1.6: Box diagram of the ML-EM reconstruction process.

imization (ML-EM) algorithm [63]. This algorithm updates the proposed distribution Λ by successively applying:

$$\lambda_i^{t+1} = \left(\frac{\lambda_i^t}{\sum_{l=1}^{L^*} h_{i,l}^T}\right)\left(\sum_{l=1}^{L} \frac{h_{i,l}^T x_l}{\sum_{i=1}^{I} h_{l,i}\lambda_i^t}\right). \tag{1.19}$$

The optimization operation presented in equation 1.19 can be divided into three parts. First, the second denominator is a projection operation as described in equation 1.13 using the current distribution Λ^t, which results in the expected counts $\hat{x}_l \in \hat{X}$. Second, the numerator of the second part of equation 1.19 is a back-projection operation as defined in equation 1.15, where the expected LoR count x_n is weighted by the expected events, $x_l/\hat{x}_l$, resulting in a correction image $\Lambda_{\mathtt{Corr.}}^t$. Finally the first part of equation 1.19 defines the update proportion given the normalization matrix defined by $\sum_{l=1}^{L*} h_{i,l}^T$, where L^* is the set of all possible LoRs (in sinogram mode $L = L^*$). This iterative process is represented in figure 1.6.

This algorithm generates an acceptable quality image in 20 to 50 iterations, starting with low frequency structures and enhancing the higher frequency structures with each iteration. The convergence expectation maximization of the method is not guaranteed to the reach global maxima and as the iterations advance the variance of the image raises. For this reason the method is stopped before convergence using a stopping rule [64]. The ML-EM algorithm is normally applied in practice using the *Ordered Subset-Expectation Maximization* (OS-EM) [65], where the projected data is divided into several sub-sets using different criteria such as angular projection order, equidistant projections, accumulative groups or acquisition time. The OS-EM algorithm has a faster convergence than ML-EM (as many times as sub-groups) adding little noise respect to the ML-EM algorithm.

Other iterative methods expand the objective function to not only act as a metric of how well the proposed image matches the acquired LoRs, but also to include some desirable image properties such as local smoothness [66, 67] or anatomical constraints [68]. These methods are called *maximum a posteriori* (MAP) reconstruction. These methods seek to maximize the posterior probability of the objective function using expectation maximization or other optimization techniques.

1.3.2 Attenuation corrections

The Attenuation Map (AM) is used during the reconstruction process to correct the detection probability of each LoR due to the attenuation along its path. This correction is normally included in the imaging model by adding to each LoR an attenuation factor:

$$x_l^{\mu} = e^{-\int AM(s_l)ds_l}, \tag{1.20}$$

and in matrix notation:

$$x_l^{\mu} = e^{-\sum_{u=1}^{U} h_{l,u}^{\mu}\mu_u}, \tag{1.21}$$

where x_l^{μ} is the attenuation of the l^{th} LoR, U is the total number of elements or voxels representing the AM, μ_u is the u^{th} element of AM and $h_{j,u}^{\mu}$ is an element of H^{μ}, the imaging matrix of the AM. After obtaining the set of all attenuation coefficients X^{μ}, they can be embedded into the imaging matrix by multiplication of each row of H with the corresponding factor x_j^{μ}:

$$H^{+\texttt{attenuation}} = diag(X^{\mu})\, H. \tag{1.22}$$

1.3.3 Random events correction

The correction of the incoming data from random events can be obtained through two ways. In sinogram mode a random effect can be corrected with the acquisition of a separate sinogram using a delayed coincidence window. This coincidence window acquires LoRs from unrelated *single* data wich is sepparated by an arbitrary large time (i.e. 200 `ns`). The resulting sinogram is an estimation of the random events inside the meassured sinogram and is substracted from the real sinogram [69]. After the subtraction the reconstruction procedure is applied in the corrected sinogram. The second method that is compatible with list mode data is the estimation of the random contribution at each LoR using the *single* count rate in the detectors. By applying equation [70]:

$$x_l^{\texttt{rand.}} = 2\tau\iota_1\iota_2, \tag{1.23}$$

where x_l^{rnd} is the rate of random counts in the l^{th} LoR, τ is the coincidence window time and ι_1 and ι_2 are the *single* rate at the detectors where the l^{th} LoR was detected. Using this value is possible to correct the reconstruction by subtraction of the reconstructed random image [71] or by adding the random contribution inside the reconstruction iterations [72]:

$$\lambda_i^{t+1} = \frac{\lambda_i^t}{\sum_{l=1}^{L^*} h_{i,l}^T} \sum_{l=1}^{L} \frac{h_{i,l}^T x_l}{\sum_{i=1}^{I} h_{l,i}\lambda_i^t + x_l^{\texttt{rand.}}}. \tag{1.24}$$

1.3.4 Scattered events correction

The Compton scattering is the dominant interaction mechanism of gamma photons with matter in middle energies (from 100 keV to 1 MeV), making it one of the dominant undesired effects in PET tomography. In Compton scattering the gamma photon of energy E_γ interacts with a free electron of energy E_e imparting an energy to the electron and changing its direction a given angle ϕ:

$$E'_\gamma = E_\gamma(1 + \alpha - \alpha\cos(\phi))^{-1}, \tag{1.25}$$

with α, the fine structure constant, equal to:

$$\alpha = \frac{E_\gamma}{m_e c^2}, \tag{1.26}$$

where m_e is the electron mass and c is the speed of light in the vacuum. The scattered events are primarily filtered from the acquisition using an energy window. Since scattered photons deposit some of their energy during the Compton interaction, this energy is lower than the expected 511 keV value. The quality of this filtering depends on the detector's energy resolution. This value is represented by the *Full Width-Half Maximum* (FWHM) value of the photopeak in the energy histogram and is calculated as the energy span between the photopeak and the point where its counts fall below 50% of this value at each side. In the same way the *Full Width-Tenth Maximum* (FWTM) is often defined, as the span where the number of events falls to the 10% of the photo peak. The FWHM and FWTM are calculated over a Gaussian function fitted to the photopeak, summed to a linear or exponential function describing the background counts. The energy window is normally defined during the calibration of the PET scanner, where the detector performs an energy histogram of all detected *singles* when measuring a monotonic 511 keV gamma source. The width of the FWHM depends on the scintillation material, detector coupling and detector electronics. The selection of the energy window is a trade-off between the scattered events and the total number of detected events. While a higher number of events reduces the statistical noise of the reconstructed image, a broader window allows more scatter into the reconstruction thus degrading the image quality. Normally the energy window is selected to be between the FWHM and the FWTM. In figure 1.7 the energy histogram of a detector with the FWHM and FWTM limits is shown.

Even after the filtering using the energy window the image needs to be corrected from scattered events that are included in the measurement. These methods can be image, sinogram or model based. The image based models, make use of a dual window to detect scattered events, one window is the central or photopeak window and an additional one below at lower energies. The correction is done by reconstructing an image with the photopeak information (Λ^{phk}) and an other image with the lower window ($\Lambda^{scat.}$). Finally the corrected image is obtained by subtracting the photopeak image with a weighted scatter image $\Lambda = \Lambda^{phk} - K\Lambda^{scat.}$, where K is a constant obtained throug calibration of the scanner [73]. Other methods divide the photopeak in a low and high energy window. The high energy window is placed starting at 511 keV to avoid scattered events and serve as a *scatter-free* acquisition and is used to estimate

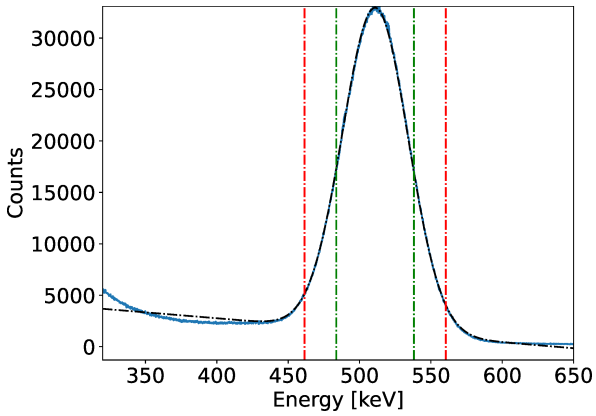


Figure 1.7: Energy histogram from a NaI(Ti) detector. In green the limits of the FWHM and in red the limits of the FWTM, around 511 `keV` central peak. The back dotted line represents the fitted function used to calculate the FWHM and FWTM values.

the scatter in the whole photopeak window [74]. Sinogram based models, such as the convolution-subtraction method [75], use the information form the photopeak. The scatter distribution is estimated by iteratively convolving the photopeak projections with a mono-exponential kernel and then subtracting the photopeak sinogram with the scattered sinogram. Finally analytic methods use the estimated activity and the attenuation map to calculate the expected scatter at each LoR direction. These algorithms can be implemented using full *Monte Carlo* (MC) simulations, but is generally too costly in terms of computation. For this reason they are normally implemented using a simplified model based on *single scatter simulation* (SSS).

Single Scatter Simulation algorithm

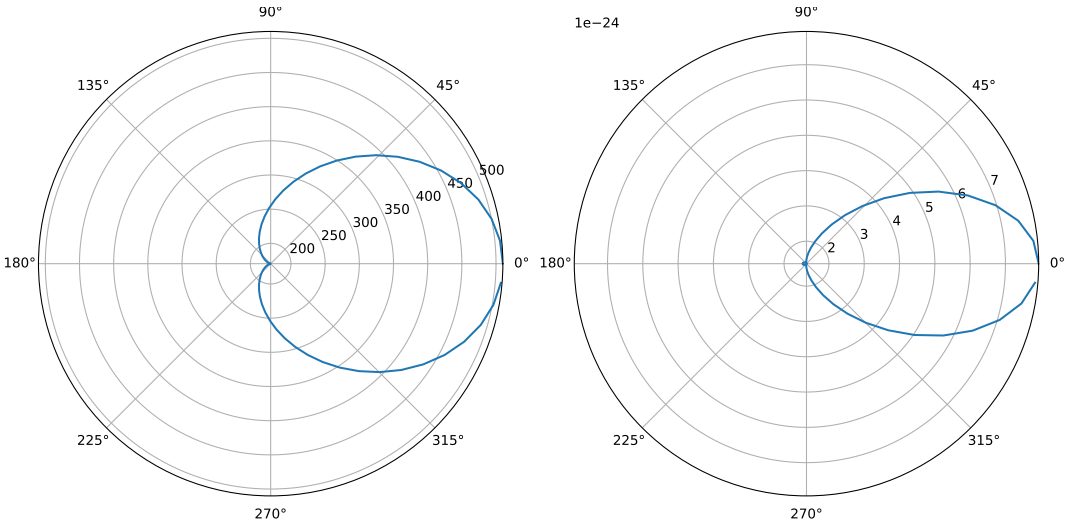
Scatter correction using SSS was proposed in [76], and implemented sucessfully on PET imaging [77, 78]. They are based on the simulation of the contribution of single scattered events. The probability of scattering of a gamma photon is given by the cross section σ_c, whose expression was first derived by Klein and Nishina [79]:

$$\sigma_c = \frac{3}{4}\sigma_o \left[\frac{2(1+\alpha)^2}{\alpha^2(1+2\alpha)} + \frac{\ln(1+2\alpha)}{\alpha}\left(\frac{1}{2} - \frac{1+\alpha}{\alpha^2}\right) - \frac{1+3\alpha}{(1+2\alpha)^2} \right], \tag{1.27}$$

where σ_o is the classical scattering cross section, which is function of r_o, the electron radius:

$$\sigma_o = \frac{8\pi r_o^2}{3}. \tag{1.28}$$

This the scattering angle, ϕ, is also a random variable whose distribution can be



(a) (b)

Figure 1.8: Energy of scattered 511 keV photon as function of the angle (a), angle distribution of scattered photon (b).

calculated using the differential cross section. It is defined by the cross section as:

$$\sigma_c = \int_0^{\pi} \left(\frac{\partial \sigma_c}{\partial \Omega}\right)_{\phi} 2\pi \sin(\phi) d\phi. \tag{1.29}$$

The resulting cross section is given by equation 1.30 [80]. The resulting angle distribution with its corresponding energy, for a 511 keV photon are shown in figure 1.8.

$$\left(\frac{\partial \sigma_c}{\partial \Omega}\right)_{\phi} = \frac{r_o^2}{2}\left(1 + \frac{\alpha^2(1-\cos\phi)^2}{(1+\cos^2\phi)[1+\alpha(1-\cos\phi)]}\right)\frac{1+\cos^2\phi}{[1+\alpha(1-\cos\phi)]^2}. \tag{1.30}$$

With the cross-section expression of equation 1.27 and the differential cross-section in equation 1.30 it is possible to calculate the scatter fraction for each LoR following the expression defined in [78]:

$$x_l^{\texttt{scatter}} = \sum_{u=1}^{U} \frac{\sigma_{D_a,u}\sigma_{D_b,u}}{4\pi d_{D_a,u}^2 d_{D_b,u}^2} \frac{\mu_u}{\sigma_c(E_\gamma,\phi)} \frac{d\sigma(E_\gamma,\phi)}{d\Omega} [K^{D_a} + K^{D_b}], \tag{1.31}$$

where D_a and D_b represent the extrema of the l^{th} LoR (impact detectors), $\sigma_{D_a,u}$ and $\sigma_{D_b,u}$ are the geometrical cross-section of the gamma photon going from each detector to the u^{th} attenuation element, $d_{D_a,u}$ and $d_{D_b,u}$ are the distances between each detector and the attenuation element, $\sigma_c(E_\gamma,\phi)$ and $\frac{d\sigma(E_\gamma,\phi)}{d\Omega}$ are the cross-section and differential cross-section for the given energy and angle respectively. The values K^{D_a} and K^{D_b} are

calculated as:

$$K^{D_a} = \upsilon_{D_a,u}\upsilon'_{D_b,u} e^{-(\sum_{u=1}^{U\in\overline{(D_a,u)}} h^{\mu}_{l,u}\mu_u + \sum_{u=1}^{U\in\overline{(D_b,u)}} h^{\mu}_{l,u}\mu'_u)} \sum_{i=1}^{I\in\overline{(D_a,u)}} h_{l,i}\lambda_i, \tag{1.32}$$

$$K^{D_b} = \upsilon'_{D_a,u}\upsilon_{D_b,u} e^{-(\sum_{u=1}^{U\in\overline{(D_a,u)}} h^{\mu}_{n,u}\mu'_u + \sum_{u=1}^{U\in\overline{(D_b,u)}} h^{\mu}_{l,u}\mu_u)} \sum_{i=1}^{I\in\overline{(D_b,u)}} h_{l,i}\lambda_i, \tag{1.33}$$

where $\upsilon_{D_a,u}$ and $\upsilon_{D_b,u}$ is the detector efficiency for a photon coming in the location of the u^{th} attenuation element. Note that the attenuation and projection operations are done only in the elements that cross the path of the LoR segment, $U \in \overline{(D_a,u)}$ and $U \in \overline{(D_b,u)}$ for the attenuation image and $I \in \overline{(D_a,u)}$ and $I \in \overline{(D_b,u)}$ for the activity image. The primed values represent the value at the energy of the scattered element (given by equation 1.25). Finally the scatter contribution is added to the ML-EM algorithm:

$$\lambda_i^{t+1} = \frac{\lambda_i^t}{\sum_{l=1}^{L^*} h_{i,l}^T} \sum_{l=1}^{L} \frac{h_{i,l}^T x_l}{\sum_{i=1}^{I} h_{l,i}\lambda_i^t + x_n^{\texttt{scatter}}}. \tag{1.34}$$

The calculation of this model adds a complexity of $\mathcal{O}(4NU)$ (plus tracing of LoR segments), which is expensive but, nevertheless, it less expensive than MC methods.

1.3.5 Activity decay correction

In long acquisitions, where the total time T is larger than a half of the radionuclide half-life, it is required to correct the reconstructed distribution. The decay does not directly affect the image quality (if the count rate is high during the whole process) but it affects the quantification of the dose. This can happen in whole body scans or dynamic PET procedures where several time-frames are reconstructed. To correct this effect the acquisition is divided into multiple time frames and the reconstructed activity is weighted by a *Decay Correction Factor* (DCF):

$$DCF = \frac{t_d \Delta t}{1 - e^{-t_d \Delta t}} e^{-t_d t_s}, \tag{1.35}$$

where t_d is the isotope disintegration constant, Δt the size of the time frame and t_s the absolute time of the start of the frame.

1.3.6 Geometric sensibility correction

When working in 3D mode, the geometry of the PET scanner generates zones within the FoV that are more densely sampled than others. These inhomogeneities arise due to various factors, such as:

- Dead zones between the photodetectors, modulating the event position estimation.

- Gain dispersion due to electronics of each single photodetector, which translates into positioning error or loss of counts.
- Different gain of each pixel scintillator (in pixelated PET scanners).
- Uneven detector depth for different LoRs (this happens more severely in continuous and/or thick scintillators, where the detection of probability of LoR is higher when the LoR path is not normal to the surface of the scintillator).

The correction of these factors can be done by careful calibration of the detectors, a *flat front* gamma field is used to scan the dead zones, gain dispersion and pixel dispersion of the scanner. The gain is then added to each LoR probability by:

$$x_l^{\texttt{gain}} = D_g^1 D_g^2, \tag{1.36}$$

where D_g^1 and D_g^2 are the gain correction factor for the detectors at the extrema of the l^{th} LoR. The resulting correction set $X^{\texttt{gain}}$ is then added to the imaging matrix:

$$H^{+\texttt{gain}} = diag(X^{\texttt{gain}})\, H. \tag{1.37}$$

The geometric factors are corrected in the image domain using a back-projection of all possible LoRs with $x_l = 1$. This term exists in the ML-EM expression (equation 1.19), as $\sum_{l=1}^{L^*} h_{i,l}^T$, in list mode L^* is too large to be calculated and is replaced by $l \xleftarrow{uniform} L^*$, an uniform random sampling of L^*.
In sinogram mode these corrections can also be achieved by acquisition of an uniform phantom that covers the FoV. The sinogram obtained from this study is used as a normalization map for the acquired sinogram.

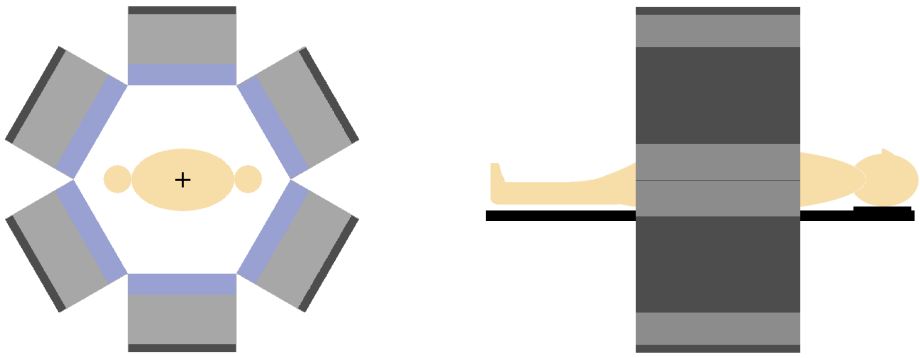
2 AR-PET Prototype

Here the AR-PET prototype scanner is presented, which is used throughout the thesis for testing. First the prototype's objective, capability and construction blocks are described. Then the focus is set on the scintillation cameras, their construction and optimization. For the construction of the AR-PET scintillation cameras, a selection criteria of the PMT units is developed. This new construction framework makes use of genetic algorithms to improve the homogeneity of the PMT gains. After, in the optimization aspect of the scintillation cameras, a new positioning scheme is presented. Such a positioning scheme is based on neural networks and is trained with a simple flood acquisition. The resulting positioning improves center of gravity positioning algorithm, thus enlarging the sensible area of the scintillation cameras.

2.1 AR-PET Prototype

The AR-PET [81, 82] is a prototype scanner designed to reduce the cost of the PET studies and enable a wide-spread use of the technique in lower income populations such as Latin-American states. In these communities the number of PET cameras per million inhabitant is near 0.5 against 1.5 in Europe and 7 in the United States of America [83]. The prototype is being developed by the Instrumentation and Control department of the *National Atomic Energy Commission* (CNEA) with collaboration from the *Artificial Intelligence and Robotics Group* (GIAR) of the *National Technological University* (UTN). Currently the first prototype is being tested at the nuclear medicine center of the *Hospital de Clínicas José de San Martín* in Buenos Aires, Argentina.

The scanner features six large scintillation cameras arranged in a hexagon, as shown in figure 2.1. In figure 2.2 the 3D view of the internal construction of the AR-PET is shown. Since great part of the overall cost of a commercial PET scanner comes from the crystals and the difficulty in fabricating small pixels, each scintillation camera is composed of a continuous NaI(TI) crystal. These continuous crystals present a large geometrical coverage fraction G and good stopping power υ at a low cost. These factors enable to reduce the radiation dose on the patient and reduce the overall study time (see equation 1.1), both limiting factors of the broader application of PET studies. A similar configuration to the AR-PET, using 4 continuous crystals was successfully applied in small-animal PET scanners [84]. To reduce the effects of blind spots and detection inhomogeneities of the hexagonal arrangement, the scanner cameras rotate around the axial axis at 2 `rpm`. Due to the rotational acquisition, the large detector



(a) (b)

Figure 2.1: Front (a) and lateral (b) views of the AR-PET scanner geometry. In grey the scanner structure, in light blue the detector surface and in black the patient couch.

surface and Depth of Interaction (DoI) capability, the scanner collects data in List-Mode. Finally the reconstruction of the radiotracer distribution is done on an external server using the ML-EM algorithm presented in section 1.3. The scanner does not posses the capability to scan the attenuation map, currently recurring to matching of external studies to compensate for attenuation effects in the reconstruction.

2.2 Scintillation cameras

Each of the scintillation cameras in the scanner is composed of a $406.4 \times 304.8 \times 25.4$ $\mathtt{mm}^3$ continuous NaI(TI) crystal, populated by an array of 8×6 *Hamamatsu R1534* PMTs. These have a square profile of 52.4×52.4 $\mathtt{mm}^2$ that completely covers the crystal's surface without a complex coupling mechanic. A representation of the scintillation crystal and detector camera can be seen in figure 2.3.

To cope with the limitations of a monolithic scintillation crystal the AR-PET prototype features a distributed processing architecture on each PTM [85]. Each PMT has its own polarization amplification and digitalization hardware, controlled by a *Spartan3e* FPGA. This FPGA process the electric signal from the PMT, integrates the pulse energy and calculates the interaction time stamp with 400 `ps` accuracy. After the pulse detection and energy integration, the FPGA communicates with an *Artix 7-100* FPGA within the detector camera using Low-Voltage Differential Signaling (LVDS). This last FPGA is responsible to translate the 48 energies and time stamps (one pair for each PMT in the camera) into a valid *single*, composed of the spatial coordinates of the event within the crystal, the deposited energy and the time stamp. This FPGA applies

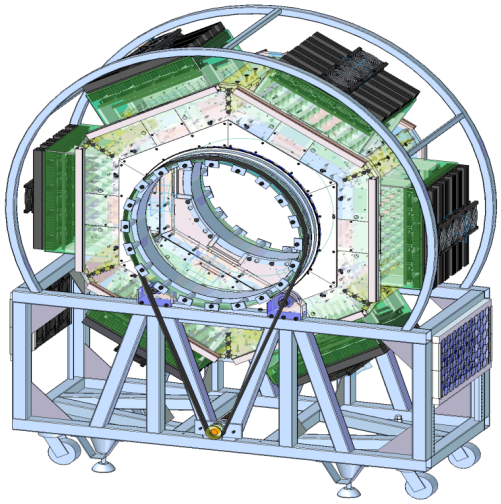
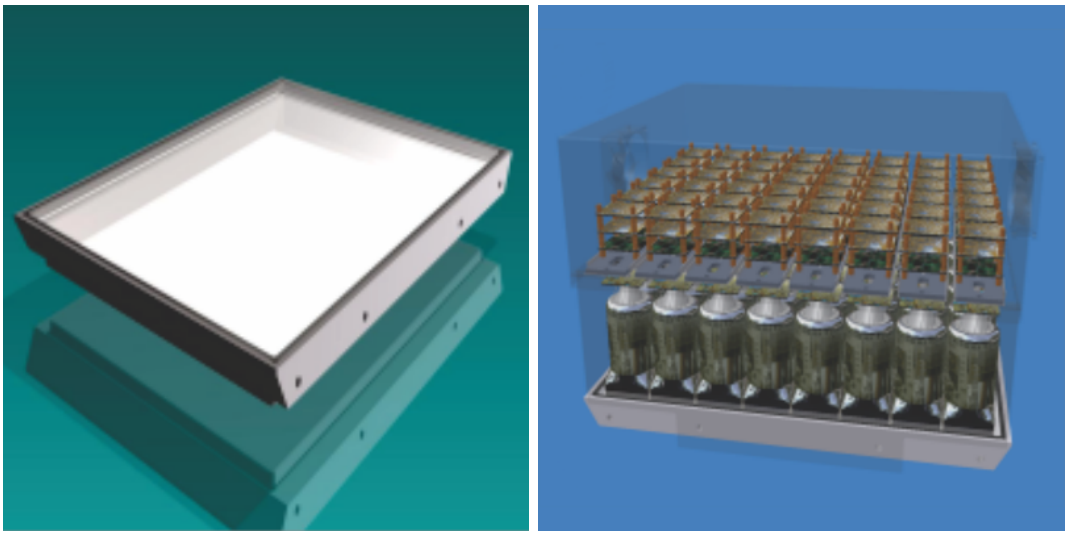
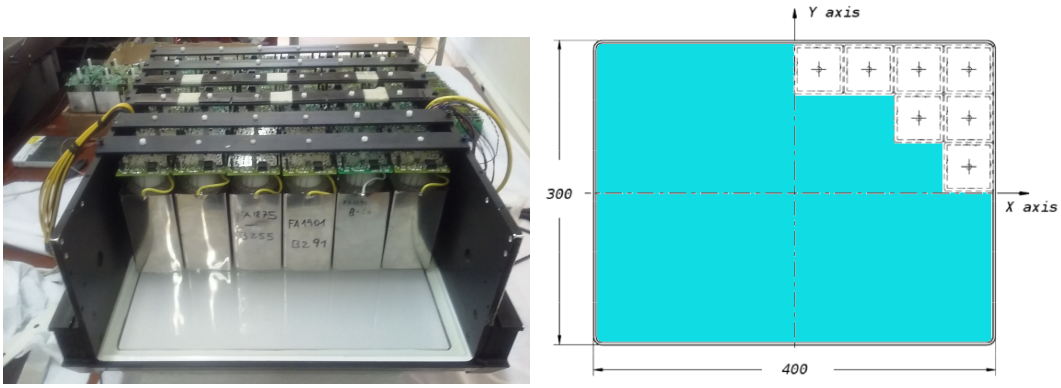


Figure 2.2: Perspective view of the internal disposition with the rotation system of the AR-PET scanner prototype.



(a) (b)

Figure 2.3: Scintillator crystal (a) and full detector camera (b) of the AR-PET scanner.



(a) (b)

Figure 2.4: Photomultiplier tube disposition in camera unit (a) and schematic of placement (b). In (b) the detector unit is presented in white over the coloured scintillation crystal (Image courtesy of E. Venialgo, 2011 IEEE N.S.S).

the energy calibration and the positioning algorithms and yields clean information to the coincidence sorter FPGA, located outside the camera. The coincidence FPGA, an other *Artix 7-100*, gathers the information from the six cameras and applies the coincidence sorting logic. The coincidence logic finds events from opposite cameras within a $\tau = 20$ `ns` time window. The found coincidences are communicated to the scanner's onboard PC via Ethernet and saved. Each scintillation camera possess 49 FPGAs. With six cameras in the scanner, this results in 295 FPGAs. All of these FPGAs work by sharing the same clock and in perfect synchronism to ensure the time accuracy of the system. Since the detection cameras are stand-alone units that can function as gamma cameras, the AR-PET scanner is able to function with less than six cameras, at the cost of losing geometrical coverage.

2.3 Photomultiplier tube selection with genetic algorithms

A known problem with the construction of large scintillation cameras with continuous crystals are the effects of a wrong configuration that directly affect the final resolution of the camera. This problem becomes even more complex when the number of PMTs to be used is large. This is the case of the AR-PET where 48 PMTs are placed in each of the six scintillation cameras according to figure 2.4.

After the interaction of the gamma photon with the scintillation crystal, hundreds or thousands of luminous photons are created in an omnidirectional flash. Some of these photons are directly absorbed by PMTs and some are reflected against the crystal walls before reaching a PMT. Therefore the PMTs near the edge of the crystal receive significantly more light than those of the center. Also, the PMT tubes have a high gain dispersion from unit to unit given the same electrical operation conditions.

To achieve a uniform count rate for the whole surface of the scintillation crystal it is necessary to equalize the energy gain and quantum efficiency of each PMT. In addition, for PET application, it is necessary to equalize the transit time of each PMT. Such a transit time of the PMT defines the time that takes the PMT to transform and amplify the optical signal to a measurable electrical pulse. This time must be equal in all PMTs to avoid integration losses in the energy calculation of the gamma photon. The transit time depends on the high voltage (HV) applied to the PMT [86]. Therefore it is fixed for all PMTs on the camera. Thus to control the PMT gain, the voltage between the first dynodes is controled [87], without changing the transit time of the unit. Several approaches exist to reduce the effect of the photomultiplier gain dispersion. The most common is using software coefficients and count skimming [88] or performing single-PMT calibration using its spectral information [89]. Hardware methods have also been studied like *PMT-quadrant-sharing* techniques [90] and also including *Light Emitting Diodes* (LEDs) inside the scintillation camera to calibrate PMTs by groups [91] or directly calculating correction matrices at hardware level [92]. The reduction of photomultiplier groups by segmenting the crystal is also common design choice which reduces this problem [93] but not applicable to continuous crystals. All of the cited methods are focused on the correction of the PMT gain after it is placed inside the scintillation camera, taking for a fact that every unit can be placed to work at every location inside the scintillation crystal. This might hold if one is able to choose from a large set of PMT units and can, therefore, choose those of similar characteristics. In the case where the amount of PMTs to be placed is not large enough to disregard the large PMT gain variation some PMT localization intelligence must be applied.
In order to create working scintillation cameras with few spare PMT units while improving the initial acquisition efficiency, a new method was developed which automatically generate sub-optimal PMTs configuration for each scintillation camera using k-nearest neighbours (kNN) clustering and genetic algorithms to explore the possible configurations.

2.3.1 Crystal Reflection

The definition of an adequate configuration of PMTs in the scintillation camera requires knowing the behavior of both the scintillation crystal edges and the PMT dynode modulation. The first effect is a consequence of the construction of the crystal, which is covered with a reflective paint in order to direct all scintillation photons to the crystal surface, the structure of the camera which holds the PMTs and the optic coupling of the PMTs with the crystal. The reflective paint is important to avoid losing light over the edges of the crystal and enables the chamber to correctly measure the deposited energy of the gamma photon. As a counter-effect, a PMT placed in the edge gathers more light if a gamma photon hits at its center than the amount of light it would gather if a photon hits the same relative position when the PMT is placed in the center of the crystal. Using the Monte Carlo simulation software *Geant4 Application for Tomographic Emission* (GATE) simulation software a scintillation camera was modeled, following the AR-PET design and using ideal PMT units (perfect photon counting). A

uniform gamma flat source was placed in front of the camera. The scintillation effect and visible photon transportation was simulated, obtaining the number of photons received by each PMT in every location of the crystal. The simulation results showed three distinct sensibility groups: the center, the edges and the corners. Where the center group is composed of those PMTs that neighbour only other PMTs (not the camera structure). From these three groups two factors were derived. These factors relate the energy observed when detecting an event in the corner or edge of the camera to detecting the same event at the center of the camera:

$$\nu_{c,c}^{\texttt{Simulation}} = \frac{\sum_{p=1}^{P\in\texttt{Corner}} \zeta_p}{\sum_{p=1}^{P\in\texttt{Center}} \zeta_p} = 1.52, \tag{2.1}$$

$$\nu_{c,e}^{\texttt{Simulation}} = \frac{\sum_{p=1}^{P\in\texttt{Edge}} \zeta_p}{\sum_{p=1}^{P\in\texttt{Center}} \zeta_p} = 1.21, \tag{2.2}$$

where P is the total number of PMTs, $P \in \texttt{Center}$, $P \in \texttt{Border}$ and $P \in \texttt{Edge}$ represent the sub group of PMTs in each region of the camera and ζ is the mean number of detected visible photons. They represent the relation in photon count for the *center to corner* and *center to edge* respectively.

2.3.2 Photomultiplier Gain

Due to the non-linearity and high gain dispersion between different PMTs for the same HV and dynode values, the PMTs are difficult to characterize. To test their behaviour a custom calibration procedure is used. In the calibration process, the PMT is coupled with a 25.4 `mm` thick, 76.2 `mm` diameter NaI(TI) crystal and placed within a dark chamber. The detector set is then exited with a Caesium-137 ($^{137}_{55}$Cs) source and the peak ADC channel is annotated. This is repeated for each pre-defined working point. For each PMT four distinct anode-cathode HV values and five dynode values were tested, totalling 20 different working points. The dynode values are expressed as a function of the controlling Digital to Analog Converter (DAC) which controls the final voltage value. The dynode DAC has 12 `bits` resolution, thus a range of $[0, 4095]$. The measured working points of a single PMT unit can be represented by four curves as shown in figure 2.5. It can be seen that the working points of the PMT vary with the dynode value following a non-linear function. The observed behaviour is fitted using a sigmoid curve. It reflects the non-linearity, both at the beginning, when the dynode is short-circuited and at the end, when it saturates.

2.3.3 Model contrasting

Since the calibration curves are generated outside the actual scintillation cameras, a correspondence factor $\chi_{\texttt{cal}}$ between the calibration crystal and the camera is derived. This factor is calculated by testing a group of PMTs in each possible position of the scintillation camera and then calibrated. Since the assembly of a scintillation camera is a complex process, only one group of 48 PMTs was tested. The obtained factor relates

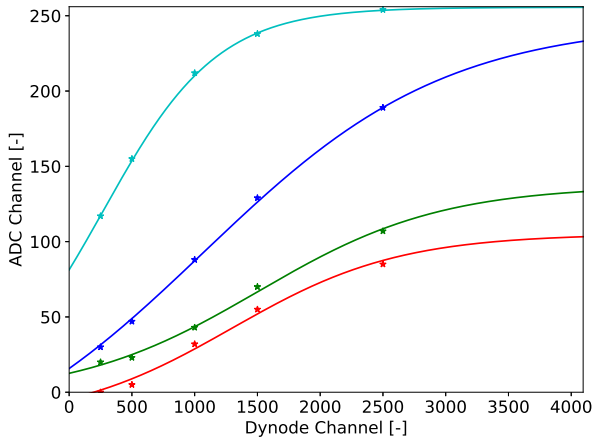


Figure 2.5: Photomultiplier tube working points in ADC value as function of dynode DAC value, $[0, 4095]$, and the approximated curve. HV values: 1300 V (light blue), 1200 V (dark blue), 1100 V (green) and 1000 V (red).

the amount of light received during the calibration and the different positions within the camera. The factor is calculated as the ratio between the ADC channel during calibration (which is proportional to the detected photons $\zeta_{\texttt{cal}}$) and ADC channel within the camera (proportional to $\zeta_{\texttt{cam}}$), as showed by equation 2.3.

$$\chi_{\texttt{cal}}^{\texttt{region}} = \frac{\zeta_{\texttt{cal}}}{\zeta_{\texttt{cam}}^{\texttt{region}}}. \tag{2.3}$$

Using the 48 tested PMTs the relation of light for each PMT can be derived. The values of the empirical correspondence factors and the amount of photons detected in the simulation (ζ_p) are shown as the pixel intensity in figure 2.6. Taking the mean value over each region (center, edge and corner), a mean correspondence factor is obtained. The resulting mean factors were: $\chi_{\texttt{cal}}^{\texttt{center}} = 0.44$, $\chi_{\texttt{cal}}^{\texttt{edge}} = 0.54$ and $\chi_{\texttt{cal}}^{\texttt{corner}} = 0.71$. Finally by comparing the gains at each position of the camera, the same factors as such obtained through simulation in section 2.3.1 can be derived:

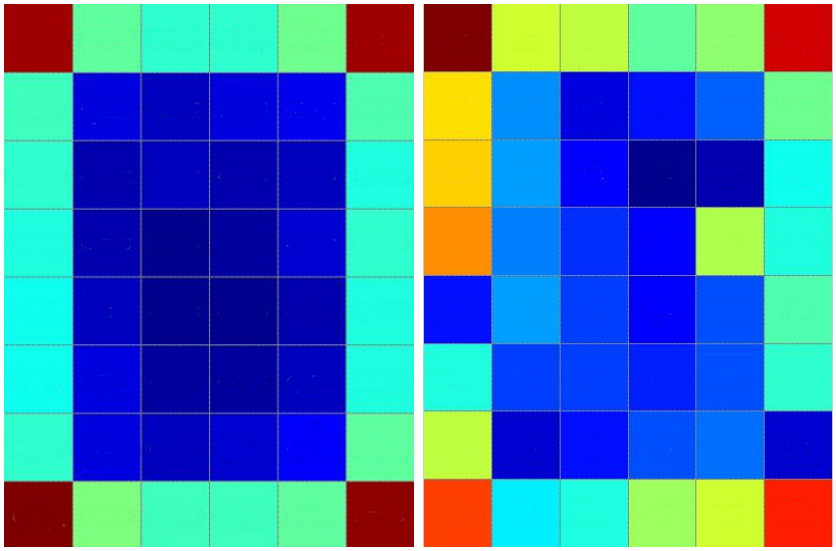
$$\nu_{c,c}^{\texttt{Empirical}} = \frac{\chi_{\texttt{cal}}^{\texttt{corner}}}{\chi_{\texttt{cal}}^{\texttt{center}}} = 1.61, \tag{2.4}$$

$$\nu_{c,e}^{\texttt{Empirical}} = \frac{\chi_{\texttt{cal}}^{\texttt{edge}}}{\chi_{\texttt{cal}}^{\texttt{center}}} = 1.22. \tag{2.5}$$

It can be noted that these factors are very similar to the simulated values $\nu_{c,c}^{\texttt{Simulation}} = 1.52$ and $\nu_{c,e}^{\texttt{Simulation}} = 1.21$, with a maximum difference of 6%. This supports the accuracy of the calibration procedure.

2.3.4 K-means pre-processing

The starting point of the PMT positioning algorithm is the preprocessing of the problem using k-means clustering. The possible starting configurations are given by the



(a) (b)

Figure 2.6: Ratio of the expected amount of light received in the center of the AR-PET camera to the light received in each PMT for the GATE simulation (a) and the empirical test (b). The color in (a) represents the number of detected photons, ranging from more photons (red) to fewer photons (blue). In (b) the color represent the correspondence factors derived from equation 2.3.

amount of PMTs available and the amount of PMTs to be placed in the different scintillation cameras to be constructed. This is given by the permutation formula shown in equation 2.6, where P' is the total number of PMT units and P the units to be placed.

$$\texttt{Perm} = \frac{P'!}{(P' - P)!}\,. \tag{2.6}$$

In order to generate a good starting point, which helps the convergence of the later genetic algorithm, the PMTs are clustered around common HV. The Clusters are created by choosing a target ADC channel for the peak energy and measuring the similarity of the dynode values. By making use of the k-means algorithm [94] three clusters of PMTs are generated for each HV value. The centers are in units of dynode value and their value is not given a-priori. The algorithm converges by using the target ADC channel and the correspondence factors $\chi_{\texttt{cal}}^{\texttt{center}}$, $\chi_{\texttt{cal}}^{\texttt{edge}}$ and $\chi_{\texttt{cal}}^{\texttt{corner}}$. This means that one cluster per position in the scintillation camera. Each cluster is given in units of ADC. From these groups, the elements nearer to the cluster center are extracted, for a given HV, which holds the lower dispersion of dynode values. If multiple cameras are to be generated, the PMTs for the first camera are extracted from the list and the selection process is repeated until all cameras are populated.

2.3.5 Genetic Algorithm overview

Since an exhaustive exploration of the problem is not possible a sub-optimal search is applied. The genetic algorithms are fit for this task and are proven to converge [95]. This technique requires the description of *individuals*, which in this case are feasible configurations of the scintillation cameras. The individual is described as the order of the PMTs inside each of the scintillation cameras to be constructed and the HV value of each of these cameras. The resulting *genome* of the individual is a vector of $[(P+1) \times D] + P_{\texttt{slk}}$ elements, where D is the number of cameras (or detectors) to be constructed and $P_{\texttt{slk}}$ is a slack variable, which represents the number of remaining PMTs which are selected as *spares* for the cameras. These spare PMTs will not be considered in the fitness calculation but are used in the mutation process. At the beginning a population of $\Xi = 30$ individuals is created using the k-means clustering. The starting fitness of the population is calculated and then the selection process is applied. This selection process is done by mutation and elitism (or replication) [96]. For each iteration a given amount $\Xi_{\texttt{elit}} > 0$ of parent individuals is selected to be copied unmodified to the next iteration of the algorithm, this is called *elitism*. Then a *mutation* process is applied to the population, the *crossover* process is not used. The fitness value ξ of each individual of the new population is computed and the probability p_m of the m^{th} individual to survive is given by:

$$p_m = \frac{\xi_m}{\sum_{n=1}^{\Xi} \xi_n}. \tag{2.7}$$

After the selection process, a new population is created and the algorithm proceeds to the next iteration. This is repeated until an individual full-fills the target criteria or the number of iterations is exhausted.

2.3.6 Fitness calculation

The fitness reflects the worst dynode value difference between PMTs at the same HV. The dynode value is proportional to the HV but it is given in bits units. In this sense, the lower the fitness value the better is the configuration (the individual). The fitness of the configuration is calculated as the largest of the maximum difference of dynode values computed inside each scintillation camera. This measure is described by:

$$\xi_n = \max(\Delta \mathtt{v}^1_{\mathtt{dn}}, \Delta \mathtt{v}^2_{\mathtt{dn}}, ..., \Delta \mathtt{v}^D_{\mathtt{dn}}), \tag{2.8}$$

with:

$$\Delta \mathtt{v}_{\mathtt{dn}} = \max(\overline{\mathtt{v}_{\mathtt{dn}}}) - \min(\overline{\mathtt{v}_{\mathtt{dn}}}), \tag{2.9}$$

where $\overline{\mathtt{v}_{\mathtt{dn}}}$ is the set of the P dynode voltages of each PMT in the camera. This metric was chosen since the worst case is given when a PMT is limited by a low dynode voltage and other is limited by a high dynode voltage. Such situation limits the fine-tuning possibilities of the dynode value after the selection process.

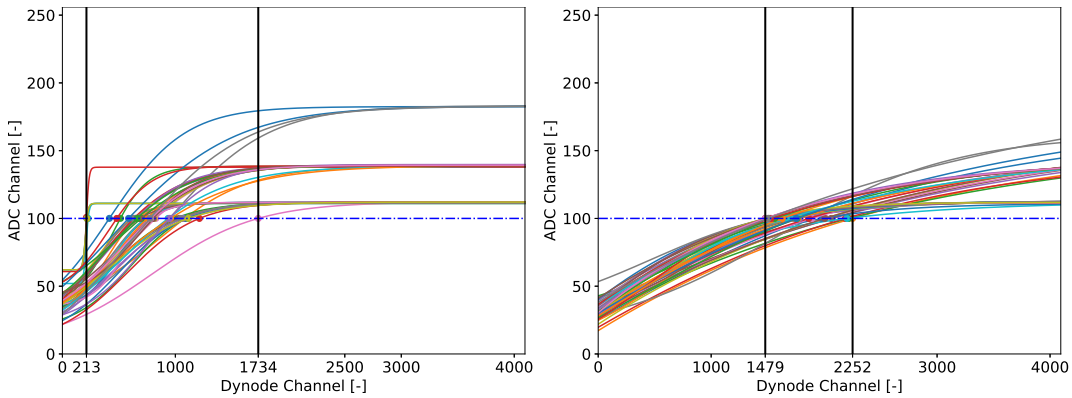
2.3.7 Mutation

The mutation process can occur in two different ways. The first way is the most probable, with a probability below 10% for each of the scintillation cameras that compose the individual. If a camera is selected to be mutated one of its PMTs is randomly selected, then, another camera of the same individual is selected randomly and an other PMT from this camera is also randomly selected. These two selected PMTs are swapped and both cameras are reconfigured using all the PMTs which belong to their respective groups. If there are spare PMTs, they are considered in both processes, swapping and reconfiguration. The second mutation process is much less probable (with a probability of mutation lower than 1%), and it changes the HV of the camera. If a camera is chosen, its HV is selected randomly to any calibrated value.

2.3.8 Results and Discussion

The procedure was used to generate the $D = 6$ scintillation cameras of the AR-PET, each consisting of $P = 48$ PMTs. For the construction of this cameras a total of $P' = 307$ PMT units were available. Using equation 2.6 the total amount of possible configurations can be calculated and it surpasses $6 \times 10^{+614}$. The constructed cameras were compared to an existing prototype scintillation camera. The prototype scintillation camera was created by hand-picking each PMT from the total P' PMTs, assigning those with lower gain to the corners, where they receive more light, those with average gain to the edges and finally, the ones with the higher gains to the center. This process was done without any restrain in order to build the other five cameras, since it was a prototype. The chosen objective ADC channel for the peak energy of the Caesium-137 ($^{137}_{55}$Cs) was 100 (within a $[0, 255]$ range). The HV was chosen to be 1200 v. The resulting scintillation camera had a fitness value of 1521, calculated by equation 2.9. In figure 2.7(a) the interpolated curves of each PMT for the chosen HV composing camera are drawn. These curves are corrected by the gain factors $\chi_{\texttt{cal}}$. The AR-PET cameras were generated using the described process, using a population of $\Xi = 30$ individuals, an elitism value of 10%, probability of mutation of 6% for the first type and 0.6% for the second type and 1000 iterations. No ending criteria was used. The resulting fitness values of the resulting six cameras are summarized, along with the prototype camera, in table 2.1. In figure 2.7 the PMT curves of the camera with the worst fitness value, which define the fitness of the individual and the prototype scintillation camera are shown. It can be seen that the fitness value, represented by the span of the vertical black lines, is much lower in the generated camera.

The k-means clustering along with the genetic algorithm proved to reduce the dispersion of dynode values within the scintillation camera significantly. It was also able to found out a viable configuration for all the requested cameras with a spare amount of PMTs of less than the 7% of the total available units. The cameras constructed with this method achieve an FWHM of 15% (without fine-tuning), improving previous results of 20% [97]. Moreover the assembled cameras have working points near the lineal dynode voltage vs gain zones of the PMTs.



(a) (b)

Figure 2.7: PMT curves of the prototype scintillation camera, created with hand picked PMTs (a) and camera number 4 of generated cameras (b). The dashed horizontal line represents the target ADC peak, the coloured dot in the intersection of each curve with the line indicates the working point of the PMT. The space between the vertical black lines represent the maximum deviation in working points, which illustrates the fitness value.

Table 2.1: Scintillation cameras fitness values

Camera	HV [v]	Fitness
1	1200	759
2	1100	771
3	1100	741
4	1300	773
5	1200	757
6	1200	684
Prototype	1200	1521

2.4 Event localization algorithms

Most clinical scanners have their event localization resolutions defined by the size of the crystal pixels. The pixel size is normally around 4 mm, and the most modern systems reduce the size up to 3.2 mm [98]. Normally the commercial pixelated scanners do not provide depth of interaction (DoI) information and their efforts to reduce the error from this source is to use thin crystals with high interaction probability υ. Since the AR-PET scanner features a monolithic large crystal per detection camera, the localization of each gamma photon interaction is obtained using the intensity values of the PMTs in the scintillation camera. The output of the p^{th} PMT is a value proportional to the amount of detected photons $e_p \approx \zeta_p$. The position is derived mathematically from the acquired energy vector $\bar{e}$. This can be done using linear or non-linear procedures.

2.4.1 Anger

The most simple technique of event localization is the Anger or center of gravity method. This method estimates the $\boldsymbol{r} = (r_x, r_y)$ positions in the surface of the detector by weighting the position of the PTMs detected values and their locations in the detector's surface. The position of an event using the Anger method is estimated by:

$$r_x = \underbrace{\begin{bmatrix} e'_1 & e'_2 & \dots & e'_P \end{bmatrix}}_{\overline{e}'} \underbrace{\begin{bmatrix} c_1^x \\ c_2^x \\ \vdots \\ c_P^x \end{bmatrix}}_{C_x}, \tag{2.10}$$

and

$$r_y = \underbrace{\begin{bmatrix} e'_1 & e'_2 & \dots & e'_P \end{bmatrix}}_{\overline{e}'} \underbrace{\begin{bmatrix} c_1^y \\ c_2^y \\ \vdots \\ c_P^y \end{bmatrix}}_{C_y}. \tag{2.11}$$

The vector $\overline{e}'$ represents the normalized energy of each PMT in the scintillation camera, with $e'_p = (e_p / \sum_{p=1}^{P} e_p)$ and C_x and C_y are the Anger coefficient vectors used to calculate the r_x and r_y positions of the event along the X and Y axis respectively. These vectors are obtained through a calibration procedure using a matrix of centroid events $\overline{e}_{\texttt{centroid}}$. This is a square matrix consisting of stacked the energy vectors $\overline{e}$. Each of the stacked vectors corresponds to the energy distribution when an event impact below the center of mass of each PMT. Knowing the position of each PMT center $(r_x^{\texttt{PMT}_p}, r_y^{\texttt{PMT}_p})$ in the scintillation crystal it is possible to calculate the matrices C_x and C_y by solving:

$$\begin{bmatrix} c_1^x, c_1^y \\ c_2^x, c_2^y \\ \vdots, \vdots \\ c_P^x, c_P^y \end{bmatrix} = \underbrace{\begin{bmatrix} e_1^{\texttt{PMT}_1} & e_2^{\texttt{PMT}_1} & \dots & e_P^{\texttt{PMT}_1} \\ e_1^{\texttt{PMT}_2} & e_2^{\texttt{PMT}_2} & \dots & e_P^{\texttt{PMT}_2} \\ \vdots & \vdots & \vdots & \vdots \\ e_1^{\texttt{PMT}_P} & e_2^{\texttt{PMT}_P} & \dots & e_P^{\texttt{PMT}_P} \end{bmatrix}}_{\overline{e}_{\texttt{centroid}}} \setminus \begin{bmatrix} r_x^{\texttt{PMT}_1}, r_y^{\texttt{PMT}_1} \\ r_x^{\texttt{PMT}_2}, r_y^{\texttt{PMT}_2} \\ \vdots, \vdots \\ r_x^{\texttt{PMT}_P}, r_y^{\texttt{PMT}_P} \end{bmatrix}. \tag{2.12}$$

The resulting positioning scheme provides a very fast algorithm, with only $2P$ multiplications and summations to achieve the event bidimensional position. However it has severe problems in the location of events in the perimeter of the crystals. This effect is due to the light spreading inside the crystal. The visible photons generated by the interaction of the gamma photon and the crystal spread isotropically. The amount of visible photons per area unit, without considering reflection in the crystal walls, can

be modelled using the inverse square law:

$$\zeta(\boldsymbol{r}, \boldsymbol{r}_0) = \frac{\zeta^{\text{Tot}}}{4\pi|\boldsymbol{r}_i - \boldsymbol{r}_0|^2}, \tag{2.13}$$

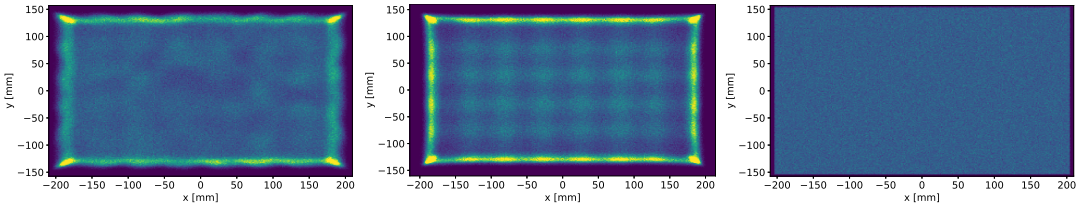
where $\boldsymbol{r}$ is the coordinate at the surface of the PMT array and $\boldsymbol{r}_0$ the coordinate of the impact of the gamma photon. The equation 2.13 is symmetric with the event location $\boldsymbol{r}_0$, making the Anger algorithm correct where this symmetry can be preserved with the P PMTs. In the edges of the crystal such symmetry is broken since the reflections dominate the distribution, resulting in a bad performance of linear methods such as Anger. This area extends up to the size of one PMT, resulting, in the case of the AR-PET in a loss of approximately 50% of the detector's surface.

2.4.2 Look-up tables

To reduce the impact of Anger positioning the most common alternative, with DoI capability, is the use of *Look-up Tables* (LuTs) [38–40]. These methods first construct a LuT using a calibration method, like collimated gamma beams, and modelling of the light distribution to achieve a list of energy distributions $\overline{e}_n$ with their corresponding tridimensional event location $\boldsymbol{r}_n$. Then using MLE [38], kNNs [39] or least-squares [40] to find the closest $\overline{e}_n$ to the detected event energy distribution $\overline{e}_m$. These methods are able to estimate the $\boldsymbol{r}$ with better accuracy than Anger methods, but they are limited by the size of the LuT and the convergence cycles needed to find the best candidate. This becomes a problem given the limited calculation and storage capabilities of FPGAs and the short time between events.

2.4.3 Neural networks

Besides the LUTs, statistical methods were also proposed, such as gradient tree boosting [99], which trains a statistical positioning model based on a fast fan-bean calibration process. The use of NNs was also proposed. Artificial neural networks are known to be able to perform such task and a network with more than 1 hidden layer and enough neurons is able to approximate a wide variety of non-lineal function [95]. In [100] and [101] they propose to train a NN using a dataset constructed from empirical measurements, such as collimated beams. In [37] and [102] they propose to train the NN with a dataset created from a mathematical model, in order to reduce calibration time and complexity. All these methods rely on the generation of a dataset containing paired light distribution and position information. Such dataset is not easy to obtain and is one of the major obstacles in the implementation of these techniques. Also, just as with LuT methods, their final accuracy depends on how dense the dataset is. However they are less memory intensive than LuTs and do not require to iterate to find a solution.



(a) Experimental (b) GATE (c) Expected

Figure 2.8: 2D histogram of the empirical (a) and simulated (b) COG distribution, and expected distribution (c), on the (x, y) plane of the AR-PET detector camera. The non homogeneities on (a) and (b) are caused by the Anger positioning errors.

2.5 Event Localization using distribution matching Neural Networks

Here, a method to correct the distortion of the Anger algorithm and enable DoI estimation using NNs is proposed. The method uses a single acquisition which only requisite is to known its expected interaction distribution. The method does not requires to perform any collimated beam scanning on the crystal surface. It is based on the transformation of the Anger interaction distribution to the expected distribution, correcting the placement interaction by interaction. Moreover, the method is capable of estimating DoI position following a light dispersion positioning strategy. Furthermore the method improves the Full Width at Half Maximum (FWHM) of the COG possitioning method. The method was tested using an acquisition from a detector module of the AR-PET scanner prototype and a Monte Carlo simulation. The method is compared to the COG method and a DoI estimation based on the light dispersion, using collimated beam acquisitions.

2.5.1 Introduction

The distribution of gamma photon interactions within a scintillation crystal depends on the energy, incident angle, position of the source and the stopping power of the crystal's material. This distribution can be empirically observed by applying a 3D histogram to a large set of interactions. A perfect positioning algorithm will recover the theoretical interaction distribution within the crystal, however, a real positioning algorithm will generate a different distribution. For example, a point source placed far from a detector's center will produce a uniform distribution of interactions, along the surface of the detector. If these interactions are positioned using the Anger algorithm, which suffers from distortion at the crystal edges, the resulting histogram will posses a packing of interactions in the detector's edge as seen in figure 2.8. The packing effect observed in the boundary of the detector leads to disable of part of the sensible area, since the localization is not trustworthy. This area has the size of a PMT unit and is called *dead zone*.

This difference between the expected distribution and the estimated distribution can be used as a metric of the packing effect, or distortion, of the positioning algorithm. More formally, if the expected interaction distribution $\boldsymbol{R}$ and the positioning or estimated distribution $\hat{\boldsymbol{R}}$ are known, then a distance measure between the distributions can be expressed as:

$$d = J_{\texttt{dist}}(\boldsymbol{R}, \hat{\boldsymbol{R}}), \tag{2.14}$$

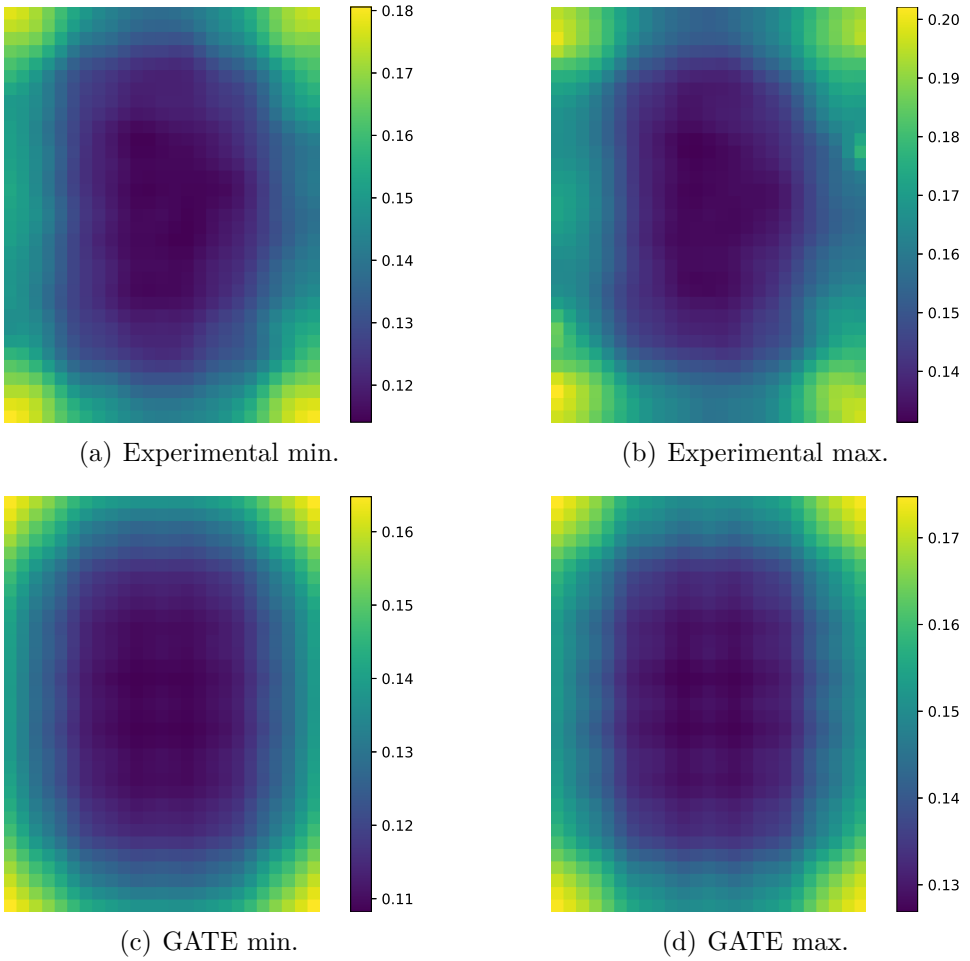
where $J_{\texttt{dist}}()$ is a distance function. The shape of the distance function should take into account each particular event separately, since their individual position is important. These metrics are normally used in the Optimal Transport (OT) theory. They measure this distance as the minimum cost of moving each sample from one distribution to the position of a sample in the objective distribution. Possible candidates for $J_{\texttt{dist}}()$ are the Earth Movers Distance (EMD), also known as Wasserstein metric [103] (implemented for large scale OT in [104]), or the Chamfer Distance Metric (CDM). The CDM is easier to calculate than EMD and provides accurate gradients resulting in similar performace than EMD in NN training [105]. The CDM finds for each event in the first set its nearest neighbour in the second set, calculates the squared distance between them, and adds all these distances. It can be expressed as:

$$J_{\texttt{dist}}(\boldsymbol{R}, \hat{\boldsymbol{R}}) = \sum_{\boldsymbol{r}_n \in \boldsymbol{R}} \min_{\hat{\boldsymbol{r}}_m \in \hat{\boldsymbol{R}}} \|(\boldsymbol{r}_n - \hat{\boldsymbol{r}_m})\|_2^2 + \sum_{\hat{\boldsymbol{r}}_m \in \hat{\boldsymbol{R}}} \min_{\boldsymbol{r}_n \in \boldsymbol{R}} \|(\boldsymbol{r}_n - \hat{\boldsymbol{r}}_m)\|_2^2, \tag{2.15}$$

where $\boldsymbol{r}$ is the 3D location of an interaction. Using equation 2.15 a NN can be optimized to reduce this distance, provided a known objective distribution $\boldsymbol{R}$ from which is possible to sample. Hence, minimizing d results in the minimization of the error between the expected interaction position and the estimated event position.

2.5.2 Data acquisition

The data required to train the network is obtained through flood acquisition using a distant isotropic gamma source. Although no distribution is enforced by the presented method, for simplicity a uniform and orthogonal field is used.The field is created using a $^{22}\texttt{Na}$ β^+ (creating 511 $\texttt{keV}$ photons by means of electron-positron annihilation) source placed at a large distance $d_s > 3$ $\texttt{m}$, aligned to the detector's surface center. Given the distance of the source to the detector and the size of the detector's surface ($406.4 \times 304.8 \times 25.4$ $\texttt{mm}^3$), the incident gamma photons can be considered perpendicular to the crystal surface. Using this configuration, $N = 3 \times 10^7$ sample points (interactions) are collected. The events were filtered using an energy window between 470 $\texttt{keV}$ and 530 $\texttt{keV}$. Each interaction n is recorded as the energy distribution $e_p^n \in \overline{e}_n$ detected on each detector p of the scintillation camera. The baseline dataset $\tilde{\boldsymbol{R}}$ is then constructed by calculating for each event $\overline{e}_n$ its estimated interaction position $\tilde{\boldsymbol{r}}_n = (r_x^n, r_y^n, r_z^n)$. The position on the plane (r_x^n, r_y^n) is calculated using the Anger method. The DoI (r_z^n) is obtained using a simple estimation based on the detected light dispersion, a known indicator of DoI [106]. This metric is built using the dataset energy distribution information. First the detectors surface is divided into D pixels of size 12×12 $\texttt{mm}^2$.



(a) Experimental min. (b) Experimental max.

(c) GATE min. (d) GATE max.

Figure 2.9: Minimum (a,c) and maximum (b,d) values of the dispersion feature per virtual pixel of the AR-PET detector for the experimental acquisition (a, b) and the GATE simulation (c, d).

Then, using the Anger information the dataset is split into subgroups (a group for each pixel). For each of the events in each group, the relation of the energy detected in the 4 PMTs with higher energy to the total energy is calculated:

$$\eta_n^k = \frac{\sum_{p=1}^{4} e_p^{n*}}{\sum_{p=1}^{48} e_p^{n*}}, \tag{2.16}$$

where $\overline{e}_n^*$ are the four largest values n^{th} event, k marks the corresponding pixel and p the PMT number. For each pixel k, the distribution of the η^k values is analysed to remove outliers, and then it is fitted using a normal distribution. Finally a minimum $\eta_{\texttt{min}}$ and maximum $\eta_{\texttt{max}}$ map of η^k values is constructed using the 2σ interval of the fitted distributions. The maximum and minimum maps can be seen in figure 2.9.

Knowing the maximum and minimum dispersion per pixel, the DoI of an event n

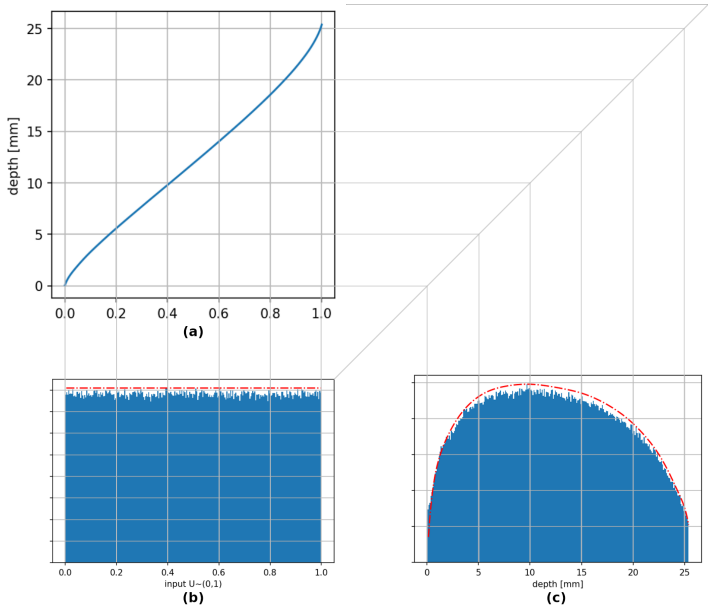


Figure 2.10: Sampling process of Z-axis distribution. in (b) is the $U(0,1)$ sampler, which is converted trough the inverse CDF in (a) and results in the objective distribution shown in (c). The red dotted lines represent the input and output PDF, below them the histogram bins are presented.

interacting in the k^{th} pixel is estimated linearly:

$$r_z^n = D_{\text{thkns}} \frac{\eta_n - \eta^k_{\texttt{max}}}{\eta^k_{\texttt{min}} - \eta^k_{\texttt{max}}}, \tag{2.17}$$

where D_{thkns} is the detector's thickness. A simulated dataset was also generated with a Monte Carlo simulation of the experimental setup using the GATE software [107]. The simulation included the tracking of optical photons and all reflective surfaces of the scintillation camera. The simulated dataset collected similar amount of events than the experimental setup.

2.5.3 Objective Distribution Sampling

The objective distribution is obtained by a GATE Monte Carlo simulation of the experimental setup. Using this simulation the cumulative density function (CDF) in each axis is obtained. Note that this simulation does not require optical photon tracking. Given the experimental conditions, the (X, Y) coordinates of the interactions follow $U(-203.0, 203.0)$ and $U(-152.0, 152.0)$ respectively. The Z-axis the expected distribution for a NaI(TI) crystal [79]. The z-axis is then sampled using the inverse CDF method [108]. To achieve this, the CDF is calculated and the inverse of the CDF is approximated using a polynomial fit. The polynomial function is then used to convert samples drawn from $U(0,1)$ to the objective distribution. In figure 2.10 the sampling procedure is shown.

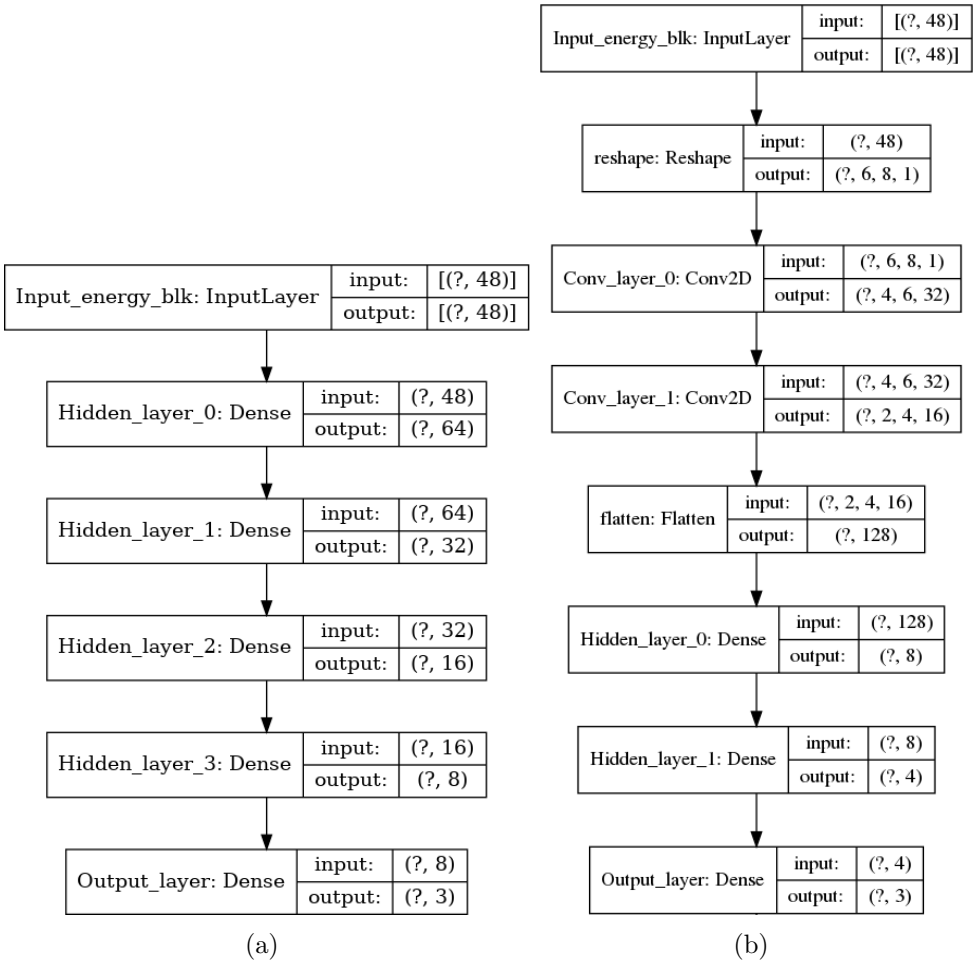


Figure 2.11: Fully connected (a) and convolutional (b) event positioning network topologies.

2.5.4 Neural Network Topology

In order to fit the interaction localization function two network topologies are tested, a Fully Connected Network (FCN) and a Convolutional Neural Network (CNN)[1]. The network topologies can be seen in figure 2.11. The first topology (in figure 2.11(a)) is a fully connected network, composed of five dense layers of 64, 32, 16 and 8 neurons, followed by an output fully connected layer with 3 neurons (one per output). The second topology (in figure 2.11(b)) is composed of two convolutional layers with kernel size of 3×3 with 32 and 16 channels respectively, followed by two dense layers of 8 and 4 neurons and finalizing with a 3 neuron fully connected layer. All activations, except the outputs, are Rectifier Linear Units (ReLU) to avoid the use of non linear functions in the detectors FPGAs. The output neurons use a linear activation. The

[1] A larger introduction to NN is given in chapter 4

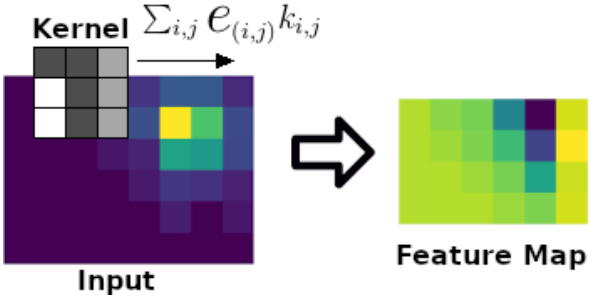


Figure 2.12: Input convolutional operation of the convolutional event positioning neural network.

ReLU activation is described as:

$$ReLU(x) = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases}. \tag{2.18}$$

Each topology features their own characteristics. Fully connected networks are simple to compute but require many parameters (one per each input to neuron pair). On the other hand convolutional layers with a *valid-convolution* strategy reduce the network size and they are well suited to capture the spatial information of the light shield. In figure 2.12 the initial convolution operation of the convolutional network is displayed, where the input energies of each PMTs (the intensity of a 6×8 px 2D image) are convolved using an input kernel. Given the size of the *Hamamatsu R1534* PMTs the input kernel size was chosen to be 3×3 px.

Thus, each type of layer will have a different number of parameters. For fully connected layers the number of parameters of the i^{th} layer is calculated as:

$$\mathtt{param}_{\mathtt{fc}}^{[i]} = \boldsymbol{W}^{[i]} + \boldsymbol{b}^{[i]} = \boldsymbol{n}_x^{[i]} \boldsymbol{n}_h^{[i]} + \boldsymbol{n}_h^{[i]}, \tag{2.19}$$

where $\boldsymbol{W}^{[i]}$ is the number of neuron weights, $\boldsymbol{b}^{[i]}$ the number of bias, calculated using the number of hidden neurons in the layers $\boldsymbol{n}_h^{[i]}$ and the number of input parameters $\boldsymbol{n}_x^{[i]}$. The number of parameters for a convolutional layer is calculated as:

$$\mathtt{param}_{\mathtt{conv}}^{[i]} = \boldsymbol{k}_x^{[i]} \boldsymbol{k}_y^{[i]} p_c^{\mathtt{in}} \boldsymbol{n}_f^{[i]} + \boldsymbol{n}_f^{[i]}, \tag{2.20}$$

where $(\boldsymbol{k}_x^{[i]}, \boldsymbol{k}_y^{[i]})$ is the size of the kernel or filter, $p_c^{\mathtt{in}}$ is the number of channels of the input image or feature map and $\boldsymbol{n}_f^{[i]}$ is the number of filters. The number of operations performed by each type of layer also differs, for a fully connected layer the number of operations is:

$$\mathcal{O}_{\mathtt{fc}}^{[i]} = (2\boldsymbol{n}_x^{[i]} \boldsymbol{n}_h^{[i]} + \boldsymbol{n}_h^{[i]}) + \boldsymbol{n}_h^{[i]} \mathcal{O}(g^{[i]}) = 2\,\mathtt{param}_{\mathtt{fc}}^{[i]} + \boldsymbol{n}_h^{[i]} (\mathcal{O}(g^{[i]}) - 1), \tag{2.21}$$

where $\mathcal{O}(g^{[i]})$ is the cost of the non-linearity operation. The number of operations for

Topology	Number of parameters	Number of operations
Fully Connected	5907	11814
Convolutional	6027	92438

Table 2.2: Number of parameters and computational cost (in number of operations) of each tested topology.

the convolutional layer is calculated as:

$$\mathcal{O}^{[i]}_{\texttt{conv}} = p^{\texttt{out}}_x p^{\texttt{out}}_y (2\,\texttt{param}^{[i]}_{\texttt{conv}} + \mathcal{O}(g^{[i]})) + \boldsymbol{n}^{[i]}_h, \tag{2.22}$$

with (for a valid convolution operation):

$$p^{\texttt{out}}_x = p^{\texttt{in}}_x - \lceil \frac{\boldsymbol{k}^{[i]}_x}{2} \rceil, \tag{2.23}$$

$$p^{\texttt{out}}_y = p^{\texttt{in}}_y - \lceil \frac{\boldsymbol{k}^{[i]}_y}{2} \rceil, \tag{2.24}$$

where $(p^{\texttt{in}}_x, p^{\texttt{in}}_y)$ is the size of the input image or feature map. The number of parameters and computational cost are an important factor when the network must be embedded in FPGAs within the detector cameras. The number of parameters and costs for both topologies is presented in table 2.2. The network topology sizes were chosen to have similar weight number.

2.5.5 Training Scheme

The training of the NN (FC or CNN) is a two step procedure. First the network is trained to copy the positioning of the Anger method and the depth estimation. The second step concerns the correction of the positioning errors through a distribution matching training. During the first stage the NN is trained to learn a base positioning task (such as Anger positioning) using a simple euclidean loss:

$$J_{\texttt{base}}(\tilde{\boldsymbol{R}}, \hat{\boldsymbol{R}}) = \frac{1}{N} \sum_{n=1}^{N} \|\tilde{\boldsymbol{r}}_n - \hat{\boldsymbol{r}}_n\|_2^2, \tag{2.25}$$

where N is the number of interactions in the dataset and $\tilde{\boldsymbol{r}}_n \in \tilde{\boldsymbol{R}}$ the position of the interaction obtained using the base method. This training is done using the Adaptive Moments Estimation (ADAM) algorithm [109], with learning rate $\epsilon = 0.001$. The NN is trained for 500 epochs and with a batch size of $N' = 5 \times 10^3$ interactions. On the second training step the optimization is not fully supervised, since there is no information on the exact interaction of each event within the crystal. Instead, the network is trained for 500 epochs using the ADAM algorithm, with indirect supervision using the distribution of the events within a batch of training data. This is achieved

by adding the matching loss term presented in equation 2.15:

$$J_{\texttt{match}} = K\, J^{*}_{\texttt{base}}(\tilde{\boldsymbol{R}}, \hat{\boldsymbol{R}}) + J_{\texttt{dist}}(\boldsymbol{R}, \hat{\boldsymbol{R}}), \tag{2.26}$$

where $K = 0.001$ is a coupling constant and $J^{*}_{\texttt{base}}()$ is a weighted version of equation 2.25:

$$J^{*}_{\texttt{base}}(\tilde{\boldsymbol{R}}, \hat{\boldsymbol{R}}) = \frac{1}{N'} \sum_{n=1}^{N'} f_{\texttt{trust}}(\tilde{\boldsymbol{r}}_n) \|\tilde{\boldsymbol{r}}_n - \hat{\boldsymbol{r}}_n\|_2^2, \tag{2.27}$$

where $f_{\texttt{trust}}(\boldsymbol{r}_n)$ a trust modulation term for the sample estimated location using the base method $\tilde{\boldsymbol{r}}_n$. The trust modulation function $f_{\texttt{trust}}()$ is a custom function that provides a value between 0 and 1 for each region of the crystal. It allows the distribution matching function to focus on the regions where the estimation methods fail. This function's behaviour is based on the light distribution function presented in equation 2.13, where events in the middle of the crystal are able to distribute their light shield along a large number of PMTs, making Anger method a good approximation but events near the crystals edges are highly distorted. Thus the trust map is given by:

$$f_{\texttt{trust}}(\boldsymbol{r}_n) = f^{x}_{\texttt{trust}}(r_x) f^{y}_{\texttt{trust}}(r_y) f^{z}_{\texttt{trust}}(r_z), \tag{2.28}$$

with

$$f^{x}_{\texttt{trust}}(r_x) = \begin{cases} 1 - (K_{\texttt{drop}} |D^{\texttt{min}}_{\text{X,trust}} - r_x|)^2, & \text{if } r_x \leq D^{\texttt{min}}_{\text{X,trust}} \\ 1 - (K_{\texttt{drop}} |D^{\texttt{max}}_{\text{X,trust}} - r_x|)^2, & \text{if } r_x \geq D^{\texttt{max}}_{\text{X,trust}} \\ 1, & \text{otherwise} \end{cases}, \tag{2.29}$$

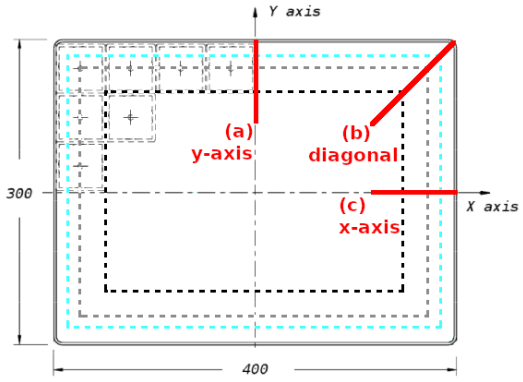
$$f^{y}_{\texttt{trust}}(r_y) = \begin{cases} 1 - (K_{\texttt{drop}} |D^{\texttt{min}}_{\text{y,trust}} - p_y|)^2, & \text{if } r_y \leq D^{\texttt{min}}_{\text{y,trust}} \\ 1 - (K_{\texttt{drop}} |D^{\texttt{max}}_{\text{y,trust}} - p_y|)^2, & \text{if } r_y \geq D^{\texttt{max}}_{\text{y,trust}} \\ 1, & \text{otherwise} \end{cases}, \tag{2.30}$$

where $K_{\texttt{drop}} = 3 \times 10^{-4}$ is a heuristic constant, $D^{\texttt{min}}_{\texttt{trust}}$ and $D^{\texttt{max}}_{\texttt{trust}}$ are the limits of the trustworthy regions of the Anger positioning method. The limits in this case are chosen to be the size of one PMT (52.4 mm) from the edges for the (X, Y) position. Resulting in $D^{\texttt{min}}_{\text{X,trust}} = -150.6$ mm, $D^{\texttt{max}}_{\text{X,trust}} = 150.6$ mm, $D^{\texttt{min}}_{\text{y,trust}} = -99.6$ mm and $D^{\texttt{max}}_{\text{y,trust}} = 99.6$ mm. For the depth estimation, the factor $f^{z}_{\texttt{trust}}(p_z)$ is a constant, since not region is justifiably more trustworthy than other.

2.5.6 Testing

The proposed method was validated using two different NNs and two different datasets. Each of the proposed topologies, the FCN and the CNN were trained with the experimental dataset and the simulated dataset. The four experiments are:

- **ANGER_FCN**: Fully connected network, trained with the experimental dataset, using Anger as base method.



(a)

Figure 2.13: Collimator placements on the (x, y) plane of the detector's surface. The y-axis series (a), the *diagonal* beam series (b) and the x-axis series (c). The black, gray and cyan dashed lines represent a margin of one, half and a quarter PMT from the detector's edge respectively.

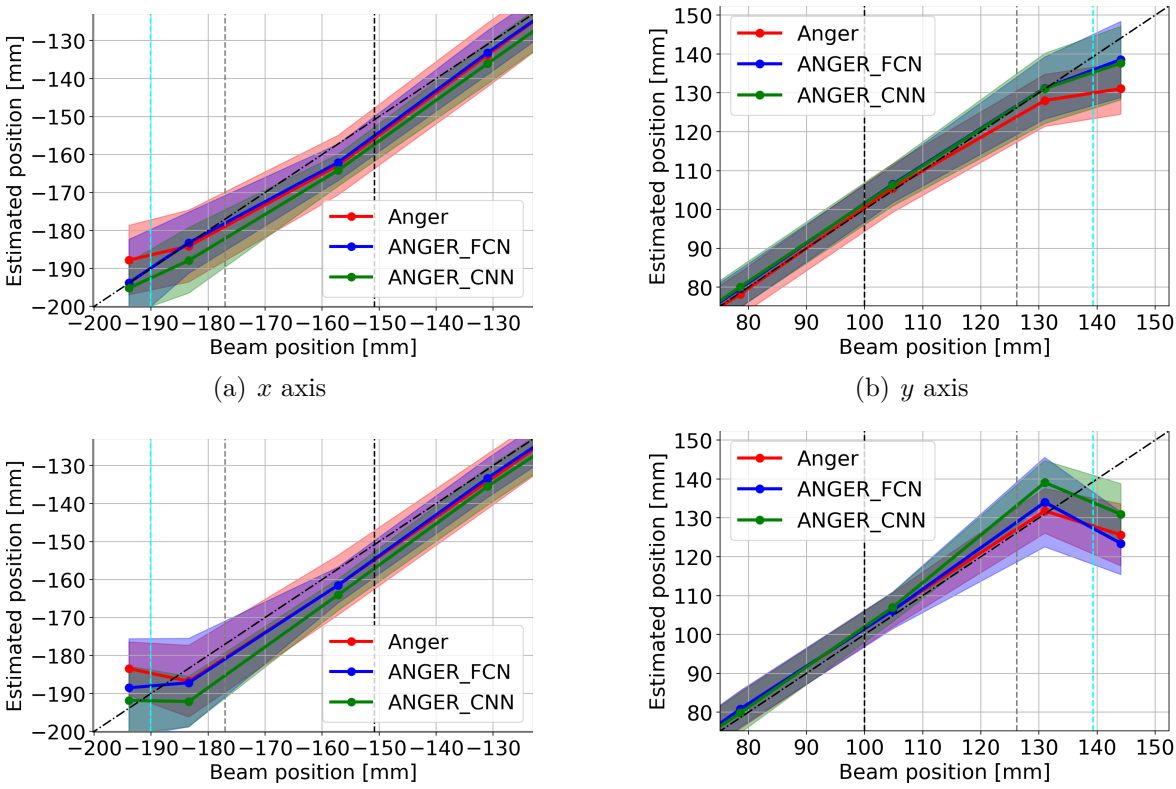
- **ANGER_CNN**: Convolutional neural network, trained with the experimental dataset, using Anger as base method.
- **Artf_ANGER_FCN**: Fully connected network, trained with the GATE simulation dataset, using Anger as base method.
- **Artf_ANGER_CNN**: Convolutional neural network, trained with the GATE simulation dataset, using Anger as base method.

2D Performance: Perpendicular collimated beams

Using the technique described in [101] three series of collimated beams were acquired: (y axis) originates at $(0.0, 72.15)$ `mm` and extends towards $(0.00, 144.38)$ `mm`, (x axis) originates at $(-123.0, 0.0)$ `mm` and extends towards $(-194.25, 0.00)$ `mm` and (*Diagonal*) starts at $(-123.0, 72.15)$ extending diagonally towards $(-194.25, 144.38)$. All series have a step size of 26.2 `mm`. The location of each series can be seen in Fig 2.13. The performance curves for the Anger method, the ANGER_FCN and the ANGER_CNN for each series, is shown in figure 2.14. The mean FWHM, mean bias and max bias for each interaction position estimator, collimated beam series and detector's dead zone are show in Table 2.3.

2D Performance: GATE test

Using the available information of the Monte Carlo simulation, the *Artf_ANGER_FCN* and the *Artf_ANGER_CNN* were tested against the ground truth and the Anger algorithm. The tested regions were the same as the collimated beams series. The simulated series have a step of 2.5 `mm`. The performance of the Artf_ANGER_FCN, the Artf_ANGER_CNN and the Anger method are shown in figure 2.15. The mean



(a) x axis

(b) y axis

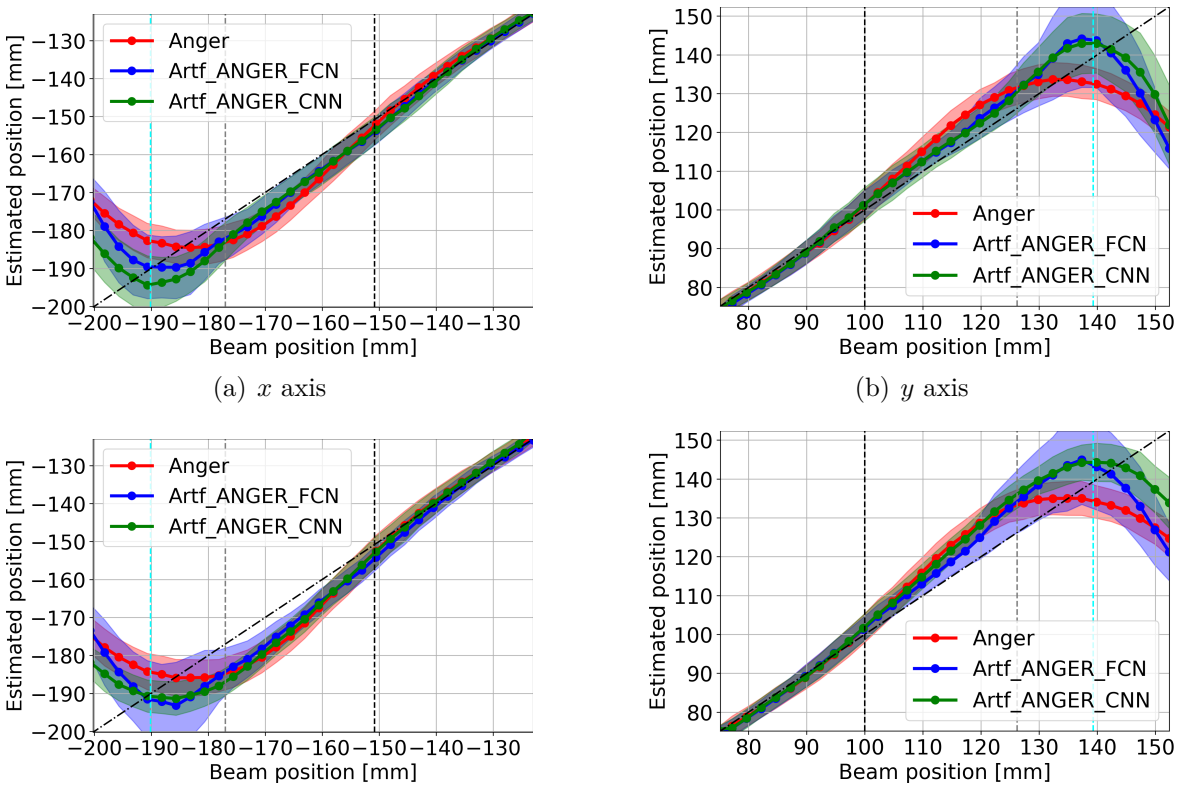
(c) Diagonal on x axis

(d) Diagonal on y axis

Figure 2.14: Experimental measurement: Performance on x axis series (a), y axis series (b), *Diagonal* series over x axis (c) and *Diagonal* series over y axis (d). The bold line represents the mean position and the coloured area represents the FWHM. The vertical black, gray and cyan dashed lines represent a distance of one, half and a quarter PMT form the detectors edge respectively.

Beam	Estimator	Dead zone size: 52.4 mm			26.2 mm			13.1 mm		
		$\overline{\text{FWHM}}$	$\bar{b}$	max b	$\overline{\text{FWHM}}$	$\bar{b}$	max b	$\overline{\text{FWHM}}$	$\bar{b}$	max b
x axis	Anger	12.08	2.74	3.72	12.20	3.15	5.63	12.64	2.85	5.63
	ANGER_FCN	10.38	2.73	3.80	10.23	3.11	5.37	10.80	2.78	5.37
	ANGER_CNN	10.35	5.07	5.07	9.27	6.02	6.96	10.94	5.51	6.96
y axis	Anger	10.38	1.44	3.22	10.67	1.24	3.22	11.04	1.58	3.22
	ANGER_FCN	10.50	2.10	3.34	10.47	1.98	3.34	11.06	1.73	3.34
	ANGER_CNN	10.40	1.82	3.34	10.52	1.73	3.34	11.36	1.41	3.34
Diagonal - x axis	Anger	13.41	2.76	3.21	13.96	3.15	4.30	14.91	3.19	4.30
	ANGER_FCN	10.06	2.39	2.80	9.88	2.75	3.82	11.62	3.05	4.26
	ANGER_CNN	9.84	4.51	4.51	9.04	5.67	6.83	10.40	6.69	8.73
Diagonal - y axis	Anger	10.25	1.13	1.89	9.88	1.21	1.89	10.19	1.13	1.89
	ANGER_FCN	9.82	1.48	2.27	9.70	1.22	2.27	12.58	1.81	4.15
	ANGER_CNN	10.33	1.24	1.24	8.88	1.69	2.15	9.64	3.82	8.08

Table 2.3: Experimental measurement: mean FWHM ($\overline{\text{FWHM}}$), mean bias ($\bar{b}$) and maximum bias (max b), for each beam series and each interaction position estimator. These metrics were tested using detector's dead zone of size one, half and one quarter PMT (52.4 mm, 26.2 mm, 13.1 mm respectively). All metrics are expressed in [mm].



(a) x axis

(b) y axis

(c) Diagonal on x axis

(d) Diagonal on y axis

Figure 2.15: GATE simulation: Performance on x axis series (a), y axis series (b), *Diagonal* series over x axis (c) and *Diagonal* series over y axis (d). The bold line represents the mean position and the coloured area represents the FWHM. The vertical black, gray and cyan dashed lines represent a distance of one, half and a quarter PMT form the detectors edge respectively.

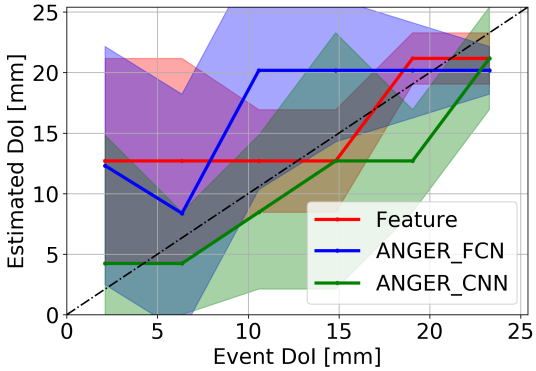
FWHM, mean bias and max bias for each position estimator and collimated beam series is show in Table 2.4.

DoI Performance: Diagonal collimated beams

To test the DoI estimation performance of the NN, a collimated beam with an incidence angle of 45° was measured in the central area of the detector's surface. The beam was located at $(0.5, 0.5)$ `mm` and its incidence followed the x axis. Given the incidence angle, the DoI is expected to follow the x axis deviation from the impact center. Also the $r_n^z = r_n^x - r^x_{\texttt{impactcenter}}$. The DoI was divided in 6 sections of 4.23 `mm` and the mean and FWHM for each section was calculated. The performance curves for the feature based DoI method, the ANGER_FCN and the ANGER_CNN are shown in figure 2.16. The correlation factor between the estimated DoI and the expected DoI was calculated and is summarized on table 2.5.

Beam	Estimator	Dead zone size								
		52.4 mm			26.2 mm			13.1 mm		
		$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$
x axis	Anger	6.00	0.88	1.46	6.99	3.32	8.31	7.37	3.32	8.31
	Artf_ANGER_FCN	5.46	0.89	3.17	6.72	2.54	5.96	8.25	2.84	5.96
	Artf_ANGER_CNN	5.90	1.19	2.92	7.85	4.19	5.39	11.31	6.84	7.74
y axis	Anger	5.81	0.80	1.30	6.64	3.05	7.31	7.02	3.04	7.31
	Artf_ANGER_FCN	5.49	1.25	2.78	6.50	2.05	4.18	9.15	2.87	8.11
	Artf_ANGER_CNN	5.78	0.88	1.42	8.49	2.61	3.41	13.61	6.01	6.94
Diagonal - x axis	Anger	6.10	0.91	1.93	7.01	3.92	9.87	7.43	3.88	9.87
	Artf_ANGER_FCN	5.16	1.05	3.64	6.25	3.42	7.84	8.63	4.04	7.84
	Artf_ANGER_CNN	6.00	1.13	2.12	6.96	7.14	9.98	8.08	6.94	9.87
Diagonal - y axis	Anger	5.94	0.89	1.55	6.70	3.65	8.87	7.02	3.58	8.87
	Artf_ANGER_FCN	5.20	0.92	1.55	6.69	2.49	7.73	9.26	3.62	8.68
	Artf_ANGER_CNN	5.20	0.96	1.85	6.58	6.26	9.80	8.12	8.79	9.91

Table 2.4: GATE simulation: mean FWHM ($\overline{\texttt{FWHM}}$), mean bias ($\bar{b}$) and maximum bias ($\max b$), for each beam series and each interaction position estimator. These metrics were tested using detector's dead zone of size one, half and one quarter PMT (52.4 mm, 26.2 mm, 13.1 mm respectively). All metrics are expressed in [mm].



(a)

Figure 2.16: Experimental measurement: DoI performance on central area. The bold line represents the mean position and the coloured area represents the FWHM.

Estimator	Center Area	Edge Area
Feature based DoI	0.28	–
ANGER_FCN	0.21	–
ANGER_CNN	0.23	–

Table 2.5: Experimental measurement: Correlation factor between the expected DoI and the estimated DoI in the central and edge areas of the detector's surface.

DoI Performance: GATE test

The Artf_ANGER_FCN, Artf_ANGER_CNN and the feature based DoI method was tested using a GATE simulation. Two regions were tested: A central region of the detectors surface, centred at $(0.0, 0.0)$ mm and with area 50.0×50.0 mm^2, and an edge region, centred at $(0.0, -133.75)$ mm and with area 50.0×50.0 mm^2. The reference DoI values were obtained from the simulation information. The performance curves for each region and method can be seen in figure 2.17. The correlation factor between the estimated DoI and the expected DoI was also calculated and is summarized on table 2.6.

Estimator	Center Area	Edge Area
Feature based DoI	0.79	0.50
Artf_ANGER_FCN	0.80	0.53
Artf_ANGER_CNN	0.83	0.56

Table 2.6: GATE simulation: Correlation factor between the expected DoI and the estimated DoI in the central and edge areas of the detector's surface.

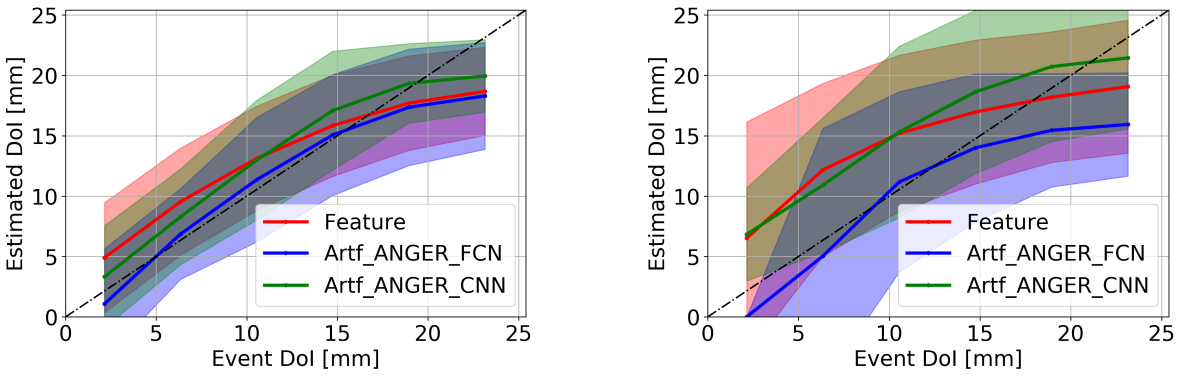


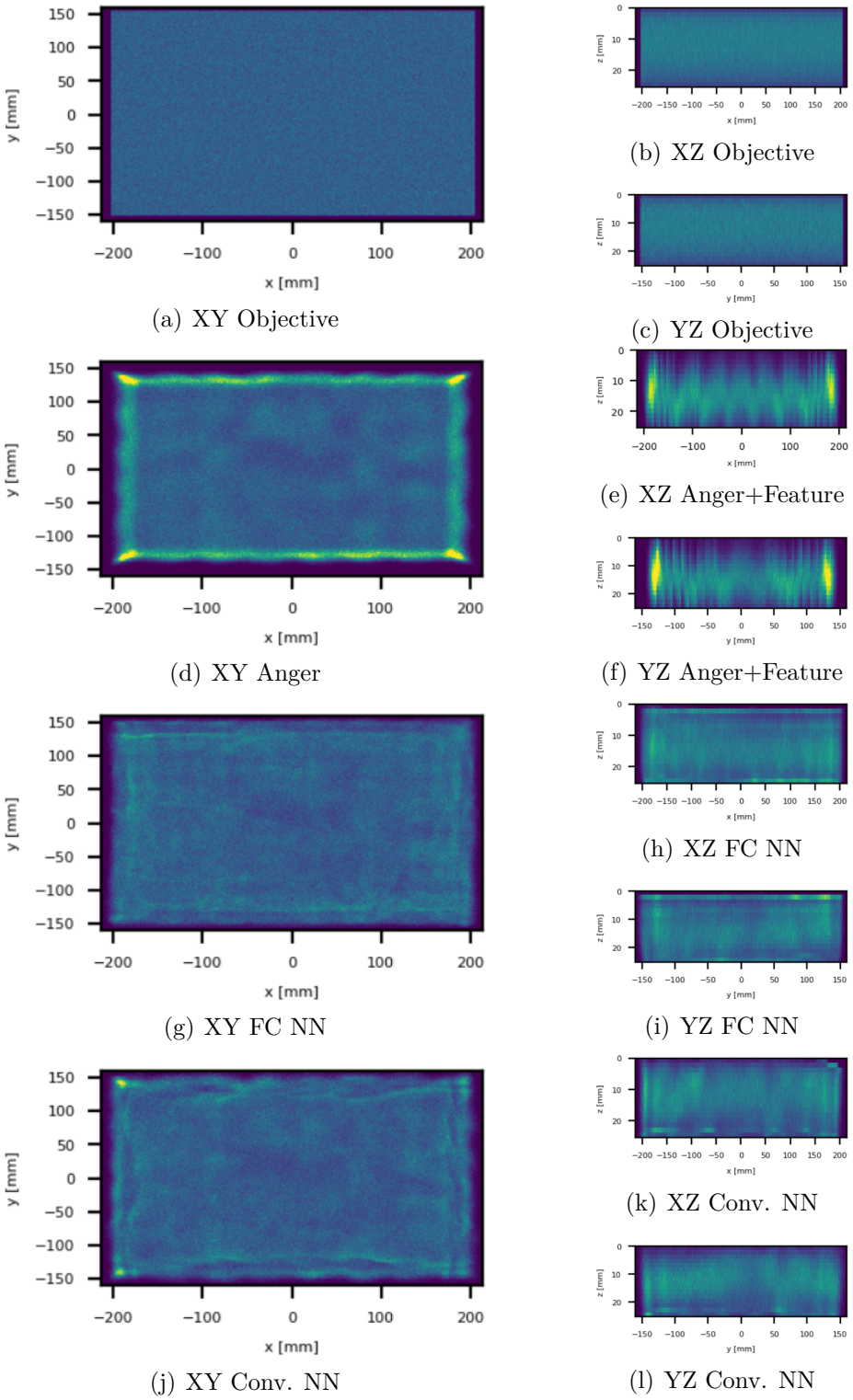
Figure 2.17: GATE simulation: DoI performance on central area (a) and edge area (b). The bold line represents the mean position and the coloured area represents the FWHM.

Distribution analysis

The packing of the resulting distributions is analysed in the (x, y) plane using a 2D histogram of the events in the dataset. For each localization method a 2D histogram with 1.56×1.56 $\texttt{mm}^2$ bin size was calculated. The resulting (x, y) histograms can be seen in figure 2.18 for the experimental measurement and in figure 2.19 for the GATE simulation. The relocation maps are presented in figure 2.20 for the experimental measurement and in figure 2.21 for the GATE simulation. The relocation maps show the initial (base method) and final position (NN correction) of each event and the color of each line reflects the length of the interaction displacement.

For each tested method the empirical density estimate along the x and y axis was calculated using the histogram over each axis. The empirical density was also computed for the z axis, in a central and an edge area. The central area was located at $(0.0, 0.0)$ $\texttt{mm}$ and the edge region was located at $(0.0, -133.75)$ $\texttt{mm}$. Both regions had an extension of 50.0×50.0 $\texttt{mm}^2$. The resulting densities, along with the objective densities are shown in figure 2.22 for the empirical measurement and the GATE simulation.

The irregularity of the distribution was measured using the normalized count difference $\Delta C = (C_{\texttt{max}} - C_{\texttt{min}})/C_{\texttt{max}}$, where $C_{\texttt{max}}$ and $C_{\texttt{min}}$ are the maximum and minimum bin values. The metric was applied on the empirical density estimation on x and y axis. For each axis three metrics were calculated, one with a 52.4 $\texttt{mm}$ margin, one with a 26.2 $\texttt{mm}$ margin and one with a 13.1 $\texttt{mm}$ margin (one, half and one quarter PMT size respectively). The results are summarized in Table 2.7.



(a) XY Objective

(b) XZ Objective

(c) YZ Objective

(d) XY Anger

(e) XZ Anger+Feature

(f) YZ Anger+Feature

(g) XY FC NN

(h) XZ FC NN

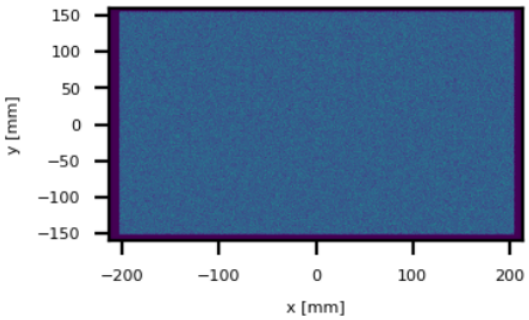
(i) YZ FC NN

(j) XY Conv. NN

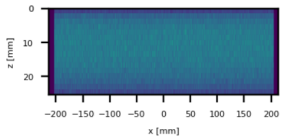
(k) XZ Conv. NN

(l) YZ Conv. NN

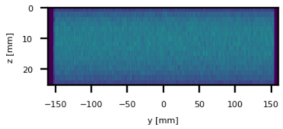
Figure 2.18: Experimental measurement: Histograms of events in (x, y) plane (a,d,g,j), (x, z) plane (b,e,h,k) and (y, z) plane (c,f,i,l) of the objective distribution (a,b,c), the estimated distribution (d,e,f), the ANGER_FCN (g,h,i) and the ANGER_CNN (j,k,l).



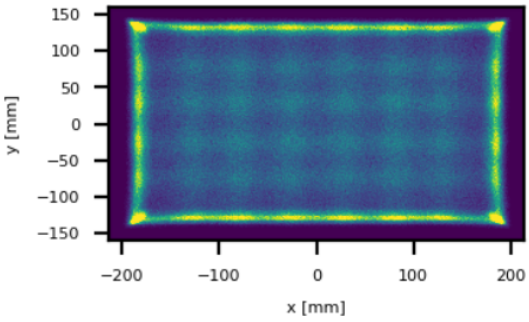
(a) XY Objective



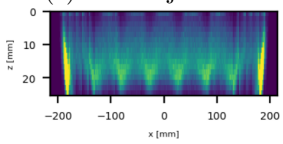
(b) XZ Objective



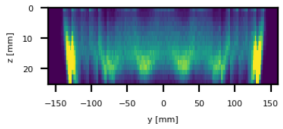
(c) YZ Objective



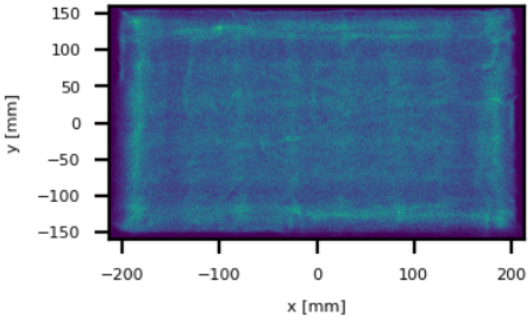
(d) XY Anger



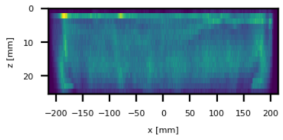
(e) XZ Anger+Feature



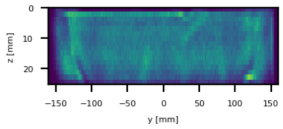
(f) YZ Anger+Feature



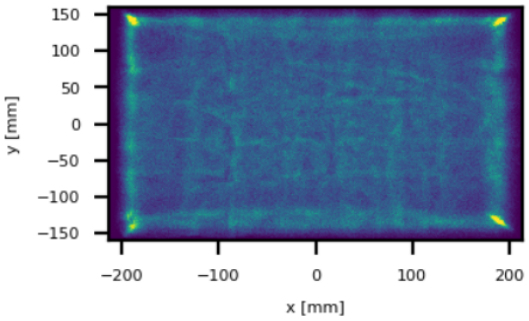
(g) XY FC NN



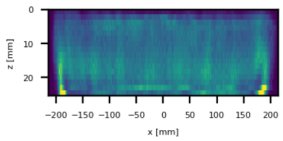
(h) XZ FC NN



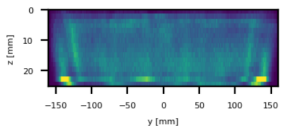
(i) YZ FC NN



(j) XY Conv. NN



(k) XZ Conv. NN



(l) YZ Conv. NN

Figure 2.19: GATE simulation: Histograms of events in (x, y) plane (a,d,g,j), (x, z) plane (b,e,h,k) and (y, z) plane (c,f,i,l) of the objective distribution (a,b,c), the estimated distribution (d,e,f), the Artf_ANGER_FCN (g,h,i) and the Artf_ANGER_CNN (j,k,l).

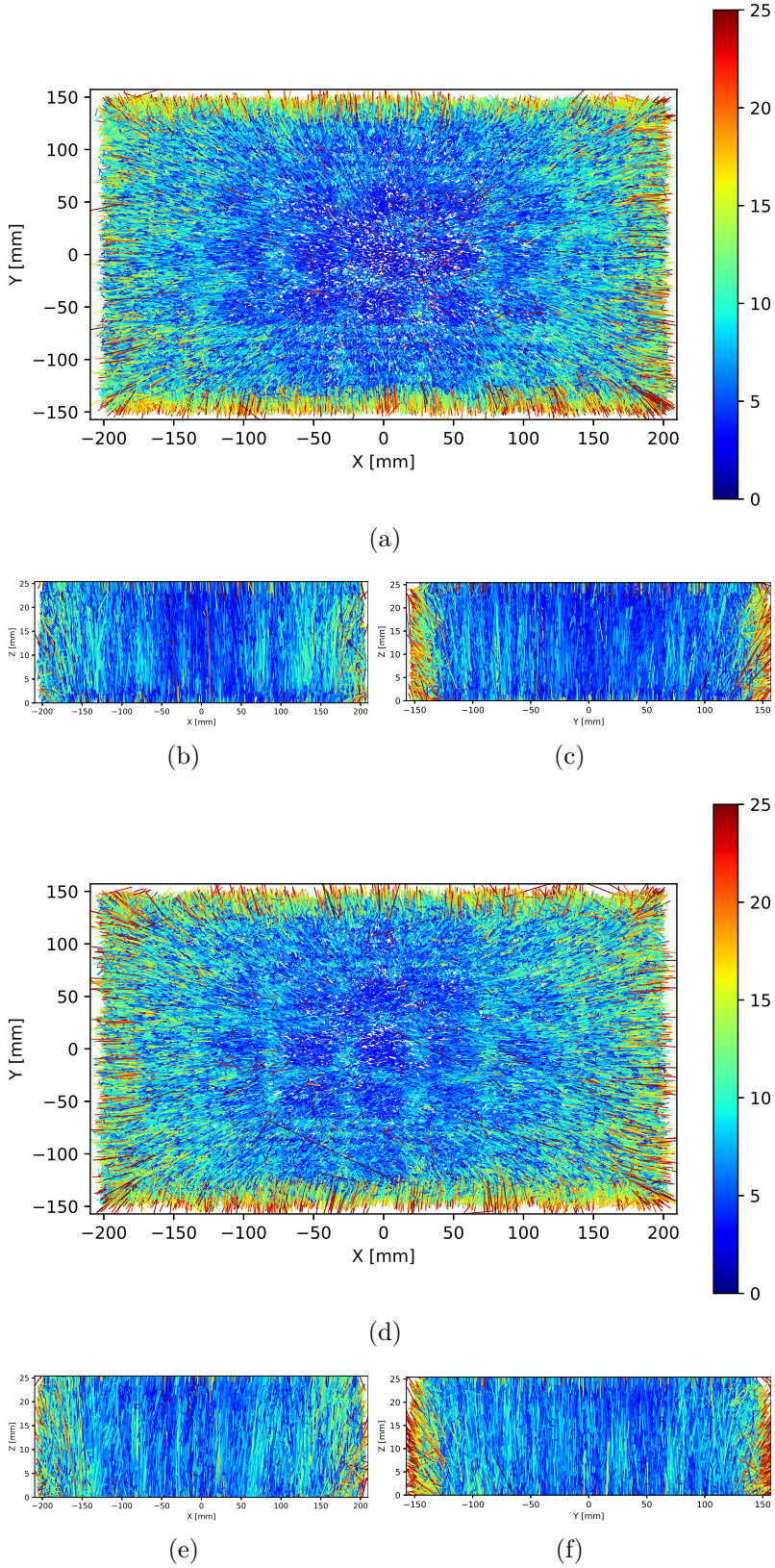
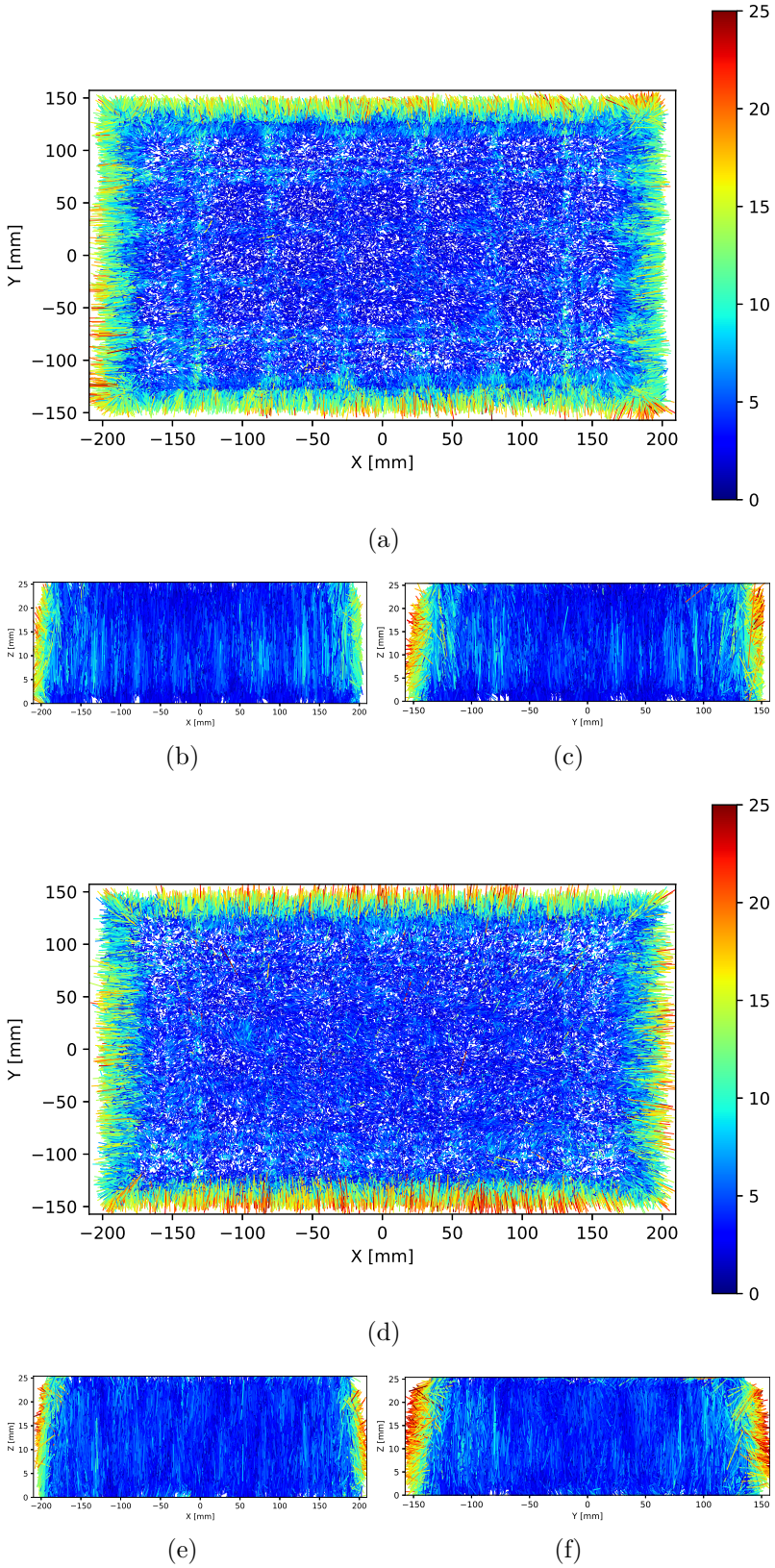
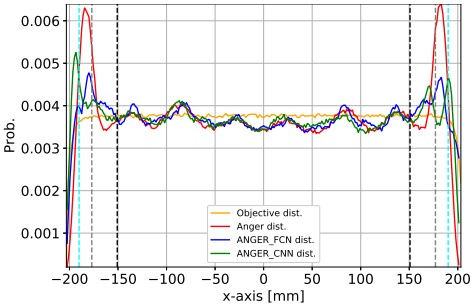


Figure 2.20: Experimental measurement: Relocation maps of events in (x, y) plane (a,d), (x, z) plane (b,e) and (y, z) plane (c,f) of the the ANGER_FCN topology (a,b,c) and the ANGER_CNN topology (d,e,f). The line color represents the line length.

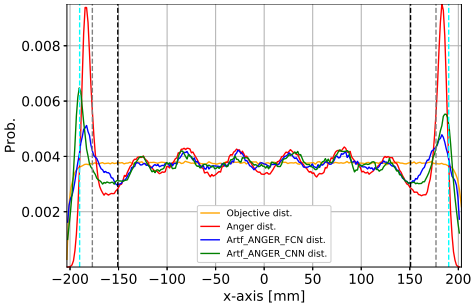


(a) (b) (c) (d) (e) (f)

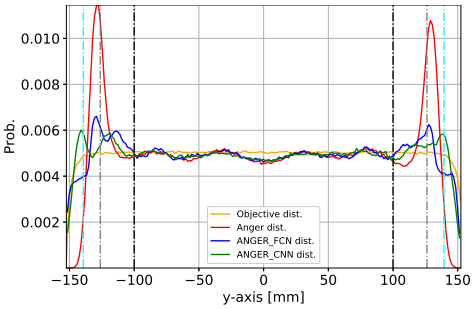
Figure 2.21: GATE simulation: Relocation maps of events in (x, y) plane (a,d), (x, z) plane (b,e) and (y, z) plane (c,f) of the the Artf_ANGER_FCN topology (a,b,c) and the Artf_ANGER_CNN topology (d,e,f). The line color represents the line length.



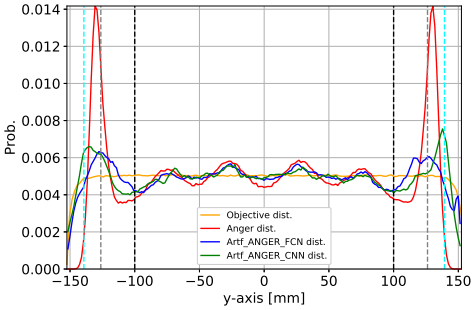
(a) x axis (Experimental)



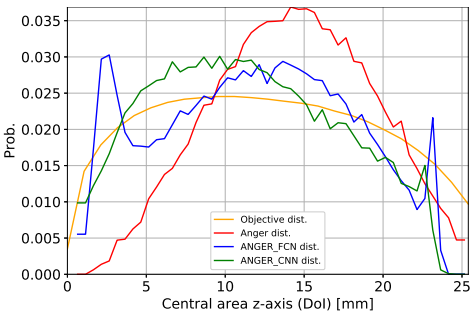
(b) x axis (Simulation)



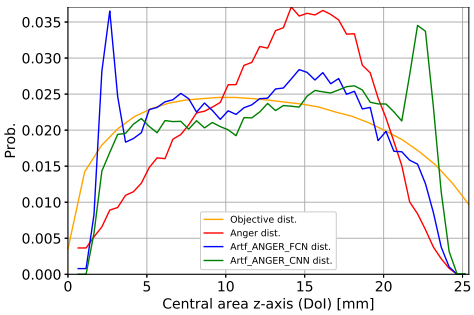
(c) y axis (Experimental)



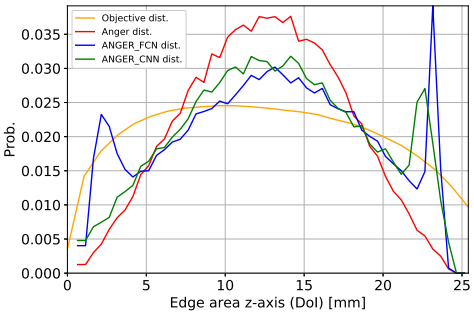
(d) y axis (Simulation)



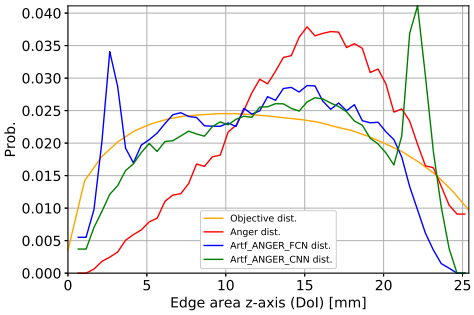
(e) Central area z axis (Experimental)



(f) Central area z axis (Simulation)



(g) Edge area z axis (Experimental)



(h) Edge area z axis (Simulation)

Figure 2.22: Experimental measurements (a,c,e,g) and GATE simulation (b,d,f,h): Empirical density estimation along the x axis (a, b), the y axis (c, d), the z axis on the central (e, f) area of the detector and the z axis in the edge area of the detector (g, h). The vertical black, gray and cyan dashed lines on (a,b) and (c,d) represent a distance of one, half and a quarter PMT form the detectors edge respectively.

Dead zone size	52.4 mm		26.2 mm		13.1 mm	
	x axis	y axis	x axis	y axis	x axis	y axis
Anger	0.17	0.13	0.38	0.58	0.47	0.83
ANGER_FCN	0.17	0.12	0.24	0.24	0.29	0.39
ANGER_CNN	0.18	0.11	0.25	0.21	0.28	0.21
Anger (GATE)	0.34	0.34	0.54	0.67	0.73	0.89
Artf_ANGER_FCN	0.29	0.27	0.31	0.34	0.42	0.35
Artf_ANGER_CNN	0.28	0.26	0.29	0.35	0.53	0.47

Table 2.7: Normalized count difference for each axis empirical density. Each metric was tested using one, half and one quarter PMT (52.4 mm, 26.2 mm, 13.1 mm respectively), representing three different dead zone thresholds of the detector. A lower value means a more uniform density.

2.5.7 Model Quantization

To enable the trained network to work with lower precision or integer-only computation devices, their weights and operations need to be quantized. This step also reduces the network size and makes calculations potentially faster. To test the networks resistance to this operation, both topologies were quantized to signed integer precision with size 16, 12 and 8 bits. The conversion is done using quantization aware retraining [110], implemented in the TensorFlow [111] toolkit. The quantization creates, for each variable, a 8 bit value using the following scheme:

$$\mathtt{f} = (\mathtt{q} - \mathtt{z}) \times \mathtt{s}, \tag{2.31}$$

where f is the floating point value, q is the quantized (integer) value, the zero point z is a quantized variable of the same type as q indicating the quantized value of the real value *zero point* and the scale s is a real valued scale factor. The scale factor, which is a floating point number, can be represented using fixed point multiplications and bit shifting, thus, making it compatible with integer only devices. The number of bits used for the quantized precision of the neural network weights can be arbitrary, however the computation accumulations and bias variable are performed using 32 bits precision. The bias is coded with a larger word size since it is added to many output activations and any quantization error impacts many operations. The values for weights are quantized for each tensor, resulting in a single z and s value per tensor. The values are calculated in order to optimize the dynamic range of the representation (i.e. if a quantization of 8 bits is chosen, then the maximum value of the tensor is set to 127 and the minimum to -127).

The presented topologies were retrained using the quantization aware scheme. The *quantization aware* training scheme uses during training a floating-point arithmetic that mimics the behaviour of the quantization, while allowing the gradients to flow without suffering precision loss. This way the network takes into account the quantization effect during training. During inference (after training), the network uses integer-only arithmetic. To this end a new representative dataset is built for each

network. This dataset consists in 3×10^6 inputs and outputs of the fully trained floating point neural network. The neural network is quantized and retrained using this representative dataset for 10 epochs. The resulting quantized network is then tested for accuracy (mean squared difference to reference neural network) and positioning. A summary of each topology accuracy to each word length can be seen in table 2.8. As expected the error is reduced with the increase of the word length. A word length of 12 `bits` has an error in the same range of a 16 `bits` word, however this value only reflects the re-training error and not the final performance of the NN. The positioning error of the perpendicular beam series, for each word length, is summarized in table 2.9 and table 2.10 for the ANGER_FCN and ANGER_CNN respectively. The GATE simulation networks, Artf_ANGER_FCN and Artf_ANGER_CNN, are shown in table 2.11 and table 2.12 respectively. The variation of the correlation factors for the DoI, in the center and edge areas are shown for each word length and network in table 2.13 and table 2.14 for the experimental dataset and GATE simulation respectively. Finally the normalized count difference in each axis is presented, for each word length in table 2.15.

Method	8 `bits`	12 `bits`	16 `bits`
ANGER_FCN	3.81	1.42	1.51
ANGER_CNN	8.41	5.50	3.70
Artf_ANGER_FCN	6.9	1.31	1.26
Artf_ANGER_CNN	5.27	1.56	1.22

Table 2.8: Mean squared error in mm of each quantized neural network topology to its reference neural network.

Both networks show a reduction of the error with the increase of word length, reaching values near the floating point precision for 16 `bits` words. This behaviour is also reflected in the correlation coefficients for the DoI estimation, presented in table 2.13.

The 2D histograms of the positioning are shown in figure 2.23 for the ANGER_FCN, in figure 2.24 for the ANGER_CNN, in figure 2.25 the Artf_ANGER_FCN and in figure 2.26 for the Artf_ANGER_CNN. These histograms show the quantization effects on the spacial localization of the events. It can be seen that the most extreme positions (the borders of the detectors) are more affected by the quantization. This is reflected in the table 2.15. This is expected due to the quantization procedure which affects more heavily the outliers weights and activations.

2.5.8 Discussion

Here a gamma interaction positioning technique based on NNs was presented. The method requires only a single flood acquisition and the knowledge of the expected interaction distribution in the scintillation volume. The 2D performance of the proposed technique was tested using a series collimated beams, showing an improvement in the mean bias and FWHM values when compared to the COG algorithm. This improvement is seen when the dead zone is reduced to half PMT (26.2 `mm`). An improvement

Beam	ANGER_FCN Estimator	Dead zone size								
		52.4 mm			26.2 mm			13.1 mm		
		$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$
x axis	8 bits	10.57	3.57	3.57	10.30	5.00	6.42	11.19	3.97	6.42
	12 bits	10.22	2.30	2.30	9.65	3.44	4.59	11.30	2.34	4.59
	16 bits	10.36	1.46	1.46	9.50	2.63	3.80	11.04	2.17	3.80
	float32	10.38	2.73	3.80	10.23	3.11	5.37	10.80	2.78	5.37
y axis	8 bits	10.37	1.72	2.48	10.46	1.43	2.48	11.03	1.57	2.48
	12 bits	9.96	1.32	2.46	9.90	1.40	2.46	10.89	1.26	2.46
	16 bits	10.35	2.67	3.56	10.12	2.68	3.56	10.87	2.18	3.56
	float32	10.50	2.10	3.34	10.47	1.98	3.34	11.06	1.73	3.34
Diagonal - x axis	8 bits	10.91	3.96	3.96	10.30	5.02	6.09	12.10	4.95	6.09
	12 bits	9.41	2.67	2.67	9.30	3.66	4.65	12.94	3.76	4.65
	16 bits	9.71	1.44	1.44	9.57	2.66	3.89	13.96	2.57	3.89
	float32	10.06	2.39	2.80	9.88	2.75	3.82	11.62	3.05	4.26
Diagonal - y axis	8 bits	9.21	2.89	2.89	9.45	2.76	2.89	13.94	2.67	2.89
	12 bits	8.38	1.89	1.89	8.73	1.64	1.89	13.34	2.04	2.83
	16 bits	8.82	3.38	3.38	9.08	3.17	3.38	14.66	3.45	4.00
	float32	9.82	1.48	2.27	9.70	1.22	2.27	12.58	1.81	4.15

Table 2.9: Experimental measurement (ANGER_FCN): Quantization accuracy comparison for each word length. The tables presents the mean FWHM ($\overline{\texttt{FWHM}}$), mean bias ($\bar{b}$) and maximum bias ($\max b$), for each beam series and each interaction position estimator. The metrics were tested using detector's dead zone of size one, half and one quarter PMT (52.4 mm, 26.2 mm, 13.1 mm respectively). All metrics are expressed in [mm].

Beam	ANGER_CNN Estimator	Dead zone size: 52.4 mm $\overline{\texttt{FWHM}}$	52.4 mm $\bar{b}$	52.4 mm $\max b$	26.2 mm $\overline{\texttt{FWHM}}$	26.2 mm $\bar{b}$	26.2 mm $\max b$	13.1 mm $\overline{\texttt{FWHM}}$	13.1 mm $\bar{b}$	13.1 mm $\max b$
x axis	8 bits	11.18	10.02	10.02	10.58	10.61	11.21	10.52	10.41	11.21
	12 bits	10.67	5.18	5.18	10.33	6.22	7.26	11.68	5.86	7.26
	16 bits	11.01	4.52	4.52	10.21	5.59	6.67	11.60	4.82	6.67
	float32	10.35	5.07	5.07	9.27	6.02	6.96	10.94	5.51	6.96
y axis	8 bits	10.66	1.06	1.97	9.88	0.92	1.97	11.13	0.76	1.97
	12 bits	10.02	1.73	3.54	10.24	1.59	3.54	11.17	1.39	3.54
	16 bits	11.02	2.48	3.70	11.06	2.50	3.70	11.72	2.35	3.70
	float32	10.40	1.82	3.34	10.52	1.73	3.34	11.36	1.41	3.34
Diagonal - x axis	8 bits	10.72	9.07	9.07	9.35	10.44	11.80	9.35	10.61	11.80
	12 bits	9.33	4.68	4.68	9.00	6.12	7.56	10.05	6.76	8.05
	16 bits	10.02	3.64	3.64	9.37	5.32	6.99	11.06	5.68	6.99
	float32	9.84	4.51	4.51	9.04	5.67	6.83	10.40	6.69	8.73
Diagonal - y axis	8 bits	9.34	1.01	1.01	8.53	0.96	1.01	10.08	2.54	5.71
	12 bits	9.93	0.38	0.38	8.70	0.19	0.38	9.91	1.95	5.49
	16 bits	10.58	2.52	2.52	9.14	2.84	3.16	9.10	4.59	8.08
	float32	10.33	1.24	1.24	8.88	1.69	2.15	9.64	3.82	8.08

Table 2.10: Experimental measurement (ANGER_CNN): Quantization accuracy comparison for each word length. The tables presents the mean FWHM ($\overline{\texttt{FWHM}}$), mean bias ($\bar{b}$) and maximum bias ($\max b$), for each beam series and each interaction position estimator. The metrics were tested using detector's dead zone of size one, half and one quarter PMT (52.4 mm, 26.2 mm, 13.1 mm respectively). All metrics are expressed in [mm].

Beam	Artf_ANGER_FCN Estimator	Dead zone size								
		52.4 mm			26.2 mm			13.1 mm		
		$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$
x axis	8 bits	7.69	1.38	3.35	10.63	2.42	3.88	16.98	1.44	1.73
	12 bits	8.52	1.64	2.58	11.78	2.32	3.56	17.64	1.81	2.34
	16 bits	9.14	1.56	2.50	12.20	2.56	3.83	18.39	2.05	2.74
	float32	5.46	0.89	3.17	6.72	2.54	5.96	8.25	2.84	5.96
y axis	8 bits	6.78	2.17	4.16	10.34	6.73	8.24	19.50	8.18	8.89
	12 bits	6.28	1.19	1.79	10.27	2.43	4.16	19.32	4.37	4.99
	16 bits	5.59	8.37	8.40	9.44	1.40	1.89	20.37	2.05	2.92
	float32	5.49	1.25	2.78	6.50	2.05	4.18	9.15	2.87	8.11
Diagonal - x axis	8 bits	8.66	14.74	14.95	11.49	2.03	3.27	20.02	1.94	5.24
	12 bits	8.29	1.60	2.76	11.56	4.44	5.67	18.86	3.26	4.77
	16 bits	8.66	1.42	2.59	12.33	4.58	5.67	20.05	2.68	3.85
	float32	5.16	1.05	3.64	6.25	3.42	7.84	8.63	4.04	7.84
Diagonal - y axis	8 bits	7.14	1.47	2.95	10.97	3.94	7.07	19.95	5.09	6.65
	12 bits	6.30	0.76	1.36	10.81	4.25	7.89	20.31	6.58	7.89
	16 bits	6.05	8.32	8.40	9.76	2.57	4.43	21.08	3.61	4.68
	float32	5.20	0.92	1.55	6.69	2.49	7.73	9.26	3.62	8.68

Table 2.11: GATE simulation (Artf_ANGER_FCN): Quantization accuracy comparison for each word length. The tables presents the mean FWHM ($\overline{\texttt{FWHM}}$), mean bias ($\bar{b}$) and maximum bias ($\max b$), for each beam series and each interaction position estimator. The metrics were tested using detector's dead zone of size one, half and one quarter PMT (52.4 mm, 26.2 mm, 13.1 mm respectively). All metrics are expressed in [mm].

Beam	Artf_ANGER_CNN Estimator	Dead zone size								
		52.4 mm			26.2 mm			13.1 mm		
		$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$
x axis	8 bits	5.00	12.75	14.95	5.91	0.92	2.77	10.16	3.83	4.80
	12 bits	5.84	1.16	1.82	7.21	2.79	3.94	11.03	5.47	6.58
	16 bits	5.76	1.30	2.50	7.59	3.57	5.11	10.46	6.71	7.83
	float32	5.90	1.19	2.92	7.85	4.19	5.39	11.31	6.84	7.74
y axis	8 bits	5.70	1.40	2.25	7.55	0.73	1.23	12.51	2.20	2.89
	12 bits	5.79	0.85	2.64	8.18	3.66	4.51	13.33	6.43	7.31
	16 bits	5.53	0.94	1.38	7.93	2.37	3.23	13.13	5.65	6.64
	float32	5.78	0.88	1.42	8.49	2.61	3.41	13.61	6.01	6.94
Diagonal - x axis	8 bits	5.42	12.75	14.95	7.60	5.54	9.90	7.38	6.04	9.32
	12 bits	5.97	1.26	1.81	7.16	6.69	10.01	8.15	6.90	9.84
	16 bits	6.21	1.24	3.04	7.42	7.48	10.48	8.24	7.50	10.29
	float32	6.00	1.13	2.12	6.96	7.14	9.98	8.08	6.94	9.87
Diagonal - y axis	8 bits	4.72	9.34	9.40	6.71	4.64	8.97	7.40	7.95	9.42
	12 bits	5.24	0.98	2.50	6.40	6.84	10.19	8.42	9.28	10.26
	16 bits	5.50	1.07	2.21	6.28	6.88	10.35	7.96	9.60	10.60
	float32	5.20	0.96	1.85	6.58	6.26	9.80	8.12	8.79	9.91

Table 2.12: GATE simulation (Artf_ANGER_CNN): Quantization accuracy comparison for each word length. The tables presents the mean FWHM ($\overline{\texttt{FWHM}}$), mean bias ($\bar{b}$) and maximum bias ($\max b$), for each beam series and each interaction position estimator. The metrics were tested using detector's dead zone of size one, half and one quarter PMT (52.4 mm, 26.2 mm, 13.1 mm respectively). All metrics are expressed in [mm].

Estimator	Word length	Center Area	Edge Area
ANGER_FCN	8 bits	0.18	–
	12 bits	0.20	–
	16 bits	0.21	–
	float32	0.21	–
ANGER_CNN	8 bits	0.22	–
	12 bits	0.22	–
	16 bits	0.20	–
	float32	0.23	–

Table 2.13: Experimental measurement: Quantization accuracy comparison for each word length. The table shows the correlation factor between the expected DoI and the estimated DoI in the central and edge areas of the detector's surface.

Estimator	Word length	Center Area	Edge Area
Artf_ANGER_FCN	8 bits	0.74	0.44
	12 bits	0.72	0.43
	16 bits	0.70	0.42
	float32	0.80	0.53
Artf_ANGER_CNN	8 bits	0.82	0.55
	12 bits	0.82	0.56
	16 bits	0.83	0.56
	float32	0.83	0.56

Table 2.14: GATE simulation: Quantization accuracy comparison for each word length. The table shows the correlation factor between the expected DoI and the estimated DoI in the central and edge areas of the detector's surface.

Estimator	Word length	Dead zone size					
		52.4 mm		26.2 mm		13.1 mm	
		x axis	y axis	x axis	y axis	x axis	y axis
ANGER_FCN	8 bits	0.22	0.20	0.25	0.29	0.30	0.33
	12 bits	0.17	0.12	0.25	0.26	0.32	0.37
	16 bits	0.17	0.12	0.29	0.27	0.31	0.39
	float32	0.17	0.12	0.24	0.24	0.29	0.39
ANGER_CNN	8 bits	0.24	0.23	0.28	0.27	0.30	0.32
	12 bits	0.23	0.17	0.26	0.24	0.35	0.27
	16 bits	0.19	0.11	0.25	0.26	0.35	0.30
	float32	0.18	0.11	0.25	0.21	0.28	0.21
Artf_ANGER_FCN	8 bits	0.31	0.30	0.38	0.39	0.38	0.49
	12 bits	0.27	0.28	0.38	0.38	0.39	0.45
	16 bits	0.28	0.29	0.39	0.41	0.40	0.54
	float32	0.29	0.27	0.31	0.34	0.42	0.35
Artf_ANGER_CNN	8 bits	0.33	0.33	0.38	0.39	0.59	0.55
	12 bits	0.36	0.29	0.33	0.35	0.55	0.48
	16 bits	0.32	0.27	0.33	0.35	0.54	0.45
	float32	0.28	0.26	0.29	0.35	0.53	0.47

Table 2.15: Normalized count difference for each axis empirical density. Quantization accuracy comparison for each word length. Each metric was tested using one, half and one quarter PMT (52.4 mm, 26.2 mm, 13.1 mm respectively), representing three different dead zone thresholds of the detector. A lower value means a more uniform density.

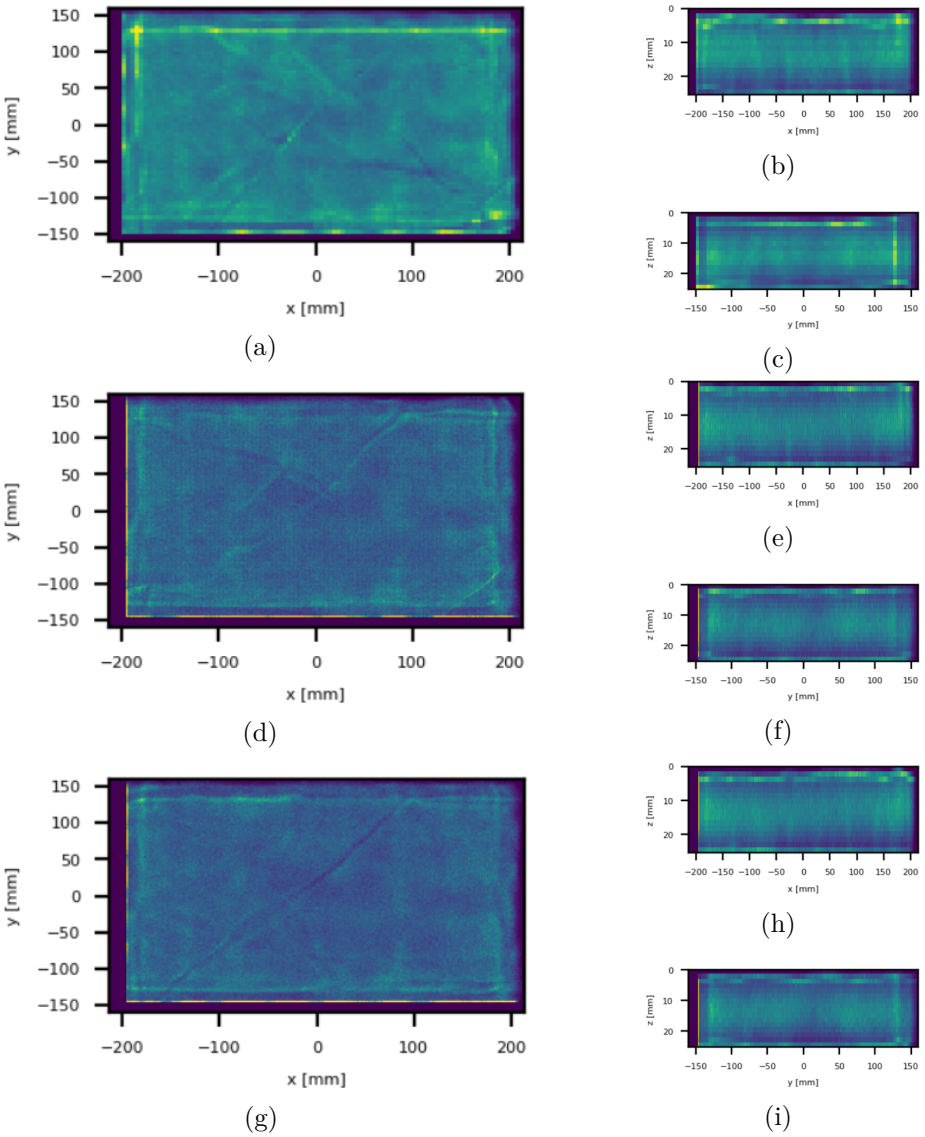


Figure 2.23: Experimental measurement: Histograms of events in (x, y) plane (a,d,g), (x, z) plane (b,e,h) and (y, z) plane (c,f,i) of the 8 `bit` (a,b,c), the 12 `bit` (d,e,f) and the 16 `bit` quantization (g,h,i) of the fully connected topology.

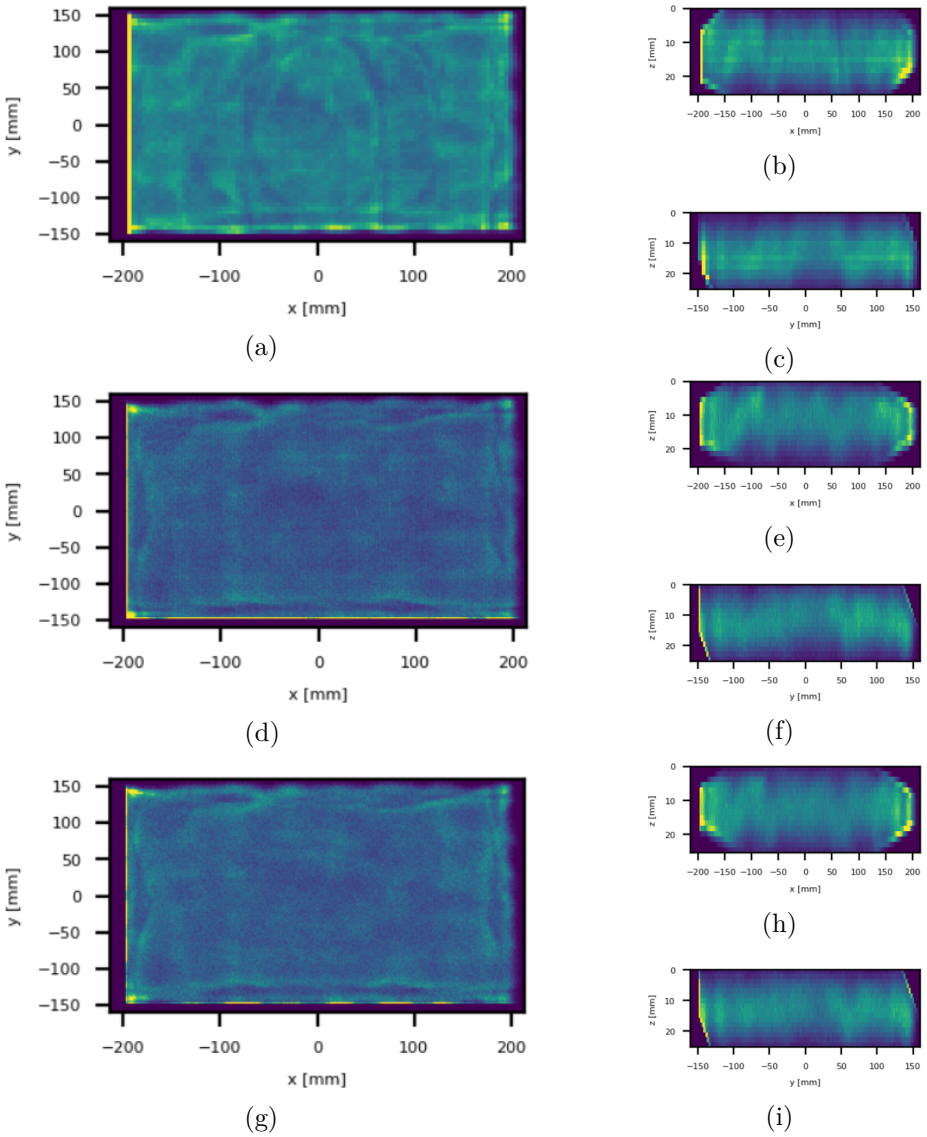


Figure 2.24: Experimental measurement: Histograms of events in (x, y) plane (a,d,g), (x, z) plane (b,e,h) and (y, z) plane (c,f,i) of the 8 `bit` (a,b,c), the 12 `bit` (d,e,f) and the 16 `bit` quantization (g,h,i) of the convolutional topology.

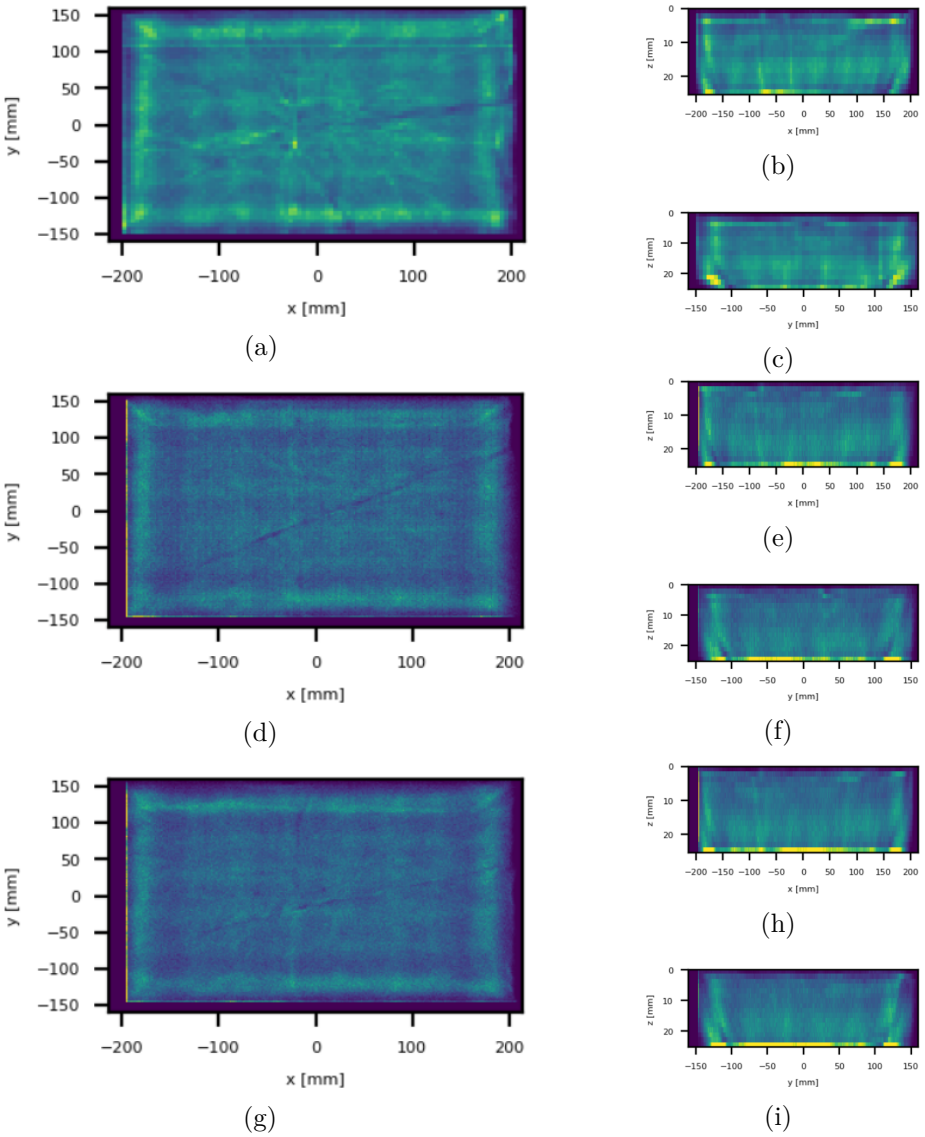


Figure 2.25: GATE simulation: Histograms of events in (x, y) plane (a,d,g), (x, z) plane (b,e,h) and (y, z) plane (c,f,i) of the 8 `bit` (a,b,c), the 12 `bit` (d,e,f) and the 16 `bit` quantization (g,h,i) of the fully connected topology.

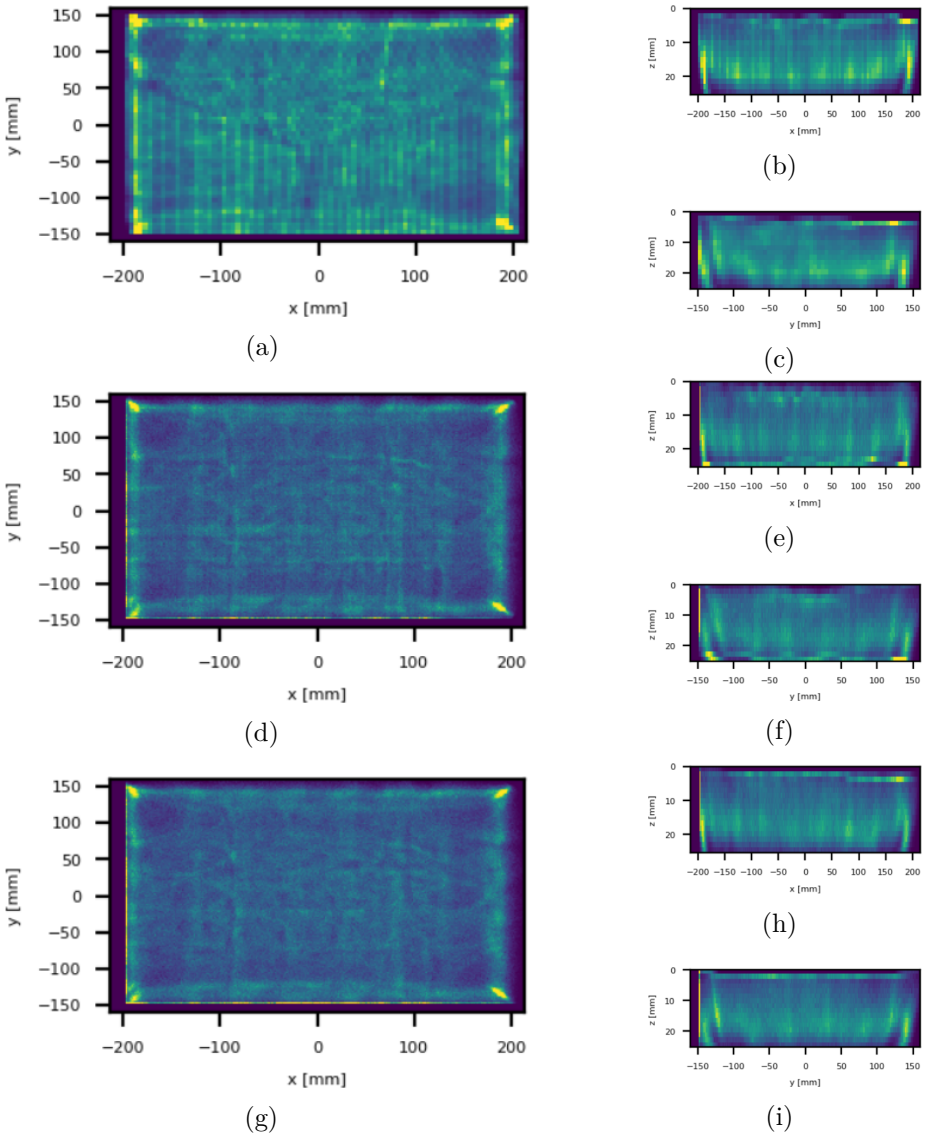
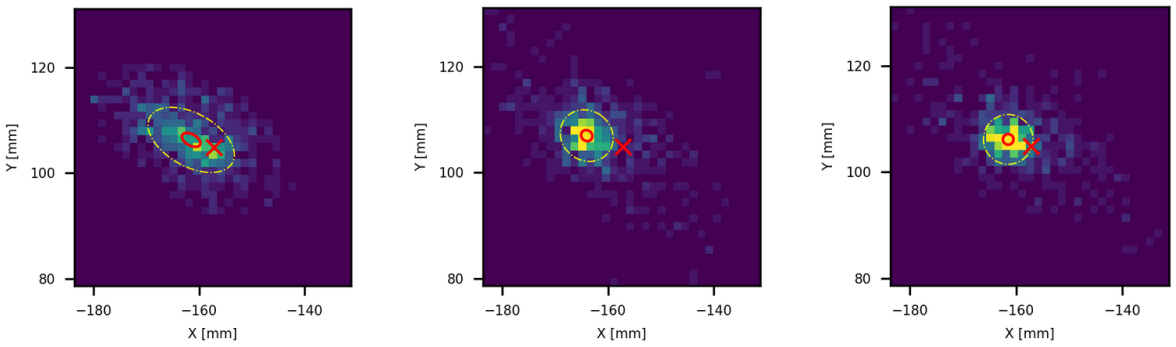


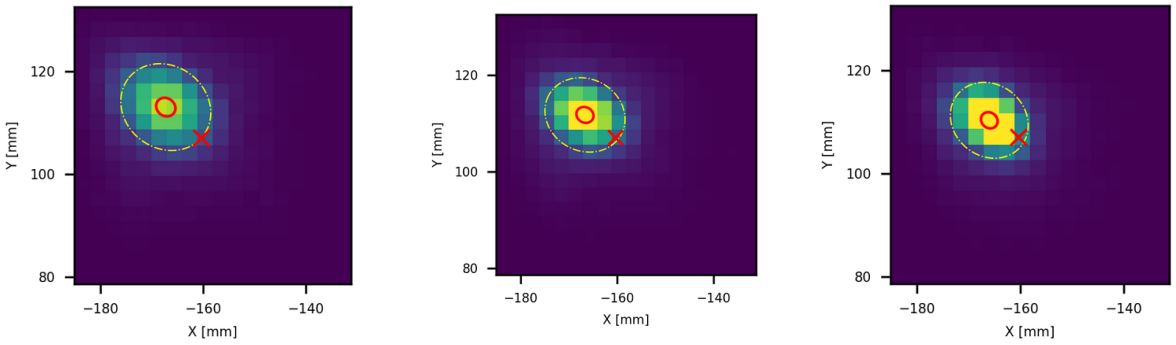
Figure 2.26: GATE simulation: Histograms of events in (x, y) plane (a,d,g), (x, z) plane (b,e,h) and (y, z) plane (c,f,i) of the 8 `bit` (a,b,c), the 12 `bit` (d,e,f) and the 16 `bit` quantization (g,h,i) of the convolutional topology.

of the FWHM is observed in the collimated beam series on the x axis and diagonal, presented in Table 2.3 and Table 2.4. The FWHM recovery can be seen on Fig 2.27 and Fig 2.28, where the 2D histograms of the interaction positions for a collimated beam near the corner of the detector surface are shown (for the empirical measurement and GATE simulation respectively).



(a) COG (b) COG_NN (c) COG_NN

Figure 2.27: Experimental measurement: interaction histogram on the (x, y) plane obtained using the Anger method (a), the ANGER_FCN (b) and the ANGER_CNN (c). The red cross represents the expected position, the red ellipse marks the estimated position and the yellow ellipse marks the FWHM of the estimated position.



(a) COG (b) COG_NN (c) Artf_COG_NN

Figure 2.28: GATE simulation: interaction histogram on the (x, y) plane obtained using the Anger method (a), the Artf_ANGER_FCN (b) and the Artf_ANGER_CNN (c). The red cross represents the expected position, the red ellipse marks the estimated position and the yellow ellipse marks the FWHM of the estimated position.

The DoI capability of the proposed technique was tested using a beam acquisition with an angle of 45°. The NNs achieved a correlation factor of $\texttt{Corr}^{\texttt{ANGER\_FCN}}_{\texttt{center}} = 0.21$ and $\texttt{Corr}^{\texttt{ANGER\_CNN}}_{\texttt{center}} = 0.23$ against a $\texttt{Corr}^{\texttt{Feature}}_{\texttt{center}} = 0.28$ of the baseline method. In the

case of the GATE simulation the NNs method achieved a $\texttt{Corr}_{\texttt{center}}^{\texttt{Artf\_ANGER\_FCN}} = 0.80$ and $\texttt{Corr}_{\texttt{center}}^{\texttt{Artf\_ANGER\_CNN}} = 0.83$ against a $\texttt{Corr}_{\texttt{center}}^{\texttt{Feature GATE}} = 0.79$ of the base method. A direct enhancement of the DoI accuracy was not observed however the relatively small size of the network was able to reproduce the baseline DoI positioning method. This is important since the base method required sorting and LUTs to produce a DoI estimation while the NN is composed only of multiplications and summations. Implementing better algorithms to detect the DoI will enable the NNs to reduce the FWHM observed in the experimental measurement (figure 2.16), as the network capacity is enough to produce better DoI estimation (as observed in Fig 2.17).

The Packing effect of the Anger algorithm was highly reduced. As observed in the histograms presented in figure 2.18 and figure 2.19 the surface present less boundary effects when compared to their references in Fig 2.8(a) and Fig 2.8(b) respectively. The proposed method relocated the "*packed*" interactions to their closest location, following an OT scheme. As shown in figure 2.20 and figure 2.21 the relocations are more severe in those regions where the COG method fails. Also the relocation map shows the zones between the PMT units where the event location is also corrected. This correction is also seen in the empirical density estimation on each axis, presented in figure 2.22. It can be seen that the density estimation obtained with the proposed method is closer to the expected distribution of the events. A count difference metric was calculated to reflect this effect and is presented in Table 2.7. The table shows the sharp reduction in the count variability in the flood acquisition. The performance improvement and the reduction of the count difference enabled the reduction of the detector's *dead zone* from 52.4 $\texttt{mm}$ to 26.2 $\texttt{mm}$, increasing the effective area of the detectors surface from 48.7% to 72.1% of the scintillation crystal surface if a maximum ΔC of 25% is tolerated. This results in an increase of 50% of single event detection efficiency and a $0.721^2/0.487^2 = 2.19$ times increase in the detection probability of a coincidence in PET. The proposed method was tested using a simple interaction positioning method such as the Anger as base method, nevertheless it can be used to enhance any other base method. Improving the performance of the base method will result in an enhanced positioning quality of the NN, provided that the NN has enough capacity. The presented method requires a simple acquisition to be implemented, in contrast to other methods that rely on carefully acquired dataset to train their NNs. Moreover the method can be used to correct or adapt model based methods, such as the ones presented in [37] and [102]. This adaptation is required when the physical model does not perfectly match the implementation or the detector construction process is not error free. The NN training scheme can be improved by including smoothness constrains in the objective function. This way the effects of low statistics in the train dataset (low $\texttt{imapacts/mm}^3$) can be alleviated. The size of the NN is also a crucial factor in the final performance of the interaction localization. However the implementation of NN based methods in PET detectors requires low processing times. While both topologies were able to improve the base method performance, the fully connected neural network showed to be more robust to quantization and requires less operations per event (see table 2.2). Although synthetising a simple NN in field-programmable gate arrays is possible, we expect to further investigate the effects of network size, topology (fully connected or convolutional) and quantization. This way it will be possible to determine a correct

trade-off between NN performance and processing bottleneck. Nevertheless the size of the NNs, selected in this study, was small and make them suitable for embedded operation. The effect of quantization and full-integer execution was tested and suffered only small loses for word sizes of 12 and 16 `bits`. As expected, the quantization errors are more severe in the borders of the detection surface, nevertheless this area falls into the blind zone. The quantization error is observed to be lower for the fully connected networks, which makes them a better fit for implementation in the detectors FPGAs.

2.6 Discussion and Remarks

In this chapter the AR-PET prototype was presented and its technical capabilities were described. The AR-PET prototype requirements inspired the methods developed in this chapter. First, using genetic algorithms, an adaptive PMT selection method was developed. This method has shown to be better than hand-picking PMTs for the camera construction. Moreover the method proved to be fitted to find viable camera configurations when the number of cameras to build is more than one and the number of available PMT units is heavily constrained. The second presented work focused on the localization of the gamma photon interactions within the scintillation crystal. This task plays a central role in the final accuracy of the scanner. Since the AR-PET cameras use large, thick and solid scintillation crystals with large PMTs, the localization task is not trivial and new positioning schemes had to be studied. The proposed method took advantage of the signal processing capabilities of the AR-PET cameras, to develop non-linear positioning algorithms. These algorithms are based on neural networks and are able to be fully embedded in the cameras FPGA units. Since the AR-PET cameras do not provide any other information than the values of the PMT units arranged on the detectors crystal surface, a novel way to train these networks was devised. This training procedure uses only information from a simple flood acquisition to obtain all the necessary training data and then tries to match the event distribution along the crystal volume to the expected distribution. The method was tested on the AR-PET cameras, transforming the Anger distribution into an expected distribution, improving the sensitivity area and fitting the camera with DoI capability at low computational cost. However the method is also valid for matching any other initial positioning method in any other camera model, under the conditions of known PMT/SiPM positions and energies, known objective distribution and a known initial positioning method.
The following chapters focus on PET image methods. They implement reconstruction algorithms and image methods to enable PET imaging without attenuation scans, taking a more direct approach to the objective of reducing the PET dose study.

3 Mesh representation on tomogaphic images

In this chapter the problem of PET image reconstruction by means of Finite Element (FE) mesh representation is studied. First the concept of FE and the basic notions of mesh construction are introduced. Then the focus is put on the existing PET and Single Photon Emission Computed Tomography (SPECT) image reconstruction methods using mesh and present our contributions. Finally the basics of Single Scatter Simulation (SSS) using mesh representation and their advantages is layed out.

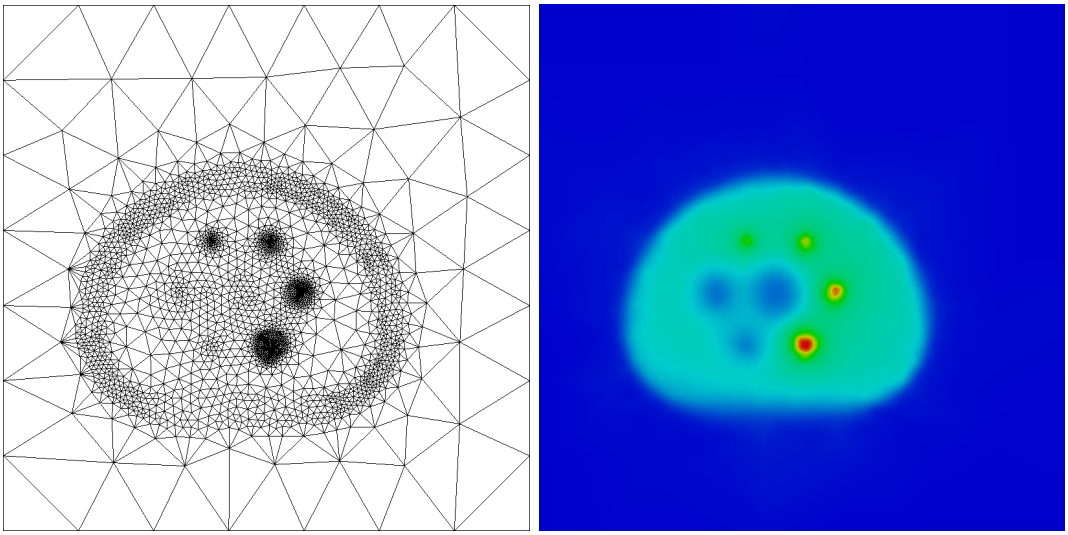
3.1 Introduction

The concept of using meshes for FE calculation roots in the discretization of a bounded domain into simple elements. It seeks to perform a piecewise approximation using simple polynomials of a complex domain. These polynomials are only defined in small regions (elements) of the domain, where their approximation is valid, and they are expressed in terms of the values at the nodes (or vertices) of the elements. The set of these elements defines the mesh. In opposition to the pixelized or voxelized representations, where all elements have the same size and are ordered by their position, a mesh does not have such constraints. A mesh is defined by two main features:

- Geometry: The positions of the nodes (or vertices) of the mesh elements in the domain space.
- Topology: The connection list of the vertex labels, which describe each element.

With such characteristics (geometry and topology), the sampling of an image can be adapted to describe its support more efficiently. In figure 3.1 the mesh representation of a National Electrical Manufacturers Association (NEMA) image quality phantom [112] is presented. It can be seen that the mesh is optimized to assign larger elements to sections of the image with almost linear activity variations and smaller elements to those sections with non-linear variations. This way the mesh representation also embeds the activity as part of the geometry information.

The idea of using mesh support for medical image representation is not new. It has been widely studied for hemodynamics [113] simulation, respiratory systems [114] simulation and medical image registration [115, 116]. In the context of image reconstruction the mesh methods have not been widely adopted, nevertheless applications



(a) (b)

Figure 3.1: Mesh representation (a) of NEMA phantom and the corresponding reconstructed activity (b).

for PET and SPEC exist. This subject will be further developed in section 3.3.1.

The mesh construction and analysis presented in this chapter correspond to 2D meshes which adapt successfully to the PET image reconstruction task (see section 3.3). Interested readers can take a look at [117] for 3D implementations of the FEM construction methods presented in the following section.

3.2 Mesh Construction

A valid mesh for FE calculation covers all points of the domain with a non-overlapping collection of elements and also meets the boundary conditions of the domain. The boundaries of a mesh can be adapted to a specific region and do not need to be rectangular as in pixel representation. The size of the elements, their nodal spacing and shape must be controlled and they define the *quality* of a mesh. A high quality mesh reduces the discretization or numerical error of the FE calculation. To optimize computation, a FE mesh must have a specific element shape with consistent numbering and orientation. These conditions make the construction of a FE mesh with high quality more difficult than a regular mesh, requiring optimization steps to enhance quality and reduce errors[1]. One way to obtain a high quality mesh is to adapt it to reduce the discretization error. For example, by assigning smaller element size to areas with large interpolation error and large elements to areas with low interpolation error. The adaptive methods use such criteria to refine a given mesh and enhance its quality.

[1]Throughout the chapter the term "mesh" will be used to refer to "FE mesh", to simplify notation.

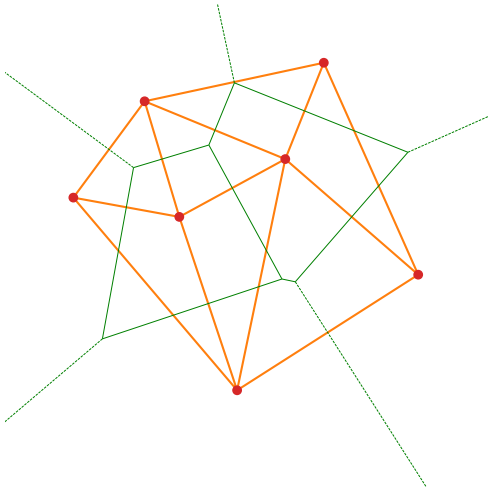


Figure 3.2: Voronoi diagram and corresponding Delaunay triangulation of a set of points. The computed points are shown in red, the thin green lines correspond to the faces of the Voronoi polygons and the orange lines correspond to the faces of the Delaunay triangulation elements.

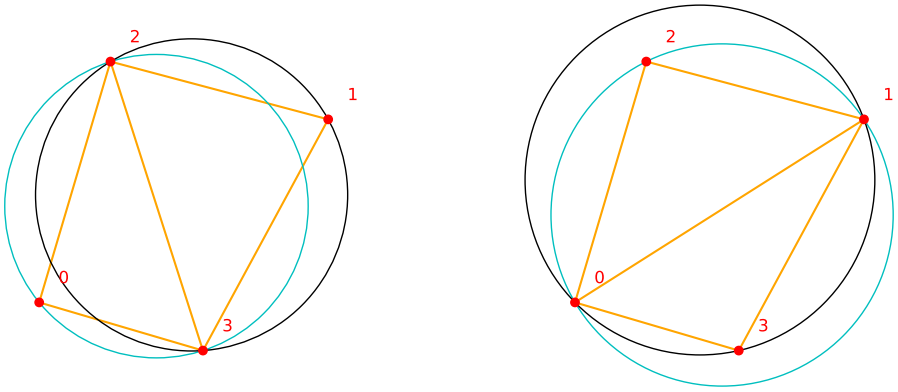
The most common procedures to build a mesh are the Delaunay insertion kernel [118, 119] and the the advancing front [120]. The first method works by incremental triangulation of a set of vertices which are successively added to the domain, starting from the boundaries and adding points one by one into the meshed domain. The second method works by adding vertices in the smaller segments of the boundary to create near equilateral elements and then update the boundary until no point can be added. It is possible to achieve combination of both strategies [121], such strategy is more efficient and robust than previous methods and is presented in section 3.2.2. The internal node generation of these methods can be achieved by using several methods and criteria, nevertheless the use of a control space to measure the quality of the elements to be generated is a simple and robust approach [122]. This technique is developed in section 3.2.3.

3.2.1 Delaunay Insertion Kernel

The Delaunay triangulation was developed as the dual of the Voronoi diagram, which partitions a domain into a set of polygons. These *Voronoi* polygons represent the region of the domain where the closest point is a central node. The Voronoi polygons are constructed by joining the Dirichlet tessellation around the central node. Then, the Delaunay triangulation is constructed by linking the central nodes of these polygons into a set of triangles. In figure 3.2 the Voronoi diagram and the Delaunay triangulation of a set points is represented.

The triangulation obtained using this method ensures the following properties [123]:

- The generated triangles are not obtuse.



(a) (b)

Figure 3.3: Delaunay triangulation, where no node is inside of the circumcircle of an other element (a) and non-Delaunay triangulation (b), where the circumcircle of element (0,1,2), in cyan, contains the external node 3 and the circumcircle of element (0,3,1), in black, contains the external node 2.

- Maximizes the minimum internal angle.
- Minimizes the circumcircle of each element.
- The union of the circumcircles of different elements is the smallest.
- No other vertex lies in the circumcircle of a given element (also known as the empty-circle property or Delaunay criterion).
- The weighted sum of squares of the edge lengths is the smallest, creating compact triangulations.

In figure 3.3 an example of valid element (figure 3.3(a)), and a non-Delaunay element (figure 3.3(b)) are shown.

The Delaunay insertion Kernel (DK) constructs a mesh by gradually inserting a node into the meshed domain and remeshing those elements which violate the Delaunay criterion. Starting from a simple mesh with four nodes (corresponding to the boundaries of the domain) a simple triangulation is constructed and then each node in the domain is added by means of the DK. This way the resulting mesh respects the Delaunay criterion and satisfy all these properties.

The steps to construct (or fill) a given mesh using a robust implementation of the DK are illustrated in figure 3.4 and are following following [124]:

step 1

Given a domain $\boldsymbol{\Omega}$ with a mesh representation $\mathcal{F}(\boldsymbol{\Omega})$, a new point p is inserted (figure 3.4(a)). The element containing the point $\boldsymbol{r}$ is identified as the *base* element, $\texttt{Base}(\boldsymbol{r})$.

step 2

The k elements whose circumcircle contains the point $\boldsymbol{r}$ are identified as the *cavity* set, $\texttt{Cavity}(\boldsymbol{r}) = \{k \,|\, \boldsymbol{r} \in \texttt{Disk}(k)\}$, where $\texttt{Disk}(k)$ is the circumcircle of the k^{th} element.

step 3

The constructed $\texttt{Cavity}(\boldsymbol{r})$ needs to be star-shaped (connected), which is not the case in figure 3.4(b) where the cavity elements are coloured. It can be seen that the cavity is disconnected due to numerical errors in the calculation of $\texttt{Disk}(1)$. To detect such ill-shaped cavities the created elements are tested. For each element in the $\texttt{Cavity}(\boldsymbol{r})$ the borders F_j^k are extracted and new elements are constructed using the inserted point $K_j^* = (F_j^k, \boldsymbol{r})$. A visibility check is performed on each of these elements, composed of points $(\boldsymbol{r}^a, \boldsymbol{r}^b, \boldsymbol{r}^c)$, by means of the determinant operation:

$$\texttt{Det}(K) = \begin{pmatrix} r_x^b - r_x^a & r_x^c - r_x^a \\ r_y^b - r_y^a & r_y^c - r_y^a \end{pmatrix}. \tag{3.1}$$

If any $\texttt{Det}(K_j^*) \leq 0$ the element k is removed from $\texttt{Cavity}(\boldsymbol{r})$. Applying this check on the element $(\boldsymbol{r}^a, \boldsymbol{r}^b, \boldsymbol{r}^c)$ of figure 3.4(b) will fail (due to the oriented notation of the elements in the mesh), and the resulting corrected cavity is shown in figure 3.4(c).

step 4

With the corrected $\texttt{Cavity}(\boldsymbol{r})$ the $\texttt{Ball}(\boldsymbol{r})$ is created. The *ball* is the set of new elements, constructed using the borders F_j^k of the elements in $\texttt{Cavity}(\boldsymbol{r})$ and the new node $\boldsymbol{r}$. In figure 3.4(d) the $\texttt{Ball}(\boldsymbol{r})$ is shown, inserted in the new mesh $\mathcal{F}^*(\boldsymbol{\Omega})$. The new mesh is constructed as:

$$\mathcal{F}^*(\boldsymbol{\Omega}) = (\mathcal{F}(\boldsymbol{\Omega}) - \texttt{Cavity}(\boldsymbol{r})) \cup \texttt{Ball}(\boldsymbol{r}). \tag{3.2}$$

3.2.2 Delaunay–advancing-front 2D mesh construction

The Delaunay–Advancing-Front (DAF) procedure consists in the meshing of a domain using variable element size. In the DAF each node is generated to create equilateral elements around the boundary. After the creation of a new element the boundary is updated to include the new element and the process is repeated. The method is defined as:

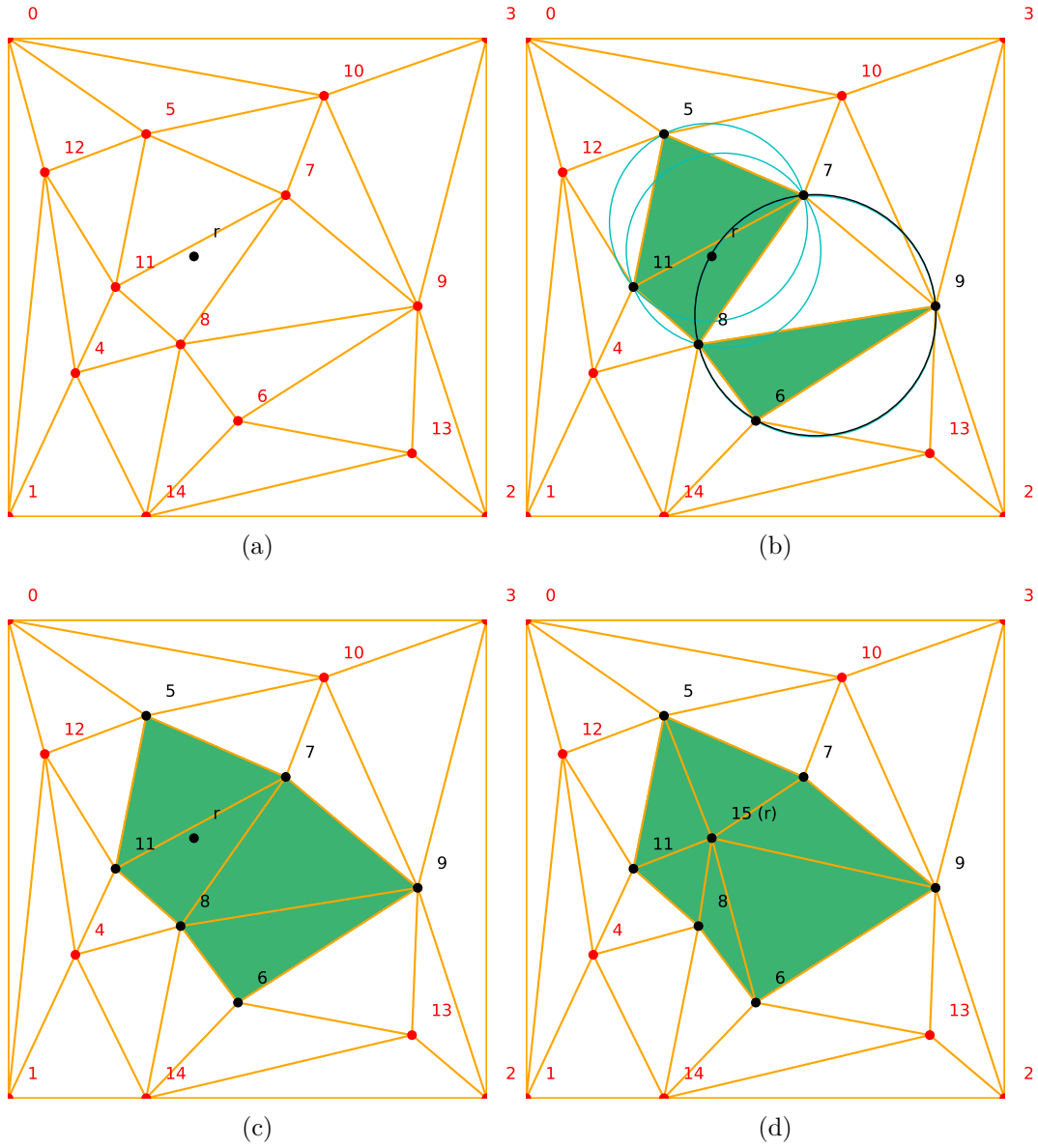


Figure 3.4: Delaunay insertion kernel process. In (a) the initial mesh with the new node $\boldsymbol{r}$. In (b) the elements of the initial cavity with numerical error is shown in light green, due to the near cyclic nodes (7,8,6,9). The circumcircles of the selected elements are shown in cyan and the circumcircle of the excluded element is shown in black. In (c) the corrected cavity is shown. Finally in (d) the *Ball* is shown.

Step 0

The DAF method starts by defining a *segment front* Γ containing all the segments $\mathcal{L}$ of the boundary $\mathcal{B}$ of the domain $\mathbf{\Omega}$. These segments $\mathcal{L}_i$ are defined *counter-clockwise*, and in case there is an inner boundary, those segments will be defined in *clockwise* manner. This way the meshing front is always located on the left of the front. The creation of Γ starts with the definition of four corners, from which a square domain is represented. Then the sides are subdivided into segments $\mathcal{L}_i$ following a nodal spacing or metric function $h()$. The nodal spacing function provides a specified segment length for each node or point in space. Based on the nodal spacing function value at each pair of corners, several segments $\mathcal{L}_i$ are created using a geometrical series. Assuming that the initial and final nodes of a segment $\mathcal{L}_i = (\boldsymbol{r}_i^s, \boldsymbol{r}_i^e)$ have the corresponding metrics $h(\boldsymbol{r}_i^s)$ and $h(\boldsymbol{r}_i^e)$, a sequence can be defined such that:

$$\mathcal{L}_i^{n+1} = \mathcal{L}_i^n \, s_r, \tag{3.3}$$

with

$$\mathcal{L}_i^0 = s_r \, h(\boldsymbol{r}_i^s) \text{ and } \mathcal{L}_i^{N_s} = s_r^{-1} \, h(\boldsymbol{r}_i^e), \tag{3.4}$$

respecting

$$\mathcal{L}_i = \sum_{n=1}^{N_s} \mathcal{L}_i^n, \tag{3.5}$$

$$h(\boldsymbol{r}_i^e) = s_r^{N_s+2} h(\boldsymbol{r}_i^s), \tag{3.6}$$

where s_r is the sought ratio and N_s the total number of divisions required. Resulting in:

$$s_r = \frac{\mathcal{L}_i + h(\boldsymbol{r}_i^e)}{\mathcal{L}_i + h(\boldsymbol{r}_i^s)}, \tag{3.7}$$

and

$$N_s = \frac{\ln(h(\boldsymbol{r}_i^e)/h(\boldsymbol{r}_i^s))}{\ln(s_r)} - 2. \tag{3.8}$$

However N_s must be an integer number, so the nearest integer is chosen $N_s = \lfloor N_s \rceil$ and s_r is re-calculated:

$$s_r = \left(\frac{h(\boldsymbol{r}_i^s)}{h(\boldsymbol{r}_i^e)} \right)^{1/(N_s+2)}. \tag{3.9}$$

The resulting boundary $\mathcal{B}$ is triangulated using Delaunay and illustrated for a square domain, in figure 3.5(a).

Step 1

From the list of segments composing Γ the shortest is selected. The selected segment is called *base* segment $\mathcal{L}_b = (\boldsymbol{r}_b^s, \boldsymbol{r}_b^e)$. Using the metric space, a new node $\boldsymbol{r}$ is created so that the new element k^* is equilateral in the metric space [125]. Such equilarity is referred to the unit length criteria, in which the length of a segment $\mathcal{L}_i = (\boldsymbol{r}_i^s, \boldsymbol{r}_i^e)$ is normalized by the expected length in the metric space:

$$\mathcal{L}_i^{\mathcal{N}} = \|\mathcal{L}_i\| \int_{\boldsymbol{r}_i^s}^{\boldsymbol{r}_i^e} \frac{1}{h(t)} dt, \tag{3.10}$$

where $\|\mathcal{L}_i\|$ is the euclidean length of the segment. Then the position of $\boldsymbol{r}$ is determined by the metrics at their extrema $h(\boldsymbol{r}_b^s)$ and $h(\boldsymbol{r}_b^e)$. Two points are determined such that the formed elements $K_0 = (\boldsymbol{r}_b^s, \boldsymbol{r}_b^e, \boldsymbol{r}_0)$ and $K_1 = (\boldsymbol{r}_b^s, \boldsymbol{r}_b^e, \boldsymbol{r}_1)$ are equilateral and of unit length. Then the point $\boldsymbol{r}$ is computed as $\boldsymbol{r} = (\boldsymbol{r}_0 + \boldsymbol{r}_1)/2$. Finally the position of $\boldsymbol{r}$ is iteratively adjusted by:

- Calculate $\boldsymbol{r}_0 = \boldsymbol{r}_b^s + (\boldsymbol{r}_0/\mathcal{L}_0^{\mathcal{N}})$ and $\boldsymbol{r}_1 = \boldsymbol{r}_b^e + (\boldsymbol{r}_1/\mathcal{L}_1^{\mathcal{N}})$, with $\mathcal{L}_0^{\mathcal{N}} = (\boldsymbol{r}_b^s, \boldsymbol{r}_0)$ and $\mathcal{L}_1^{\mathcal{N}} = (\boldsymbol{r}_1, \boldsymbol{r}_b^e)$.
- Set $\boldsymbol{r} = (\boldsymbol{r}_0 + \boldsymbol{r}_1)/2$.

Step 2

Find other point $\boldsymbol{r}^*$ from the nodes in Ω which creates the most equilateral element. Compare the geometries of $k_0 = (\boldsymbol{r}_b^s, \boldsymbol{r}_b^e, \boldsymbol{r})$ and $k_1 = (\boldsymbol{r}_b^s, \boldsymbol{r}_b^e, \boldsymbol{r}^*)$ using the equilarity measure:

$$\gamma(\boldsymbol{r}^a, \boldsymbol{r}^b, \boldsymbol{r}^c) = 2\sqrt{3} \frac{\mathcal{L}^{\mathcal{N}}(\boldsymbol{r}^a, \boldsymbol{r}^b)\, \mathcal{L}^{\mathcal{N}}(\boldsymbol{r}^a, \boldsymbol{r}^c)}{\mathcal{L}^{\mathcal{N}}(\boldsymbol{r}^a, \boldsymbol{r}^b)^2 + \mathcal{L}^{\mathcal{N}}(\boldsymbol{r}^a, \boldsymbol{r}^c)^2 + \mathcal{L}^{\mathcal{N}}(\boldsymbol{r}^c, \boldsymbol{r}^b)^2}. \tag{3.11}$$

If $\gamma(k_0) > \gamma(k_1)$ and the node $\boldsymbol{r}$ is not too close to other node in Γ, the point $\boldsymbol{r}$ is included in the domain as shown in figure 3.5(b). Otherwise the node $\boldsymbol{r}$ is discarded in favour of $\boldsymbol{r}^*$.

Step 3

if the node $\boldsymbol{r}$ is kept, the DK is applied to the new node $\boldsymbol{r}$ to obtain a new mesh.

Step 4

Update Γ by removing $\mathcal{L}_b$. Then:

- If $\boldsymbol{r}$ was selected, the segments $\mathcal{L}_0 = (\boldsymbol{r}_b^s, \boldsymbol{r})$ and $\mathcal{L}_1 = (\boldsymbol{r}, \boldsymbol{r}_b^e)$ are added to Γ.
- If $\boldsymbol{r}^*$ was selected:
  - if $\mathcal{L}_0 = (\boldsymbol{r}_b^s, \boldsymbol{r}^*)$ belongs to Γ, remove it from the front, otherwise add it.
  - if $\mathcal{L}_0 = (\boldsymbol{r}^*, \boldsymbol{r}_b^e)$ belongs to Γ, remove it from the front, otherwise add it.

The resulting front is shown in figure 3.5(c).

Step 5

If Γ is not empty, return to step 1. The mesh is thus constructed by an advancing front, as shown in figure 3.5(d).

Step 6

After the mesh $\mathcal{F}(\mathbf{\Omega})$ is constructed, it must be optimized such that the mean segment length is close to the unity. To this end an optimization process can be applied on the mesh node positions, such that the mean and minimum quality are maximized:

$$\mathcal{F}^*(\mathbf{\Omega}) = \max_{\mathcal{F}(\mathbf{\Omega})} \left(\frac{1}{K} \sum_{k \in \mathcal{F}(\mathbf{\Omega})} \gamma(k) + \min_{\mathcal{F}(\mathbf{\Omega})} \gamma(k) \right). \tag{3.12}$$

In order to do this, normally, Laplacian smoothing [126], gradient-guided methods to improve node position and/or diagonal swapping [127] is applied.

3.2.3 Adaptive scheme

To produce a mesh with controlled interpolation error, an adaptation process is performed. This process improves the mesh using the metrics on a previous mesh as error estimation. Since the real geometry of the image (i.e. radiotracer distribution of a PET image) is unknown, the error estimation is done *a posteriori*, based on the FE solution on the current mesh. The objective of the remeshing adaptation process is to minimize the error:

$$e_{\mathcal{F}} = \mathcal{S} - \mathcal{S}_{\mathcal{F}}, \tag{3.13}$$

where $\mathcal{S}$ is the exact unknown solution and $\mathcal{S}_{\mathcal{F}}$ is the solution obtained by FE calculation. The correct estimation of equation 3.13 is problematic, since the FE solution on a given node is not the real value at that node (the error at the nodes location is not zero). Nevertheless the error $e_{\mathcal{F}}$ can be assumed to be bounded by an indirect measurement:

$$\|e_{\mathcal{F}}\| \leq C \|\tilde{e_{\mathcal{F}}}\|, \tag{3.14}$$

where C is a constant and $\tilde{e_{\mathcal{F}}} = \mathcal{S} - \tilde{\mathcal{S}_{\mathcal{F}}}$ is the deviation between the exact solution $\mathcal{S}$ and an interpolation of $\mathcal{S}$ in the mesh $\mathcal{F}$. Thus, that consists in simplifying the problem and building a new mesh $\mathcal{F}'$ on which, for a given interpolation of the solution $\tilde{\mathcal{S}_{\mathcal{F}}}$, the interpolation error is limited by a given threshold δ. The interpolation error can be bounded locally, in the vicinity of a node, using a simple approximation based on the local deviation of the solution surface [125]. By defining the vicinity of node $\boldsymbol{r}$ using a Taylor expansion of second order:

$$\phi(\Delta r_x, \Delta r_y) = \boldsymbol{r} + \phi'_x \Delta r_x + \phi'_y \Delta r_y + \frac{1}{2}[\phi''_{xx} \Delta r_x^2 + \phi''_{xy} \Delta r_x \Delta r_y + \phi''_{yy} \Delta r_y^2] + o(\Delta r_x^2 + \Delta r_y^2)\texttt{err}, \tag{3.15}$$

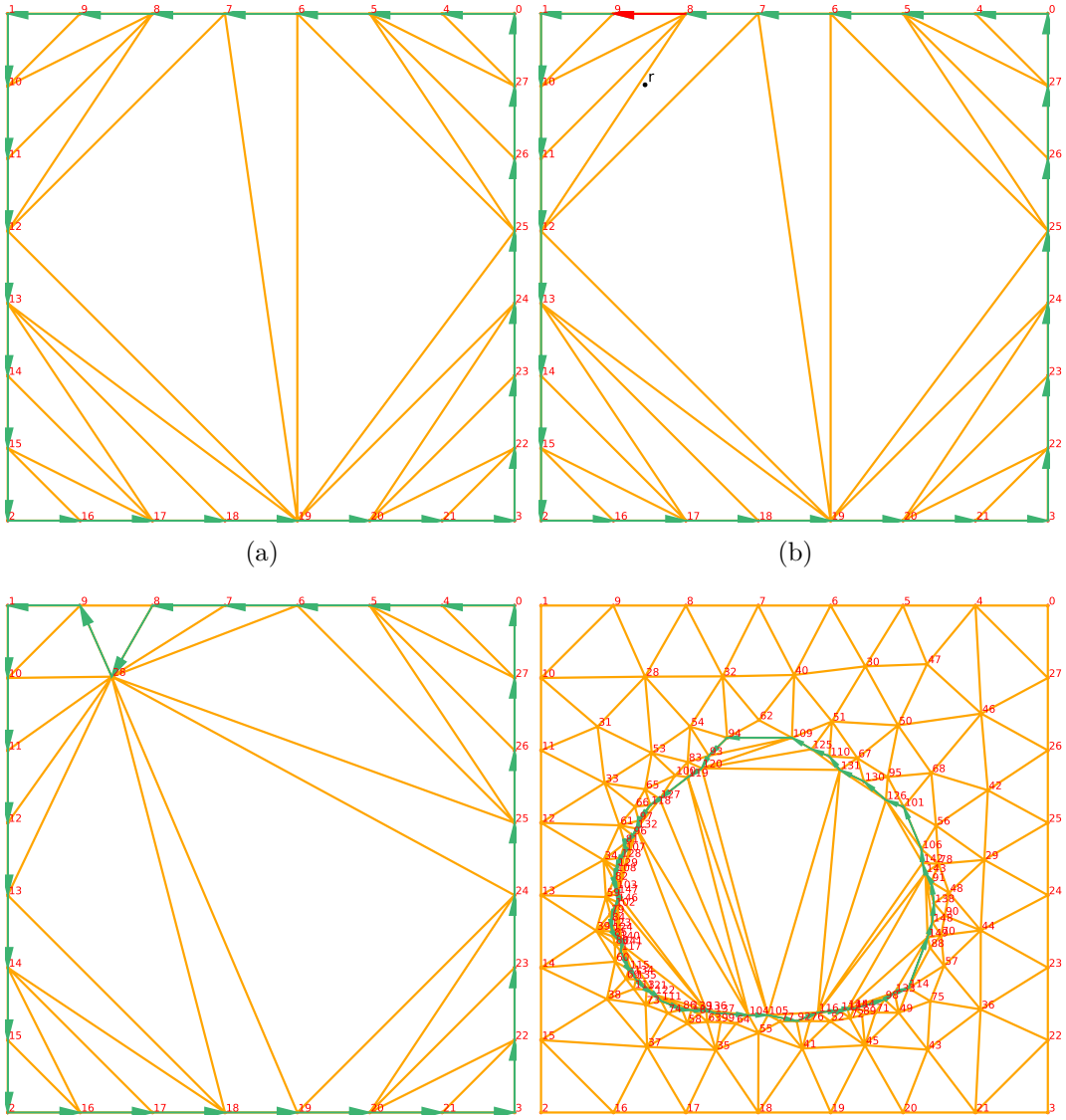


Figure 3.5: Delaunay advancing front mesh construction of NEMA phantom. In (a) the initial step, which applies the DK on the segmented boundary. The front Γ is coincident with the boundary and is shown as series of green arrows. In (b) the smallest segment of Γ is chosen (red arrow) and an internal node $\boldsymbol{r}$ is generated (in black). In (c) the DK is applied and the front Γ is updated to include the new node. In (d) the step 150 of the DAF is shown, where the front Γ is completely separated from the boundary. The final meshed support can be seen in figure 3.1.

where the derivatives of $\phi()$ are evaluated at $\boldsymbol{r}$. The deviation to the normal plane at the node $\boldsymbol{r}$ to an adjacent node $\boldsymbol{r}^{\mathtt{ad}}$ can then be defined as:

$$\mathtt{err}(\boldsymbol{r}, \boldsymbol{r}^{\mathtt{ad}}) = \langle \mathcal{V}(\boldsymbol{r}), (\phi(\Delta \boldsymbol{r}_x^{\mathtt{ad}}, \Delta \boldsymbol{r}_y^{\mathtt{ad}}) - \phi(0,0)) \rangle, \tag{3.16}$$

where $\langle , \rangle$ denotes the scalar product and $(\Delta \boldsymbol{r}_x^{\mathtt{ad}}, \Delta \boldsymbol{r}_y^{\mathtt{ad}})$ is the deviation from node $\boldsymbol{r}$ to node $\boldsymbol{r}^{\mathtt{ad}}$. The equation 3.16 can be written in terms of the Hessian as:

$$\mathtt{err}(\boldsymbol{r}, \boldsymbol{r}^{\mathtt{ad}}) = \frac{1}{2}[\langle \mathcal{V}(\boldsymbol{r}), \phi''_{xx} \rangle (\Delta \boldsymbol{r}_x^{\mathtt{ad}})^2 + \langle \mathcal{V}(\boldsymbol{r}), \phi''_{xy} \rangle \Delta \boldsymbol{r}_x^{\mathtt{ad}} \Delta \boldsymbol{r}_y^{\mathtt{ad}} + \langle \mathcal{V}(\boldsymbol{r}), \phi''_{yy} \rangle (\Delta \boldsymbol{r}_y^{\mathtt{ad}})^2] + o(\Delta(\boldsymbol{r}_x^{\mathtt{ad}})^2 + (\Delta \boldsymbol{r}_y^{\mathtt{ad}})^2). \tag{3.17}$$

Then the maximal deviation from the normal plane at point $\boldsymbol{r}$ to all its neighbouring nodes $\boldsymbol{r}^{\mathtt{ad}}$ is defined as:

$$\mathcal{M}(\boldsymbol{r}, \mathcal{S}(\boldsymbol{r})) = \max_{\boldsymbol{r}^{\mathtt{ad}}} \mathtt{err}(\boldsymbol{r}, \boldsymbol{r}^{\mathtt{ad}}). \tag{3.18}$$

Resulting in a size metric of:

$$h(\boldsymbol{r}) = \frac{\delta}{\mathcal{M}(\boldsymbol{r}, \mathcal{S}(\boldsymbol{r}))}. \tag{3.19}$$

By applying equation 3.19 on each node of the mesh $\mathcal{F}(\Omega)$, a size-map of the domain $\mathcal{H}(\Omega)$, associated with the nodes in $\mathcal{F}$ is obtained.

Construction of a mesh using the adaptive scheme

The adaptive remeshing scheme starts with a coarse initial mesh which is generated using only the corners of a rectangular domain and the expected segment size as input. The initial four segments, defined by the corner coordinates, are subdivided into N_s segments following equation 3.8 and equation 3.9, using $h()$ equal to the expected segment size. The resulting segments compose the front Γ, which is meshed using the DAK method presented in section 3.2.2. This way the initial mesh $\mathcal{F}^{t=0}$ is created, an uniform mesh with no previous information of the domain. Using the mesh $\mathcal{F}^{t=0}$, the adaptation process starts. The FE calculation is processed (a reconstruction process in the case of PET imaging) on the initial mesh, obtaining the interpolated solution $\tilde{\mathcal{S}}$

3.3 PET activity reconstruction on 2D meshes

In this section the PET image reconstruction in the mesh domain is presented. First the current approaches for PET and SPECT image reconstruction on meshes are discussed. Then a new method, based on series of adaptive meshing of series of 2D meshes that represent the FoV volume, is described. This method is based on the meshing methods presented in section 3.2, using an adapted ML-EM method on the mesh domain in place

Algorithm 1: Adaptive remeshing based on the Delaunay Advancing Front

```
t = 0
while e_s > δ do
    Perform the FE calculation on F^(t)(Ω).
    Calculate the size map H(Ω) using equation 3.19.
    Obtain F^(t+1)(Ω) by means of the DAK method, using the size map H(Ω).
      Evaluate error "e" between neighbouring nodes of F^(t+1)(Ω), using trilinear
      interpolation.
    t+ = 1
end
```

of FE calculation. In the last part of this section, the results of the proposed method are presented and compared to the voxelized support reconstruction.

3.3.1 State of the art

The use of mesh support for reconstruction procedures for medical imaging has been studied in the past for PET [128–131] and Single Photon Emission Computed Tomography (SPECT) [132]. These reconstructions are based on Maximum Likelihood-Expectation Maximization (MLEM) or Ordered Subset Expectation Maximization (OSEM). These studies shown that projection through a reduced support can be faster than a pixel support in 2D reconstruction [132]. Also, the works of Sitek, Pereira and Boutchko [128,130,131] reported similar or improved image quality when using a mesh or point-cloud based support. In these studies the quality of the support is highlighted as an important aspect of the reconstruction, however simple mesh updating techniques are applied, due the cost of computing an optimal mesh support.

Fixed mesh domain

The PET image reconstruction on mesh domain proposed in [133] uses a fixed mesh based on the meshing of a CT image. This approach uses the mesh to reduce the number of unknowns in the reconstruction process of the PET image and improves the image quality overall by including the anatomical priors of the CT image. The PET reconstruction is carried out using non refined 2D meshes using a ML-EM algorithm. The model was tested on a simulation containing only 2D LoRs (included in the meshed slice), showing improved bias-variance curves in the task of separating white and gray matter in brain reconstructions. The model did not include information on 3D extension of the algorithm nor mesh quality improvement. The algorithm was further developed and applied to SPEC image reconstruction [132]. This method was defined for 2D and 3D meshes. However the definition for 2D meshes uses only LoRs contained in the mesh slice. The reconstruction is done using OSEM algorithm. The mesh support is based on an initial voxelized image, obtained through Filtered Back Projection (FBP). This initial image is meshed using a content-adaptive mesh

model [134] based on non uniform sampling of a feature map obtained from the spatial distribution of the second directional derivative of the initial image. The meshing method then controls the error by controlling the number of nodes that are placed, which they subsequently control using the minimum description length (MDL) principle [135]. The method was tested on a cardiac torso simulation slice, showing improved image quality when compared to pixel support reconstruction.

Adaptive mesh domain

In [129] they implement a 2D mesh representation that dynamically adapts to the underlying tracer distribution by means of remeshing. The initial representation is a coarse random mesh which adapts to the tracer distribution at every reconstruction step. The mesh is guided by a feature map based on the hessian of reconstructed image and the nodes are located using the Floyd–Steinberg error-diffusion algorithm [136]. Their reconstruction method and representation is 2D, no extension to 3D reconstruction is given. In [128] a simple dynamic 3D mesh reconstruction algorithm based on ML-EM is presented. It consists in considering the reconstruction volume as a set of points (or point cloud) which is meshed using a Delaunay tetrahedrization method. It that case, the mesh is adapted to the domain by adding points in the centers of elements whose variation is the largest along their vertex. This is a simple approach that requires low computational power but the resulting number of nodes and mesh quality are not controlled. Such a method is tested on the Jaszczak anthropomorphic cardiac torso phantom, achieving marginally lower resolution than the voxelized support. In a following work [130] they test several approaches improving the node inclusion within large variation tetrahedra, following a histogram based priority update and a base image as guide. Finally in Ref. [131] they propose to start with a highly dense grid which is sequentially coarsened to obtain a mesh support that does not miss out image details. In this last work they point out need of enhancing the mesh support quality to avoid wedge-shaped artifacts and numerical instability due to degenerate mesh elements. However a more sophisticated adaptive remeshing process exists and was not explored in the literature. It consists in a total remeshing of the full domain by controlling both the size shape and the quality of the mesh though an a priori error map estimator based on the Hessian of the physical solution (see [137–140]) such as presented in section 3.2.3. Such adaptive remeshing process has shown its efficiency, particularly for optimization in complex FE calculations [140].

3.3.2 3D tomographic reconstruction problem using 2D meshes

The presented method differs from previous PET image reconstruction on mesh attempts in two main aspects: first the mesh support is based on a collection of 2D meshes that represents a 3D volume. This representation allows to implement fully 3D reconstruction while using simpler 2D remeshing techniques which can be processed in parallel. Second the mesh support is dynamically adapted to the distribution, starting from a coarse mesh without any particular configuration. The location and number of the mesh nodes is determined using a strategy that controls the interpolation error

and mesh quality. The combination of these techniques produces a high quality compact representation of the reconstruction process. The accuracy is governed through the interpolation error of the support. It is shown that such combination enables to decrease the number of parameters by more than one order of magnitude.

The proposed model is shown in figure 3.6 . It can be seen that instead of using a 3D mesh to describe the FoV, a multi slice 2D mesh model is proposed to describe the distribution $\Lambda^{(t)}$.

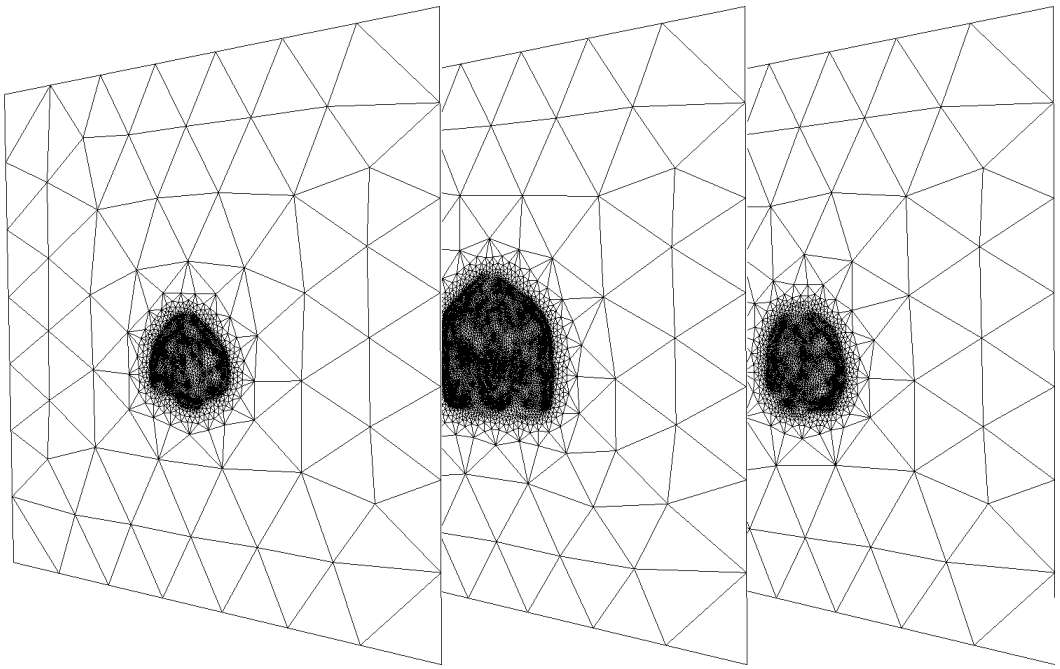


Figure 3.6: Series of 2D meshes representing a volume. Each mesh represents a given part of the axial axis.

The reconstruction operation to be applied for this new support is based on the ML-EM algorithm described in section 1.3.1. The projection and back-projection operations were modified to work using the mesh support.

The presented method uses List Mode representation. This mode is chosen to fit the needs of the *AR-PET* prototype (see chapter 2). Since the cameras are in continual rotation, the sinogram based reconstruction methods are unfitted to fully exploit the rotational and depth of interaction capability of the scanner. List Mode data is better suited for this application as shown in [141], where it was applied in voxelized support reconstruction.

Projection operation on 2D mesh support

Given that the activity within an unknown volume is continually distributed, it can be represented with a continuous 3D spatial function Λ. This function can be used to obtain the expected amount of events detected in a Line of Response (LoR) by means

of modeling each event count in the trajectory of a LoR using a Poisson distribution. The parameter of this distribution is the activity along the LoR trajectory. Then the number of expected events in the trajectory of each LoR can be calculated by the integration of the LoR trajectory in the current estimate $\Lambda^{(t)}$, as presented in equation 3.20, where s_l is the line differential along the l^{th} LoR.

$$\hat{x}_l = \int \Lambda^{(t)}(s_l) ds_l. \tag{3.20}$$

The volumetric activity $\Lambda^{(t)}$ is sampled using the total number N_s of axial slices where the meshes are generated. The activity along the z-axis of a single slice is considered constant (as in voxel representation), hence the activity within a slice only depends on coordinates (x, y). In this representation scheme, the resulting element is a triangular prism (or *thick* triangle, with thickness Δz) which can be treated as a triangular element for interpolation in a slice. The activity in a slice can be presented as the bilinear interpolation of the values at each node of each current element. Each mesh element then applies a triangular shape function Φ to approximate the value of $\Lambda^{(t)}$ at any given position. The shape function can be written as finding the value at the coordinates of the point in a plane (a plane is defined by 3 coplanar points). The unknown coordinates of a 3D LoR are defined by using the slope-intercept notation $\boldsymbol{r}(a) = \boldsymbol{r}^P.a + \boldsymbol{r}^O$. The points $\boldsymbol{r}^O = [r_x^O, r_y^O, r_z^O]$ and $\boldsymbol{r}^P = [r_x^P, r_y^P, r_z^P]$ are the three dimensional coordinates of the origin and the slope of the LoR. However, the LoR inside a 2D mesh drops the z-axis coordinate since the activity along this axis is constant within the mesh. Thus the LoR within the mesh is defined by $\boldsymbol{r}^O = [r_x^O, r_y^O]$ and $\boldsymbol{r}^P = [r_x^P, r_y^P]$. Each vertex $\boldsymbol{r}^{v,e}$ (with $v \in [1, 2, 3]$) for each element e of the mesh is described by its position and activity estimation $\boldsymbol{r}^v = [r_x^{v,e}, r_y^{v,e}, \lambda_{v,e}]$. The activity at the node position is given by the volumetric activity function $\lambda_v^e = \Lambda^{(t)}(r_x^{v,e}, r_y^{v,e}, z_s)$, where z_s is the position of the mesh slice in the z-axis of the FoV. Also, the coordinates of a vertex $\boldsymbol{r}^{v,e}$ represent the coordinates of the mesh x and y and the activity value λ. In figure 3.7 the notion of the extended node coordinates is represented. By using this notation the shape function is obtained by derivation of the point-normal form of a plane described by three points: $\boldsymbol{r}^{0,e}$, $\boldsymbol{r}^{1,e}$ and $\boldsymbol{r}^{2,e}$. This function approximates $\Lambda^{(t)}$ at any position inside of an arbitrary element e and can be written as:

$$\Phi^e(\Lambda^{(t)}, e, \boldsymbol{r}^P, \boldsymbol{r}^O, a) = \begin{cases} \frac{d^e - n_x^e(r_x^P a + r_x^O) - n_y^e(r_y^P a + r_y^O)}{n_\lambda^e}, & \text{inside } e \\ 0, & \text{outside } e \end{cases} \tag{3.21}$$

where $\bar{n^e} = [n_x^e, n_y^e, n_\lambda^e]$ is the normal vector of the plane on the activity surface defined by the e^{th} element and distance $d^e = \bar{n^e} \cdot \boldsymbol{r}^{v,e} = (r_y^{3,e} r_x^{2,e} - r_x^{3,e} r_y^{2,e})\ \lambda_1^e + (r_y^{1,e} r_x^{3,e} - r_x^{1,e} r_y^{3,e})\ \lambda_2^e + (r_y^{1,e} r_x^{2,e} - r_x^{1,e} r_y^{2,e})\ \lambda_3^e$ is part of the plane linear equation: $n_x^e\ x + n_y^e\ y + n_\lambda^e\ \lambda - d^e = 0$. The projection operation on the mesh is then described as:

$$\hat{x}_l = x_l^\mu x_l^{\mathtt{eff}\cdot} \sum_{e=1}^{E} \int \Phi^e(\Lambda^{(t)}, e, \boldsymbol{r}_l^P, \boldsymbol{r}_l^O, a) da. \tag{3.22}$$

The elements x_l^μ and $x_l^{\mathtt{eff}\cdot}$ are, respectively, the attenuation correction and detection

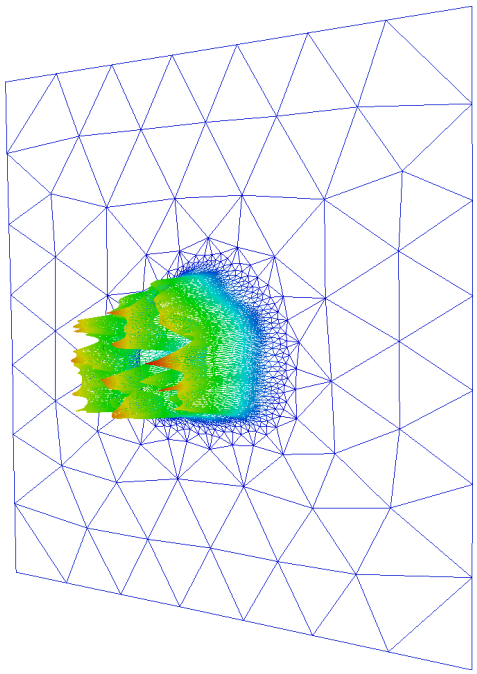


Figure 3.7: 3D representation of a 2D mesh. The height of each node is given by the activity value at the node position.

efficiency of the l^{th} LoR. Operating on equation 3.21 and replacing on equation 3.22, the expression can be reduced to a multiplicative form:

$$\hat{x}_l = x_l^{\mu} x_l^{\texttt{eff.}} \sum_{e=1}^{E} \sum_{v=1}^{2} \frac{\mathcal{K}_{l,e,v} \lambda_v}{n_{\lambda}^{k}}, \tag{3.23}$$

with

$$\begin{aligned} \mathcal{K}_{l,e,1} = [(r_y^{2,e} - r_y^{3,e}).r_x^O + (r_x^{3,e} - r_x^{2,e}).r_y^O + (r_y^{3,e} r_x^{2,e} - r_x^{3,e} r_y^{2,e})](a_{n,e}^{\texttt{out}} - a_{n,e}^{\texttt{in}}) \\ + [(r_y^{2,e} - r_y^{3,e}).r_x^P + (r_x^{3,e} - r_x^{2,e}).r_y^P] \left[\frac{(a_{n,e}^{\texttt{out}})^2 - (a_{n,e}^{\texttt{in}})^2}{2} \right], \end{aligned} \tag{3.24}$$

$$\begin{aligned} \mathcal{K}_{n,e,2} = [(r_y^{3,e} - r_y^{1,e}).r_x^O + (r_x^{1,e} - r_x^{3,e}).r_y^O + (r_y^{1,e} r_x^{3,e} - r_x^{1,e} r_y^{3,e})](a_{n,e}^{\texttt{out}} - a_{n,e}^{\texttt{in}}) \\ + [(r_y^{3,e} - r_y^{1,e}).r_x^P + (r_x^{1,e} - r_x^{3,e}).r_y^P] \left[\frac{(a_{n,e}^{\texttt{out}})^2 - (a_{n,e}^{\texttt{in}})^2}{2} \right], \end{aligned} \tag{3.25}$$

$$\mathcal{K}_{n,e,3} = [(r_y^{1,e} - r_y^{2,e}).r_x^O + (r_x^{2,e} - r_x^{1,e}).r_y^O + (r_y^{1,e}r_x^{2,e} - r_x^{1,e}r_y^{2,e})](a_{n,e}^{\mathtt{out}} - a_{n,e}^{\mathtt{in}}) + [(r_y^{1,e} - r_y^{2,e}).r_x^P + (r_x^{2,e} - r_x^{1,e}).r_y^P] \left[\frac{(a_{n,e}^{\mathtt{out}})^2 - (a_{n,e}^{\mathtt{in}})^2}{2}\right]. \quad (3.26)$$

The elements $a_{n,e}^{\mathtt{in}}$ and $a_{n,e}^{\mathtt{out}}$ of equations 3.24, 3.25 and 3.26 are the input and output value of the scalar a that describe the initial and final point of the l^{th} LoR inside the e^{th} mesh element. These values are obtained by tracing the LoR inside the e^{th} element. The elements $\mathcal{K}_{l,e,v}$, of equations 3.23 , presented in equations 3.24, 3.25 and 3.26, are the contribution of the sampled point to the probability of detecting an event originated at position $\boldsymbol{r}(a)$ in the direction of the l^{th} LoR. This value is derived from the estimation of a point value inside a plane defined by three points. The upper-index v represents the vertices that compose the triangular mesh element e. The total E elements are composed by the I nodes in the mesh. For simplicity the element notation is dropped and the ecuation 3.23 can be rewritten as summation along the I nodes of the mesh, where the matrix of the system can be directly observed:

$$\hat{x}_l = x_l^{\mu} x_l^{\mathtt{eff.}} \sum_{i=1}^{I} \mathcal{K}'_{l,i} \lambda_i, \quad (3.27)$$

with

$$K'_{l,i} = \sum_{e=1,v=i}^{E} \frac{K_{l,e,v}}{n_{\lambda}^{e}}. \quad (3.28)$$

Back-Projection Operation

The inverse operation of the projection, needed to complete the Expectation Maximization (EM) algorithm is constructed as a summation of the influence factor of each of the L LoRs in the list. This operation results in the activity λ for each of the nodes that define the distribution $\Lambda^{(t)}$. The Back-Projection (BP) operation is defined as:

$$\lambda_i = \sum_{l=1}^{L} \frac{x_l^{\mu} x_l^{\mathtt{eff.}} \mathcal{K}'_{l,i}}{\mathtt{SC}_i} x_l. \quad (3.29)$$

The BP operation includes a new Sampling Normalization Factor $\mathtt{SC}_i$ which is needed to correct the non uniform sampling of the mesh representation. This effect is shown figure 3.8, where two different sampling points at positions $\boldsymbol{r}^A$ and $\boldsymbol{r}^B$ contribute to the definition of the activity value in different sized volumes Vol_A and Vol_B, respectively. The activity at each node i is used to interpolate the activity value at every coordinate inside all the elements conformed by it. The combined area of all these elements and the distance between mesh slices creates a volume in which the i node value has influence. Since the elements in the mesh have different sizes, then the volume of influence of each node is different. The relation of the area of influence of node i to

the area of the largest area, define $\mathtt{SC}_i$ value.

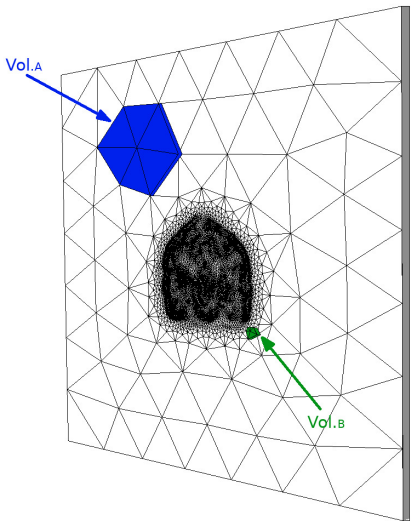


Figure 3.8: Volume of influence of two different nodes. In the upper part of the image, the volume around the central node "A" (Vol_A) and near the center of the image the smaller volume around central node "B" (Vol_B).

Operation on oblique LoRs

The total activity of a 3D LoR crossing within a triangular prism depends on:

1. The activity value at each of the six nodes of the element, 3 from the *front* triangle and 3 from the *back* triangle.
2. The input coordinate.
3. The output coordinate of the LoR.

The crossing of a LoR within the 3D element is represented in figure 3.9. The activity of an arbitrary point $\boldsymbol{r}(a)$ within the 3D element can be calculated using trilinear interpolation. This is achieved by first finding the activity value at the edges of the triangular plane ($\boldsymbol{r}^0$, $\boldsymbol{r}^1$, $\boldsymbol{r}^2$) which contains such point using linear interpolation:

$$\lambda_v = (\frac{\lambda_v^b - \lambda_v^f}{\Delta z})(r_z^P a + r_z^O) + \lambda_v^f, \tag{3.30}$$

and then apply equation 3.21 using the values of the $\boldsymbol{r}^{n,e}$ nodes. However, since it was choosen to make the activity constant along the z-axis of the mesh slice, the activity at nodes of the front and back triangles are the same ($\lambda_v^f = \lambda_v = \lambda_v^b$). Thus the trilinear interpolation is simplified to the bilinear interpolation presented in equation 3.21.

All LoRs that do not fit into a single 2D mesh slice are multi-slice oblique LoRs. These LoRs are segmented on each crossed mesh slice and each segment is projected or

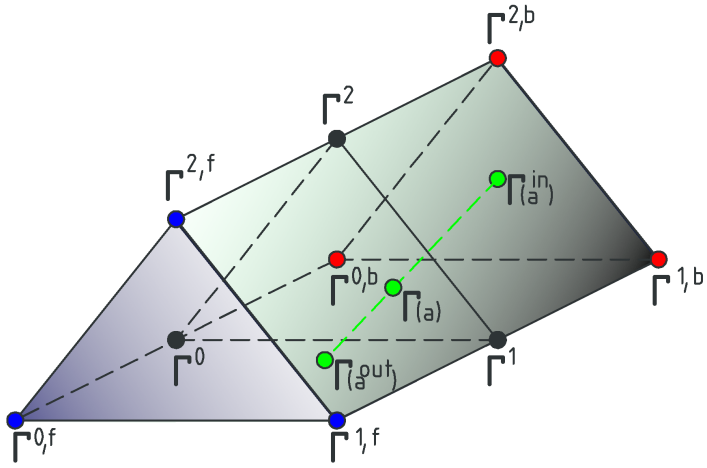


Figure 3.9: Triangular prism representing a 3D mesh element. The blue dots ($\boldsymbol{r}^{0,f}$, $\boldsymbol{r}^{1,f}$, $\boldsymbol{r}^{2,f}$) and the red dots ($\boldsymbol{r}^{0,b}$, $\boldsymbol{r}^{1,b}$, $\boldsymbol{r}^{2,b}$) represent the front and back limits of the prism. The green dotted line with limits $\boldsymbol{r}(a^{\mathtt{in}})$ and $\boldsymbol{r}(a^{\mathtt{out}})$ represents a LoR passing through the element. The green dot $\boldsymbol{r}(a)$ is a coplanar point included in an arbitrary triangular plane within the prism, represented by the black dots ($\boldsymbol{r}^0$, $\boldsymbol{r}^1$, $\boldsymbol{r}^2$).

back-projected using the equations 3.27 and 3.29 respectively. The segmentation of a multi-slice oblique LoR is done by computing the parameter a of the input and output location for each mesh slice n_s: $z_s^{\mathtt{in}} = z_s - \Delta z/2$ and $z_s^{\mathtt{out}} = z_s + \Delta z/2$. Resulting in two points, $\boldsymbol{r}(a_s^{\mathtt{in}})$ and $\boldsymbol{r}(a_s^{\mathtt{out}})$. Then the elements $K_{l,e,v}$ are computed on those values within the segment $\overline{\boldsymbol{r}(a_s^{\mathtt{out}}) - \boldsymbol{r}(a_s^{\mathtt{in}})}$. Finally the LoR is weighted by the its length proportion within the slice.

ML-EM formulation on 2D Mesh

Using the projection and back-projection operation previously defined, the ML-EM expression (defined in equation 1.19), can be re-written as:

$$\lambda_i^{(t+1)} = \frac{\lambda_i^{(t)}}{\sum_{l=1}^{L^*} \frac{x_l^{\mu} x_l^{\mathtt{eff}} \cdot \mathcal{K}'_{l,i}}{\mathtt{SC}_i}} \sum_{l=1}^{L} \frac{1.0}{\hat{x}_l} \frac{x_l^{\mu} x_l^{\mathtt{eff}} \cdot \mathcal{K}'_{l,i}}{\mathtt{SC}_i}, \tag{3.31}$$

where $\sum_{l=1}^{L^*} \frac{AC_i DC_i \mathcal{K}'_{l,i}}{\mathtt{SC}_i}$ is the summation over the densely sampled list of LoRs from the distribution of all measurable LoRs. Each step of the function $\lambda_i^{(t+1)}$ in equation 3.31 increases monotonically the likelihood of the activity distribution $\Lambda^{(t)}$ to the detected LoR set X.

3.3.3 ML-EM Reconstruction using Adaptive Remeshing Scheme

Since the scanner is only able to capture coincidences perpendicular or with a limited inclination to the z-axis, it was chosen to partition the image using series of 2D meshes

along the axial axis of the scanner (the z-axis). When compared to a 3D mesh, the proposed representation requires less computation, at the cost of higher interpolation error of the mesh support on the z-axis. Each of the meshes in the proposed representation is computed as 2D mesh, hence simplifying the remeshing calculation. The volume of the mesh slice is defined by the 2D size of the mesh and the separation to the following slice. Since the activity within a mesh slice is assumed constant along the z-axis, the 3D can be reduced to consider only 2D mesh for each slice. By using the mesh representation node elements can be strategically placed in regions where more compact representation is needed (for sharp edges or complex textures). The ability to control the nodes positions enables the description of the image with potentially less elements than the voxel representation. One strategy to guide the nodes positions is using the adaptive remeshing process [142, 143], presented in section 3.2.

By using the presented reconstruction procedure as the physical model in the remeshing process, a reconstruction operation with bounded interpolation error is achieved. The magnitude of this error is directly related to the accuracy of the projection operation (see equation 3.27). Also the obtained mesh representation is more compact, without losing accuracy, and requires the computation of fewer nodes as the mesh structure converges. The reconstruction procedure is then carried out in the following manner:

Algorithm 2: Reconstruction Process

```
Create N_s initial coarse meshes F_s^(t=0)(Λ^(t=0))
for S_L steps do
    for S_R steps do
        Compute the intensity on each F_s^(t)(Λ^(t)), as a volume (Reconstruction
         Step).
    end
    for all F in N_s do
        Obtain F_s^(t+1)(Λ^(t+1)) by applying remeshing for S_remesh steps (See
         algorithm 1).
    end
end
```

Each of the reconstruction steps in the algorithm 2 is a series of S_R MLEM steps. The number of MLEM reconstructions steps to be applied between remeshing steps is arbitrary. It is not required that the estimated activity reaches an optimal value before updating the mesh support. Since the remeshing process is governed by the threshold δ, the spatial positions of the nodes are adapted in order to minimize the interpolation error. Thus the new support has a bounded error, with respect to the previous support. This results in a controlled change in the projection value $\hat{x}_l$ which in turn controls the change in the objective function (equation. 1.18). Therefore, each remeshing iteration achieves a better quality representation of the unknown activity, with a bounded impact on the objective function.

Computational Cost

The complexity of the remeshing reconstruction iteration is given by the complexity of the remeshing process and the reconstruction process. For the remeshing process, the mesher is based on the incremental construction of a Delaunay triangulation [144]. Namely, internal nodes are generated using an advancing front strategy and are connected together by the Delaunay Kernel (the Delaunay method to insert a node in an existing triangulation). The speed of the Delaunay kernel depends on how fast the vicinity of a point can be identified. It is known that triangulation algorithms take $\mathcal{O}(I \log I)$, where I is the number of nodes, times and specially for that of the construction of the Delaunay triangulation. Actually, for a reasonable number of points and using special data structures, the neighboring of a point can be found in $\mathcal{O}(1)$ time and in this case, the mesher (in this case the BL2D [145]) take $\mathcal{O}(I)$ time. For the reconstruction on the mesh domain the computational cost resides in the tracing of the LoR through the meshes. The reconstruction uses an initial step to sort the nodes and elements of each mesh into N_G sub-groups. This step also builds a data structure of the vicinity of each element in the mesh before starting the projection and back-projection operation. The complexity of this initial step is $\mathcal{O}(N_s\, I\, E)$, where N_s is the number of axial 2D meshes used to describe the volume and E the number of triangle elements in the mesh. Nevertheless this operation is fully parallelizable with GPUs, resulting in no additional cost. Then, the tracing inside a single mesh takes $\mathcal{O}(L\,(E' + E_M))$, where L is the number of LoRs, E' is the maximum number of elements by sub-group, typically $E' \approx E/(N_G^2)$ and E_M is the number of elements touched by the LoR inside the mesh. Finally the computational cost of the remeshing and the two projection operations of the volumetric reconstruction can be written as the sum of the remeshing and reconstruction operations multiplied by the number of meshes N_s:

$$\mathcal{O}_{mesh} = \mathcal{O}(N_s\, I) + 2\,\mathcal{O}(L\, N_s\,(E' + E_M)). \tag{3.32}$$

The cost of the projection and back-projection in the voxelized support can be approximated as $\mathcal{O}(L\, E_V)$, where E_V is the number of voxels touched by the LoR, then the reconstruction cycle takes:

$$\mathcal{O}_{voxel} = 2\,\mathcal{O}(L\, E_V). \tag{3.33}$$

3.3.4 Experiments

The proposed reconstruction strategy is tested using two phantoms: a Monte Carlo (MC) simulation of the Hoffman Brain 3D Phantom and a National Electrical Manufacturers Association (NEMA) image quality phantom [112] acquisition. The simulation was done using the GATE [107] software, where the AR-PET geometry was simulated. The NEMA acquisition was done using the pre-clinical AR-PET prototype. The proposed reconstruction method is compared to the voxelized MLEM reconstruction method described in Ref. [146]. All reconstructions share the same data (Event List), the same mathematical model of the scanner and data correction techniques.

Hoffman Brain phantom simulation

The phantom is placed in the center of the field of view (FoV). The simulation is free of attenuation effects and event positioning errors in order to better reflect the remeshing compression capabilities. During the simulation a total of 1.74×10^8 coincidences are collected (including true and random coincidences).

In order to assess the proposed reconstruction behavior at different levels of statistical noise, the Hoffman brain phantom simulation was reconstructed using four different levels of statistics, 1×10^5, 1×10^6, 1×10^7 and 1.7×10^8 LoRs. The reconstruction on mesh support uses 2 mm slices, an error threshold of $\delta = 0.01$ and $h_{min} = 1.0$ mm. The initial mesh is a near Cartesian grid with a distance between nodes of 4 mm. The process is applied for 10 iterations with $S_R = 10$ reconstruction steps and $S_{\texttt{remesh}} = 5$ remeshing steps, resulting in 50 remesh iterations and 100 reconstruction iterations. The voxel reconstruction is done with $S_R = 100$ iterations and uses the same input list as the presented method. The voxel reconstruction is filtered using a Gaussian filter with standard deviation $\sigma = 1$ mm, to soften any high frequency noise from the reconstruction computation. This filtering is not necessary for the mesh support, it avoids this by re-sampling the support where it does not meet the error threshold δ. Two voxel size configurations were tested, one with a voxel size of $2 \times 2 \times 2$ mm^3, resulting in $256 \times 256 \times 256 = 16777216$ voxels and an other with a voxel size of $4 \times 4 \times 4$ mm^3, resulting in $128 \times 128 \times 128 = 2097152$ voxels. In order to compare the reconstructions, the presented method is projected into a volume of voxel size of $2 \times 2 \times 2$ mm^3 using bilinear interpolation at each slice. The low resolution voxel reconstruction is linearly interpolated to a voxel size of $2 \times 2 \times 2$ mm^3 for comparison.

To compare the reconstructed images, the Signal to Noise Ratio (SNR), the Mean Squared Error (MSE) and white-grey matter contrast ($\mathtt{C}_{\texttt{grey}}$) metrics were used. The contrast metric is calculated as:

$$\mathtt{C}_{\texttt{grey}} = \frac{\overline{\mathcal{V}_{\texttt{grey}}} - \overline{\mathcal{V}_{\texttt{white}}}}{\overline{\mathcal{V}_{\texttt{white}}}}, \tag{3.34}$$

where $\overline{\mathcal{V}_{\texttt{grey}}}$ and $\overline{\mathcal{V}_{\texttt{white}}}$ are the mean value of the voxels identified as grey and white matter respectively. The segmentation of the voxels was done using a threshold on the ground truth image. Since the image occupies a small fraction of the total FoV, a bounding box around the phantom was described around image and the metrics were applied only inside this box. The size of the bounding box for the Hoffman phantom is $102 \times 102 \times 102 = 1.06 \times 10^6$ voxels.

NEMA phantom acquisition

The phantom is constructed following the guidelines provided by the NEMA and placed at the center of the field of view (FoV). The acquisition is static (single bed position) and the detectors are rotating at 2 rpm. The acquisition has a duration of 47 minutes and 2.0×10^7 prompt coincidences are collected. The injected dose is 1.85 `mCi`, the hot lesions are filled with a concentration of 4 times that of the background.

The attenuation images are acquired using an Aquilion Lightning, Canon Medical Systems 16-row 32 slice helical CT and manually co-registered by using a non attenuation corrected preliminary reconstruction.

The NEMA Phantom was also tested using different amounts of statistics, in this case the sets included 1×10^5, 1×10^6 and 2×10^7 LoRs. The mesh support reconstruction parameters where the same as in the Hoffman Brain simulation reconstruction. The reconstruction on remesh support is applied for 20 iterations with $S_R = 5$ and $S_{\texttt{remesh}} = 5$, resulting in 100 remesh iterations and 100 reconstruction iterations. The voxel reconstruction was done with the same parameters as in the Hoffman Brain simulation. As before, to compare the reconstructions, the presented method is projected into a volume of equal resolution using bilinear interpolation at each slice.

The NEMA phantom adquisition was tested using the metrics described in the NEMA 2012 document [112].

3.3.5 Results

Hoffman Phantom Simulation

The real activity and the reconstruction on mesh support and high resolution voxel support, for the complete set of LoRs, can be seen in figura 3.10, where cuts along of each axis are displayed.

The resulting metrics for each set of LoRs and reconstruction are given in Table 3.1. The evolution of the metrics as a function of the statistics, for the last reconstruction iteration is shown in figure 3.11. Using these metrics the convergence of the reconstruction can be observed. In figure 3.12 the metrics as a function of the iterations, for the largest statistics set, are presented.

Table 3.1: Computed metrics for the Hoffman Brain simulation on each set of statistics.

LoRs	Reconstruction		Metric		
	Support	Resolution	MSE	SNR	$C_{\texttt{grey}}$
1×10^5	Voxel	4 mm	0.17	4.88	0.33
		2 mm	0.21	3.93	0.35
	Mesh	-	0.28	2.66	0.29
1×10^6	Voxel	4 mm	0.11	6.90	0.40
		2 mm	0.11	6.86	0.44
	Mesh	-	0.13	6.15	0.45
1×10^7	Voxel	4 mm	0.099	7.21	0.42
		2 mm	0.094	7.46	0.46
	Mesh	-	0.093	7.51	0.49
1.7×10^8	Voxel	4 mm	0.099	7.22	0.41
		2 mm	0.092	7.51	0.46
	Mesh	-	0.080	8.18	0.57

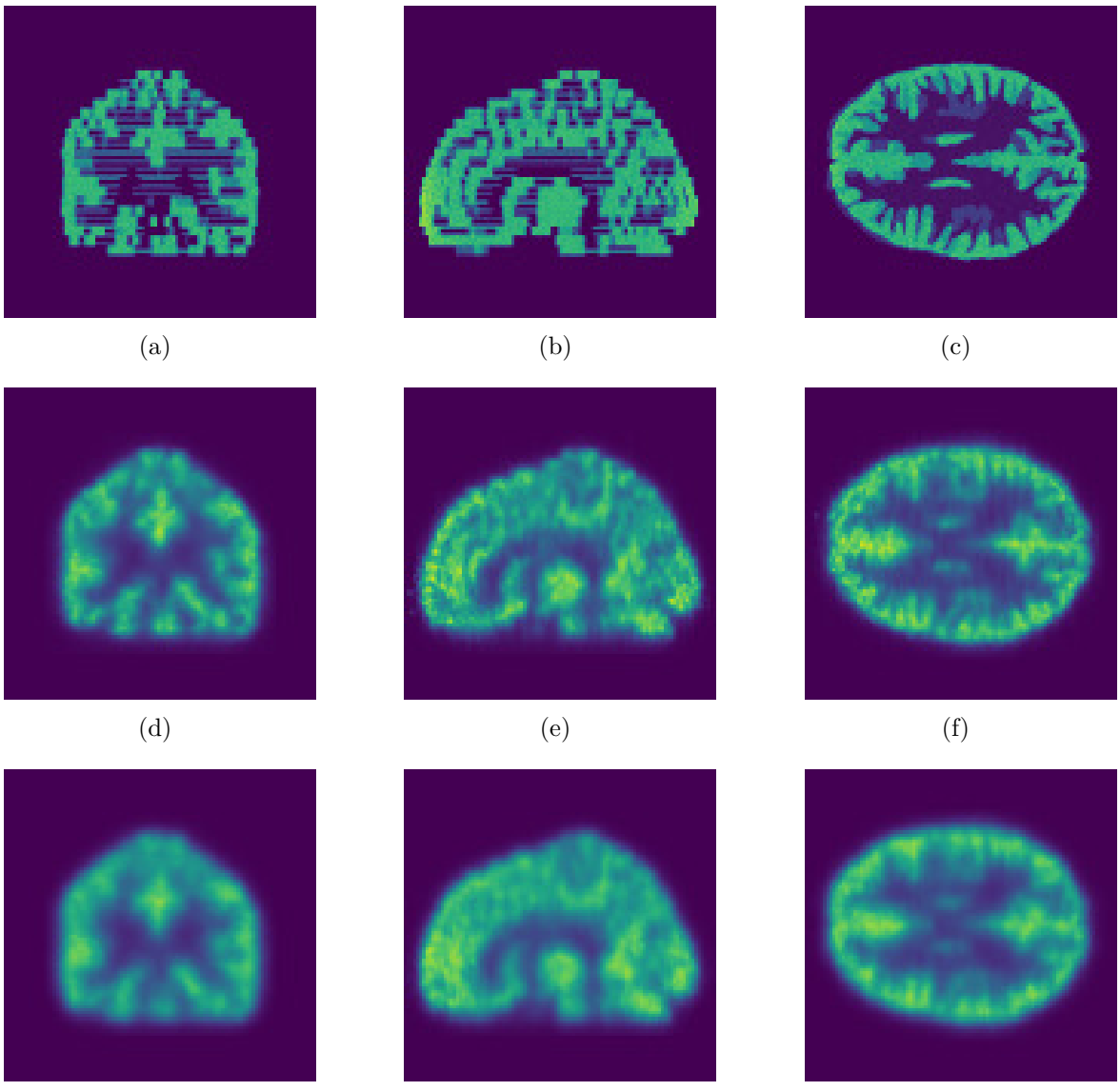


Figure 3.10: Comparison between the real distribution of the activity (a, b, c),the interpolation of the mesh reconstruction into 2 mm pixel support (d, e, f) and the 2 mm voxel support reconstruction (g, h, i) of the Hoffman Brain simulation. The displayed images were reconstructed using the complete set of LoRs.

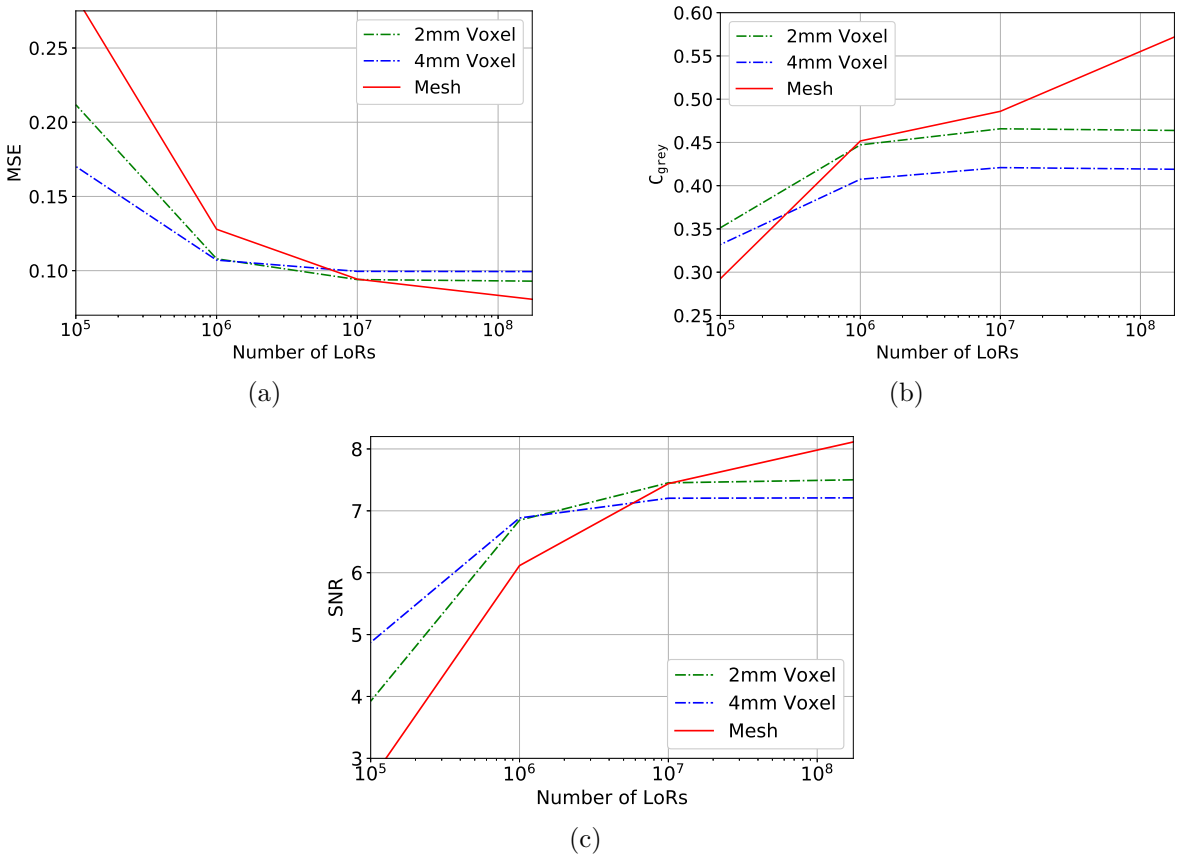


Figure 3.11: Evolution of the MSE (a), $\mathtt{C_{grey}}$ (b) and SNR (c) as function of the amount of collected statistics for the Hoffman Brain simulation. The green dotted line represents the $2 \times 2 \times 2$ mm^3 voxel support reconstruction, the blue dotted line represents the $4 \times 4 \times 4$ mm^3 voxel support and the solid red line represents the mesh support reconstruction.

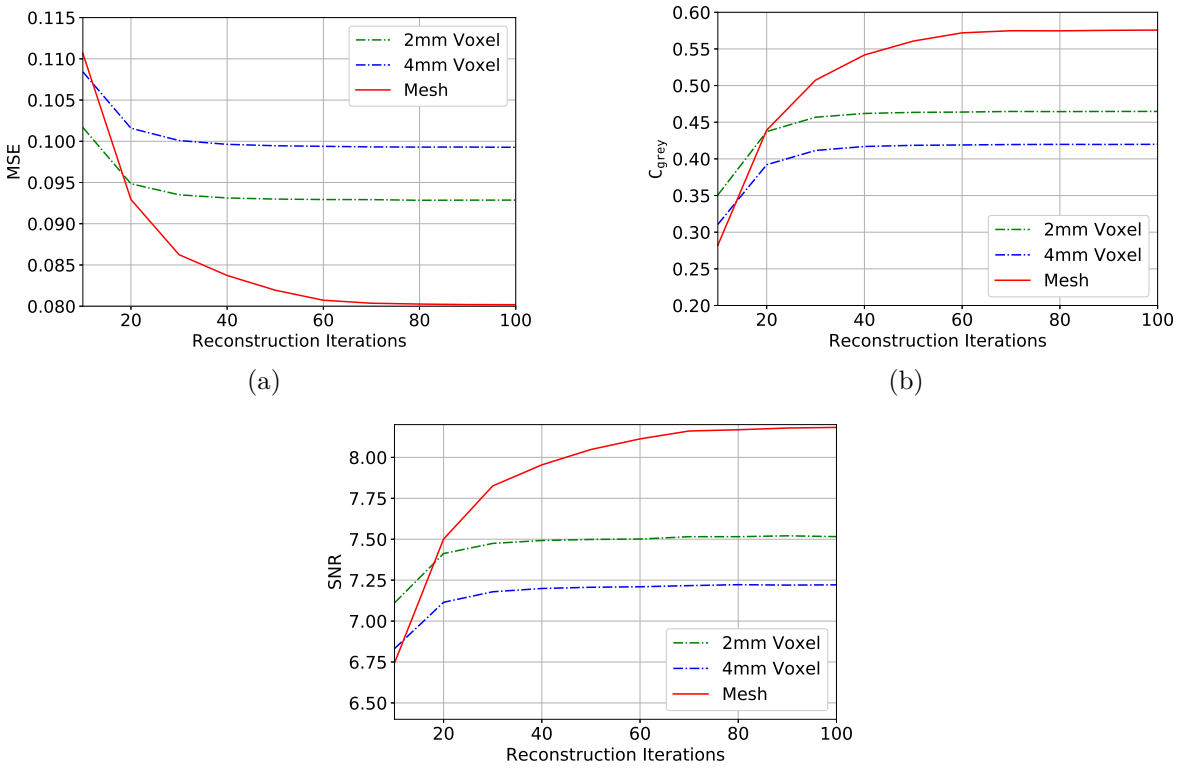


Figure 3.12: Evolution of the MSE (a), $\mathtt{C_{grey}}$ (b) and SNR (c) as function of the iteration for the largest statistics set of the Hoffman Brain simulation. The green dotted line represents the $2 \times 2 \times 2$ mm^3 voxel support reconstruction, the blue dotted line represents the $4 \times 4 \times 4$ mm^3 voxel support and the solid red line represents the mesh support reconstruction.

NEMA Phantom Acquisition

The phantom shape and the reconstruction on mesh support and the reconstruction on the high resolution voxel support can be seen in figure 3.13, where cuts along of each axis are displayed.

The accuracy of attenuation and scatter corrections ($C_{\texttt{lung}}$, lower values are better), contrast on cold spheres ($Q_{\texttt{cold}}$, higher values are better), contrast on hot spheres ($Q_{\texttt{hot}}$, higher values are better) and mean background variability ($N_{\texttt{mean}}$, lower values are better) are summarized, for each set of statistics and resolution, in table 3.2. In figure 3.14 the metrics as a function of the iterations, for the largest statistics set, are presented.

Table 3.2: NEMA 2012 metrics for the phantom acquisition.

LoRs	Reconstruction		Metric			
	Support	Resolution	$C_{\texttt{lung}}$	$Q_{\texttt{cold}}$	$Q_{\texttt{hot}}$	$N_{\texttt{mean}}$
1×10^5	Voxel	4 mm	0.52	0.17 ± 0.06	0.53 ± 0.14	0.61 ± 0.17
		2 mm	0.43	0.15 ± 0.07	0.31 ± 0.05	0.87 ± 0.21
	Mesh	-	0.52	0.23 ± 0.32	0.21 ± 0.29	0.60 ± 0.29
1×10^6	Voxel	4 mm	0.39	0.24 ± 0.09	0.38 ± 0.07	0.23 ± 0.07
		2 mm	0.36	0.23 ± 0.09	0.33 ± 0.06	0.29 ± 0.08
	Mesh	-	0.42	0.18 ± 0.06	0.18 ± 0.06	0.26 ± 0.11
2×10^7	Voxel	4 mm	0.30	0.26 ± 0.00	0.41 ± 0.11	0.065 ± 0.025
		2 mm	0.29	0.26 ± 0.01	0.42 ± 0.11	0.082 ± 0.025
	Mesh	-	0.37	0.33 ± 0.14	0.33 ± 0.14	0.043 ± 0.023

The reconstruction process is complete when the reconstruction and the mesh support converge. The remeshing process produces a spatial configuration of nodes ensuring an interpolation error lower than the given threshold δ. The convergence of the mesh representation can then be observed in terms of the amount of nodes used to represent the volume. In figure 3.15 the evolution of the total number of nodes and nodes in a central central slice, for each phantom, is presented. The selected slice for each reconstruction is presented in figure 3.16.

Evolution of Computational Cost

The worst-case scenario for a LoR trace is that where the LoR touches the most voxels or triangles. In the voxel domain this value can be approximated as an oblique LoR crossing the FoV $E_V \approx \sqrt[2]{(d_x/\mathtt{v}_x)^2 + (d_y/\mathtt{v}_x)^2 + (d_z/\mathtt{v}_x)^2}$, where $\mathtt{v}_x$ is the voxel size and d is the length of the side of the FoV in each axis. In the case of the mesh domain this value varies with the number of slices N_s and evolution of the mesh but an approximation can be calculated by dividing the volume with a virtual voxel will have a size of $\mathtt{v}'_x = \sqrt[3]{Vol_{FoV}/\sum_{n=1}^{N_s} E_n^t}$, where Vol_{FoV} is the volume of the FoV and $\sum_{n=1}^{N_s} E_n^t$ is the sum of the elements of all meshes at step t. Then the number of triangles touched by the LoR inside a single mesh is given by $E_M \approx 2\sqrt[2]{(d_x/\mathtt{v}'_x)^2 + (d_y/\mathtt{v}'_x)^2 + (d_z/\mathtt{v}'_x)^2}$.

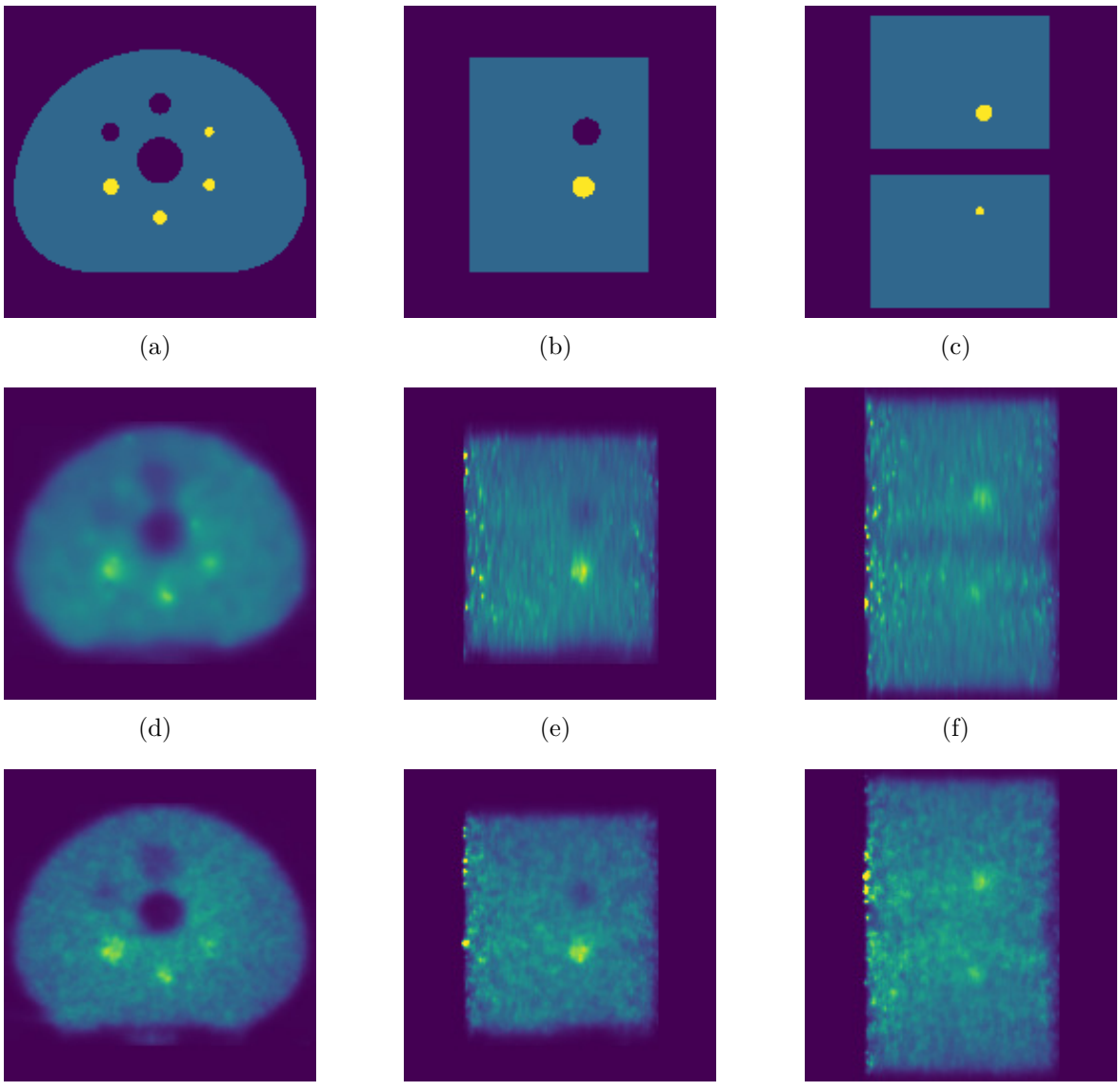


Figure 3.13: Comparison between the expected distribution of the activity (a, b, c), the interpolation of the mesh reconstruction into 2 mm pixel support (d, e, f) and the 2 mm voxel support reconstruction (g, h, i) of the NEMA phantom acquisition. The displayed images were reconstructed using the complete set of LoRs.

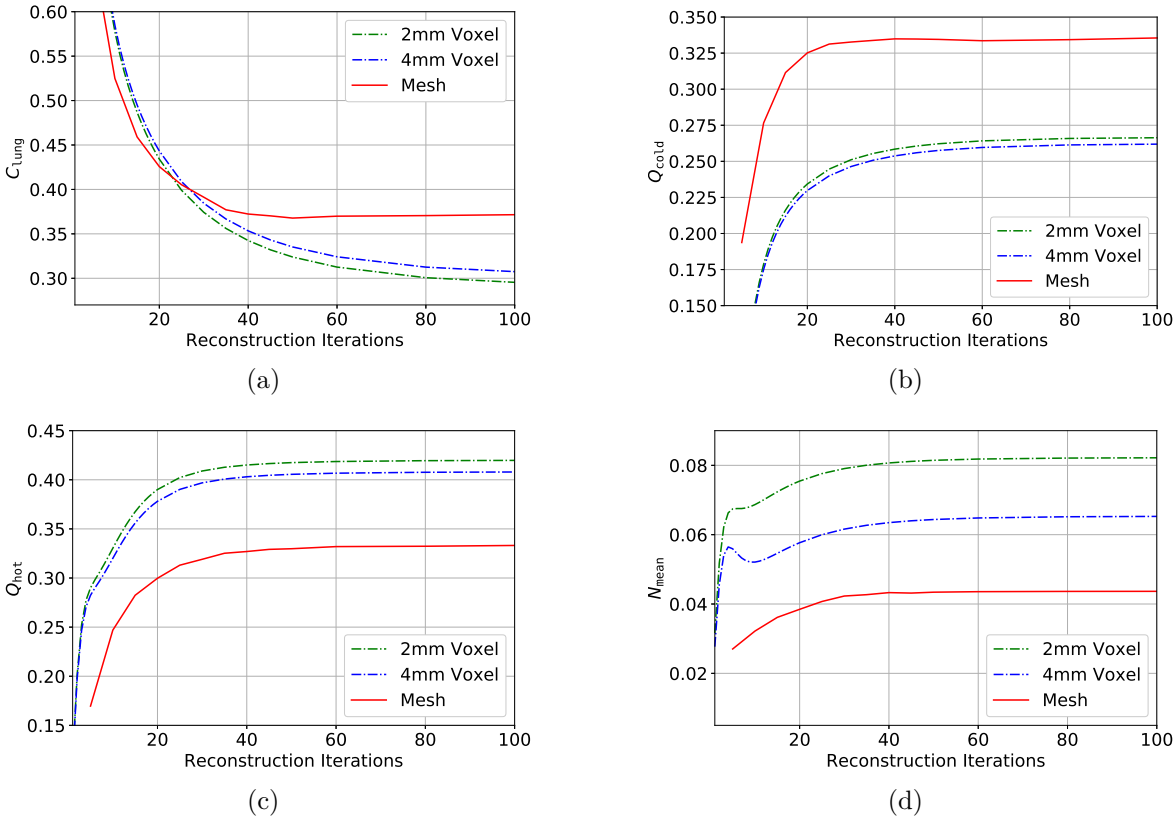
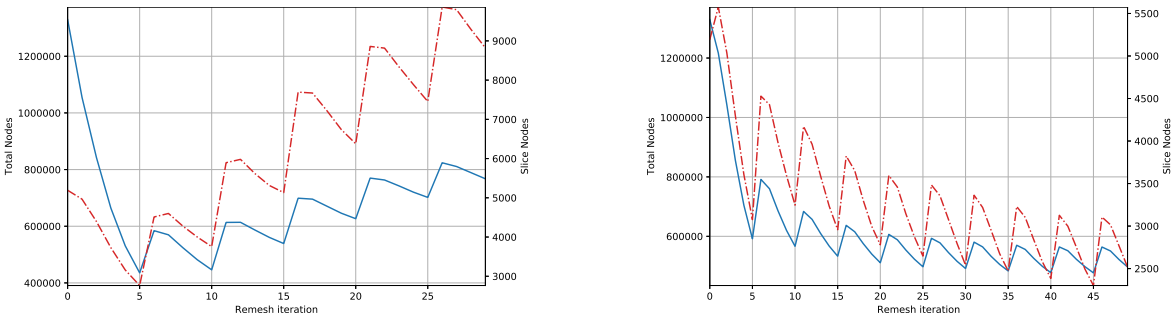


Figure 3.14: Evolution of the attenuation and scatter corrections $C_{\texttt{lung}}$ (a), contrast on cold spheres $Q_{\texttt{cold}}$ (b), contrast on hot spheres $Q_{\texttt{hot}}$ (c) and mean background variability $N_{\texttt{mean}}$ (d) as function of the iteration for the largest statistics set of the NEMA phantom. The green dotted line represents the $2\times2\times2$ mm^3 voxel support reconstruction, the blue dotted line represents the $4\times4\times4$ mm^3 voxel support and the solid red line represents the mesh support reconstruction.



(a) (b)

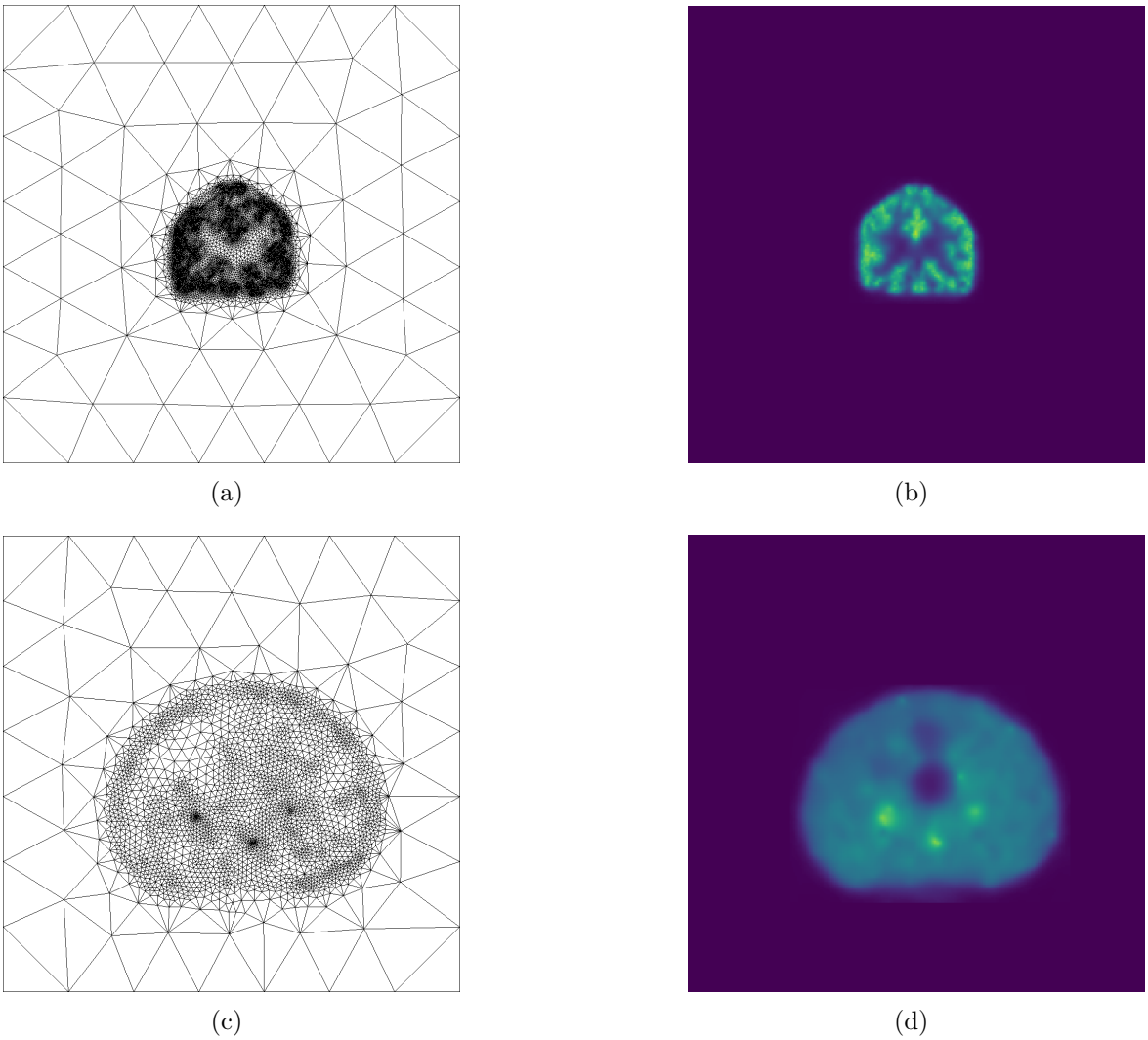
Figure 3.15: Evolution of the number of nodes used to sample the Hoffman Brain simulation (a) and the NEMA phantom acquisition (b). The solid line represents the total number of nodes (left scale) and the dotted line the evolution of the number of nodes at a central slice of the Phantom (right scale). The x-axis represents the remesh process iteration, every five iterations a reconstruction step was applied.

Using this approximation the evolution of the cost of the proposed algorithm against a voxelized support of voxel size $2 \times 2 \times 2$ mm^3, is shown in figure 3.17.

3.3.6 Discussion

The presented method achieved a superior quality on the reconstruction of the Hoffman brain phantom, improving the MSE from 0.092 to 0.080, the SNR from 7.51 `db` to 8.18 `db` and the grey matter contrast from 0.46 to 0.57 when compared to the voxelized reconstruction. For the NEMA reconstruction, the mesh achieved partial success, improving the contrast on cold spheres from 0.26 to 0.33 and the background variability from 0.082 to 0.043 but deteriorating the attenuation correction from 0.29 to 0.37 and the hot spheres contrast from 0.42 to 0.33.

It can be seen that the Hoffman Brain simulation metrics (Table 3.1) for the mesh support and high resolution voxel are very similar for the statistics sets of 10×10^6 and 10×10^7 LoRs. When the number of LoRs is above 1×10^7, the mesh support is superior to the voxelized support. However the mesh reconstruction quality decays faster with the number of LoRs than the voxelized reconstruction. This effect is seen in figure 3.11, where the MSE and SNR metrics of the final iteration of the mesh reconstruction are lower than the voxelized reconstruction for statistics lower than 1×10^7. It is worth noting that this also happens when comparing the voxelized reconstructions, the 4 mm support achieves better or similar results than the 2 mm voxel support for 1×10^6 or lower statistics sets. The same is observed for the NEMA phantom in Table 3.2, where the mesh reconstructions for 1×10^6 LoRs achieve lower scores. In the case of the NEMA phantom acquisition the mesh presents better contrast of cold spheres and background variability, however it achieves lower performance in the attenuation correction and hot spheres contrast. This behaviour can be related to failures in the mesh representation, which is more susceptible to fewer or noisier data.



(a) (b)

(c) (d)

Figure 3.16: Mesh representation (a, c) and corresponding activity distribution (b, d) of a detailed slice from the Hoffman Brain phantom (a,b) and the NEMA phantom (c, d) reconstructions.

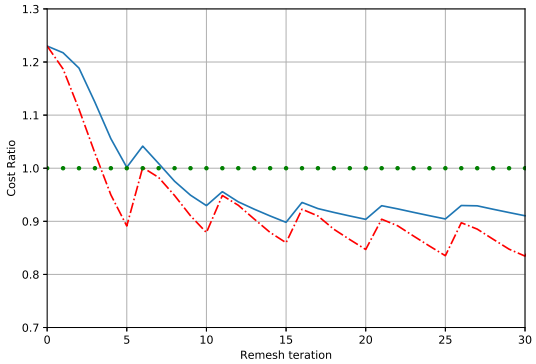


Figure 3.17: Evolution of the computational cost ratio for the Hoffman Brain simulation (solid blue line) and the NEMA phantom acquisition (dashed red line). The cost ratio is calculated as the cost of the reconstruction in mesh support over the voxelized support. The threshold (dotted green line) line represents voxelized reference cost for a $2 \times 2 \times 2$ mm^3 voxel size. The x-axis represents the remesh process iteration, every five iterations a reconstruction step was applied.

The application the MLEM algorithm on mesh support can improve the image quality for larger statistics and reduce the amount of parameters required to represent the activity distribution. The Hoffman brain requires 7.67×10^5 nodes to be represented and the NEMA phantom 5.22×10^5 nodes, the high resolution voxel reconstruction requires 1.7×10^7 voxels, and the low resolution 2.1×10^6. This is more than an order of magnitude in the total number of elements to present the same information. In order to reduce the effect of large vacant zones in the FoV, one can analyze a detailed slice of the reconstruction. Such slice requires 8.82×10^3 nodes for the Hoffman Brain and 2.55×10^3 nodes for the NEMA phantom. The same slice in voxelized support requires $256 \times 256 = 6.55 \times 10^4$ and $128 \times 128 = 1.64 \times 10^4$ voxels for the high resolution and low resolution representation respectively. Also in this case the representation uses at most half of the parameters in the worst case. This improvement can be exploited to implement more complex reconstruction procedures and dynamic PET imaging which benefit from a reduced system matrix.

As in previous studies [128–132] the implementation of a mesh support resulted in similar or superior reconstruction quality and a reduction of the dimensionality of the problem. These studies also reported a reduction in the background noise when reconstructing real acquisitions [129, 131], this was also observed in our experiments, with the decrease of the mean background variability metric on the NEMA phantom. In addition to the new approach to activity reconstruction on mesh support, the variability of the reconstruction quality with respect to the amount of statistics was explored. This was unexplored in the literature and it was shown that it affects the reconstruction accuracy on mesh support.

The theoretical cost of computing the MLEM algorithm using presented method against voxelized representations is smaller when the target quality is above that of a $2 \times 2 \times 2$ mm^3 voxel. As it can be seen in figure 3.17, after 10 remeshing steps the com-

putational cost of the algorithm is almost 10% lower than the $2 \times 2 \times 2$ mm^3 voxelized representation. However, as the cost is linear with the voxel size, a representation of voxel size $4 \times 4 \times 4$ mm^3 (the mean voxel size in commercial scanners, which ranges from 3 to 5 mm [31]), is faster than the presented method.

In order to avoid wrong representations of the attenuation image due to the mesh being adapted to the activity distribution, it was choosen to use a secondary mesh to represent the attenuation map. While this approach requires tracing of the LoR over a new domain, the attenuation can be pre-computed, since it will not change over time. Optimization of a single support for attenuation and activity, to be used in algorithms such as Maximum Likelihood estimation of Activity and Attenuation, is left as future work.

3.3.7 Conclusions

The presented strategy for PET image reconstruction based on MLEM and remeshing results in a high quality image with reduced dimensionality. The remeshing process being governed by the interpolation error, the method is adapted to treat any existing structures, avoiding losing details in the process of compacting the mesh and the need of manually selecting RoIs in the image. It was shown that the method is able to reconstruct the activity images and to describe the volume by using series of 2D meshes, without needing complex and computationally costly 3D remeshing strategies. Moreover the mesh slices contain less nodes than a volumetric mesh and are independent thus they can be optimized using parallel processing techniques, greatly reducing the computational cost of the mesh optimization algorithm. The reduction in the optimization cost can be evidenced by means of the asymptotic computational cost, while a 3D mesh composed of K nodes takes at least $\mathcal{O}(K)$ operations, the parallel optimization of S 2D meshes results in $\mathcal{O}(K/S)$ operations. The results have shown that not only the proposed strategy produces better resolution relatively to state of the art methods but also it permits to reduce significantly the number of parameters in the image (voxels v.s. nodes). This reduction has a greater impact when small voxels size are used. The combination of these techniques enables the reconstruction process to include complex models that require computational resources, such as the single scatter simulation techniques, AM reconstruction and facilitate MC scatter correction in list mode algorithms. Moreover, the inclusion of more robust models can reduce the impact of low statistics in the reconstruction process, which is crucial to reduce the dose of PET studies without loosing image quality.

3.4 Scatter Correction on Meshes

The correction of the scattering effect on mesh support can be done using the Single Scatter Simulation (SSS) algorithm presented in section 1.3.4. However the sampling strategy defined in equation 1.31 needs to be updated. When implementing the SSS equation on voxel support the algorithm takes advantage of the uniform sampling of the FoV to obtain an uniform sampling of scattered angles, by means of testing scattering

against each voxel in the Attenuation Map (AM) (summation of the U elements of the AM). In a mesh the sampling of the FoV is optimized to reduce the number of elements used, resulting in nodes assigned to areas with higher frequencies. A direct application of the summation over the U elements of the mesh results in an uneven sampling of the scattering angles ϕ. To correct this, a new sampling over the possible scattering angles must be defined, resulting in:

$$x_l^{\texttt{scatter}} = \sum_{u=1}^{U'} \frac{\sigma_{D_a,u}\sigma_{D_b,u}}{4\pi d_{D_a,u}^2 d_{D_b,u}^2} \cdot \frac{\mu_u}{\sigma_c(E_\gamma,\phi)} \cdot \frac{d\sigma(E_\gamma,\phi)}{d\Omega} \cdot [K^{D_a} + K^{D_b}], \tag{3.35}$$

where U' represents now a set of points in the AM which sample the scatter angles ϕ uniformly. The creation of such a set is done using a range of ϕ angles where the scattered event energy E'_γ, obtained using equation 1.25, falls within the energy window of the coincidence sorter of the scanner. Nevertheless, a given value of ϕ can be consequence of an event scattering at different positions of the FoV. The possible positions are defined by all the *scattering triangles* that can be formed using a base of length d_{D_a,D_b}, given by the position of the detectors, and the opposite angle defined by the complement of ϕ. Analytically they are obtained using the sine law:

$$k = \frac{d_{D_a,D_b}}{\sin(\pi-\phi)} = \frac{d_{D_a,\boldsymbol{r}^u}}{\sin(\gamma)} = \frac{d_{\boldsymbol{r}^u,D_n}}{\sin(\delta)}, \tag{3.36}$$

where $d_{D_a,\boldsymbol{r}^u}$ and $d_{\boldsymbol{r}^u,D_b}$ are the segments of the scattered LoR going from detector D_a to scatter point $\boldsymbol{r}^u$ and scatter point $\boldsymbol{r}^u$ to detector D_b respectively. Thus, using the relationship in equation 3.36 and operating, this segments can be defined in a 2D plane as:

$$(r_x^{D_a} - r_x^u)^2 + (r_y^{D_a} - r_y^u)^2 = d_{D_a,\boldsymbol{r}^u}^2 = (\sin(\beta)k)^2, \tag{3.37}$$

and

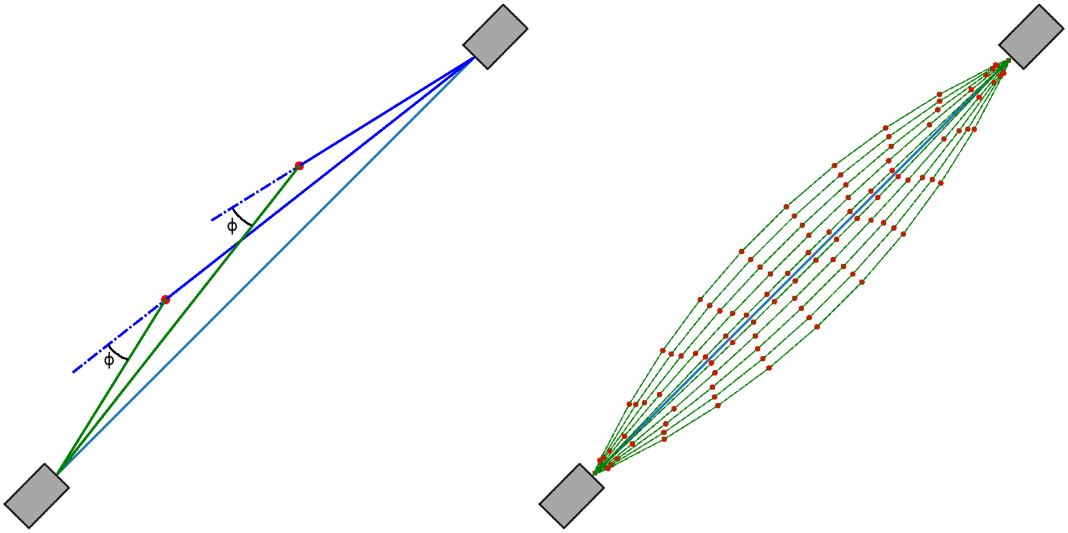
$$(r_x^u - r_x^{D_b})^2 + (r_y^u - r_y^{D_b})^2 = d_{\boldsymbol{r}^u,D_b}^2 = (\sin(\pi-\phi+\beta)k)^2, \tag{3.38}$$

The position of the scatter point is then obtained by choosing a value of $\beta \in [0,\phi)$ and solving:

$$(r_x^{D_a}-r_x^u)^2+(r_y^{D_a}-\sqrt{(k\,sen(\beta))^2-(r_x^u-r_x^{D_b})^2}-r_y^{D_b})^2-(k\,\sin(\pi-\phi+\beta))^2=0. \tag{3.39}$$

In figure 3.18(a) two scatter triangles are shown, each sharing the same angle ϕ. The collection of all possible positions of the scatter point for the same detected LoR (base segment) and scatter angle ϕ define a pair of arcs in 2D. In figure 3.18(b) the different arcs, formed by different scatter angles are shown.

This sampling method can be extended for 3D FoV easily by revolving the arc around the base segment. For a given point in the 2D arc a plane including the scatter point, normal to the base segment can be defined. In this plane a circumference is then defined with center in the base segment and radius defined by the scatter point



(a) (b)

Figure 3.18: Two *scatter triangles* formed by the same scatter angle ϕ (a) and the arcs formed by the collection of scatter points for five different ϕ values. The red dots represent the scattering position and the base segment (detected LoR), defined by the detectors, are presented in light blue. In (a) the blue line represents non-scattered LoR trajectory and in green the scattered LoR segment, the three segments (light blue, blue and green) form the triangle that define scattered LoR. In (b), the arcs formed by each scatter position are represented with thin green lines, and 10 possible scatter positions for each arc are presented with a red dot.

distance. Finally for each scatter point in the 2D plane one of such circumferences is built and sampled.

3.4.1 Verification of SSS algorithm on mesh support

The SSS algorithm, using the alternative scatter point sampling strategy, is now compared against a Monte Carlo simulation. To this end a modified NEC-R phantom was simulated in GATE software. The phantom consists of a polyethylene cylinder with a radius of 101.5 mm, with 3 perforations, first a rod of 1.6 mm filled with water, covered by 0.8 mm of low density polyethylene, placed at $(0.0, -45.0)$ mm. The second perforation is an empty rod of 10.0 mm radius placed at $(-20.0, 0.0)$ mm and the final perforation is a rod of 5.0 mm radius placed at $(30.0, 0.0)$ mm, filled with bone tissue. The simulated activity is placed within the water vial at $(0.0, -45.0)$ mm. The simulated phantom and activity images, along their corresponding meshes, can be seen in figure 3.19.

Using the GATE simulation 22×10^6 single scattered true coincidences are collected. The scattered coincidences are from a single slice of 5 mm thick. All coincidences are processed into a single 2D sinogram with 64 angle (θ) bins and 64 distance (ρ) bins (see section 1.3.1 for more information on the sinogram construction). The resulting sinogram can be seen in figure 3.20(a). Using the SSS method and the proposed sampling method, each of the $64 \times 64 = 4096$ LoRs defined by each of the sinogram bins is computed using 16 values of ϕ, ranging from the maximum possible ϕ value for the given energy window and 0, and 64 points per ϕ value, corresponding to a total of 1024 test points. The resulting SSS values for each bin are presented as a sinogram in figure 3.20(b). The obtained sinograms were normalized by their mean values and subtracted, resulting in the difference sinogram presented in figure 3.21. The MSE between each normalized sinogram was calculated and its value is $\texttt{MSE} = 0.249$. The differences between the reference sinogram and the SSS sinogram are within the expected error value. Such the error value does not present structures or artifacts that could derive from the mesh representation. The presented method permits to control the number of test points of the SSS algorithm and enables its application on mesh support. Moreover, such method permits to reduce the number of computations by means of using a more compact representation during the tracing of LoR segments.

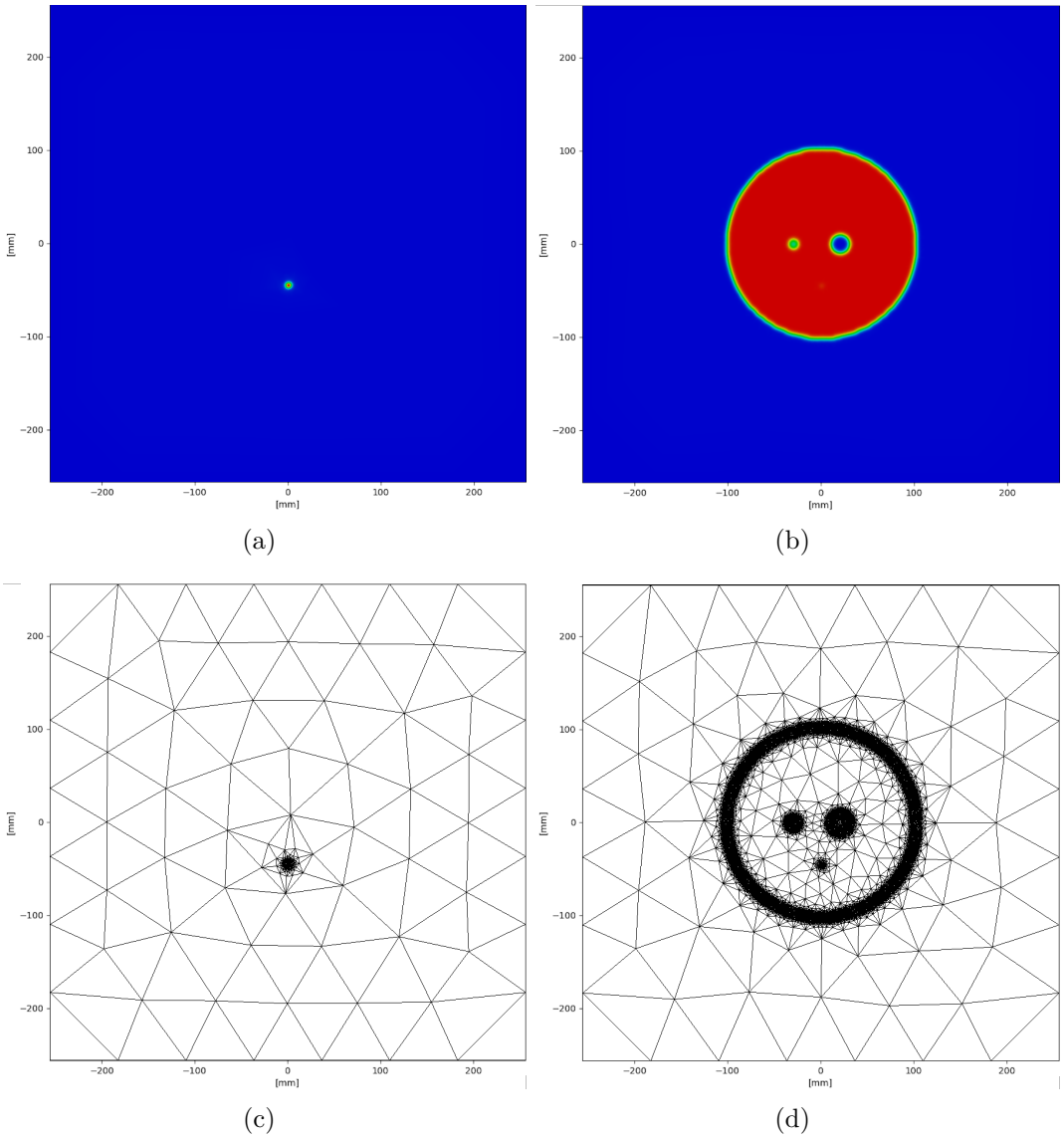
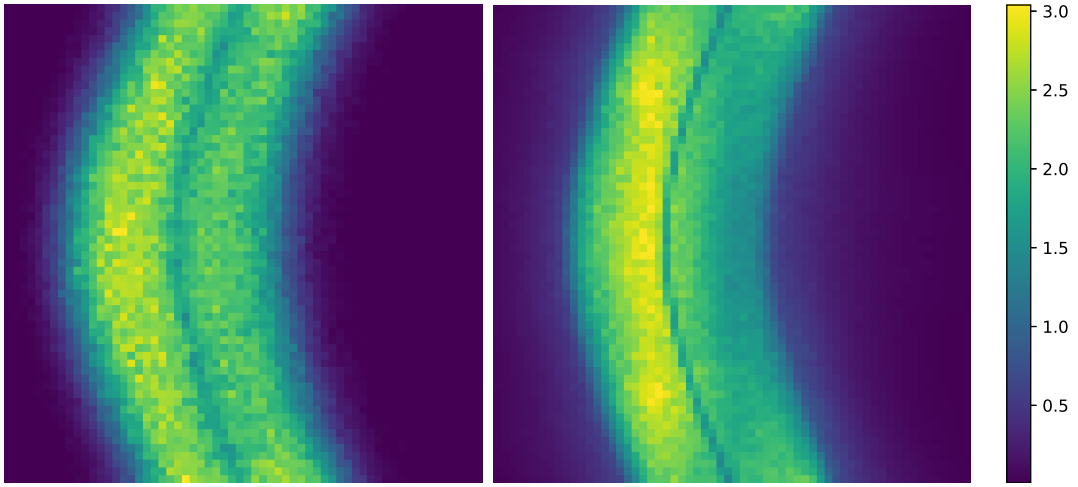


Figure 3.19: Activity distribution (a), attenuation structures (b) and their corresponding meshes (c,d) of the modified NEC-R phantom.



(a) (b)

Figure 3.20: Sinogram of single scattered true events obtained using GATE (a) and calculated SSS values for each LoR representing the sinogram (b).

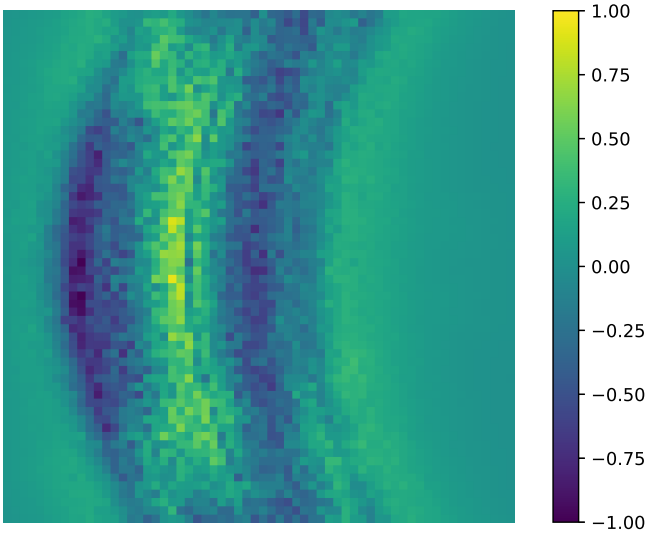


Figure 3.21: Difference sinogram between the GATE simulation of single scattered true events and SSS values, normalized by their mean values before subtraction.

3.5 Discussion and Remarks

In this chapter the mesh application to PET image reconstruction was presented. The chapter first presents the Delaunay–advancing-front method, used in the mesh construction of the PET image reconstruction procedure. Then the basis of PET construction using mesh is presented followed by a novel reconstruction procedure. The developed method shows that a series of 2D mesh is able to represent complex 3D tracer distributions. Such representation is not limited to 2D LoR projections and it is able to represent all 3D projections of a regular PET scanner (being only limited to fully transversal LoRs that are not feasible in existent PET scanners). Moreover the mesh based PET reconstruction algorithm adapts dynamically to the underlying geometry of the tracer, starting from a coarse mesh representation and adapting as the image details appear. The resulting representation is more compact and potentially requires less computation than a voxelized FoV to represent the same data. This representation means that less parameters are required and makes LoR projections less computationally expensive in those areas where no detail is needed. In the last section it was shown that the application of the Single Scatter Simulation algorithm is straightforward in mesh support. The application of SSS requires only to redefine the tested sample points in the FoV (restricted to voxel positions in voxelized format). Using a novel scatter point sampling method the mesh is able to compute the SSS approximation with user-defined granularity.

Here concludes the mesh techniques presented in this thesis. It was shown that is possible to express the PET reconstruction problem using less parameters without loosing image quality. Also the mesh representation, coupled with the possibility of applying the SSS algorithm, is a step towards the mutual reconstruction of tracer activity and Attenuation Map at lower computational costs. The final chapter of this thesis will focus in further reducing the complexity of the mutual activity and AM reconstruction by means of automatic generation of an AM prior.

4 Attenuation map generation using Deep Neural Networks

4.1 Introduction

The use of deep learning methods in image segmentation, classification and generation is a strong trend in several fields. With the presentation of AlexNet [147] in 2012, many problems started to be re-thought with some influence of Deep Neural Networks (DNN). This is reflected in the exponential increase of publications, across a wide variety of fields, which include keywords such as *machine learning* or *neural network*. The field of medical image analysis has increased the use of these novel tools. In this field the trend in designing problem-specific handcrafted features shifted to learned representations, exploiting the power of deep models [148]. The Convolutional Neural Networks (CNN) began to be the predominant technique in the medical imaging field in 2015, applying a large variety of models to different problems [149]. Recently the artificial generation of medical images began to be a noticeable part of the machine learning trend. In 2018 the medical image synthesys works reached 49% of the publications in the field [150]. The most of the modern synthetic image generation algorithms are based on the adversarial training of deep neural topologies, a technique presented in 2014 [151]. These generative algorithms have achieved impressive results in fields where large datasets are available, like face generation [152]. Although data is not as abundant in the field of medical imaging, the possible image representations in the field of medical imaging are also reduced when compared to other image domains such as face photographies. Also, exploiting the regularity of the medical images, the adversarial models can be successfully applied for different types of imaging modalities [153].

This chapter takes on the subject of DNN in medical imaging. It focus on the generation synthetic Attenuation Maps (AMs) to be used for correction of PET images. First, the most common Neural Network (NN) topologies are presented. These basic topologies make the base of the complex NN topologies used through the chapter. Then the specific topologies used in image analysis are shown, describing their working principles and focusing on 3D representations. After, the image generation topologies are derived, introducing the concepts of generative and adversarial models. Finally the AM generation for PET images is developed. This work is a fundamental step towards a transmission-free PET scan. Images generated with the methods presented in this chapter are intended to be used as a coarse estimation of the AM to directly correct the attenuation effect on a PET image or serve as a patient specific prior for mutual activity and AM reconstruction. Moreover, it will be shown, that the presented scheme

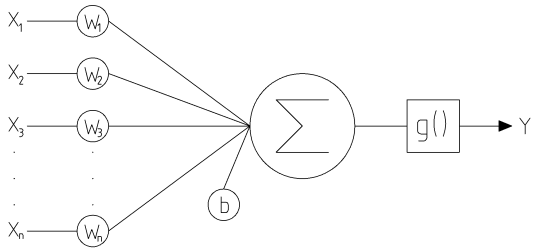


Figure 4.1: Single neuron diagram. In the left the inputs, labelled x_1 to x_n, each of them is multiplied by its input weight w_1 to w_n and then summed, along the bias b, at the $\sum$ node. Finally the activation function $g()$ is applied.

works in different patient, lesion and scanner data.

4.2 Basic Notions of Classic and Deep Neural Networks

Neural networks are function approximators based on stacking of a simple unit called *neuron*. These neurons are simple mathematical functions, loosely based on biological brain neurons. A neuron works by accumulating all these inputs, weighted by an activation multiplier and outputting a value based on this accumulation and an activation function. A single neuron is graphically represented in figure 4.1 and its mathematical expression can be written as:

$$y = g\left(\sum_{i=1}^{\boldsymbol{n}_h} w_i\, x_i + b\right) = g(\bar{W}^T\, \bar{X} + b), \tag{4.1}$$

where $\boldsymbol{n}_h$ is the number of connections, w_i are the connection weights, x_i are the connection inputs, b is the bias and $g()$ is the activation function. The activation function is used to add non linearity to the network. The most used functions are the hyperbolic tangent (**tanh()**), the sigmoid (`sigm()`), the Rectified Linear Unit (`ReLU()`) and the identity or linear output (no function is applied). The shape of these activations can be seen in figure 4.2.

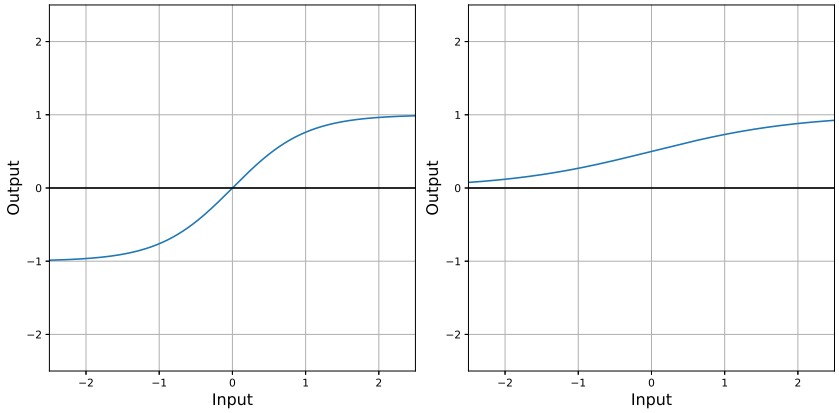
The normal procedure to obtain the values of the different weights and bias is by an optimization process called *training*. During this process the weight vector $\bar{W}$ and bias b are optimized to minimize a given cost function:

$$J(\hat{y}, y) = J(\bar{X}, \bar{W}, b, y), \tag{4.2}$$

where y is the expected output, x a given input and $\bar{W}$ and b the parameters of the NN. This subject is further developed in section 4.2.1.

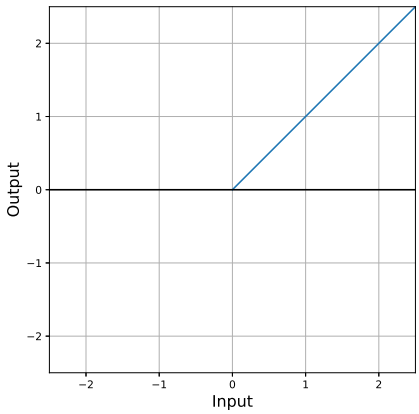
Shallow Networks

The power of a single neuron as an approximator is poor, being restricted only to linear functions and non-linearities equal to its activation. In terms of classification,

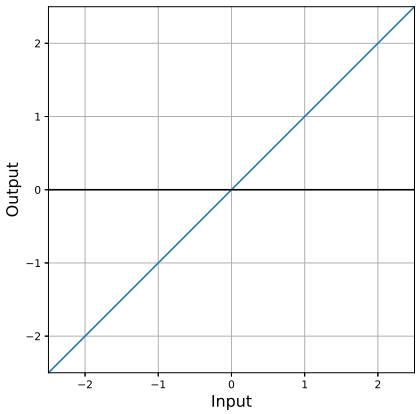


(a) tanh()

(b) `sigm`()



(c) `ReLU`()



(d) Linear

Figure 4.2: Common activation functions of neurons. The hyperbolic tangent (a), the sigmoid (b), the Rectifier Linear Unit (c) and the identity or linear output (d).

a single neuron can only draw a hyperplane to divide two clases. The power of NNs arise when several of these units are stacked in layers. The stacking is performed by using the output y of a neuron as the input x_i of an other, as shown in figure 4.3(a). This stacking form is called *feed-forward*, since the information flows from the previous layer output to the new layers inputs. The output of a neuron in a given layer is not connected to the input of a previous or current layer neuron (the information never flows back)[1]. Adding a single layer, results in several input neurons and one or more output neurons. It is expressed as:

$$y = g\left(\sum_{j=1}^{\boldsymbol{n}_h^{[1]}} w_j^{[1]}\, g\left(\sum_{i=1}^{\boldsymbol{n}_h^{[0]}} w_i^{[0]}\, x_i + b^{[0]}\right) + b^{[1]}\right), \tag{4.3}$$

where now the $\boldsymbol{n}_h^{[0]}$ represent the number of inputs (or layer 0) connections and $\boldsymbol{n}_h^{[1]}$ the number of neurons in the first layer, which are the inputs of the second layer. The topology can now draw complex functions and approximate convex regions. Nevertheless it is unable to create isolated regions, such as the ones required in an exclusive-or classification task. When an other neuron layer is added, creating a topology with an input layer, middle or hidden layer and output layer (shown in figure 4.3(b)) the topology is able to approximate a wide variety of continuous functions. Its form now responds to:

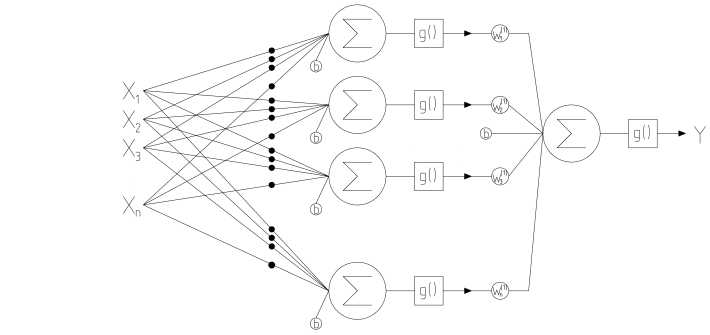
$$y = g\left(\sum_{k=1}^{\boldsymbol{n}_h^{[2]}} w_k^{[2]}\, g\left(\sum_{j=1}^{\boldsymbol{n}_h^{[1]}} w_j^{[1]}\, g\left(\sum_{i=1}^{\boldsymbol{n}_h^{[0]}} w_i^{[0]}\, x_i + b^{[0]}\right) + b^{[1]}\right) + b^{[2]}\right). \tag{4.4}$$

Networks with one hidden layer are regarded as universal approximators. This property is known as the universal approximation theorem [155–157], which states that given a large enough number of hidden neurons and non-polynomial squashing functions (such as **tanh()** or `sigm()`) activations, a NN with one hidden layer can approximate any $\mathcal{R}^n \to \mathcal{R}^n$ function with a desired non-zero error. However the theorem does not states how to determine the parameters of the network nor if such parameters can be learned. In praxis these networks are called *shallow* topologies and are not normally applied. This is due to infeasibility of computing extremely large hidden layers and the inability to learn each parameter correctly.

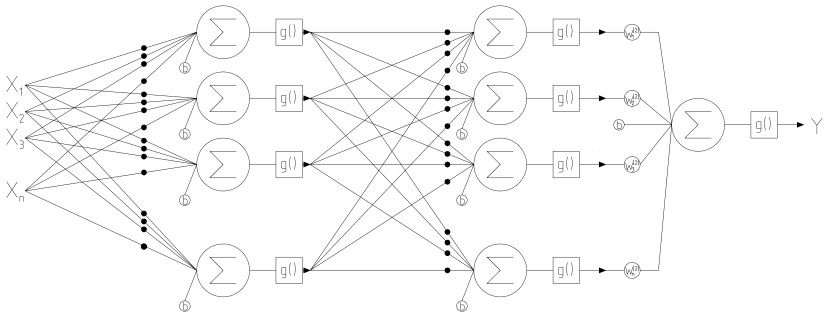
Deep Networks

An other type of feed-forward networks are the *deep networks*, which use many hidden layers. They have fewer number of parameters and generally they achieve lower generalization error than shallow networks. Moreover, it was shown that deep topologies with `ReLU()` activations can subdivide a number regions that is exponential with the

[1]The focus will be on feed-forward stacking, for other types of topologies the reader can refer to chapter 10 of [154]



(a)



(b)

Figure 4.3: Two layer fully connected neural network (a) and three layer fully connected neural network (b). Black dots represent the weights of each hidden neuron.

depth of the network [158]. Such subdivisions would require an exponential number of neurons in a shallow topology. Choosing deep topologies over shallow ones is also a design choice. Since the higher layers of a deep topology are compositions of lower layers, then by using a deep topology it is indirectly assumed that the problem is a composition of smaller problems (or that the function is a composition of simpler ones). However, such as in the case of shallow networks, the number of parameters and the number of layers of the network is not defined for a given task. In fact this is an active area of research and currently the size of a neural network is empirically defined.

Convolutional Networks

A particular type of NN is the Convolution Neural Network (CNN). These networks perform a convolutional operation between the neuron inputs and a kernel of given size. The convolutional operation is defined at element j as:

$$y = (X * K)(j) = \sum_{i=-\infty}^{\infty} x(i)\, K(j-i), \tag{4.5}$$

where K is the convolutional kernel. In practice the summation is finite and spans along the input samples. Also the convolution is in fact implemented as a cross correlation in many machine learning frameworks. Then, the output of a convolutional neuron for the input element j is defined as:

$$y(j) = g\left((X * K)(j) + b\right) = g\left(\sum_{i=-\boldsymbol{k}_x/2}^{\boldsymbol{k}_x/2} x(j+i)\, K(i) + b\right), \tag{4.6}$$

where $\boldsymbol{k}_x$ is the size of the convolutional kernel. Normally a layer of convolutional neurons share the same kernel (or weights). This way, the output of the layer can be regarded as the filtering of the input by a kernel. For this reason the number of neurons in the layer depend on the convolution mode:

- *Valid*: A valid convolution performs the convolution operation only on those elements j of the input which use valid elements of $\bar{X}$. The convolution starts in $j = \lfloor(\boldsymbol{k}_x/2)\rfloor$ and ends in $j = p_x - \lfloor(\boldsymbol{k}_x/2)\rfloor$, where p_x is the number of elements in the input vector $\bar{X}$. The resulting output has $2\lfloor(\boldsymbol{k}_x/2)\rfloor$ less elements than the input.

- *Same*: This convolution uses as many neurons as input elements p_x, resulting in an output vector of the same size as the input. To be able to perform the convolution on those elements j where the input is not fully defined (i.e. $x(p_x + 1)$), the operation uses an auxiliary value. There are several techniques to create this value:
  - *Constant padding*: This method fills the undefined value with zero. i.e. when padding 2 positions of the vector $[1, 2, 3]$ it results in $[\mathbf{0}, \mathbf{0}, 1, 2, 3, \mathbf{0}, \mathbf{0}]$
  - *Symmetric padding*: This technique repeats the last value and restarts. In this case the vector $[1, 2, 3]$ results in $[\mathbf{2}, \mathbf{1}, 1, 2, 3, \mathbf{3}, \mathbf{2}]$.

– *Reflective padding*: Here the last value is not repeated, the vector $[1, 2, 3]$ results in $[\mathbf{3}, \mathbf{2}, 1, 2, 3, \mathbf{2}, \mathbf{1}]$.

The convolutional neurons can be stacked as normal neurons and also be combined with them, creating NN with mixed neuron types. Since CNN kernels $K()$ can be shared among several layers, the convolutional layers use fewer parameters (weights) than normal neurons (also known as *Fully-Connected* layers). For this reason they are normally used in the first layers of a NN topology. These layers are most often applied in problems where it is expected that input posses shift invariant features, such as signals and images. The extension of these operations to higher dimensional inputs, such as images and volumes, is done by extending the kernel and convolution operations along the new dimension. The padding operation follows the same rule. This topic will be expanded in section 4.3.

4.2.1 Training Algorithms

The process of determining the values of each weight w_i and bias b in a NN is a complex and suboptimal process called *training*. The NN normally starts with random parameters drawn from an uniform distribution, i.e. the weights in layer i are initialized using an uniform distribution $U(-1/\sqrt{\boldsymbol{n}^{[i-1]}}, 1/\sqrt{\boldsymbol{n}^{[i-1]}})$, where $\boldsymbol{n}^{[i-1]}$ is the size of the previous layer. During the training process the network is presented with examples, containing an input and expected output. The network is expected to *learn* the mapping from the input to the expected output. In fact the process of training a NN is based on the minimization of a cost function defined by the parameters of the NN, the inputs and the outputs of a dataset:

$$\begin{aligned} \arg\min_{W} \quad & J(\hat{Y}, Y), \\ \text{with} \quad & \hat{Y} = f(X, Y, W) \end{aligned} \tag{4.7}$$

where X are the inputs of the dataset, Y the expected outputs and $\hat{Y}$ the outputs from the NN. For simplicity now the network parameter b is included in the parameter matrix W. The cost function $J()$ is a central part of every NN training procedure. Its shape is problem dependent, since it embeds information on how to judge the performance of the NN. Common shapes for $J()$ are:

- Mean Squared Error (MSE), for function approximation problems:

$$J(\hat{Y}, Y) = \frac{1}{N} \sum_{n=1}^{N} (\hat{y}_n - y_n)^2, \tag{4.8}$$

  where N is the number of samples in the train dataset.

- Categorical Cross-Entropy, for classification problems:

$$J(\hat{Y}, Y) = -\mathbf{E}_p[\log(q)] = -\frac{1}{N} \sum_{n=1}^{N} \sum_{m=1}^{N_c} y_{n,m} \log(\hat{y}_{n,m}), \tag{4.9}$$

where $\hat{y}_{n,m}$ and $y_{n,m}$ are now interpreted as the probabilities $\boldsymbol{q}_{n,m}$ and $\boldsymbol{p}_{n,m}$ respectively. They represent the probability $\boldsymbol{q}$ assigned by the NN to the n^{th} sample belonging to the target class m, subject to the real probability $\boldsymbol{p}$. Note that now $\sum_{m=1}^{N_c} \hat{y}_{n,m} = 1.0$.

The training dataset is a collection of inputs x_n with their corresponding expected outputs y_n. It should be sufficiently large and representative for the problem. There is no method to determine the size of a dataset or the types of samples it requires to ensure a low generalization error. Basic guidelines in this subject are that the larger the NN topology, the larger the dataset needs to be. Also if the number of features and the dimensionality of each sample x_n is large, the dataset needs to be larger.

The cost function is commonly optimized using the Gradient Descent (GD) method. This method is formulated as:

$$W^{t+1} = W^t - \epsilon \nabla_W J(X, Y, W^t), \tag{4.10}$$

where ϵ is the learning rate and $\nabla_W J()$ is the gradient of the cost function with respect to the NN parameters[2]. The equation 4.10 results in a new set of NN parameters W^{t+1} which will produce a lower cost value than previous values W^t. This method will improve the cost until it reaches a stationary point, where the gradient is zero in all parameter directions. Such point is a minimum, but it is not ensured to be a global minimum, nor an acceptable minimum. To avoid falling into local minima with a poor performance (local minima too far from the global minima) the NN training hyper-parameters are *tuned*. The hyper-parameters are all the parameters manually set, hence they are not learned. The fundamental hyper-parameter is ϵ, the learning rate, which must be carefully tuned. While a large ϵ avoids falling into local minima it also can lead to oscillations around a minimum. On the other hand, setting ϵ to an extremely low value can result into overly large training times. However GD is normally applied using one of its many variants which seek to improve its speed and accuracy while reducing the hyper-parameter sensibility. Probably the most widespread variation is the Stochastic Gradient Descent (SGD). It is based on the division of the dataset into (mini) batches. This is due both to a computational constrain and a generalization advantage. Large datasets used to train NN become too large to be completely processed in a single training step, instead, each training step takes a batch of randomly selected samples from the dataset to calculate the gradient:

$$W^{t+1} = W^t - \epsilon \frac{1}{N_m} \nabla_W \sum_{n=1}^{N_m} J(x_n, y_n, W^t), \tag{4.11}$$

where N_m is the size of the randomly sampled batch from the dataset. The size of the sample introduces sampling error of the mean of $\sigma/\sqrt{N_m}$ but, for small enough learning rates, it provides regularization and convergence speed [159].

[2]Normally this derivative is found using automatic differentiation, a fundamental tool of the rapid growth of NN methods.

The second modification of the GD, that is also implemented in SGD, is the addition of momentum to the optimization step. The momentum in the optimization step helps to overcome noisy gradients or small local minima. I works by adding a new variable v to the optimization algorithm which plays the role of the step direction and speed. The momentum in the learning rate is then the resistance of the learning step to change its direction, as it were a mass moving though the parameters space:

$$W^{t+1} = W^t - v^t. \tag{4.12}$$

The velocity is then calculated as an Exponentially Weighted Moving Average (EWMA) of the gradient:

$$v^t = \beta v^{t-1} - \epsilon \frac{1}{N_m} \nabla_W \sum_{n=1}^{N_m} J(x_n, y_n, W^t), \tag{4.13}$$

where $\beta \in (0, 1]$ is the decay rate of the velocity, the larger is its value, the heavier are previous gradients in the step direction of the current step. Also if successive gradients point in the same direction, they will accumulate and produce larger learning rates. On the other hand if the gradients are noisy or not aligned, they will cancel out and produce smaller learning rates as in response to the uncertainty in the direction of the minimum.

Another of the most popular extensions to the SDG is the use of parameter dependent learning rates. This modification is based on the notion that different parameters of the NN require different learning rates and may benefit from adapting the moments of each parameter in a separate fashion. Two of the mos popular optimizing algorithms for NN use combinations of these concepts, they are Root Mean Square Propagation (*RMSProp*) and Adaptive Moment estimation (*ADAM*).

RMSProp

The RMSProp algorithm [154] adapts the learning rate of each parameter w_i according to an exponentially weighted moving average. For each parameter in the NN the RMSProp inversely scales the applied learning rate by these historical values. The optimization algorithm is described as:

ADAM

The ADAM [109] is a combination of RMSProp with momentum. ADAM adds a first order momentum and second order momentum terms with bias correction, resulting in a robust optimization algorithm:

The creation of training algorithms for NN is an active research field, there exist many other training procedures. However they are outside the scope of this thesis, the reader may refer to chapter 8 of [154] for more information on the subject.

Algorithm 3: RMSProp training algorithm.

input: learning rate (ϵ), decay rate (β), initial parameters (W)
$\delta = 1e-8$
$r = 0$
while *not* `convergence_check()` **do**
  Sample mini batch of size N_m from dataset.
  Get gradient: $g = \nabla_W \sum_{n=1}^{N_m} J(x_n, y_n, W^t)$
  Accumulate in moving average: $r = \beta r + (1-\beta)\, g \odot g$
  Get update (parameter-wise): $\Delta_W = \frac{\epsilon}{\sqrt{\delta + r}} \odot g$
  Update: $W^{t+1} = W^t - \Delta_W$
  $t+ = 1$
end

Algorithm 4: ADAM training algorithm.

input: learning rate (ϵ), first moment decay rate (β_1), second moment decay rate (β_2), initial parameters (W)
$\delta = 1e-8$
$r = 0$
$s = 0$
while *not* `convergence_check()` **do**
  Sample mini batch of size N_m from dataset.
  Get gradient: $g = \nabla_W \sum_{n=1}^{N_m} J(x_i, y_i, W^t)$
  Get biased first moment: $s = \beta_1 s + (1-\beta_1) g$
  Get biased second moment: $r = \beta_2 r + (1-\beta_2)\, g \odot g$
  Get un-biased first moment: $\hat{s} = \frac{s}{(1-\beta_1^T)}$
  Get un-biased second moment: $\hat{r} = \frac{r}{(1-\beta_2^T)}$
  Get update (parameter-wise): $\Delta_W = \epsilon \frac{\hat{s}}{\sqrt{\hat{r} + \delta}}$
  Update: $W^{t+1} = W^t - \Delta_W$
  $t+ = 1$
end

Batch Normalization

A common technique to help the convergence of DCNs is the batch normalization technique [160]. Batch normalization reduces the NN sensitivity to hyper-parameters by reducing how changes in a given layer affect the following layers. Normally in a NN when one changes the weight of a given neuron all the proceeding neurons are affected by this change. Also, changing the weights in a given layer, changes the distribution of the outputs of the layer, which is the input distribution of the following layers. This effect is called *internal covariance-shift*. This shift can be so large that it can vanish the gradient or make certain neurons saturate, if using activations such as sigmoid. Part of this is solved by means of ReLU (non-saturating) activations and careful initialization of the weights. However, it is more efficient to seek to preserve some properties in the distribution of each layer output. This promotes a more efficient training and does not requires to re-adjust the layer to changes in its input distribution.
The batch normalization strategy normalizes the output $X = [x^{(1)}, x^{(2)}, ..., x^{(j)}]$ of each dimension in the feature space:

$$\hat{x}^{(j)} = \frac{x^{(j)} - \mathbf{E}[x^{(j)}]}{\sigma(x^{(j)})}. \tag{4.14}$$

And then it is linearly transformed using:

$$y^{(j)} = \hat{x}^{(j)}\, \psi_1^{(j)} + \psi_2^{(j)}, \tag{4.15}$$

where $\psi_1^{(j)}$ and $\psi_2^{(j)}$ are two new parameters[3] that make explicit the normalization to the optimization algorithm. In other words, these parameters will be explicitly learned by the network to account for the representation loss of the normalization in equation 4.14. During training the expected value and standard deviation of $x^{(j)}$ is computed using mini-batch statistics. During production, the mean and standard deviation are no longer calculated from mini-batch statistics (since there is no mini-batch at all). Instead the whole training dataset mean and deviation are used. These statistics are calculated using running means and variance calculation during training. The result is a normalization procedure that can be applied to any NN that accept mini-batch training and takes the form of a simple normalization and linear transformation.

4.3 Image analysis using convolutional neural networks

The problem of computer vision has spawned a whole sub-branch in the field of NN. The tasks of segmentation and clasification of images have created their own topologies, focused in these types of problems. Image analysis with NN specializes in extracting information from 2D image data, normally encoded in raster (pixel) format. The topologies are created to extract features from large inputs with high local correlation. The most successful topology is based on the first stage of the brain's visual cortex, where certain neurons are exited only by distinct patterns [161]. This behaviour can

[3] Actually two per feature in the output vector of the layer

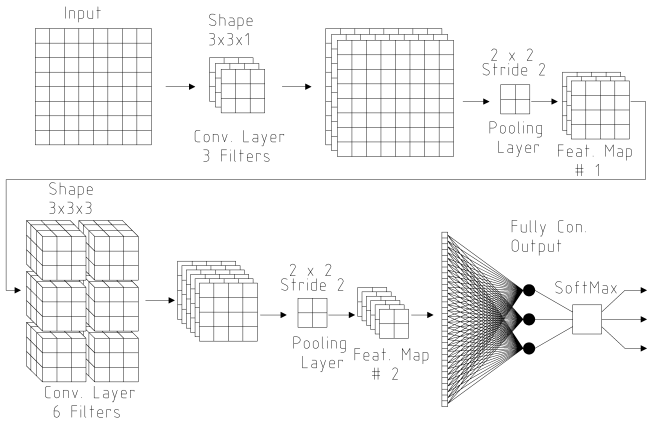


Figure 4.4: Basic convolutional neural network topology for 2D image classification. The input is a 2D raster image which is processed by two convolutional and pooling layers. The convolutional layers have a kernel of size $3 \times 3\mathtt{px}^2$. These layers reduce the network size and expand the receptive field of deeper layers. The output is a flatten operation and a fully connected layer with 3 output neurons and a softmax layer.

be reflected mathematically by the convolution or cross-correlation operation, which applied in convolutional neurons. An other property of the visual cortex is that it posses more complex detection mechanisms that are invariant to light intensity and position of the pattern created in previous neurons. This translated, mathematically, in the use of pooling layers within the NN topology. The poling layers are composed of special neurons that take small regions of the input and produce a single value. This output value normally responds to the mean or the maximum value of the input. The result is a reduction of the input size, which gather and compress information from multiple outputs in the lower layers. This knowledge lead to the creation of the basic image processing NN which combines convolutional layers with pooling layers [162]. This topology is known today as the Convolutional Neural Network (CNN), the basic building block of many modern image analysis topologies. The topology of the CNN can be seen in figure 4.4, where the network output is first processed by a series of convolutional neurons (also known as filters) and then the pooling operation reduces its size by half by taking the mean or maximum value in the filter outputs, using 2×2 pixel patches. This filtering-pooling operation is repeated until the output is small enough to be processed by a Fully Connected (FC) layer.

Classification Task

The processing operations inside a CNN are more easily explained using an example. Given a classification topology for 5 objective classes, as the one presented in figure 4.5, with 5 valid convolutional and pooling layers, a single fully connected layer (dense layer) and a final soft-max operation. The input of this network is a single channel image of size (256×256 $\mathtt{px}^2$), the first convolutional layer applies 8 different filters of size 3×3 $\mathtt{px}^2$, resulting in a feature tensor of size ($256 \times 256 \times 8$). These, and

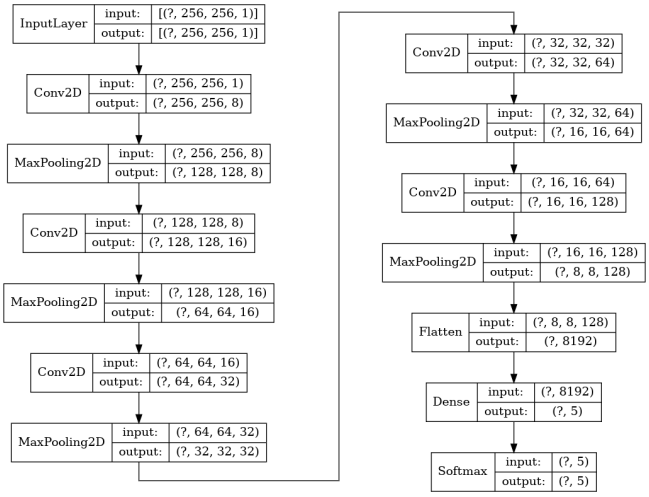


Figure 4.5: Convolutional neural network architecture used for image classification, containing 5 convolutional layers with max-pooling operations, a 5 neuron dense layer and a softmax layer as output. Each layer is represented by its name, input dimension and output dimension. The plot is based on the Keras toolkit plot function. The interrogation sign refers to the *batch* dimension used during training.

all convolution operations in the model, use *same* convolutions with zero-padding. The convolutional layer adds an extra dimension of size 8. Then this tensor is fed to the pooling operation, in this case a max pooling with size (2×2). The max-pooling operation applies the operation $y(i) = \max([\text{ft}(i-1, i-1), \text{ft}(i, i-1), \text{ft}(i-1, i), \text{ft}(i, i)])$, where ft() is the value of the feature tensor at a given coordinate. The pooling operation is non-overlapping, meaning that the stride of i is equal to the size of the pooling patch, in this case the stride is $i = 2$. The feature vector after the pooling operation has a size of $(128 \times 128 \times 8)$, no pooling operation is performed along the last dimension of the tensor, since each slice correspond to a different filter output. Then the feature tensor is applied to the second convolutional layer (second Conv2D block of figure 4.5), which implements 16 filters. In this case the input is a tensor of size $(128 \times 128 \times 8)$, this means that the applied filter must have an additional dimension, to capture information from all filters. The filter size in this case will be $(3 \times 3 \times 8)$. As in the first convolutional layer, the output of the layer will copy the size of the first two dimensions and create a third dimension for every applied filter, resulting in a tensor of size $(128 \times 128 \times 16)$. Once again the max pooling layer will reduce the feature map, creating a tensor of size $(64 \times 64 \times 16)$. This process is repeated for the last three convolutional and pooling layers, resulting in a tensors of size $(32 \times 32 \times 32)$, $(16 \times 16 \times 64)$ and finally $(8 \times 8 \times 128)$. This last feature tensor is flattened, the feature tensor dimension is lost and it transforms into a vector of shape $8192 = (8 \times 8 \times 128)$. This vector is processed by the FC layer and outputs a tensor of 5 values. Finally the soft-max operation

creating a normalized option of its values by:

$$x_n = \frac{\exp(x_n)}{\sum_{j=1}^{5} \exp(x_j)}. \tag{4.16}$$

This last layer of the network is applied in order to obtain an output compatible with a statistical loss such as the cross-entropy, commonly used in classification problems. With this configuration the network takes an image of size 256×256 $\mathtt{px}^2$ and outputs a Probability Density Function (PDF) of the five objective classes for each input. The same network using a shallow topology would have required $256 \times 256 \times 5 + 5 = 327685$ parameters, while the CNN requires 139205, this is 2.35 times less parameters.

Segmentation

The other main task of computer vision is object segmentation. This task seeks to classify and locate different objects or structures in an image. The output of the segmentation is not a probability density but a label map indicating the class of each pixel in an image, for example separate different types of tissues (bone, grey/white matter, tumorous, etc.) in an MRI image. For this task one of the most successful topologies in the U-Net [163]. This NN takes its name from the shape of its topology, which can be seen in figure 4.6. It implements a CNN to reduce the input image into a smaller feature tensor and then adds a second path which gradually dilates the feature tensor back in to the initial resolution while incorporating feature tensors from the initial convolutional layers through skip-connections. The topology presented in figure 4.6 takes as input a single channel image of size $(256 \times 256 \times 1)$ and has three resolution levels. Each level applies convolutional layers with 8, 16 and 32 filters respectively followed be a pooling operation of size 2×2. The feature tensor at the bottom of the topology will have size $(32 \times 32 \times 32)$. The process is the same as the one described for the classification task, except that no FC layers are applied. Also, instead of mean or max-pooling operations, a convolution operation with stride equal to two pixels is used. This stride-2 convolution operation can be seen as a weighted average whose weights are learned. At the bottom, the U-Net starts to dilate the feature tensor to recover an image. This process is performed using two operations, a convolutional layer and a up-sampling layer. The convolutional layers work just as the ones used in the downward path, and they are used to extract features. The main difference is the replacement of the stride-2 convolution operation with an up-sampling operation. This up-sampling operation can be implemented using a linear upsampling or a transposed convolution. The U-Net implements the transposed convolution, a convolution procedure which recovers the shape of the tensor before being convolved (since the values cannot be recovered). The transposed convolution first pads the image into the output shape by including zeros between the image elements and then applies the filter. The result after each up-sampling operation is a weighted-upsampled tensor. It can be seen in figure 4.6 that after each up-sampling, the deep features are concatenated (linked together in the feature dimension) with the features from the previous path. This is a copy-and-concatenate operation that seeks to fusion

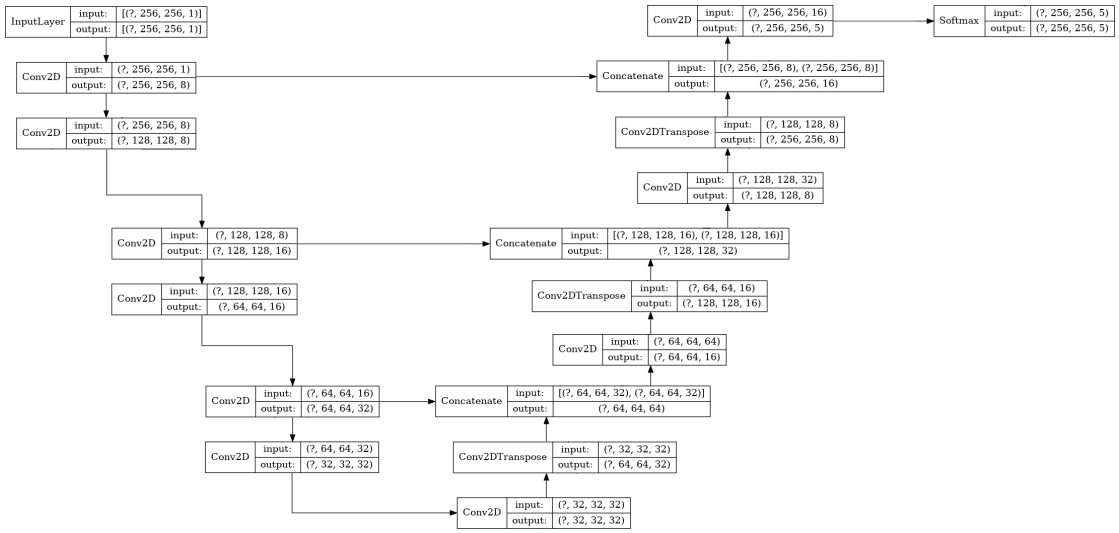


Figure 4.6: Example of a U-Net image segmentation network topology. This NN is composed of 3 convolutional layers in the downward (left) path and 3 convolutional layers in the upward (right) path. Each convolutional layer in the downward path is followed by a stride-2 convolution to perform downsampling and each convolutional layer in the upward path is followed by a transpose convolution with stride-2 to perform upsampling. The output is a convolutional filter with kernel (1×1) and as many layers as output classes, followed by a pixel-wise softmax layer. Each layer is represented by its name, input dimension and output dimension. The plot is based on the Keras toolkit plot function. The interrogation sign refers to the *batch* dimension used during training.

the deep feature, containing information from a broader receptive field with more localized features from narrower receptive fields. The concatenation is done in the last dimension, to keep different features in different channels. If each convolution in the upward path has a size of 32, 16 and 8 respectively, after the first convolution and up-sampling the feature tensor has a size of $(64 \times 64 \times 32)$. This is concatenated with the last feature tensor from the downward path, generating a compound feature tensor of size $(64 \times 64 \times 64)$. This compound tensor is processed by the next convolutional layer in the upward path, creating a new tensor of $(64 \times 64 \times 16)$. Once again up-sampled to $(128 \times 128 \times 16)$ and concatenated with the tensor of the same size from the downward path, resulting in a tensor of size $(128 \times 128 \times 32)$. This operation is repeated until the last upward layer which will have a size of $(256 \times 256 \times 8)$ and is concatenated with the first feature tensor, resulting in high resolution feature tensor of shape $(256 \times 256 \times 16)$. At this point the final layer is applied, which applies a kernel of size (1×1) and as many filters as objective classes of the segmentation map. This will result in an output tensor of size $(256 \times 256 \times N_c)$, where $N_c = 5$ is the number of classes. Normally a pixel-wise soft-max is applied to transform each $(1 \times 1 \times N_c)$ tensor into a PDF of the objective classes.

Network topologies for 3D images

The presented NN topologies and functions were applied to 2D image processing problems. Nevertheless the medical image analysis field, and specifically the tomographic image analysis, is concerned with 3D or even higher dimensional spaces. The additional dimensions greatly affect the performance of NNs, even when growing from 2D to 3D. First, the data size grows linearly with the additional dimension, requiring more computational resources to perform the same operations. The memory growth is critical in NN processing, which is normally executed on Graphic Processing Units (GPUs) with limited memory (for each element in the additional dimension the maximum batch size is reduced by the same amount). The additional dimension also results in an exponential growth of the possible combinations in the input vector. This translates in increased feature space and thus requiring higher network capacity.

Extensions of the presented NN topologies for 3D inputs are straight forward. They require only to extend the 2D convolution, pooling, up-sampling or transposed convolution in the additional dimension. While a 3D volume can be analysed using a 2D architecture in a slice-by-slice fashion, they ignore correlation between different slices in the volumetric data. The tomographic images like PET, CT and MRI present such correlation between neighbouring voxels, since they are representations of physical volumes. For this reason the 3D NN are more effective than their 2D implementations since they can create filters that capture volumetric features. A direct application of this concept is the 3D U-Net [164] derived from the original U-Net which replaces all 2D convolutional layers with 3D convolutional ones and was tested in the volumetric segmentation of Xenopus kidney. A similar work is the V-Net [165], which also uses 3D convolutional layers and was applied on prostate segmentation task. The success of these previous studies show the fitness of 3D DCN for segmentation in volumetric data.

The analysis of 3D volume data with neural networks is currently limited for lack data and computational cost. However 3D image analysis is an emerging trend outside the medical field. In the recent years the interest in analysing 2D images and 3D volumes for creation of volumetric depth maps has grown [105, 166, 167]. Also the widespread use of 3D-scanners, which gather data in the form of point clouds, has ignited a trend to analyse data directly in this format [168–171]. This has lead to the creation of specific tools to manipulate 3D data [172].

4.4 Deep generative models

So far the presented NN models were used for classification or function approximation tasks. An other interesting application of neural networks is their capacity to learn features from data. This task is actually a sub-product of the presented models. In every model the input is processed and condensed into a lower dimensional representation that should retain all the relevant information of the problem under consideration (input information). Generative models are NN topologies designed to explicitly exploit this feature. They are normally trained to learn representations and then operate in this lower dimensional mapping of the information.

4.4.1 Autoencoders

One of the simplest generative models is the Autoencoder (AE). It is a bottleneck topology which explicitly aims to compress and then recover the input, passing all the information through a low dimensional bottleneck tensor. Its operation can be expressed as:

$$\hat{x} = \texttt{Dec}(\texttt{Enc}(x)), \tag{4.17}$$

where $\texttt{Enc}()$ is a NN called *encoder* and $\texttt{Dec}()$ is the *decoder* NN. The encoder takes the input x and processes it into a reduced feature space, then the decoder uses only the information contained in the reduced feature space to recreate the input. The AE learns to create an internal representation of the input in a lower dimensional space by minimizing a given loss $J(x, \hat{x})$.
The success of AE can be interpreted as the NN learning a mapping of the data. Suppose that the AE is fed with human skeleton volumes, it is known that the human skeleton has typical shapes, pieces and relative locations. This means that the probability that each voxel will be filled or not is not independent from the others. Also, the possible modes of the volume is restricted to those that are recognizable as human skeletons. Thus it is possible to imagine the set of all possible human skeletons as a manifold in the n-dimensional space of the voxels within the volume $\boldsymbol{V}$. Such manifold could then be mapped into a lower dimensional feature space $\boldsymbol{F}$. Then, the AE task is to learn the encoding map $\texttt{Enc}() : \boldsymbol{V} \rightarrow \boldsymbol{F}$ and decoding projection $\texttt{Dec}() : \boldsymbol{F} \rightarrow \boldsymbol{V}$. A NN topology that can be used to implement such AE is presented in figure 4.7.

This AE is similar to the U-net topology, but removing the skip connections. The downward branch is the encoder $\texttt{Enc}()$, composed of convolutional layers and pooling

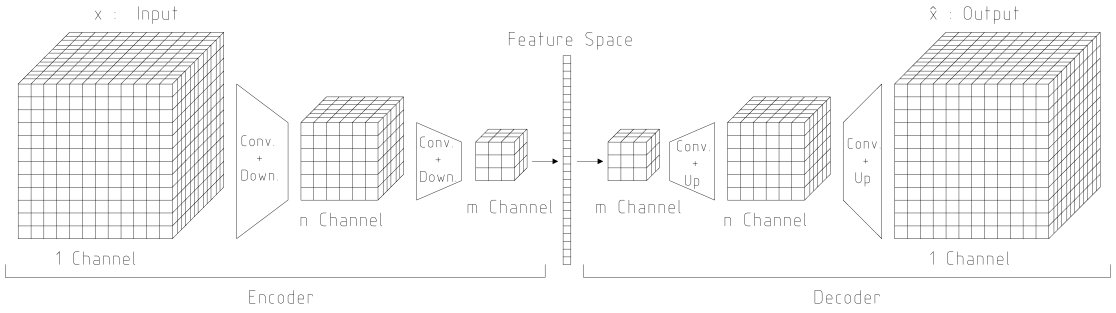


Figure 4.7: Basic autoencoder topology for 3D volumes. The input x is processed by a series of convolutional and downsampling layers until reaching the target feature space size, after the decoder applies convolutional and upsampling layers to obtain an output $\hat{x} \approx x$.

operations. The upward path is the decoder $\texttt{Dec}()$, composed of convolutional layers and up-sampling operations. The size at the bottom layer defines the dimensionality of the feature space $\boldsymbol{F}$.
While a direct application of the AE as formulated in equation 4.18 is limited, the concepts that it applies can be exploited for a wide variety of tasks. The AE were applied in medical imaging to the problem of image de-noising [173], where the input x is corrupted with noise and fed to the NN:

$$\hat{x} = \texttt{Dec}(\texttt{Enc}(\tilde{x})), \tag{4.18}$$

where $\tilde{x}$ is a corrupted version of x. This AE is trained minimizing the loss $J(x, \hat{x})$, just as a basic AE. The result is a NN that can be fed with noisy data to retrieve the free noise version, that is unknown. An other interesting application of AE is to use them for feature extraction, in [174] the resulting feature mapping of a trained AE is used to penalize segmentation structures. The idea behind [174] is that if an AE is trained on the objective data of a given problem then its feature space contains a mapping of the objective manifold. Thus, when training an other NN, the objective and produced samples can be mapped using the trained encoder and measure their distance in the lower dimensional mapping. This not only improves the overall quality but also heavily penalizes shapes that are not part of the objective manifold which is hard to reflect using metrics in the manifold space [175].

4.4.2 Variational Autoencoders

From the AE yet an other concept rises. Idealy, given a trained AE, the feature space $\boldsymbol{F}$ could be sampled to obtain novel samples of the objective class. This basic idea is the fundament of many deep generative topologies. Nevertheless to successfully sample the feature space some constrains should be introduced in the generation of the feature space, such as continuity and compactness. The insight of turning the latent space into a compact distribution is what turns AEs into Variational Autoencoders (VAEs),

and was presented in [176]. The intuitive way to think of VAEs is an AE where now the latent space is diffuse. In other words, an input x_n does not correspond to a single point in z_n in the feature (or latent) space, but to a distribution. Then, the encoder now outputs the parameters of this distribution instead of a single point. If the distribution is chosen to be a Gaussian, then the encoder will calculate $\texttt{Enc}(x) = (\mu_z, \sigma_z)$ and the decoder will do $\texttt{Dec}(z) = \hat{x}$, with $z \sim \mathcal{N}(\mu_z, \sigma_z)$, thus forcing continuity in the latent space and encoding uncertainty in the process.

More formally, the central task of a generative model is to learn a distribution. The objective distribution is often intractable (for the majority of interesting generative models at least). Suppose that given an input x it is expected to find the distribution on the latent space (the task of the encoder). The conditional distribution of the latent variable z given the input x is given by:

$$\boldsymbol{p}(z|x) = \int \frac{\boldsymbol{p}(x|z)\,\boldsymbol{p}(z)}{\boldsymbol{p}(x)} dz, \tag{4.19}$$

where $\boldsymbol{p}(z)$ is the prior distribution of the latent space, $\boldsymbol{p}(x)$ the input distribution and $\boldsymbol{p}(x|z)$ the likelihood of x given the latent space. The equation 4.19 presents several problems for training generative models, first it requires to compute the complete feature space z and second it requires to know the input distribution $\boldsymbol{p}(x)$ which is often unknown. To circumvent these problems the variational inference proposes to use another, known, distribution to approximate this posterior. Generative models as VAEs, model this distribution using a neural network. Then the VAEs seek to minimize the distance between the unknown posterior $\boldsymbol{p}(z|x)$ with $\boldsymbol{q}_W(z|x)$, a distribution parameterized by the weight matrix W. A way to measure the difference between the real posterior and the parameterized posterior is the Kullback-Leibler (KL) divergence:

$$D_{\texttt{KL}}(\boldsymbol{q}_W(z|x)||\boldsymbol{p}(z|x)) = -\int \boldsymbol{q}_W(z|x) \log \frac{\boldsymbol{p}(z|x)}{\boldsymbol{q}_W(z|x)} dz. \tag{4.20}$$

Equation 4.20 still contains the unknown posterior $\boldsymbol{p}(z|x)$. However, this is then circumvented using the second part of the VAE NN, the decoder. The decoder now is interpreted as a parameterized likelihood function $\boldsymbol{p}_W(x|z)$ that given a point z in the latent space recovers the probability density in the input (or target) manifold. Using Bayes theorem it can be expressed that:

$$\boldsymbol{p}(z|z) = \frac{\boldsymbol{p}_W(x|z)\boldsymbol{p}(z)}{\boldsymbol{p}(x)}. \tag{4.21}$$

Replacing equation 4.21 in equation 4.20:

$$D_{\texttt{KL}}(\boldsymbol{q}_W(z|x)||\boldsymbol{p}(z|x)) = -\int \boldsymbol{q}_W(z|x) \log \left(\frac{\boldsymbol{p}_W(x|z)\boldsymbol{p}(z)}{\boldsymbol{q}_W(z|x)\boldsymbol{p}(x)} \right) dz. \tag{4.22}$$

Applying logarithm properties and distribution of the integral, it takes the form:

$$D_{\mathtt{KL}}(\boldsymbol{q}_W(z|x)||\boldsymbol{p}(z|x)) = -\int \boldsymbol{q}_W(z|x)\log\left(\frac{\boldsymbol{p}_W(x|z)\boldsymbol{p}(z)}{\boldsymbol{q}_W(z|x)}\right)dz + \int \boldsymbol{q}_W(z|x)\log(\boldsymbol{p}(x))dz, \tag{4.23}$$

where the second term in the right hand of the equation can be simplified further, since it integrates to unity:

$$D_{\mathtt{KL}}(\boldsymbol{q}_W(z|x)||\boldsymbol{p}(z|x)) = -\int \boldsymbol{q}_W(z|x)\log\left(\frac{\boldsymbol{p}_W(x|z)\boldsymbol{p}(z)}{\boldsymbol{q}_W(z|x)}\right)dz + \log(\boldsymbol{p}(x)). \tag{4.24}$$

Also, remembering that by definition $D_{\mathtt{KL}} \geq 0$, it is possible to rearrange the expression into an inequality:

$$\log(\boldsymbol{p}(x)) \geq \int \boldsymbol{q}_W(z|x)\log\left(\frac{\boldsymbol{p}_W(x|z)\boldsymbol{p}(z)}{\boldsymbol{q}_W(z|x)}\right)dz. \tag{4.25}$$

Once again, using logarithm properties the likelihood term $\boldsymbol{p}_W(x|z)$ is removed:

$$\log(\boldsymbol{p}(x)) \geq \int \boldsymbol{q}_W(z|x)\log\left(\frac{\boldsymbol{p}(z)}{\boldsymbol{q}_W(z|x)}\right)dz + \int \boldsymbol{q}_W(z|x)\log\boldsymbol{p}_W(x|z)dz, \tag{4.26}$$

in which two terms can be defined:

$$\log(\boldsymbol{p}(x)) \geq D_{\mathtt{KL}}(\boldsymbol{q}_W(z|x)||\boldsymbol{p}(z)) + \mathbf{E}_{\sim \boldsymbol{q}_W(z|x)}(\log\boldsymbol{p}_W(x|z)). \tag{4.27}$$

Equation 4.27 is known as the Evidence Lower Bound (ELBO) or variational lower bound, since it bounds the log probability $\log(\boldsymbol{p}(x))$ of the data. Thus maximizing the ELBO maximizes the probability of the data, which is the objective of a generative model. In the ELBO two main parts are distinguishable. The first part is the KL divergence of the latent prior $\boldsymbol{p}(z)$ and the parameterized posterior distribution $\boldsymbol{q}_W(z|x)$. This means that it is possible to select a distribution, and hence control the continuity of the latent space. The second part is the expected outcome of the logarithm of the decoder, $\log\boldsymbol{p}_W(x|z)$, given the encoding distribution $\boldsymbol{q}_W(z|x)$. This term is also known as the *reconstruction term*, since it relates the output of the decoder with the input of the encoder. While it is possible to choose any form for the latent prior and the parameterized posterior, not every distribution shields a closed form. One of the most common choices for these distributions are Gaussians:

$$\boldsymbol{p}(z) = \frac{1}{\sqrt{2\pi\sigma_p^2}}\exp\left(-\frac{(x-\mu_p)^2}{2\sigma_p^2}\right), \tag{4.28}$$

$$\boldsymbol{p}_W(z|x) = \frac{1}{\sqrt{2\pi\sigma_q^2}}\exp\left(-\frac{(x-\mu_q)^2}{2\sigma_q^2}\right). \tag{4.29}$$

Replacing in the KL divergence term of equation 4.27:

$$D_{\mathtt{KL}}(\boldsymbol{q}_W(z|x)||\boldsymbol{p}(z)) = \int \frac{1}{\sqrt{2\pi\sigma_q^2}} \exp\left(-\frac{(x-\mu_q)^2}{2\sigma_q^2}\right) \log\left(\frac{\frac{1}{\sqrt{2\pi\sigma_p^2}} \exp\left(-\frac{(x-\mu_p)^2}{2\sigma_p^2}\right)}{\frac{1}{\sqrt{2\pi\sigma_q^2}} \exp\left(-\frac{(x-\mu_q)^2}{2\sigma_q^2}\right)}\right) dz. \tag{4.30}$$

Selecting $\mu_p = 0$ and $\sigma_p = 1$ and operating on equation 4.30 shields:

$$D_{\mathtt{KL}}(\boldsymbol{q}_W(z|x)||\boldsymbol{p}(z)) = \frac{1}{2}\left[1 + \log(\sigma_q^2) - \sigma_q^2 - \mu_q^2\right]. \tag{4.31}$$

Equation 4.31 is then a metric of similarity between the learned posterior and the selected prior (a centered Gaussian with unity deviation), for a single dimension in the latent space. Normally the latent space is n-dimensional, thus the equation 4.27, for the particular case of the centered Gaussian is written as:

$$\log(\boldsymbol{p}(x)) \geq \frac{1}{2}\sum_{q=1}^{N_q}\left[1 + \log(\sigma_q^2) - \sigma_q^2 - \mu_q^2\right] + \mathbf{E}_{\sim q_W(z|x)}(\log p_W(x|z)), \tag{4.32}$$

where N_q is the number of dimensions of the latent space. Note that all distributions, for any posterior, are pushed against the same distribution, thus making the posterior compact and continuous. Now the second term remains disconnected from the maximization objective. It can be seen that the second term is related to the latent posterior distribution $\boldsymbol{q}_W(z|x)$, and to decode a sample x_n in the manifold it is required to sample form the posterior distribution $\boldsymbol{q}_W(z|x_n)$. To allow the gradient to flow from the generated sample $\hat{x_n}$ to the input sample x_n it is required to sample the latent posterior in a way that the distribution parameters μ_q and σ_q, obtained from application of the variational encoder, are explicit. This called *re-parametrization trick*, and it is simply to sample q_W using a batch of samples:

$$z_m = \mu_q + \sigma_q e_m, \tag{4.33}$$

where e_m is a random sample from $\mathcal{N}(0,1)$, with size N_R. Finally the VAE loss of the n^{th} sample can be expressed as:

$$\mathcal{L}_n = -\frac{1}{2}\sum_{q=1}^{N_q}\left[1 + \log(\sigma_q^2) - \sigma_q^2 - \mu_q^2\right] - \frac{1}{N_R}\sum_{m=1}^{N_R}\mathbf{E}_{\sim \boldsymbol{q}_W(z|x)}(\log \boldsymbol{p}_W(x_n|z_m)), \tag{4.34}$$

where the term $\mathbf{E}_{\sim \boldsymbol{q}_W(z|x)}(\log \boldsymbol{p}_W(x_n|z_m))$ is a reconstruction loss between $\hat{x_n}$ and x_n, normally euclidean distance or cross-entropy.

Currently VAEs are one of the most successful models to learn representation of the data. It can be applied to enhance the accuracy of many AE applications where the features space needs to be sampled or when metrics are applied in the feature space. However they are not currently the best models to produce high quality samples, they were recently relegated by the *adversarial models*.

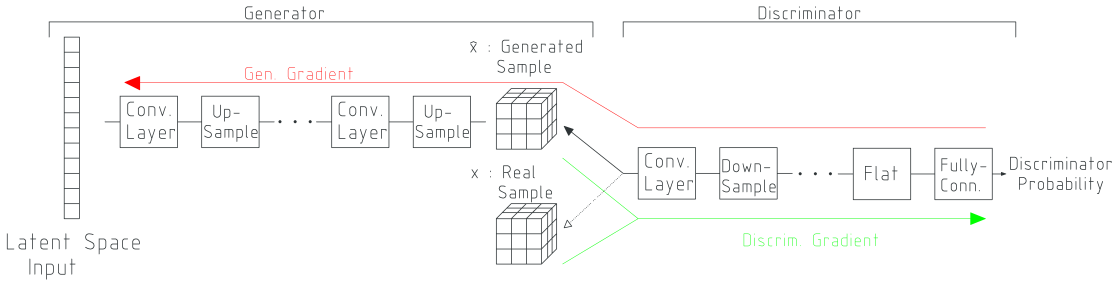


Figure 4.8: GAN topology for volume generation. The input of the generator is a latent vector which is transformed into a volume $\hat{x}$ by means of a convolutional NN. The discriminator is fed with real x or generated $\hat{x}$ samples and seeks to identify each class. The gradient of the generator network propagates through the generator, as illustrated by the red arrow. The gradient of the discriminator flows only through the discriminator network, from the generated and real sample, as shown by the green arrow.

4.4.3 Adversarial Networks

Adversarial networks are a special type of generative models where two neural networks compete with each other, one tries to generate samples from an objective manifold while the other tries to detect generated samples from real samples drawn from the objective manifold. These networks are called *generator* and *discriminator*, respectivelly. The generator can be think as a decoder from a AE which tries to learn $\boldsymbol{p}(x)$ sampling from n-dimensional noise, such as a normal distribution $z \sim \mathcal{N}(0, 1)$, instead of using the encoder output. Therefore the generator is seeking to learn the data distribution given the latent space:

$$\boldsymbol{p}(x|z) = \int \frac{\boldsymbol{p}(z|x)\,\boldsymbol{p}(x)}{\boldsymbol{p}(z)} dx. \tag{4.35}$$

This situation is similar to the one presented in equation 4.19. This setup requires to calculate the probability over all the output space, and it also requires to know the data probability distribution $\boldsymbol{p}(x)$, hence making it intractable. Instead of trying to learn the explicit distribution $\boldsymbol{p}(x|z)$ using a likelihood metric, an implicit model is used, defined by the generator:

$$\hat{x} = \texttt{Gen}_W(z), \tag{4.36}$$

where $\texttt{Gen}_W()$ is the parametrized generator, a NN. To be able to learn $\texttt{Gen}_W()$ without knowledge of $\boldsymbol{p}(x)$, a second NN can be proposed, $\texttt{Disc}_W(x)$ that learns to discriminate samples from $\boldsymbol{p}(x)$ from other samples. Then the generator $\texttt{Gen}_W()$ and discriminator $\texttt{Disc}_W()$ could be trained in the same data but in opposite task. This is known as a Generative Adversarial Network (GAN) [151]. The concept is that both the generator network $\texttt{Gen}_W()$ and the discriminator network $\texttt{Disc}_W()$ are trained at the same time in an antagonist manner. The generator seeks to generate data that copy $\boldsymbol{p}(x)$, sampling a prior distribution $\boldsymbol{p}_z(z)$ and the discriminator learns how to distinguish generated data $\hat{x}$ from real data x. The GAN network training scheme is shown in figure 4.8.

The generator loss is built using the cross-entropy measure between $\boldsymbol{p}(x)$ and $\texttt{Disc}_W(x)$, and since the input can be real or fake (generated), the discriminator output is a binary (or Bernoulli) loss:

$$H(\boldsymbol{p}, \texttt{Disc}_W) = -\frac{1}{N_m^{true}} \sum_{n=1}^{N_m^{true}} \boldsymbol{p}(1|true)\ \log(\texttt{Disc}_W(x_n)) \\ - \frac{1}{N_m^{fake}} \sum_{n=1}^{N_m^{fake}} [1 - \boldsymbol{p}(1|fake)]\ \log(1 - \texttt{Disc}_W(\texttt{Gen}_W(z))), \quad (4.37)$$

where N_m is the number of samples within a training batch. Knowing that $\boldsymbol{p}(1|true) = 1$ and $\boldsymbol{p}(1|fake) = 0$ it is possible to simplify the equation 4.37:

$$J_d = -\frac{1}{N_s^{true}} \sum_{n=1}^{N_s^{true}} \log(\texttt{Disc}_W(x_n)) - \frac{1}{N_s^{fake}} \sum_{n=1}^{N_s^{fake}} \log(1 - \texttt{Disc}_W(\texttt{Gen}_W(z))). \quad (4.38)$$

The generator adversarial cost should maximize the opposite of the discriminator, thus the training would require a min-max optimization. The function to maximize would be:

$$\mathcal{G}_g^{theo.} = \frac{1}{N_m^{true}} \sum_{n=1}^{N_m^{true}} \log(\texttt{Disc}_W(x_n)) + \frac{1}{N_m^{fake}} \sum_{n=1}^{N_m^{fake}} \log(1 - \texttt{Disc}_W(\texttt{Gen}_W(z))). \quad (4.39)$$

But since no gradient coming from the samples x_n flows through the generator NN the first term can be dropped. An other problem of the loss presented in equation 4.39 is that the discriminator task is most of the time easier to optimize than the generator part. Then, the discriminator will provide low probability to fake images at the beginning of the training. These low probabilities mean a low value of $\texttt{Disc}_W(\texttt{Gen}_W(z))$ which in turn will generate a gradient close to zero [151]. To circumvent this problem, instead of maximizing equation 4.39, the heuristic loss presented in equation 4.40 is minimized:

$$J_g = -\frac{1}{N_m^{true}} \sum_{n=1}^{N_m^{true}} log(\texttt{Disc}_W(\texttt{Gen}_W(z))). \quad (4.40)$$

The generator and discriminator networks are trained using equation 4.38 and 4.40 respectively. However since the discriminator should be a good proxy of $\boldsymbol{p}(x)$ it must be trained near the optimum for each train step of the generator. Otherwise the discriminator would easily fool the discriminator and no informative gradient will flow. The training loop of a GAN is shown in algorithm 5.

A successfully trained GAN is a state of the art generative model, creating sharp representations in complex problems such as medical image generation [16]. However they require delicate tuning to successfully converge. While a low trained discriminator generates bad quality gradients that are non-informative, and thus the generator cannot learn $\boldsymbol{p}(x)$, an exceedingly trained discriminator would give the generated samples $\hat{X}$ a too low score, and hence no significant gradient would flow to the generator.

Algorithm 5: GAN training algorithm.

input: data (X), discriminator train steps per generator step (S_d), generator train method ($\mathcal{T}_g$), discriminator train method ($\mathcal{T}_d$), batch size N_m.
$t_d = 0$
$t_g = 0$
while *not* ***convergence_check()*** **do**
  $\texttt{step\_discrim.} = 0$
  while ***step_discrim.*** $< S_d$ **do**
    Sample mini batch X of size N_m from dataset.
    Sample mini batch Z of size N_m from latent space.
    Generate fake mini batch: $\hat{X} = \texttt{Gen}_W^{t_g}(Z)$
    optimize discriminator: $\texttt{Disc}_W^{t_d+1} \leftarrow \mathcal{T}_d(\texttt{Disc}_W^{t_d}, J_d(X, \hat{X}))$
    $\texttt{step\_discrim.} += 1$
    $t_d += 1$
  end
  Sample mini batch Z of size N_m from latent space.
  Generate fake mini batch: $\hat{X} = \texttt{Gen}_W^{t_g}(Z)$
  Optimize generator: $\texttt{Gen}_W^{t_g+1} \leftarrow \mathcal{T}_g(\texttt{Gen}_W^{t_g}, J_g(\hat{X}, \texttt{Disc}_W^{t_d}))$
  $t_g += 1$
end

Yet an other problem is that the generator could ignore the input prior z and produce a single valid data point $\hat{x}$, this is known as *model collapse*, since the generator is able to fool the discriminator but it is not able to generate any data except the collapsed mode. These problems are enhanced when higher dimensional data, such as 3D volumes are processes, probably due to the relative reduction of the model capacity and the asymmetry in the task difficulties between the generator and discriminator.

Wasserstein GAN

An interesting improvement of the basic GAN method is the Wasserstein Generative Adversarial Network (WGAN) [177]. WGANs propose to modify the discriminator and turn it into a *critic* in order to reduce the impact of unbalanced training in GANs, which produces an uninformative gradient from the discriminator. Intuitively, the critic is a discriminator that instead of producing a saturated *real-fake* probability as output, produces a continuous linear metric of the difference between the real $\boldsymbol{p}(x)$ and fake $\texttt{Gen}_W(z)$ distributions. Such metric is the Wasserstein or Earth-Movers distance:

$$\mathcal{W}(\boldsymbol{p}(x), \texttt{Gen}_W(z)) = \inf_{\Psi \in \Pi(\boldsymbol{p}(x), \texttt{Gen}_W(z))} \mathbf{E}_{(x,y)\sim\Psi}[\|x - y\|], \tag{4.41}$$

where $\Pi(\boldsymbol{p}(x), \texttt{Gen}_W(z))$ is the set of all joint distributions $\Psi(x, y)$, whose marginals are $\boldsymbol{p}(x)$ and $\texttt{Gen}_W(z)$. The result can be interpreted as the cost of moving the mass

from one distribution to an other, using an optimal transport plan. However, since the infimum in equation 4.41 is intractable , the Kantorovich-Rubinstein dual is used:

$$\mathcal{W}(\boldsymbol{p}(x), \texttt{Gen}_W(z)) = \sup_{\|f\|_L<1} \mathbf{E}_{x\sim\boldsymbol{p}(x)}[f(x)] - \mathbf{E}_{\hat{x}\sim\texttt{Gen}_W(z)}[f(\hat{x})], \tag{4.42}$$

where f is a 1-Lipschitz function [178], meaning it has gradients with norm at most equal to 1 everywhere. The function f can be parameterized using a NN, this network is the critic $\texttt{Crit}_W(x)$. Hence the problem can be rewritten as a maximization:

$$\mathcal{G}_c = \max_W \mathbf{E}_{x\sim\boldsymbol{p}(x)}[\texttt{Crit}_W(x)] - \mathbf{E}_{z\sim\boldsymbol{p}_z(z)}[\texttt{Crit}_W(\texttt{Gen}_W(z))], \tag{4.43}$$

which can be finally expressed in the same terms as equation 4.38, of the traditional GAN:

$$J_c = \frac{1}{N_m^{true}} \sum_{n=1}^{N_m^{true}} \texttt{Crit}_W(x_n) - \frac{1}{N_m^{fake}} \sum_{n=1}^{N_m^{fake}} \texttt{Crit}_W(\texttt{Gen}_W(z)). \tag{4.44}$$

Nevertheless the critic $\texttt{Crit}_W$ should be constrained to remain a 1-Lipschitz function. In the original formulation this is done through weight clipping, forcing the elements of W to remain in the interval $[-k_c, k_c]$ (normally $k_c = 0.01$). However this is not a good way to ensure it to be k-Lipschitz, with k dependent on k_c. A better way to impose the k-Lipschitz is by constraining the norm of the gradient, as proposed in [179]. This is done by adding a penalization term to equation 4.44:

$$J_c^{\texttt{GP}} = J_c + k_{\texttt{GP}} \frac{1}{N_m^{true}} \sum_{n=1}^{N_m^{true}} \mathbf{E}_{\tilde{x}\sim\boldsymbol{p}(\tilde{x}_n)}[(\|\nabla_{\tilde{x}_n}\texttt{Crit}_W(\tilde{x}_n)\|_2 - 1)^2], \tag{4.45}$$

where $k_{\texttt{GP}}$ is a coupling coeficient (normally $k_{\texttt{GP}} = 10$) and $\boldsymbol{p}(\tilde{x})$ is an implicit defined distribution to avoid the intractability that arises if using $\boldsymbol{p}(x)$. This distribution is defined as an uniform distribution between a sample from the dataset x and a generated sample $\hat{x}$. This is an approximation that works empirically and motivated by the notion that an optimal critic has gradient norm equal to one in the stright lines connecting coupled points from $\boldsymbol{p}(x)$ and $\texttt{Gen}_W(z)$ [179]. Using this setup inhibits the use of batch normalization techniques, since now the gradient penalty is enforced sample by sample. Finally a training loop for the WGAN with gradient penalty constrain is shown in algorithm 6.

Progressive Growing GAN

A final extension to the WGAN with gradient penalty is the Progressive Growing GAN (ProGAN) [152]. This framework is proposed as an additional step towards a stable training of GANs and is especially useful in cases of large dimensionality of the objective distribution, such as 3D medical images. These generative NNs are based on the previous WGAN and are trained in a scalable fashion. The scheme of this NN is shown in figure 4.9. The strategy is to train a small generative model using scaled versions of the latent and output space. These scaled spaces are built

Algorithm 6: Wasserstein GAN training algorithm with gradient penalty constrain.

input: data (X), critic train steps per generator step (S_c), generator train method ($\mathcal{T}_g$), critic train method ($\mathcal{T}_c$), batch size N_m, dimensionality of the objective space V, coupling coefficient $k_{\texttt{GP}}$.
$t_c = 0$
$t_g = 0$
while *not* `convergence_check()` **do**
  `step_critic` $= 0$
  while `step_discrim.` $< S_c$ **do**
    Sample mini batch X of size N_m from dataset.
    Sample mini batch Z of size N_m from latent space.
    Generate fake mini batch: $\hat{X} = \texttt{Gen}_W^{t_g}(Z)$
    $n = 0$
    while $n < N_m$ **do**
      Sample random vector $\boldsymbol{V}$ of size V from $U(0, 1)$.
      Create weighted intermediate sample: $\tilde{x}^{(n)} = \boldsymbol{V}\, x^{(n)} + (1 - \boldsymbol{V})\, \hat{x}^{(n)}$
      Calculate gradient penalty: $\nabla_{\texttt{GP}} = (\|\nabla_{\tilde{x}^{(i)}} \texttt{Crit}_W(\tilde{x}^{(n)})\|_2 - 1)^2$
      Calculate sample loss: $\mathcal{L}_c^{\texttt{GP},(n)} = \mathcal{L}_c(x^{(n)}, \hat{x}^{(n)}) + k_{\texttt{GP}} \nabla_{\texttt{GP}}$
      $n+ = 1$
    end
    optimize critic: $\texttt{Crit}_W^{t_c+1} \leftarrow \mathcal{T}_c(\texttt{Crit}_W^{t_c}, J_c^{\texttt{GP}})$
    `step_critic`$+ = 1$
    $t_c+ = 1$
  end
  Sample mini batch Z of size N_s from latent space.
  Generate fake mini batch: $\hat{X} = \texttt{Gen}_W^{t_g}(Z)$
  Optimize discriminator: $\texttt{Gen}_W^{t_g+1} \leftarrow \mathcal{T}_g(\texttt{Gen}_W^{t_g}, J_g(\hat{X}, \texttt{Crit}_W^{t_d}))$
  $t_g+ = 1$
end

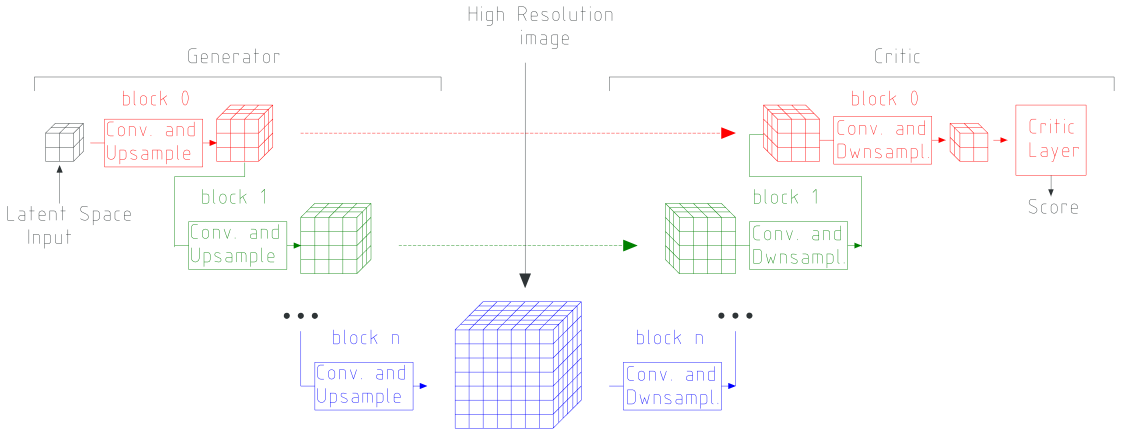


Figure 4.9: Progressive growing GAN topology for 3D volume generation. Each resolution block is shown in a different color. The NN starts with the `block` 0, shown in red, during the initial training the generator resolution `block` 0 is connected directly with the resolution `block` 0 of the critic (through the dashed line). Once this resolution level is trained this connection is eliminated and the resolution `block` 1 is connected (shown in green). This process continues until the resolution `block` n is added which shields the desired resolution.

using interpolation on higher dimensional spaces. Training in lower dimensional spaces is much more stable since fewer modes are possible. Each block is initially trained following like a normal GAN model (i.e. following algorithm 6). After the block is fully trained a new block is added. The new block is slowly faded into the NN topology using a special training stage. This *fade-in* training step performs a weighted sum of the new block output and an upsampling of the lower resolution block. This is done to reduce the impact of the newly added block in the total gradient, thus preventing the network to quickly *forget* the learned weights. This fade-in step is described in algorithm 7.

Using this technique higher dimensional generative models are easier to train, reaching excellent results in 1024×1024 $\texttt{px}^2$ images and high quality $64 \times 64 \times 64$ $\texttt{px}^3$ structures in vertebral structures generation [180].

Conditional GAN

The generative models presented up to this point focused on the generation of random points in the objective manifold. These models are normally used to generate new samples from a distribution (e.g. generating music with VAEs [181] or faces of non-existant people with GANs [182]). In medical image processing this can be used, e.g. to generate new samples for physician training [183], however the generation of specific patient data should not be random. This is the case of Attenuation Map (AM) generation for PET image correction. Here the expected generated map must be subject to the patient anatomy. This information can be given to the generative model by replacing the random input $z \sim \boldsymbol{p}_z(z)$ with data drawn from a known distribution

Algorithm 7: Progressive Growing GAN, fade-in of higher dimensional block.

input: data (X), critic train steps per generator step (S_c), generator train method ($\mathcal{T}_g$), critic train method ($\mathcal{T}_c$), batch size (N_m), low dimensionality trained generator and critic models ($\texttt{Gen}_W^{(ld)}$, $\texttt{Crit}_W^{(ld)}$), higher dimensionality models to be faded in ($\texttt{Gen}_W^{(hd)}$, $\texttt{Crit}_W^{(hd)}$), fade-in steps (S_f).
$s = 0$
$t_c = 0$
$t_g = 0$
$k_{\texttt{fade}} = 0$
while $s < S_f$ **do**
  Update fade-in weight: $k_{\texttt{fade}} = s/(S_f - 1)$
  $\texttt{step\_critic} = 0$
  while $\textit{step\_critic} < S_c$ **do**
    Sample mini batch X of size N_m from dataset.
    Sample mini batch Z of size N_m from latent space.
    Generate fake mini batch using low dim. (ld) model: $\hat{X}^{(ld)} = \texttt{Gen}_W^{t_g,(ld)}(Z)$
    Generate fake mini batch using full dim. (fd) model: $\hat{X}^{(fd)} = \texttt{Gen}_W^{t_g,(ld)}(Z)$
    Create faded fake mini batch: $\hat{X} = (1 - k_{\texttt{fade}})\hat{X} + k_{\texttt{fade}}\,\hat{X}^{(ld)}$
    Calculate critic cost: $J_c^{\texttt{GP}}$
    optimize critic: $\texttt{Crit}_W^{t_c+1} \leftarrow \mathcal{T}_c(\texttt{Crit}_W^{t_c}, J_c^{\texttt{GP}})$
    $\texttt{step\_critic} += 1$
    $t_c += 1$
  end
  Sample mini batch Z of size N_m from latent space.
  Generate fake mini batch using low dim. (ld) model: $\hat{X}^{(ld)} = \texttt{Gen}_W^{t_g,(ld)}(Z)$
  Generate fake mini batch using full dim. (fd) model: $\hat{X}^{(fd)} = \texttt{Gen}_W^{t_g,(ld)}(Z)$
  Create faded fake mini batch: $\hat{X} = (1 - k_{\texttt{fade}})\hat{X}^{(fd)} + k_{\texttt{fade}}\,\hat{X}^{(ld)}$
  Optimize discriminator: $\texttt{Gen}_W^{t_g+1} \leftarrow \mathcal{T}_g(\texttt{Gen}_W^{t_g}, J_g(\hat{X}, \texttt{Crit}_W^{t_d}))$
  $t_g += 1$
  $n += 1$
end

or dataset. This is the case of conditional generative models, where the generated sample is conditioned by a known input. A special case of the conditional generative models, and of special importance in this work, is the image-to-image translation. In this case the input of the model is an image in certain modality and the output is the same image in an other modality. A generative model trained this way performs a domain translation task. The output quality of such model will not only depend on the available data, but also in the shared information between the modes to be translated.

The following sections will focus on this problem, in the specific case of NAC PET to CT translation, showing the capabilities and necessity of generative models in these tasks. The models presented so far will be used to obtain a patient specific artificial AM for PET image correction, using as input the information available in NAC PET images (PET images reconstructed with emission data only) and CT scans as objective modality.

4.5 Deep neural networks for Attenuation Map generation

In this section the implementation of AM generation methods using NN for PET image correction is discussed. Specifically, the task of image translation between NAC PET and CT is developed. This is achieved using several of the presented techniques in their fully 3D implementations. Since the generated AMs are intended to serve as a high-quality prior to other mutual attenuation and activity reconstruction process or to directly correct attenuation effects in reconstruction, the focus will be set on whole-body images. This image translation task is especially difficult in whole-body images, since the NAC PET presents incomplete anatomical information. It will be shown that in this scenario, where the translation process also needs to fill information blanks, the NN and specifically generative models, are specially powerful.

The application of GANs in image to image translation tasks has been successfully exploited in many medical imaging domains, including PET attenuation map synthesizing. However, most methods of attenuation map generation analyze the MRI to CT translation using convolutional networks [49] and GANs with paired [153] and unpaired data [184], requiring a co-registered MRI image wich contains anatomical information that is not present in NAC-PET images. The PET (and NAC-PET) to Computed Tomography (CT) image transalation remains as one of the less explored domains, specially in whole body scans [185]. The studies in this particular domain focus on PET-CT image translation of corrected images on head scans. Liu [13] proposes the use of a 2D U-Net architecture to translate NAC-PET head scans to CT, showing promising results in head region scans. Armanious [153] proposes a general GAN application composed of a cascaded 2D U-Net generator and a discriminator used to evaluate the perceptual loss and style of the generated image. They show the capability of the topology to translate PET scans to CT, using only axial slices and, again, only for head region scans. Both methods provide no information on their

capability for whole-body image translation which is a harder problem to solve, given that the possible modes in the attenuation structures is larger. The whole-body image translation was explored by Dong [14] using a GAN trained with cycle consistency and by Armanious [15] using a 2D GAN based on a cascaded U-Net, however they trained and tested their studies only on state-of-the-art PET scanners with Time of Flight (ToF) cappabilities. The reconstruction of the attenuation map was also studied using maximum-likelihood reconstruction of attenuation and activity (MLAA) methods [51], showing promise in PET scanners with ToF capabilities only when combined with neural networks [186]. However without using a post-processing step MLAA outputs noisy attenuation maps, with or without ToF information [187]. Solving the problem of generating an attenuation map directly from image data can enable non-ToF scanners to use synthetic attenuation maps.

In this chapter several NN topologies are proposed to achieve a NAC PET to AM image translation. The models are trained on a public available data-set from the Cancer Image Archive [188], the data-set contains series of registered CT, PET and NAC-PET whole body scans of Head and Neck Scamorous Cell Carcinoma (HNSCC) [189], acquired with Discovery ST/STE/RX General Electric scanners. For the testing process 8 different datasets are used, containing 5 different types of carcinomas and multiple scanners models: Discovery ST/STE/RX/LS/IQ/610/690, from General Electric and Biograph from Siemens. Only the Discovery 690 possess ToF capabilities. The models presented in this chapter along with the database and pre-processing pipeline are constructed using the Keras [190] framework with TensorFlow 2.0 [111] back-end and are publicly available[4].

4.5.1 Generation of the dataset

As in every machine learning problem, a dataset is needed in order to adjust the model variables and to be able perform inferences. The objective of the dataset is to train and test different AM generation topologies, also this dataset should contain matched samples of the source distribution (NAC PET images) and the target distribution (AC images or CT images). The different image modalities within each dataset sample should be co-registered to enable voxel based accuracy metrics. Finally the dataset should be as large as possible and contain samples from several expected body sizes, patient positions (i.e. arms up and down), observed lesions and scanners models. Finding such dataset, publicly available is difficult, however, The Cancer Image Archive (TCIA) [188] contains a large database that matches these requirements.

Training and Validation Dataset

One of the largest datasets in TCIA, containing NAC PET and CT data, is the Head and Neck Scamorous Cell Carcinoma (HNSCC) [189] dataset. The complete HNSCC dataset is composed of 215 patients who underwent basal or upper body scans using PET-CT scanners. The images were acquired at *MD Anderson Cancer Center* from

[4]https://github.com/RawthiL/PET_DeepAttCorrection

2003 to 2013 and include 765 studies, but not all of them fit the criteria needed to be part of a training dataset. The original purpose of the dataset is the patient-specific planning and delivery of radiotherapy, a primary determinant of head and neck cancer outcomes. A major drawback is that it only contains HNSCC lesions. Nevertheless since the lesions it contains are not massive or produce large captation within the body, they proved to be useful for the training procedure. The dataset, after pre-processing, resulted in 118 images from 71 different patients.

Testing Datasets

A separate dataset is used for testing and it is also created using TCIA. It is built using several public datasets, since no other large dataset as the HNSCC was found. The selected datasets include different types of lesions, patients and scanner technologies. They compose a total of 133 test samples, whose origins are:

- Non-Small Cell Lung Cancer (NSCLC) [191], 73 samples.
- The Cancer Genome Atlas - Head-Neck Squamous Cell Carcinoma (TCGA-HNSC) [192], 25 samples.
- The Cancer Genome Atlas Lung Adenocarcinoma (TCGA-LUAD) [193], 20 samples.
- The Cancer Genome Atlas - Thyroid Cancer (TCGA-THCA) [194], 1 sample.
- Clinical Proteomic Tumor Analysis Consortium - Lung Adenocarcinoma (CPTAC-LUAD) [195], 4 samples.
- Clinical Proteomic Tumor Analysis Consortium - Pancreatic Ductal Adenocarcinoma (CPTAC-PDA) [196], 1 sample.
- Clinical Proteomic Tumor Analysis Consortium - Uterine Corpus Endometrial Carcinoma (CPTAC-UCEC) [197], 3 samples.
- Clinical Proteomic Tumor Analysis Consortium - Lung Squamous Cell Carcinoma (CPTAC-LSCC) [198], 6 samples.

The presented amount of training/validation and testing samples are obtained after a pre-processing procedure. A process that seeks to normalize certain characteristics of the input image, such as voxel value dynamic range and positioning and remove external structures like the patient couch. Also the labels for the different types of tissues must be derived to be used in the segmentation task.

Data Cleaning

The first process to be performed is to clean the dataset from invalid samples. As mentioned before, the original objective of the dataset is not the synthetic AM generation nor the segmentation of different tissues. Thus, the dataset contains incomplete samples, mostly those that lack the NAC PET images. It also contains multiple samples per study, and non-matched samples that must be discarded. A non matching sample is a series of NAC PET, PET and CT/AC images that are not in the same FoV (they correspond to different sections of the body). Finally the data should be divided by patient and not by study to avoid correlation when dividing the dataset in train and validation sets.
The first operation performed on the set is the creation of a database including all samples which contained at least one image of each of the images modalities (NAC PET, PET and CT/AC). This information is obtained from the Digital Imaging and Communications in Medicine (DICOM) information on each sample, which includes the PatientID, the study ID, the image type, the amount of axial slices, the physical axial size and the path to the sample. Using this database the dataset is filtered by performing two operations. First each PatientID is inspected and if it does not posses at least one of each image types (NAC PET, PET and CT/AC) all entries of the patient are erased. The second operation is applied to each patient study ID. All studies of the selected patient are tested for overlapping on the axial dimension. All studies containing non overlapped samples are erased. If a study contains all types of images but they are not overlapped, the whole study is erased. In the case of a duplicated image modality within a study, only the image with higher overlap is kept and the duplicate is erased. The overlapping detection is done by finding the pair of images with the smallest distance between extrema and then checking if their limits define an overlap. The result of this operation is saved into a new database that now contains the information of the different matching images, along with their shared sizes.

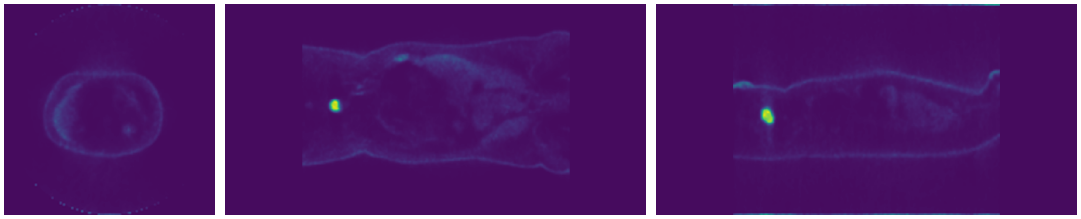
Finally the database is sorted by the PatientID. In the case of the training/validation dataset, it is divided into the train and validation sets. After the filtering the total number of matched studies in the training and validation dataset is 118, divided into 71 unique patients. In order to avoid correlation between the sets the division is performed by PatientID. The 10% of the dataset patients are chosen randomly and all their studies are extracted from the train dataset to create a validation dataset. The final size of the train dataset is 108 samples from 64 unique patients and the validation dataset is composed of 10 samples from 7 unique patients.

Image Normalization

The image normalization preprocessing is responsible of generating a common format along all samples. It represents the information in a way that is favourable for the NN training.

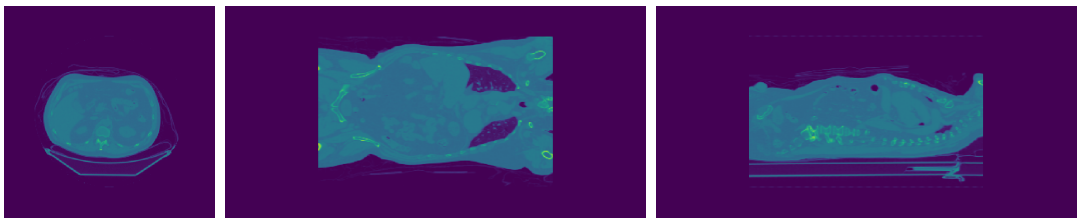
The first step is to define a common FoV and voxel size into which all samples will be projected. Using the data size information of all the filtered samples the absolute limits

of the FoV are calculated. This is achieved by keeping the global maxima and minima along the three axes of every sample of the dataset. With these values, the normalized FoV size is determined. The resulting FoV size is $700.0 \times 700.0 \times 1304.22$ $\texttt{mm}^3$. This FoV is divided into $256 \times 256 \times 512$ `voxels` , resulting in a near cubic voxel of size $2.73 \times 2.73 \times 2.54$ $\texttt{mm}^3$. All images are then centered in this FoV and re-sampled to fit the new voxel size. The cuts along each axis for a NAC PET and CT samples are shown in figure 4.10 and figure 4.11 respectively. All images in the following figures correspond to the processing of these NAC PET and CT volumes.



(a) (b) (c)

Figure 4.10: Axial (a), Coronal (b) and Sagital (c) cuts of the NAC PET re-sampled image, previous normalization of voxel values. Image from patient *08-22-1999-PETCT HEADNECK CA IN-05597*, study *08-22-1999-PETCT HEADNECK CA IN-05597*



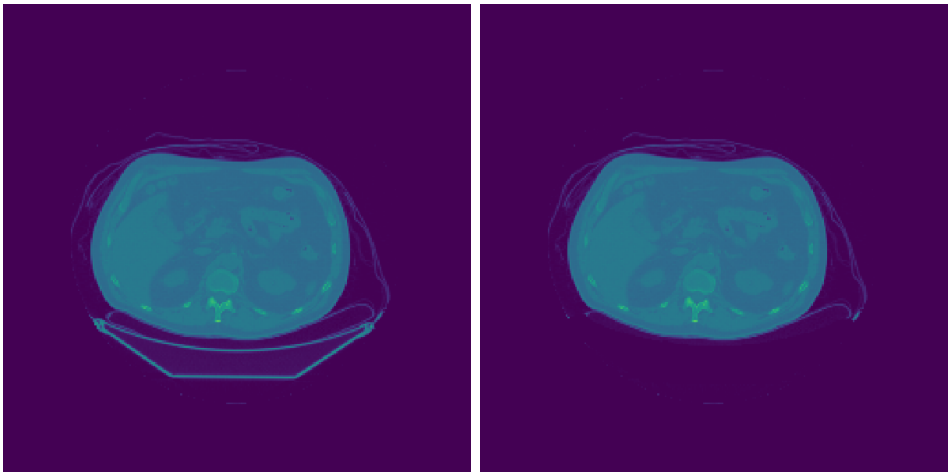
(a) (b) (c)

Figure 4.11: Axial (a), Coronal (b) and Sagital (c) cuts of the CT re-sampled image, previous normalization of voxel values. Image from patient *08-22-1999-PETCT HEADNECK CA IN-05597*, study *08-22-1999-PETCT HEADNECK CA IN-05597*

CT Couch Removal

Since the couch material is not invisible to the energies used in the CT scan (120 `keV`) it appears in the CT/AM image. The couch, which it is not part of the patient, is not observed on the PET image, since it measures indirectly the metabolism of the body. Thus the couch should be removed because it holds no relation with the attenuation structures of the patient and its shape and position depends on the PET

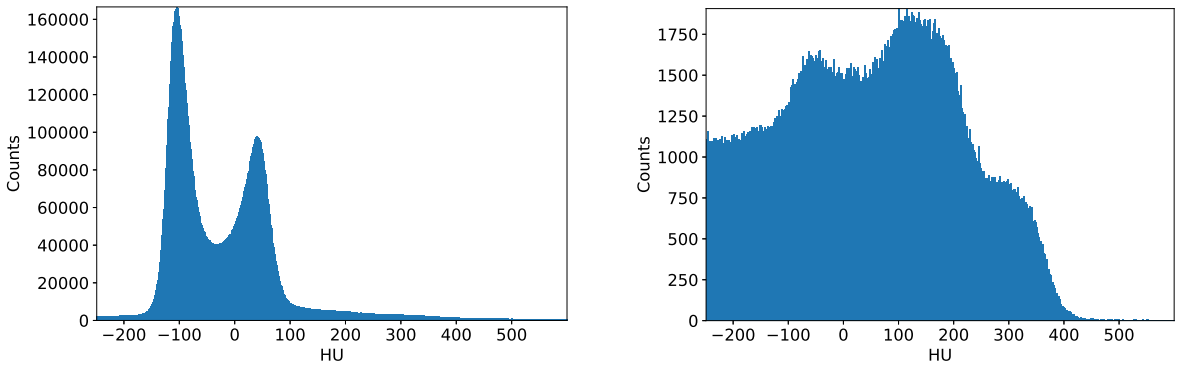
scanner model. In figure 4.12 the initial and final representations of a successful couch removal are shown. The removal of the couch is not straightforward, the couch cannot



(a) (b)

Figure 4.12: Central slice of the original CT image (a) and the same slice with couch removed (b).

be removed by means of its voxel values. It can be seen in figure 4.13 that the voxels values of the couch fall within the values of the body. To automatically separate the



(a) (b)

Figure 4.13: Histogram of the whole CT image (a) and histogram of the couch voxels (b). Note that the counts (Y axis) are not in the same scale. The histogram is restricted to the range $[-250, 600]$ HU to highlight the overlapping.

couch from the CT image a method based on the voxel variance along the axial axis is applied [199]. In order to apply the method the images are first clipped between -400 and 400 HU, eliminating noise and artefacts from strange materials. Then the sample dynamic range is normalized to $[0.0, 1.0]$. From the normalized volume, N_s evenly distributed consecutive slices are chosen. On these slices the multiple correlation is

calculated for each pixel. This calculation requires two steps, first the image is filtered using a disk kernel which defines the correlation area. This operation is defined for a single pixel (x, y) by:

$$\texttt{PxCorr}_n(x,y) = \sum_{i=-(\boldsymbol{k}_x/2)}^{(\boldsymbol{k}_x/2)} \sum_{j=-(\boldsymbol{k}_y/2)}^{(\boldsymbol{k}_y/2)} K_{corr.}(i,j)\boldsymbol{V}_n(x+i, y+j), \tag{4.46}$$

where $K_{corr.}$ is the disk kernel, $\boldsymbol{k}_x$ and $\boldsymbol{k}_y$ are the size of the kernel and $\boldsymbol{V}_n$ is the n^{th} slice of the N_s selected slices. Then the multiple correlation is calculated by subtracting initial slice value to the filtered slice:

$$\texttt{Corr}(x,y) = \boldsymbol{V}_n(x,y) - \texttt{PxCorr}_n(x,y). \tag{4.47}$$

Finally the raw couch mask is obtained by multiplying the N_s slices:

$$\texttt{Corr}(x,y) = \prod_{n=0}^{N_s} \texttt{Corr}_n(x,y), \tag{4.48}$$

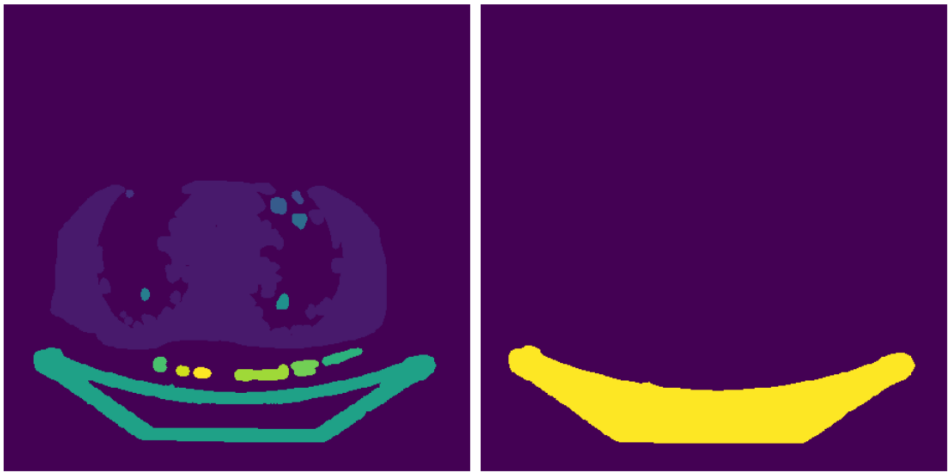
and keeping those pixels respecting the condition:

$$\texttt{Corr}_{raw}(x,y) = \begin{cases} 1 & \text{if } \texttt{Corr}(x,y) \neq 0, \\ 0 & \text{else.} \end{cases} \tag{4.49}$$

This mask is a coarse delimitation of the couch positions, and it contains multiple objects that need to be clustered into a single object, that represents the coach. The first operation is to apply an open operation to the raw mask using a disk shaped kernel for the binary erosion and dilation operations. This operation seeks to separate different structures into isolated blobs. Then each remaining blob is tagged, resulting in a configuration like the one presented in figure 4.14(a). Each blob is then identified using a connected component labelling algorithm [200], provided by the *scikit-image* [201] Python package. To select the correct blob from the constellation of structures the distance of each blob to the center of the Y axis is calculated. This is reliable since the patient is centered in the FoV and the couch is always on the lower extrema of the patient. The distance is obtained by the following equation:

$$d_i = \sqrt[4]{\sum_{j=1}^{V_{\texttt{Blob}}} (r_y^{j,i})^4}, \tag{4.50}$$

where d_i is the L4 distance of the i^{th} blob to the center of the Y axis. The axis is considered centered (the center is 0) and $r_y^{j,i}$ is the Y coordinate on the j^{th} pixel of the i^{th} blob. The L4 distance is used by recommendation of the original implementation of the algorithm. Finally the blob with the highest d_i value is chosen and it is heavily dilated, filled and eroded to obtain a couch mask as shown in figure 4.14(b).



(a) (b)

Figure 4.14: Raw initial mask, where each color represents a different blob (a) and the final couch mask (b).

The values used for the training and validation dataset are presented in table 4.1. These parameters were stable for most of the samples. Nevertheless some samples required manual tuning since the filling operation failed to generate a complete couch mask or there was too much contact between the body of the patient and the couch.

N_s	20	slices
Correlation Disk	6	pixels
First dilatation	4	pixels
First erosion	2	pixels
Second dilatation	15	pixels
Second erosion	13	pixels

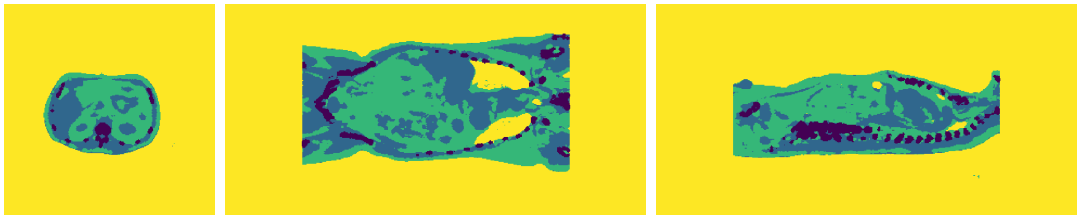
Table 4.1: Values used for couch removal implementation in HNSCC training and validation dataset

Body masking and label generation

By using the clean CT image several masking operations are performed using the typical HU values of different tissues. Four labels classes are extracted using a voxel value threshold and open and closing filters. The threshold values for each mask are shown in Table 4.2. The whole body mask is obtained by making a binary image from all pixel whose value is above the fat tissue lower limit. The final mask is a multichannel volume $\mathtt{v}^{\mathtt{TL}}$, containing the labels of each tissue. In figures 4.15 the produced labels are shown.

Mask	Threshold [HU]	
	Low	High
Air-Lung	-1000	-125
Fluids-Fat	-125	10
Parenchyma	10	90
Bone	90	1300

Table 4.2: Low and high threshold values for each label mask.



(a) (b) (c)

Figure 4.15: Axial (a), Coronal (b) and Sagital (c) cuts of the composed tissue mask. In yellow the air or background, in light green the fat-fluid, in light blue the parenchyma and in dark blue the bone.

CT Unit Normalization

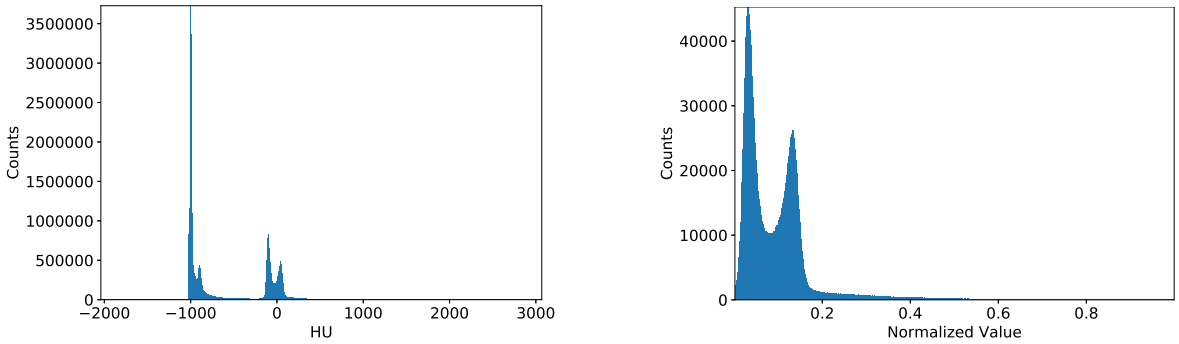
Since the inputs and outputs of the NN have a fixed dynamic range, the images in the dataset need to be normalized. To avoid the loss of information due to numeric precision in the normalization process, the dynamic range of the image should be optimized for the information that is important. Since the network is expected to detect soft tissue and bone structures, the dynamic range must be set to allow the maximum dynamic range between these limits. The expansion is applied by first clipping the values:

$$\boldsymbol{V}^{\mathtt{CT+Clip}}(x,y,z) = \begin{cases} -125\ \mathtt{HU} & \text{if } \boldsymbol{V}^{\mathtt{CT\,raw}}(x,y,z) \leq -125\ \mathtt{HU}, \\ 1300\ \mathtt{HU} & \text{if } \boldsymbol{V}^{\mathtt{CT\,raw}}(x,y,z) \geq 1300\ \mathtt{HU}, \\ \boldsymbol{V}^{\mathtt{CT\,raw}}(x,y,z) & \text{else}, \end{cases} \tag{4.51}$$

where $\boldsymbol{V}^{\mathtt{CT\,raw}}(x,y,z)$ is the value of a voxel of the CT image volume ($\boldsymbol{V}^{\mathtt{CT\,raw}}$). The normalization is then applied by:

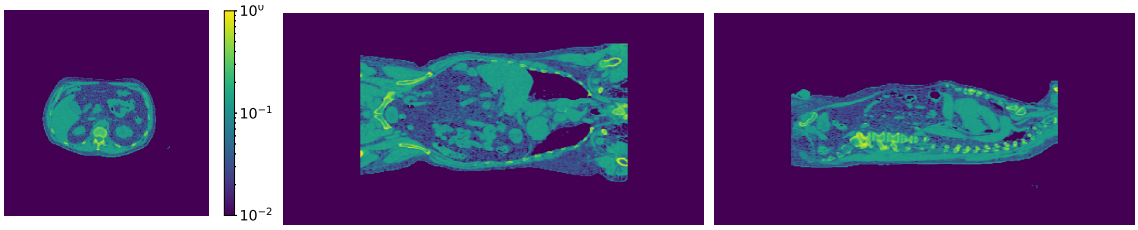
$$\boldsymbol{V}^{\mathtt{CT}}(x,y,z) = \frac{\boldsymbol{V}^{\mathtt{CT+Clip}}(x,y,z) - \min(\boldsymbol{V}^{\mathtt{CT+Clip}})}{\max(\boldsymbol{V}^{\mathtt{CT+Clip}}) - \min(\boldsymbol{V}^{\mathtt{CT+Clip}})}. \tag{4.52}$$

The resulting histogram is now in the expected range. The initial and final histograms can be seen in figure 4.16. The resulting CT image can be seen in figure 4.17.



(a) (b)

Figure 4.16: Initial image histogram (a) and normalized histogram (b) of the CT volume. The initial and last bin of both histograms were removed for display proposes.



(a) (b) (c)

Figure 4.17: Axial (a), Coronal (b) and Sagital (c) cuts of the CT re-sampled image, after normalization of voxel values.

PET Unit Normalization

Finally, as it is done with the CT volume, the NAC PET image should be also normalized. PET images can be stored using several units such as *counts* (direct number of detected counts coming from the voxel), *proportional to counts* (number that is proportional to *counts* value), *counts per second* (number of counts from voxel received within a second), *propotional to counts per second* and *Standard Uptake Value* (SUV) [202]. The SUV is the proportion of activity between the injected activity and the detected activity, divided by the mass density of a pixel[5] . The calculation of the SUV_{lbw} value is [203]:

$$\mathrm{SUV}_{lbw} = \frac{A_{\mathrm{img.}}}{A_{\mathrm{inj.}}/LBW}, \tag{4.53}$$

where LBW is the *Lean Body Weight*, which is patient dependant, $A_{\mathrm{img.}}$ is the image activity and $A_{\mathrm{inj.}}$ is the injected activity. The LBW is further normalized by

[5]More information: http://www.turkupetcentre.net/petanalysis/model_suv.html (last accessed 08/07/2019)

gender [203]. For male patients as:

$$LBW^m = \frac{9.27 \times 10^3 \text{Weight}}{6.68 \times 10^3 + 216 \frac{\text{Weight}}{\text{Height}^2}}, \tag{4.54}$$

and for female patients as:

$$LBW^f = \frac{9.27 \times 10^3 \text{Weight}}{8.78 \times 10^3 + 244 \frac{\text{Weight}}{\text{Height}^2}}, \tag{4.55}$$

using $[kg]$ and $[cm]$ for the Weight and Height respectively.

All NAC PET images in the set were converted to *counts per second* and then transformed to SUV_{lbw} using the information available in the DICOM files provided by the dataset. Although the conversion was possible, the resulting images do not show a standardized value. This was expected since the SUV value has been criticized for oversimplification, varying largely with the administered dose, the administration route, systemic condition of the body and biochemical factors, among others [204,205]. To address this problem the normalization of the image values was proposed to be done using the mean value observed in the voxels that correspond to the bones. This is done using the bone mask ($\boldsymbol{V}^{\texttt{bone}}$):

$$\boldsymbol{V}^{\texttt{NPT,dl.}} \leftarrow \frac{\boldsymbol{V}^{\texttt{NPT,raw}}}{\texttt{mean}(\boldsymbol{V}^{\texttt{NPT,raw}} \cdot \boldsymbol{V}^{\texttt{bone}})}, \tag{4.56}$$

the volume is now in a dimensionless (dl.) scale referred to the bone dose value. To achieve a volume normalization the NAC PET voxels values are clipped using a user defined value $\texttt{Lim}_{PET}$. This value was chosen empirically to be $\texttt{Lim}_{PET} = 10.0$.

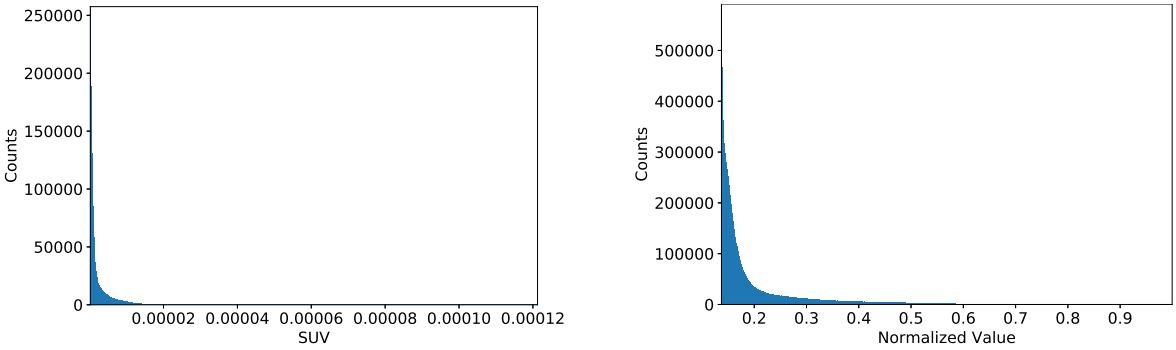
$$\boldsymbol{V}^{\texttt{NPT,dl.c.}}(x,y,z) = \begin{cases} \texttt{Lim}_{PET} & \texttt{if}\, \boldsymbol{V}^{\texttt{NPT,dl.}}(x,y,z) \geq \texttt{Lim}_{PET}, \\ \boldsymbol{V}^{\texttt{NPT,dl.}}(x,y,z) & \text{else}, \end{cases} \tag{4.57}$$

resulting in a dimensionless and capped (dl.c.) volume. finally the volume is normalized between 0.0 and 1.0

$$\boldsymbol{V}^{\texttt{NPT}}(x,y,z) \leftarrow \frac{\boldsymbol{V}^{\texttt{NPT,dl.c.}}(x,y,z) - \min(\texttt{V}^{\texttt{NACPET,dl.c.}})}{\max(\boldsymbol{V}^{\texttt{NPT,dl.c.}}) - \min(\boldsymbol{V}^{\texttt{NPT,dl.c.}})}. \tag{4.58}$$

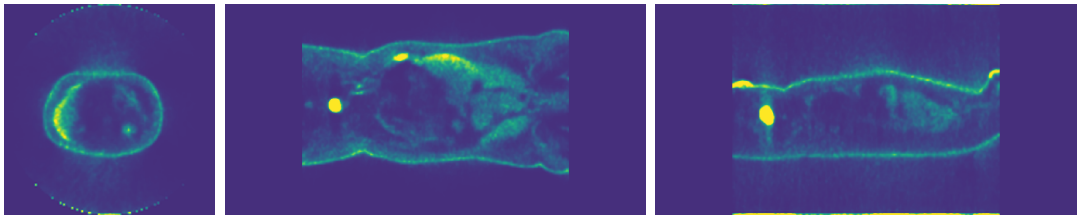
The resulting histogram is now in the expected range. The initial and final histograms can be seen in figure 4.18. In contrast to the CT histogram displayed in figure 4.16(b) the normalized histogram for the NAC PET image shown in figure 4.18(b) is more distributed over the normalized scale. The resulting NAC PET image can be seen in figure 4.19.

As it has been described, the NAC PET normalization is user defined (by means of the $\texttt{Lim}_{PET}$ parameter) and referred to bone caption. This can lead to a bias in the network training if used directly. However further processes are applied to this histogram before feeding it to the NN, as it will be described later. The validation



(a) (b)

Figure 4.18: Initial image histogram (a) and normalized histogram (b) of the NAC PET volume. The initial and last bin of both histograms were removed for display proposes.



(a) (b) (c)

Figure 4.19: Axial (a), Coronal (b) and Sagital (c) cuts of the NAC PET re-sampled image, previous normalization of voxel values.

dataset is also normalized using this procedure. In the testing datasets this normalization is not performed since the required information is not available if the target CT is unknown.

4.5.2 Tissue segmentation and AM generation using 3D U-Nets

The first and most direct method of AM generation is to implement an U-net topology which is trained to classify tissue labels. This approach seeks to generate a simple AM containing only a finite number of attenuation values. By restricting the AM to a handful of mutually exclusive classes the cost of the task is greatly reduced, compared to recreating a CT image. The chosen topology is based on a 3D U-Net, and is shown in figure 4.20. The 3D operation was chosen over the multiple 2D slices since it has been noted that due to the high volumetric correlation between slices, 3D convolutional method is better [206].

The input of the NN is the NAC PET image and the output a 4-channel segmentation volume containing the voxel wise PDF of each class (Air-Lung, Fluids-Fat, Parenchyma and Bone). The network possesses 5 resolution levels, each of them composed of two convolutional layers with filter shape of $3 \times 3 \times 3$ and ReLU activation. Each

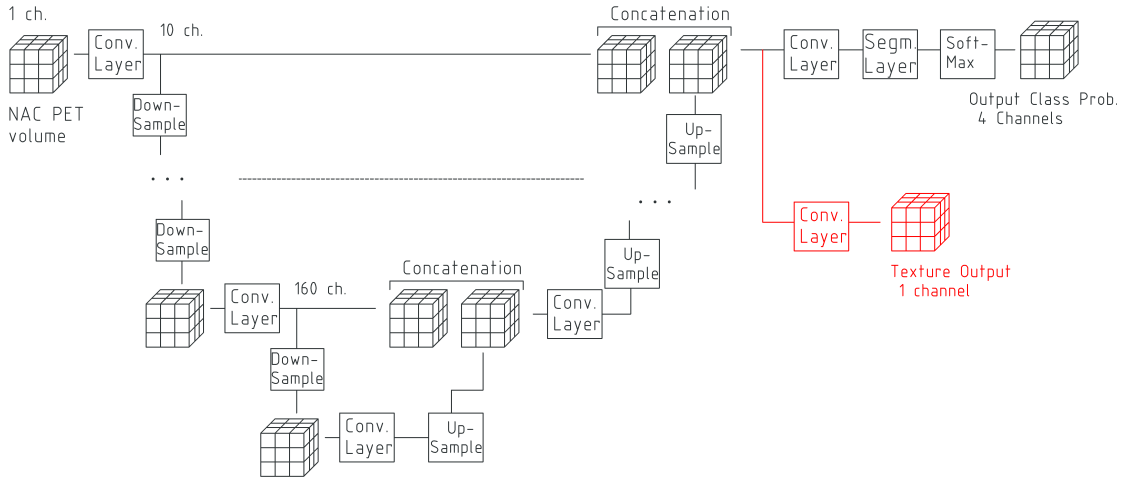


Figure 4.20: 3D U-Net topology implemented for NAC PET to CT image translation and tissue segmentation. In the segmentation network topology is shown in black, which outputs a 4 channel volume containing each tissue type probability. The additional CT generation layer is presented in red, connected to the last layer of the 3D U-Net topology, and outputs a synthetic CT image.

convolutional filter, going from higher to lower resolutions, have $10, 20, 40, 80$ and 160 channels respectively. Each resolution level posses a skip connection between the down-sample and up-sample path. The segmentation path uses 4 channels filters with kernel size $3 \times 3 \times 3$. Instead of convolutional resampling the resolution changes are performed using trilinear up- or down-sampling. After each convolutional layer, voxel normalization along feature maps is applied. The voxel normalization divides each voxel value by $\sqrt{\frac{1}{p_c}\sum_{i=1}^{p_c}(\mathtt{v}_i)^2 + e}$, where p_c is the number of channels in the feature map, $\mathtt{v}_i$ is the voxel value of the i^{th} voxel and $e = 1.0 \times 10^{-8}$. Instead of performing a carefull initialization of the weights, a normal distribution is used $\mathcal{N}(0, 1)$. Then each filter kernel is weighted by a factor based on He's [207] scaled initialization of weights, this was reported to have similar effects than complex initialization methods [152]. Finally the output layer is softmaxed along the last dimension (objective classes).

Texture generation

A simple addition to the segmentation topology enables the NN to produce synthetic CT (sCT) or AM images. This addition is shown in figure 4.20, as the added block is red color. The texturing layer is composed of a single convolutional layer with kernel size $1 \times 1 \times 1$ $\mathtt{vx}^3$ and a sigmoid activation. This layer is trained along the segmentation layers and makes use of the features extracted by it to generate a real valued image instead a class map.

Network input generation

The sample resolution of the dataset is $256 \times 256 \times 512$ $\mathtt{vx}^3$, such input would not fit in the 12 GB RAM of a high-end GPU, the current limit of commercial desktop devices. For this reason the input is scaled to $128 \times 128 \times 256$ $\mathtt{voxels}^3$ and cropped into $128 \times 128 \times 32$ $\mathtt{vx}^3$ slices. Then, the input of the NN is a slice of the sample of size $128 \times 128 \times 32$ $\mathtt{vx}^3$. During training, the input slice is transformed before being passed to the NN. This process is called *data augmentation*, a mandatory step when only a handful of annotated examples exist. This is the common scenario of most medical imaging problems. Since the training dataset used in this work is composed of only 108 samples, from 64 unique patients, data augmentation is necessary. The first enhancing operation is the scaling and slicing. This is necessary for two reasons: the first is the current technological capability restricts the use of full resolution volumes and the second is that the slicing operation not only reduces the overall memory requirements of the NN, it also multiplies the number of samples. Although the different slices are highly correlated. The slicing is performed at random, by choosing an arbitrary axial position z. As a result the different slices show different structures at different positions, helping the network to avoid over-fitting. An other common data augmentation technique in image processing is the image rotation and flipping. The transformation involves the flipping (or mirroring) of the image and/or rotation along one or different axis. The transformation selected to be applied is only a rotation along the axial axis. This rotation can be performed to each sample by choosing a random rotation value between 0.0 and π and then rotated and interpolated using a bilinear interpolation. However it was found that in medical images such as PET, this has little effect, probably due to the fact that patients have always the same orientation or can be flipped to match the training.

The problem of low statistics in the dataset also impacts the normalization of the inputs. The dataset is composed of images from few scanner models and only one type of pathology. This affects the representation power of the dataset. To increase the variance of the dataset while trying to keep meaningful relations and structures within the input image, a histogram shifting and clipping of each sample is applied. This transformation is performed by first adding a random value ($-0.1 \leq \boldsymbol{V}_{\mathtt{rnd}} \leq 0.1$) to the image:

$$\boldsymbol{V}^{\mathtt{NPT}} \leftarrow \boldsymbol{V}^{\mathtt{NPT}} + \boldsymbol{V}_{\mathtt{rnd}}, \tag{4.59}$$

then clipping all values back to the interval $[0.0, 1.0]$

$$\boldsymbol{V}^{\mathtt{NPT}}(x,y,z) \leftarrow \begin{cases} 1.0 & \text{if } \boldsymbol{V}^{\mathtt{NPT}}(x,y,z) \geq 1.0, \\ 0.0 & \text{if } \boldsymbol{V}^{\mathtt{NPT}}(x,y,z) \leq 0.0, \\ \boldsymbol{V}^{\mathtt{NPT}}(x,y,z) & \text{else,} \end{cases} \tag{4.60}$$

and finally re-normalizing the histogram

$$\boldsymbol{V}^{\mathtt{NPT}} \leftarrow \frac{\boldsymbol{V}^{\mathtt{NPT}} - \min(\boldsymbol{V}^{\mathtt{NPT}})}{\max(\boldsymbol{V}^{\mathtt{NPT}}) - \min(\boldsymbol{V}^{\mathtt{NPT}})}. \tag{4.61}$$

The final process of data augmentation is the addition of noise to the input image values. This is a standard procedure in many data augmentation pipelines. The noise added to the images has a normal distribution with $\overline{\boldsymbol{V}_{\texttt{rnd}}} = 0.0$ and $\sigma(\boldsymbol{V}_{\texttt{rnd}}) = 0.025$.

Segmentation Objective Metric

For the segmentation task a multi-class DICE loss is used. This objective function can be seen in equation 4.62:

$$\mathcal{L}_D = \frac{1}{p_c} \sum_{i=1}^{p_c} \frac{2 \sum_{j=1}^{V} \hat{\mathtt{v}}_{j,i}^{\texttt{TL}} \mathtt{v}_{j,i}^{\texttt{TL}}}{\sum_{j=1}^{V} (\hat{\mathtt{v}}_{j,i}^{\texttt{TL}})^2 + \sum_{j}^{V} (\mathtt{v}_{j,i}^{\texttt{TL}})^2}, \tag{4.62}$$

where p_c is the number of objective classes, V the number of voxels in the volume, $\mathtt{v}_{j,i}^{\texttt{TL}}$ are the voxels of the tissue labels ground truth and $\hat{\mathtt{v}}_{j,i}^{\texttt{TL}}$ the values of estimated labels (the softmaxed output of the network). The dice loss ranges from 0 to 1. It produces its maximum value when all voxels of the ground truth ($\mathtt{v}_{j,i}^{\texttt{TL}}$) have the same value as the softmaxed output voxels ($\hat{\mathtt{v}}_{j,i}^{\texttt{TL}}$). Since the output is softmaxed the denominator of equation 4.62 is always larger than its numerator except when $\mathtt{v}_{j,i}^{\texttt{TL}}$ and $\hat{\mathtt{v}}_{j,i}^{\texttt{TL}}$ are identical. In the case of a multi-class problem ($p_c > 1$) the final value is divided by the number of classes. This metric was adapted from [165] in order to be used in multi-class segmentation problems.

sCT Generation Objective Metric

The sCT generation branch or texturing branch tries to replicate the real valued CT corresponding to the NAC PET input. Thus the NN is trained using the euclidean distance between the sCT and the objective CT image:

$$\mathcal{L}_T = ||\boldsymbol{V}^{\texttt{sCT}}, \boldsymbol{V}^{\texttt{CT}}||^2. \tag{4.63}$$

Complete Objective Metric

Finally the 3D U-Net loss is built by combining the segmentation and texturing loss:

$$\mathcal{L}^{\texttt{U-Net}} = (1 - \mathcal{L}_D) + k_e \mathcal{L}_T, \tag{4.64}$$

where k_e is a coupling constant. Note that the segmentation layers will only receive gradient from $\mathcal{L}_D$ and the texturing layers only from $\mathcal{L}_T$. The 3D U-Net, whose task is to extract features from the NAC PET image, receives gradients from the weighted combination of the two tasks being trained.

Training

The network is trained for 100 epochs with 100 mini-batches. The mini-batch size is 4 samples. The optimizer is ADAM, with learning rate $\epsilon = 0.001$, first moment decay rate $\beta_1 = 0.9$, second moment decay rate $\beta_2 = 0.99$.

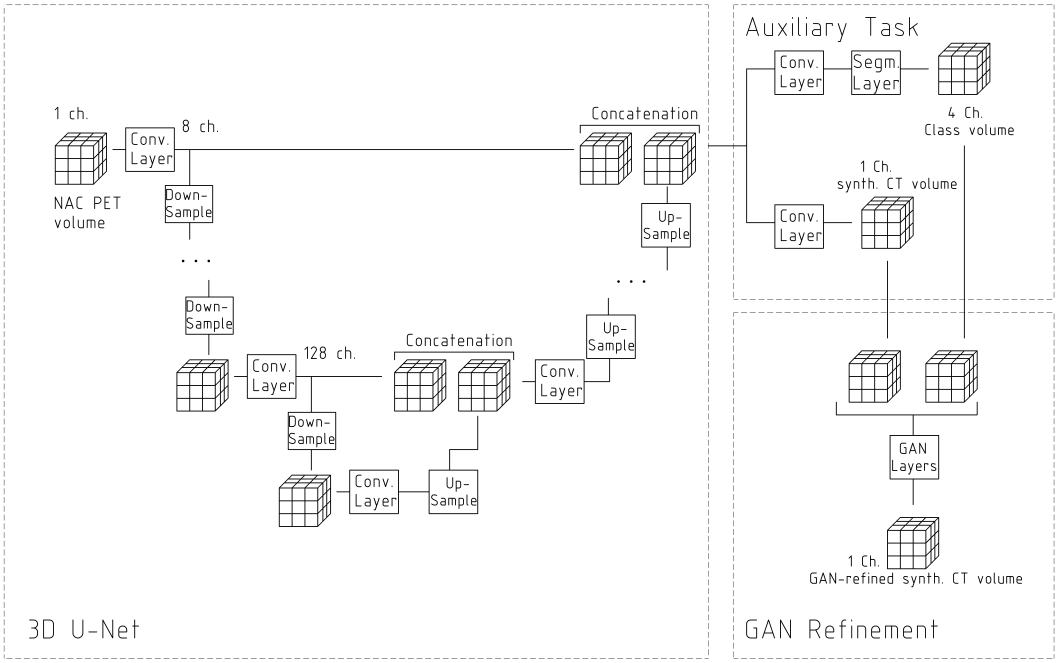


Figure 4.21: Generator network topology. Top branch is used for label segmentation and bottom branch for artificial CT generation. The bottom output is the GAN refined synthetic CT built from the U-Net outputs.

4.5.3 Texture generation using hybrid adversarial loss

To improve the AM quality generated by the 3D U-Net while conserving the informative gradients from the segmentation task, a hybrid generative approach is implemented. To this end an adversarial loss block is added to enhance the image resolution. The resulting topology is composed of a 3D U-Net generator and a convolutional critic (or discriminator).

Generator

The generator is based on the 3D U-Net network presented in the previous section (4.5.2), where additional layers to improve the sCT quality were added. The added GAN layers use the outputs of the U-Net as input. They are a collection of 5 convolutional layers with 8 filters each. All convolutional operations use a filter of size $3 \times 3 \times 3$ except the output layer which use a $1 \times 1 \times 1$ filter. The complete topology of the generator is shown in figure 4.21.

Critic

The critic or discriminator network is a fully convolutional network with ReLU activation in all layers except the last layer which has a linear activation. The input of this network is a two channel volume composed of the NAC PET volume and the real or sCT image. The output of the network is a value proportional to quality value of the

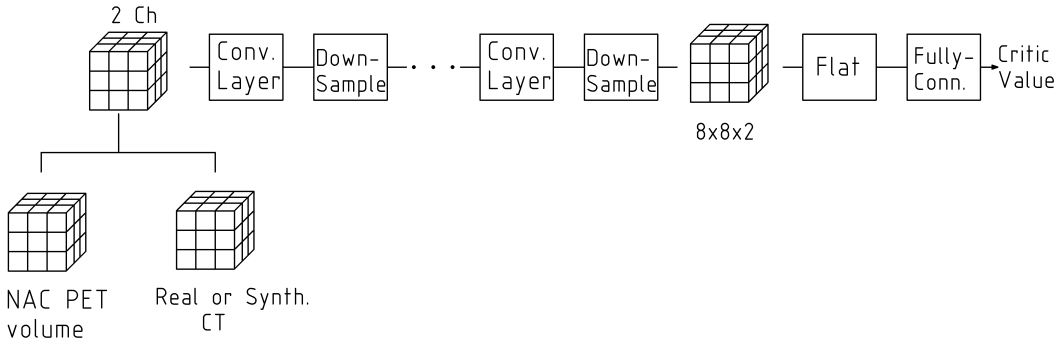


Figure 4.22: Critic or discriminator network topology.

generated image. The critic topology is show in figure 4.22. The network is conformed by 4 resolution levels with two convolutional layers per level. Each convolution has a filter size of $3 \times 3 \times 3$ and ReLU activation. Neither batch nor pixel normalization is applied. The last two layers of the critic are a flatten operation followed by a single dense layer with linear activation.

Network input generation

The input generation is the same as the one presented in section 4.5.2.

Training Scheme

The training of the network is divided into two stages: first the generator network is trained in a supervised manner using a composed loss. Up to the added GAN layers, the NN uses the losses described in previous section (4.5.2). After the initial training, the adversarial training starts. The adversarial training uses the Wasserstein-GAN (W-GAN) strategy [177], resulting in a generator loss as shown in equation 4.65,

$$\mathcal{L}_W = -\texttt{Crit}_W(\texttt{Gen}_W(\boldsymbol{V}^{\texttt{NPT}}), \boldsymbol{V}^{\texttt{NPT}}), \tag{4.65}$$

where $\texttt{Crit}_W()$ is the critic network function, $\texttt{Gen}_W()$ is the generator network function and $\boldsymbol{V}^{\texttt{NPT}}$ is the input NAC PET image. Since the critic $\texttt{Crit}_W()$ is now a conditional discriminator, it receives the sCT $\boldsymbol{V}^{\texttt{sCT}} = \texttt{Gen}_W(\boldsymbol{V}^{\texttt{NPT}})$ and the reference volume $\boldsymbol{V}^{\texttt{NPT}}$. During the adversarial training the GAN layers become active and are trained using the W-GAN loss. The gradient of the GAN does not flow into the 3D U-Net layers. The resulting generator is shown in equation 4.66, it is composed of the WGAN loss with the additional supervised loss coupled using the $k_{\texttt{sup}}$ constant.

$$\mathcal{L}_{gen}^{GAN} = \mathcal{L}_W + k_{\texttt{sup}}\mathcal{L}^{\texttt{U-Net}}. \tag{4.66}$$

The critic is trained using a coupled pairs of NAC PET and CT/sCT images, real or fake. It is trained using the wasserstein loss cost in equation 4.67,

$$J_{crit.}^{GAN} = \frac{1}{N_m^{true}} \sum_{n=1}^{N_m^{true}} \texttt{Crit}_W(\boldsymbol{V}_n^{\texttt{NPT}}, \boldsymbol{V}_n^{\texttt{CT}}) - \frac{1}{N_m^{fake}} \sum_{n=1}^{N_m^{fake}} \texttt{Crit}_W(\boldsymbol{V}_n^{\texttt{NPT}}, \texttt{Gen}_W(\boldsymbol{V}_n^{\texttt{NPT}})) + k_{\texttt{gp}} \frac{1}{N_m^{true}} \sum_{n=1}^{N_m^{true}} \nabla_{\texttt{GP}}(\texttt{Gen}_W(\boldsymbol{V}_n^{\texttt{NPT}}), \boldsymbol{V}_n^{\texttt{CT}}), \quad (4.67)$$

where $\boldsymbol{V}_n^{\texttt{CT}}$ is the real CT of the n^{th} sample, $\boldsymbol{V}_n^{\texttt{NPT}}$ is the n^{th} NAC PET input, $\nabla_{\texttt{GP}}$ is the gradient penalty and $k_{\texttt{gp}} = 10.0$. The critic is trained 5 steps for each generator step. At the initial step of the GAN training stage the critic is optimally trained before starting the GAN training loop. The generators are trained using the ADAM optimizer with parameters $\beta_1 = 0.0$, $\beta_2 = 0.99$ and learning rate $\epsilon = 0.0001$. The discriminator uses a RMSprop optimizer with learning rate $\epsilon = 0.0005$. During the GAN training the adversarial loss gradient flow is limited to the last part of the network. This prevents the GAN texturing layers to modify the features generated using the supervised loss. The network is trained for 100 epochs with 100 mini-batches. The mini-batch size is $N_m = 4$ samples.

4.5.4 Texture generation with Progressive GANs

The final studied topology is a ProGAN. The motivation behind this topology is that the dimensionality of the 3D volumes is comparable to that of high resolution 2D images. The generator, as in previous topologies is based on a 3D U-Net, since the objective is to build a conditional generative network. This topology is also similar to a Pix2Pix [208] with 3D filters plus a progressive growing approach. The generator topology follows the 3D U-Net presented in previous sections. Composed of convolutional blocks of $10, 20, 40, 80$ and 160 filters each, on resolutions of $(128 \times 128 \times 32)$, $(64 \times 64 \times 16)$, $(32 \times 32 \times 8)$ and $(16 \times 16 \times 4)$ respectively. The NN grows from the $(16 \times 16 \times 4)$ up to the $(128 \times 128 \times 32)$ resolution following the strategy presented in section 4.4. The generator topology is shown in figure 4.23(a), where the different blocks are marked.

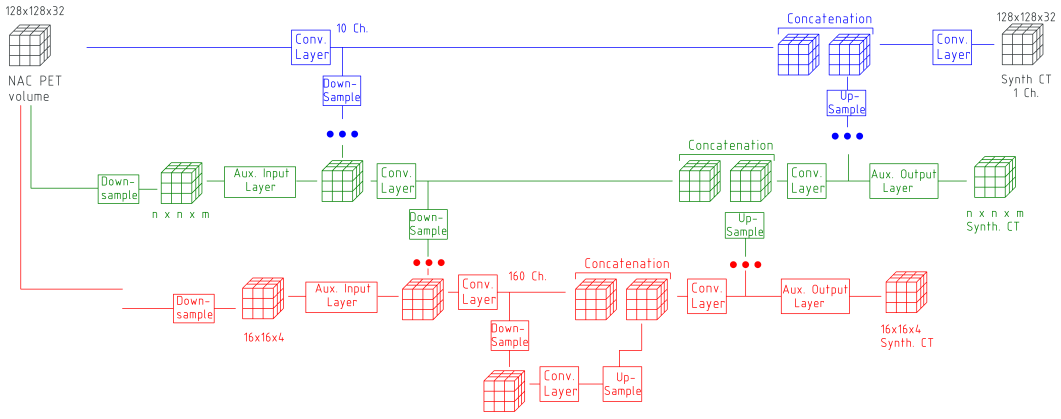
The critic has the same topology as the critic presented in previous section. Just as the generator, the critic also grows following the current resolution. The discriminator topology is shown in figure 4.23(b).

Network input generation

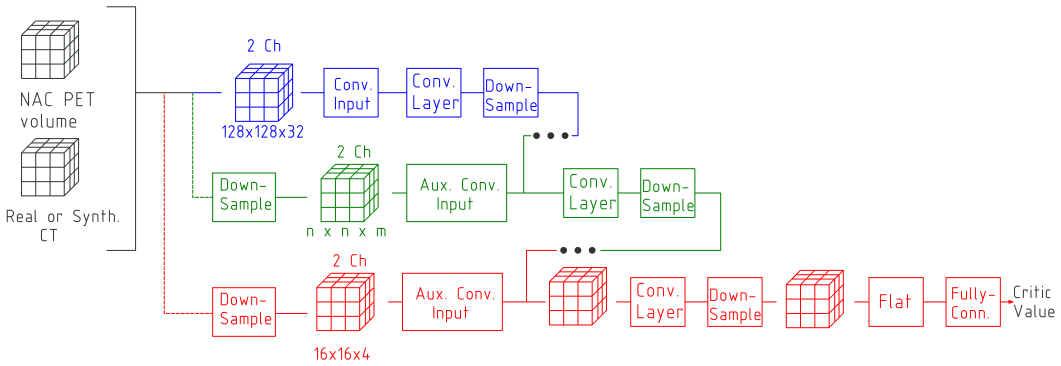
The input generation is the same as the one presented in the previous section (4.5.2). With the addition of the downsampling. At each resolution stage the input is downsampled (before the data augmentation process) to match the current network resolution.

Training Scheme

The network uses only the adversarial losses. The generator use equation 4.65 and critic cost in equation 4.67. The training begins with the lowest resolution block,



(a)



(b)

Figure 4.23: Progressive generator (a) and critic (b) NN topologies. Each coloured section is a bock that grows over the previous structure. The NN starts with the structure shown in red, once this resolution level is trained this connection is eliminated and the next resolution level is connected (an intermediate resolution structure is shown in green). This process continues until the final resolution structure is added which shields the desired resolution. The block marked with *Aux.* are only used during the training of its specific resolution and then discarded.

$(16 \times 16 \times 4)$, and is trained for 1500 epochs with 100 mini-batches and mini-batch size of $N_m = 4$ samples. At each following resolution the number of epochs increases, resulting in 3000, 6000, 8000, 15000 epochs respectively. The fade-in process takes place after each resolution training is finished and it is applied for half the number of epochs of the training epochs in the current resolution.

4.5.5 Tests and Results

Each NN topology was tested on the quality of the generated CT images using four different metrics:

- Structural Similarity Index (SSIM) [209], Used to measure the similarity between two images:

$$\texttt{SSIM}(\boldsymbol{V}^{\texttt{CT}}, \boldsymbol{V}^{\texttt{sCT}}) = \frac{1}{S_m} \sum_{m=1}^{S_m} \frac{2\,\overline{\boldsymbol{V}}_m^{\texttt{CT,win}}\,\overline{\boldsymbol{V}}_m^{\texttt{sCT,win}}\,(2\,\sigma_{\boldsymbol{V}_m^{\texttt{CT,win}},\boldsymbol{V}_m^{\texttt{sCT,win}}})}{((\overline{\boldsymbol{V}}_m^{\texttt{CT,win}})^2\,(\overline{\boldsymbol{V}}_m^{\texttt{sCT,win}})^2)\,(\sigma^2_{\boldsymbol{V}_m^{\texttt{CT,win}}}\,\sigma^2_{\boldsymbol{V}_m^{\texttt{sCT,win}}})}, \tag{4.68}$$

  where $\boldsymbol{V}_m^{\texttt{CT,win}}$ and $\boldsymbol{V}_m^{\texttt{sCT,win}}$ are $5 \times 5 \times 5$ windows and S_m is the number of windows required to scan the complete volume.

- Peak Signal to Noise Ratio (PSNR), The ratio between the *signal* and the corrupting *noise* in the image:

$$\texttt{PSNR}(\boldsymbol{V}^{\texttt{CT}}, \boldsymbol{V}^{\texttt{sCT}}) = 10 \log_{10} \left(\frac{1.0}{\texttt{MSE}(\boldsymbol{V}^{\texttt{CT}}, \boldsymbol{V}^{\texttt{sCT}})} \right), \tag{4.69}$$

  with

$$\texttt{MSE}(\boldsymbol{V}^{\texttt{CT}}, \boldsymbol{V}^{\texttt{sCT}}) = \frac{1}{V} \sum_{i=1}^{V} |\texttt{v}_i^{\texttt{sCT}} - \texttt{v}_i^{\texttt{CT}}|^2, \tag{4.70}$$

  where V is the number of voxels in the image.

- Mean Absolute Error (MAE):

$$\texttt{MAE}(\boldsymbol{V}^{\texttt{CT}}, \boldsymbol{V}^{\texttt{sCT}}) = \frac{1}{V} \sum_{i=1}^{V} |\texttt{v}_i^{\texttt{sCT}} - \texttt{v}_i^{\texttt{CT}}|. \tag{4.71}$$

- Normalized Cross Correlation (NCC):

$$\texttt{NCC}(\boldsymbol{V}^{\texttt{CT}}, \boldsymbol{V}^{\texttt{sCT}}) = \frac{\frac{1}{V} \sum_i^V (\boldsymbol{V}^{\texttt{CT}} - \overline{\boldsymbol{V}}^{\texttt{CT}}])(\boldsymbol{V}^{\texttt{sCT}} - \overline{\boldsymbol{V}}^{\texttt{sCT}})}{\sigma_{\boldsymbol{V}^{\texttt{CT}}} \sigma_{\boldsymbol{V}^{\texttt{sCT}}}}. \tag{4.72}$$

During the testing procedure the NAC PET volume is fed in consecutive slices of $128 \times 128 \times 32$ voxels. The intensity is re-scaled to $[0, 1]$ for each slice. The resulting sCT/AM slices are composed into a single volume using a weighted average operation. The weighting is done using a Gaussian function with value 1 on the central slice and

a standard deviation of 4 slices. The metrics were applied on the occupied voxels of the full sized volumes to avoid the effect of the large vacant sections in the volumes.

The metrics are presented in the figure 4.24, each of the box plots corresponds to the metrics of the 3D U-Net, 3D U-Net+GAN and ProGAN on the different test sets. The datasets are presented by source since the number of samples of the individual datasets is too small in some cases.

An ablation test was done to assess the influence of the different network properties on the 3D-Unet topologies. Five different networks were trained for the same number of epochs and have the same number of parameters. The baseline network has no supervised loss (shown in equation 4.64), the adversarial gradient is allowed through the U-Net and is pre-trained. The code *Sup.* indicates supervised loss during GAN training, *(No) Pre-Train* indicates whether or not the network has a supervised pre-train step, *No GAN.Loss* indicates no adversarial training and *RestrictedGrad.* indicates that the adversarial gradient is not allowed into the U-Net. An additional row is added for the ProGAN, which does not share the same number of training epoch as the other topologies, nevertheless it has a similar number of parameters.

The results of the study using the PSNR, MAE and MCC metrics of all datasets is summarized in table 4.3. The highlighted line represents the proposed hybrid WGAN and U-Net network.

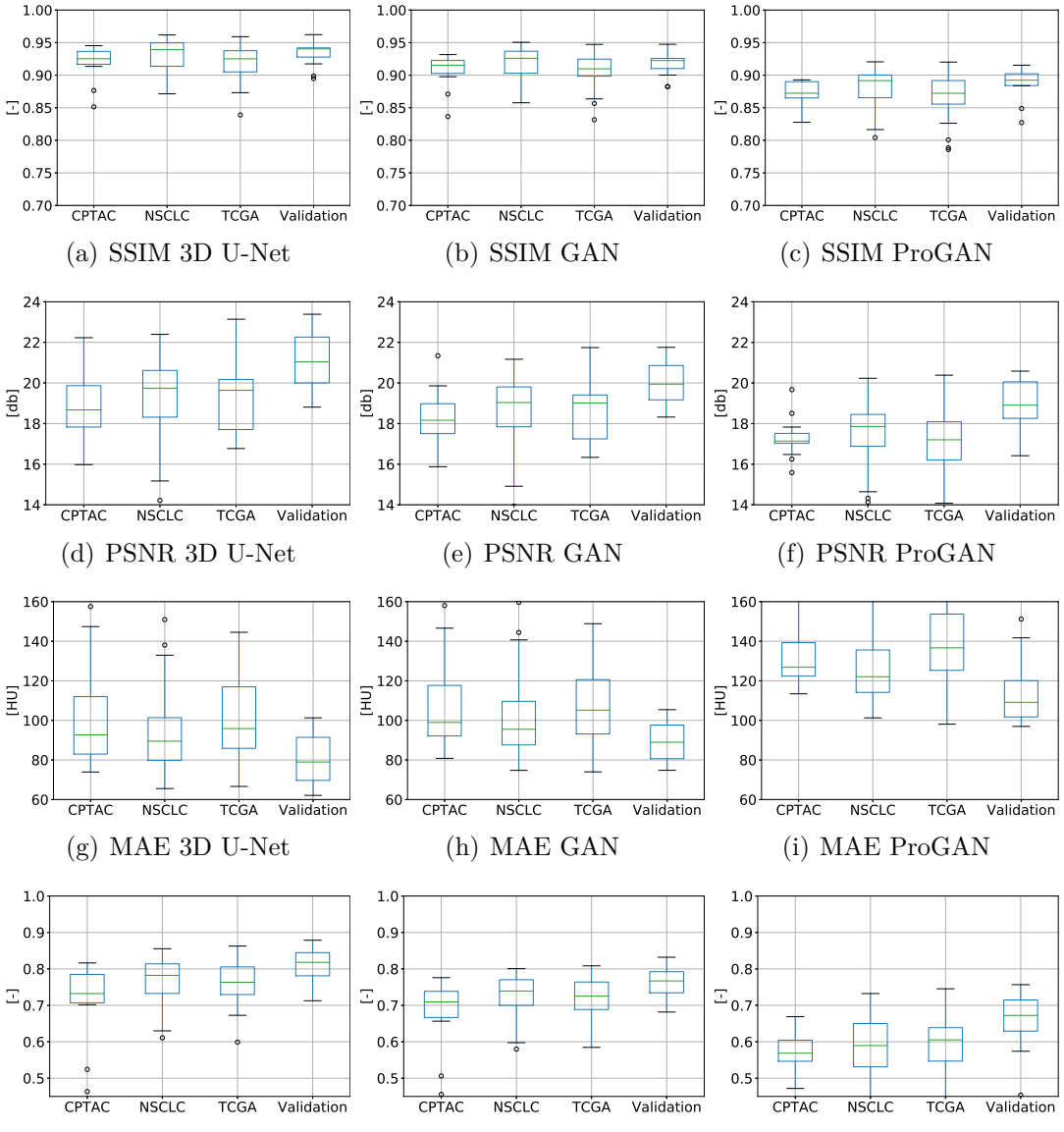
The resulting sCT of three samples in the test set and one from the validation dataset are presented at figures 4.25, 4.26, 4.27, 4.28, 4.29m 4.30, 4.31 and 4.32. Each sample was processes using the three presented topologies. First the slices of the HNSCC-01-0148 sample from the validation dataset are shown figure 4.25 and its 3D volumes in figure 4.26. Then the three validations samples, AMC-009 from NSCLC Radiogenomics is shown in figures 4.27 and 4.28, C3N-00957 from CPTAC-PDA is shown in figures 4.29 and 4.30 and TCGA-BB-7863 from TCGA-HNSC is shown in figures 4.31 and 4.32.

Finally, the attenuation correction of the generated attenuation maps was tested by means of attenuation sinograms. Each CT and sCT was first converted from $[HU]$ scale to linear attenuation $[1/\text{cm}^2]$, at 120 `keV`. Then the linear attenuation was scaled to the PET energy, 511 `keV`. This conversion was linear, using the relation between the water coefficient at 120 `keV` and 511 `keV`. The resulting factor is $(9.687\times10^{-2}/1.607\times10^{-1}) = 0.603$. Using the real and generated AM a pair of attenuation sinograms was created. The attenuation sinograms are a collection of LoRs, with different (ρ, θ, q) parameters that cover the whole FoV. For each of these LoRs the attenuation coefficient was calculated as presented in equation 1.21, $x_l^\mu = e^{-\sum_{u=1}^{U} h_{l,u}^\mu \mu_u}$. Using the provided CT as ground truth, the differences in the attenuation sinograms was measured using the mean difference:

$$\overline{x^\mu} = \frac{1}{L}\sum_{l=1}^{L} \frac{x_l^{\mu,\texttt{sCT}} - x_l^{\mu,\texttt{CT}}}{x_l^{\mu,\texttt{CT}}} \times 100\%, \tag{4.73}$$

the standard deviation:

$$\sigma(x^\mu) = \sqrt{\frac{\sum_{l=1}^{L}(x_l^{\mu,\texttt{sCT}} - x_l^{\mu,\texttt{CT}})100 - (\overline{x^\mu})}{N}}, \tag{4.74}$$



(a) SSIM 3D U-Net (b) SSIM GAN (c) SSIM ProGAN

(d) PSNR 3D U-Net (e) PSNR GAN (f) PSNR ProGAN

(g) MAE 3D U-Net (h) MAE GAN (i) MAE ProGAN

(j) NCC 3D U-Net (k) NCC GAN (l) NCC ProGAN

Figure 4.24: Boxplot of SSIM (a,b,c), PSNR (d,e,f), MAE (g,h,i) and NCC (j,k,l) for each test data source and validation dataset. The figures (a,d,g,j), (b,e,h,k) and (c,f,i,l) are corresponding to the 3D U-Net, the network with adversarial refinement and the ProGAN, respectively.

Topology	Dataset	SSIM [−]	PSNR [dB]	MAE [HU]	NCC [−]
Baseline	Validation	0.888 ± 0.028	18.3 ± 1.3	121 ± 20	0.640 ± 0.080
	Testing	0.890 ± 0.029	17.3 ± 1.5	133 ± 22	0.600 ± 0.070
Sup. Pre-Train	Validation	0.923 ± 0.022	19.6 ± 1.2	92 ± 14	0.750 ± 0.060
	Testing	0.916 ± 0.027	18.3 ± 1.4	105 ± 20	0.700 ± 0.070
Sup. No Pre-Train	Validation	0.868 ± 0.042	17.5 ± 1.8	147 ± 38	0.54 ± 0.14
	Testing	0.884 ± 0.031	17.0 ± 1.4	143 ± 25	0.570 ± 0.080
Sup. Pre-Train No GAN.Loss.	Validation	0.933 ± 0.021	21.0 ± 1.5	80 ± 13	0.800 ± 0.050
	Testing	0.926 ± 0.024	19.3 ± 1.7	97 ± 20	0.760 ± 0.060
Sup. Pre-Train RestrictedGrad.	Validation	$\mathbf{0.916 \pm 0.020}$	$\mathbf{19.9 \pm 1.3}$	$\mathbf{88.9 \pm 10.5}$	$\mathbf{0.760 \pm 0.050}$
	Testing	$\mathbf{0.913 \pm 0.024}$	$\mathbf{18.6 \pm 1.5}$	$\mathbf{103 \pm 18}$	$\mathbf{0.720 \pm 0.060}$
ProGAN	Validation	0.886 ± 0.026	18.8 ± 1.3	115 ± 18	0.650 ± 0.090
	Testing	0.876 ± 0.027	17.4 ± 1.3	132 ± 22	0.580 ± 0.070

Table 4.3: Validation (10 samples) and testing (133 samples) scores for topologies with additional features. Supervised loss (*Sup.*), pre-train step (*Pre-Train*), no adversarial training (*No GAN.Loss.*) and adversarial gradient restricted to last layers (RestrictedGrad.). The presented hybrid topology is in bold text.

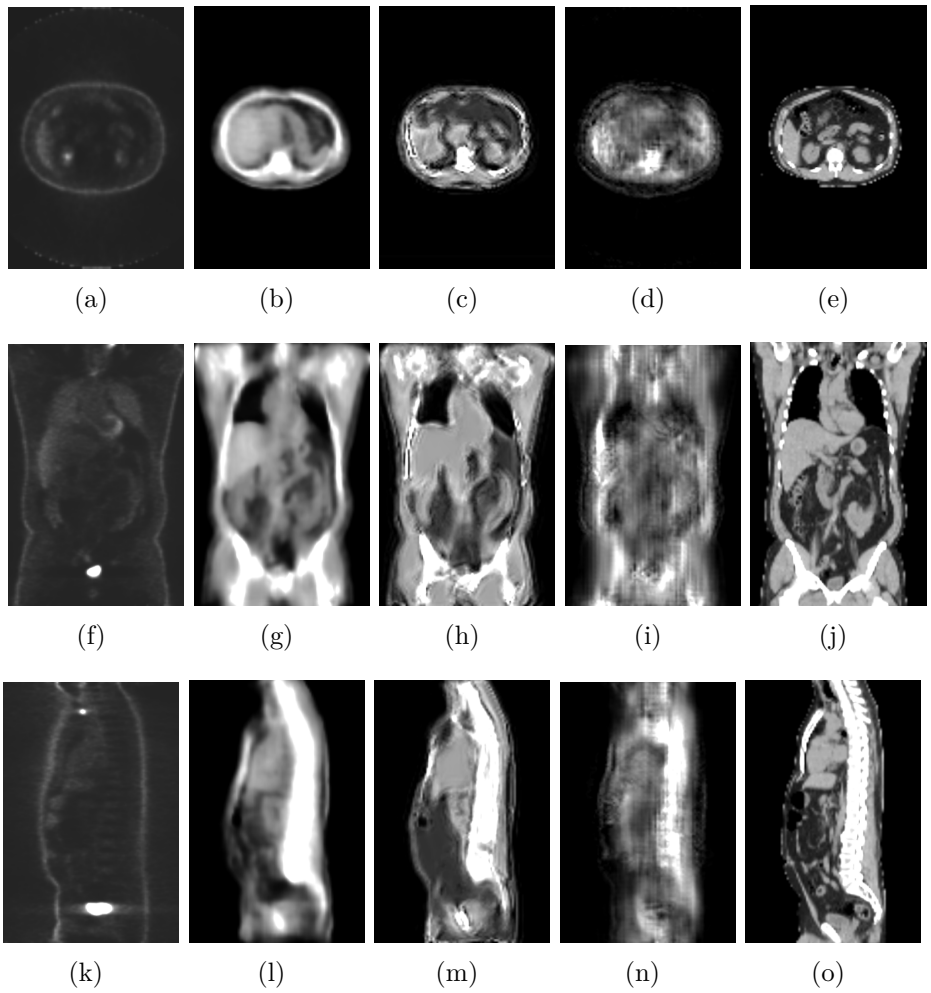


Figure 4.25: Input NAC PET (a, f, k), 3D-Unet sCT (b, g, l), 3D GAN sCT (c, h, m), 3D ProGAN (d, i, n) and reference CT (e, j, o), axial (a, b, c, d, e), coronal (f, g, h, i, j) and sagital (k, l, m, n, o) of the HNSCC-01-0148 sample from the validation dataset.

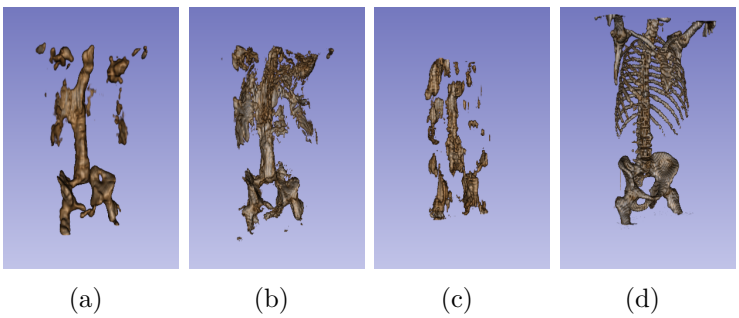


Figure 4.26: 3D renders of bone tissue, from the D-Unet sCT (a), 3D GAN sCT (b), 3D ProGAN (c) and reference CT (d) of the HNSCC-01-0148 sample from the validation dataset.

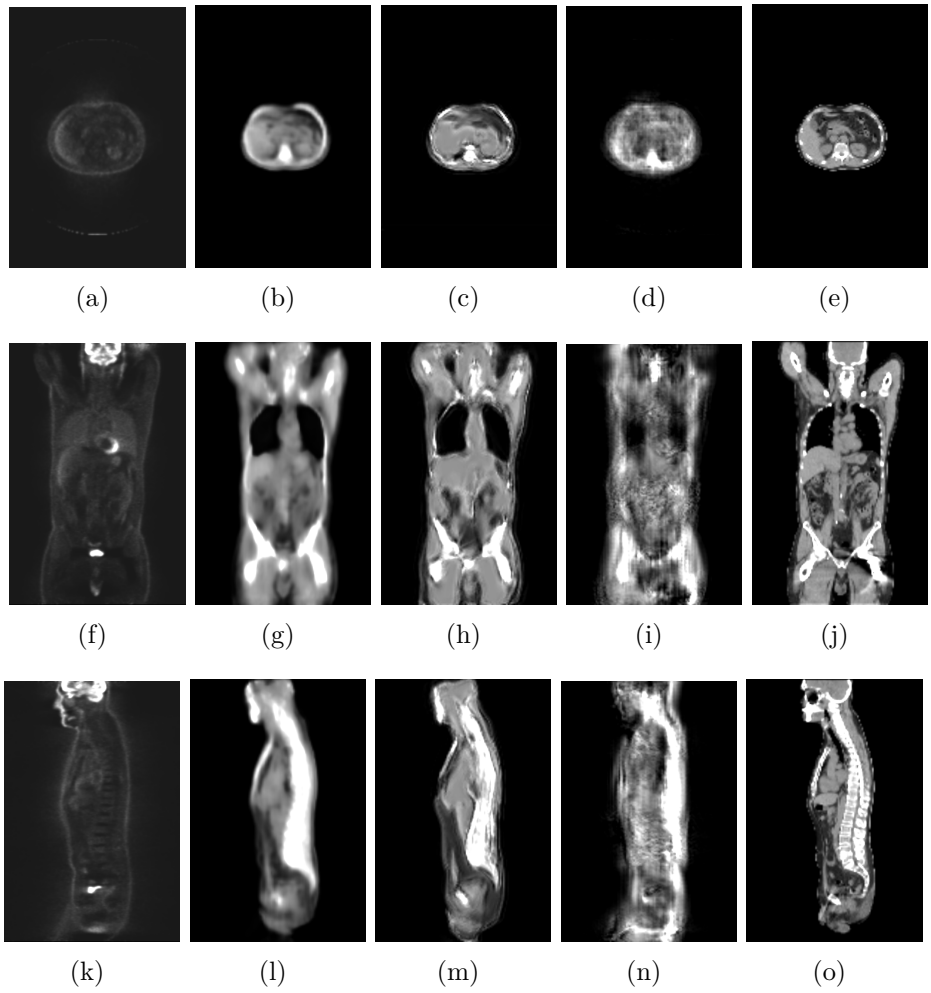


Figure 4.27: Input NAC PET (a, f, k), 3D-Unet sCT (b, g, l), 3D GAN sCT (c, h, m), 3D ProGAN (d, i, n) and reference CT (e, j, o), axial (a, b, c, d, e), coronal (f, g, h, i, j) and sagital (k, l, m, n, o) of the AMC-009 sample from NSCLC Radiogenomics.

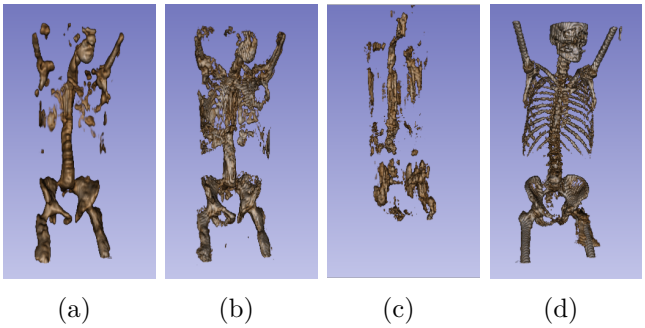


Figure 4.28: 3D renders of bone tissue, from the D-Unet sCT (a), 3D GAN sCT (b), 3D ProGAN (c) and reference CT (d) of the AMC-009 sample from NSCLC Radiogenomics.

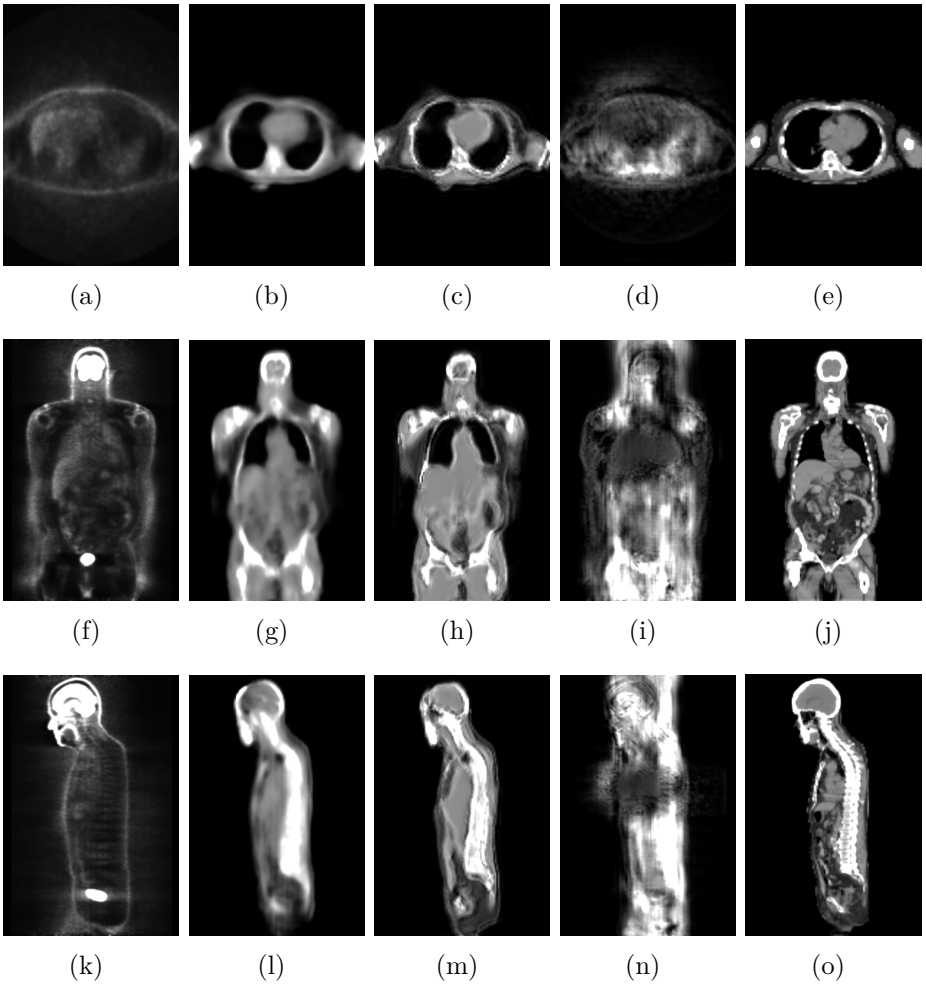


Figure 4.29: Input NAC PET (a, f, k), 3D-Unet sCT (b, g, l), 3D GAN sCT (c, h, m), 3D ProGAN (d, i, n) and reference CT (e, j, o), axial (a, b, c, d, e), coronal (f, g, h, i, j) and sagital (k, l, m, n, o) of the C3N-00957 sample from CPTAC-PDA dataset.

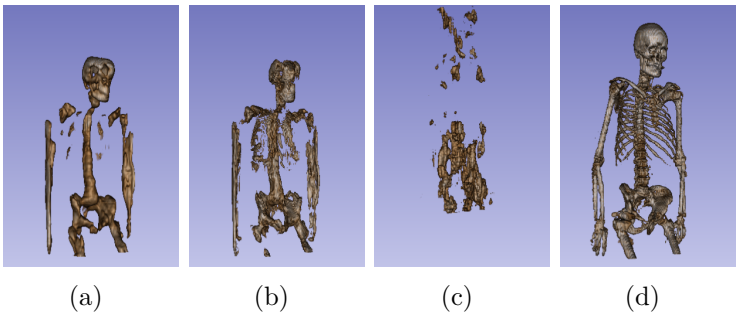


Figure 4.30: 3D renders of bone tissue, from the D-Unet sCT (a), 3D GAN sCT (b), 3D ProGAN (c) and reference CT (d) of the C3N-00957 sample from CPTAC-PDA dataset.

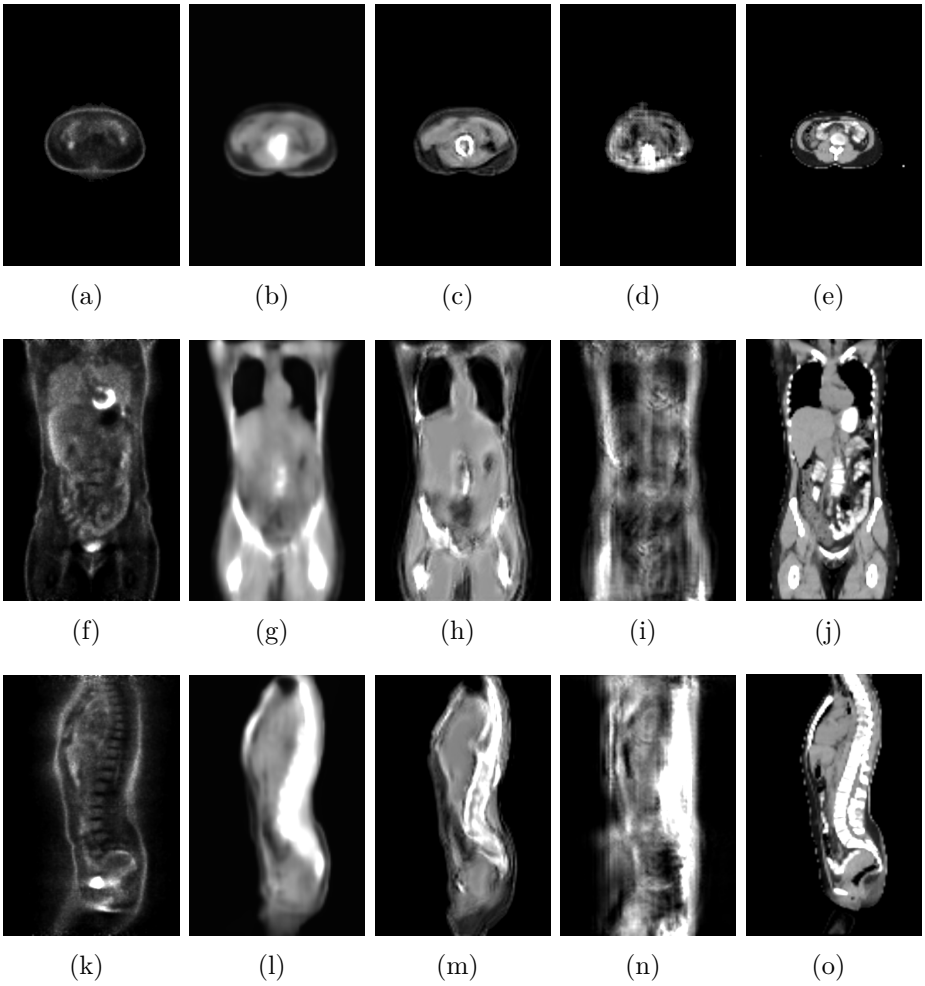


Figure 4.31: Input NAC PET (a, f, k), 3D-Unet sCT (b, g, l), 3D GAN sCT (c, h, m), 3D ProGAN (d, i, n) and reference CT (e, j, o), axial (a, b, c, d, e), coronal (f, g, h, i, j) and sagital (k, l, m, n, o) of the TCGA-BB-7863 sample from TCGA-HNSC dataset.

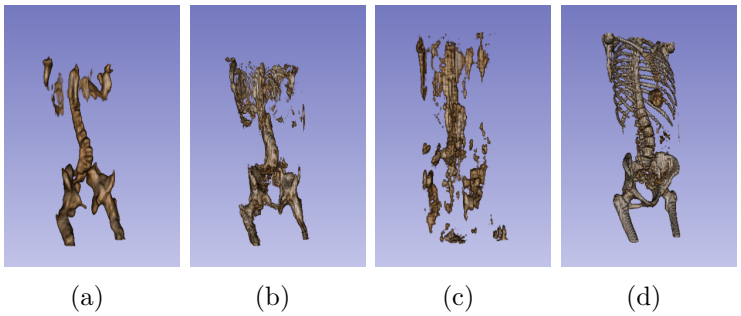


Figure 4.32: 3D renders of bone tissue, from the D-Unet sCT (a), 3D GAN sCT (b), 3D ProGAN (c) and reference CT (d) of the TCGA-BB-7863 sample from TCGA-HNSC dataset.

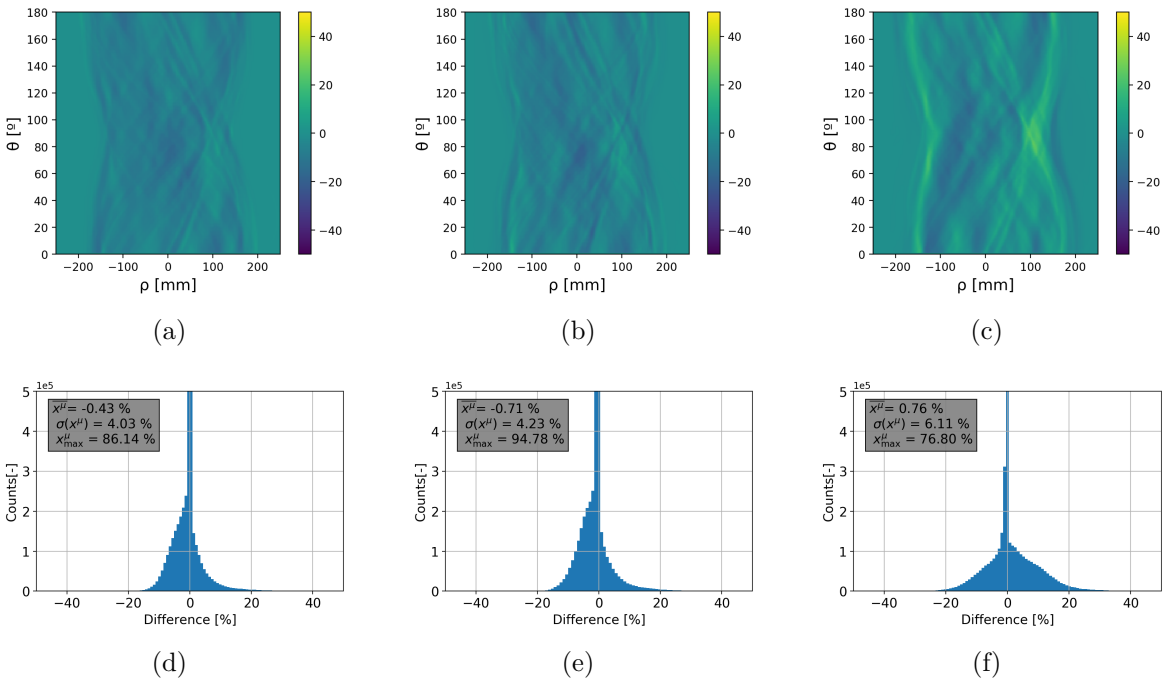


Figure 4.33: Central ring from the difference of the attenuation sinograms from the reference CT and the sCT (a,b,c) and the histograms (d,e,f) of the LoR attenuation value differences, of the 3D-Unet sCT (a,d), 3D GAN sCT (b,e) and 3D ProGAN (c,f) of the HNSCC-01-0148 sample from the validation dataset.

and absolute maximum difference:

$$x^{\mu}_{\max} = \max_{l \in L} \left| \frac{x_l^{\mu,\mathtt{sCT}} - x_l^{\mu,\mathtt{CT}}}{x_l^{\mu,\mathtt{CT}}} \right| \times 100\%. \tag{4.75}$$

The difference of the attenuation sinograms, in a central ring and the histogram of the LoR attenuation values are shown in figure 4.33 for the HNSCC-01-0148 sample from the validation dataset, and for samples AMC-009 from NSCLC Radiogenomics, C3N-00957 from CPTAC-PDA and TCGA-BB-7863 from TCGA-HNSC in figure 4.34, figure 4.35 and figure 4.36, respectively.

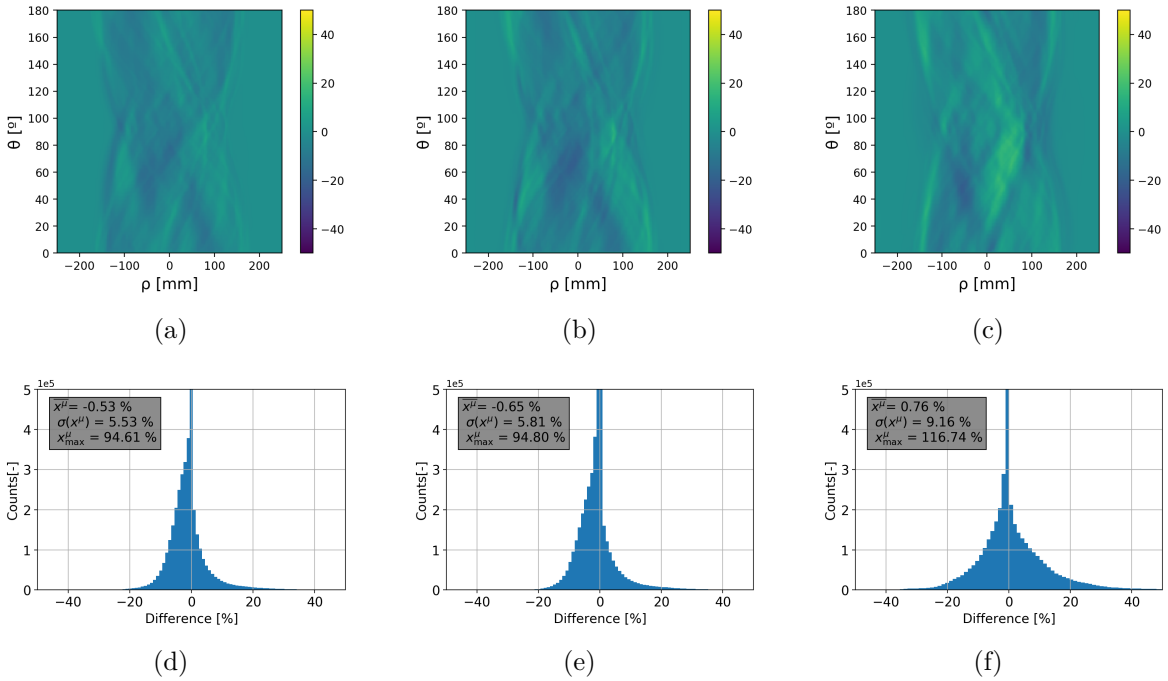


Figure 4.34: Central ring from the difference of the attenuation sinograms from the reference CT and the sCT (a,b,c) and the histograms (d,e,f) of the LoR attenuation value differences, of the 3D-Unet sCT (a,d), 3D GAN sCT (b,e) and 3D ProGAN (c,f) of the AMC-009 sample from NSCLC Radiogenomics dataset.

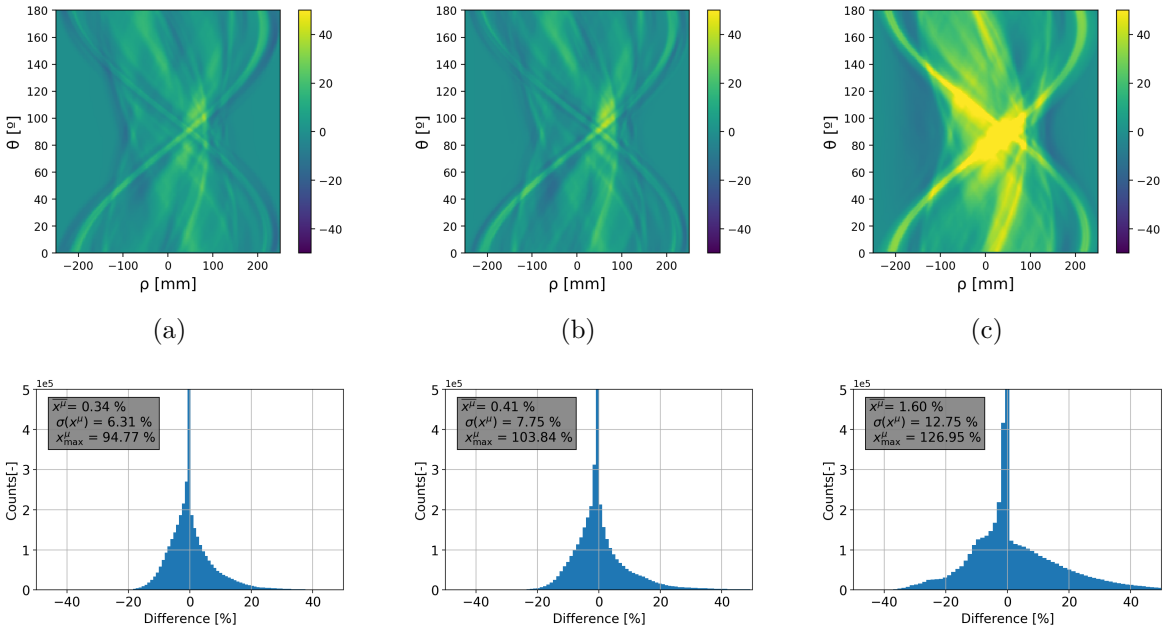


Figure 4.35: Central ring from the difference of the attenuation sinograms from the reference CT and the sCT (a,b,c) and the histograms (d,e,f) of the LoR attenuation value differences, of the 3D-Unet sCT (a,d), 3D GAN sCT (b,e) and 3D ProGAN (c,f) of the C3N-00957 sample from CPTAC-PDA dataset.

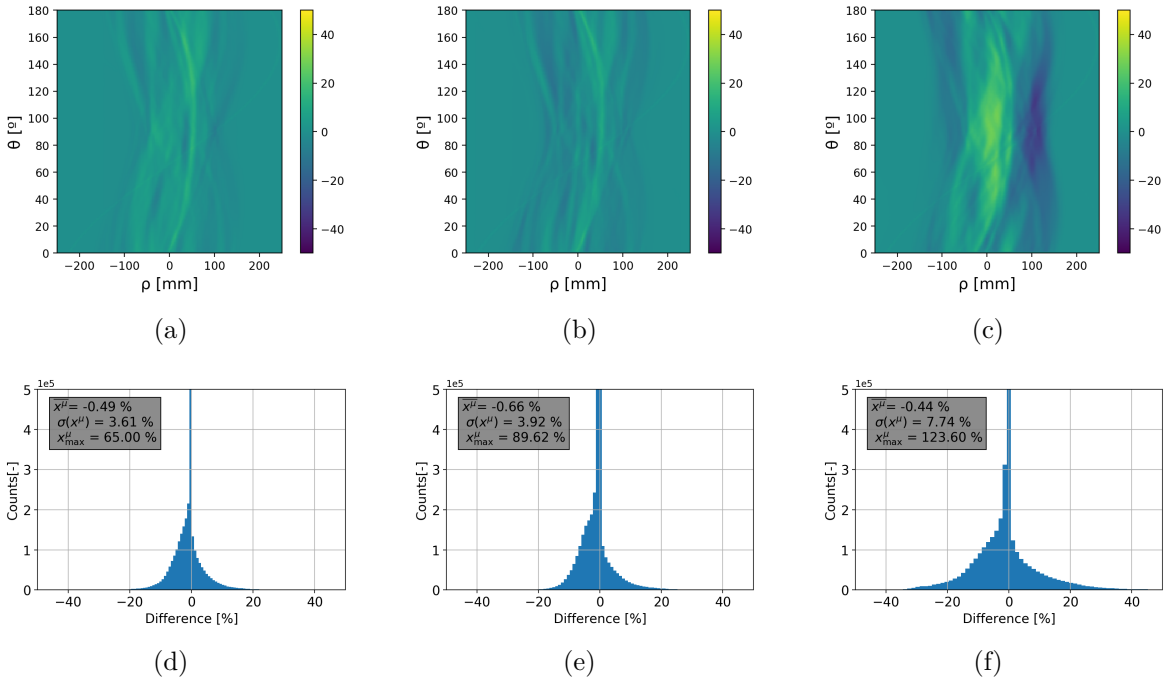


Figure 4.36: Central slice from the difference of the attenuation sinograms from the reference CT and the sCT (a,b,c) and the histograms (d,e,f) of the LoR attenuation value differences, of the 3D-Unet sCT (a,d), 3D GAN sCT (b,e) and 3D ProGAN (c,f) of the TCGA-BB-7863 sample from TCGA-HNSC dataset.

4.6 Discussion and Remarks

In this chapter the artificial generation of AM for PET image correction using NN was presented. It was shown that a NN is capable to recover AM from a NAC PET image with useful quality. Specifically three network topologies were tested, a classic 3D U-Net topology with a supervised learning approach, a GAN network based on a progressive growing scheme and Wasserstein loss and a hybrid network which mix the supervised learning gradients with a custom adversarial gradient used for image refinement. All these networks were built with a similar number of parameters, in order to equalize their capabilities. These NNs were trained and tested on public dataset showing good performance across a wide variety of PET scanner models, dose administration procedures, reconstruction techniques and pathologies. This is reflected in the metrics, resumed in figure 4.24. The basic 3D U-Net topology generates synthetic attenuation correction images with a SSIM of 0.926 ± 0.024, PSNR of 19.3 ± 1.6 dB, a MAE of 96 ± 20 HU and NCC of 0.760 ± 0.064. The addition of the GAN layers achieve a SSIM of 0.913 ± 0.024,a PSNR of 18.6 ± 1.4 dB, a MAE of 103 ± 18 HU and NCC of 0.720 ± 0.059. The ProGAN NN achieves lower scores overall, with a SSIM of 0.876 ± 0.027, PSNR of 17.4 ± 1.3 dB, a MAE of 132 ± 22 HU and NCC of 0.580 ± 0.074. These values, along those of the ablation test in table 4.3, show that a straightforward implementation of GAN losses does not perform well on problems of high dimensionality such as 3D volumes. Nevertheless the addition of a contained GAN gradient to improve the image quality is possible. The hybrid adversarial loss enables the network to learn higher frequency details, as shown in figures 4.25, 4.27, 4.29 and 4.31. It can be seen that the supervised network is unable to generate fine bone structures while the hybrid GAN network shows a more complete structure, including part of the rib cage and the sacrum and coccyx. However the ProGAN NN fails to generate good quality images and its quality is heavily affected when tested in samples from outside the HNSCC dataset. This is also reflected in the 3D plots of the bone structures, in figures 4.26, 4.28, 4.30 and 4.32. While quality improvement by the hybrid GAN NN can be observed, it is not reflected in the performance of the metrics, resulting in lower but not significantly important changes when compared to the base 3D U-Net. This illustrates the limits of simplistic metrics to reflect anatomic improvements in the image generation process. It is expected that more specific losses based on learnt representations could reflect this improvement [210].

An observed problem in all the trained topologies is that they fail to generalize the upper section of the body where less training data were available, or the arms that appear in different positions inside the dataset. This could be mitigated by anatomically matching all the training data and train region specific networks. Further improvement in this direction will be to train the network using a full-size intermediate space to map each anatomical section, such as the intermediate representations presented in [211] and [212].

The test on the attenuation sinograms present a low error, the basic 3D U-Net and the 3D GAN have an average error below 1% with a standard deviation below 8% in the tested samples, the ProGAN achieves a mean error above 1% with standard deviation reaching 12% in the validation sample. The obtained results are also comparable with

more complex NN topologies such as cycle-GANs. The presented method metrics are comparable to the method proposed by Dong [213], which achieves a MAE of 108 ± 19 HU in the reconstruction of a sCT image. In their study obtain their dataset with controlled dose administration and reconstruction procedures for the Discovery 690 PET/CT scanner (General Electric) with time of flight capabilities. Nevertheless the presented models are more compact than their proposed cycle-GAN architecture, requiring less parameters and in the presented model the impact of the adversarial gradient is controlled, resulting in a more stable training procedure. Also the presented models are tested on multiple PET scanners. Moreover, most of the tested scanners do not posses ToF capabilities, for which other methods such as MLAA, result in lower quality whole body attenuation maps.

Some aspects of the implementation of GAN generated images in praxis should be considered. As it is observed in figure 4.31(j), where CT contrast in the stomach is observed, such structure is not reproduced by any of the tested NN topologies. This is inline with [214] where they observed that NN fails to generate unseen structures or that it adds structures that are not present, resulting in erroneous images that may not be fit for clinical analysis. For this reason, until this problem can be solved, the generated AMs should be regarded more as high-quality prior for mutual activity and attenuation reconstruction method as the ones presented in section 1.2.3.

An important detail of the presented work is the use of public datasets to train the models. A known issue in the field of ML is the reproducibility of the works. Many works are presented with superficial description of the methodology and very few information on the dataset conformation. These results in very difficult reproduction of their results or even lead to the publication of overfitted models. To allow a more direct comparison of different models many ML disciplines use common databases to test their models, such as the MNIST [215] for 2D images. In medical imaging there is no consensus in a public dataset to use for ML method testing. Also, many works make use of private datasets that cannot be released due to patient privacy issues. Nevertheless TCIA has a large database of medical images, which can be exploited to create a public dataset for benchmarking medical image ML applications. In this work it is shown that a number of these datasets can be processed to successfully train and test a NN topology. Also the dataset generation pipeline and NN code is released. This enables the direct reproduction and testing of the presented models, contributing to a healthier research environment for the medical image ML field.

5 Final conclusions and perspectives

5.1 Main Discussion

Throughout this thesis different aspects of the PET tomography were studied, focusing on the enhancement of low dose scans image quality. The problem of reducing the patient dose during a PET study was analysed from three different fronts.

The AR-PET prototype

First the optimization of the AR-PET scanner was studied. This scanner prototype, designed to be low cost and to enable low income communities to access PET screening studies, features large and solid scintillation crystals and no attenuation mapping hardware. The characteristics of the AR-PET prototype played a central role in the investigation paths pursued during this thesis. Specifically, two problems were studied for this hardware: First the matching of PMTs gains with the large scintillation cameras and second the localization of gamma photon interactions within the crystals.

In section 2.3 the matching of the PMTs with the cameras was analysed and resulted in an optimization that was applied during the construction of the AR-PET prototype. In this study the gain dispersion of the PMTs under different working conditions was analysed and a simplified model of the AR-PET scintillation camera was built. This model, and an optimization procedure based on genetic algorithms, enabled the selection of groups of PMT for the construction of each of the AR-PET scintillation cameras under highly restraint conditions. The application of this technique translated on an improvement of the base performance of the cameras. While the application of this study was tested on the AR-PET scanner construction, the concept of the PMT characterization and modelling of the scintillation camera can be applied to any type of scintillation camera with solid crystals.

The localization of gamma photons within the solid crystal was studied in section 2.5, where a novel localization algorithm based on distribution matching neural networks was developed. The main contribution of this work was the development of localization algorithm with depht of interaction estimation capabilities which does not requires a complex dataset acquisition. While in current NN localization methods the dataset must contain the exact location of each event (which is normally difficult to obtain) the developed method only needs to know the spacial distribution of these events, ignoring each particular event localization. Then the method can improve the shortcomings of a base localization method by matching the distribution obtained through the base method to the expected distribution of the events. The method was applied in the correction of the event localization in the AR-PET scintillation cameras, improving

the base Anger positioning method. The sensitivity area of the cameras was improved by more than 23% and DoI estimation was added. Moreover, the NN require only a reduced amount of parameters and computational cost. The effects of embedding the NN in the localization hardware of the scintillation cameras was explored. The embedding of the neural network is possible with integer variables and word sizes of 12 or 16 `bits`. The limiting factor of quantization is the number of parameters in the NN, as larger number of parameters suffers more the quantization process. However the selection of an optimal network size for embedded operation is left to be explored.

PET imaging on mesh support

The second subject of study is the reduction of the number of parameters used to describe the radiotracer distribution in a PET image. Specifically the representation of volumetric PET images using meshes was studied. This reduction of parameters is very important for two reasons: first it could translate into an improvement of speed in the image reconstruction procedure and second it lowers the dimensionality of the data, which can be exploited in reconstruction algorithms based on optimization techniques. To test these hypothesis in section 3.3 an improved mesh based reconstruction algorithm was developed. Instead of using a 3D polyhedral mesh to describe the volume, a series of 2D meshes which represent slices of the volume were used. This was possible since the information used to reconstruct the PET images, the LoRs, is limited to certain angles, also a LoR could never go through a slice in a perpendicular angle. This form of describing the volume enables the refinement of the 2D meshes wich have lower number of nodes than 3D meshes. Moreover, the refinement of such meshes can be completely parallelized, resulting in lower processing times. The developed technique was implemented using a custom LoR tracing algorithm, used to model the AR-PET acquisition procedure and the BL2D-V2 remeshing software. The developed method was tested using an error-free simulation of a complex human brain phantom and an acquisition of a NEMA image quality phantom using the AR-PET pre-clinical prototype. These phantoms were reconstructed using a mesh based MLEM algorithm. Using these experiments it was possible to show that the application of a series of 2D meshes to describe the volumetric tracer does not results in a loss of image quality while it reduces the number of elements required to describe a volumetric image by more than one order of magnitude when compared to a voxelized support. This reduction is even more notorious when high resolution voxelized images are used. Nevertheless, using a simple MLEM reconstruction model, the quality of the images and the mesh support degrades faster than the voxelized support when fewer number of LoRs are used in the reconstruction procedure. However, the reduction in the number of nodes can be used to improve the model quality, at lower costs than the voxelized support, and thus leverage the effects of low statistics during the reconstruction. Finally a Single Scatter Simulation (SSS) algorithm was implemented in mesh support in section 3.4. It was possible to show that the SSS can be implemented in mesh support, by changing the voxel sampling with a scatter-angle sampling strategy.

Attenuation Map generation using machine learning methods

The third and final subject of study was the generation of attenuation maps for PET image reconstruction correction using ML methods. This chapter focused on analyzing different strategies to generate an AM using only the NAC PET images applying state of the art generative tools from the field of ML. To explore the ML methods a dataset of NAC PET images and their co-registered CT scans was built in section 4.5.1. The dataset was created using publicly available images from TCIA. The chosen topology to implement the AM generation was a deep neural network based on the 3D version of the U-Net. Three variations of this topology were studied in section 4.5.2, each of them trained using a distinct approach: a fully adversarial training, a fully supervised training and a combination of these two. In the fully adversarial training approach two topologies were tested, a 3D U-Net and a progressive growing topology, designed to reduce training instabilities in large images. Despite the widespread positive results of GANs their direct implementation on 3D images, with restricted number of samples fell short against other methods. The fully adversarial training did not generate good quality images. The fully supervised method was able to generate AM with relative good quality and it was also able to segment different tissue labels. Nevertheless the generated images presented a loss of high frequency details. This was expected since the optimization procedure used a L2 distance metric between the objective and generated CT. To improve the quality of the fully supervised method a hybrid network was built. This network combined the supervised training with a restricted adversarial loss used to enhance the high frequency details of the image. As a result the generated images presented more realistic features without loosing the specificity of the supervised training. The use of the hybrid training strategy achieved state of the art scores. Moreover the network was tested on different PET scanner models, dose administration procedures and image reconstruction processes, showing good results overall. The generated AMs quality is high enough to be directly applied for PET image correction, according to a recent study. However it was observed that some foreign or unusual structures in the CT images (e.g. CT contrast, prostheses or calcifications) can be mistakenly removed from the generated CT images. For this reason it is advised to refine the generated AM. Such refinement can be performed using simultaneous activity and attenuation reconstruction methods that use the generated AM as prior.

5.2 Perspectives and future work

The path towards a low cost PET system without additional dose from CT scans is long and in this thesis some steps towards this goal were made. Nevertheless many are yet to be done. The main foundations of an improved activity and attenuation reconstruction algorithm were developed here, an optimized representation and reconstruction method for PET images and an image based high quality AM prior generation. While it was not possible to further develop the potential of mesh representation for simultaneous reconstruction of activity distribution and attenuation map (due to time constraints), the basis of activity distribution reconstruction and SSS on meshes was

developed, along with a high quality AM prior generation using deep neural networks. These works are the fundamental parts of a simultaneous activity and attenuation reconstruction algorithm. Extension of these works includes the extension of the reconstruction model to for simultaneous AM generation, implementing a mesh based MLAA, following [11,51,52]. An other promising path is the optimization of the SSS to match lower energy (scattered) coincidences and using a high quality AM prior to reduce convergence problems [54, 55]. Also, exploiting the reduced dimensionality of the mesh representation the reconstruction models can be enhanced or substituted with more realistic models in order to enhance the quality of low statistics PET images [19].

The AM generation through ML methods can also be improved in several aspects. The generation models benefit greatly from uniform data. This is part of the success of large generative networks such as [216]. Histologically matching the samples of the dataset may boost the training efficiency and hence the quality of the generated AMs. Other improvement of the model can be the use of an intermediate full resolution latent space from which the generative model can sample. This is proposed in [212] where they implement such latent space to improve 3D representation of images. Also the development of new tools for analysing 3D data, such as [172], should be tested. These tools are enabling rapid implementation of 3D models that analyse voxelized or point cloud data. Point cloud analysis, such as [167–170], can be directly implemented over the mesh nodes of the PET image representation, exploiting the reduced dimensionality and avoiding projection errors.

Finally, it is also expected that the AM generation could be tested on patient data from the AR-PET scanner. The work presented in this thesis helped the construction and testing of the AR-PET. The scanner prototype is currently under test and awaiting acceptance from the National Administration of Drugs, Foods and Medical Devices (Administración Nacional de Medicamentos, Alimentos y Tecnología Médica, ANMAT) in Argentina to begin testing on patients.

Bibliography

[1] R. Minamimoto, M. Senda, S. Jinnouchi, T. Terauchi, T. Yoshida, T. Murano, H. Fukuda, T. Iinuma, K. Uno, S. Nishizawa, *et al.*, "The current status of an fdg-pet cancer screening program in japan, based on a 4-year (2006–2009) nationwide survey," *Annals of nuclear medicine*, vol. 27, no. 1, pp. 46–57, 2013.

[2] M. Ide and Y. Suzuki, "Is whole-body fdg-pet valuable for health screening?," *European journal of nuclear medicine and molecular imaging*, vol. 32, no. 3, pp. 339–341, 2005.

[3] N. Ghotbi, M. Iwanaga, A. Ohtsuru, Y. Ogawa, S. Yamashita, *et al.*, "Cancer screening with whole-body pet/ct for healthy asymptomatic people in japan: re-evaluation of its test validity and radiation exposure," *Asian Pacific Journal of Cancer Prevention*, vol. 8, no. 1, p. 93, 2007.

[4] A. Kaushik, A. Jaimini, M. Tripathi, M. D'Souza, R. Sharma, A. Mondal, A. K. Mishra, and B. S. Dwarakanath, "Estimation of radiation dose to patients from 18fdg whole body pet/ct investigations using dynamic pet scan protocol," *The Indian journal of medical research*, vol. 142, no. 6, p. 721, 2015.

[5] J. Valentin *et al.*, *The 2007 recommendations of the international commission on radiological protection*, vol. 37. Elsevier Oxford, 2007.

[6] M. Weckesser and O. Schober, "Is whole-body fdg-pet valuable for health screening?," *European journal of nuclear medicine and molecular imaging*, vol. 32, no. 3, pp. 342–343, 2005.

[7] N. Jallow, P. Christian, J. Sunderland, M. Graham, J. M. Hoffman, and J. A. Nye, "Diagnostic reference levels of ct radiation dose in whole-body pet/ct," *Journal of Nuclear Medicine*, vol. 57, no. 2, pp. 238–241, 2016.

[8] D. J. Brenner and E. J. Hall, "Computed tomography—an increasing source of radiation exposure," *New England Journal of Medicine*, vol. 357, no. 22, pp. 2277–2284, 2007.

[9] E. Prieto, M. J. García-Velloso, M. Rodríguez-Fraile, V. Morán, B. García-García, F. Guillén, M. I. Morales, L. Sancho, I. Peñuelas, J. Á. Richter, *et al.*, "Significant dose reduction is feasible in fdg pet/ct protocols without compromising diagnostic quality," *Physica Medica*, vol. 46, pp. 134–139, 2018.

[10] T.-H. Wu, T.-C. Chu, Y.-H. Huang, L.-K. Chen, S.-P. Mok, J.-K. Lee, T. Yeu-Sheng, and J. J. Lee, "A positron emission tomography/computed tomography (pet/ct) acquisition protocol for ct radiation dose optimization," *Nuclear medicine communications*, vol. 26, no. 4, pp. 323–330, 2005.

[11] A. Mihlin and C. S. Levin, "Gpu formulated mlem joint estimation of emission activity and photon attenuation in positron emission tomography," in *2014 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, pp. 1–2, IEEE, 2014.

[12] L. Pang, W. Zhu, Y. Dong, Y. Lv, and H. Shi, "Zero-extra-dose pet delayed imaging with data-driven attenuation correction estimation," *Molecular Imaging and Biology*, vol. 21, no. 1, pp. 149–158, 2019.

[13] F. Liu, H. Jang, R. Kijowski, G. Zhao, T. Bradshaw, and A. B. McMillan, "A deep learning approach for 18 f-fdg pet attenuation correction," *EJNMMI physics*, vol. 5, no. 1, p. 24, 2018.

[14] X. Dong, Y. Lei, T. Wang, K. Higgins, T. Liu, W. J. Curran, H. Mao, J. A. Nye, and X. Yang, "Deep learning-based attenuation correction in the absence of structural information for whole-body pet imaging," *Physics in Medicine & Biology*, 2019.

[15] K. Armanious, T. Hepp, T. Küstner, H. Dittmann, K. Nikolaou, C. La Fougère, B. Yang, and S. Gatidis, "Independent attenuation correction of whole body [18 f] fdg-pet using a deep learning approach with generative adversarial networks," *EJNMMI research*, vol. 10, no. 1, pp. 1–9, 2020.

[16] K. Armanious, C. Yang, M. Fischer, T. Küstner, K. Nikolaou, S. Gatidis, and B. Yang, "Medgan: medical image translation using gans," *arXiv preprint arXiv:1806.06397*, 2018.

[17] D. F. Bauer, A.-K. Schnurr, T. Russ, S. Goerttler, L. R. Schad, F. G. Zoellner, and K. Chung, "Synthesis of ct images using cyclegans: Enhancement of anatomical accuracy," in *Procedings of 2019 Medical Imaging with Deep Learning Conference (OpenReview.net)*, 2019.

[18] F. Gao, J. Xu, H. Liu, and P. Shi, "Efficient point clouds based scatter correction for fully 3d pet," in *IEEE Nuclear Science Symposuim & Medical Imaging Conference*, pp. 2931–2934, IEEE, 2010.

[19] M. D. Walker, M. Asselin, P. J. Julyan, M. Feldmann, P. Talbot, T. Jones, and J. Matthews, "Bias in iterative reconstruction of low-statistics pet data: benefits of a resolution model," *Physics in Medicine & Biology*, vol. 56, no. 4, p. 931, 2011.

[20] H. Zaidi, "Comparative evaluation of scatter correction techniques in 3d positron emission tomography," *European journal of nuclear medicine*, vol. 27, no. 12, pp. 1813–1826, 2000.

[21] R. G. Rodríguez Colmeiro and C. A. Verrastro, "Heuristic method for photodetectors localization over continuous crystal scintillation cameras," *Journal of Computer Science & Technology*, vol. 18, 2018.

[22] R. R. Colmeiro, C. Verrastro, D. Minsky, and T. Grosges, "Event localization in continuous crystal scintillation cameras using distribution matching neural networks," in *2021 Argentine Conference on Electronics (CAE)*, pp. 7–12, IEEE, 2021.

[23] R. R. Colmeiro, C. Verrastro, D. Minsky, and T. Grosges, "Reconstruction of positron emission tomography images using adaptive sliced remeshing strategy," *Journal of Medical Imaging*, vol. 8, no. 2, pp. 1 – 18, 2021.

[24] R. R. Colmeiro, C. Verrastro, and T. Grosges, "Multimodal brain tumor segmentation using 3d convolutional networks," in *International MICCAI Brainlesion Workshop, Lecture Notes in Computer Science*, vol. 10670, pp. 226–240, Springer, 2018.

[25] R. G. Rodríguez Colmeiro, C. Verrastro, D. Minsky, and T. Grosges, "Towards a whole body [18f] fdg positron emission tomography attenuation correction map synthesizing using deep neural networks," *Journal of Computer Science and Technology*, vol. 21, p. e4, Apr. 2021.

[26] M. Hatanaka, "Transport of sugars in tumor cell membranes," *Biochimica et Biophysica Acta (BBA)-Reviews on Cancer*, vol. 355, no. 1, pp. 77–104, 1974.

[27] R. Kubota, K. Kubota, S. Yamada, M. Tada, T. Ido, and N. Tamahashi, "Microautoradiographic study for the differentiation of intratumoral macrophages, granulation tissues and cancer cells by the dynamics of fluorine-18-fluorodeoxyglucose uptake," *Journal of Nuclear Medicine*, vol. 35, no. 1, pp. 104–112, 1994.

[28] K. Kubota, "From tumor biology to clinical pet: a review of positron emission tomography (pet) in oncology," *Annals of nuclear medicine*, vol. 15, no. 6, pp. 471–486, 2001.

[29] S. Vandenberghe, E. Mikhaylova, E. D'Hoe, P. Mollet, and J. S. Karp, "Recent developments in time-of-flight pet," *EJNMMI physics*, vol. 3, no. 1, p. 3, 2016.

[30] T. Tomitani, "Image reconstruction and noise evaluation in photon time-of-flight assisted positron emission tomography," *IEEE Transactions on Nuclear Science*, vol. 28, no. 6, pp. 4581–4589, 1981.

[31] E. Berg and S. R. Cherry, "Innovations in instrumentation for positron emission tomography," in *Seminars in nuclear medicine*, vol. 48, pp. 311–331, Elsevier, 2018.

[32] S. Surti, J. Karp, and G. Muehllehner, "Evaluation of pixelated nai (tl) detectors for pet," *IEEE Transactions on Nuclear Science*, vol. 50, no. 1, pp. 24–31, 2003.

[33] S. España, L. Fraile, J. Herraiz, J. M. Udías, M. Desco, and J. Vaquero, "Performance evaluation of sipm photodetectors for pet imaging in the presence of magnetic fields," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 613, no. 2, pp. 308–316, 2010.

[34] D. Renker, "New trends on photodetectors," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 571, no. 1-2, pp. 1–6, 2007.

[35] M. Ruiz-Gonzalez, V. Bora, and L. R. Furenlid, "Maximum-likelihood estimation of scintillation pulse timing," *IEEE transactions on radiation and plasma medical sciences*, vol. 2, no. 1, pp. 1–6, 2017.

[36] E. Berg and S. R. Cherry, "Using convolutional neural networks to estimate time-of-flight from pet detector waveforms," *Physics in Medicine & Biology*, vol. 63, no. 2, p. 02LT01, 2018.

[37] P. Conde, A. Iborra, A. González, L. Hernández, P. Bellido, L. Moliner, J. Rigla, M. Rodríguez-Álvarez, F. Sánchez, M. Seimetz, *et al.*, "Determination of the interaction position of gamma photons in monolithic scintillators using neural network fitting," *IEEE Transactions on Nuclear Science*, vol. 63, no. 1, pp. 30–36, 2016.

[38] C. W. Lerche, A. Ros, J. Monzó, R. Aliaga, N. Ferrando, J. Martínez, V. Herrero, R. Esteve, R. Gadea, R. Colom, *et al.*, "Maximum likelihood positioning for gamma-ray imaging detectors with depth of interaction measurement," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 604, no. 1-2, pp. 359–362, 2009.

[39] G. Borghi, V. Tabacchini, and D. R. Schaart, "Towards monolithic scintillator based tof-pet systems: practical methods for detector calibration and operation," *Physics in Medicine & Biology*, vol. 61, no. 13, p. 4904, 2016.

[40] M. Morrocchi, W. Hunter, A. Del Guerra, T. Lewellen, P. Kinahan, L. MacDonald, M. G. Bisogni, and R. Miyaoka, "Evaluation of event position reconstruction in monolithic crystals that are optically coupled," *Physics in Medicine & Biology*, vol. 61, no. 23, p. 8298, 2016.

[41] M. Ito, S. J. Hong, and J. S. Lee, "Positron emission tomography (pet) detectors with depth-of-interaction (doi) capability," *Biomedical Engineering Letters*, vol. 1, no. 2, p. 70, 2011.

[42] H. Zaidi and B. Hasegawa, "Determination of the attenuation map in emission tomography," *Journal of Nuclear Medicine*, vol. 44, no. 2, pp. 291–315, 2003.

[43] A. C. Kak, M. Slaney, and G. Wang, "Principles of computerized tomographic imaging," *Medical Physics*, vol. 29, no. 1, pp. 107–107, 2002.

[44] C. Burger, G. Goerres, S. Schoenes, A. Buck, A. Lonn, and G. Von Schulthess, "Pet attenuation coefficients from ct images: experimental evaluation of the transformation of ct into pet 511-kev attenuation coefficients," *European journal of nuclear medicine and molecular imaging*, vol. 29, no. 7, pp. 922–927, 2002.

[45] Y. Nakamoto, M. Osman, C. Cohade, L. T. Marshall, J. M. Links, S. Kohlmyer, and R. L. Wahl, "Pet/ct: comparison of quantitative tracer uptake between germanium and ct transmission attenuation-corrected images," *Journal of Nuclear Medicine*, vol. 43, no. 9, pp. 1137–1143, 2002.

[46] T. Pan, O. Mawlawi, S. A. Nehmeh, Y. E. Erdi, D. Luo, H. H. Liu, R. Castillo, R. Mohan, Z. Liao, and H. Macapinlac, "Attenuation correction of pet images with respiration-averaged ct images in pet/ct," *Journal of Nuclear Medicine*, vol. 46, no. 9, pp. 1481–1487, 2005.

[47] D. W. McRobbie, E. A. Moore, M. J. Graves, and M. R. Prince, *MRI from Picture to Proton*. Cambridge university press, 2017.

[48] M. Hofmann, I. Bezrukov, F. Mantlik, P. Aschoff, F. Steinke, T. Beyer, B. J. Pichler, and B. Schölkopf, "Mri-based attenuation correction for whole-body pet/mri: quantitative evaluation of segmentation-and atlas-based methods," *Journal of Nuclear Medicine*, vol. 52, no. 9, pp. 1392–1399, 2011.

[49] D. Nie, X. Cao, Y. Gao, L. Wang, and D. Shen, "Estimating ct image from mri data using 3d fully convolutional networks," in *Deep Learning and Data Labeling for Medical Applications*, pp. 170–178, Springer, 2016.

[50] B. Stimpel, C. Syben, T. Würfl, K. Mentl, A. Dörfler, and A. Maier, "Mr to x-ray projection image synthesis," *arXiv preprint arXiv:1710.07498*, 2017.

[51] J. Nuyts, P. Dupont, S. Stroobants, R. Benninck, L. Mortelmans, and P. Suetens, "Simultaneous maximum a posteriori reconstruction of attenuation and activity distributions from emission sinograms," *IEEE transactions on medical imaging*, vol. 18, no. 5, pp. 393–403, 1999.

[52] V. Y. Panin, G. L. Zeng, and G. T. Gullberg, "A method of attenuation map and emission activity reconstruction from emission data," *IEEE Transactions on Nuclear Science*, vol. 48, no. 1, pp. 131–138, 2001.

[53] A. Rezaei, M. Defrise, G. Bal, C. Michel, M. Conti, C. Watson, and J. Nuyts, "Simultaneous reconstruction of activity and attenuation in time-of-flight pet," *IEEE transactions on Medical Imaging*, vol. 31, no. 12, pp. 2224–2233, 2012.

[54] S. C. Cade, S. Arridge, M. J. Evans, and B. F. Hutton, "Use of measured scatter data for the attenuation correction of single photon emission tomography without transmission scanning," *Medical physics*, vol. 40, no. 8, p. 082506, 2013.

[55] Y. Berker, F. Kiessling, and V. Schulz, "Scattered pet data for attenuation-map reconstruction in pet/mri," *Medical physics*, vol. 41, no. 10, p. 102502, 2014.

[56] L. Brusaferri, A. Bousse, N. Efthimiou, E. Emond, D. Atkinson, S. Ourselin, B. F. Hutton, S. Arridge, and K. Thielemans, "Potential benefits of incorporating energy information when estimating attenuation from pet data," in *2017 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, pp. 1–4, IEEE, 2017.

[57] A. P. Leynes, J. Yang, F. Wiesinger, S. S. Kaushik, D. D. Shanbhag, Y. Seo, T. A. Hope, and P. E. Larson, "Direct pseudoct generation for pelvis pet/mri attenuation correction using deep convolutional neural networks with multi-parametric mri: zero echo-time and dixon deep pseudoct (zedd-ct)," *Journal of Nuclear Medicine*, pp. jnumed–117, 2017.

[58] A. Sitek and A. M. Celler, "Limitations of poisson statistics in describing radioactive decay," *Physica Medica*, vol. 31, no. 8, pp. 1105–1107, 2015.

[59] B. Bendriem and D. W. Townsend, *The theory and practice of 3D PET*, vol. 32. Springer Science & Business Media, 2013.

[60] D. E. Dudgeon and R. M. Mersereau., *Multidimensional digital signal processing.* Prentice-Hall, Englewood Cliffs, NJ, 1995.

[61] P. A. Toft, "The radon transform-theory and implementation," 1996.

[62] R. L. Siddon, "Fast calculation of the exact radiological path for a three-dimensional ct array," *Medical physics*, vol. 12, no. 2, pp. 252–255, 1985.

[63] L. A. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE transactions on medical imaging*, vol. 1, no. 2, pp. 113–122, 1982.

[64] V. E. Johnson, "A note on stopping rules in em-ml reconstructions of ect images," *IEEE transactions on medical imaging*, vol. 13, no. 3, pp. 569–571, 1994.

[65] H. M. Hudson and R. S. Larkin, "Accelerated image reconstruction using ordered subsets of projection data," *IEEE transactions on medical imaging*, vol. 13, no. 4, pp. 601–609, 1994.

[66] T. Hebert and R. Leahy, "A generalized em algorithm for 3-d bayesian reconstruction from poisson data using gibbs priors," *IEEE transactions on medical imaging*, vol. 8, no. 2, pp. 194–202, 1989.

[67] A. R. De Pierro, "A modified expectation maximization algorithm for penalized likelihood estimation in emission tomography," *IEEE transactions on medical imaging*, vol. 14, no. 1, pp. 132–137, 1995.

[68] G. Gindi, M. Lee, A. Rangarajan, and I. G. Zubal, "Bayesian reconstruction of functional images using anatomical information as priors," *IEEE Transactions on Medical Imaging*, vol. 12, no. 4, pp. 670–680, 1993.

[69] D. Brasse, P. E. Kinahan, C. Lartizien, C. Comtat, M. Casey, and C. Michel, "Correction methods for random coincidences in fully 3d whole-body pet: impact on data and image quality," *Journal of nuclear medicine*, vol. 46, no. 5, pp. 859–867, 2005.

[70] T. Lewellen, R. Miyaoka, and W. Swan, "Pet imaging using dual-headed gamma cameras: an update.," *Nuclear medicine communications*, vol. 20, no. 1, pp. 5–12, 1999.

[71] S. Vandenberghe, Y. Asseler, M. Koole, L. Bouwens, R. Van de Walle, I. Lemahieu, and R. Dierckx, "Randoms correction for gamma camera based pet list-mode reconstruction," in *2001 IEEE Nuclear Science Symposium Conference Record (Cat. No. 01CH37310)*, vol. 4, pp. 2031–2035, IEEE, 2001.

[72] Y. Yamakawa, K. Kitamura, Y. Yamada, N. Hashizume, A. Kawashima, and Y. Kumazawa, "Random correction using singles count rates for doi positron emission mammography," in *2008 IEEE Nuclear Science Symposium Conference Record*, pp. 5129–5132, IEEE, 2008.

[73] M. A. King, G. J. Hademenos, and S. J. Glick, "A dual-photopeak window method for scatter correction," *J Nucl Med*, vol. 33, no. 4, pp. 605–12, 1992.

[74] L.-E. Adam, J. Karp, and R. Freifelder, "Scatter correction using a dual energy window technique for 3d pet with nai (tl) detectors," in *1998 IEEE Nuclear Science Symposium Conference Record. 1998 IEEE Nuclear Science Symposium and Medical Imaging Conference (Cat. No. 98CH36255)*, vol. 3, pp. 2011–2018, IEEE, 1998.

[75] D. L. Bailey and S. R. Meikle, "A convolution-subtraction scatter correction method for 3d pet," *Physics in medicine and biology*, vol. 39, no. 3, p. 411, 1994.

[76] C. C. Watson, D. Newport, and M. E. Casey, "A single scatter simulation technique for scatter correction in 3d pet," in *Three-dimensional image reconstruction in radiology and nuclear medicine*, pp. 255–268, Springer, 1996.

[77] S. C. Cade, S. Arridge, M. J. Evans, and B. F. Hutton, "Attenuation map estimation without transmission scanning using measured scatter data," in *2011 IEEE Nuclear Science Symposium Conference Record*, pp. 2657–2663, IEEE, 2011.

[78] C. C. Watson, "New, faster, image-based scatter correction for 3d pet," *IEEE Transactions on Nuclear Science*, vol. 47, no. 4, pp. 1587–1594, 2000.

[79] G. F. Knoll, *Radiation detection and measurement*. John Wiley & Sons, 2010.

[80] C. Tsoumpas, "Implementation and evaluation of scatter estimation algorithms in positron emission tomography," *Interdepartmental Post-graduate Course in Biomedical Engineering, University of Patras and National Technical*, 2004.

[81] C. Verrastro, M. Cabrejas, D. Estryk, E. Venialgo, S. Marinsek, M. Belzunce, J. Zalcman, L. Giuliodori, R. Barneda, F. Carmona, *et al.*, "Ar-pet: Primer tomógrafo por emisión de positrones argentino," *Actas de la XXXV Reunión Anual de la Asociación Argentina de Tecnología Nuclear*, 2008.

[82] C. Verrastro, M. Belzunce, J. C. Gomez, D. Estryk, E. Venialgo, F. Carmona, and D. De Biase, "De la primera radiografía al primer tomógrafo por emisión de positrones argentino," *Proyecciones*, vol. 7, no. 1, pp. 49–76, 2009.

[83] D. Páez, P. Orellana, C. Gutiérrez, R. Ramirez, F. Mut, and L. Torres, "Current status of nuclear medicine practice in latin america and the caribbean," *Journal of Nuclear Medicine*, vol. 56, no. 10, pp. 1629–1634, 2015.

[84] S. Espana, R. Marcinkowski, V. Keereman, S. Vandenberghe, and R. Van Holen, "Digipet: sub-millimeter spatial resolution small-animal pet imaging using thin monolithic scintillators," *Physics in Medicine & Biology*, vol. 59, no. 13, p. 3405, 2014.

[85] D. Estryk, C. Verrastro, S. Marinsek, M. Belzunce, and E. Venialgo, "Fpga hierarchical architecture for a positron emission tomography scanner," in *2010 VI Southern Programmable Logic Conference (SPL)*, pp. 113–118, IEEE, 2010.

[86] HAMAMATSU PHOTONICS K.K, "Photomultiplier tubes," 2007.

[87] Y. Liu, H. Li, Y. Wang, T. Xing, S. Xie, J. Uribe, H. Baghaei, R. Ramirez, S. Kim, and W.-H. Wong, "A gain-programmable transit-time-stable and temperature-stable pmt voltage divider," in *Nuclear Science Symposium Conference Record, 2003 IEEE*, vol. 5, pp. 3101–3104, IEEE, 2003.

[88] G. Knoll and M. Schrader, "Computer correction of camera nonidealities in gamma ray imaging," *IEEE Transactions on Nuclear Science*, vol. 29, no. 4, pp. 1271–1279, 1982.

[89] Y. Wang, W.-H. Wong, M. Aykac, J. Uribe, H. Li, H. Baghaei, Y. Liu, and T. Xing, "An iterative energy-centroid method for recalibration of pmt gain in pet or gamma camera," in *Nuclear Science Symposium Conference Record, 2001 IEEE*, vol. 4, pp. 1965–1968, IEEE, 2001.

[90] H. Li, W.-H. Wong, Y. Wang, Y. Liu, T. Xing, J. Uribe, H. Baghaei, and R. Farrell, "Front-end electronics based on high-yield-pileup-event-recovery method for a high resolution pet camera with pmt-quadrant-sharing detector modules," in *Nuclear Science Symposium Conference Record, 2002 IEEE*, vol. 2, pp. 699–703, IEEE, 2002.

[91] H. Li, Y. Liu, T. Xing, Y. Wang, J. Uribe, H. Baghaei, S. Xie, S. Kim, R. Ramirez, and W.-H. Wong, "An instantaneous photomultiplier gain calibration method for pet or gamma camera detectors using an led network," in *Nuclear Science Symposium Conference Record, 2003 IEEE*, vol. 4, pp. 2447–2451, IEEE, 2003.

[92] E. Stoub, "Method and circuit for stabilizing conversion gain of radiation detectors of a radiation detection system," Apr. 15 1986. US Patent 4,583,187.

[93] T. E. Peterson and L. R. Furenlid, "Spect detectors: the anger camera and beyond," *Physics in medicine and biology*, vol. 56, no. 17, p. R145, 2011.

[94] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[95] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*. John Wiley & Sons, 2012.

[96] J. H. Holland *et al.*, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

[97] E. Venialgo, C. Verrastro, D. Estryk, and M. Belzunce, "Método de calibración para cámara gamma y pet con una medición única de campo inundado," *XIII reunión de Trabajo en Procesamiento de la Información y Control*, pp. 1010–1015, 2009.

[98] M. Casey, Z. Burbar, H. Rothfuss, V. Panin, and D. Bharkhada, "A next generation sipm based pet/ct system with improved time and spatial resolution," *Journal of Nuclear Medicine*, vol. 58, no. supplement 1, pp. 1332–1332, 2017.

[99] F. Müller, D. Schug, P. Hallen, J. Grahe, and V. Schulz, "Gradient tree boosting-based positioning method for monolithic scintillator crystals in positron emission tomography," *IEEE Transactions on Radiation and Plasma Medical Sciences*, vol. 2, no. 5, pp. 411–421, 2018.

[100] P. Bruyndonckx, S. Leonard, S. Tavernier, C. Lemaitre, O. Devroede, Y. Wu, and M. Krieguer, "Neural network-based position estimators for pet detectors using monolithic lso blocks," *IEEE Transactions on Nuclear Science*, vol. 51, no. 5, pp. 2520–2525, 2004.

[101] E. Venialgo, C. Verrastro, D. Estryk, M. Belzunce, A. Carimatto, E. da Ponte, L. M. Garbino, and J. Alarcon, "Pet calibration method of nonlinear position estimation algorithms for continuous nai (tl) crystals," in *2011 IEEE Nuclear Science Symposium Conference Record*, pp. 3359–3364, IEEE, 2011.

[102] A. Iborra, A. González, A. González-Montoro, A. Bousse, and D. Visvikis, "Ensemble of neural networks for 3d position estimation in monolithic pet detectors," *Physics in Medicine & Biology*, vol. 64, no. 19, p. 195010, 2019.

[103] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *International journal of computer vision*, vol. 40, no. 2, pp. 99–121, 2000.

[104] A. Genevay, M. Cuturi, G. Peyré, and F. Bach, "Stochastic optimization for large-scale optimal transport," in *Advances in neural information processing systems*, pp. 3440–3448, 2016.

[105] H. Fan, H. Su, and L. J. Guibas, "A point set generation network for 3d object reconstruction from a single image," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 605–613, 2017.

[106] C. W. Lerche, J. Benlloch, F. Sanchez, N. Pavon, B. Escat, E. Gimenez, M. Fernandez, I. Torres, M. Gimenez, A. Sebastia, *et al.*, "Depth of/spl gamma/-ray interaction within continuous crystals from the width of its scintillation light-distribution," *IEEE Transactions on Nuclear Science*, vol. 52, no. 3, pp. 560–572, 2005.

[107] D. Strulab, G. Santin, D. Lazaro, V. Breton, and C. Morel, "Gate (geant4 application for tomographic emission): a pet/spect general-purpose simulation platform," *Nuclear Physics B-Proceedings Supplements*, vol. 125, pp. 75–79, 2003.

[108] L. Devroye, "Nonuniform random variate generation," *Handbooks in operations research and management science*, vol. 13, pp. 83–121, 2006.

[109] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[110] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, and D. Kalenichenko, "Quantization and training of neural networks for efficient integer-arithmetic-only inference," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2704–2713, 2018.

[111] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015. Software available from tensorflow.org.

[112] NEMA, "Nema standards publication nu 2-2012: performance measurements of positron emission tomographs," *National Electrical Manufacturers Association*, 2012.

[113] M. Spiegel, T. Redel, Y. J. Zhang, T. Struffert, J. Hornegger, R. G. Grossman, A. Doerfler, and C. Karmonik, "Tetrahedral vs. polyhedral mesh size evaluation on flow velocity and wall shear stress for cerebral hemodynamic simulation," *Computer methods in biomechanics and biomedical engineering*, vol. 14, no. 01, pp. 9–22, 2011.

[114] C. Kleinstreuer and Z. Zhang, "Airflow and particle transport in the human respiratory system," *Annual review of fluid mechanics*, vol. 42, pp. 301–334, 2010.

[115] F. Alam, S. U. Rahman, A. Khalil, S. Khusro, and M. Sajjad, "Deformable registration methods for medical images: a review based on performance comparison," *Proc Pak Acad Sci A: Phys Comput Sci*, vol. 53, pp. 111–30, 2016.

[116] M. Wang and P. Li, "A review of deformation models in medical image registration," *Journal of Medical and Biological Engineering*, vol. 39, no. 1, pp. 1–17, 2019.

[117] D. S. Lo, *Finite element mesh generation*. CRC Press, 2014.

[118] A. Bowyer, "Computing dirichlet tessellations," *The computer journal*, vol. 24, no. 2, pp. 162–166, 1981.

[119] D. F. Watson, "Computing the n-dimensional delaunay tessellation with application to voronoi polytopes," *The computer journal*, vol. 24, no. 2, pp. 167–172, 1981.

[120] S. Lo, "A new mesh generation scheme for arbitrary planar domains," *International Journal for Numerical Methods in Engineering*, vol. 21, no. 8, pp. 1403–1426, 1985.

[121] H. Borouchaki, P. Laug, and P.-L. George, "Parametric surface meshing using a combined advancing-front generalized delaunay approach," *International Journal for Numerical Methods in Engineering*, vol. 49, no. 1-2, pp. 233–259, 2000.

[122] H. Borouchaki, P. L. George, F. Hecht, P. Laug, and E. Saltel, "Delaunay mesh generation governed by metric specifications. part i. algorithms," *Finite elements in analysis and design*, vol. 25, no. 1-2, pp. 61–83, 1997.

[123] V. Rajan, "Optimality of the delaunay triangulation in rd," *Discrete & Computational Geometry*, vol. 12, no. 2, pp. 189–202, 1994.

[124] H. Borouchaki and P. L. George, "Aspects of 2-d delaunay mesh generation," *International Journal for Numerical Methods in Engineering*, vol. 40, no. 11, pp. 1957–1975, 1997.

[125] H. Borouchaki, P. Laug, A. Cherouat, and K. Saanouni, "Adaptive remeshing in large plastic strain with damage," *International Journal for Numerical Methods in Engineering*, vol. 63, no. 1, pp. 1–36, 2005.

[126] D. A. Field, "Laplacian smoothing and delaunay triangulations," *Communications in applied numerical methods*, vol. 4, no. 6, pp. 709–712, 1988.

[127] C. Cherfils and F. Hermeline, "Diagonal swap procedures and characterizations of 2d-delaunay triangulations," *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique*, vol. 24, no. 5, pp. 613–625, 1990.

[128] A. Sitek, R. H. Huesman, and G. T. Gullberg, "Tomographic reconstruction using an adaptive tetrahedral mesh defined by a point cloud," *IEEE Transactions on medical imaging*, vol. 25, no. 9, pp. 1172–1179, 2006.

[129] H. Zhu, H. Shu, J. Zhou, X. Dai, and L. Luo, "Conditional entropy maximization for pet image reconstruction using adaptive mesh model," *Computerized Medical Imaging and Graphics*, vol. 31, no. 3, pp. 166–177, 2007.

[130] N. Pereira and A. Sitek, "Evaluation of a 3d point cloud tetrahedral tomographic reconstruction method," *Physics in Medicine & Biology*, vol. 55, no. 18, p. 5341, 2010.

[131] R. Boutchko, A. Sitek, and G. T. Gullberg, "Practical implementation of tetrahedral mesh reconstruction in emission tomography," *Physics in Medicine & Biology*, vol. 58, no. 9, p. 3001, 2013.

[132] J. G. Brankov, Y. Yang, and M. N. Wernick, "Tomographic image reconstruction based on a content-adaptive mesh model," *IEEE Transactions on medical imaging*, vol. 23, no. 2, pp. 202–212, 2004.

[133] J. G. Brankov, Y. Yang, R. M. Leahy, and M. N. Wernick, "Multi-modality tomographic image reconstruction using mesh modeling," in *Biomedical Imaging, 2002. Proceedings. 2002 IEEE International Symposium on*, pp. 405–408, IEEE, 2002.

[134] Y. Yang, M. N. Wernick, and J. G. Brankov, "A fast algorithm for accurate content-adaptive mesh generation," in *Proceedings 2001 International Conference on Image Processing (Cat. No. 01CH37205)*, vol. 3, pp. 868–871, IEEE, 2001.

[135] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–471, 1978.

[136] R. W. Floyd, "An adaptive algorithm for spatial gray-scale," in *Proc. Soc. Inf. Disp.*, vol. 17, pp. 75–77, 1976.

[137] H. Borouchaki, T. Grosges, and D. Barchiesi, "Improved 3d adaptive remeshing scheme applied in high electromagnetic field gradient computation," *Finite Elements in Analysis and Design*, vol. 46, no. 1-2, pp. 84–95, 2010.

[138] T. Grosges, H. Borouchaki, and D. Barchiesi, "Three dimensional adaptive meshing scheme applied to the control of the spatial representation of complex field pattern in electromagnetics," *Applied Physics B*, vol. 101, no. 4, pp. 883–889, 2010.

[139] F. Mezghani, D. Barchiesi, A. Cherouat, T. Grosges, and H. Borouchaki, "Comparison of 3d adaptive remeshing strategies for finite element simulations of electromagnetic heating of gold nanoparticles," *Advances in Mathematical Physics*, vol. 2015, 2015.

[140] P. L. George, H. Borouchaki, F. Alauzet, P. Laug, A. Loseille, and L. Marechal, *Meshing, Geometric Modeling and Numerical Simulation, Volume 2: Metrics, Meshes and Mesh Adaptation*. John Wiley & Sons, 2019.

[141] L. Cao, R. Bugalho, N. Matela, M. Martins, P. Almeida, J. Peter, and J. Varela, "List-mode maximum-likelihood reconstruction for the clearpem system," in *Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), 2011 IEEE*, pp. 4171–4174, IEEE, 2011.

[142] T. Grosges, H. Borouchaki, and D. Barchiesi, "Improved scheme for accurate computation of high electric near-field gradients," *Optics express*, vol. 15, no. 3, pp. 1307–1321, 2007.

[143] T. Grosges, H. Borouchaki, and D. Barchiesi, "New adaptive mesh development for accurate near-field enhancement computation," *Journal of microscopy*, vol. 229, no. 2, pp. 293–301, 2008.

[144] D.-T. Lee and B. J. Schachter, "Two algorithms for constructing a delaunay triangulation," *International Journal of Computer & Information Sciences*, vol. 9, no. 3, pp. 219–242, 1980.

[145] P. Laug and H. Borouchaki, "Bl2d-v2: mailleur bidimensionnel adaptatif." `http://www-rocq1.inria.fr/gamma/cdrom/www/bl2d-v2/INDEX.html`, 2003.

[146] A. J. Reader, K. Erlandsson, M. A. Flower, and R. J. Ott, "Fast accurate iterative reconstruction for low-statistics positron volume imaging," *Physics in Medicine & Biology*, vol. 43, no. 4, p. 835, 1998.

[147] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[148] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[149] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. Van Der Laak, B. Van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Medical image analysis*, vol. 42, pp. 60–88, 2017.

[150] T. Wang, Y. Lei, Y. Fu, W. J. Curran, T. Liu, and X. Yang, "Medical imaging synthesis using deep learning and its clinical applications: A review," *arXiv preprint arXiv:2004.10322*, 2020.

[151] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, pp. 2672–2680, 2014.

[152] T. Karras, T. Aila, S. Laine, and J. Lehtinen, "Progressive growing of gans for improved quality, stability, and variation," *arXiv preprint arXiv:1710.10196*, 2017.

[153] K. Armanious, C. Jiang, M. Fischer, T. Küstner, T. Hepp, K. Nikolaou, S. Gatidis, and B. Yang, "Medgan: Medical image translation using gans," *Computerized Medical Imaging and Graphics*, p. 101684, 2019.

[154] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[155] M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken, "Multilayer feedforward networks with a nonpolynomial activation function can approximate any function," *Neural networks*, vol. 6, no. 6, pp. 861–867, 1993.

[156] K. Hornik, M. Stinchcombe, H. White, *et al.*, "Multilayer feedforward networks are universal approximators.," *Neural networks*, vol. 2, no. 5, pp. 359–366, 1989.

[157] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Mathematics of control, signals and systems*, vol. 2, no. 4, pp. 303–314, 1989.

[158] G. F. Montufar, R. Pascanu, K. Cho, and Y. Bengio, "On the number of linear regions of deep neural networks," in *Advances in neural information processing systems*, pp. 2924–2932, 2014.

[159] D. R. Wilson and T. R. Martinez, "The general inefficiency of batch training for gradient descent learning," *Neural networks*, vol. 16, no. 10, pp. 1429–1451, 2003.

[160] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[161] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *The Journal of physiology*, vol. 195, no. 1, pp. 215–243, 1968.

[162] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological cybernetics*, vol. 36, no. 4, pp. 193–202, 1980.

[163] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, Springer, 2015.

[164] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "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, 2016.

[165] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *3D Vision (3DV), 2016 Fourth International Conference on*, pp. 565–571, IEEE, 2016.

[166] G. Gkioxari, J. Malik, and J. Johnson, "Mesh r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9785–9795, 2019.

[167] X. Qi, R. Liao, J. Jia, S. Fidler, and R. Urtasun, "3d graph neural networks for rgbd semantic segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5199–5208, 2017.

[168] C. R. Qi, O. Litany, K. He, and L. J. Guibas, "Deep hough voting for 3d object detection in point clouds," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9277–9286, 2019.

[169] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph cnn for learning on point clouds," *arXiv preprint arXiv:1801.07829*, 2018.

[170] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, vol. 1, no. 2, p. 4, 2017.

[171] D. Maturana and S. Scherer, "Voxnet: A 3d convolutional neural network for real-time object recognition," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 922–928, IEEE, 2015.

[172] N. Ravi, J. Reizenstein, D. Novotny, T. Gordon, W.-Y. Lo, J. Johnson, and G. Gkioxari, "Pytorch3d." `https://github.com/facebookresearch/pytorch3d`, 2020.

[173] L. Gondara, "Medical image denoising using convolutional denoising autoencoders," in *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pp. 241–246, IEEE, 2016.

[174] O. Oktay, E. Ferrante, K. Kamnitsas, M. Heinrich, W. Bai, J. Caballero, S. A. Cook, A. De Marvao, T. Dawes, D. P. O'Regan, *et al.*, "Anatomically constrained neural networks (acnns): application to cardiac image enhancement and segmentation," *IEEE transactions on medical imaging*, vol. 37, no. 2, pp. 384–395, 2018.

[175] A. J. Larrazabal, C. Martinez, and E. Ferrante, "Anatomical priors for image segmentation via post-processing with denoising autoencoders," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 585–593, Springer, 2019.

[176] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[177] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proceedings of the 34th International Conference on Machine Learning* (D. Precup and Y. W. Teh, eds.), vol. 70 of *Proceedings of Machine Learning Research*, (International Convention Centre, Sydney, Australia), pp. 214–223, PMLR, 06–11 Aug 2017.

[178] C. Villani, "Grundlehren der mathematischen wissenschaften [fundamental principles of mathematical sciences]," 2009.

[179] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, "Improved training of wasserstein gans," in *Advances in neural information processing systems*, pp. 5767–5777, 2017.

[180] E. Iarussi, F. Thomsen, and C. Delrieux, "Generative modelling of 3d in-silico spongiosa with controllable micro-structural parameters," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 785–794, Springer, 2020.

[181] A. Roberts, J. Engel, and D. Eck, "Hierarchical variational autoencoders for music," in *NIPS Workshop on Machine Learning for Creativity and Design*, 2017.

[182] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8110–8119, 2020.

[183] S. Vitale, J. I. Orlando, E. Iarussi, and I. Larrabide, "Improving realism in patient-specific abdominal ultrasound simulation using cyclegans," *International Journal of Computer Assisted Radiology and Surgery*, vol. 15, no. 2, pp. 183–192, 2020.

[184] J. M. Wolterink, A. M. Dinkla, M. H. Savenije, P. R. Seevinck, C. A. van den Berg, and I. Išgum, "Deep mr to ct synthesis using unpaired data," in *International Workshop on Simulation and Synthesis in Medical Imaging*, pp. 14–23, Springer, 2017.

[185] T. Wang, Y. Lei, Y. Fu, W. J. Curran, T. Liu, J. A. Nye, and X. Yang, "Machine learning in quantitative pet: A review of attenuation correction and low-count image reconstruction methods," *Physica Medica*, vol. 76, pp. 294–306, 2020.

[186] L. Shi, J. A. Onofrey, E. M. Revilla, T. Toyonaga, D. Menard, J. Ankrah, R. E. Carson, C. Liu, and Y. Lu, "A novel loss function incorporating imaging acquisition physics for pet attenuation map generation using deep learning," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 723–731, Springer, 2019.

[187] J. Hamill and V. Panin, "Tof-mlaa for attenuation correction in thoracic pet/ct," in *2012 IEEE Nuclear Science Symposium and Medical Imaging Conference Record (NSS/MIC)*, pp. 4040–4047, IEEE, 2012.

[188] K. Clark, B. Vendt, K. Smith, J. Freymann, J. Kirby, P. Koppel, S. Moore, S. Phillips, D. Maffitt, M. Pringle, *et al.*, "The cancer imaging archive (tcia): maintaining and operating a public information repository," *Journal of digital imaging*, vol. 26, no. 6, pp. 1045–1057, 2013.

[189] B. Tatiana, M. De Ornelas Couto, and I. B. Mihaylov, "Head-and-neck squamous cell carcinoma patients with ct taken during pre-treatment, mid-treatment, and post-treatment dataset.," 2018.

[190] F. Chollet *et al.*, "Keras." `https://keras.io`, 2015.

[191] S. Bakr, O. Gevaert, S. Echegaray, K. Ayers, M. Zhou, M. Shafiq, H. Zheng, W. Zhang, A. Leung, M. Kadoch, *et al.*, "Data for nsclc radiogenomics collection," *The Cancer Imaging Archive*, 2017.

[192] M. L. Zuley, R. Jarosz, S. Kirk, Y. Lee, R. Colen, K. Garcia, and N. Aredes, "Radiology data from the cancer genome atlas head-neck squamous cell carcinoma [tcga-hnsc] collection," *Cancer Imaging Archive. doi*, 2016.

[193] B. Albertina, M. Watson, C. Holback, R. Jarosz, S. Kirk, Y. Lee, and J. Lemmerman, "Radiology data from the cancer genome atlas lung adenocarcinoma [tcga-luad] collection," *Cancer Imaging Arch*, 2016.

[194] S. Kirk, Y. Lee, C. Roche, *et al.*, "Radiology data from the cancer genome atlas thyroid cancer [tcga-thca] collection," *Cancer Imaging Archive. doi*, 2016.

[195] A. Meldo, L. Utkin, A. Lukashin, V. Muliukha, and V. Zaborovsky, "Database acquisition for the lung cancer computer aided diagnostic systems," in *2019 25th Conference of Open Innovations Association (FRUCT)*, pp. 220–227, IEEE, 2019.

[196] A. B. Menegotto, C. D. L. Becker, and S. C. Cazella, "Computer-aided hepatocarcinoma diagnosis using multimodal deep learning," in *International Symposium on Ambient Intelligence*, pp. 3–10, Springer, 2019.

[197] N. C. I. C. P. T. A. C. C. National Cancer Institute Clinical Proteomic Tumor Analysis Consortium (CPTAC), "Radiology data from the clinical proteomic tumor analysis consortium uterine corpus endometrial carcinoma (cptac-ucec) collection," 2019.

[198] N. C. I. C. P. T. A. C. C. National Cancer Institute Clinical Proteomic Tumor Analysis Consortium (CPTAC), "Radiology data from the clinical proteomic tumor analysis consortium lung squamous cell carcinoma [cptac-lscc] collection," 2018.

[199] P. Bandi, N. Zsoter, L. Seres, Z. Toth, and L. Papp, "Automated patient couch removal algorithm on ct images," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 7783–7786, IEEE, 2011.

[200] K. Wu, E. Otoo, and A. Shoshani, "Optimizing connected component labeling algorithms," in *Medical Imaging 2005: Image Processing*, vol. 5747, pp. 1965–1976, International Society for Optics and Photonics, 2005.

[201] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu, and the scikit-image contributors, "scikit-image: image processing in Python," *PeerJ*, vol. 2, p. e453, 6 2014.

[202] K. R. Zasadny and R. L. Wahl, "Standardized uptake values of normal tissues at pet with 2-[fluorine-18]-fluoro-2-deoxy-d-glucose: variations with body weight and a method for correction.," *Radiology*, vol. 189, no. 3, pp. 847–850, 1993.

[203] A. K. Tahari, D. Chien, J. R. Azadi, and R. L. Wahl, "Optimum lean body formulation for correction of standardized uptake value in pet imaging," *Journal of Nuclear Medicine*, vol. 55, no. 9, pp. 1481–1484, 2014.

[204] J. W. Keyes Jr *et al.*, "Suv: standard uptake or silly useless value?," *Journal of Nuclear Medicine*, vol. 36, no. 10, pp. 1836–1839, 1995.

[205] S.-C. Huang, "Anatomy of suv," *Nuclear medicine and biology*, vol. 27, no. 7, pp. 643–646, 2000.

[206] Y. Wang, B. Yu, L. Wang, C. Zu, D. S. Lalush, W. Lin, X. Wu, J. Zhou, D. Shen, and L. Zhou, "3d conditional generative adversarial networks for high-quality pet image estimation at low dose," *Neuroimage*, vol. 174, pp. 550–562, 2018.

[207] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

[208] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

[209] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[210] L. Mansilla, D. H. Milone, and E. Ferrante, "Learning deformable registration of medical images with anatomical constraints," *Neural Networks*, vol. 124, pp. 269–279, 2020.

[211] J.-Y. Zhu, Z. Zhang, C. Zhang, J. Wu, A. Torralba, J. Tenenbaum, and B. Freeman, "Visual object networks: image generation with disentangled 3d representations," in *Advances in Neural Information Processing Systems*, pp. 118–129, 2018.

[212] T. Nguyen-Phuoc, C. Li, L. Theis, C. Richardt, and Y.-L. Yang, "Hologan: Unsupervised learning of 3d representations from natural images," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 7588–7597, 2019.

[213] X. Dong, T. Wang, Y. Lei, K. Higgins, T. Liu, W. J. Curran, H. Mao, J. A. Nye, and X. Yang, "Synthetic ct generation from non-attenuation corrected pet images for whole-body pet imaging," *Physics in Medicine & Biology*, vol. 64, no. 21, p. 215016, 2019.

[214] J. P. Cohen, M. Luck, and S. Honari, "Distribution matching losses can hallucinate features in medical image translation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 529–536, Springer, 2018.

[215] Y. LeCun and C. Cortes, "MNIST handwritten digit database," 2010.

[216] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4401–4410, 2019.

[217] R. R. Colmeiro, C. Verrastro, D. Minsky, and T. Grosges, "Whole body positron emission tomography attenuation correction map synthesizing using 3d deep generative adversarial networks," *Prerpint*, 2020.

[218] H. O. Anger, "Scintillation camera," *Review of scientific instruments*, vol. 29, no. 1, pp. 27–33, 1958.

[219] G. Borghi, V. Tabacchini, S. Seifert, and D. R. Schaart, "Experimental validation of an efficient fan-beam calibration procedure for k-nearest neighbor position estimation in monolithic scintillator detectors," *IEEE Transactions on Nuclear Science*, vol. 62, no. 1, pp. 57–67, 2015.

Abbreviations and Nomenclature

List of Abbreviations

ADAM	**Ada**ptive **M**oment estimation
AE	**A**uto**e**ncoder
AM	**A**ttenuation **M**ap
ANMAT	**A**dministración **N**acional de **M**edicamentos, **A**limentos y **T**ecnología médica (Administration of Drugs, Foods and Medical Devices)
BGO	**B**ismuth **G**ermanium **O**xide
BP	**B**ack-**P**rojection
CDF	**C**umulative **D**ensity **F**unction
CDM	**C**hamfer **D**istance **M**etric
CNEA	**C**omisión **N**acional de **E**nergía **A**tómica (National Atomic Energy Commission)
CNN	**C**onvolutional **N**eural **N**etwork
COG	**C**enter **O**f **G**ravity
CPTAC	**C**linical **P**roteomic **T**umor **A**nalysis **C**onsortium
CT	**C**omputed **T**omography
DAC	**D**igital to **A**nalog **C**onverter
DAF	**D**elaunay–**A**dvancing-**F**ront
DCF	**D**ecay **C**orrection **F**actor
DCN	**D**eep **C**onvolutional **N**etworks
DICOM	**D**igital **I**maging and **C**ommunications in **M**edicine
DK	**D**elaunay insertion **K**ernel
DNN	**D**eep **N**eural **N**etworks
DoI	**D**epth **o**f **I**nteraction
ELBO	**E**vidence **L**ower **Bo**und
EM	**E**xpectation **M**aximization
EMD	**E**arth **M**over's **D**istance
EWMA	**E**xponentially **W**eighted **M**oving **A**verage
FBP	**F**iltered **B**ack-**P**rojection
FCN	**F**ully **C**onnected **N**etwork
FDG	^{18}**F**-fluoro-2-deoxy-**D**-**g**lucose
FE	**F**inite **E**lement
FEM	**F**inite **E**lement **M**ethod
FoV	**F**ield **o**f **V**iew
FPGA	**F**ield-**P**rogrammable **G**ate **A**rray
FWHM	**F**ull **W**idth **H**alf **M**aximum
FWTM	**F**ull **W**idth **T**enth **M**aximum

GAN ..... **G**enerative **A**dversarial **N**etworks
GATE ..... **G**eant4 **A**pplication for **T**omographic **E**mission
GD ..... **G**radient **D**escent
GIAR ..... **G**rupo de **I**nteligencia **A**rtificial y **R**obótica (Artificial Intelligence and Robotics Group)
GPU ..... **G**raphic **P**rocessing **U**nit
HNSCC ..... **H**ead **N**eck **S**camorous **C**ell **C**arcinoma
HU ..... **H**ounsfield **U**nit
HV ..... **H**igh **V**oltage
ICRP ..... **I**nternational **C**ommission on **R**adiation **Protection**
KL ..... **K**ullback-**L**eibler
kNN ..... **k**-**N**earest **N**eighbours
LED ..... **L**ight **E**mitting **D**iodes
LoR ..... **L**ine **o**f **R**esponse
LSCC ..... **L**ung **S**quamous **C**ell **C**arcinoma
LUAD ..... **Lu**ng **Ad**enocarcinoma
LuT ..... **L**ook-**u**p **T**able
LVDS ..... **L**ow-**v**oltage **D**ifferential **S**ignalling
LYSO ..... **L**utetium-**y**ttrium oxyorthosilicate,
MC ..... **M**onte **C**arlo
ML ..... **M**achine **L**earning
ML EM ..... **M**aximum **L**ikelihood **E**xpectation **M**aximization
MRI ..... **M**agnetic **R**esonance **I**maging
MSE ..... **M**ean **S**quared **E**rror
NAC ..... **N**on **A**ttenuation **C**orected (PET)
NEMA ..... **N**ational **E**lectrical **M**anufacturers **A**ssociation
NN ..... **N**eural **N**etwork
NSCLC ..... **N**on-**S**mall **C**ell **L**ung **C**ancer
OS EM ..... **O**rdered **S**ubset **E**xpectation **M**aximization
PDA ..... **P**ancreatic **D**uctal **A**denocarcinoma
PDF ..... **P**robability **D**ensity **F**unction
PET ..... **P**ositron **E**mission **T**omography
PMT ..... **P**hoto**m**ultiplier **T**ube
pt ..... **Pa**t**i**ent
px ..... **Pix**el
ReLU ..... **Re**ctified **L**inear **U**nit
RMSProp ..... **R**oot **M**ean **S**quare **Prop**agation
rnd ..... **R**a**nd**om
RoI ..... **R**egion **o**f **I**nterest
sCT ..... **s**ynthetic **C**omputed **T**omography
SGD ..... **S**tochastic **G**radient **D**escent
SiPM ..... **Si**licon **P**hoto**m**ultiplier
SNR ..... **S**ignal to **N**oise **R**atio
SPECT ..... **S**ingle **P**hoton **E**mission **C**omputed **T**omography
SSS ..... **S**ingle **S**catter **S**imulation

SUV ..... **S**tandard **U**ptake **V**alue
TCGA ..... **T**he **C**ancer **G**enome **A**tlas
TCIA ..... **T**he **C**ancer **I**mage **A**rchive
THCA ..... **Th**yroid **Ca**ncer
thkns ..... **th**ic**k**en**s**s
ToF ..... **T**ime **o**f **F**light
UCEC ..... **U**terine **C**orpus **E**ndometrial **C**arcinoma
UTN ..... **U**niversidad **T**ecnológica **N**acional (National Technological University)
UTT ..... **U**niversité de **T**echnologie de **T**royes (University of Technology of Troyes)
VAE ..... **V**ariational **A**uto**e**ncoder
vx ..... **V**o**x**el
WGAN ..... **W**asserstein **G**enerative **A**dversarial **N**etworks

Nomenclature - Standards

α ..... [−] Fine structure constant of the Compton scattering.
β^+ ..... [−] Positron.
$\mathcal{O}$ ..... [−] Asymptotic computational cost.
μ ..... [$\mathtt{mm}^{-1}$] Linear Attenuation Coefficient.
ϕ ..... [`rad`.] Scattering angle of a gamma photon.
σ ..... [$\mathtt{m}^2$] Geometrical cross-section.
σ_c ..... [$\mathtt{m}^2$] Interaction cross-section of gamma photon.
σ_o ..... [$\mathtt{m}^2$] Classical scattering cross-section.
c ..... [$\frac{\mathtt{m}}{\mathtt{s}}$] Speed of light in the vacuum (299792458 $\frac{\mathtt{m}}{\mathtt{s}}$)
m_e ..... [`Kg`] The electron mass (9.109×10^{-31} `Kg`).
r_o ..... [`m`] The electron radius (2.8179×10^{-15} `m`).
t_d ..... [−] Disintegration constant of the radionuclide.

Nomenclature - PET

ΔE_γ ..... [`keV`] Energy size of the gamma photon detection window.
$\varnothing_{\text{pt.}}$ ..... [`mm`] External diameter of the patient inside the PET scanner.
$\varnothing_{FoV}$ ..... [`mm`] Diameter of the FoV.
$\text{SR}_{\texttt{ToF}}$ ..... [`mm`] Distance of the LoR that can be measured using ToF capabilities of the PET scanner.
τ ..... [`nSeg`.] Time of the coincidence window.
A ..... [`Sv`] Total injected activity.
E_γ ..... [`keV`] Energy of a detected gamma photon.
E_e ..... [`keV`] Energy of an electron.
G ..... [−] Geometric coverage fraction of PET the scanner.
$k_{\text{pt.}}$ ..... [−] Fraction of true coincidence of patient.
L ..... [−] Total number of LoRs or coincidences in a PET scan.
L^* ..... [−] The set of all possible LoRs (in sinogram mode $L^* = L$).
S ..... [−] Number of steps in a given process (reconstruction, remeshing, training, etc.).

T ..... [Seg.] Total acquisition time of PET scan.

Nomenclature - Gamma Detectors

$\chi_{\texttt{cal.}}$ ..... [−] Correction factor of PMT sensibility in the calibration procedure to the scintillation camera.
η ..... [−] Light dispersion feature of an event. Used in DoI estimation.
ι ..... [−] Single detection rate in a detector or a detector's pixel.
$\nu_{c,c}$ ..... [−] Relation in photon count of the PMTs in the AR-PET scintillation camera, for the *center to corner* positions.
$\nu_{c,e}$ ..... [−] Relation in photon count of the PMTs in the AR-PET scintillation camera, for the *center to edge* positions.
$\texttt{V}_{\texttt{dn}}$ ..... [v] Voltage of a dynode in a PMT.
υ ..... [−] Probability of interaction of a 511 keV gamma photon in the detector (photo-peak efficiency).
ζ ..... [−] Number of visible photons detected by a sensible element.
D ..... [−] Detector units (or pixels) of the PET scanner.
D_g ..... [−] Detector unit (or pixel) gain.
D_{px} ..... [mm^2] Area of a detector's pixel.
D_{thkns} ..... [mm^2] Thickness of a detector's scintillation crystal.
D_{x} ..... [mm^2] Size of a detector's scintillation crystal in the X axis.
D_{y} ..... [mm^2] Size of a detector's scintillation crystal in the Y axis.
$f_{\texttt{trust}}(\boldsymbol{r})$ ..... [−] Trust map function used to modulate the weight of the linear positioning methods on the detector, according to their spatial coordinates.
P ..... [−] Number of PMTs in a scintillation camera.

Nomenclature - Reconstruction

ρ ..... [mm] Distance of LoR to the center of the FoV (used in sinogram notation).
Θ ..... [−] Number of angular positions in a sinogram.
θ ..... [rad.] LoR angular position in sinogram notation.
H ..... [−] System matrix of the PET scanner, contains the relationship coefficients of each voxel or element in the distribution Λ to each LoR in the set X.
H^{μ} ..... [−] System matrix of the PET scanner for attenuation coefficients, contains the relationship coefficients of each voxel or element in the AM to each LoR attenuation factor in the set X^{μ}.
$h^{\mu}_{l,u}$ ..... [−] Element of matrix H^{μ} corresponding to the influence of μ_u on the attenuation coefficient x^{μ}_{l}.
$h_{l,i}$ ..... [−] Element of matrix H corresponding to the influence of λ_i on the counts detected in x_l.
Q ..... [−] Total number of rings or divisions of the the axial FoV.
q ..... [−] Ring or axial plane number.
$Rad(\rho,\theta)$ ..... [−] Radon transformation value for the LoR defined by ρ and

θ. This value is equal to x_l when the l^{th} LoR is defined by the same ρ and θ parameters.

X ..... [−] Set of L detected LoRs x_l.

$X^{\texttt{gain}}$ ..... [−] Set of L gain correction factors corresponding to the LoR set X.

X^{μ} ..... [−] Set of L attenuation factors corresponding to the LoR set X.

x_l ..... [−] Amount of detected counts of the l^{th} LoR.

$x_l^{\texttt{gain}}$ ..... [−] Gain correction factor of the l^{th} LoR.

x_l^{μ} ..... [−] Attenuation factor of the l^{th} LoR.

$x_l^{\texttt{rnd.}}$ ..... [−] Rate of random counts of the l^{th} LoR.

$x_l^{\texttt{scatter}}$ ..... [−] Scatter fraction of the l^{th} LoR.

Nomenclature - Volume description

$\boldsymbol{V}$ ..... [−] Set of voxels in a volumetric image.

$\boldsymbol{V}^{\texttt{CT}}$ ..... [HU] Computed Tomography volumetric image.

$\boldsymbol{V}^{\texttt{NPT}}$ ..... [suv] Non Attenuation corrected PET volumetric image.

$\boldsymbol{V}^{\texttt{PT}}$ ..... [suv] PET volumetric image.

$\boldsymbol{V}^{\texttt{sCT}}$ ..... [HU] Synthetic Computed Tomography volumetric image.

$\boldsymbol{V}^{\texttt{TL}}$ ..... [−] Tissue label volumetric map.

$\Lambda(x, y, z)$ ..... [−] Volumetric distribution of the radiotracer.

λ_i ..... [−] Activity of the i^{th} element of the discrete distribution Λ.

$\texttt{v}_n$ ..... [−] Value of the n^{th} voxel of a volumetric image.

$\texttt{v}_x$ ..... [−] Voxel or pixel size in the x-axis direction.

$\texttt{v}_y$ ..... [−] Voxel or pixel size in the y-axis direction.

$\texttt{v}_z$ ..... [−] Voxel or pixel size in the z-axis direction.

μ_u ..... [$\texttt{mm}^{-1}$] Attenuation of the u^{th} element of AM.

AM ..... [$\texttt{mm}^{-1}$] Attenuation Map of a PET scan, composed of U elements μ_u.

d_x ..... [−] Length of an image or volume in the x-axis direction.

d_y ..... [−] Length of an image or volume in the y-axis direction.

d_z ..... [−] Length of an image or volume in the z-axis direction.

I ..... [−] Number of elements in the discrete activity distribution Λ.

p_c ..... [−] Number of channels of an image or volume.

p_x ..... [−] Number of elements of an image or volume in the x-axis direction.

p_y ..... [−] Number of elements of an image or volume in the y-axis direction.

p_z ..... [−] Number of elements of an image or volume in the z-axis direction.

U ..... [−]Total number of elements or voxels representing the Attenuation Map.

V ..... [−] Number of voxels in a volumetric image.

Nomenclature - Genetic Algorithms

Ξ ..... [−] Number of population individuals used in optimization via genetic algorithms (used in the configuration of the scintillation cameras).
ξ ..... [−] Fitness value of an individual of a genetic algorithm.

Nomenclature - Event positioning

$\bar{e}$ ..... [−] Set of all energy values detected in each PMT within a detector.
c_p^x ..... $[\frac{\texttt{mm}}{\texttt{keV}}]$ Element of the matrix C_x for the p^{th} PMT.
c_p^y ..... $[\frac{\texttt{mm}}{\texttt{keV}}]$ Element of the matrix C_y for the p^{th} PMT.
C_x ..... [−] Matrix of Anger coefficients for the X axis.
C_y ..... [−] Matrix of Anger coefficients for the Y axis.
e_p ..... [keV] Energy value of the p^{th} PMT of a detector.

Nomenclature - Neural Networks

$\bar{W}$ ..... [−] Weight vector of a NN, composed of $\boldsymbol{W}$ elements w_i.
β ..... [−] Moment decay of learning rate of NN.
$\boldsymbol{b}^{[i]}$ ..... [−] Number bias of the i^{th} layer.
$\boldsymbol{k}_x$ ..... [−] Number of elements of a convolutional kernel in the x-axis direction.
$\boldsymbol{k}_y$ ..... [−] Number of elements of a convolutional kernel in the y-axis direction.
$\boldsymbol{k}_z$ ..... [−] Number of elements of a convolutional kernel in the z-axis direction.
$\boldsymbol{n}_x$ ..... [−] Number of inputs of the layer.
$\boldsymbol{n}_f^{[i]}$ ..... [−] Number of filters of the i^{th} a convolutional layer.
$\boldsymbol{n}_h^{[i]}$ ..... [−] Number of hidden neurons of the i^{th} a fully connected layer.
$\boldsymbol{W}^{[i]}$ ..... [−] Number weights of the i^{th} layer.
ϵ ..... [−] Learning rate of NN.
$\mathcal{L}()$ ..... [−] Loss function (on a single sample).
$K()$ ..... [−] Convolutional kernel of a NN layer.
$b^{[i]}$ ..... [−] Bias term of the i^{th} layer of a NN.
$g()$ ..... [−] Activation function of a NN.
$J()$ ..... [−] Cost function (over whole dataset).
N ..... [−] Number of samples in a set.
v ..... [−] Learning momentum of NN.
w_i ..... [−] Weight of a NN.

Nomenclature - Mesh

$\boldsymbol{\Omega}$ ..... [−] Domain of FEM calculation.
$\boldsymbol{R}$ ..... [−] Set of point coordinates $\boldsymbol{r}$ in space.
$\boldsymbol{r}$ ..... [−] Coordinates of a point (or mesh node) in space, $\boldsymbol{r} = (r_x, r_y, r_z)$.
δ ..... [−] Tolerated physical deviation of the FEM.
Γ ..... [−] Segment front of DAF method.
$\gamma(\boldsymbol{r}^a, \boldsymbol{r}^b, \boldsymbol{r}^c)$ ..... [−] Equilarity measure function of an element defined by the

points $(\boldsymbol{r}^a, \boldsymbol{r}^b, \boldsymbol{r}^c)$.

$\mathcal{B}$ ..... [−] Boundary of the meshed domain $\mathcal{F}(\boldsymbol{\Omega})$.

$\mathcal{F}(\boldsymbol{\Omega})$ ..... [−] Mesh representation of a given domain $\boldsymbol{\Omega}$.

$\mathcal{H}(\Omega)$ ..... [−] Element size-map of domain Ω.

$\mathcal{K}'_{l,k}$ ..... [−] Contribution of the sampled point k, from the discrete distribution Λ, to the probability of detecting an event in the direction of the l^{th} LoR.

$\mathcal{L}$ ..... [−] Segment or side of element (simplex containing two points).

$\mathcal{M}(\boldsymbol{r}, S(\boldsymbol{r})$ ..... [−] Maximal deviation from the normal plane at point $\boldsymbol{r}$ to all its neighbouring nodes.

$\mathcal{S}$ ..... [−] Exact solution of PDE.

$\mathcal{S}_{\mathcal{F}}$ ..... [−] Solution of PDE through FE calculation.

$\Phi()$ ..... [−] Mesh element shape function. Used to interpolate values within a mesh element.

$\mathtt{SC}_k$ ..... [−] Sampling correction factor for the k^{th} element of the discrete distribution Λ.

$\tilde{e_{\mathcal{F}}}$ ..... [−] Interpolation error of the FEM.

$\tilde{S_{\mathcal{F}}}$ ..... [−] Interpolated solution of PDE through FE calculation.

E ..... [−] Number of elements (i.e. triangles) in a mesh.

$e_{\mathcal{F}}$ ..... [−] Physical deviation of the FEM.

$h()$ ..... [−] Metric function that defines the Segment size on a given point of the domain $\boldsymbol{\Omega}$.

I ..... [−] Number of nodes $\boldsymbol{r}$ in a mesh.

N_s ..... [−] Total number 2D meshes used to describe a volume.

Spanish Extended Abstract

Contents

Introducción

La Tomografía por Emisión de Positrones (conocida por su sigla en idioma inglés, PET) continúa siendo una de las herramientas más poderosas para la detección y seguimiento del cáncer, medición del metabolismo y estudio de medicamentos. Hoy en día los escáneres PET son ampliamente utilizados en la medicina usando el trazador ^{18}F-fluoro-2-deoxy-D-glucose (FDG), mayormente para pacientes con sospecha o diagnostico de cáncer. También el uso se estudios PET para estudios de cribado en pacientes asintomáticos esta siendo estudiado [1,2]. Sin embargo el uso de esta técnica implica un riesgo considerable para el paciente. Dicho riesgo viene de la radiación absorbida por el mismo, la cual es estimada entre 10 `mSv` a 18 `mSv` para un estudio de cribado [3,4]. Esta dosis es considerada una *dosis sustancial* para la Comisión Internacional de Protección Radiológica (conocida por su sigla en idioma inglés, ICRP) [5] y es uno de los argumentos utilizados en contra del uso del PET como estudio de cribado en pacientes asintomáticos [6]. Pero la dosis de radiación de un estudio PET no proviene enteramente del FDG, entre 1.3 `mSv` y 8.0 `mSv` de la dosis provienen de la imagen de Tomografía Computada (CT) en un estudio de cribado promedio (si se utiliza un CT de baja dosis) [3,4,7]. El estudio CT es erróneamente considerado un estudio de *baja dosis* por los médicos y el mismo debería ser evitado cuando sea posible [8]. Por lo tanto la reducción de la dosis en estudios PET y CT, en particular, son campos activos de investigación en las imágenes PET [9,10]. Se podría reducir aún más la dosis del estudio removiendo completamente el estudio CT del estudio PET, sin perder exactitud ni calidad de imagen. Con este objetivo, la idea de reconstruir el Mapa de Atenuación (AM), sin utilizar el estudio CT esta siendo explorada con modelos cada vez más complejos [11,12], y más recientemente usando modelos de aprendizaje automático basados en datos de la imagen PET Sin Corrección de Atenuación (NAC PET) [13–15] u otro tipo de modalidades [16,17].

En este ámbito, la contribución de esta tesis es la aplicación de técnicas de Aprendizaje Automático (ML) y muestreo disperso para reducir la dosis del paciente en estudios PET debido al estudio CT. Más específicamente, los métodos para obtener un AM de imágenes NAC PET son estudiados, posibilitando el uso de métodos con costo computacional más elevado. Esto es logrado por medio de utilizar una representación de mallas de la imagen y generando imágenes de AM de alta calidad a priori usando métodos de ML. Para esto, primero se construye un esquema de reconstrucción sobre mallas adaptativas, utilizando mallas de alta calidad para aplicar métodos de reconstrucción de mayor complejidad, reduciendo el costo computacional asintótico y permitiendo la generación o mejora de los AM y mejorando la calidad de imagen PET en general. Segundo, las Redes Neuronales Convolucionales Profundas (DCN) y las Redes Neuronales Generativas Adversarias (GAN) son estudiadas, para inferir AM, obteniendo resultados superiores al estado del arte. Mediante la generación de AMs a

priori de alta calidad con técnicas de ML y reduciendo el el costo computacional de modelos de construcción complejos, se espera reducir el costo y mejorar la convergencia de los métodos de reconstrucción de imágenes y corrección de atenuación basados en Simulación de Dispersión Simple (SSS) [18], modelos de resolución [19] y simulaciones Monte Carlo [20].

Esquema del manuscrito

El objetivo de esta tesis allana el camino para un estudio PET de baja dosis, eliminando la necesidad de un estudio CT o estudio de transmisión gamma. Las contribuciones están basadas en simulaciones, bases de datos públicas y mediciones de un prototipo de tomógrafo PET de baja dosis. Este resumen esta dividido en cuatro capítulos principales. En el capitulo A el prototipo AR-PET es rápidamente presentado junto con dos trabajos de optimización de sus detectores gamma de cristal continuo. Luego en el capitulo B el uso de mallas para la reconstrucción de imágenes PET es presentada. Aquí la habilidad de las mallas para adaptarse y describir una imagen PET es demostrada. Las mallas representan la imagen PET sin perder calidad y reduciendo el número de parámetros cuando es comparada con una representación voxelizada. Finalmente el capitulo C toma la temática del uso de DCN y GAN para el problema de la generación de AM desde una imagen NAC PET. Las arquitecturas y el conjunto de datos utilizado son presentados, mostrando las capacidades y los defectos de dichas técnicas. El código para reproducir estos resultados junto con el conjunto de datos son provistos.

En el último capitulo se presentan las conclusiones globales de la tesis y los resultados y proyecciones de sus contribuciones son discutidos.
Para facilitar la lectura de la versión completa de la tesis, escrita en idioma inglés, todos las abreviaciones y símbolos matemáticos se conservan de la misma manera que en el manuscrito original por lo que corresponden a su significado en idioma inglés.

Los resultados presentados en esta tesis están apoyados en las siguientes publicaciones revisadas:

- "Heuristic Method for Photo-detectors Localization over Continuous Crystal Scintillation Cameras" [21].
- "Event Localization in Continuous Crystal Scintillation Cameras using Distribution Matching Neural Networks" (Aceptada en "Third Argentine Conference on Electronics" de la IEEE, 10-12 Marzo de 2021).
- "Reconstruction of Positron Emission Tomography images using adaptive sliced remeshing strategy" [23]
- "Multimodal Brain Tumor Segmentation Using 3D Convolutional Networks" [24]
- "Whole Body [^{18}]F FDG Positron Emission Tomography Attenuation Correction Map Synthesizing using 3D Deep Generative Adversarial Networks" (Enviada para revisión)[1]

[1]Preprint disponible [217]. El código se puede encontrar en: https://github.com/RawthiL/PET_DeepAttCorrection

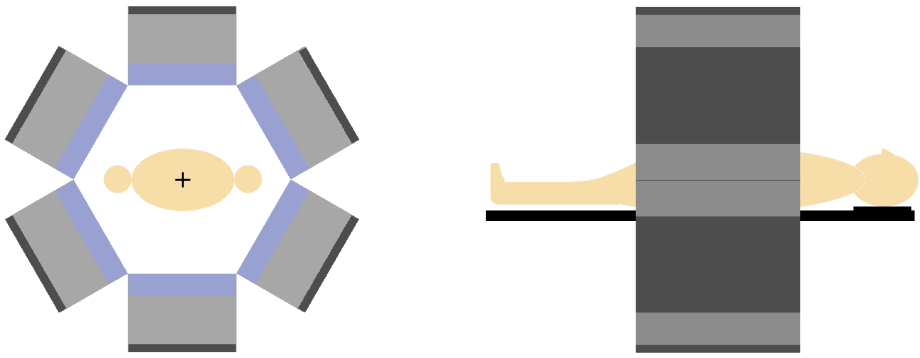
A Prototipo AR-PET

En esta sección se presenta el prototipo de tomógrafo AR-PET, el cual es utilizado a lo largo de toda la tesis. Aquí se describe rápidamente el diseño del tomógrafo y se hace énfasis en las camaradas de detección gamma, para las cuales un criterio de selección de Tubos FotoMultiplicadores (PMT) es desarrollado. Este nuevo esquema de construcción hace uso de algoritmos genéticos para mejorar la homogeneidad de las ganancias de los PMT. Luego, para la optimización de las cámaras gamma, se presenta un nuevo algoritmo de posicionamiento de eventos. Dicho esquema de posicionamiento esta basado en redes neuronales y entrenado con una única adquisición de campo inundado. El posicionamiento resultante presenta mejoras sobre el algoritmo de centro de gravedad, por lo que aumenta el área sensible de la cámara gamma.

A.1 El AR-PET

El AR-PET [81,82] es un prototipo de tomógrafo diseñado para reducir el costo de los estudios PET y posibilitar la extensión del uso de este tipo de técnicas en poblaciones de bajos recursos, como los estados latinoamericanos. En estas comunidades el número de cámaras PET por millón de habitante esta cerca de 0.5 contra 1.5 en Europa y 7 en los Estados Unidos de América [83]. El prototipo esta siendo desarrollado por el departamento de Instrumentación y Control de la Comisión Nacional de Energía Atómica (CNEA) con colaboración del Grupo de Inteligencia Artificial y Robótica (GIAR) de la Universidad Tecnológica Nacional (UTN). Actualmente el primer prototipo esta siendo probado en el centro de medicina nuclear del Hospital de Clínicas José de San Martín en Buenos Aires, Argentina.

El tomógrafo posee seis grandes cámaras de centelleo dispuestas en forma hexagonal, como se muestra en la figura A.1. En la figura A.2 se observa una vista 3D de la construcción interna del tomógrafo. Ya que gran parte del costo de un tomógrafo PET comercial viene de los cristales y la dificultad de fabricar pixeles pequeños, cada cámara de centelleo esta compuesta por un cristal continuo de NaI(TI). Estos cristales continuos presentan una gran cobertura geométrica y un buen poder de frenado a un bajo costo. Esto permite reducir la dosis de radiación en el paciente y también el tiempo total del estudio, ambos factores limitantes para ela amplia aplicación de estudios PET. Una configuración similar a la del AR-PET, usando 4 cristales continuos fue exitosamente aplicada en tomógrafos PET para animales pequeños [84]. Para reducir los efectos de zonas ciegas e inhomogeneidades del arreglo hexagonal, las cámaras gamma del tomógrafo rotan alrededor del eje axial a 2 `rpm`. Debido al modo de adquisición rotacional, la gran superficie de detección y capacidad de detección de Profundidad de Interacción (DoI), el tomógrafo acumula datos en Modo Lista. Final-



(a) (b)

Figure A.1: Vista frontal (a) y lateral (b) de la geometría del tomógrafo AR-PET. En gris se muestra la estructura del tomógrafo, en azul claro la superficie de los detectores y en negro la camilla del paciente.

mente la recontrucción de la distribución del radiotrazador se realiza en un servidor externo utilizando el algoritmo ML-EM [146]. El tomógrafo no posee capacidad de medir el mapa de atenuación, actualmente se recurre al apareado de estudios externos para compensar por los efectos de atenuación durante la reconstrucción.

A.2 Cámaras de centelleo

Cada una de las cámara de centelleo esta compuesta por cristales continuos de NaI(TI) de $406.4 \times 304.8 \times 25.4$ $\mathtt{mm}^3$, poblados por una colección de 8×6 PMTs *Hamamatsu R1534*. Estos PMT tienen un perfil cuadrado de 52.4×52.4 $\mathtt{mm}^2$ que cubren por completo la superficie del cristal sin un acoplamiento mecánicamente complejo. Una representación del cristal centellador y de la cámara de detección puede verse en la figura A.3.

Para hacer frente a las limitaciones de usar cristales monolíticos, el prototipo AR-PET cuenta con una arquitectura de procesamiento distribuido en cada PMT [85]. Los 48 PMT poseen una matriz de puertas lógicas programable en campo (conocida como FPGA por su sigla en idioma inglés) tipo *Spartan3e* las cuales se comunican por una señal diferencial de bajo voltaje a una *Artix 7-100* FPGA que centraliza todas la información de la cámara. Esto permite por un lado que cada cámara del AR-PET funcione de manera independiente como una cámara gamma, no necesitando las 6 cámaras para poder funcionar. Por otro lado provee un alto poder de procesamiento de alta velocidad sobre las señales de energía detectadas.

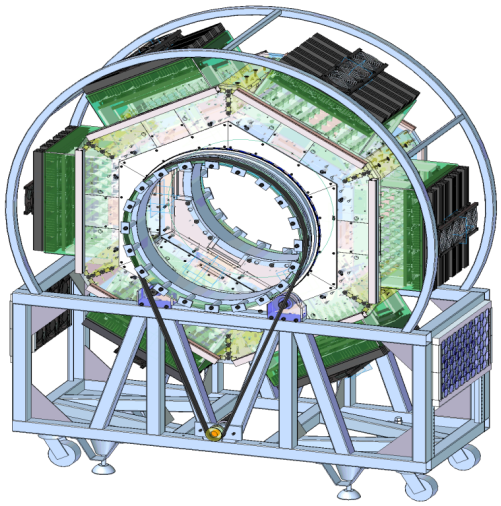
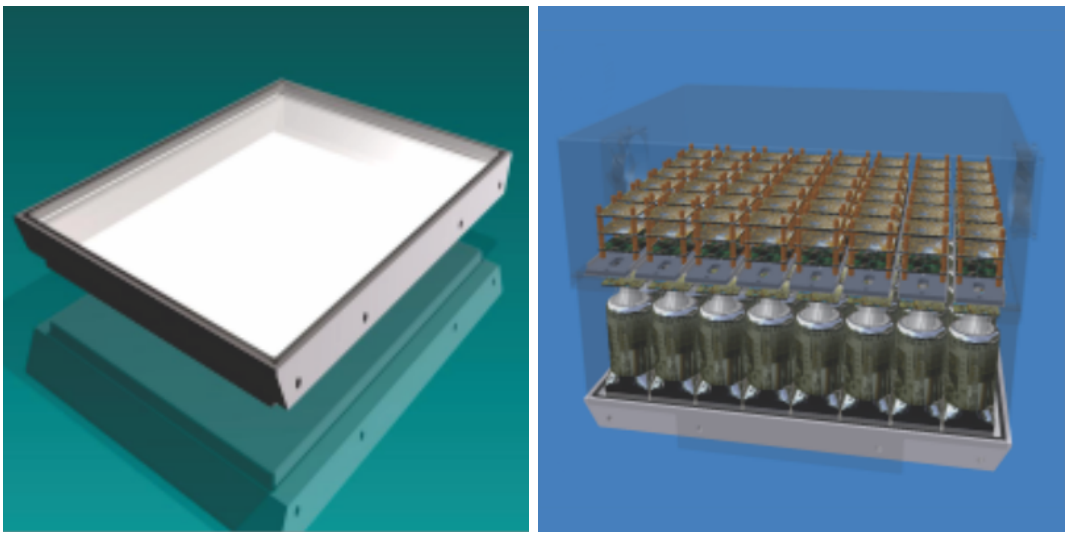
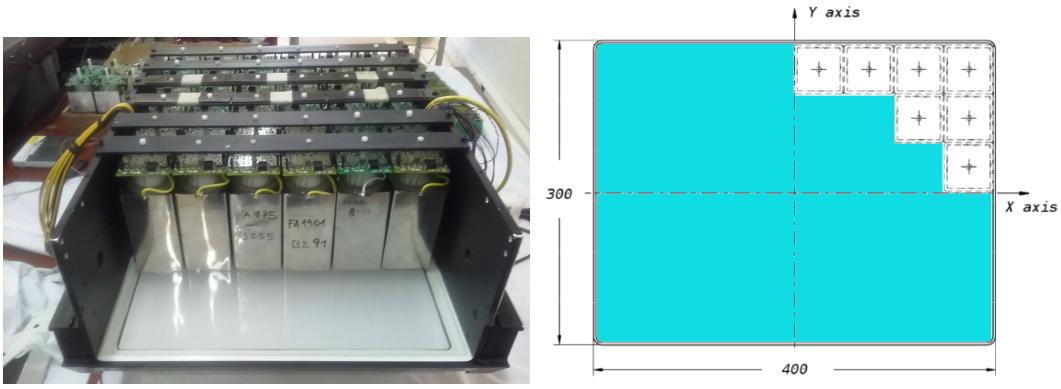


Figure A.2: Vista en perspectiva de la disposición interna con el sistema de rotación del prototipo AR-PET.



(a) (b)

Figure A.3: Cristal centellador (a) y cámara de detección completa (b) del tomógrafo AR-PET.



(a) (b)

Figure A.4: Disposición de los tubos fotomultiplicadores en la cámara (a) y el esquema de posicionamiento (b). En (b) la unidad de detección es presentada en blanco sobre un cristal de centelleo coloreado (Imagen cortesía de E. Venialgo, 2011 IEEE N.S.S).

A.3 Selección de tubos fotomultiplicadores con algoritmos genéticos

Un problema conocido en la construcción de cámaras de centelleo de gran tamaño con cristales continuos son los efectos de una mala configuración que afecta directamente la resolución final de la cámara. Este problema se hace aún más complejo cuando el número de PMTs usado es grande. Éste es el caso del AR-PET, donde 48 PMT son ubicados en cada una de las seis cámaras de centelleo según la figura A.4. Luego de la interacción del fotón gamma con el cristal centellador, cientos o miles de fotones luminosos son creados en un destello omnidireccional. Algunos de estos fotones son directamente absorbidos por PMTs y algunos son reflejados contra las paredes del cristal antes de llegar a un PMT. Por lo tanto los PMTs cerca del borde del cristal reciben una cantidad de luz significativamente mayor que aquellos en el centro. Además los PMT tienen una gran dispersión de unidad a unidad, dadas las mismas condiciones eléctricas de operación. Para alcanzar un conteo uniforme para toda la superficie del cristal es necesario ecualizar la ganancia energética y la eficiencia cuántica de cada PMT. Además, para la aplicación PET,es necesario ecualizar el tiempo de tránsito de cada PMT. Dicho tiempo de transito del PMT está define el tiempo que le lleva al PMT transformar y amplificar la señal óptica en un pulso eléctrico medible. Este tiempo debe ser igual en todos los PMT para evitar pérdidas de integración en el cálculo de la energía del fotón gamma. El tiempo de tránsito depende del valor de Alta Tensión (HV) aplicada en el PMT [86]. Por lo tanto esta está fija para todos los PMT de la cámara. Por lo tanto para controlar la ganancia del PMT, la tensión entre los primeros dinodos es controlada [87], sin cambiar el tiempo de tránsito de la unidad. Existen distintas aproximaciones para reducir el efecto de la dispersión de ganancia del fotomultiplicador. Lo más común es utilizar coeficientes programáticos y

recorte de cuentas (*count skimming*) [88] o haciendo calibración a cada PMT usando su información espectral [89]. Los métodos por hardware también fueron estudiados, como técnicas de cuadrante compartido de PMT [90] y también incluyendo Diodos Emisores de Luz (LED) dentro de la cámara de centelleo para calibrar los PMT por grupos [91] o directamente calcular las matrices de correlación a nivel hardware [92]. La reducción de los grupos de fotomultiplicadores por medio de segmentar el cristal es también una decisión diseño que reduce el impacto de este problema [93] pero no es aplicable para cristales continuos. Todos los métodos citados están enfocados en la corrección de la ganancia de los PMT luego de que los mismos hallan sido instalados dentro de la cámara centelladora, dado por sentado que cada una de las unidades puede ser posicionada en cualquier lugar sobre el cristal centellador. Esto puede ser cierto si uno puede elegir dentro de un conjunto de PMT grande y puede, por lo tanto, elegir aquellos PMT con características similares. En el caso que la cantidad de PMT a ser utilizados no es lo suficientemente grande como para descartar grandes variaciones de ganancia entre los PMT, alguna estrategia de posicionamiento debe ser aplicada.

Para poder crear cámaras de centelleo funcionales utilizando pocos PMT sobrantes y así mismo mejorando la eficiencia de adquisición inicial, un nuevo método fue desarrollado el cual genera automáticamente configuraciones subóptimas de PMT para cada cámara de centelleo utilizando aglomeración por k-vecinos más cercanos (kNN) y algoritmos genéticos para explorar las posibles configuraciones.

A.3.1 Reflexiones en el cristal

La definición de una configuración de PMTs dentro de la cámara de centelleo requiere conocer el comportamiento de tanto los bordes del cristal como de la modulación del dinodo del PMT. El primer efecto es una consecuencia de la construcción del cristal, el cual esta recubierto con una pintura reflectante para dirigir todos los fotones generados hacia la superficie del cristal, la estructura de la cámara que contiene los PMT y el acople óptico de los PMT con el cristal. La pintura reflectante es importante para evitar perdidas de luz en los bordes del cristal y permite a la cámara medir correctamente la energía depositada por el fotón gamma. Como efecto adverso, un PMT ubicado en el borde recibe más luz si un fotón gamma impacta sobre su centro que si el mismo PMT estuviera en el centro del cristal y un fotón impactara sobre esté. Usando el programa *Geant4 Application for Tomographic Emission* (GATE, Aplicación de Geant4 para Tomografía de Emisión) se realizó una simulación de Monte Carlo de la cámara de centello del AR-PET usando PMT ideales (conteo de fotones perfecto). Una fuente gamma uniforme y plana fue ubicada frente a la cámara. El efecto de centelleo y transporte de fotones visibles fue simulado, obteniendo el número de fotones recibido por cada PMT en cada posición del cristal. Los resultados de la simulación muestran tres grupos distintos de sensibilidad: el centro, los bordes y las esquinas. Estando compuesto el grupo central por aquellos PMT que no se encuentran directamente sobre un borde del cristal. De estos tres grupos se obtuvieron dos factores. Estos factores relacionan la energía observada cuando se detecta un evento en un borde o en

una esquina de la cámara contra la detección del mismo evento en centro de la misma:

$$\nu_{c,c}^{\texttt{Simulation}} = \frac{\sum_{p=1}^{P\in\texttt{Corner}} \zeta_p}{\sum_{p=1}^{P\in\texttt{Center}} \zeta_p} = 1.52, \tag{A.1}$$

$$\nu_{c,e}^{\texttt{Simulation}} = \frac{\sum_{p=1}^{P\in\texttt{Edge}} \zeta_p}{\sum_{p=1}^{P\in\texttt{Center}} \zeta_p} = 1.21, \tag{A.2}$$

donde P es el número total de PMT, $P \in \texttt{Center}$, $P \in \texttt{Border}$ y $P \in \texttt{Edge}$ representan el subgrupo de PMTs en cada región de la cámara y ζ es el número medio de fotones visibles detectados. Estos factores representan la relación en la cuenta de fotones para el *centro a esquina* y *centro a borde*, respectivamente.

A.3.2 Ganancia del Fotomultiplicador

Debido a la no-linealidad y la alta dispersión entre los diferentes PMTs para el mismo valor de HV y dinodos, los PMT son difíciles de caracterizar. Para probar su comportamientos se realizo un proceso de calibración personalizado. En el proceso de calibración el PMT es acoplado a un cristal de NaI(TI) de 25.4 `mm` de ancho y 76.2 `mm` de diámetro y posicionado dentro de una cámara oscura. El detector es entonces excitado con una fuente de Cesio-137 ($^{137}_{55}$Cs) y el valor del pico del ADC es anotado. Esto es repetido para cada punto de trabajo predefinido. Para cada PMT se testearon cuatro valores de HV y cinco valores de dinodo, conformando 20 puntos de trabajo distintos. Los valores de dinodo son expresados como una función del Conversor Analógico Digital (DAC) que controla el voltaje final del dinodo. El DAC del dinodo tiene 12 `bits` de resolución, lo que significa un rango de $[0, 4095]$. Los puntos de trabajo de un solo PMT pueden ser representados por cuatro curvas como se muestra en la figura A.5. Allí se observa que los puntos de trabajo de un PMT varían con el valor del dinodo siguiendo una función no lineal. El comportamiento observado es ajustado con una curva sigmoidal. Esta refleja la no-linealidad, tanto al comienzo, cuando el dinodo esta cortocircuitado, como al final, donde el mismo satura.

A.3.3 Contraste del modelo

Ya que las curvas de calibración son generadas fuera de la cámara donde funcionará el PMT, un factor de correspondencia $\chi_{\texttt{cal}}$ entre el cristal de calibración y la cámara es derivado. Este factor es calculado mediante el posicionamiento de un grupo de PMTs en cada posible posición de la cámara de centelleo y luego calibrados. Debido a que la construcción de una cámara de centelleo es un proceso complejo, solo un grupo de 48 PMTs fueron medidos. El factor obtenido relaciona la cantidad de luz observada durante la calibración y las diferentes posiciones del PMT dentro de la cámara. El factor es calculado como la proporción entre el canal ADC durante la calibración (el cual es proporcional a los fotones detectados $\zeta_{\texttt{cal}}$) y el canal de ADC dentro de la

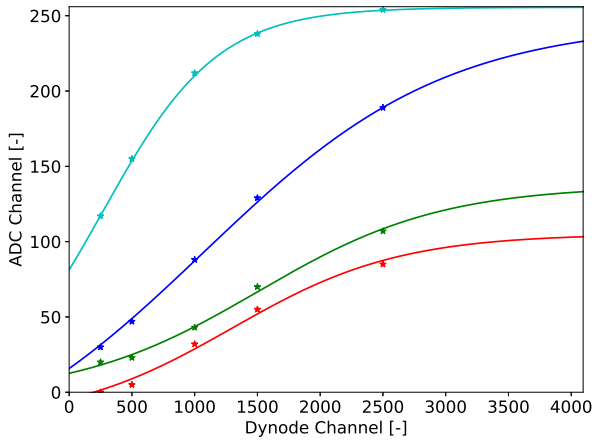


Figure A.5: Puntos de trabajo del fotomultiplicador como valor del ADC en función del valor del DAC del dinodo, $[0, 4095]$, y curvas ajustadas. Valores de HV: 1300 V (azul claro), 1200 V (azul oscuro), 1100 V (verde) y 1000 V (rojo).

cámara (proporcional a $\zeta_{\texttt{cam}}$), como se muestra en la ecuación A.3.

$$\chi_{\texttt{cal}}^{\texttt{region}} = \frac{\zeta_{\texttt{cal}}}{\zeta_{\texttt{cam}}^{\texttt{region}}}. \tag{A.3}$$

Usando los 48 PMTs la relación de luz para cada PMT puede ser extraída. Los valores de los factores de correspondencia empírica y la cantidad de fotones detectados en la simulación (ζ_p) son mostrados en la figura A.6. Tomando el valor medio sobre cada región (centro, esquina y borde), un factor de correspondencia media es obtenido. Los factores medios resultantes fueron: $\chi_{\texttt{cal}}^{\texttt{center}} = 0.44$, $\chi_{\texttt{cal}}^{\texttt{edge}} = 0.54$ y $\chi_{\texttt{cal}}^{\texttt{corner}} = 0.71$. Finalmente, comparando las ganancias en cada posición de la cámara, los mismos factores son obtenidos mediante la simulación de la sección A.3.1:

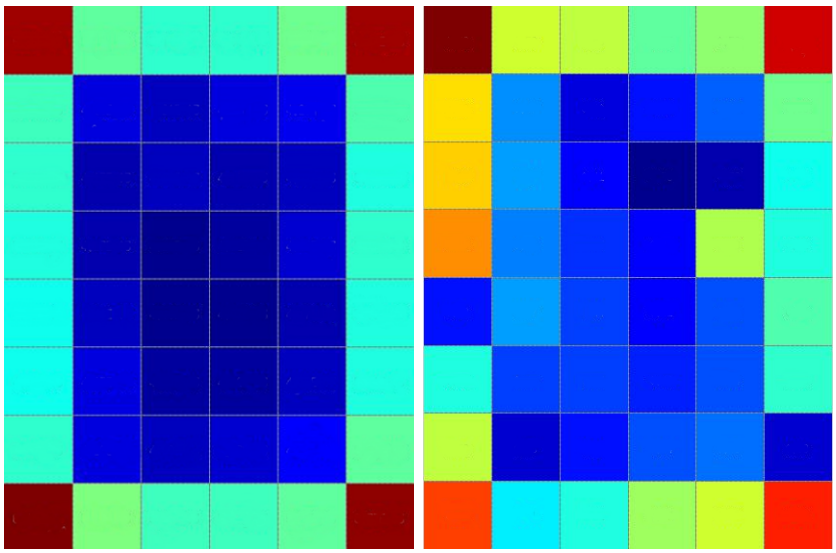
$$\nu_{c,c}^{\texttt{Empirical}} = \frac{\chi_{\texttt{cal}}^{\texttt{corner}}}{\chi_{\texttt{cal}}^{\texttt{center}}} = 1.61, \tag{A.4}$$

$$\nu_{c,e}^{\texttt{Empirical}} = \frac{\chi_{\texttt{cal}}^{\texttt{edge}}}{\chi_{\texttt{cal}}^{\texttt{center}}} = 1.22. \tag{A.5}$$

Se puede ver que estos factores son muy similares a los valores simulados $\nu_{c,c}^{\texttt{Simulation}} = 1.52$ y $\nu_{c,e}^{\texttt{Simulation}} = 1.21$, teniendo una diferencia máxima del 6%. Esto refuerza la exactitud del procedimiento de calibración.

A.3.4 Preprocesamiento por K-medias

El punto de partida del algoritmo de posicionamiento de los PMTs es un agrupamiento por K-medias. El número de las configuraciones iniciales posibles para una cámara están dadas por la cantidad de PMTs disponibles y la cantidad de PMTs a posicionar en las distintas cámaras de centello a ser construidas. Este valor se obtiene a través



(a) (b)

Figure A.6: Proporción entre la cantidad de luz esperada en el centro de la cámara gamma del AR-PET y la luz recibida en cada uno de los PMT en la simulación de GATE (a) y en el estudio experimental (b). El color de (a) representa el número de fotones detectado, que va desde más fotones (rojo) a menos fotones (azul). En (b) el color representa la correspondencia entre los factores derivados de la ecuación A.3.

de la fórmula de las permutaciones en la ecuación A.6, donde P' es el número total de PMTs y P es el número de PMTs a posicionar.

$$\texttt{Perm} = \frac{P'!}{(P'-P)!}\,. \tag{A.6}$$

Para poder generar un buen punto de partida, el cual ayuda en la convergencia del posterior algoritmo genético, los PMTs son agrupados alrededor de un valor de HV en común. Los grupos son creados eligiendo el canal objetivo del ADC para el pico de energía y midiendo la similaridad entre los valores de dinodo. Haciendo uso del algorimo k-medias [94] tres grupos de PMTs son generados para cada valor de HV. Los centros o representantes de estos grupos están dados en unidades de dinodos y sus valores no son dados a-priori. El algoritmo converge utilizando el canal ADC objetivo y los factores de correspondencia $\chi_{\texttt{cal}}^{\texttt{center}}$, $\chi_{\texttt{cal}}^{\texttt{edge}}$ y $\chi_{\texttt{cal}}^{\texttt{corner}}$. Esto significa que hay un grupo por posición en la cámara de centelleo. Si múltiple cámaras deben ser generadas, los PMTs de la primer cámara son extraídos de la lista y el proceso de selección continua hasta que todas las cámaras son pobladas.

A.3.5 Algoritmo Genético

Ya que una búsqueda exhaustiva del problema no es posible, una búsqueda subóptima es aplicada. Los algoritmos genéticos están preparados para esta tarea y está probada su convergencia [95]. Esta técnica requiere la descripción de *individuos*, los cuales en este caso son las configuraciones posibles de las cámaras de centelleo. El individuo es descrito como el orden de los PMTs dentro de cada cámara de centelleo a ser construida y el valor de HV para cada una de estas cámaras. El *genoma* resultante de cada individuo es un vector de $[(P+1)\times D] + P_{\texttt{slk}}$ elementos, donde D es el número de cámaras (o detectores) a ser construidos y $P_{\texttt{slk}}$ es la variable de afloje, que representa cuantos PMTs de *repuesto* poseen las cámaras. Estos PMT de repuesto no son considerados en el cálculo de aptitud pero son utilizados durante el proceso de mutación. Al comienzo una población de $\Xi = 30$ individuos es creada usando el agrupamiento de k-medias. La aptitud inicial de la población es calculada y luego el proceso de selección comienza. Este proceso de selección se realiza mediante la mutación y el elitismo (o réplica) [96]. Para cada iteración una cantidad $\Xi_{\texttt{elit}} > 0$ de individuos es elegida para ser copiada sin modificación a la siguiente etapa del algoritmo, esto es llamado *elitismo*. Luego un proceso de *mutación* es aplicado a la población, el proceso de *cruza* no es utilizado. El valor de aptitud ξ de cada individuo de la nueva población es calculado y la probabilidad de sobrevivir p_m del m^{esimo} individuo es dada por:

$$p_m = \frac{\xi_m}{\sum_{n=1}^{\Xi}\xi_n}. \tag{A.7}$$

Luego de este proceso de selección, una nueva población es creada y el algoritmo procede a la siguiente iteración. Esto es repetido hasta que un individuo alcanza la aptitud deseada o el número de iteraciones totales es alcanzado.

A.3.6 Cálculo de Aptitud

La aptitud de un individuo refleja la peor diferencia entre valores de dinodo entre dos PMTs a un mismo HV. Este valor de dinodo es proporcional al HV, pero es dado en unidades de bits. En este sentido, cuanto menor es el valor de aptitud mejor es la configuración (el individuo). La aptitud de la configuración es calculada como la más grande diferencia entre valores de dinodo entre todas las cámaras de centelleo. Esta métrica esta descrita por:

$$\xi_n = \max(\Delta \mathtt{v}^1_{\mathtt{dn}}, \Delta \mathtt{v}^2_{\mathtt{dn}}, ..., \Delta \mathtt{v}^D_{\mathtt{dn}}), \tag{A.8}$$

con:

$$\Delta \mathtt{v}_{\mathtt{dn}} = \max(\overline{\mathtt{v}_{\mathtt{dn}}}) - \min(\overline{\mathtt{v}_{\mathtt{dn}}}), \tag{A.9}$$

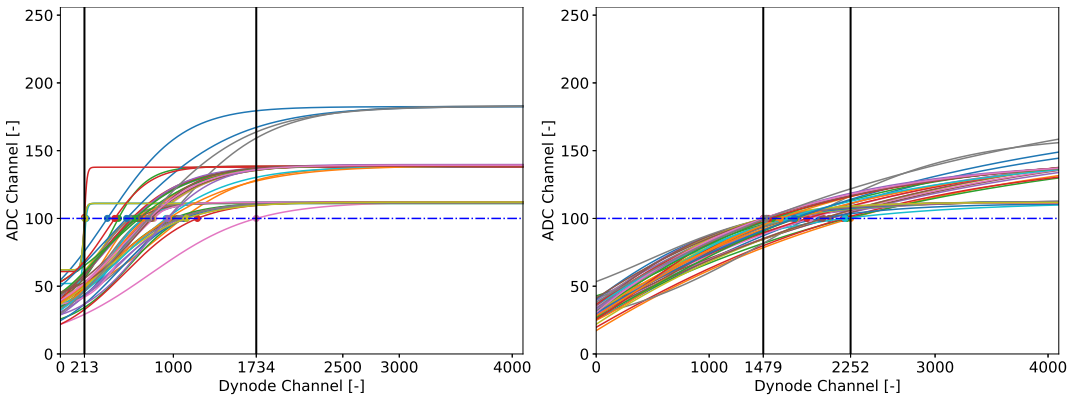
donde $\overline{\mathtt{v}_{\mathtt{dn}}}$ es el conjunto de los P voltajes de dinodo de cada PMT dentro de la cámara. Esta métrica fue seleccionada debido a que la peor configuración se da cuando un PMT esta limitado por un bajo valor de dinodo y otro esta limitado por una alto valor de dinodo. Esta situación limita las posibilidades del ajuste fino luego del proceso de selección.

A.3.7 Mutación

El proceso de mutación puede ocurrir de dos formas diferentes. La primera es la más probable, con una probabilidad del 10% para cada cámara de centelleo que compone a un individuo. Si una cámara es seleccionada para ser mutada uno de sus PMTs es seleccionado al azar, luego, otra cámara del mismo individuo es seleccionada y otro PMT de dicha cámara es seleccionado al azar. Estos dos PMTs son intercambiados y ambas cámaras son reconfiguradas usando todos los PMTs que pertenecen a cada uno de sus grupos. Si existen PMTs de repuesto, estos son considerados en ambos procesos, tanto para el intercambio como para la reconfiguración. El segundo tipo de mutación es mucho menos probable (con una probabilidad menor al 1%), y el mismo cambia el valor de HV de la cámara. Si una cámara es elegida, su nuevo valor de HV es seleccionado al azar dentro de los valores de calibración.

A.3.8 Resultados y Discusión

El proceso aquí descrito fue usado para generar $D = 6$ cámaras de centelleo para el AR-PET, cada una consistiendo en $P = 48$ PMTs. Para la construcción de las cámaras un total de $P' = 307$ PMTs estaban disponibles. Usando la ecuación A.6 el número total de configuraciones posibles puede ser calculado, el cual supera los $6 \times 10^{+614}$. Las cámaras construidas fueron comparadas con un prototipo de cámara existente. El prototipo de cámara fue creada eligiendo los PMTs manualmente del total P' PMTs, asignando aquellos de menor ganancia a las esquinas, donde iban a recibir más luz, aquellos de ganancia media a los bordes y finalmente los de alta ganancia al centro.



(a) (b)

Figure A.7: Curvas de PMTs de la cámara de centelleo prototipo, creada por selección manual de los PMTs (a) y la cámara número 4 de las cámaras generadas (b). La linea punteada horizontal representa el canal ADC objetivo, el punto coloreado en la intersección de la curva con la linea representa el punto de trabajo del PMT. El espacio entre las lineas negras verticales representa la máxima desviación entre los puntos de trabajo, lo que ilustra el valor de aptitud.

Este proceso se realizo sin ningún reparo para construir las cinco cámaras restantes, ya que era un prototipo. El canal objetivo de ADC para el pico de energía del Cesio-137 ($^{137}_{55}$Cs) fue 100 (dentro de un rango de $[0, 255]$). El valor de HV fue seleccionado en 1200 `v`. La cámara resultante tenía un valor de aptitud de 1521, calculado con la ecuación A.9. En la figura A.7(a) se muestran las curvas interpoladas para cada uno del PMTs elegidos en el HV correspondiente a la cámara prototipo. Estas curvas ya esta corregidas por los factores de ganancia $\chi_{\texttt{cal}}$. Las cámaras del AR-PET fueron generadas con el proceso descrito, usando una población de $\Xi = 30$ individuos, un valor de elitismo del 10%, una probabilidad de mutación del 6% para el primer tipo y de 0.6% para el segundo y se procesó durante 1000 iteraciones. No se aplico un criterio de finalización temprana. Los valores de aptitud de las seis cámaras están resumidos, junto con la cámara prototipo, en la tabla A.1. En la figura A.7 las curvas para los PMTs de la cámara generada con el peor valor de aptitud, el cual define la aptitud del individuo y la cámara prototipo son mostradas. Se puede observar que el valor de aptitud, representado por el ancho entre las lineas negras verticales es mucho menor en la cámara generada.

El agrupamiento por k-medias junto con el algoritmo genético probaron reducir la dispercion de los valores de dinodo dentro de la cámara de centelleo de manera significativa. También fue posible encontrar configuraciones viables para todas las cámaras requeridas con una cantidad de PMTs sobrantes menor al 7% del total de unidades disponibles. Las cámaras construidas con este método alcanzaron un 15% de FWHM (sin ajuste fino posterior), mejorando los resultados previos del 20% [97]. Aún más, las cámaras construidas tienen puntos de trabajo en la zona linear de la curva

Table A.1: Valores de aptitud de las cámaras de centelleo

Cámara	HV [v]	Aptitud
1	1200	759
2	1100	771
3	1100	741
4	1300	773
5	1200	757
6	1200	684
Prototipo	1200	1521

voltaje de dinodo-ganancia de los PMTs.

A.4 Algoritmos de Localización de Eventos

La mayoría de los tomógrafos tienen su resolución de localización de eventos sobre sus detectores fijadas por el tamaño de los pixeles del mismo. Un pixel se encuentra normalmente alrededor de 4 `mm` y llega hasta 3.2 `mm` en sistemas modernos [98]. Los sistemas comerciales en general no proveen información sobre la profundidad de interacción de los eventos (DoI), por lo que se busca que los cristales sena lo más finos posibles. Como el tomógrafo AR-PET utiliza cristales monolíticos y de gran tamaño, la localización de los eventos debe hacerse por medio de los valores de intensidad detectados sobre los PMT cuando ocurre una interacción de un fotón gamma. La salida del p^{esimo} PMT es un valor proporcional al total de los fotones detectados $e_p \approx \zeta_p$. La posición es entonces derivada matemáticamente del vector de energía adquirido $\overline{e}$. Esto puede hacer por medio de distintos métodos. El método estándar para posicionar las interacciones es el algoritmo de Centro de Gravedad (CoG) o Anger [218], el cual provee un método simple para la localización de la interacción sobre la superficie del cristal. Sin embargo el algoritmo CoG posee dos grandes desventajas: primero, no provee información de DoI que es crítica para tomógrafos de con un gran Campo de Visión (FoV) axial o cuando el ancho del cristal es comparable con el FoV (como es el caso de tomógrafos PET para animales pequeños). Segundo, el posicionamiento de la interacción es severamente distorsionado en los bordes del detector debido a la reflexión de luz o perdidas en los bordes del cristal de centelleo y la falta de información de otro PMT o SiPM vecino. Las limitaciones del algoritmo CoG llevaron a proponer distintos métodos para mejorar la localización de interacciones gamma. Uno de estos métodos es la utilización de Tablas de Búsqueda (LuTs) construidas usando una serie de mediciones y luedo cada nuevo evento es comparado con los existentes en la tabla y la posición del evento más cercano en la tabla es obtenida. Varias técnicas fueron propuestas para hacer la búsqueda y estimar la posición del evento, como por ejemplo máxima verosimilitud [38], k-vecinos más cercanos [219] y error cuadrático medio [40]. Además de las LuTs, métodos estadísticos fueron propuestos, como potenciación del gradiente de árboles [99], el cual entrena un modelo de posicionamiento estadístico

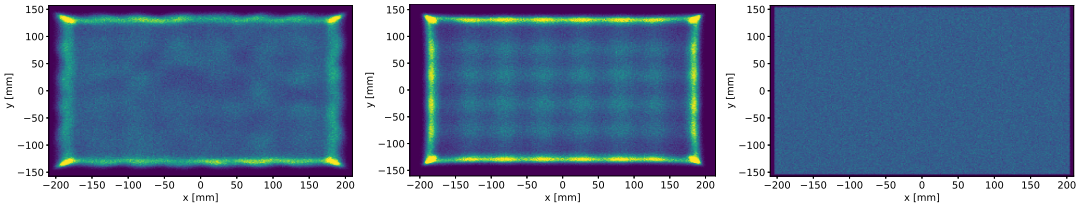
basado en una medición de calibración rápida realizada por un abanico de rayos. El uso de Redes Neuronal (NNs) también fue propuesto. En Bruyndonckx [100] y Venialgo [101] proponen entrenar una NN usando un conjunto de datos construido por medio de mediciones empíricas, como rayos colimados. En Conde [37] y Ibora [102] proponen entrenar la NN con un conjunto de datos generado a partir de un modelo matemático, para reducir el tiempo de calibración y la complejidad. Todos estos métodos se basan en la generación de un conjunto de datos que contiene para cada posición de impacto en el cristal lo valores de energía esperados. Un conjunto de datos de estas características no es fácil de construir y es una de los mayores impedimentos para implementar estas técnicas.

A.5 Localización de Eventos utilizando una Red Neuronal de Emparejamiento de Distribuciones

Aquí se propone un método para corregir la distorsión del algoritmo de CoG y obtener un valor de DoI utilizando NNs y una adquisición simple de campo inundado cuyo único requisito es conocer la distribución de eventos esperada. El método no requiere que se realice ningún exploración de la superficie con rayos colimados. Se basa en la transformación de la distribución de eventos obtenida mediante el algoritmo de CoG (u otro) en la distribución esperada, corrigiendo la posición de aun evento a la vez. El método es incluso capas de estimar la DoI siguiendo una estrategia de dispersión de luz. Aún más, el método mejora el FWHM del algoritmo CoG. El método fue probado usando una adquisición de un modulo de detección del AR-PET y una simulación de Monte Carlo. El método es comparado con el método de CoG (o Anger) y una estimación de DoI basada en la dispersión de luz, utilizando adquisiciones de rayos colimados.

A.5.1 Introducción

La distribución de las interacciones de los fotones gamma dentro del cristal depende de la energía, el ángulo de incidencia, la posición de la fuente y el poder de frenado del material del cristal. Esta distribución puede ser observada empíricamente aplicando un histograma 3D a un gran número de interacciones. Un algoritmo de posicionamiento perfecto recuperaría la distribución de interacciones teórica dentro del cristal, sin embargo, un algoritmo de posicionamiento real va a generar una distribución distinta. Por ejemplo, una fuente puntual posicionada lejos del centro del detector producirá una distribución de interacciones uniforme, a lo largo de la superficie del detector. Si estas interacciones son posicionadas con el algoritmo de Anger, el cual sufre de distorsión en los bordes del cristal, el histograma resultante va a mostrar efectos de acumulación de eventos en los bordes del detector, como se observa en la figura A.8. El efecto de acumulación observado en los limites del detector lleva a tener que desestimar parte del área sensible, ya que la localización no es confiable. Este área tiene el tamaño de un PMT y se la denomina *zona muerta* (ó *dead zone* en idioma inglés).



(a) Experimental (b) GATE (c) Expected

Figure A.8: Histograma 2D de la distribución empírica (a) y la simulada (b) del método de CoG, y la distribución esperada (c), en el plano (x, y) de la cámara de detección del AR-PET. Las inhomogeneidades en (a) y (b) son causadas por los errores de posicionamiento del algoritmo de CoG.

Esta diferencia entre la distribución esperada y la estimada puede ser usada como una métrica del efecto de acumulación, o distorsión, del algoritmo de posicionamiento. Más formalmente, si la distribución esperada $\boldsymbol{R}$ y la distribución del posicionamiento, o estimada, $\hat{\boldsymbol{R}}$ son conocidas, entonces se puede obtener una métrica de la distancia entre ambas:

$$d = J_{\texttt{dist}}(\boldsymbol{R}, \hat{\boldsymbol{R}}), \tag{A.10}$$

donde $J_{\texttt{dist}}()$ es una función de distancia. La forma de la función de distancia debe tener en cuenta cada evento en particular por separado, ya que la posición individual de estos es importante. Este tipo de métricas son normalmente usadas en el campo del Transporte Óptimo (OT). Ellas miden esta distancia como el costo de mover cada una de las muestras de una distribución a la posición de una de las muestras en la distribución objetivo. Posibles candidatos para $J_{\texttt{dist}}()$ son la distancia del movimiento de tierra (EMD), también conocida como métrica de Wasserstein [103] (implementada para problemas de OT de gran escala en [104]), o la Métrica de Distancia de Chamfer (CDM). La CDM es más fácil de calcular que la EMD y provee gradientes precisos que resultan en un desempeño similar que la EMD para el entrenamiento de NNs [105]. La CDM encuentra para cada evento en el primer conjunto su vecino más cercano en el segundo conjunto, calcula la distancia cuadrada entre ellos y acumula todas estas distancias. Puede ser expresada como:

$$J_{\texttt{dist}}(\boldsymbol{R}, \hat{\boldsymbol{R}}) = \sum_{\boldsymbol{r}_n \in \boldsymbol{R}} \min_{\hat{\boldsymbol{r}}_m \in \hat{\boldsymbol{R}}} \|(\boldsymbol{r}_n - \hat{\boldsymbol{r}_m})\|_2^2 + \sum_{\hat{\boldsymbol{r}}_m \in \hat{\boldsymbol{R}}} \min_{\boldsymbol{r}_n \in \boldsymbol{R}} \|(\boldsymbol{r}_n - \hat{\boldsymbol{r}}_m)\|_2^2, \tag{A.11}$$

donde $\boldsymbol{r}$ es la posición 3D de una interacción. Usando la ecuación A.11 se puede optimizar a una NN para reducir esta distancia, siempre y cuando se conozca la distribución objetivo $\boldsymbol{R}$ y se puede muestrear de ella. Por lo tanto, minimizando d resulta en una minimización entre la posición esperada y la estimada de un evento.

A.5.2 Generación del conjunto de datos

El conjunto de datos se obtuvo utilizando una cámara de centelleo del AR-PET. La cámara esta compuesta por un cristal continuo de NaI(TI) de $406.4 \times 304.8 \times 25.4$ $\texttt{mm}^3$ con una guía de luz de 9.5 $\texttt{mm}$, poblada con una formación de 8×6 PMTs R1534 de Hamamatsu. Estos PMTs tienen un perfil rectangular de 52.4×52.4 $\texttt{mm}^2$ que cubre completamente la superficie del cristal. Aunque el método no requiere ninguna distribución objetivo en particular, por simplicidad se utiliza una distribución uniforme y ortogonal. El campo de eventos es generado utilizando una fuente de sodio $^{22}\texttt{Na}$ β^+ (que genera fotones 511 $\texttt{keV}$ mediante descomposición electrón-positrón) ubicada a una grán distancia $d_s > 3$ $\texttt{m}$ y alineada con el centro de la superficie de detección del detector. Dada la distancia de la fuente a la superficie del detector, se puede considerar que los fotones son perpendiculares a la superficie del cristal. Usando esta configuración $N = 3 \times 10^7$ interacciones son obtenidas. Estos eventos fueron filtrados por una ventana energética entre 470 $\texttt{keV}$ y 530 $\texttt{keV}$. Cada interacción n es guardada como la distribución de energía $e_p^n \in \overline{e}_n$ detectada en cada detector p de la cámara de centelleo. El conjunto de datos base $\tilde{\boldsymbol{R}}$ es entonces construido mediante el calculo de la posición de interacción estimada para evento $\tilde{\boldsymbol{r}}_n = (r_x^n, r_y^n, r_z^n)$. La posición en el plano (r_x^n, r_y^n) es calculada usando el algoritmo de CoG. El DoI (r_z^n) se obtiene a partir de una estimación simple basada en la dispersión de luz, un conocido indicador del DoI [106]. Esta métrica se construye usando la información de la distribución de energía en el conjunto de datos. Primero la superficie del detector es dividida en D píxeles de 12×12 $\texttt{mm}^2$. Luego, usando el posicionamiento del CoG el conjunto de datos es dividido en subgrupos, la relación de energía detectada en los 4 PMTs con mayor energía contra la energía total es calculada:

$$\eta_n^k = \frac{\sum_{p=1}^{4} e_p^{n*}}{\sum_{p=1}^{48} e_p^{n*}}, \tag{A.12}$$

donde $\overline{e}_n^*$ son los cuatro valores de energía más altos para el n^{esimo} evento, k marca el píxel correspondiente y p el número de PMT. Por cada píxel k, la distribución de los valores η^k es ajustada usando una distribución normal. Finalmente un mapa de mínimos $\eta_{\texttt{min}}$ y máximos $\eta_{\texttt{max}}$ de valores η^k es construido usando el intervalo 2σ de las distribuciones ajustadas. Sabiendo la dispersión máxima y mínima por píxel, el DoI de un evento n interactuando en el píxel k^{esimo} puede ser estimado linealmente:

$$r_z^n = D_{\text{thkns}} \frac{\eta_n - \eta_{\texttt{max}}^k}{\eta_{\texttt{min}}^k - \eta_{\texttt{max}}^k}, \tag{A.13}$$

donde D_{thkns} es el ancho del detector. Un conjunto de datos generado con una simulación Monte Carlo del experimento fue generado con el software GATE. La simulación incluyo el seguimiento de los fotones ópticos y todos los elementos reflectantes de la cámara de centelleo. El conjunto de datos simulado juntó aproximadamente la misma cantidad de eventos que el conjunto de datos empírico.

A.5.3 Muestreo de la Distribución Objetivo

La distribución objetivo es obtenida mediante una simulación Monte Carlo de la configuración experimental en el software GATE. Usando esta simulación se obtiene la función de distribución acumulada (CDF) en cada eje. Se debe notar que esta simulación no requiere el seguimiento de los fotones óptimos. Dadas las condiciones experimentales, las interacciones sobre los ejes (X, Y) siguen las distribuciones $U(-203.0, 203.0)$ y $U(-152.0, 152.0)$ respectivamente. El eje Z sigue la distribución esperada para un cristal de NaI(TI) [79]. El eje Z es entonces muestreado utilizando el método de la CDF inversa [108]. Para lograr esto, la CDF es calculada y luego la inversa de la misma es aproximada utilizado un ajuste por polinomios. La función polinomial obtenida es entonces utilizada para convertir muestras obtenidas de una distribución $U(0, 1)$ en la distribución objetivo.

A.5.4 Red Neuronal

Para poder ajustar la función de localización de las interacciones, se utilizo una NN totalmente conexa. La entrada de la NN es el vector de entrada $\overline{e}_n$, compuesto de los 48 valores de energía. La NN posee una capa de entrada de 64 neuronas, seguida por tres capas ocultas de 32, 16 y 8 neuronas respectivamente, todas *rectificadoras* (ReLu). La capa de salida tiene 3 neuronas, con activación lineal y salidas correspondientes a cada uno de los ejes (x, y, z). La red neuronal es entrenada en dos pasos, primero es entrenada para aprender el posicionamiento del método base (como lo es el CoG) usando una función objetivo simple, como la distancia euclídea:

$$J_{\texttt{base}}(\tilde{\boldsymbol{R}}, \hat{\boldsymbol{R}}) = \frac{1}{N} \sum_{n=1}^{N} \|\tilde{\boldsymbol{r}}_n - \hat{\boldsymbol{r}}_n\|_2^2, \tag{A.14}$$

donde N es el número de interacciones en el conjunto de datos y $\tilde{\boldsymbol{r}}_n \in \tilde{\boldsymbol{R}}$ es la posición de la interacción obtenida con el método base. Este entrenamiento esta hecho con el algoritmo de Estimación Adaptativa de Momentos (ADAM) [109], con una taza de aprendizaje de $\epsilon = 0.001$. La red es entrenada por 500 épocas y con un tamaño de lote de $N' = 5 \times 10^3$ interacciones. Luego que el entrenamiento se completa la NN es reentrenada para corregir los errores de posición mediante la inclusión de una función de pérdida por apareo, presentada en A.11:

$$J_{\texttt{match}} = K\, J^*_{\texttt{base}}(\tilde{\boldsymbol{R}}, \hat{\boldsymbol{R}}) + J_{\texttt{dist}}(\boldsymbol{R}, \hat{\boldsymbol{R}}), \tag{A.15}$$

donde $K = 0.001$ es una constante de acoplamiento. Una vez más, la NN es entrenada utilizando el algoritmo ADAM por 500 épocas.

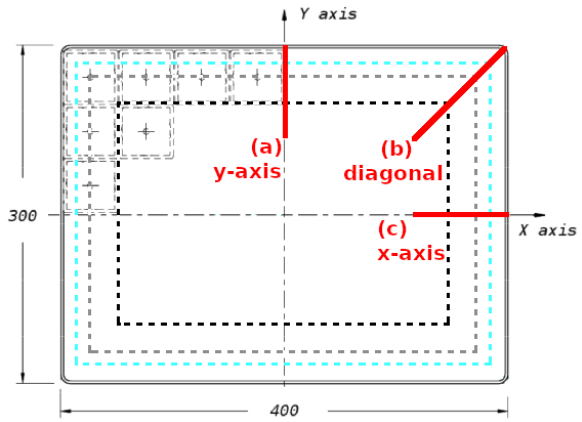
A.5.5 Resultados

El método propuesto fue probado en dos NN diferentes. La primera esta preentrenada usando el conjunto de datos experimental y el algoritmo de CoG como método base y luego refinada usando la perdida por apareo de distribuciones propuesta para

corregir las efectos de acumulación en el algoritmo de CoG, esta NN se denominó *ANGER_FCN*. La segunda sigue los lineamientos de la *ANGER_FCN* pero utiliza el conjunto de datos artificial, esta NN es llamada *Artf_ANGER_FCN* y fue estudiada de manera separada.

Desempeño 2D: Rayos Colimados Perpendiculares

Usando la técnica descrita en [101] tres series de rayos colimados fueron adquiridos: el **eje y** (**y axis**) que se origina en $(0.0, 72.15)$ `mm` y se extiende hacia $(0.00, 144.38)$ `mm`, el **Diagonal** comenzando en $(123.0, 72.15)$ y extendiéndose diagonalmente hacia $(194.25, 144.38)$ y el **eje x** (**x axis**) originándose en $(123.0, 0.0)$ `mm` y extendiéndose hacia $(194.25, 0.00)$ `mm`. Todas las series tienen un paso de 26.2 `mm`. La ubicación de cada serie se puede ver en figura A.9. Las curvas de desempeño para el método de CoG (Anger) y para la *ANGER_FCN*, para cada serie, se muestra en la figura A.10. El valor medio de FWHM, tendencia media y tendencia máxima para cada estimador de posición de interacción serie de rayo colimado y zona muerta del detector, se muestra en la tabla A.2.



(a)

Figure A.9: Posiciones del colimador en el plano (x, y) de la superficie del detector. La serie "y-axis" (a), la serie "Diagonal" (b) y la serie "x-axis" (c). Las lineas punteadas negra, gris y cyan, representan un margen de uno, medio y un cuarto de PMT desde el borde del detector respectivamente.

Desempeño 2D: Prueba en GATE

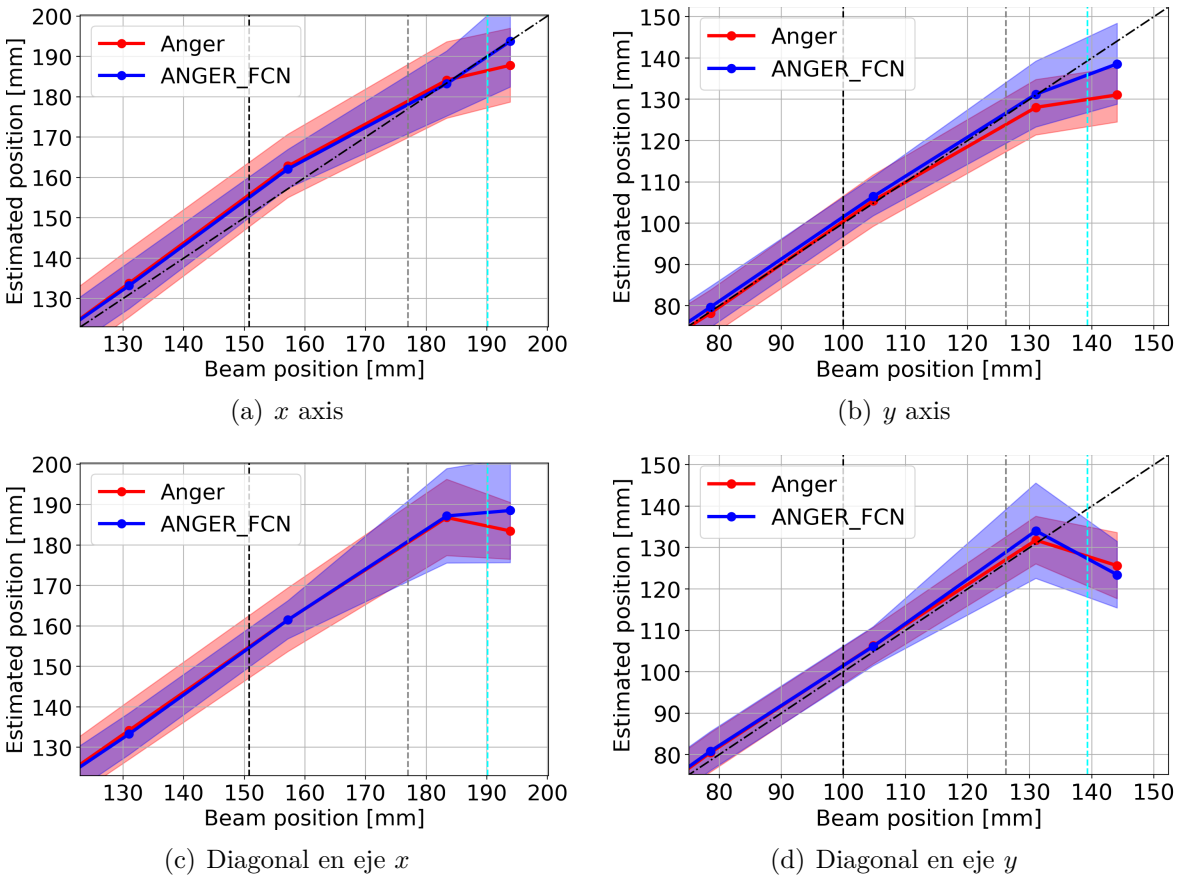
Usando la información disponible de la simulación de Monte Carlo, la Artf_ANGER_FCN fue probada contra los valores verdaderos y el algoritmo de Anger. Las regiones probadas fueron las mismas que las series de rayos colimados. Las series simuladas tienen un paso de 2.5 `mm`. El desempeño de la Artf_ANGER_FCN y el método de Anger se muestra en la figura A.11. El valor medio de FWHM, tendencia media y tendencia máxima para cada estimador de posición de interacción serie de rayo colimado y zona muerta del detector, se muestra en la tabla A.3.

Serie	Estimador	Tamaño de la zona muerta								
		52.4 mm			26.2 mm			13.1 mm		
		$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$
x axis	Anger	12.08	2.74	3.72	12.20	3.15	5.63	12.64	2.85	5.63
	ANGER_FCN	10.38	2.73	3.80	10.23	3.11	5.37	10.80	2.78	5.37
y axis	Anger	10.38	1.44	3.22	10.67	1.24	3.22	11.04	1.58	3.22
	ANGER_FCN	10.50	2.10	3.34	10.47	1.98	3.34	11.06	1.73	3.34
Diagonal - eje x	Anger	13.41	2.76	3.21	13.96	3.15	4.30	14.91	3.19	4.30
	ANGER_FCN	10.06	2.39	2.80	9.88	2.75	3.82	11.62	3.05	4.26
Diagonal - eje y	Anger	10.25	1.13	1.89	9.88	1.21	1.89	10.19	1.13	1.89
	ANGER_FCN	9.82	1.48	2.27	9.70	1.22	2.27	12.58	1.81	4.15

Table A.2: Medición empírica: FWHM medio ($\overline{\texttt{FWHM}}$), tendencia media ($\bar{b}$) y tendencia máxima ($\max b$), para cada una de las series de rayos y cada posición estimada. Las métricas fueron probadas utilizando zonas muertas del detector de uno, medio y un cuarto del tamaño de un PMT (52.4 mm, 26.2 mm, 13.1 mm respectivamente). Todas las métricas están expresadas en [mm].

Serie	Estimador	Tamaño de la zona muerta								
		52.4 mm			26.2 mm			13.1 mm		
		$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$
x axis	Anger	6.00	0.88	1.46	6.99	3.32	8.31	7.37	3.32	8.31
	Artf_ANGER_FCN	5.46	0.89	3.17	6.72	2.54	5.96	8.25	2.84	5.96
y axis	Anger	5.81	0.80	1.30	6.64	3.05	7.31	7.02	3.04	7.31
	Artf_ANGER_FCN	5.49	1.25	2.78	6.50	2.05	4.18	9.15	2.87	8.11
Diagonal - eje x	Anger	6.10	0.91	1.93	7.01	3.92	9.87	7.43	3.88	9.87
	Artf_ANGER_FCN	5.16	1.05	3.64	6.25	3.42	7.84	8.63	4.04	7.84
Diagonal - eje y	Anger	5.94	0.89	1.55	6.70	3.65	8.87	7.02	3.58	8.87
	Artf_ANGER_FCN	5.20	0.92	1.55	6.69	2.49	7.73	9.26	3.62	8.68

Table A.3: Simulación en GATE: FWHM medio ($\overline{\texttt{FWHM}}$), tendencia media ($\bar{b}$) y tendencia máxima ($\max b$), para cada una de las series de rayos y cada posición estimada. Las métricas fueron probadas utilizando zonas muertas del detector de uno, medio y un cuarto del tamaño de un PMT (52.4 mm, 26.2 mm, 13.1 mm respectivamente). Todas las métricas están expresadas en [mm].



(a) x axis

(b) y axis

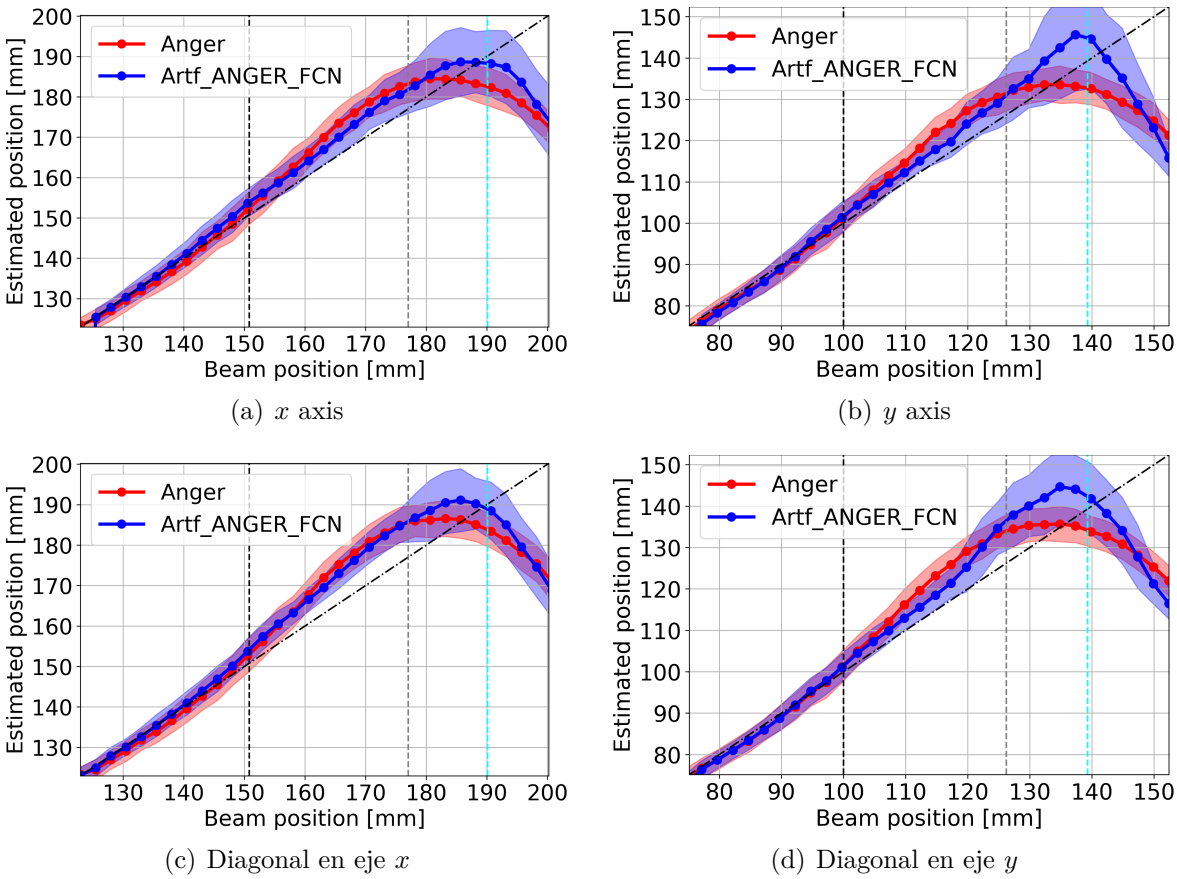
(c) Diagonal en eje x

(d) Diagonal en eje y

Figure A.10: Medición empírica: Desempeño en la serie "x axis" (a), serie "y axis" (b), serie "Diagonal" sobre el eje x (c) y serie "Diagonal" sobre eje y (d). La linea llena representa la posición media y el área coloreada representa el FWHM. Las lineas punteadas verticales en color negro, gris y cyan representan un margen de uno, medio y un cuarto de PMT desde el borde del detector respectivamente.

Desempeño en Profundidad de Interacción: Rayos Colimados Diagonales

Para poder probar el desempeño de la estimación de DoI de la NN, un rayo colimado con un ángulo de incidencia de 45° fue medido en el área central de la superficie de detección. El rayo se colocó en la posición $(0.5, 0.5)$ $\mathtt{mm}$ y su incidencia seguía el eje x. Dado el ángulo de incidencia, la DoI sigue la desviación de la posición en el eje x respecto al punto de impacto. Es decir, $r_n^z = r_n^x - r^x_{\mathtt{centrodeimpácto}}$. La DoI fue dividida en 6 secciones de 4.23 $\mathtt{mm}$ y el FWHM medio para cada sección fue calculado. El factor de correlación entre la DoI estimada y la esperada fue calculada, resultando en $\mathtt{Corr}^{\mathtt{Disp}}_{\mathtt{centro}} = 0.28$ y $\mathtt{Corr}^{\mathtt{ANGER\_FCN}}_{\mathtt{centro}} = 0.21$.



(a) x axis

(b) y axis

(c) Diagonal en eje x

(d) Diagonal en eje y

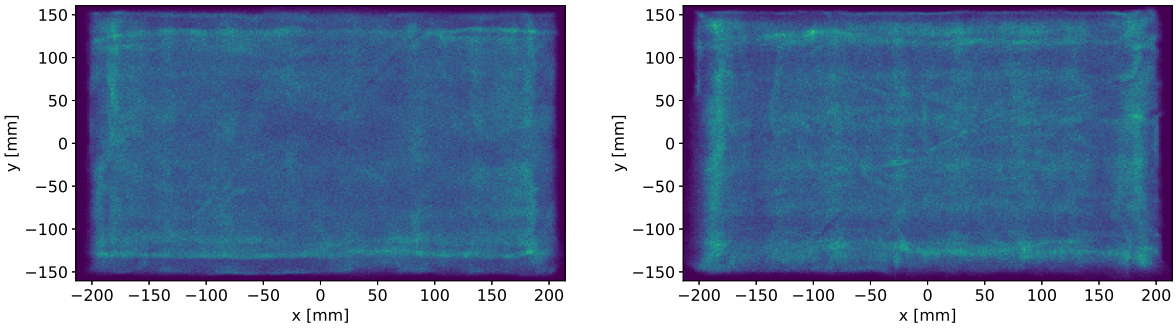
Figure A.11: Simulación en GATE: Desempeño en la serie "x axis" (a), serie "y axis" (b), serie "Diagonal" sobre el eje x (c) y serie "Diagonal" sobre eje y (d). La linea llena representa la posición media y el área coloreada representa el FWHM. Las lineas punteadas verticales en color negro, gris y cyan representan un margen de uno, medio y un cuarto de PMT desde el borde del detector respectivamente.

Desempeño en Profundidad de Interacción: Prueba en GATE

La Artf_ANGER_FCN y la estimación de DoI por el método basado en la dispersión de luz fueron probados utilizando una simulación en GATE. Dos regiones fueron probadas: Una región central de la superficie del detector, centrada en $(0.0, 0.0)$ $\mathtt{mm}$ y con un área de $50.0{\times}50.0$ $\mathtt{mm}^2$, y luego una región del borde centrada en $(0.0, -133.75)$ $\mathtt{mm}$ y con un área de 50.0×50.0 $\mathtt{mm}^2$. El valor de referencia de la DoI fue obtenido de los datos de la simulación. El factor de correlación entre la DoI estimada y la esperada fue calculada, resultando en $\mathtt{Corr}_{\mathtt{center}}^{\mathtt{Artf\_ANGER\_FCN}} = 0.80$, $\mathtt{Corr}_{\mathtt{center}}^{\mathtt{ANGER\ GATE}} = 0.79$, $\mathtt{Corr}_{\mathtt{edge}}^{\mathtt{Artf\_ANGER\_FCN}} = 0.53$ y $\mathtt{Corr}_{\mathtt{edge}}^{\mathtt{ANGER\ GATE}} = 0.50$.

Análisis de la Distribución

El apilamiento de las distribuciones resultantes es analizado en al plano (x, y) utilizando un histograma 2D de los eventos en el conjunto de datos. Para cada método de localización un histograma 2D con un tamaño de casilla de 1.56×1.56 $\texttt{mm}^2$ fue calculado. Los histogramas (x, y) resultantes pueden verse en la figura A.12.



(a) COG_NN (x, y) histogram (b) Artf_COG_NN (x, y) histogram

Figure A.12: Histogramas 2D de la ANGER_FCN (a) y la Artf_ANGER_FCN (b).

Para cada método analizado la densidad empírica a lo largo del eje x e y fue calculada usando el histograma sobre cada eje. La irregularidad e la distribución fue medida utilizando la diferencia normalizada de cuentas $\Delta C = (C_{\texttt{max}} - C_{\texttt{min}})/C_{\texttt{max}}$, donde $C_{\texttt{max}}$ y $C_{\texttt{min}}$ son los valores de cuenta máximos y mínimos dentro de los casilleros del histograma. La métrica fue aplicada en la estimación de la de densidad empírica sobre los ejes x e y. Para cada eje tres métricas fueron calculadas, una con un margen de 52.4 $\texttt{mm}$ una con un margen de 26.2 $\texttt{mm}$ y finalmente una con un margen de 13.1 $\texttt{mm}$ (uno, medio y un cuarto del tamaño de un PMT respectivamente). Estos resultados se resumieron en la tabla A.4.

Tam. zona muerta	52.4 $\texttt{mm}$		26.2 $\texttt{mm}$		13.1 $\texttt{mm}$	
	eje x	eje y	eje x	eje y	eje x	eje y
Anger	0.17	0.13	0.38	0.58	0.47	0.83
ANGER_FCN	0.17	0.12	0.24	0.24	0.29	0.39
Artf_Anger (GATE)	0.34	0.34	0.54	0.67	0.73	0.89
Artf_ANGER_FCN	0.29	0.27	0.31	0.34	0.42	0.35

Table A.4: Diferencia de cuentas normalizada para cada eje de la densidad empírica. Cada métrica fue probada usando uno, medio y un cuarto de PMT (52.4 $\texttt{mm}$, 26.2 $\texttt{mm}$, 13.1 $\texttt{mm}$ respectivamente), representando tres límites de zonas muertas diferentes sobre el detector. Un valor bajo indica mayor uniformidad.

Discusión

En este trabajo se presentó una técnica para el posicionamiento de interaccionas de fotones gamma basado en NNs. El método requiere solo una adquisición de campo

inundado y el conocimiento de la distribución esperada de las interacciones dentro del volumen del centellador. El desempeño 2D de la técnica propuesta fue probada usando una serie de rayos colimados, mostrando una mejora en los valores de tendencia y FWHM medio cuando se compara con el algoritmo de CoG (Anger). Esta mejora se aprecia cuando la zona muerta es reducida a medio PMT (26.2 mm). Una mejora sobre el FWHM es observada en la serie de rayos colimados en el "x-axis" y "Diagonal", presentados en las tablas A.2 y tabla A.3.

La capacidad de la técnica propuesta para estimar la DoI fue testeada usando un rayo colimado con un ángulo de incidencia de 45°. La NN alcanzó un factor de correlación de $\mathtt{Corr}^{\mathtt{ANGER\_FCN}}_{\mathtt{centro}} = 0.21$ contra $\mathtt{Corr}^{\mathtt{Disp}}_{\mathtt{centro}} = 0.28$ del método base. En el caso de la simulación en GATE el método de NN alcanzó $\mathtt{Corr}^{\mathtt{Artf\_ANGER\_FCN}}_{\mathtt{center}} = 0.80$ contra $\mathtt{Corr}^{\mathtt{ANGER\ GATE}}_{\mathtt{center}} = 0.79$ del método base. Un mejoramiento directo de la estimación de la DoI no fue observado, sin embargo el tamaño relativamente pequeño de la NN fue capás de reproducir el método de estimación de la DoI de base. Esto es importante ya que el método base requiere el uso de LuTs para producir la DoI, mientras que la NN esta compuesta de solo de multiplicaciones y sumas. La implementación de mejores algoritmos para estimar el DoI le permitirá a las NNs reducir el FWHM observado en la medición experimental, ya que la capacidad de la red es suficiente como para producir una mejor estimación de la DoI.

El efecto de apilamiento del algoritmo de CoG fue muy reducido. Como se observa en los histogramas presentado en las figuras A.12(a) y A.12(b) la superficie presenta menos efectos de borde cuando se comparan contra sus referencias en las figuras A.8(a) y A.8(b). El método propuesto reposicionó las interacciones "apiladas" a su ubicación más cercana, siguiendo un esquema de transporte óptimo. Una métrica de diferencia de cuentas fue calculada para reflejar este efecto y es presentada en la tabla A.4. La tabla muestra una fuerte reducción en la variabilidad de cuentas en la adquisición de campo inundado.

La mejora en la aptitud y la reducción de la diferencia de cuentas posibilita la reducción de la zona muerta del detector de 52.4 mm a 26.2 mm, incrementado el área efectiva de los detectores del 48.7% al 72.1% de la superficie del cristal de centelleo si un máximo ΔC de 25% es tolerado. Esto resulta en un incremento del 50% en la detección de eventos simples $0.721^2/0.487^2 = 2.19$ veces más de probabilidad de detección de una coincidencia en un sistema PET.

El método propuesto fue probado utilizando una técnica de posicionamiento base simple como lo es el CoG, sin embargo el mismo puede ser utilizad con cualquier otro método de posicionamiento base. Mejorando la aptitud del método base resulta en una mejora de la aptitud final dela NN. siempre y cuando la NN tenga la capacidad suficiente. El método presentado requiere una única adquisición para ser implementado, en contraste con otros métodos que requieren conjuntos de datos cuidadosamente adquiridos para entrenar sus NNs. Por otra parte el método puede usarse para corregir o adaptar los métodos basados en modelos, como los presentados en [37] y [102]. Esta adaptación es requerida cuando el modelo teórico no se aparea perfectamente con el detector real. El entrenamiento de la NN puede ser mejorado incluyendo limitaciones de suavidad en la función objetivo. De esta manera los efectos de la baja estadística en el conjunto de datos de entrenamiento (una baja relación de $\mathtt{imapacts/mm}^3$) pueden

ser aliviados. El tamaño de la NN es también un factor crucial para la aptitud final de la red. Sin embargo la implementación de la NN en detectores PET requiere tiempos de procesamiento bajos.

A.6 Discusión y Comentarios

En este capitulo se presentó el AR-PET y sus capacidades técnicas. Los requerimientos técnicos del AR-PET inspiraron los métodos desarrollados en este capitulo. Primero usando algoritmos genéticos se desarrolló un método de selección de PMTS. Éste mostró ser mejor que la selección manual de los mismos. Además el método probó ser apto para encontrar configuraciones de cámaras cuando el número de PMTs disponible esta altamente limitado. El segundo trabajo se enfocó en la localización de interacciones de fotones gamma dentro del cristal de centelleo. Esta tarea desarrolla un papel central en la exactitud final del tomógrafo. Ya que las cámaras del tomógrafo AR-PET son grandes con cristales sólidos y PMTs de gran tamaño, la tarea de localización no es trivial y nuevas técnicas debieron ser estudiadas. El método propuesto tomó ventaja de la capacidad de procesamiento de señal del AR-PET para implementar algoritmos de posicionamiento no lineales. Estos algoritmos están basados en redes neuronales y es posible embeberlos completamente dentro de FPGAs. Ya que las cámaras del AR-PET no proveen otra información que los valores de los PMTs ubicados sobre la superficie del cristal centellador, una nueva forma de entrenar las redes fue ingeniada. Este procedimiento de entrenado utiliza solamente la información de una única adquisición de campo inundado para obtener todos los datos de entrenamiento y luego trata de aparear la distribución de los eventos dentro del volumen del cristal a la distribución esperada de los mismos. El método fue probado en las cámaras del AR-PET, transformando la distribución del CoG (Anger) en la distribución esperada, mejorando el área sensible de la cámara y proveyéndola con capacidad de estimar la DoI a un costo computacional bajo. Sin embargo el método es valido para aparear cualquier otro método de posicionamiento inicial en cualquier otro modelo de cámara, siempre que se conozca la posición de los PMTs/SiPMs, sus energías, la distribución objetivo y un método de posicionamiento inicial.

Los capítulos subsiguientes se concentran en los métodos de imagen PET. En ellos se implementan algoritmos de reconstrucción y métodos de análisis de imagen para permitir un estudio PET sin necesidad de estudios adicionales para obtener la imagen de atenuación, tomando un camino más directo hacia la reducción de la dosis en un estudio PET.

B Representación de Imágenes Tomográficas por medio de Mallas

En este capitulo se estudia el problema de la reconstrucción de imágenes por medio representación en mallas de Elementos Finitos (FE). Primero se introduce, rápidamente, el concepto de FE y luego se enfoca en la reconstrucción de imágenes de tomografía PET sobre mallas.

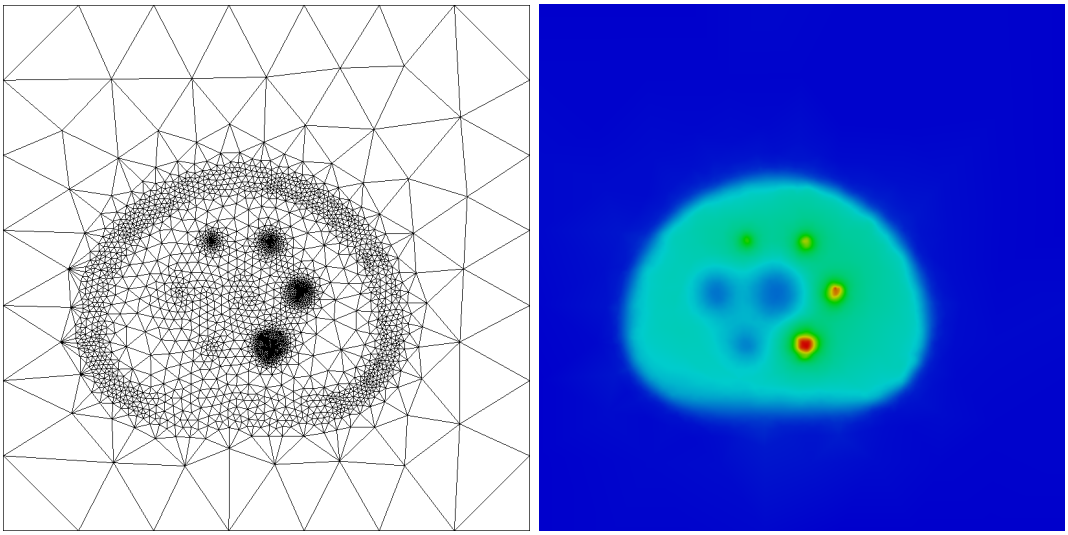
B.1 Introducción a las Mallas

El concepto de usar mallas para el cálculo de FE se origina en la discretización de un dominio encerrado en elementos sencillos. El FE busca realizar un aproximación por partes de un dominio complejo utilizando polinomios simples. Estos polinomios están solamente definidos en pequeñas regiones (elementos) del domino, donde su aproximación es válida, y luego estos son expresados en términos de los valores de los nodos (o vértices) de dichas regiones (o elementos). El conjunto de todos estos elementos define la malla. En oposición a las representaciones pixeladas o voxelizadas, donde todos los elementos tienen el mismo tamaño y están ordenados por su posición, una malla no tiene este tipo de restricciones. Una malla esta definida por sus dos características principales:

- Geometría: La posición de los nodos (o vértices) de la malla en el espacio del dominio.
- Topología: La lista de conexiones de los vértices, que describen cada elemento.

Con estas características (geometría y topología), el muestreo de una imagen puede ser adaptado para describir su soporte de manera más eficiente. En la figura B.1 se muestra la representación por mallas de un fantoma de calidad de imagen de la Asociación Nacional de Fabricantes Eléctricos (NEMA, del inglés National Electrical Manufacturers Association). Se puede ver que la malla esta optimizada para asignar elementos más grandes a las secciones de la imagen donde su actividad varía muy poco o de manera lineal, y elementos más pequeños en aquellas secciones con variaciones no-lineales. De esta manera la representación por mallas también embebe parte de la actividad como parte de su geometría.

La idea de utilizar soporte de mallas para las imágenes médicas no es nueva. Ésta ha sido ampliamente estudiada para la simulaciones hemodinámicas [113], simulación de sistemas respiratorios [114] y registro de imágenes médicas [115, 116]. En el contexto



(a) (b)

Figure B.1: Representación por mallas (a) del fantoma NEMA y su correspondiente actividad reconstruida (b).

de la reconstrucción de imágenes los métodos de mallas no han sido adoptados, sin embargo aplicaciones para PET y SPECT existen.

B.2 Reconstrucción de la Imagen de Actividad PET sobre Mallas 2D

B.3 Introducción

El uso de un soporte por mallas par los procesos de reconstrucción en imágenes médicas ha sido estudiado en el pasado para PET [128–131] y SPECT [132]. Estos algoritmos de reconstrucción están basados en MLEM u OSEM. Estos estudios muestran que la proyección a través de un soporte reducido puede ser más rápida que en un soporte pixelado para reconstrucción 2D [132]. También los trabajos de Sitek, Pereira y Boutchko [128,130,131] reportaron una calidad de imagen similar o superior cuando se utiliza un soporte de mallas o de nubes de puntos. En estos estudios la calidad del soporte es resaltada como un aspecto importante de la reconstrucción, sin embargo solo técnicas simples de actualización de la malla son aplicadas, dado el costo de calcular un soporte óptimo. La representación utilizada en [132] es fija y basada en una imagen inicial voxelizada, obtenida a través de Retro-Proyección Filtrada (FBP). Esta imagen inicial es mallada usando un modelo de malla adaptado al contexto [134] basado en el muestreo no uniforme de un mapa de características obtenido de la distribución espacial de la segunda derivada direccional de la imagen inicial. El método de mallado entonces controla el error mediante el control del número de nodos que son colocados,

los cuales controlan subsecuentemente usando el principio de largo de descripción mínima (MDL) [135]. En [129] se implementa una representación por mallas 2D que se adapta dinámicamente a la distribución del radiotrazador por medio del remallado. La representación inicial es una malla gruesa aleatoria que se adapta a la distribución del radiotrazador en cada paso de la reconstrucción. La malla es guiada por un mapa de características basado en el hessiano de la imagen reconstruida y los nodos son colocados usando el algoritmo de difusión de error de Floyd–Steinberg [136]. Su método de reconstrucción y representación es 2D, no se da una extensión del mismo para reconstrucción 3D. La generación de mallas presentada en [128] es dinámica y basada en un simple movimiento de puntos donde los nodos son agregados dentro los tetraedros que presentan la mayor variación entre sus cuatro nodos, esta es una técnica simple de bajo costo computacional pero el número final de nodos y la calidad de la malla no es controlada. En un trabajo posterior [130] proponen empezar con una grilla densa la cual es subsecuentemente rebajada para obtener un soporte de malla que no pierda ningún detalle. En este último trabajo ese destaca la necesidad de mejorar la calidad de la malla soporte para evitar artefactos en forma de cuña e inestabilidad numérica debido a elementos de la malla degenerados.

El método que se presenta aquí difiere de los anteriores en dos aspectos principales: primero el soporte de malla esta basado en una colección de mallas 2D que representan un volumen 3D. Esta representación permite implementar una reconstrucción 3D mientras se utilizan técnicas de remallado 2D que son más simples y pueden ser implementadas en paralelo. Segundo, el soporte de malla se adapta dinámicamente a la distribución, empezando por una malla gruesa sin ninguna configuración particular. La ubicación y el número de nodos de la malla está determinada usando una estrategia que controla el error de interpolación y la calidad de la malla. La combinación de estas técnicas produce una representación compacta de alta calidad del proceso de reconstrucción. La exactitud está gobernada a través del error de interpolación del soporte. Se muestra que dicha combinación permite reducir el número de parámetros por más de un orden de magnitud.

El método presentado usa una representación en modo lista. Este modo se eligió debido a las necesidades del AR-PET.

B.4 Materiales, Métodos y Teoria

En esta sección se describen las operaciones de proyección y retro-proyección sobre la malla, seguido por los métodos de optimización de actividad y de la malla. Finalmente el algoritmo completo de reconstrucción es presentado y la disposición experimental presentada.

B.4.1 Trazado de la linea de respuesta sobre un soporte de mallas múltiples

B.4.1.1 Operación de Proyección

Dado que la actividad dentro de un volumen desconocido está distribuida de manera continua, la misma puede ser representada por una función espacial 3D continua Λ. Esta función puede ser usada para obtener la cantidad de eventos detectados en una Linea de Respuesta (LoR) por medio del modelado de la cantidad de eventos detectados en una LoR utilizando una distribución de Poisson. El parámetro de esta distribución es la actividad a través de la trayectoria de la LoR. Entonces el número de eventos esperados en la trayectoria de cada LoR puede ser calculado como la integración de la trayectoria de la LoR en la estimación actual $\Lambda^{(t)}$, como es presentado en la ecuación B.1, donde s_l es el diferencial de linea sobre la l^{esima} LoR.

$$\hat{x}_l = \int \Lambda^{(t)}(s_l) ds_l. \tag{B.1}$$

Aquí se propone el uso de un modelo de mallas 2D múltiples, como es mostrado en la figura B.2, para describir la distribución $\Lambda^{(t)}$. La actividad es muestreada utilizando un total de N_s cortes axiales donde las mallas son generadas. La actividad a lo largo del eje z de una de los cortes es considerado constante (como en la representación voxelizada), por lo que la actividad dentro de un corte depende solamente de las coordenadas (x, y). En este esquema de representación, el elemento resultante es un prisma triangular (o triángulo *ancho*, con ancho Δz) el cual puede ser tratado como un elemento triangular para la interpolación dentro de un corte.

La actividad dentro de un corte puede ser presentada como la interpolación bi-lineal de los valores de actividad en cada uno de los nodos del elemento actual. Cada malla entonces aplica una función de forma Φ para aproximar el valor de $\Lambda^{(t)}$ en cualquier posición. La forma de la función puede ser escrita como encontrar el valor en las coordenadas de un punto en un plano (un plano descrito por tres puntos coplanares). Las coordenadas desconocidas de una LoR 3D son definidas usando la notación *pendiente-ordenada al origen* $\boldsymbol{r}(a) = \boldsymbol{r}^P.a + \boldsymbol{r}^O$. Los puntos $\boldsymbol{r}^O = [r_x^O, r_y^O, r_z^O]$ y $\boldsymbol{r}^P = [r_x^P, r_y^P, r_z^P]$ son las coordinadas tri-dimensionales de la ordenada al origen y la pendiente, respectivamente, de la LoR. Sin embargo, dentro de una malla 2D se descarta la coordenada del eje z dado que la actividad a lo largo de ese eje es constante dentro de la malla. Entonces la LoR dentro de una malla está definida por $\boldsymbol{r}^O = [r_x^O, r_y^O]$ y $\boldsymbol{r}^P = [r_x^P, r_y^P]$. Cada vértice $\boldsymbol{r}^{v,e}$ (con $v \in [1, 2, 3]$) para cada elemento e de la malla es descrito por su posición y la estimación de su actividad $\boldsymbol{r}^v = [r_x^{v,e}, r_y^{v,e}, \lambda_{v,e}]$. La actividad en la posición del nodo esta dad por la función de actividad volumétrica $\lambda_v^e = \Lambda^{(t)}(r_x^{v,e}, r_y^{v,e}, z_s)$, donde z_s es la posición del corte en el eje z del FoV. Utilizando esta notación se obtiene la función de forma por derivación de la representación *punto y normal* de un plano descrito por tres puntos: $\boldsymbol{r}^{0,e}$, $\boldsymbol{r}^{1,e}$ y $\boldsymbol{r}^{2,e}$. Esta función aproxima $\Lambda^{(t)}$ en cualquier

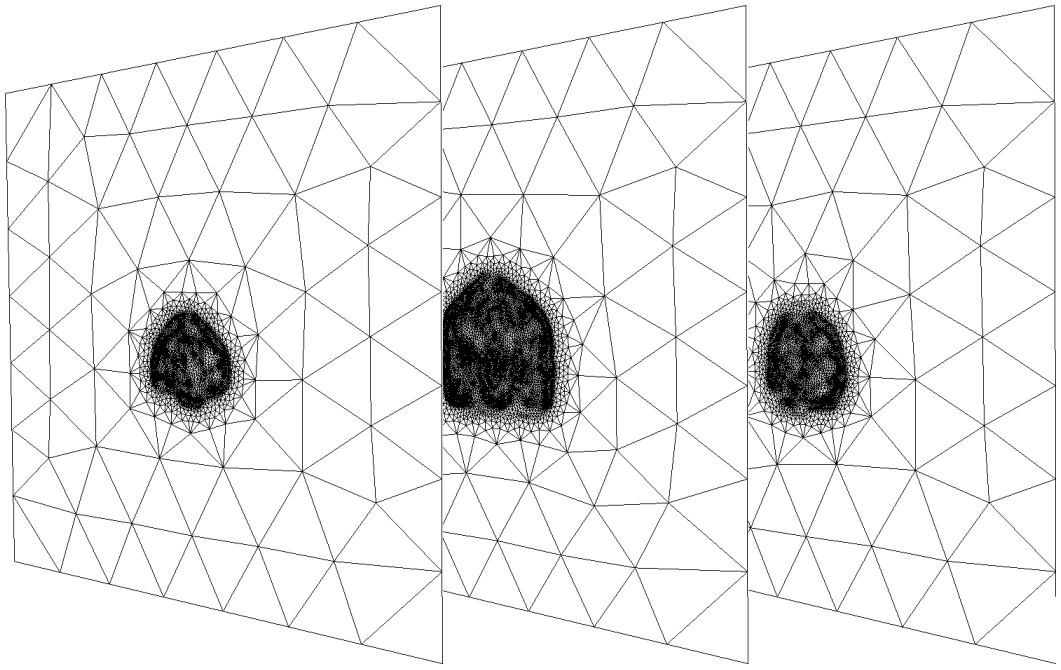


Figure B.2: Series de mallas 2D representado un volumen. Cada malla representa una parte del eje axial.

posición dentro de un elemento arbitrario e y puede escribirse como:

$$\Phi^e(\Lambda^{(t)}, e, \boldsymbol{r}^P, \boldsymbol{r}^O, a) = \begin{cases} \frac{d^e - n_x^e(r_x^P a + r_x^O) - n_y^e(r_y^P a + r_y^O)}{n_\lambda^e}, & \text{inside } e \\ 0, & \text{outside } e \end{cases} \tag{B.2}$$

donde $\bar{n^e} = [n_x^e, n_y^e, n_\lambda^e]$ es el vector normal al plano en la superficie de actividad definida por el elemento e^{esimo} y la distancia $d^e = \bar{n^e} \cdot \boldsymbol{r}^{v,e} = (r_y^{3,e} r_x^{2,e} - r_x^{3,e} r_y^{2,e})\ \lambda_1^e + (r_y^{1,e} r_x^{3,e} - r_x^{1,e} r_y^{3,e})\ \lambda_2^e + (r_y^{1,e} r_x^{2,e} - r_x^{1,e} r_y^{2,e})\ \lambda_3^e$ es parte de la ecuación lineal del plano: $n_x^e\ x + n_y^e\ y + n_\lambda^e\ \lambda - d^e = 0$. Entonces la operación de proyección en la malla se describe como:

$$\hat{x}_l = x_l^\mu x_l^{\mathtt{eff}\cdot} \sum_{e=1}^{E} \int \Phi^e(\Lambda^{(t)}, e, \boldsymbol{r}_l^P, \boldsymbol{r}_l^O, a) da. \tag{B.3}$$

Los elementos x_l^μ y $x_l^{\mathtt{eff}\cdot}$ son, respectivamente, el Factor de Corrección de Atenuación y el Factor de Normalización por eficiencia de Detección de la l^{esima} LoR. Operando en B.2 y reemplazando en la ecuación B.3, la expresión puede ser reducida a una forma multiplicativa:

$$\hat{x}_l = x_l^\mu x_l^{\mathtt{eff}\cdot} \sum_{e=1}^{E} \sum_{v=1}^{2} \frac{\mathcal{K}_{l,e,v} \lambda_v}{n_\lambda^k}, \tag{B.4}$$

con

$$\mathcal{K}_{l,e,1} = [(r_y^{2,e} - r_y^{3,e}).r_x^O + (r_x^{3,e} - r_x^{2,e}).r_y^O + (r_y^{3,e}r_x^{2,e} - r_x^{3,e}r_y^{2,e})](a_{n,e}^{\mathtt{out}} - a_{n,e}^{\mathtt{in}}) + [(r_y^{2,e} - r_y^{3,e}).r_x^P + (r_x^{3,e} - r_x^{2,e}).r_y^P] \left[\frac{(a_{n,e}^{\mathtt{out}})^2 - (a_{n,e}^{\mathtt{in}})^2}{2}\right], \quad \text{(B.5)}$$

$$\mathcal{K}_{n,e,2} = [(r_y^{3,e} - r_y^{1,e}).r_x^O + (r_x^{1,e} - r_x^{3,e}).r_y^O + (r_y^{1,e}r_x^{3,e} - r_x^{1,e}r_y^{3,e})](a_{n,e}^{\mathtt{out}} - a_{n,e}^{\mathtt{in}}) + [(r_y^{3,e} - r_y^{1,e}).r_x^P + (r_x^{1,e} - r_x^{3,e}).r_y^P] \left[\frac{(a_{n,e}^{\mathtt{out}})^2 - (a_{n,e}^{\mathtt{in}})^2}{2}\right], \quad \text{(B.6)}$$

$$\mathcal{K}_{n,e,3} = [(r_y^{1,e} - r_y^{2,e}).r_x^O + (r_x^{2,e} - r_x^{1,e}).r_y^O + (r_y^{1,e}r_x^{2,e} - r_x^{1,e}r_y^{2,e})](a_{n,e}^{\mathtt{out}} - a_{n,e}^{\mathtt{in}}) + [(r_y^{1,e} - r_y^{2,e}).r_x^P + (r_x^{2,e} - r_x^{1,e}).r_y^P] \left[\frac{(a_{n,e}^{\mathtt{out}})^2 - (a_{n,e}^{\mathtt{in}})^2}{2}\right]. \quad \text{(B.7)}$$

Los elementos $a_{n,e}^{\mathtt{in}}$ y $a_{n,e}^{\mathtt{out}}$ de las ecuaciones B.5, B.6 y B.7 son el valor de entrada (*in*put) y salida (*out*put) del escalar a que describe el punto inicial y final de la l^{esima} LoR dentro del e^{esimo} elemento de la malla. Estos valores son obtenidos mediante el trazado de la LoR dentro del elemento e^{esimo}. Los elementos $\mathcal{K}_{l,e,v}$, de la ecuación B.4, presentados en las ecuaciones B.5, B.6 y B.7, son la contribución del punto muestreado a la probabilidad de detectar un evento que se origina en la posición $\boldsymbol{r}(a)$, en la dirección de la l^{esima} LoR. Este valor es derivado de la estimación del valor de un punto en un plano definido por tres puntos. El subíndice v representa los vértices que componen el elemento triangular e de la malla. El total de E elementos esta compuesto por los I nodos en la malla. Para simplicidad la notación en base a elementos es dejada de lado y la ecuación B.4 puede ser reescrita como una sumatoria sobre los I nodos de la malla, donde puede observarse la matriz del sistema de manera directa:

$$\hat{x}_l = x_l^{\mu} x_l^{\mathtt{eff.}} \sum_{i=1}^{I} \mathcal{K}'_{l,i} \lambda_i, \quad \text{(B.8)}$$

con

$$K'_{l,i} = \sum_{e=1,v=i}^{E} \frac{K_{l,e,v}}{n_{\lambda}^{e}}. \quad \text{(B.9)}$$

B.4.1.2 Operación de Retro-Proyección

La operación inversa de la proyección, requerida para completar el algoritmo de Maximización de la Expectativa (EM), es construida como una sumatoria del factor de influencia de las L LoRs en la lista. Esta operación resulta en la actividad λ para cada uno de los nodos definidos en la distribución $\Lambda^{(t)}$. La operación de Retro-Proyección

(BP) es definida como:

$$\lambda_i = \sum_{l=1}^{L} \frac{x_l^{\mu} x_l^{\texttt{eff.}} \mathcal{K}'_{l,i}}{\texttt{SC}_i} x_l. \tag{B.10}$$

La operación de BP incluye una nuevo Factor de Normalización de Muestreo $\texttt{SC}_i$ el cual es necesario para corregir el muestreo no uniforme de la representación por mallas. Este efecto es mostrado en la figura B.3 donde los puntos de muestreo distintos en posiciones $\boldsymbol{r}^A$ y $\boldsymbol{r}^B$, contribuyen a la definición de la actividad en volúmenes distintos, Vol_A y Vol_B respectivamente. La actividad en cada nodos i es usada para interpolar la actividad en cada coordenada dentro de todos los elementos que conforman este volumen. El área combinada de cada uno de estos elementos y la distancia entre los cortes crea un volumen en donde el nodo i tiene influencia. Dado que los elementos en la malla tienen distintos tamaños, entonces el volumen de influencia de cada nodo es distinto. La relación del área de influencia del nodo i al área mayor existente define el valor $\texttt{SC}_i$.

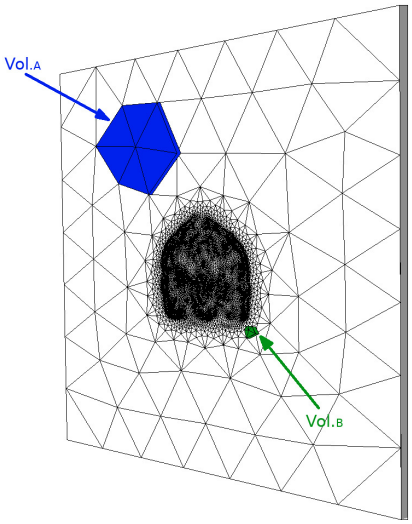


Figure B.3: Volumen de influencia de dos nodos distintos. En la parte superior de la imagen, el volumen alrededor del nodo central "A" (Vol_A) y cerca del centro de la imagen el volumen alrededor del nodo central "B" (Vol_B).

B.4.1.3 Operación en LoRs oblicuas

La actividad total de una LoR 3D cruzando dentro de un prisma triangular depende de: El valor de actividad en cada uno de los seis nodos del elemento, 3 del triangulo delantero y 3 del triangulo trasero ; La coordenada de ingreso (in) ; La coordenada de egreso (out). En la figura B.4 el cruce de una LoR dentro del elemento 3D es representada. La actividad de un punto aleatorio $\boldsymbol{r}(a)$ dentro del elemento 3D puede ser calculado usando interpolación tri-lineal. Esto es logrado mediante el cálculo inicial

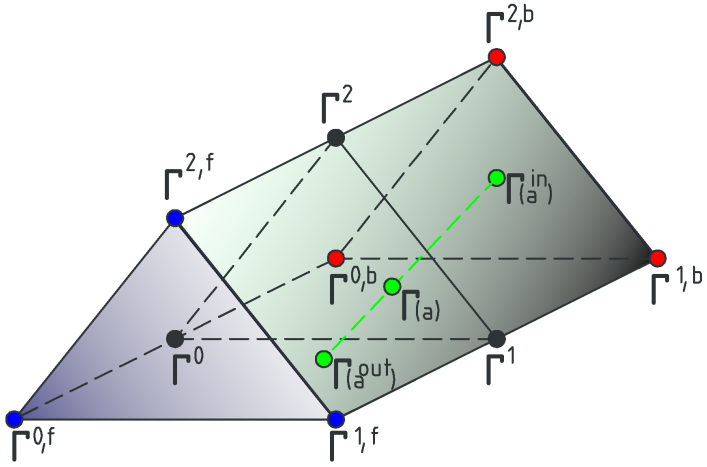


Figure B.4: Prisma triangular representado el elemento 3D de la malla. Los puntos azules ($\boldsymbol{r}^{0,f}$, $\boldsymbol{r}^{1,f}$, $\boldsymbol{r}^{2,f}$) y los puntos rojos ($\boldsymbol{r}^{0,b}$, $\boldsymbol{r}^{1,b}$, $\boldsymbol{r}^{2,b}$) representan los límites frontales y traseros del prisma. La línea punteada verde con límites $\boldsymbol{r}(a^{\texttt{in}})$ y $\boldsymbol{r}(a^{\texttt{out}})$ representa el pasaje de la LoR por el elemento. El punto verde $\boldsymbol{r}(a)$ es un punto coplanar incluido en un triangulo arbitrario dentro del prisma, representado por los puntos negros ($\boldsymbol{r}^0$, $\boldsymbol{r}^1$, $\boldsymbol{r}^2$).

de la actividad en cada uno de los bordes del plano triangular ($\boldsymbol{r}^0$, $\boldsymbol{r}^1$, $\boldsymbol{r}^2$) que contiene el punto a interpolar utilizando interpolación lineal:

$$\lambda_v = (\frac{\lambda_v^b - \lambda_v^f}{\Delta z})(r_z^P a + r_z^O) + \lambda_v^f, \tag{B.11}$$

y luego aplicando la ecuación B.2 usando los valores de los nodos $\boldsymbol{r}^{n,e}$. Sin embargo como se eligió que la actividad a lo largo del eje z de un corte sea constante, la actividad en el triangulo delantero y trasero son la misma ($\lambda_v^f = \lambda_v = \lambda_v^b$). Entonces la interpolación tri-lineal es simplificada a la bi-lineal presentada en la ecuación B.2.

Todas las LoRs que no entran en una única malla 2D son LoRs oblicuas multi-corte. Estas LoRs son segmentadas en cada corte que es cruzado y cada segmento es proyectado o retro-proyectado usando las ecuaciones B.8 y B.10, respectivamente. La segmentación de una LoR es realizada mediante el cómputo del parámetro a de la entrada y la salida para cada corte n_s: $z_s^{\texttt{in}} = z_s - \Delta z/2$ y $z_s^{\texttt{out}} = z_s + \Delta z/2$. Resultando en dos puntos, $\boldsymbol{r}(a_s^{\texttt{in}})$ y $\boldsymbol{r}(a_s^{\texttt{out}})$. Entonces los elementos $K_{l,e,v}$ son calculados en aquellos valores dentro del segmento $\overline{\boldsymbol{r}(a_s^{\texttt{out}}) - \boldsymbol{r}(a_s^{\texttt{in}})}$. Finalmente la LoR es modulada por el largo de la misma dentro del corte.

B.4.2 Maximización de la Verosimilitud

Para optimizar la distribución de actividad propuesta $\Lambda^{(t)}$ a la medición x_l, la verosimilitud logarítmica es maximizada:

$$\ell(X|\Lambda^t) = \sum_{l=1}^{L} -\hat{x}_l + x_l \ln(\hat{x}_l). \tag{B.12}$$

La expresión para el modo lista del algoritmo de MLEM [146] es entonces expresada como:

$$\lambda_i^{(t+1)} = \frac{\lambda_i^{(t)}}{\sum_{l=1}^{L^*} \frac{x_l^{\mu} x_l^{\mathtt{eff}} \cdot \mathcal{K}'_{l,i}}{\mathtt{SC}_i}} \sum_{l=1}^{L} \frac{1.0}{\hat{x}_l} \frac{x_l^{\mu} x_l^{\mathtt{eff}} \cdot \mathcal{K}'_{l,i}}{\mathtt{SC}_i}, \tag{B.13}$$

donde $\sum_{l=1}^{L^*} \frac{AC_i DC_i \mathcal{K}'_{l,i}}{\mathtt{SC}_i}$ es la sumatoria sobre una lista densamente muestreada de la distribución de todas las LoRs medibles. Cada paso de la función B.13 aumenta de manera monotónica la verosimilitud de la distribución $\Lambda^{(t)}$ al conjunto de LoRs detectadas $\mathcal{X}$.

B.4.3 Esquema de remallado

Dado que el tomógrafo solo puede capturar coincidencias perpendiculares o con una inclinación limitada con respecto al eje z, se eligió dividir la imagen usando una serie de mallas 2D a lo largo del eje axial del tomógrafo (el eje z). Cuando se lo compara con una malla 3D, la representación propuesta requiere menos computo, a cambio de mayor error de interpolación en el eje z. Cada una de las mallas en la representación propuesta es calculada como una malla 2D, por lo tanto se simplifica el proceso de remallado. El volumen de un corte es definido por el tamaño de la malla 2D y la separación con la siguiente malla. Dado que la actividad dentro de un corte se asume constante a lo largo del eje z, la representación 3D puede considerarse solo 2D para cada corte. Usando esta representación se pueden posicionar los nodos de manera estratégica en las regiones donde una representación más compacta es necesaria (para bordes agudos o texturas complejas). La habilidad de controlar las posiciones de los nodos posibilita la descripción de la imagen con potencialmente menos elementos que en la representación voxelizada. Una estrategia para posicionar los nodos es usar un proceso de remallado adaptativo [142, 143]. Para producir una malla con un error de interpolación controlado, un proceso de adaptación es requerido. Este proceso mejora la malla usando métricas sobre una malla previa como estimación del error. Dado que la distribución real del radiotrazador es desconocida, la estimación del error se realiza a posteriori, basa en la reconstrucción sobre la malla actual. El error es confinado localmente, en las cercanías del nodo, usando una aproximación simple basada en el desvío local de la superficie de solución [125]. Esta métrica, basada en el Hessiano discreto, resulta en un mapa de tamaños el cual es usado para producir la nueva malla

adaptada. Cada elemento del mapa puede escribirse como:

$$h(i) = \frac{\delta}{\eta(i, \lambda_i)}, \tag{B.14}$$

donde $\eta(i, \lambda_i)$ es el desvío máximo desde los nodos adyacentes al plano tangente en este nodo. El proceso de remallado se describe como:

Algorithm 8: Proceso de remallado

```
while e_s > δ do
    Calcular el mapa de tamaños F^(t)(Λ), basado en la solución del Hessiano
     discreto de Λ.
    Actualizar F_s^(t+1)(Λ) dado el mapa de tamaños.
    Evaluar el error "e" de F_s^(t)(Λ), usando interpolación lineal.
end
```

La operación de remallado basada en un mapa de tamaño asegura que el error de interpolación entre nodos esta confinado por un rango mínimo y máximo ($h_{min} < h(i) < h_{max}$), seleccionando de manera cuidadosa h_{min} es posible evitar crear estructuras artificiales de alta frecuencia (estructural) y artefactos de ruido. En aquellos nodos donde $h = h_{min}$, el error no estará confinado, sin embargo eligiendo el valor de h_{min} lo suficientemente bajo, estas diferencias quedan por debajo de la resolución del tomógrafo.

B.4.4 Iteraciones de Reconstrucción

Utilizando el procedimiento de reconstrucción presentado como el modelo físico en el proceso de remallado una operación de reconstrucción con un error de interpolación confinado es lograda. La magnitud de este error esta directamente relacionada con la precisión de la operación de proyección (ver ecuación B.8). Entonces la malla obtenida es una representación más compacta, que no pierde precisión y que requiere el computo de menos nodos a medida que la malla converge. El proceso de reconstrucción es entonces llevado a cabo de la siguiente manera:

Cada uno de los pasos de reconstrucción en el algoritmo 9 es una serie de S_R pasos de reconstrucción MLEM. El número de pasos a aplicar es arbitrario. No es requerido que la estimación de actividad llegue a un valor óptimo antes de actualizar la malla de soporte. Dado que el proceso de remallado esta gobernado por el limite σ, las posiciones espaciales de los nodos están adaptadas para poder minimizar el error de interpolación. Esto resulta en un cambio controlado en el valor de proyección $\hat{x}_l$ el cual a su vez controla el cambio en la función objetivo (ecuación B.12). Por lo tanto, cada iteración consigue una mejor calidad de representación de la actividad desconocida, con un impacto confinado en la función objetivo.

Algorithm 9: Proceso de reconstrucción

```
Crear N_s mallas iniciales gruesas F_s^(t=0)(Λ^(t=0))
for S_L pasos do
    for S_R pasos do
        Calcular la intensidad en cada F_s^(t)(Λ^(t)), como un volumen (Paso de
         reconstrucción).
    end
    for todas las F en N_s do
        Obtener F_s^(t+1)(Λ^(t+1)) mediante la aplicación de remallado durante
         S_remesh pasos (Ver algoritmo 8).
    end
end
```

B.4.5 Experimentos

La estrategia de reconstrucción propuesta es probada utilizando dos fantomas: una simulación de Monte Carlo del Cerebro de Hoffman 3D y una adquisición del fantoma de calidad de imagen de NEMA [112]. La simulación fue realizada usando el software GATE [107], donde la geometría del AR-PET fue simulada. La adquisición del fantoma NEMA fue hecha utilizando el prototipo preclínico AR-PET. El método de reconstrucción propuesto es comparado con la reconstrucción MLEM en soporte voxelizado presentado en [146]. Todas las reconstrucciones comparten los mismos datos (Lista de Eventos), el mismo modelo matemático del tomógrafo y técnicas de corrección.

Simulación del fantoma de cerebro de Hoffman

El fantoma es ubicado en el centro del campo de visión (FoV). La simulación esta libre de efectos de atenuación y errores de posicionamiento para poder reflejar de una mejor manera la capacidad de compresión de la malla. Durante la simulación un total de 1.74×10^8 coincidencias son recolectadas (incluyendo reales y aleatorias).

Para poder evaluar el comportamiento de la técnica de reconstrucción propuesta bajo distintos niveles de ruido estadístico, la simulación del cerebro 3D de Hoffman fue reconstruida utilizando cuatro niveles distintos de estadística, 1×10^5, 1×10^6, 1×10^7 y 1.7×10^8 LoRs. La reconstrucción en soporte de malla utiliza cortes de 2 mm, un $\delta = 0.01$ y $h_{min} = 1.0$ mm. La malla inicial es una grilla casi Cartesiana con una distancia entre nodos de 4 mm. El proceso es aplicado durante 10 iteraciones con $S_R = 10$ pasos de reconstrucción y $S_{\texttt{remesh}} = 5$ pasos de remallado, resultando en 50 pasos de remallado y 100 iteraciones de reconstrucción. La reconstrucción en soporte voxelizado es hecha con $S_R = 100$ iteraciones y utiliza la misma lista de entrada que el método presentado. La reconstrucción sobre voxeles es filtrada usando un filtro Gausseano con un desvío estándar de $\sigma = 1$ mm, para suavizar cualquier ruido de alta frecuencia del calculo de la reconstrucción. Este filtrado no es necesario para el soporte de malla ya que este lo evita mediante el remuestreado del soporte donde el mismo

no respeta el límite σ. Dos configuraciones voxelizadas fueron probadas, una con un tamaño de voxel de $2 \times 2 \times 2$ mm^3, resultado en $256 \times 256 \times 256 = 16777216$ voxels y otra con un tamaño de voxel de $4 \times 4 \times 4$ mm^3, resultando en $128 \times 128 \times 128 = 2097152$ voxels. Para poder comparar las reconstrucciones, el método presentado es proyectado a un volumen con tamaño de voxel de $2 \times 2 \times 2$ mm^3 usando interpolación bi-lineal en cada corte. La representación voxelizada de baja calidad es linealmente interpolada a un tamaño de voxel de $2 \times 2 \times 2$ mm^3 para ser comparado.

Para comparar las imágenes reconstruidas, la Relación de Señal a Ruido (SNR), el Error Cuadrático Medio (MSE) y el contraste de materia gris ($\mathtt{C}_{\mathtt{grey}}$) fueron utilizadas. La métrica del contraste es calculada como:

$$\mathtt{C}_{\mathtt{grey}} = \frac{\overline{\mathcal{V}_{\mathtt{grey}}} - \overline{\mathcal{V}_{\mathtt{white}}}}{\overline{\mathcal{V}_{\mathtt{white}}}}, \tag{B.15}$$

donde $\overline{\mathcal{V}_{\mathtt{grey}}}$ y $\overline{\mathcal{V}_{\mathtt{white}}}$ son el valor medio de los voxeles identificados como materia gris y blanca respectivamente. La segmentación de los voxels fue realizada usando un umbral en la imagen de verdad. Dado que la imagen ocupa una pequeña fracción del total del FoV, se construyo una cuadro delimitador alrededor del fantoma y las métricas fueron aplicadas solo dentro de este cuadro. El tamaño del cuadro delimitado para el fantoma de Hoffman es de $102 \times 102 \times 102 = 1.06 \times 10^6$ voxels.

Adquisición del fantoma NEMA

El fantoma se construye siguiendo las guías propuestas por NEMA y ubicado en el centro del FoV. La adquisición es estática (una única posición de la camilla) y los detectores están rotando a 2 rpm. La adquisición tiene una duración de 47 minutos y 2.0×10^7 coincidencias son guardadas. La dosis inyectada es de 1.85 `mCi`, las lesiones calientes están llenas con una concentración de 4 veces la del fondo.

Las imágenes de atenuación son adquiridas utilizando un CT helicoidal *Aquilion Lightning, Canon Medical Systems* de 16 filas y 32 cortes. Las imágenes son coregistradas manualmente utilizando una reconstrucción preliminar sin corrección de atenuación.

El fantoma NEMA también fue probado utilizando distintos niveles de estadística, en este caso los conjuntos incluyen 1×10^5, 1×10^6 y 2×10^7 LoRs. Los parámetros de reconstrucción en el soporte de malla fueron los mismos que en la reconstrucción de la simulación del cerebro de Hoffman. La reconstrucción en el soporte de malla es aplicada durante 20 iteraciones con $S_R = 5$ y $S_{\mathtt{remesh}} = 5$, resultando en 100 iteraciones de remallado y 100 iteraciones de reconstrucción. La reconstrucción en voxel fue hecha con los mismos parámetros que en la simulación. Como antes, para comparar las reconstrucciones, el método propuesto es proyectado a un volumen de igual resolución utilizando interpolación bi-lineal en cada corte.

La adquisición del fantoma NEMA fue probada utilizando las métricas descritas en le documento de NEMA 2012 [112].

B.5 Resultados

B.5.1 Simulación del fantoma de Hoffman

La actividad real y la reconstrucción en el soporte de malla y el soporte voxelizado de alta resolución, para el conjunto completo de LoRs, puede verse en la figura B.5, donde se muestran los cortes a lo largo de cada eje.

Las métricas resultantes para cada conjunto de LoRs y tipo de reconstrucción se dan en la tabla B.1. La evolución de las métricas como una función de la estadística, para la última iteración, es mostrada en la figura B.6. Usando estas métricas la convergencia de la reconstrucción puede ser observada. En la figura B.7 las métricas como una función de la iteración, para el conjunto total de estadística, son mostradas.

Table B.1: Métricas calculadas para la simulación del fantoma del cerebro de Hoffman en cada conjunto de estadísticas.

LoRs	Reconstrucción		Métrica		
	Soporte	Resolución	MSE	SNR	C_{grey}
1×10^5	Voxel	4 mm	0.17	4.88	0.33
		2 mm	0.21	3.93	0.35
	Malla	-	0.28	2.66	0.29
1×10^6	Voxel	4 mm	0.11	6.90	0.40
		2 mm	0.11	6.86	0.44
	Malla	-	0.13	6.15	0.45
1×10^7	Voxel	4 mm	0.099	7.21	0.42
		2 mm	0.094	7.46	0.46
	Malla	-	0.093	7.51	0.49
1.7×10^8	Voxel	4 mm	0.099	7.22	0.41
		2 mm	0.092	7.51	0.46
	Malla	-	0.080	8.18	0.57

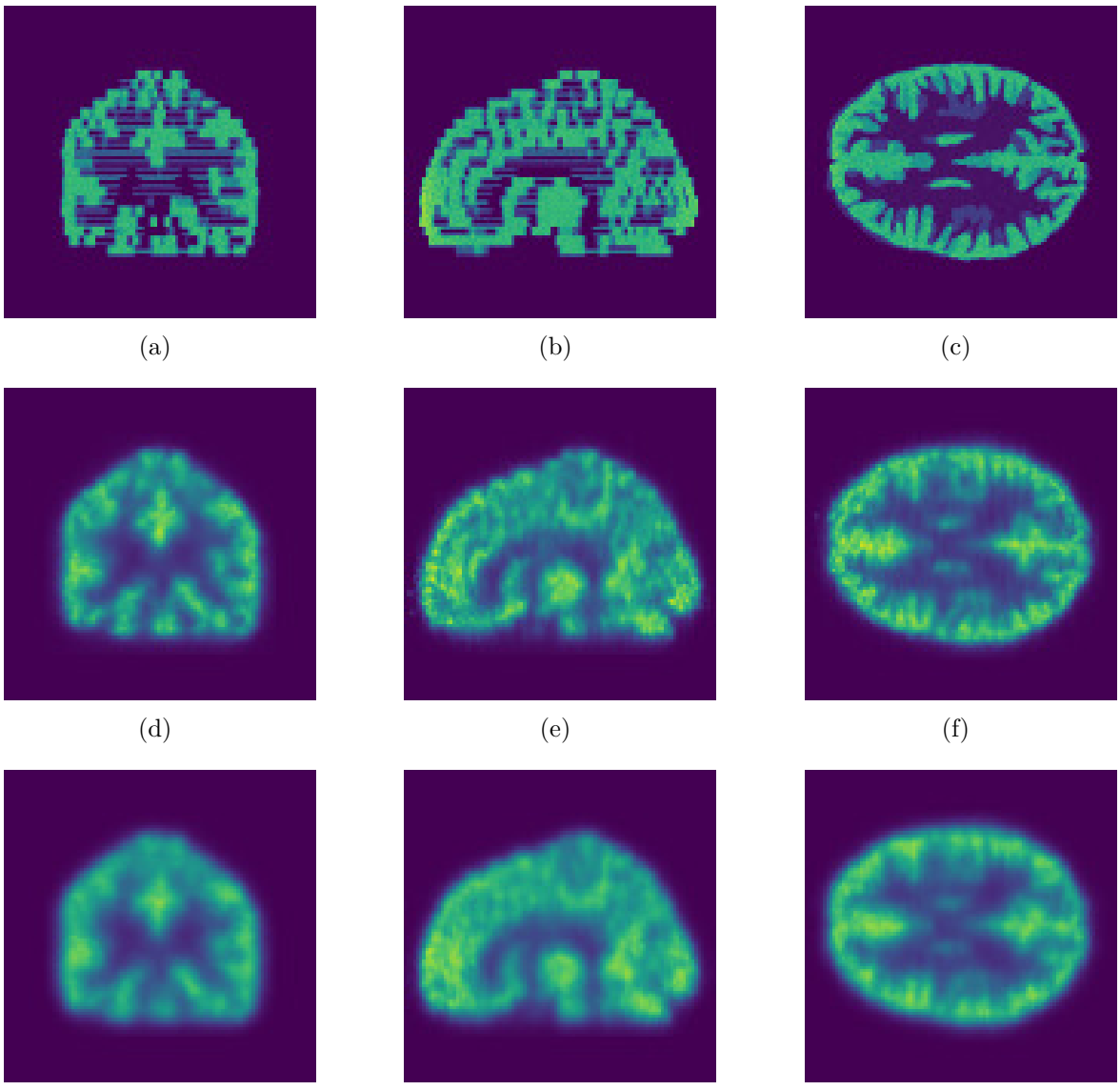


Figure B.5: Comparación entre la distribución real de la actividad (a, b, c), la interpolación de la reconstrucción sobre mallas en un soporte pixelado de 2 mm (d, e, f) y la reconstrucción en soporte voxelizado de 2 mm (g, h, i) de la simulación del cerebro de Hoffman. Las imágenes que se muestran fueron reconstruidas con el conjunto total de LoRs.

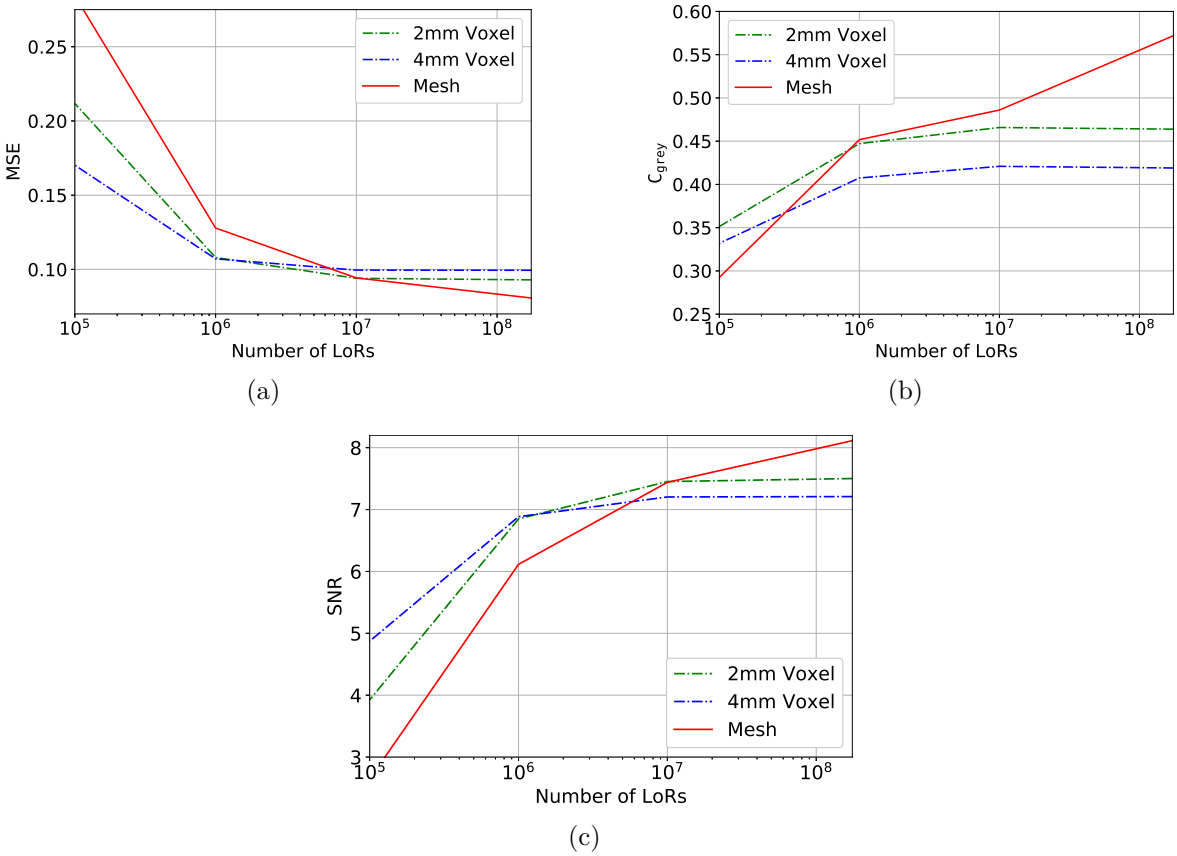


Figure B.6: Evolución del MSE (a), $\mathtt{C_{grey}}$ (b) y SNR (c) como una función de la cantidad de estadística recolectada para la simulación del cerebro de Hoffman. La linea punteada verde representa la reconstrucción en el soporte voxelizado de tamaño $2 \times 2 \times 2$ mm^3, la linea punteada azul representa la reconstrucción en el soporte voxelizado de $4 \times 4 \times 4$ mm^3 y la línea roja sólida representa la reconstrucción sobre la malla.

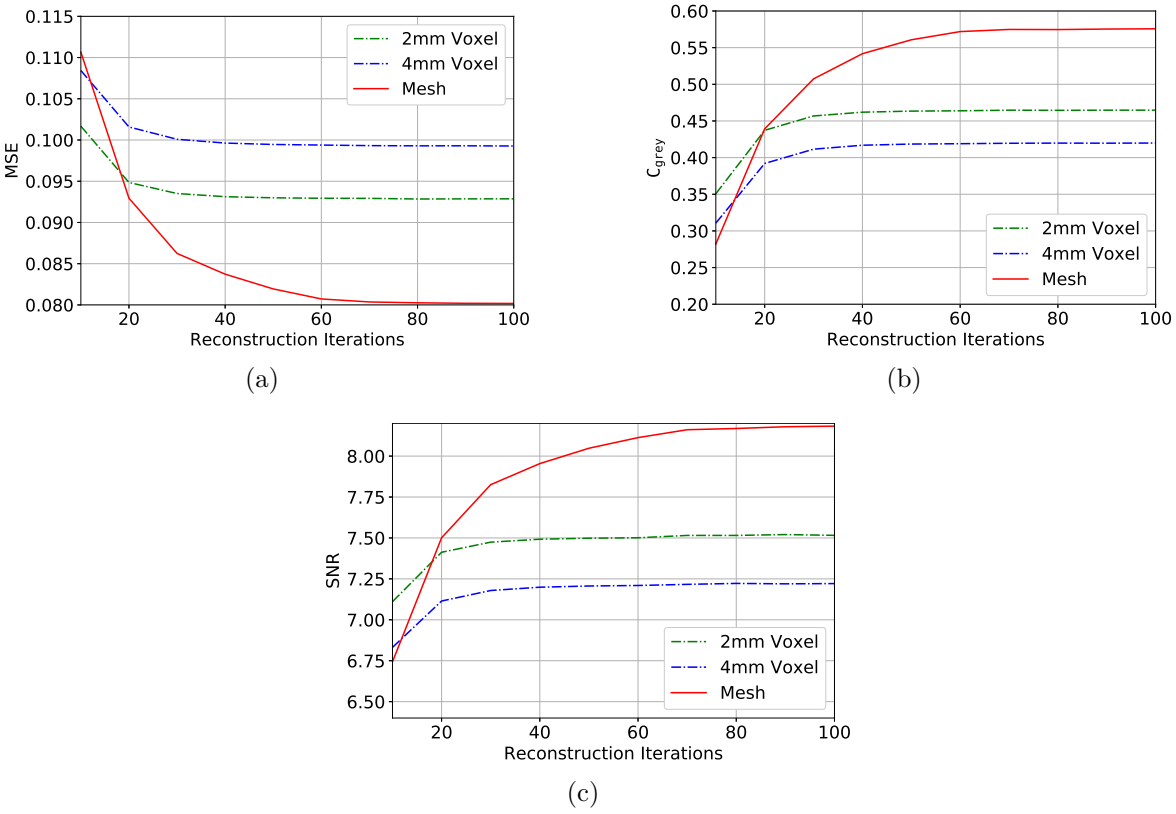


Figure B.7: Evolución del MSE (a), $\mathtt{C_{grey}}$ (b) y SNR (c) como una función de la iteración para el conjunto de estadística mayor de la simulación del fantoma del cerebro de Hoffman. La linea punteada verde representa la reconstrucción en el soporte voxelizado de tamaño $2 \times 2 \times 2$ mm^3, la linea punteada azul representa la reconstrucción en el soporte voxelizado de $4 \times 4 \times 4$ mm^3 y la línea roja sólida representa la reconstrucción sobre la malla.

B.5.2 Adquisición del fantoma NEMA

La forma del fantoma y la reconstrucción sobre la malla y la reconstrucción sobre el soporte voxelizado de alta resolución pueden verse en la figura B.8, donde los cortes a lo largo de cada eje se muestran.

La precisión de las correcciones de atenuación y dispersión ($C_{\texttt{lung}}$, cuanto más bajo mejor), contraste en esferas frías ($Q_{\texttt{cold}}$, cuanto más alto mejor), contraste en esferas calientes ($Q_{\texttt{hot}}$, cuanto más alto mejor) y variación media del fondo ($N_{\texttt{mean}}$, menores valores son mejores) son resumidas, para cada conjunto de estadísticas y resolución, en la tabla B.2. En la figura B.9 las métricas como una función de las iteraciones, para el conjunto de estadísticas mayor, son presentadas.

Table B.2: Métricas para la adquisición del fantoma NEMA 2012.

LoRs	Reconstrucción		Métrica			
	Soporte	Resolución	$C_{\texttt{lung}}$	$Q_{\texttt{cold}}$	$Q_{\texttt{hot}}$	$N_{\texttt{mean}}$
1×10^5	Voxel	4 mm	0.52	0.17 ± 0.06	0.53 ± 0.14	0.61 ± 0.17
		2 mm	0.43	0.15 ± 0.07	0.31 ± 0.05	0.87 ± 0.21
	Malla	-	0.52	0.23 ± 0.32	0.21 ± 0.29	0.60 ± 0.29
1×10^6	Voxel	4 mm	0.39	0.24 ± 0.09	0.38 ± 0.07	0.23 ± 0.07
		2 mm	0.36	0.23 ± 0.09	0.33 ± 0.06	0.29 ± 0.08
	Malla	-	0.42	0.18 ± 0.06	0.18 ± 0.06	0.26 ± 0.11
2×10^7	Voxel	4 mm	0.30	0.26 ± 0.00	0.41 ± 0.11	0.065 ± 0.025
		2 mm	0.29	0.26 ± 0.01	0.42 ± 0.11	0.082 ± 0.025
	Malla	-	0.37	0.33 ± 0.14	0.33 ± 0.14	0.043 ± 0.023

B.5.3 Convergencia del soporte mallado

El proceso de reconstrucción finaliza cuando la reconstrucción y el proceso de mallado convergen. El proceso de remallado produce una configuración de nodos espacial que aseguran que el error de interpolación esta por debajo de un umbral δ. La convergencia de la representación de la malla se puede observar en términos de la cantidad de nodos utilizados para representar el volumen. En la figura B.10 la evolución del número total de nodos y de los nodos en un corte central, para cada fantoma, es presentado. El corte seleccionado para cada reconstrucción es presentado en la figura B.11.

B.6 Discusión

El método presentado logra una calidad de reconstrucción superior en el fantoma del cerebro de Hoffman, mejorando el MSE de 0.092 a 0.080, el SNR de 7.51 db a 8.18 db y el contraste de materia gris de 0.46 a 0.57 cuando se compara con la reconstrucción voxelizada. Para la reconstrucción del fantoma NEMA, la malla tuvo un éxito parcial, mejorando el contraste en las esferas frías de 0.26 a 0.33 y la variabilidad del fondo de

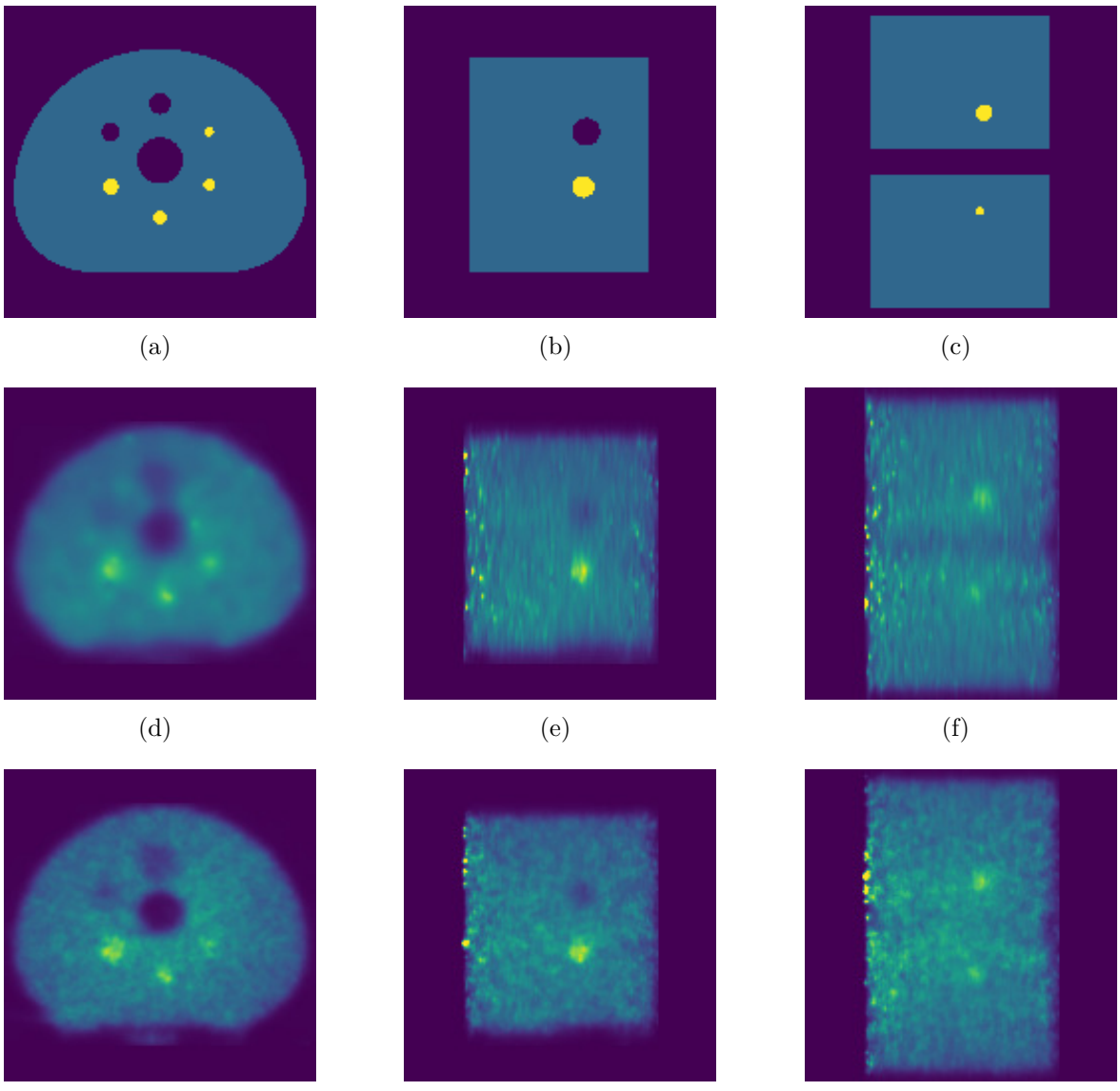


Figure B.8: Comparación entre la distribución esperada de la actividad (a, b, c), la interpolación de la reconstrucción sobre mallas en un soporte pixelado de 2 mm (d, e, f) y la reconstrucción en soporte voxelizado de 2 mm (g, h, i) de la adquisición del fantoma NEMA. Las imágenes que se muestran fueron reconstruidas con el conjunto total de LoRs.

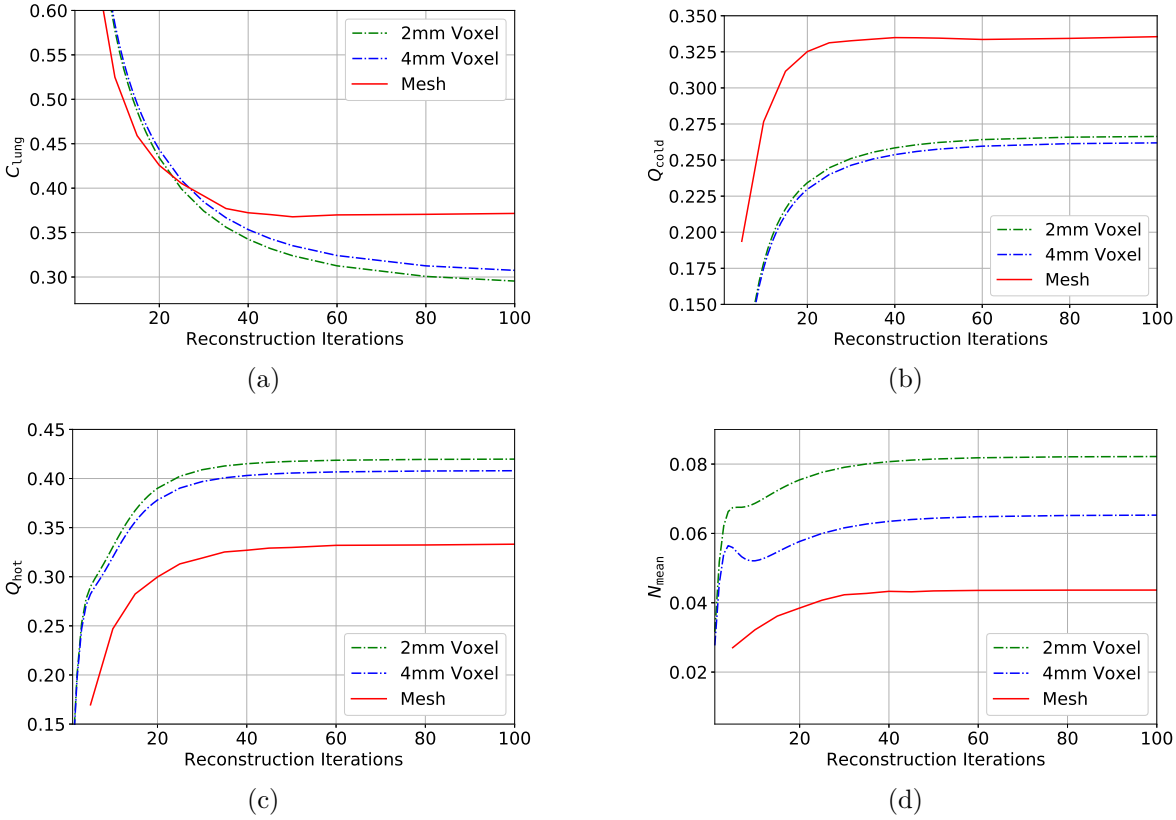
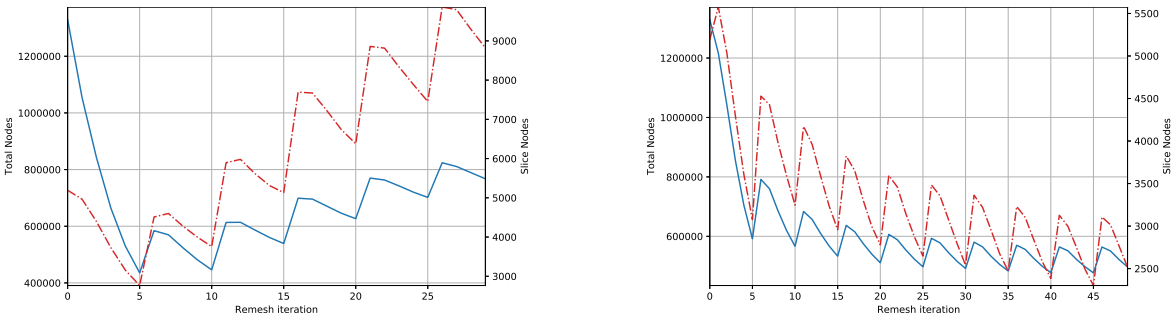


Figure B.9: Evolución de la corrección de atenuación y dispersión $C_{\texttt{lung}}$ (a),contraste en esferas frías $Q_{\texttt{cold}}$ (b), contraste en esferas calientes $Q_{\texttt{hot}}$ (c) y variabilidad media del fondo $N_{\texttt{mean}}$ (d) como una función de la iteración para el conjunto de estadística mayor de la adquisición del fantoma NEMA. La linea punteada verde representa la reconstrucción en el soporte voxelizado de tamaño $2 \times 2 \times 2$ mm^3, la linea punteada azul representa la reconstrucción en el soporte voxelizado de $4 \times 4 \times 4$ mm^3 y la línea roja sólida representa la reconstrucción sobre la malla.



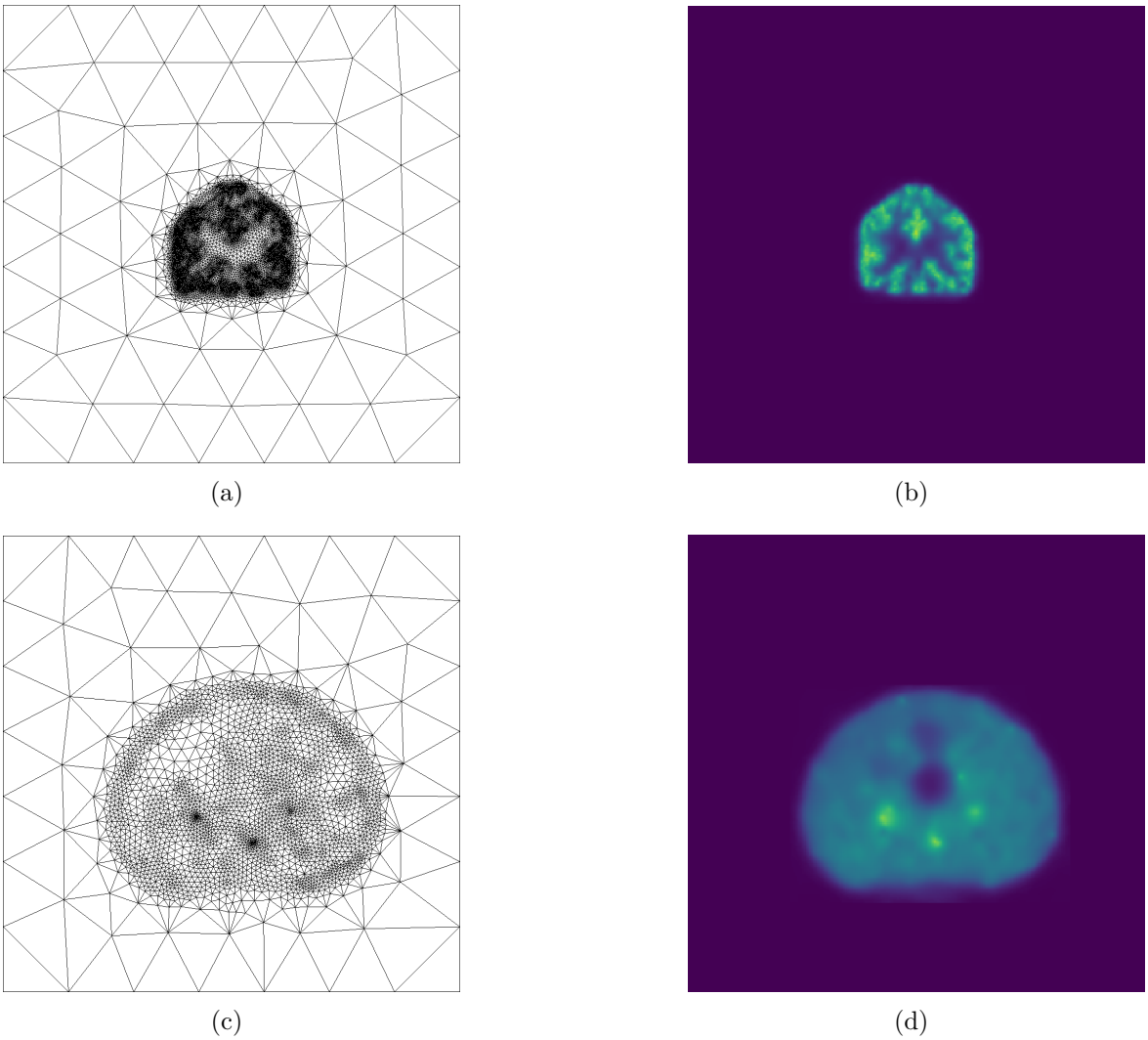
(a) (b)

Figure B.10: Evolución del número de nodos usados para muestrear la simulación del cerebro de Hoffman (a) y la adquisición del fontoma NEMA (b). La linea sólida representa el número total de nodos (escala de la izquierda) y la linea punteada la evolución de los nodos en un corte central del fantoma (escala de la derecha). El eje x representa las iteraciones en el proceso de remallado, cada cinco iteraciones un proceso de reconstrucción fue aplicado.

0.082 a 0.043 pero empeorando la corrección de atenuación de 0.29 a 0.37 y el contraste de las esferas calientes de 0.42 a 0.33.

Se puede apreciar que las métricas sobre la simulación del fantoma del cerebro de Hoffman (tabla B.1) para el soporte de malla y para el soporte voxelizado de alta resolución son muy similares para los conjuntos de estadísticas de 10×10^6 y 10×10^7 LoRs. Cuando el número de LoRs supera las 1×10^7, el soporte de malla es superior al voxelizado. Sin embargo la calidad de la reconstrucción sobre la malla decae más rápido con el número de LoRs que la reconstrucción sobre el soporte voxelizado. Este efecto se ve en la figura B.6, donde el MSE y el SNR de última iteración de reconstrucción en el soporte de malla son inferiores al del soporte voxelizado para estadísticas inferiores a 1×10^7. Vale la pena notar que esto también sucede cuando se comparan las reconstrucciones voxelizadas, el soporte de 4 mm logra una calidad similar o superior al soporte de 2 mm para conjuntos de estadísticas de 1×10^6 o menores. Lo mismo se observa en el fantoma NEMA en la tabla B.2 donde la reconstrucción sobre mallas para 1×10^6 LoRs alcanza métricas inferiores. En el caso del fantoma NEMA la malla presente un mejor contraste de las esferas frías y la variabilidad del fondo, sin embargo este logra una menor aptitud en la corrección de atenuación y en el contraste de esferas calientes. Este comportamiento puede estar relacionado con las fallas en la representación de la malla, que es más susceptible a datas más escasos o ruidosos.

La aplicación del algoritmo de MLEM sobre mallas puede mejorar la calidad en conjuntos con gran estadística y reducir la cantidad de parámetros necesarios para representar la distribución de la actividad. El cerebro de Hoffman requiere 7.67×10^5 nodos para ser representado y el fantoma NEMA 5.22×10^5 nodos, la reconstrucción voxelizada de alta resolución requiere 1.7×10^7 voxels y la de baja resolución 2.1×10^6. Esto es más de un orden de magnitud en el número de elementos para representar la misma información. Para reducir el efecto de las grandes zonas vacías en el FoV, uno



(a) (b)

(c) (d)

Figure B.11: Representación de la malla (a, c) y la correspondiente distribución de la actividad (b, d) de un corte detallado de las reconstrucciones del cerebro de fantoma de Hoffman (a, b) y el fantoma NEMA (c, d).

puede analizar un corte detallado de la reconstrucción. Dicho corte requiere 8.82×10^3 nodos para el cerebro de Hoffman y 2.55×10^3 nodos para el fantoma NEMA. El mismo corte para el soporte voxelizado requiere $256 \times 256 = 6.55 \times 10^4$ y $128 \times 128 = 1.64 \times 10^4$ voxels para la representación de alta y baja resolución respectivamente. Incluso en este caso la representación por mallas usa como máximo la mitad de los parámetros en el peor de los casos. Esta mejora puede ser utilizada para implementar algoritmos de mayor complejidad y sistemas de PET dinámicos que se benefician de una matriz del sistema reducida.

Como en estudios previos [128–132] la implementación de un soporte por medio de mallas resultó en una calidad de reconstrucción similar o superior y en una reducción de la dimencionalidad del problema. Estos estudios también reportan una reducción del ruido de fondo cuando se reconstruyen adquisiciones reales [129, 131], esto fue también observado en nuestros experimentos, con la reducción de la variación media del fondo en el fantoma NEMA. Además del nuevo enfoque a la reconstrucción de actividad sobre mallas, se exploraron los efectos de los niveles de actividad en la reconstrucción. Esto estaba sin explorar en la literatura y mostramos que afecta la reconstrucción sobre el soporte mallado.

Para evitar problemas de representación de la imagen de atenuación debido a que el soporte es adaptado a la actividad, se eligió utilizar una malla secundaria para representar el mapa de atenuación. Mientras este enfoque requiere el trazado de la LoR sobre un nuevo dominio, la atenuación puede ser precalculada, ya que la misma no varía. La optimización de un único soporte para la atenuación y la actividad, para ser utilizado en algoritmos como Maximización de la Verosimilitud de la Actividad y la Atenuación, se deja como trabajo futuro.

B.7 Discusiones y Comentarios

En este capitulo se presentaron las aplicaciones de mallado para la reconstrucción de imágenes de PET. Se presentan las bases de la reconstrucción de imágenes PET sobre mallas con un método de reconstrucción novedoso. El método muestra que una serie de mallas 2D pueden representar distribuciones 3D complejas. Dicha representación no está limitada a LoRs 2D y la misma es capás de representar todas las proyecciones 3D de un topógrafo convencional (estando solamente limitado por las LoRs totalmente transversales, las cuales no son detectables en los tomógrafos actuales). Aun mas, el soporte de malla se adapta dinámicamente a la geometría del radiotrazador, comenzando desde una malla gruesa y adaptándose a medida que los detalles de la imagen surgen. La representación resultantes es más compacta y requiere potencialmente menos cómputo que un FoV voxelizado para representar los mismos datos. Esta representación significa que menos parámetros son requeridos y hace que las proyecciones de las LoRs sean menos costosas en las zonas donde no se requiere detalle.

Aquí concluyen las técnicas de mallado incluidas en esta tesis. Se mostró que es posible expresar el problema de reconstrucción de imágenes PET usando menos parámetros sin perder calidad de imagen. La representación sobre mallas, acoplado con la posibilidad de aplicar el algoritmo de Simulación de Dispersión Simple, es un paso hacia la reconstrucción mutua de la actividad y el mapa de atenuación con costos computacionales reducidos. El último capitulo de esta tesis se enfoca en reducir aún más la complejidad de la reconstrucción mutua de actividad y atenuación por medio de la generación automática de una imagen de atenuación a priori.

C Generación de mapas de atenuación utilizando Redes Neuronales Profundas

C.1 Introducción

El uso de métodos de aprendizaje profundo en la segmentación, clasificación y generación de imágenes es una tendencia fuerte en distintos campos. Con la aparición de la AlexNet [147] en 2012, muchos problemas comenzaron a ser repensados con alguna influencia de las Redes Neuronales Profundas (DNN). Esto se refleja en el incremento exponencial de publicaciones, a lo largo de una amplia variedad de campos, que incluyen palabras clave como *aprendizaje automático* o *red neuronal*. El campo de las imágenes médicas ha aumentado el uso de estas nuevas herramientas. En este campo la tendencia de construir descriptores a la medida para problemas específicos cambió al uso de representaciones aprendidas, haciendo uso del poder de los modelos profundos [148]. La Red Neuronal Convolucional (CNN) comenzó a ser la técnica predominante en el campo de las imágenes médicas en 2015, aplicando una variedad de modelos para diferentes problemas [149]. Recientemente la generación de imágenes médicas comenzó a ser una parte notable de la tendencia de ML. En 2018 los trabajo de sintetización de imágenes médicas llegaron al 49% de las publicaciones en el campo [150]. Los algoritmos más modernos de sintetización de imágenes médicas están basados en el entrenamiento adversario de topologías neuronales profundas, una técnica presentada en 2014 [151]. Estos algoritmos generativos han alcanzado resultados impresionantes en campos donde existen grandes bases de datos, como la generación de rostros humanos [152]. Aunque los datos no son tan abundantes en el campo de las imágenes médicas, las representaciones posibles en el campo de las imágenes médicas son también reducidas cuando se compraron con otros dominios de imágenes como las fotografías de rostros. Entonces, explotando dicha regularidad en las imágenes médicas, los modelos adversarios pueden ser aplicados a distintas modalidades de imágenes [153].

Este capitulo toma la temática de las DNN en las imágenes médicas. Se enfoca en la generación de Mapas de Atenuación (AMs) para ser utilizados en la corrección de imágenes PET. Este es un paso fundamental hacia una tomografía PET sin imágenes de transmisión. Las imágenes generadas con estos métodos están destinadas a ser usada como una estimación gruesa del AM para corregir directamente la imagen PET o servir como una imagen a priori específica de cada paciente para ser utilizada en la reconstrucción mutua de la actividad y el AM. Aún más se mostrará que el esquema

presentado funciona en datos de diferentes pacientes, lesiones y tomógrafos.

C.2 Sintetización del Mapa de Atenuación para tomografía por Emisión de Positrones con $[^{18}]$F FDG usando Redes Generativas Adversarias 3D

La correcta estimación de la corrección de los mapas de atenuación para imágenes PET es fundamental para su correcta reconstrucción, pero una medición directa de esta significa radiación ionizante adicional para el paciente. Otro enfoque para obtener esta información es utilizar métodos de análisis de imágenes. Estos métodos crean una estructura de atenuación desde otra modalidad de imagen, como una Resonancia Magnética (MRI) o los la imagen PET Sin Corrección de Atenuación (NAC-PET). Esta traducción de modalidad de imagen es especialmente difícil en imágenes NAC-PET de cuerpo completo, dado que la información que presenta es incompleta. En este escenario, donde la traducción de la imagen también necesita completar vacíos de información, las Redes Generativas Adversarias (GANs) son especialmente poderosas.

La aplicación de GANs en tareas de traducción de modalidad de imágenes ha sido aplicada con éxito en muchos dominios de las imágenes médicas, incluyendo la sintetización de los mapas de atenuación de imágenes PET. Sin embargo la mayoria de los métodos analizan la generación de mapas de atenuación desde MRI hacia CT usando CNNs [49] y GANs con datos apareados [153] y sin aparear [184], requiriendo una imagen coregistrada de MRI que contenga la información que no está presente en la imagen de NAC-PET. La traducción de PET (y NAC-PET) a CT continúa siendo uno de los dominios menos explorados, especialmente en tomografías de cuerpo completo [185]. Los estudios en este dominio en particular se enfocan en traducción PET-CT en imágenes corregida de del cráneo. Liu [13] propone el uso de una arquitectura tipo U-Net 2D para traducir tomografías del cráneo de NAC-PET a CT, mostrando resultados prometedores para las regiones craneales. Armanious [153] propone una aplicación GAN generalizada compuesta de generadores tipo U-Net 2D en forma de cascada y un discriminador que evalúa la pérdida perceptual y el estilo de la imagen generada. Ellos muestran la capacidad de la topología para traducir estudios PET en CT, usando solo cortes axiales y, nuevamente, solo para regiones craneales. Ninguno de los métodos provee información en su capacidad de traducción de imágenes de cuerpo completo que es un problema más difícil de resolver, dado que existen más modos posibles en las estructuras de atenuación. La traducción de imágenes de cuerpo completo fue estudiado por Dong [14] usando una GAN entrenada con consistencia cíclica y por Armanious [15] usando una GAN 2D basada en una U-Net en forma de cascada, sin embargo ellos entrenaron y probaron sus modelos solo en tomógrafos PET del estado del arte con capacidad de medición de Tiempo de Vuelo (ToF). La reconstrucción del AM también fue estudiada utilizando reconstrucción de Máxima Verosimilitud de la Actividad y el Mapa de Atenuación (MLAA) [51], mostrando promesa en tomógrafos PET con capacidad de medición de ToF solo cuando es combinado con redes neuronales [186]. Sin embargo sin utilizar un post-procesamiento la salida del MLAA

es ruidosa, con o sin información de ToF [187]. Resolver el problema de generar un mapa de atenuación directamente de una imagen puede permitir a los tomógrafos sin capacidad de ToF el uso de mapas de atenuación sintéticos.

Aquí se presenta una topología GAN 3D con una perdida mixta para generar sinogramas de atenuación de cuerpo completo por medio de imágenes artificiales de CT desde imágenes NAC-PET. Dado que la dimensionalidad de los volúmenes 3D es comparable con imágenes 2D de alta resolución. Por lo tanto se aplica un entrenamiento en dos partes, comenzando con un entrenamiento supervisado con etiquetas y luego se agrega un bloque de pérdida adversaria para mejorar la resolución de la imagen. El modelo es entrenado con conjuntos de datos públicos disponibles en el Archivo de Imágenes de Cáncer [188], el conjunto de datos contiene series de imágenes coregistradas de tipo CT, PET y NAC-PET de cuerpo completo de Carcinomas Espinocelulares de Cabeza y Cuello (HNSCC) [189], adquiridos con el tomógrafo Discovery ST/STE/RX General Electric. Se utilizan 8 conjuntos de datos distintos para para el proceso de prueba, conteniendo 5 tipos de carcinomas y múltiples modelos de tomógrafos: Discovery ST/STE/RX/LS/IQ/610/690, de General Electric y Biograph de Siemens. Solo el Discovery 690 posee capacidad de medir ToF.

C.3 Materiales, métodos y teoría

C.3.1 Descripción de la Topología

Dos arquitecturas son probadas en este trabajo. Primero una 3D U-Net de base, entrenada de una manera totalmente supervisada. Segundo, una GAN cuyo generador esta compuesto por la 3D U-Net de base con capas adicionales entrenadas de manera adversaria contra un crítico (o discriminador) convolucional. Para reducir la inestabilidad del entrenamiento adversario, el gradiente adversario no fluye hacia la 3D U-Net de base.

Generador:

La representación del modelo puede ser vista en la figura C.1. La sección inicial del generador es una topología tipo 3D U-Net [165] (el bloque *3D U-Net*). La red U-Net posee 5 niveles de resolución, cada uno de ellos compuesto de dos capas convolucionales con filtros de tamaño $3 \times 3 \times 3$ y Unidades Rectificadoras (ReLU) como activación. Cada nivel de resolución posee una conexión de salto entre el camino de reducción de resolución y el camino de aumento de resolución. En lugar de utilizar remuestreo convolucional los cambios de resolución son realizados con remuestreo tri-linear. Luego de cada capa de convolución se aplica una normalización de voxel a lo largo de los mapas de características, dividiendo cada voxel por $\sqrt{\frac{1}{p_c}\sum_{i=1}^{p_c}(\mathtt{v}_i)^2 + e}$, donde p_c es el número de canales en el mapa de características, $\mathtt{v}_i$ es el valor del i^{esimo} voxel y $e = 1.0 \times 10^{-8}$. También se aplica en cada capa convolucional un factor de de escalado al núcleo basado en el escalado de inicialización de pesos de He [207]. Luego del bloque de 3D U-Net la red se divide en dos ramas dentro del bloque *Auxiliary*

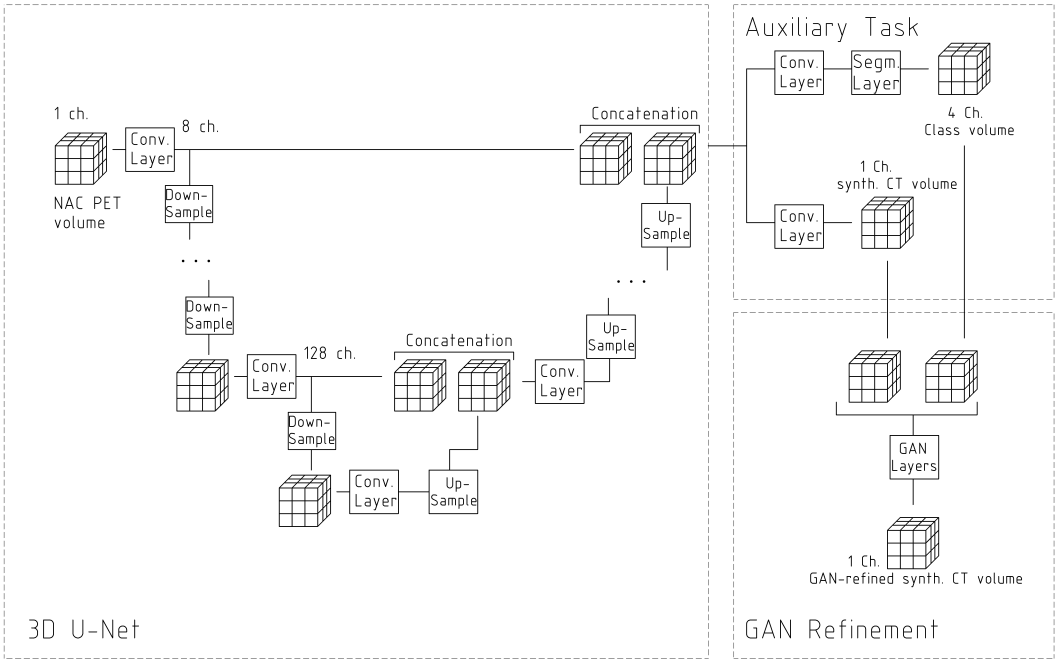


Figure C.1: Topología del generador, basada en una 3D U-Net. La rama superior en el bloque *tarea auxiliar (auxiliary task)* es utilizado para la segmentación de etiquetas y la rama inferior para la generación del CT artificial. La salida del bloque *refinamiento por GAN (GAN Refinement)* es el CT sintético refinado por las capas GAN y construido de las salidas del bloque *tarea auxiliar (auxiliary task)*.

Task. La primer rama es utilizada para segmentar y esta compuesta de tres capas convolucionales y termina en una capa de función exponencial normalizada (softmax). La segunda rama es responsable de generar un CT sintético (sCT), esta está compuesta de una capa convolucional con una activación de tangente hiperbólica. Finalmente, las salidas de la 3D U-Net son fusionadas y procesadas por las capas de GAN en el bloque *GAN Refinement*. Este último bloque con 8 filtros cada capa, es usado durante el entrenamiento adversario. Todas las operaciones de convolución usan un filtro de tamaño $3 \times 3 \times 3$ excepto la capa de salida, la cual tiene un filtro de tamaño $1 \times 1 \times 1$.

Crítico:

La red de crítico o discriminador es una red convolucional normal con activación tipo ReLU en todas sus capas, solamente la última capa no posee activación. La entrada de esta red es un volumen de dos canales: la imagen de NAC-PET y la imagen de CT real o el sCT. La salida de la red es un valor proporcional a la calidad de la imagen generada. La red esta compuesta por 4 niveles de resolución con dos capas convolucionales por nivel. Cada filtro de convolución tiene un tamaño de $3 \times 3 \times 3$ y activación tipo ReLU. No se aplica normalización por tanda o por píxel. Las últimas dos capas del crítico son una operación de aplanado seguido de una única capa densa

con salida lineal.

C.3.2 Esquema de entrenamiento

El entrenamiento de la red esta dividido en dos etapas: primero el generador es entrenado de manera supervisada usando una función objetivo compuesta. La rama de segmentación de red aplica una perdida tipo DICE 3D como se muestra en la ecuación C.1

$$\mathcal{L}_D = \frac{1}{p_c}\sum_{i=1}^{p_c}\frac{2\sum_{j=1}^{V}\hat{\mathtt{v}}_{j,i}^{\mathtt{TL}}\mathtt{v}_{j,i}^{\mathtt{TL}}}{\sum_{j=1}^{V}(\hat{\mathtt{v}}_{j,i}^{\mathtt{TL}})^2+\sum_{j}^{V}(\mathtt{v}_{j,i}^{\mathtt{TL}})^2}, \tag{C.1}$$

donde p_c es el número de las clases objetivo, V el número voxels en el volumen, $\mathtt{v}_{j,i}^{\mathtt{TL}}$ son los valores de los voxels del objetivo y $\hat{\mathtt{v}}_{j,i}^{\mathtt{TL}}$ los valores de la salida de la red. El DICE 3D varia entre 0 y 1. El mismo produce un máximo cuando todos los voxels objetivo ($\mathtt{v}_{j,i}^{\mathtt{TL}}$) tienen el mismo valor que la salida de la red ($\hat{\mathtt{v}}_{j,i}^{\mathtt{TL}}$). Dado que la salida de la red se le aplicó la operación de softmax, el denominador de la ecuación C.1 siempre es mas grande que el numerador excepto cuando $\mathtt{v}_{j,i}^{\mathtt{TL}}$ y $\hat{\mathtt{v}}_{j,i}^{\mathtt{TL}}$ son idénticas. En el caso de un problema multiclase ($p_c > 1$) el valor final es dividido por el número de clases.

La rama de sintetización de CT, hasta las capas de GAN, es entrenada usando la distancia euclídea entre el sCT y la imagen CT objetivo, $\mathcal{L}_T = ||\boldsymbol{V}^{\mathtt{sCT}}, \boldsymbol{V}^{\mathtt{CT}}||^2$, donde $\boldsymbol{V}^{\mathtt{sCT}}$ es el CT sintético y $\boldsymbol{V}^{\mathtt{CT}}$ el CT real. La función de pérdida para el entrenamiento supervisado se muestra en la ecuación C.2.

$$\mathcal{L}^{\mathtt{U-Net}} = (1-\mathcal{L}_D) + k_e\mathcal{L}_T, \tag{C.2}$$

donde k_e es una constante de acople. Luego del entrenamiento inicial, comienza el entrenamiento adversario. El entrenamiento adversario usa la estrategia de la Wassersterin-GAN (W-GAN) [177], resultando en una perdida del generador como la que se muestra en la ecuación C.3,

$$\mathcal{L}_W = -\mathtt{Crit}_W(\mathtt{Gen}_W(\boldsymbol{V}^{\mathtt{NPT}}), \boldsymbol{V}^{\mathtt{NPT}}), \tag{C.3}$$

donde $\mathtt{Crit}_W()$ es la función de la red crítico, $\mathtt{Gen}_W()$ es la función de la red generadora y $\boldsymbol{V}^{\mathtt{NPT}}$ es la imagen NAC-PET de entrada. Durante el entrenamiento adversario las capas de GAN se activan y son entrenadas utilizando la pérdida de W-GAN. El gradiente de la GAN no fluye hacia las capas de la 3D U-Net. El crítico es entrenado usando un par de de imágenes de NAC-PET y CT acopladas, real o generada. El mismo es entrenado usando la pérdida de wasserstein mostrada en la ecuación C.4,

$$J_{crit.}^{GAN} = \frac{1}{N_m^{true}}\sum_{n=1}^{N_m^{true}}\mathtt{Crit}_W(\boldsymbol{V}_n^{\mathtt{NPT}}, \boldsymbol{V}_n^{\mathtt{CT}}) - \frac{1}{N_m^{fake}}\sum_{n=1}^{N_m^{fake}}\mathtt{Crit}_W(\boldsymbol{V}_n^{\mathtt{NPT}}, \mathtt{Gen}_W(\boldsymbol{V}_n^{\mathtt{NPT}}))+ k_{\mathtt{gp}}\frac{1}{N_m^{true}}\sum_{n=1}^{N_m^{true}}\nabla_{\mathtt{GP}}(\mathtt{Gen}_W(\boldsymbol{V}_n^{\mathtt{NPT}}), \boldsymbol{V}_n^{\mathtt{CT}}), \tag{C.4}$$

donde $\boldsymbol{V}_n^{\mathtt{CT}}$ es el CT real de la muestra n^{esima}, $\boldsymbol{V}_n^{\mathtt{NPT}}$ es la entrada NAC PET n^{esima},

$\nabla_{\texttt{GP}}$ es la penalización de gradiente [179] y $k_{\texttt{gp}} = 10.0$. El crítico es entrenado durante 5 pasos por cada paso del generador. En el paso inicial del entrenamiento GAN el crítico es entrenado hasta su óptimo antes de inicializar el lazo de entrenamiento GAN. Los generadores son entrenados usando el optimizador de Estimación Adaptativa de Momentos (ADAM) con parámetros $\beta_1 = 0.0$, $\beta_2 = 0.99$ y $\epsilon = 1.0 \times 10^8$ y taza de aprendizaje $lr = 0.0001$. El discriminador usa el optimizador de RMSprop con taza de aprendizaje de $lr = 0.0005$.

C.3.3 Descripción del conjunto de datos de entrenamiento

El conjunto de datos de HNSCC esta compuesto de una serie de estudios CT y NAC-PET coregistrados de carcinomas espinocelulares de cabeza y cuello. Las muestras seleccionadas del conjunto de datos consisten de estudios de cuerpo completo. El conjunto de datos fue inicialmente despojado de todas las muestras que no contenían muestras coregistradas de NAC-PET y CT. Luego todas las muestras fueron probadas por solapamiento y recortadas a aquellos cortes axiales que cuenten con información de ambos modos de imagen. Luego que el conjunto de datos es limpiado, el mismo el normalizado a un tamaño de FoV de $128 \times 128 \times 256$ voxels con un tamaño de voxel de $5.46 \times 5.46 \times 5.08$ $\texttt{mm}^3$. El conjunto de datos final contiene 118 imágenes de 71 pacientes distintos, de los cuales 7 pacientes (10 imágenes) fueron separadas como conjunto de datos de validación. Antes de alimentar cada muestra a la red, el volumen es cortado de manera aleatoria en un volumen de $128 \times 128 \times 32$ voxels y a todos los voxels del NAC-PET se les añade un valor de $\pm 10\%$ el valor de plena escala y son renormalizadas.

Normalización del CT objetivo:

El CT objetivo debe estar libre de estructuras de la camilla, esto es logrado usando un método basado en la varianza de los voxels a lo largo del eje axial [199]. Luego el rango dinámico de la imagen es recortado entre -125 y 1300 Unidades Housfield (HU) y normalizado entre 0 y 1, para maximizar la distancia entre tejido blando y hueso.

Generación de etiquetas:

Cuatro clases de etiquetas son extraídas de las imágenes de CT no-normalizadas, sin camilla, usando umbrales en los valores de los voxels. La máscara de *Aire-Pulmones* se encuentra entre -1000 `HU` a -125 `HU`, la máscara *Fluidos-Grasa* se encuentra de -125 `HU` a 10 `HU`, la máscara *Tejido Blando* se encuentra de 10 `HU` a 90 `HU` y la máscara *Hueso* se encuentra de 90 `HU` a 1300 `HU`.

C.3.4 Descripción del conjunto de datos de prueba

Los conjuntos de datos de prueba son una serie de conjuntos de datos públicos, también del TCIA, que incluyen diferentes tipos de lesiones, pacientes y tecnologías de tomógrafos. El conjunto de datos esta compuesto por 133 muestras de prueba: 73

del conjunto de datos de Carcinoma Pulmonar No Microcítico (NSCLC) [191], 25 del conjunto de datos del Atlas del Genóma del Cancer - Carcinoma Espinocelular de Cabeza y Cuello (TCGA-HNSC) [192], 20 del conjunto de datos del Atlas del Genóma del Cancer - Adenocarcinoma Pulmonar (TCGA-LUAD) [193], 1 del conjunto de datos del Atlas del Genóma del Cancer - Cáncer Tiroideo (TCGA-THCA) [194], 4 del conjunto de datos del Consorcio del Análisis Preotónico Clínico de Tumores - Adenocarcinoma Pulmonar (CPTAC-LUAD) [195], 1 del conjunto de datos del Consorcio del Análisis Preotónico Clínico de Tumores - Adenocarcinoma Ductal Pancreático (CPTAC-PDA) [196], 3 del conjunto de datos del Consorcio del Análisis Preotónico Clínico de Tumores - Cáncer de Endometrio Uterino (CPTAC-UCEC) [197] y 6 del conjunto de datos del Consorcio del Análisis Preotónico Clínico de Tumores - Carcinoma Espinocelular Pulmonar (CPTAC-LSCC) [198]. Estos conjuntos de datos fueron limpiados de muestras no registradas y normalizadas para adaptarse a la entrada de la red. Durante el proceso de testeo el NAC-PET es dado a la red neuronal en cortes consecutivos de $128 \times 128 \times 32$ voxels. La intensidad es reescalada a $[0, 1]$ para cada corte (no se aplica ningún otro tipo de normalización). El sCT resultante se compone en un solo volumen utilizando una operación de promedio ponderado.

C.3.5 Métricas de corrección de atenuación

Dado que los sinogramas de los conjuntos de datos no están disponibles, no es posible una reconstrucción directa de la actividad del PET usando el modelo propuesto. Como evaluación cuantitativa del desempeño de la corrección de atenuación se comparó la atenuación sobre una Linea de Respuesta (LoR) usando el CT provisto y el sCT generado. La atenuación en las LoRs fue construida usando sinogramas de atenuación de tamaño (180×180) y con un paso axial de 5 `mm`. Cada CT y sCT fue inicialmente convertido de unidades HU a atenuación lineal$[1/cm^2]$, a 120 `keV`. Luego la atenuación lineal fue convertida a energía de PET 511 `keV`. Usando el CT real y el sCT, un par de sinogramas de atenuación fue creado. Por cada una de las LoR en los sinogramas, le coeficiente de atenuación fue calculado como $x_l^\mu = e^{-\sum_{u=1}^{U} h_{l,u}^\mu \mu_u}$. Usando el CT provisto como objetivo, la diferencia entre los sinogramas de atenuación fue medida usando la diferencia media y el desvío estándar de los valores del sinograma. Esto fue aplicado en cuatro muestras, HNSCC-01-0148 del conjunto de datos de validación, muestra AMC-009 de NSCLC Radiogenomics, C3N-00957 de CPTAC-PDA y TCGA-BB-7863 de TCGA-HNSC.

C.3.6 Métricas de la calidad de imagen

La calidad de imagen de los sCT generados fue probada contra el CT real usando tres métricas. Relación de Señal a Ruido Pico (PSNR), Error Absoluto Medio (MAE) y Correlación Cruzada Normalizada (NCC).

C.4 Resultados

C.4.1 Corrección de la atenuación

Las métricas resultantes de los sinogramas de atenuación son resumidas en la tabla C.1. Un anillo central del sinograma de atenuación del CT de referencia y del sCT son mostrados en la figura C.2 para la muestra de validación HNSCC-01-0148 y en la figura C.3 para la muestra AMC-009 del conjunto de datos NSCLC Radiogenomics. Los sinogramas se muestran junto con los histogramas de diferencia de valor para todos los sinogramas del FoV, mostrando la forma de la distribución del error presentado en la tabla C.1.

Topología	HNSCC-01-0148	AMC-009	C3N-00957	TCGA-BB-7863
U-Net	$-0.43 \pm 4.0\%$	$-0.53 \pm 5.5\%$	$0.34 \pm 6.3\%$	$-0.49 \pm 3.6\%$
GAN	$-0.71 \pm 4.2\%$	$-0.65 \pm 5.8\%$	$0.41 \pm 7.7\%$	$-0.66 \pm 3.9\%$

Table C.1: Media y desvío estándar de la diferencia de atenuación entre el CT real y el sCT generado, en porcentaje del valor de referencia, para una muestra de validación y tres muestras de prueba. Una buena métrica estará cerca de cero y tendrá un desvío estándar bajo.

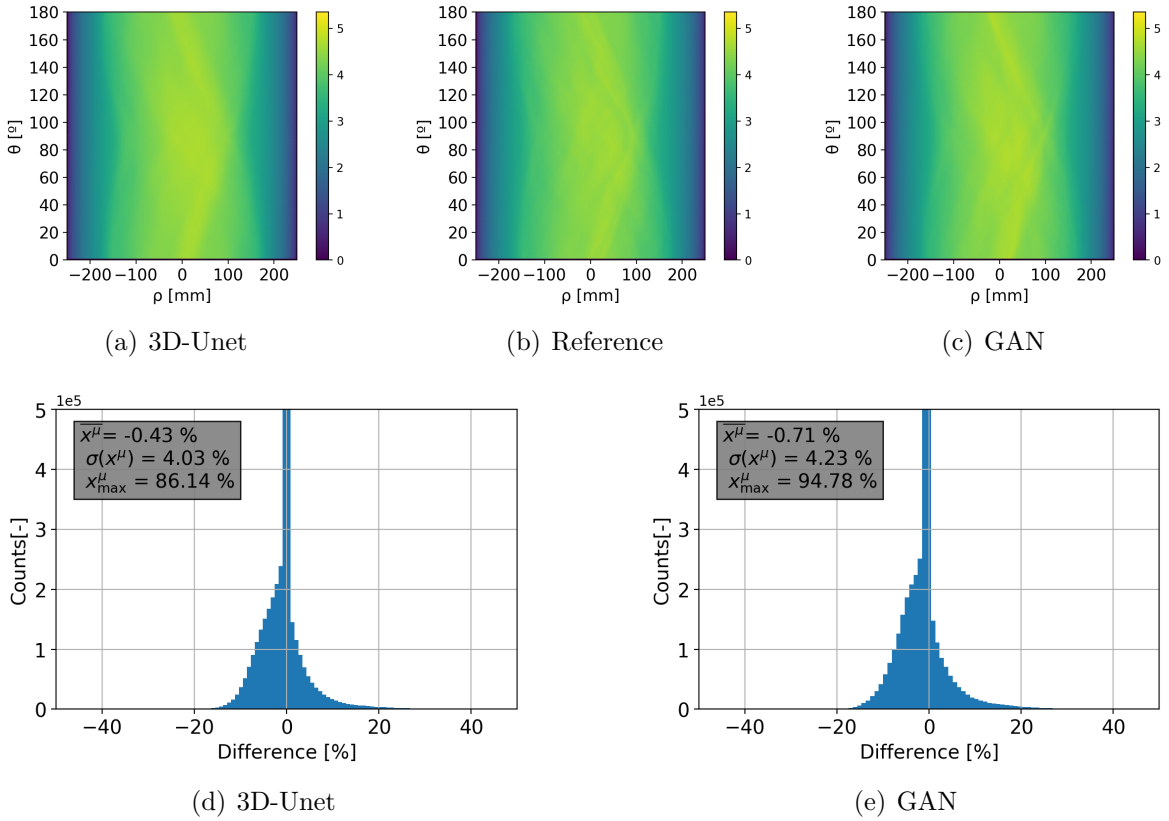
C.4.2 Calidad de imagen

Las métricas de PSNR, MAE y NCC del sCT generado contra el CT objetivo son presentadas en la figura C.4, cada uno de los diagramas de cajas corresponde a las métricas de la 3D U-Net y la GAN en distintos conjuntos de datos. Los conjuntos de datos están presentados por su fuente ya que el número de muestras en los conjuntos individuales es muy pequeña en algunos casos. Los valores para todo el conjunto de validación y de prueba son resumidos en la tabla C.2.

	Fuente	PSNR [dB]	MAE [HU]	NCC [−]
U-Net	Validación	21.0 ± 1.4	80 ± 13	0.80 ± 0.050
	Prueba	19.3 ± 1.7	96 ± 20	0.760 ± 0.060
GAN	Validación	19.9 ± 1.3	89 ± 10	0.760 ± 0.050
	Prueba	18.6 ± 1.5	103 ± 18	0.720 ± 0.060

Table C.2: Valores para el conjunto de validación (10 muestras) y de prueba (133 muestras) para la U-Net 3D y la topología GAN propuesta.

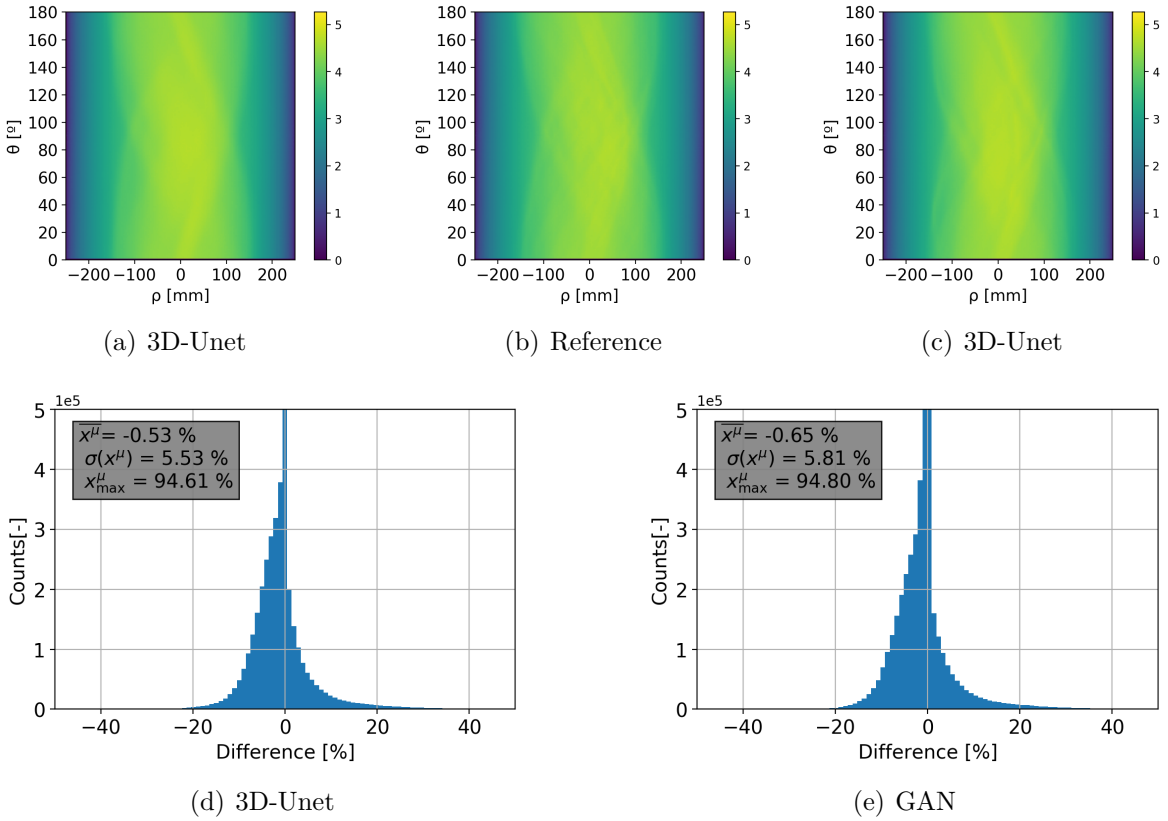
Tres muestras de tres conjuntos de datos pueden verse en las figuras C.5, C.6, yC.7, dos con pacientes con los brazos elevados sobre la cabeza (brazos arriba) y otra con los brazos posicionados al lado del cuerpo (brazos abajo). Estas imágenes corresponden a los conjuntos NSCLC Radiogenomics, CPTAC-PDA y TCGA-HNSC respectivamente.



(a) 3D-Unet (b) Reference (c) GAN

(d) 3D-Unet (e) GAN

Figure C.2: Anillo central de los sinogramas de atenuación para la 3D U-Net sCT (a), el CT referencia (b) y el sCT GAN (c) de la muestra HNSCC-01-0148 del conjunto de datos de validación. Debajo, los histogramas (d,e) de la diferencia de los valores de atenuación, del sCT de la 3D U-Net y el sCT de la GAN 3D, respectivamente.



(a) 3D-Unet (b) Reference (c) 3D-Unet

(d) 3D-Unet (e) GAN

Figure C.3: Anillo central de los sinogramas de atenuación para la 3D U-Net sCT (a), el CT referencia (b) y el sCT GAN (c) de la muestra AMC-009 del conjunto de datos de NSCLC Radiogenomics. Debajo, los histogramas (d,e) de la diferencia de los valores de atenuación, del sCT de la 3D U-Net y el sCT de la GAN 3D, respectivamente.

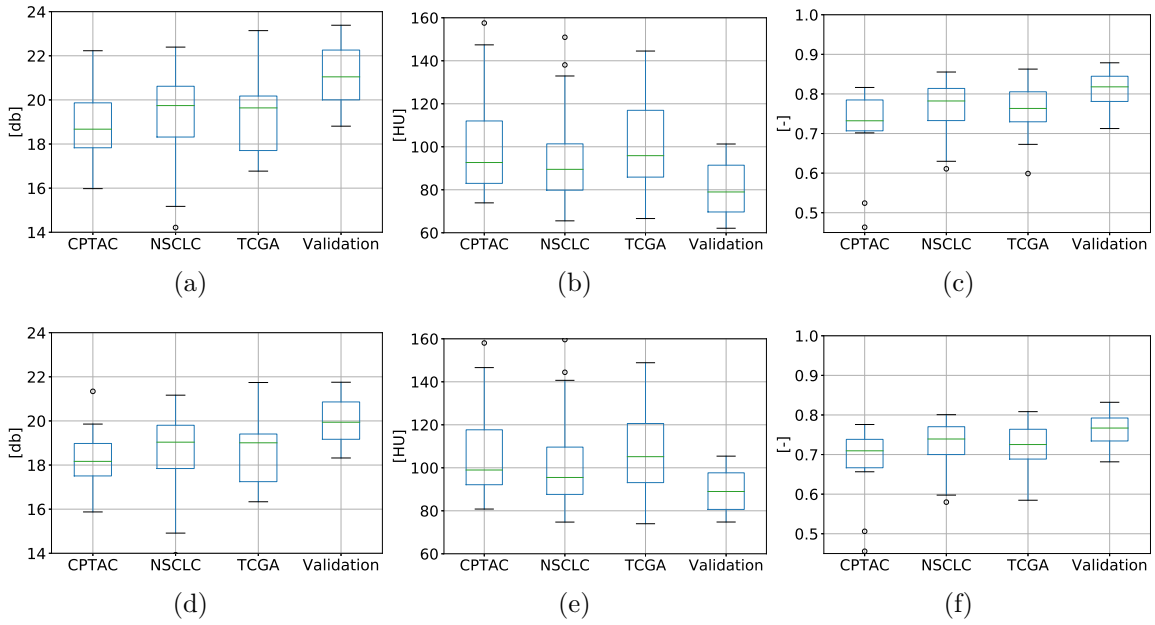


Figure C.4: Diagrama de cajas del PSNR (a, d), MAE (b, e) and NCC (c, f) para cada fuente del conjunto de datos de prueba y de validación. Las figuras (a,b,c) y (d,e,f) corresponden a la 3D U-Net y la red con entrenamiento adversario, respectivamente.

En estas figuras las imágenes de sCT generadas únicamente con la función de pérdida supervisada mostrada en la ecuación C.2, y los sCT generados con la pérdida adversaria mostrada en la ecuación C.3, son comparadas con el CT de referencia. También en estas figuras se presenta una vista 3D del esqueleto generado por cada topología y la estructura de referencia.

C.5 Discusión

La capacidad de corrección de atenuación de la red fue probada usando una serie de sinogramas de atenuación y comprando los sinogramas de los sCT a los sinogramas del CT real. Esta métrica alcanzo un error medio menor al 1% y un desvío estándar del 8% en el peor caso, como se ve en la tabla C.1, el cual puede ser mejorado usando conjuntos de datos más grandes y uniformes (en términos de cobertura anatómica). Mientras que el agregado de capas GAN mejora la apariencia visual de los sCT, una mejora en la corrección de atenuación no se percibe. Contrario al resultado esperado, el modelo completamente supervisado parece ser más adecuado. Nuestra técnica muestra ser resistente a múltiples tipos de reconstrucción y tecnologías de tomógrafos cuando se opera en la resolución seleccionada, como se ve en las métricas de prueba en la figura C.4. La topología U-Net 3D básica genera imágenes de atenuación sintéticas con un PSNR de 19.3 ± 1.7 `dB`, un MAE de 97 ± 20 `HU` y un NCC de 0.760 ± 0.064. La inclusión de las capas GAN logra un PSNR de 18.6 ± 1.4 `dB`, un MAE de 103 ± 18 `HU` y

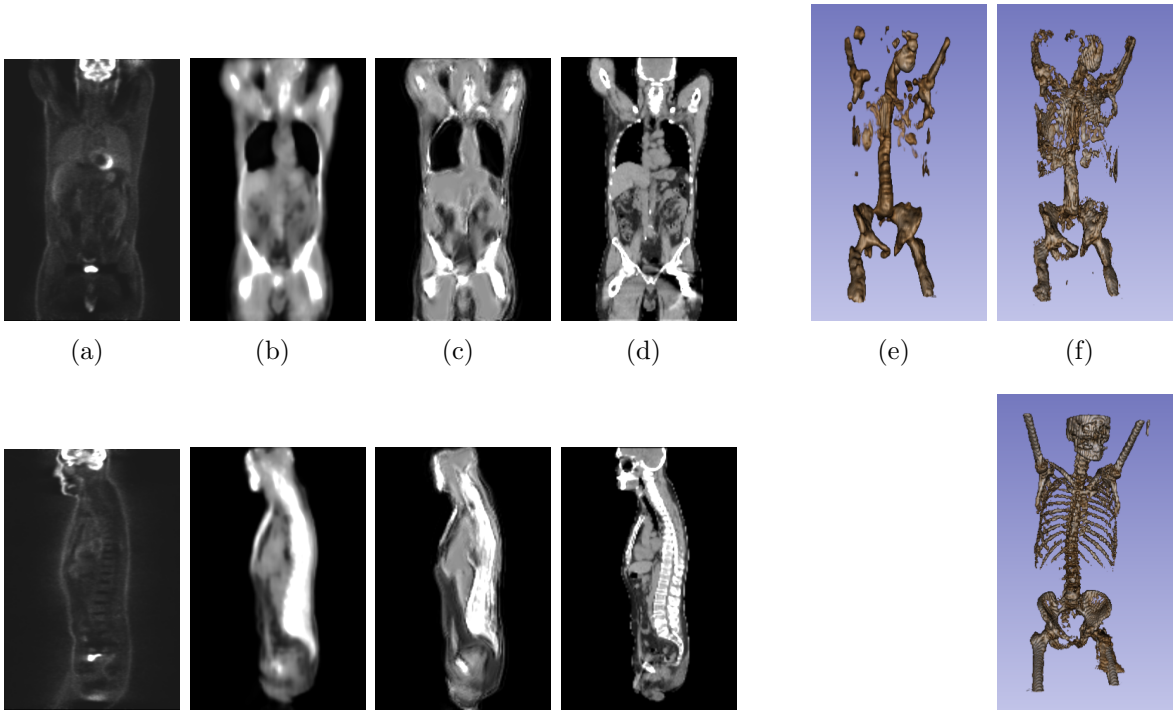
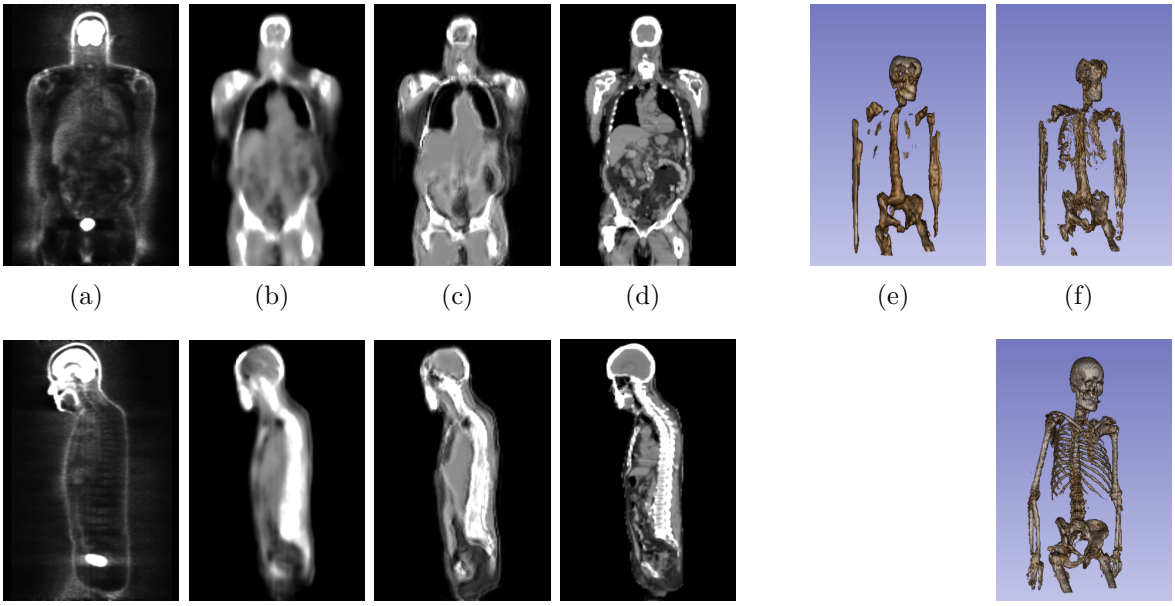


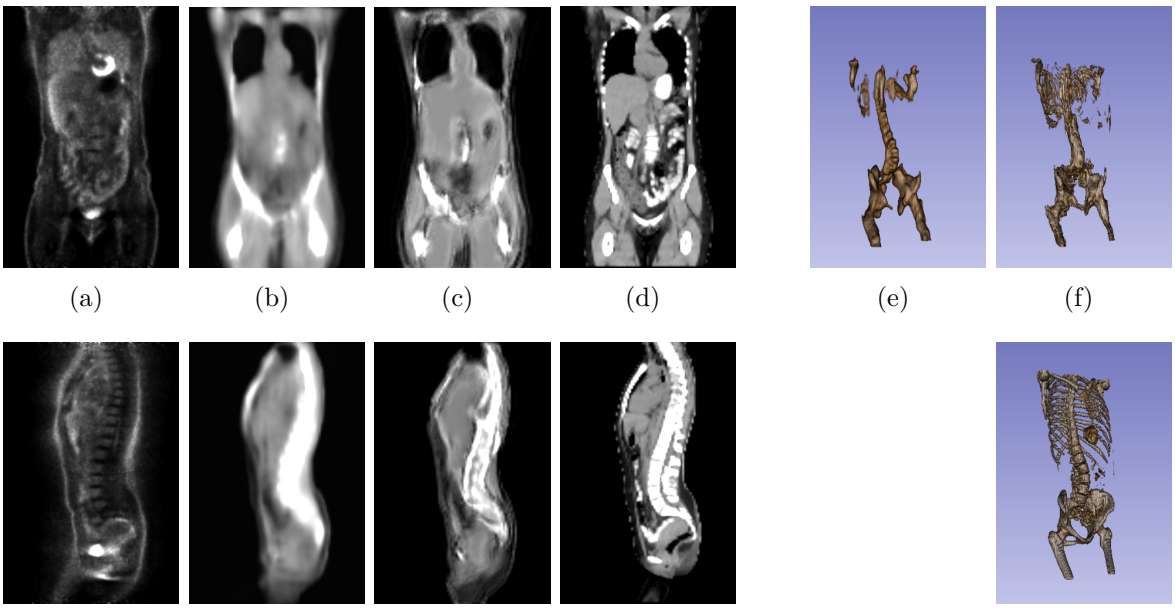
Figure C.5: NAC-PET de entrada (a,g), 3D-Unet sCT (b,e,h), 3D GAN sCT (c,f,i) y CT de referencia (d,j,k) coronal (a,b,c,d), sagital (g,h,i,j) y perspectiva 3D del esqueleto (e,f,k) de la muestra AMC-009 del conjunto NSCLC Radiogenomics. La mejora de la GAN puede verse en la disminución de la difuminación entre las imágenes (b, c) and (h, i). En la perspectiva 3D, red GAN (f) es capás de generar más detalles que la U-Net 3D (e), cuando se la compara con la referencia (k).



(a) (b) (c) (d) (e) (f)

(g) (h) (i) (j) (k)

Figure C.6: NAC-PET de entrada (a,g), 3D-Unet sCT (b,e,h), 3D GAN sCT (c,f,i) y CT de referencia (d,j,k) coronal (a,b,c,d), sagital (g,h,i,j) y perspectiva 3D del esqueleto (e,f,k) de la muestra C3N-00957 del conjunto CPTAC-PDA.



(a) (b) (c) (d) (e) (f)

(g) (h) (i) (j) (k)

Figure C.7: NAC-PET de entrada (a,g), 3D-Unet sCT (b,e,h), 3D GAN sCT (c,f,i) y CT de referencia (d,j,k) coronal (a,b,c,d), sagital (g,h,i,j) y perspectiva 3D del esqueleto (e,f,k) de la muestra TCGA-BB-7863 del conjunto TCGA-HNSC.

un NCC de 0.720 ± 0.059. Estos valores son obtenidos en muestras de distintos tomógrafos, pacientes y lesiones, mostrando que la técnica puede ser usada con distintas fuentes. Se puede ver que la red supervisada genera imágenes borrosas mientras que la GAN muestras detalles de frecuencias más altas como se ve en la figura C.5. Sin embargo estas mejoras no se ven reflejadas en las métricas, resultando inferiores pero no significativamente distintas a las obtenidas por la 3D U-Net base. Ambas, tanto la 3D U-Net como la red adversaria fallan al generalizar la sección superior del cuerpo donde menos datos de entrenamiento estaban disponibles. Esto también se refleja en los brazos, que aparecen en distintas posiciones dentro del conjunto de datos. Mientras esto puede ser solucionado con más datos de entrenamiento, también puede ser mitigado mediante el apareo anatómico de los datos de entrenamiento y entrenando redes específicas para cada región. Una mejora adicional en esta dirección puede lograrse entrenando una red en un espacio intermedio de tamaño total para mapear cada sección del cuerpo a un espacio latente específico, como lo hacen las representaciones intermedias presentadas en [211] y [212]. La generación de sCT actual puede ser también utilizada como imagen a priori para la reconstrucción de la atenuación mediante técnicas como el MLAA y modelado de dispersión simple [56], reduciendo su elevado costo computacional. Estas técnicas pueden potencialmente eliminar artefactos de los mapas de atenuación generados, tales como el contraste estomacal del CT observado en la figura C.7(d) que no está presente en la figura C.7(b) ni en la figura C.7(c).

Las métricas alcanzadas son comparables con el método propuesto por Dong [213], el cual logra un MAE de 108 ± 19 `HU` en la reconstrucción de una imagen de sCT. Sin embargo el modelo presentado modelo es más compacto que la arquitectura de GAN cíclica propuesta por ellos, requiriendo menos parámetros y además controlando el gradiente adversario, resultando en un entrenamiento más estable. También nuestro trabajo esta probado en múltiples tomógrafos PET. Aún más, la mayoría de los tomógrafos probados no poseen capacidad de medición de ToF, por lo que métodos como el MLAA, resultan en mapas de atenuación de cuerpo completo de menor calidad. Finalmente, para poder hacer evaluaciones directas entre los distintos modelos y evaluar su complejidad, es necesario poseer un conjunto de datos común. Por esta razón el conjunto de datos utilizado en este trabajo está liberado [1].

[1] https://github.com/RawthiL/PET_DeepAttCorrection

D Conclusiones Finales y Perspectivas

D.1 Discusión Principal

A lo largo de esta tesis se estudiaron los diferentes aspectos de la tomografía PET, enfocándose en la mejora de las tomografías de baja dosis. El problema de reducir la doses en el paciente durante una tomografía PET fue analizado desde tres frentes.

El prototipo AR-PET

Primero la optimización del tomógrafo AR-PET fue estudiada. Este prototipo, diseñado para ser de bajo costo y permitir el acceso a esta tecnología a comunidades de bajos ingresos, presenta cristales solidos grandes y no posee equipamiento de mapeo de atenuación. Las características del AR-PET jugaron un papel central en los caminos de investigación tomados durante esta tesis. Específicamente dos problemas fueron estudiados para este hardware: Primero el apareo de las ganancias de los PMTs con las cámaras fue analizado y segundo la localización de los eventos dentro del cristal centellador. En la sección A.3 el apareo de los PMTs con las cámaras de centelleo fue analizado y resulto en una optimización aplicada durante la construcción del AR-PET. En este estudio la optimización de la dispersión de la ganancia de los PMTs bajo distintas condiciones de trabajo fue estudiada y un modelo simplificado de la cámara del AR-PET fue construido. Este modelo, y un procedimiento de optimización basado en algoritmos genéticos, permitió la selección de grupos de PMTs para la construcción de las cámaras de centelleo de la AR-PET bajo restricciones severas. La aplicación de esta técnica se tradujo en una mejora de la eficiencia de base de las cámaras. Mientras que este estudio fue probado en la construcción del tomógrafo AR-PET, el concepto de la caracterización del PMT y el modelado de la cámara de centelleo puede ser aplicado a cualquier tipo de cámara de centelleo con cristales sólidos.

La localización de los fotones gamma dentro del cristal sólido fue estudiado en la sección A.5, donde un novedoso algoritmo de localización basado en el apareo de distribuciones con redes neuronales fue desarrollado. La principal contribución de este trabajo fue el desarrollo del algoritmo con capacidad de estimar profundidad de interacción, que no requiere un conjunto de datos complejo para su entrenamiento. Mientras que en los métodos actuales de redes neuronales el conjunto de datos debe contener la localización exacta de cada evento (la cual es normalmente difícil de obtener) el método desarrollado solo necesita conocer la distribución espacial de estos eventos, ignorando la localización particular de cada uno de ellos. Así el método propuesto puede mejorar los defectos de un método de localización de base mediante el apareo de la distribución obtenida por el método base y la distribución esperada durante la adquisi-

ción. El método fue aplicado para la corrección del posicionamiento de los eventos en las cámaras del AR-PET, mejorando el posicionamiento por centro de gravedad. El área sensible de las cámaras fuer mejorada por más de un 23% y capacidad de estimar profundidad de interacción fue incluida.

Imágenes PET en soporte de malla

El segundo tema tratado fue la reducción del número de parámetros utilizados para describir la distribución del radio-trazador en una imagen de PET. Específicamente la representación volumétrica de las imágenes de PET con mallas fue estudiada.Esta reducción de parámetros es importante por dos motivos: Primero, puede traducirse en una mejora de la velocidad de reconstrucción y segundo reduce la dimensionalidad de los datos, lo cual puede ser utilizado en algoritmos de reconstrucción basados en técnicas de optimización. Para probar estas hipótesis en la sección B.2 una reconstrucción mejorada basada en mallas fue desarrollada. En lugar de utilizar mallas 3D poligonales para describir el volumen, una serie de mallas 2D que representan cortes del volumen fue utilizada. Esto fue posible dado que la información usada para reconstruir las imágenes PET, las LoRs, están limitadas a ciertos ángulos, por lo tanto la LoR nunca puede ir a través de un corte en un ángulo perpendicular al mismo. Esta forma de describir el volumen permite el refinamiento de mallas 2D, las cuales tienen un menor número de nodos que las mallas 3D. Mas aún, el refinamiento de dichas mallas puede ser totalmente paralelizado, resultando en menores tiempos de procesamiento. La técnica desarrollada fue implementada usando un algoritmo de trazado de LoR a medida, usado para modelar la adquisición del tomógrafo AR-PET y el software de remallado BL2D-V2. El método desarrollado fue testeado usando una simulación libre de errores de un fantoma complejo del cerebro humano y una adquisición de un fantoma de calidad de imagen NEMA usando el prototipo preclínico AR-PET. Estos fantomas se reconstruyeron usando un algoritmo de MLEM basado en mallas. Usando estos experimentos fue posible mostrar que la aplicación de una serie de mallas 2D para describir una dosis volumétrica no resulta en una perdida de calidad de imagen mientras que reduce la cantidad de elementos necesario para describir la imagen volumétrica por mas de un orden de magnitud cuando se comprara con un soporte voxelizado. Esta reducción es incluso más notoria cuando imágenes voxelizadas de alta resolución son utilizadas. Sin embargo usando un algoritmo de reconstrucción simple como el MLEM, la calidad de las imágenes decae más rápidamente en el soporte de mallas que en el soporte voxelizado cuando el número de LoRs utilizadas en el proceso de reconstrucción disminuye. Sin embargo, la disminución en el número de nodos puede ser utilizada para mejorar la calidad del modelo de reconstrucción, a costos menores que en el soporte voxelizado, y aplacar el efecto de la baja estadística.

Generación del mapa de atenuación utilizando métodos de aprendizaje automático

El tercer y final tema abordado en esta tesis fue la generación de mapas de atenuación para imágenes PET por medio de métodos de ML. Este capitulo se enfocó en analizar

las distintas estrategias para generar un Mapa de Atenuación (AM) usando solamente la imagen NAC PET y aplicando herramientas generativas de ML del estado del arte. Para explorar los métodos de ML se generó un conjunto de imágenes NAC PET coregistradas con estudios CT. El conjunto de datos fue creado con imágenes públicas de la base de datos TCIA. La topología escogida para implementar la generación del AM fue una red neuronal profunda basada en una versión 3D de la U-Net. Dos variaciones se estudiaron en la sección B.2, cada una entrenada de una forma distinta: un entrenamiento totalmente supervisado y un entrenamiento adversario. El entrenamiento totalmente supervisado fue capas de generar AM con una calidad relativamente buena y también fue capas de segmentar distintos tipos de tejidos. Sin embargo las imágenes generadas poseían una perdida de altas frecuencias. Esto era esperable ya que el entrenamiento usó una métrica tipo L2 como función objetivo. Para mejorar la calidad de las imágenes se le adicionó a la red una serie de capas entrenadas de manera adversaria, lo que constituyó un entrenamiento híbrido. Esta red combina el entrenamiento supervisado con una pérdida adversaria restringida utilizada para mejorar los detalles de alta frecuencia. Como resultado las imágenes presentan características más realistas sin perder la especificidad de la red supervisada. El uso de la red híbrida alcanzo métricas del estado del arte. Además la red fue probada en distintos modelos de tomógrafos PET, procedimientos de administración de dosis y procesos de reconstrucción de imagen, mostrando buenos resultados siempre. Los AM generados son de calidad suficiente para ser aplicados directamente a la corrección de imágenes PET según un estudio reciente. Sin embargo se observo que la existencia de estructuras extrañas (como prótesis, contraste o calcificaciones) pueden ser erróneamente removidas de las imágenes de AM generadas. Por esta razón es aconsejable refinar el AM. Dicho refinamiento puede ser hecho utilizando reconstrucción mutua de actividad y atenuación y tomando el AM generado como imagen a priori.

D.2 Perspectivas y trabajo futuro

El camino hacia un sistema PET de bajo costo y sin radiación adicional por estudios de transmisión es largo y esta tesis dio algunos pasos hacia esta meta. Sin embargo hay todavía mucho por hacer. Los fundamentos principales para una reconstrucción de actividad y atenuación fueron desarrollados aquí, una representación y reconstrucción optimizada para imágenes PET y método de generación de AM a priori de alta calidad basado en imágenes. Mientras que no fue posible seguir desarrollando el potencial de la representación de mallas para la reconstrucción de la actividad y los mapas de atenuación (debido a restricciones de tiempo), las bases de la reconstrucción de actividad sobre mallas fue desarrollada junto con la generación de AMs de alta calidad por medio de redes neuronales profundas. Estos trabajos son partes fundamentales para los algoritmos de reconstrucción mutua de actividad y atenuación. La extensión de estos trabajos incluye la extensión del modelos de reconstrucción desarrollado para la reconstrucción mutua de actividad y AM, implementando un MLAA basado en mallas, siguiendo [11,51,52]. Otro camino prometedor es la optimización del método de Simulación de Dispersión Simple (SSS) para aparear las coincidencias de energía baja

(dispersa) y usando una imagen a priori de AM de alta calidad para reducir problemas de convergencia [54, 55]. También, haciendo uso de la dimensionalidad reducida de la representación por mallas, el modelo de reconstrucción puede ser mejorado para mejorar la calidad de imágenes PET de baja dosis [19].

La generación del AM por métodos de ML puede ser mejorado en varios aspectos. Los modelos de generación se benefician mucho con datos uniformes. Esto es parte del éxito de redes generativas grandes como [216]. El apareo histológico de las muestras del conjunto de datos puede mejorar la eficiencia del entrenamiento y por lo tanto la calidad de las AMs generadas. Otra mejora del modelo puede ser el uso de un espacio latente intermedio de alta resolución del cual el modelo generativo pueda muestrear. Esto es propuesto en [212] donde implementan dicho espacio latente para mejorar la representación 3D de las imágenes. También el desarrollo de nuevas herramientas para el análisis de datos 3D, como [172], debe ser probado. Estas herramientas permiten la rápida implementación de modelos 3D que analizan datos voxelizados o de nubes de puntos. El análisis de nubes de puntos, como [167–170], puede ser directamente implementado sobre los nodos de una representación por mallas, reduciendo la dimensionalidad y evitando errores de proyección.

Finalmente, se espera también que la generación del AM pueda ser probada en datos de pacientes de tomógrafo AR-PET. El trabajo presentado en esta tesis ayudó la construcción y prueba del AR-PET. El prototipo del tomógrafo se encuentra actualmente bajo prueba y espera ser aceptado por la Administración Nacional de Medicamentos, Alimentos y Tecnología Médica (ANMAT) en Argentina para comenzar las pruebas sobre pacientes.

French Extended Abstract

Contents

Introduction

La tomographie par Émission de Positrons (connue sous son acronyme en anglais, PET) continue d'être l'un des outils les plus puissants pour la détection et la surveillance du cancer, la mesure du métabolisme et l'étude des médicaments. Aujourd'hui, les scanners PET sont utilisés en médecine avec le traceur ^{18}F-fluoro-2-deoxy-D-glucose (FDG), principalement pour les patients avec un cancer suspecté ou diagnostiqué. L'utilisation des analyses PET pour les études de screening de patients asymptomatiques est également à l'étude [1, 2]. Cependant, l'utilisation de cette technique peut provoquer un risque considérable pour le patient. Ce risque provient de la radiation absorbée par le patient, qui est estimée entre 10 `mSv` et 18 `mSv` pour une analyse de screening [3, 4]. Cette dose est considérée comme une *dose importante* par la Commission Internationale de Protection Radiologique (connue sous son acronyme en anglais, ICRP) [5] et c'est l'un des arguments utilisés contre l'utilisation de la PET comme étude de screening pour les patients asymptomatiques [6]. Mais la dose de radiation d'une étude PET ne provient pas entièrement du FDG, entre 1.3 `mSv` et 8.0 `mSv` de la dose proviennent de l'image CT dans un étude de screening moyen (si vous utilisez un scanner à petite dose) [3, 4, 7]. L'étude CT est considérée à tort comme une étude *à petite dose* par les médecins et doit être évité dans la mesure du possible [8]. Par conséquent, la réduction de la dose dans les études PET et CT, en particulier, sont des champs actifs d'investigation dans les images PET [9, 10]. La dose de l'étude peut être encore réduite en supprimant complètement l'étude CT de l'étude PET, sans perte de précision ou de qualité d'image. À cette fin, l'idée de reconstruire la carte d'atténuation (AM), sans utiliser l'étude CT est explorée avec des modèles de plus en plus complexes [11,12], et plus récemment en utilisant des modèles d'apprentissage automatique basés sur des images PET sans correction d'atténuation (PET NAC) [13–15] ou d'autres modalités [16, 17].

Dans ce cadre, la contribution de cette thèse est l'application de techniques d'apprentissage automatique (ML) et d'échantillonnage dispersé pour réduire la dose des patients dans les études PET en raison de l'étude CT. Plus spécifiquement, les méthodes d'obtention d'une AM à partir d'images PET NAC sont étudiées, qui permettent l'utilisation de méthodes à coût de calcul plus élevé. Ceci est fait accompli en utilisant une représentation maillée de l'image et en générant des images AM de haute qualité a priori en utilisant des méthodes ML. Pour cela, un schéma de reconstruction est effectué sur des maillages adaptatifs, en utilisant des maillages de haute qualité pour appliquer des méthodes de reconstruction plus complexes, qui r le coût de calcul asymptotique et permettre la génération ou l'amélioration de le AM et améliore la qualité de l'image PET en général. Dans un deuxième temps, les Réseaux Neuronaux à Convolution Profonde (DCN) et les Réseaux Neuronaux Génératifs Antagonistes (GAN) sont étudiés, pour produire le AM en obtenant des résultats supérieurs à l'état de l'art. En générant

des AMs a priori de haute qualité avec des techniques ML et en réduisant le coût de calcul des modèles de constructions complexes, permettent d'espèrer réduire le coût et améliorer la convergence des méthodes de reconstruction d'image et de correction de gradation basées sur la simulation de dispersion simple (SSS) [18], modèles de résolution [19] et simulations de Monte Carlo [20].

Schème du manuscrit

L'objet de cette thèse ouvre la voie à une étude PET à petite dose, en éliminant le besoin d'une étude CT ou d'une étude de transmission gamma. Les contributions sont basées sur des simulations, des bases des données publiques et des mesures à partir d'un prototype de scanner PET à petite dose. Ce résumé est divisé en quatre chapitres principaux. Dans le chapitre E le prototype AR-PET est rapidement présenté avec deux travaux d'optimisation de ses détecteurs gamma à cristal continu. Après dans, le chapitre F, l'utilisation de maillages pour la reconstruction d'images PET est présentée. Ici, la capacité des maillages à s'adapter et à décrire une image PET est démontrée. Les maillages représentent l'image PET sans perte de qualité et réduisent le nombre de paramètres par rapport à une représentation voxélisée. Enfin le chapitre G reprend le thème de l'utilisation de DCN et GAN pour le problème de la génération de AM à partir d'une image PET NAC. Les architectures et la base des données utilisés sont présentés, montrant les capacités et les défauts de ces techniques. Le code pour reproduire ces résultats avec la base des données est fourni.

Dans le dernier chapitre, les conclusions générales de la thèse sont présentées et les résultats et les projections de leurs contributions sont discutés.

Pour faciliter la lecture de la version complète de la thèse, rédigée en anglais, toutes les abréviations et symboles mathématiques sont conservés de la même manière que dans le manuscrit original, afin qu'ils correspondent à leur signification en anglais.

Les résultats présentés dans cette thèse sont soutenus par les publications suivantes:

- "Heuristic Method for Photo-detectors Localization over Continuous Crystal Scintillation Cameras" [21].
- "Event Localization in Continuous Crystal Scintillation Cameras using Distribution Matching Neural Networks" (Accepté dans "Third Argentine Conference on Electronics" de la IEEE, 10-12 Mars 2021).
- "Reconstruction of Positron Emission Tomography images using adaptive sliced remeshing strategy" [23]
- "Multimodal Brain Tumor Segmentation Using 3D Convolutional Networks" [24]
- "Whole Body [^{18}]F FDG Positron Emission Tomography Attenuation Correction Map Synthesizing using 3D Deep Generative Adversarial Networks" (soumis)[1]

[1]Preprint disponible [217]. Le code peut être trouvé dans: https://github.com/RawthiL/PET_DeepAttCorrection

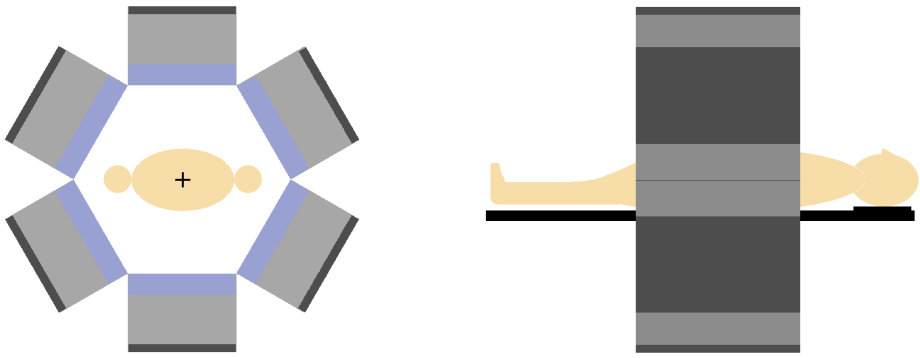
E Prototype AR-PET

Ce chapitre présente le prototype de scanner AR-PET, qui est utilisé tout au long de la thèse. Ici, la conception du scanner est rapidement décrite et l'accent est mis sur les caméra de détection gamma, pour lesquelles un critère de sélection de Tube photomultiplicateur (PMT) est développé. Ce nouveau schéma de construction utilise des algorithmes génétiques pour améliorer l'homogénéité des gains PMT. Ensuite, pour l'optimisation de la caméra gamma, un nouvel algorithme de positionnement d'événement est présenté. Ce schéma de positionnement est basé sur des réseaux de neurones et enseigné avec une seule acquisition de champ inondé. Le positionnement qui en résulte présente des améliorations par rapport à l'algorithme du centre de gravité, augmentant ainsi la zone sensible de la caméra gamma.

E.1 L'AR-PET

L'AR-PET est un prototype de scanner conçu pour réduire le coût des études PET et permettre l'extension de l'utilisation de ce type de techniques dans les populations à faibles revenus, comme les États d'Amérique latine. Dans ces communautés, le nombre de caméras PET par million d'habitants est proche de $0,5$ contre $1,5$ en Europe et 7 aux États-Unis d'Amérique. Le prototype est développé par le Département d'Instrumentation et Contrôle de la Commission Nationale de L'Énergie Atomique (CNEA) en collaboration avec le Groupe d'Intelligence Artificielle et de Robotique (GIAR) de l'Université Nationale de Technologie (UTN). Actuellement, le premier prototype est en train d'être testé dans le centre de médecine nucléaire de l'hôpital clinique José de San Martín à Buenos Aires, en Argentine.

Le tomographe se compose de six grandes caméras de scintillation disposées en forme hexagonale, comme le montre la figure E.1. Dans la figure E.2 une vue 3D de la construction interne du scanners est observée. Une grande partie du coût d'un scanner PET commercial provient des cristaux et de la difficulté de fabriquer de petits pixels, chaque caméra de scintillation est constituée d'un cristal NaI (TI) continu. Ces cristaux continus ont une grande couverture géométrique et un bon pouvoir d'arrêt à faible coût. Cela permet de réduire la dose de radiation au patient ainsi que la durée totale de l'étude, deux facteurs limitants pour une large application des études PET. Une configuration similaire à celle de l'AR-PET qui utilise 4 cristaux continus, a été appliquée avec succès en tomographie PET pour les petits animaux [84]. Pour réduire les effets des angles morts et des inhomogénéités de la disposition hexagonale les caméras gamma du scanner tournent autour de l'axe axial à 2 `rpm`. En raison du mode d'acquisition rotationnel, de la grande surface de détection et de la capacité de détection de la profondeur d'interaction (DoI), le scanner accumule les données en



(a) (b)

Figure E.1: Vues de face (a) et latérale (b) de la géométrie du scanner AR-PET. La structure du scanner est représentée en gris, la surface des détecteurs en bleu clair et la civière du patient en noir.

mode liste. Enfin, la reconstruction de la distribution des radiotraceurs est réalisée sur un serveur externe à l'aide de l'algorithme ML-EM [146]. Le scanner n'a pas la capacité de mesurer la carte d'atténuation. Actuellement, l'appariement d'études externes est utilisé pour compenser les effets d'atténuation lors de la reconstruction.

E.2 Caméras de scintillation

Chacune des caméras de scintillation est composée de cristaux de NaI(TI) continus de $406.4 \times 304.8 \times 25.4$ $\mathtt{mm}^3$, peuplé d'une collection de 8×6 PMT *Hamamatsu R1534*. Ces PMT ont un profil carré de 52.4×52.4 $\mathtt{mm}^2$ qui couvre complètement la surface du cristal sans un couplage mécaniquement complexe. Une représentation du cristal de scintillation et de la caméra de détection peut être vue sur la figure E.3.

Pour répondre aux limites de l'utilisation de cristaux monolithiques, le prototype AR-PET dispose d'une architecture de traitement distribuée dans chaque PMT [85]. Les 48 PMT ont une Matrice de Portes Logiques Programmables sur Site (connue sous le nom de FPGA sous son acronyme en anglais) de type *Spartan3e* lesquelles communiquent par un signal différentiel basse tension à un FPGA *Artix 7-100* qui centralise toutes les informations de la caméra. Cela permet, d'une part, que chaque caméra AR-PET fonctionne indépendamment comme une caméra gamma, sans avoir besoin des 6 caméras pour fonctionner. D'autre part, il fournit une puissance de traitement à grande vitesse sur les signaux d'énergie détectés.

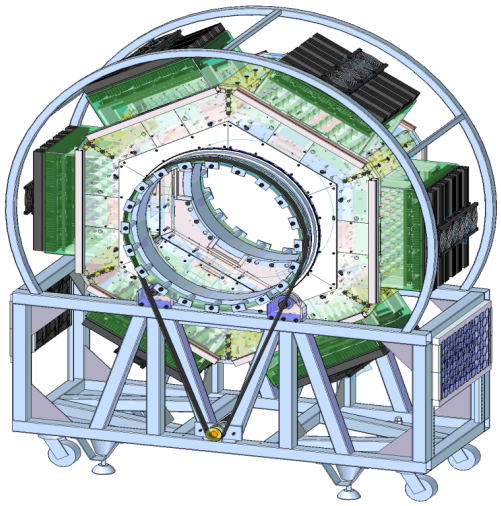
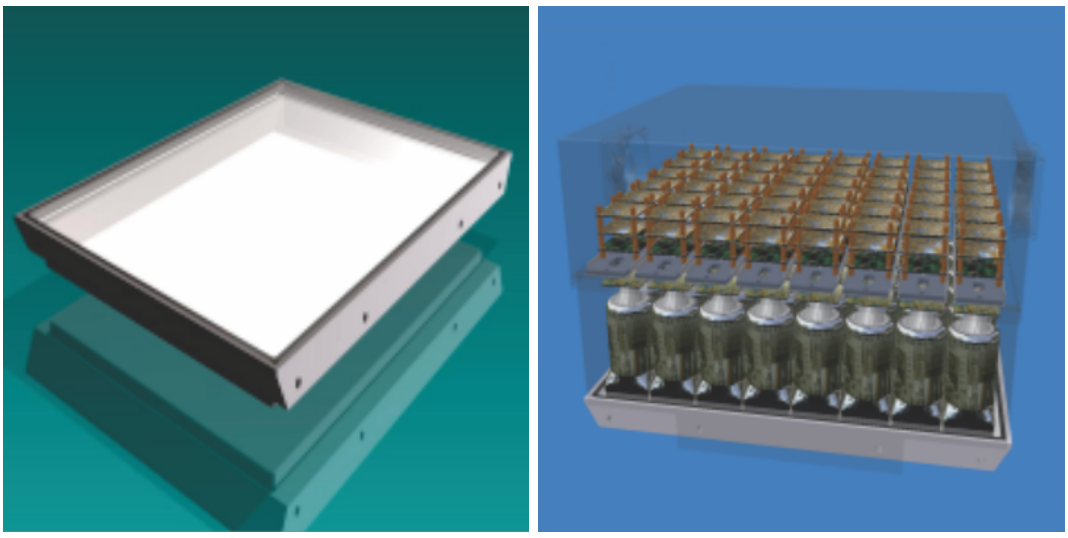
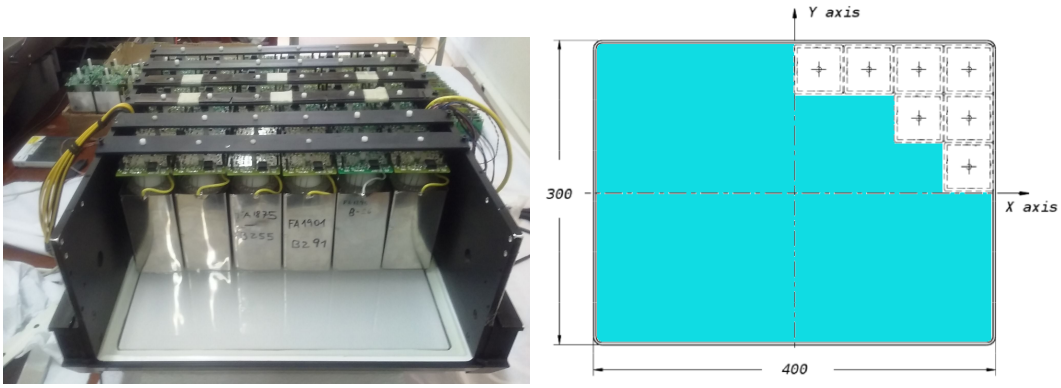


Figure E.2: Vue en perspective de l'arrangement intérieur avec le système de rotation du prototype AR-PET.



(a) (b)

Figure E.3: cristal scintillateur (a) et caméra de détection complète (b) du scanner AR-PET.



(a) (b)

Figure E.4: Disposition des tubes photomultiplicateurs dans la caméra (a) et schéma de positionnement (b). En (b) l'unité de détection est affichée en blanc sur un cristal de scintillation coloré (Image de E. Venialgo, 2011 IEEE N.S.S).

E.3 Sélection de tubes photomultiplicateurs avec des algorithmes génétiques

Un problème connu dans la construction de grandes caméras de scintillation avec des cristaux continus est les effets d'une mauvaise configuration qui affecte directement la résolution finale de la caméra. Ce problème devient encore plus complexe lorsque le nombre de PMT utilisé est important. C'est le cas de l'AR-PET, où 48 PMT sont situés dans chacune des six caméras de scintillation selon la figure E.4. Après l'interaction du photon gamma avec le cristal scintillateur, des centaines ou des milliers de photons lumineux sont créés dans un flash omnidirectionnel. Certains de ces photons sont directement absorbés par les PMT et certains sont réfléchis contre les côtés du cristal avant d'atteindre un PMT. Par conséquent, les PMT près du bord du cristal reçoivent une quantité de lumière significativement plus importante que ceux du centre. De plus, les PMT ont une grande dispersion d'une unité à l'autre, étant donné les mêmes conditions de fonctionnement électrique. Pour obtenir un comptage uniforme pour toute la surface du cristal, il est nécessaire d'égaliser le gain d'énergie et l'efficacité quantique de chaque PMT. De plus, pour l'application PET, il est nécessaire d'égaliser le temps de transit de chaque PMT. Ce temps de transit du PMT est défini par le temps qu'il faut au PMT pour transformer et amplifier le signal optique en une impulsion électrique mesurable. Ce temps doit être le même dans tous les PMT pour éviter les pertes d'intégration dans le calcul de l'énergie du photon gamma. Le temps de transit dépend de la valeur de la Haute Tension (HV) appliquée au PMT [86]. Cette HT est fixé pour tous les PMT dans une caméra. Par conséquent pour contrôler le gain du PMT, la tension entre les premières dynodes est contrôlée [87], sans changer le temps de transit de l'unité. Il existe différentes approches pour réduire l'effet de dispersion du gain du photomultiplicateur. La plus courante consiste à utiliser des coefficients programmatiques et une réduction des comptes d'écrénage (*count skimming*) [88] ou calibrer

chaque PMT à l'aide de ses informations spectrales [89]. Des méthodes de materiel ont également été étudiées, telles que les techniques de quadrant partagé PMT [90] et la inclusion des diodes électroluminescentes (LED) à l'intérieur de la chambre de scintillation pour calibrer les PMT par groupes [91] ou calculer des matrices de corrélation au niveau materiel [92]. La réduction des agrégats de photomultiplicateurs en segmentant le cristal est également une décision de conception du scanner qui réduit l'impact de ce problème [93] mais cela ne s'applique pas aux cristaux continus. Toutes les méthodes mentionnées sont axées sur la correction du gain PMT après leur installation à l'intérieur de la caméra de scintillation, en supposant que chacune des unités PMT peut être positionnée n'importe où sur le cristal scintillateur. Cela peut être le cas si l'on peut choisir parmi un grand nombre de PMT et donc choisir ces PMT avec des caractéristiques similaires. Dans le cas où la quantité de PMT à utiliser n'est pas suffisamment grande pour exclure de grandes variations de gain entre les PMT, une stratégie de positionnement doit être envisagée.

Afin de créer des caméras de scintillation fonctionnelles en utilisant peu de PMT et en améliorant également l'efficacité d'acquisition initiale, une nouvelle méthode a été développée qui génère automatiquement des groupes de PMT sous-optimaux pour chaque caméra de scintillation en utilisant l'agglomération par les k-voisins plus proches (kNN) et des algorithmes génétiques pour explorer les configurations possibles.

E.3.1 Réflexions dans le cristal

La définition d'une configuration de PMT dans la caméra de scintillation nécessite de connaître le comportement sur les bords du cristal et de la modulation du dynode PMT. Le premier effet est une conséquence de la construction du cristal, qui est enduit d'une peinture réfléchissante pour diriger tous les photons générés vers la surface du cristal, la structure de la caméra qui contient les PMT et le couplage optique des PMT avec le cristal. La peinture réfléchissante est importante pour éviter la perte de lumière sur les bords du cristal et permet à la caméra de mesurer correctement l'énergie déposée par le photon gamma. Une effet indésirable est produit par le positionnement:, un PMT situé sur le bord reçoit plus de lumière si un photon gamma frappe son centre par rapport au même PMT au centre du cristal. En utilisent le logiciel *Geant4 Application for Tomographic Emission* (GATE, Application Geant4 pour la Tomographie par Émission) une simulation Monte Carlo de la caméra de scintillation du AR-PET a été réalisée pour des PMT idéaux (comptage parfait de photons). Une source gamma uniforme a été placée devant la caméra. Les effets de scintillation et de transport des photons visibles ont été simulés, obtenant le nombre de photons reçus par chaque PMT dans chaque position du cristal. Les résultats de la simulation montrent trois groupes différents de sensibilité: le centre, les bords et les coins. Le groupe central est composé de ces PMT qui ne sont pas directement sur le bord du cristal. Deux facteurs ont été obtenus de ces trois groupes. Ces facteurs mettent en relation l'énergie observée lorsqu'un événement est détecté sur un bord ou un coin de la caméra par rapport à la

détection du même événement au centre de la chambre:

$$\nu_{c,c}^{\texttt{Simulation}} = \frac{\sum_{p=1}^{P\in\texttt{Corner}} \zeta_p}{\sum_{p=1}^{P\in\texttt{Center}} \zeta_p} = 1.52, \tag{E.1}$$

$$\nu_{c,e}^{\texttt{Simulation}} = \frac{\sum_{p=1}^{P\in\texttt{Edge}} \zeta_p}{\sum_{p=1}^{P\in\texttt{Center}} \zeta_p} = 1.21, \tag{E.2}$$

ou P est le nombre total du PMT, $P \in \texttt{Center}$, $P \in \texttt{Border}$ et $P \in \texttt{Edge}$ représentent le sous-groupe de PMT dans chaque région de chambre et ζ est le nombre moyen de photons visibles détectés. Ces facteurs représentent la relation entre le nombre de photons pour le *centre à coin* et *centre à bord*, respectivement.

E.3.2 Gain du photomultiplicateur

À cause de la non-linéarité et de la grand dispersion entre les différents PMT pour la même valeur de HV et de dynodes, les PMT sont difficiles à caractériser. Pour tester leur comportement, un processus d'étalonnage sur mesure a été réalisé. Dans le processus d'étalonnage, le PMT est couplé à un cristal NaI(TI) de 25.4 mm de large et de 76.2 mm de diamètre et positionné à l'intérieur d'une chambre sombre. Le détecteur est excité avec une source de césium-137 ($^{137}_{55}$Cs) et la valeur du pic de l'ADC est notée. Ceci est répété pour chaque point de fonctionnement prédéfini. Pour chaque PMT, quatre valeurs HV et cinq valeurs de dynode ont été testées, produisant 20 différents points de travail. Les valeurs de dynode sont exprimées en fonction du Convertisseur Analogique-Numérique (DAC) qui contrôle la tension finale de la dynode. Le DAC de la dynode est codée sur 12 bits de résolution, ce qui signifie une échelle de $[0, 4095]$. Les points de fonctionnement PMT uniques peuvent être représentés par quatre courbes, comme indiqué sur la figure E.5. Dans la figure on observe que les points de travail d'un PMT varient avec la valeur de la dynode en suivant une fonction non linéaire. Le comportement observé est ajusté a une courbe sigmoïdale. Cela reflète la non-linéarité, à la fois au début, lorsque la dynode est court-circuitée, et à la fin, où elle sature.

E.3.3 Contraste du modèle

Étant donné que les courbes d'étalonnage sont générées à l'extérieur de la chambre où le PMT fonctionnera, un facteur de correspondance $\chi_{\texttt{cal}}$ entre le cristal d'étalonnage et la caméra est dérivé. Ce facteur est calculé en positionnant un groupe de PMT dans chaque position possible de la chambre de scintillation avant et après calibration. La construction d'une chambre de scintillation étant un processus complexe, seul un groupe de 48 PMT a été mesuré. Le facteur obtenu concerne la quantité de lumière observée lors de l'étalonnage et les différentes positions du PMT à l'intérieur de la caméra. Le facteur est calculé comme le rapport entre le canal ADC pendant l'étalonnage (qui est proportionnel aux photons détectés $\zeta_{\texttt{cal}}$) et le canal ADC à l'intérieur de la caméra

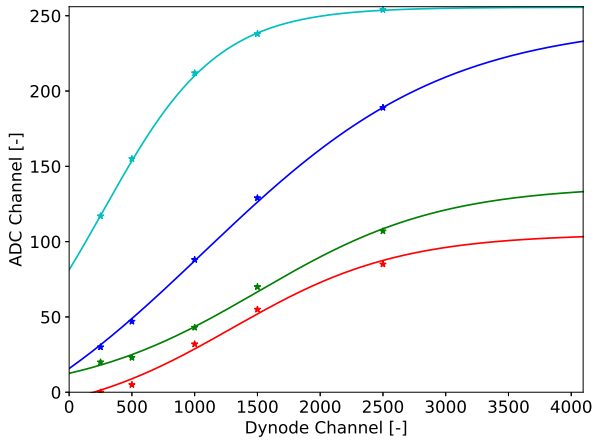


Figure E.5: Points de fonctionnement du photomultiplicateur comme valeur ADC en fonction de la valeur DAC dynode, $[0, 4095]$, et courbes ajustées. Valeurs HV : 1300 V (bleu clair), 1200 V (bleu foncé), 1100 V (vert) et 1000 V (rouge).

(qui est proportionnel aux $\zeta_{\texttt{cam}}$), comme indiqué dans l'équation E.3:

$$\chi_{\texttt{cal}}^{\texttt{region}} = \frac{\zeta_{\texttt{cal}}}{\zeta_{\texttt{cam}}^{\texttt{region}}}. \tag{E.3}$$

En utilisant les 48 PMT, le rapport de lumière pour chaque PMT peut être extrait. Les valeurs des facteurs de correspondance empirique et le nombre de photons détectés dans la simulation (ζ_p) sont représentés sur la figure E.6. En prenant la valeur moyenne sur chaque région (centre, coin et bord), un facteur d'appariement moyen est obtenu. Les facteurs moyens résultants étaient: $\chi_{\texttt{cal}}^{\texttt{center}} = 0.44$, $\chi_{\texttt{cal}}^{\texttt{edge}} = 0.54$ et $\chi_{\texttt{cal}}^{\texttt{corner}} = 0.71$. Enfin, en comparant les gains à chaque position de caméra, les mêmes facteurs sont obtenus dans la simulation de la section E.3.1:

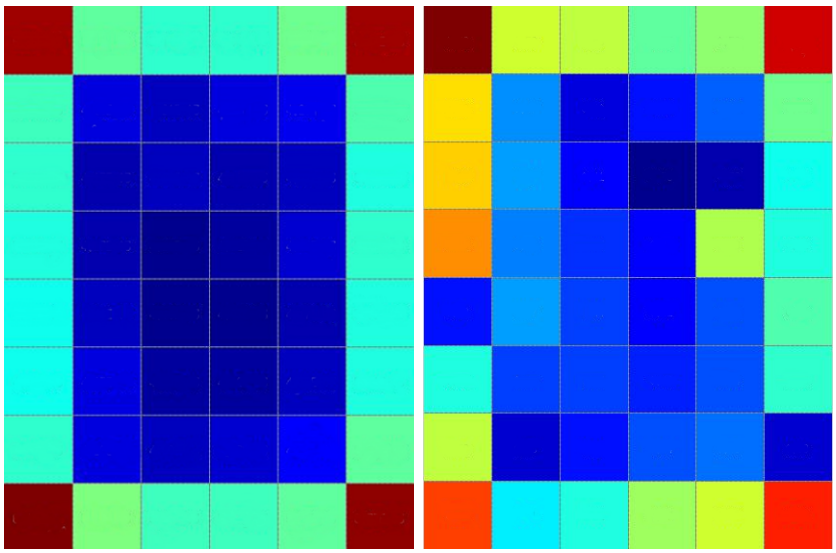
$$\nu_{c,c}^{\texttt{Empirical}} = \frac{\chi_{\texttt{cal}}^{\texttt{corner}}}{\chi_{\texttt{cal}}^{\texttt{center}}} = 1.61, \tag{E.4}$$

$$\nu_{c,e}^{\texttt{Empirical}} = \frac{\chi_{\texttt{cal}}^{\texttt{edge}}}{\chi_{\texttt{cal}}^{\texttt{center}}} = 1.22. \tag{E.5}$$

On peut voir que ces facteurs sont très similaires aux valeurs simulées $\nu_{c,c}^{\texttt{Simulation}} = 1.52$ et $\nu_{c,e}^{\texttt{Simulation}} = 1.21$, ayant une différence maximale de 6%. Cela valide la précision de la procédure d'étalonnage.

E.3.4 Prétraitement par K-means

Le point de départ de l'algorithme de positionnement des PMT est un groupement par K-means. Le nombre de configurations initiales possibles pour une caméra est donné par le nombre de PMT disponibles et le nombre de PMT à positionner dans les différentes caméras de scintillation à construire. Cette valeur est obtenue par la



(a) (b)

Figure E.6: Rapport entre la quantité de lumière attendue au centre de la caméra gamma AR-PET et la lumière reçue dans chacun des PMT dans la simulation GATE (a) et dans l'étude expérimentale (b). La couleur de (a) représente le nombre de photons détectés, allant de plus de photons (rouge) à moins de photons (bleu). Dans (b) la couleur représente la correspondance entre les facteurs dérivés de l'équation E.3.

formule de permutations dans l'équation E.6, où P' est le nombre total de PMT et P est le nombre de PMT à positionner.

$$\texttt{Perm} = \frac{P'!}{(P'-P)!}\,. \tag{E.6}$$

Afin de générer un bon point de départ, ce qui aide à la convergence de l'algorithme génétique ultérieur, les PMT sont regroupé autour d'une valeur HV commune. Les clusters sont créés en choisissant le canal objectif de l'ADC pour l'énergie de crête et en mesurant la similitude entre les valeurs de la dynode. En utilisant l'algorithme k-means [94], trois groupes de PMT sont générés pour chaque valeur de HV. Les centres, ou représentants, de ces groupes sont donnés en unités de dynodes et leurs valeurs ne sont pas données a priori. L'algorithme converge en utilisant le canal ADC objectif et les facteurs de correspondance $\chi_{\texttt{cal}}^{\texttt{center}}$, $\chi_{\texttt{cal}}^{\texttt{edge}}$ et $\chi_{\texttt{cal}}^{\texttt{corner}}$. Cela signifie qu'il y a un groupe par position dans la caméra de scintillation. Si plusieurs caméras doivent être générées, les PMT de la première caméra sont extraites de la liste et le processus de sélection se poursuit jusqu'à ce que toutes les caméras soient remplies.

E.3.5 Algorithme génétique

Puisqu'une recherche exhaustive du problème n'est pas possible, une recherche sous-optimale est appliquée. Les algorithmes génétiques sont préparés pour cette tâche et leur convergence est montrée [95]. Cette technique nécessite la description des *individus*, qui sont, dans ce cas, les configurations possibles des caméras à scintillation. L'individu est décrit comme l'ordre des PMT dans chaque caméra de scintillation à construire et la valeur HV pour chacune de ces chambres. Le *génome* résultant de chaque individu est un vecteur à $[(P+1)\times D] + P_{\texttt{slk}}$ éléments, où D est le nombre de caméras (où détecteurs) à construire et $P_{\texttt{slk}}$ est la variable de relâchement, qui représente le nombre de PMT de rechange dont disposent les caméras. Ces PMT de rechange ne sont pas pris en compte dans la function de forme mais sont utilisés pendant le processus de mutation. Au début, une population de $\Xi = 30$ individus est créée en utilisant le regroupement de k-means. Le fonction de forme initiale de la population est calculée, puis le processus de sélection commence. Ce processus de sélection se fait par mutation et élitisme (ou réplication) [96]. Pour chaque itération, un certain nombre d'individus $\Xi_{\texttt{elit}} > 0$ est choisi pour être copié sans modification à l'étape suivante de l'algorithme, c'est ce qu'on appelle l'élitisme. Ensuite, un processus de *mutation* est appliqué à la population, le processus de *croisement* n'est pas utilisé. La valeur de fonction de forme ξ de chaque individu de la nouvelle population est calculée et la probabilité de survie p_m du $m^{\text{ème}}$ individu est donnée par:

$$p_m = \frac{\xi_m}{\sum_{n=1}^{\Xi}\xi_n}. \tag{E.7}$$

Après ce processus de sélection, une nouvelle population est créée et l'algorithme passe à l'itération suivante. Ceci est répété jusqu'à ce qu'un individu atteigne la fonction de forme souhaitée ou que le nombre total d'itérations soit atteint.

E.3.6 Calcul de la fonction de forme

La fonction de forme d'un individu reflète la plus grande différence entre les valeurs de dynode entre deux PMT à la même HT. Cette valeur de dynode est proportionnelle à HV, mais est donnée en unités de bits. En ce sens, plus la valeur de fonction de forme est basse, meilleure est la configuration (l'individu). Le fonction de forme de la configuration est calculée comme la plus grande différence entre les valeurs de dynode parmi toutes les caméras à scintillation. Cette métrique est décrite par:

$$\xi_n = \max(\Delta \mathtt{v}^1_{\mathtt{dn}}, \Delta \mathtt{v}^2_{\mathtt{dn}}, ..., \Delta \mathtt{v}^D_{\mathtt{dn}}), \tag{E.8}$$

avec:

$$\Delta \mathtt{v}_{\mathtt{dn}} = \max(\overline{\mathtt{v}_{\mathtt{dn}}}) - \min(\overline{\mathtt{v}_{\mathtt{dn}}}), \tag{E.9}$$

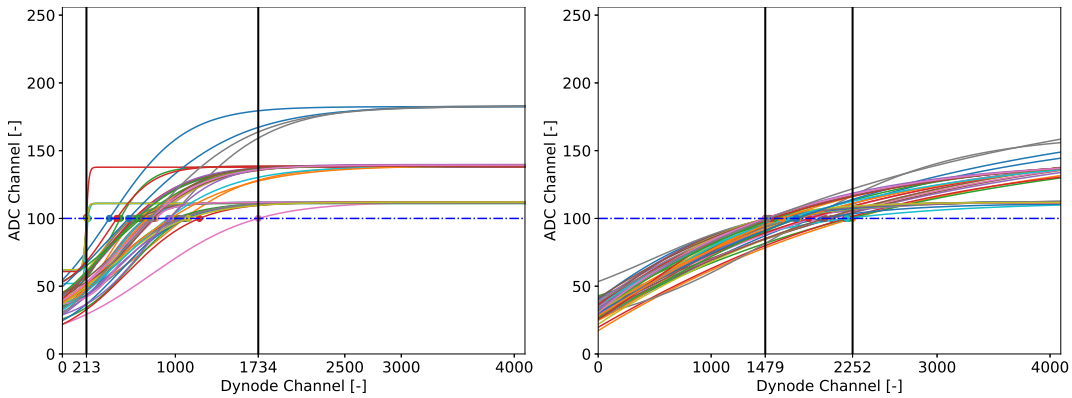
où $\overline{\mathtt{v}_{\mathtt{dn}}}$ est l'ensemble des tensions de dynode P de chaque PMT dans la chambre. Cette métrique a été sélectionnée car la moins bonne configuration se produit lorsqu'un PMT est limité par une valeur de dynode basse et un autre est limité par une valeur de dynode élevée.

E.3.7 Mutation

Le processus de mutation peut se produire de deux manières différentes. La première est la plus probable, avec une probabilité de 10% pour chaque caméra de scintillation constituant un individu. Si une chambre est sélectionnée pour être mutée, l'un de ses PMT est sélectionné au hasard, ensuite une autre caméra du même individu est sélectionnée et un autre PMT de cette chambre est sélectionné au hasard. Ces deux PMT sont échangés et les deux caméras sont reconfigurées à l'aide de tous les PMT appartenant à chacun de leurs groupes. S'il y a des PMT de rechange, ils sont pris en compte dans les deux processus, à la fois pour l'échange et pour la reconfiguration. Le deuxième type de mutation est beaucoup moins probable (avec une probabilité inférieure à 1%), et modifie la valeur HV de la caméra. Si une caméra est choisie, sa nouvelle valeur HV est sélectionnée au hasard parmi les valeurs d'étalonnage.

E.3.8 Résultats et discussion

Le processus décrit ici a été utilisé pour générer $D = 6$ chambres de scintillation pour l'AR-PET, chacune constituée de $P = 48$ PMT. Pour la construction des chambres, un total de $P' = 307$ PMT était disponible. En utilisant l'équation E.6, le nombre total de configurations possibles peut être calculé, et il dépasse $6 \times 10^{+614}$. Les caméras construites ont été comparées à un prototype de caméra existant. Le prototype de caméra a été créé en choisissant manuellement les PMT parmi les P' PMT totaux, en attribuant ceux avec un gain inférieur aux coins, où ils recevaient plus de lumière, ceux de gain moyen aux bords et enfin ceux à gain élevé au centre. Ce processus a été réalisé sans prendre en compte les cinq caméras restantes, car il s'agissait d'un prototype. Le canal objectif ADC pour l'énergie de crête du césium-137 ($^{137}_{55}$Cs) était



(a) (b)

Figure E.7: Courbes PMT du prototype de caméra à scintillation, créées par sélection manuelle des PMT (a) et la caméra numéro 4 des caméras générées (b). La ligne pointillée horizontale représente le canal ADC objectif, le point coloré à l'intersection de la courbe avec la ligne représente le point de travail du PMT. L'espace entre les lignes noires verticales représente l'écart maximal entre les points de travail, illustrant la valeur de fonction de forme.

de 100 (dans une plage de $[0, 255]$). La valeur HV a été sélectionnée dans 1200 `V`. La caméra résultante avait une valeur de fonction de forme de 1521, calculée avec l'équation E.9. La figure E.7(a) montre les courbes interpolées pour chacun des PMT choisis dans l'HV correspondant à la caméra prototype. Ces courbes sont corrigées pour les facteurs de gain $\chi_{\texttt{cal}}$. Les caméras AR-PET ont été générées avec le processus décrit, en utilisant une population de $\Xi = 30$ individus, une valeur d'élitisme de 10%, une probabilité de mutation de 6% pour le premier type et de 0.6% pour le deuxième type et il a été traité pour 1000 itérations. Aucun critère de terminaison précoce n'a été appliqué. Les valeurs de fonction de forme des six chambres sont résumées, avec la chambre prototype, dans le tableau E.1. Dans la figure E.7 les courbes pour les PMT de la caméra générée avec la plus mauvaise valeur de fonction de forme de l'individu et la caméra prototype sont affichées. On peut voir que la valeur de fonction de forme, représentée par la largeur entre les lignes noires verticales, est beaucoup plus faible dans la caméra générée.

Le regroupement par k-means avec l'algorithme génétique réduisent considérablement la dispersion des valeurs de la dynode dans la chambre de scintillation. Il a également été possible de trouver des configurations viables pour toutes les caméras requises avec une quantité de PMT en excès inférieure à 7% du total des unités disponibles. Les caméras construites avec cette méthode ont atteint 15% de FWHM (aucun autre réglage fin), améliorant les résultats précédents de 20% [97]. De plus, les caméras construites ont des points de travail dans la zone linéaire de la courbe de tension de dynode-gain des PMT.

Table E.1: Valeurs de fonction de forme de les caméras de scintillation

Caméra	HV [v]	Conformité
1	1200	759
2	1100	771
3	1100	741
4	1300	773
5	1200	757
6	1200	684
Prototype	1200	1521

E.4 Algorithmes de localisation d'événements gamma

La plupart des tomographes ont leur résolution de localisation des événements gamma sur leurs détecteurs fixée par la taille de leur pixels. Un pixel est normalement autour de 4 mm et va jusqu'à 3.2 mm dans les systèmes modernes [98]. Les systèmes commerciaux en général ne fournissent pas d'informations sur la profondeur de l'interaction des événements (DoI), il est donc prévu que les cristaux soient aussi minces que possible. Comme le scanner AR-PET utilise des cristaux monolithiques et de grandes tailles, la localisation des événements doit être effectuée au moyen des valeurs d'intensité détectées sur les PMT lorsqu'une interaction d'un photon gamma se produit. La sortie du $p^{éme}$ PMT est une valeur proportionnelle au total des photons détectés $e_p \approx \zeta_p$. La position est alors dérivée mathématiquement du vecteur d'énergie acquis $\overline{e}$. Cela peut être fait par diverses méthodes. La méthode standard pour positionner les interactions est l'algorithme du centre de gravité (CoG) ou Anger [218], qui fournit une méthode simple pour localiser l'interaction sur la surface du cristal. Cependant, l'algorithme CoG présente deux inconvénients majeurs: Premièrement, il ne fournit pas d'informations DoI qui sont essentielles pour les scanners à grand champ de vision axial (FoV) ou lorsque la largeur du cristal est comparable au FoV (comme c'est le cas avec les scanners PET pour petits animaux). Deuxièmement, le positionnement de l'interaction est fortement déformé sur les bords du détecteur en raison de la réflexion de la lumière ou des pertes dans les bords du cristal de scintillation et du manque d'informations d'un autre PMT ou SiPM voisin. Les limites de l'algorithme CoG ont conduit à proposer différentes méthodes pour améliorer la localisation des interactions gamma. L'une de ces méthodes consiste à utiliser des tables de recherche (LuT) construites à l'aide d'une série de mesures, puis chaque nouvel événement est comparé aux événements existants dans le tableau et la position de l'événement le plus proche dans le tableau est obtenue. Plusieurs techniques ont été proposées pour effectuer la recherche et estimer la position de l'événement, comme le maximum de vraisemblance [38], k-voisins les plus proches [219] et l'erreur quadratique moyenne [40]. En plus des LuT, des méthodes statistiques ont été proposées, telles que l'amélioration du gradient de décision [99], qui forme un modèle de positionnement statistique basé sur une mesure d'étalonnage rapide effectuée par un éventail de rayons. L'utilisation

de réseaux de neurones (NN) a également été proposée. Bruyndonckx [100] et Venialgo [101] proposent de former un NN à l'aide d'un ensemble des données construites à partir de mesures empiriques, telles que les rayons collimatés. Conde [37] et Ibora [102] proposent de former le NN avec un jeu des données générées à partir d'un modèle mathématique, pour réduire le temps et la complexité de l'étalonnage. Toutes ces méthodes sont basées sur la génération d'un ensemble des données contenant les valeurs d'énergies attendues pour chaque position d'impact sur le cristal. Un ensemble des données présentant ces caractéristiques n'est pas facile à construire et constitue l'un des plus grands obstacles à la mise en œuvre de ces techniques.

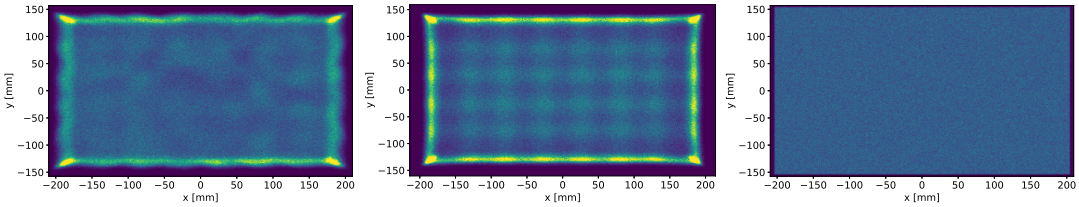
E.5 Localisation d'événements à l'aide d'un réseau neuronal correspondant à des distributions

Il est proposé ici une méthode pour corriger la distorsion de l'algorithme CoG et obtenir une valeur DoI en utilisant les NN et une simple acquisition de champ dont la seule exigence est de connaître la distribution des événements attendus. Le procédé ne nécessite aucun exploration de surface avec des rayons collimatés. Il repose sur la transformation de la distribution d'événements obtenus par l'algorithme CoG (ou autre) en la distribution attendue, corrigeant la position d'un seule événement à la fois. La méthode est même capable d'estimer le DoI suivant une stratégie de diffusion de la lumière. De plus, la méthode améliore le FWHM de l'algorithme CoG. La méthode a été testée à l'aide d'une acquisition d'un module de détection AR-PET et d'une simulation Monte Carlo. La méthode est comparée à la méthode CoG (ou Anger) et à une estimation de DoI basée sur la diffusion de la lumière, en utilisant des acquisitions de rayons collimatés.

E.5.1 Introduction

La distribution des interactions de photons gamma dans le cristal dépend de l'énergie, de l'angle d'incidence, de la position de la source et de la puissance d'arrêt du cristal. Cette distribution peut être observée empiriquement en appliquant un histogramme 3D à un grand nombre d'interactions. Un algorithme de positionnement parfait récupérerait la distribution d'interaction théorique dans le cristal, cependant, un algorithme de positionnement réel générera une distribution différente. Par exemple, une source ponctuelle positionnée loin du centre du détecteur produira une distribution uniforme des interactions, sur toute la surface du détecteur. Si ces interactions sont positionnées avec l'algorithme Anger, qui souffre d'une distorsion sur les bords du cristal, l'histogramme résultant montrera les effets de l'accumulation d'événements sur les bords du détecteur, comme indiqué sur la figure E.8. L'effet d'accumulation observé aux limites du détecteur conduit à devoir ignorer une partie de la zone sensible, car la localisation n'était pas fiable. Cette zone a la taille d'un PMT et s'appelle *zone morte* (où *dead zone* en anglais).

Cette différence entre la distribution attendue et estimée peut être utilisée comme une métrique de l'effet d'accumulation, ou distorsion, de l'algorithme de position-



(a) Experimental (b) GATE (c) Expected

Figure E.8: Histogramme 2D de la distribution empirique (a) et simulée (b) de la méthode CoG, et la distribution attendue (c), dans le plan (x, y) de la caméra de détection AR-PET. Les inhomogénéités en (a) et (b) sont causées par les erreurs de positionnement de l'algorithme CoG.

nement. Plus formellement, si la distribution attendue $\boldsymbol{R}$ et la distribution de positionnement estimée $\hat{\boldsymbol{R}}$ sont connues, alors une métrique de la distance entre elles peut être obtenue:

$$d = J_{\texttt{dist}}(\boldsymbol{R}, \hat{\boldsymbol{R}}), \tag{E.10}$$

où $J_{\texttt{dist}}()$ est une fonction de la distance. La forme de la fonction de distance doit prendre en compte chaque événement particulier séparément, car la position individuelle de ceux-ci est importante. Ces types de métriques sont normalement utilisés dans le domaine du transport optimal (OT). Ils mesurent cette distance comme le coût de déplacement de chacun des échantillons dans une distribution à la position de l'un des échantillons dans la distribution objectif. Les candidats possibles pour $J_{\texttt{dist}}()$ sont la distance de mouvement terrestre (EMD), également connue sous le nom de métrique de Wasserstein [103] (mis en œuvre pour les problèmes OT à grande échelle dans [104]), ou la métrique de distance de Chanfrein (CDM). La CDM est plus facile à calculer que EMD et fournit des gradients précis qui se traduisent par des performances similaires à EMD pour la formation des NN [105]. Le CDM trouve pour chaque événement du premier ensemble son voisin le plus proche du second ensemble, le carré de la distance entre elles et cumule toutes ces distances. Il peut être exprimé comme:

$$J_{\texttt{dist}}(\boldsymbol{R}, \hat{\boldsymbol{R}}) = \sum_{\boldsymbol{r}_n \in \boldsymbol{R}} \min_{\hat{\boldsymbol{r}}_m \in \hat{\boldsymbol{R}}} \|(\boldsymbol{r}_n - \hat{\boldsymbol{r}_m})\|_2^2 + \sum_{\hat{\boldsymbol{r}}_m \in \hat{\boldsymbol{R}}} \min_{\boldsymbol{r}_n \in \boldsymbol{R}} \|(\boldsymbol{r}_n - \hat{\boldsymbol{r}}_m)\|_2^2, \tag{E.11}$$

où $\boldsymbol{r}$ est la position 3D d'une interaction. En utilisant l'équation E.11 il est possible d'optimiser un NN pour réduire cette distance tant que la distribution objectif $\boldsymbol{R}$ est connue et elle peut être échantillonnée. Par conséquent, minimiser d entraîne une minimisation entre la position attendue et estimée d'un événement.

E.5.2 Génération de l'ensemble des données

L'ensemble des données a été obtenu en utilisant une caméra à scintillation AR-PET. La chambre est composée d'un cristal NaI (TI) continu de $406.4 \times 304.8 \times 25.4$ $\texttt{mm}^3$ avec un guide de lumière de 9.5 $\texttt{mm}$, peuplé d'une formation de 8×6 PMT R1534 de

Hamamatsu. Ces PMT ont un profil rectangulaire de 52.4×52.4 $\texttt{mm}^2$ qui recouvrent complètement la surface du verre. Bien que le procédé ne nécessite aucune distribution objectif particulière, pour simplifier, une distribution uniforme et orthogonale est utilisée. Le champ d'événements est généré à l'aide d'une source de sodium $^{22}\texttt{Na}$ β^+ (qui génère des photons 511 $\texttt{keV}$ par décomposition électron-positon) situé à une longue distance $d_s > 3$ $\texttt{m}$ et aligné avec le centre de la surface de détection du détecteur. Compte tenu de la distance de la source à la surface du détecteur, les photons peuvent être considérés comme perpendiculaires à la surface du cristal. En utilisant cette configuration, $N = 3 \times 10^7$ interactions sont obtenues. Ces événements ont été filtrés par une fenêtre énergétique entre 470 $\texttt{keV}$ et 530 $\texttt{keV}$. Chaque interaction n est stockée comme la distribution d'énergie $e_p^n \in \overline{e}_n$ détectée dans chaque détecteur p de la caméra de scintillation. L'ensemble des données de base $\tilde{\boldsymbol{R}}$ est ensuite construit en calculant la position d'interaction estimée pour l'événement $\tilde{\boldsymbol{r}}_n = (r_x^n, r_y^n, r_z^n)$. La position dans le surface (r_x^n, r_y^n) est calculée à l'aide de l'algorithme CoG. Le DoI (r_z^n) est obtenu à partir d'une estimation simple basée sur la diffusion de la lumière, un indicateur bien connu du DoI [106]. Cette métrique est construite en utilisant les informations de la distribution d'énergie dans l'ensemble des données. Tout d'abord, la surface du détecteur est divisée en D pixels de 12×12 $\texttt{mm}^2$. Ensuite, en utilisant le positionnement du CoG, l'ensemble des données est divisé en sous-groupes, la relation entre l'énergie détectée dans les PMT 4 avec une énergie plus élevée et l'énergie totale est calculée:

$$\eta_n^k = \frac{\sum_{p=1}^{4} e_p^{n*}}{\sum_{p=1}^{48} e_p^{n*}}, \tag{E.12}$$

où $\overline{e}_n^*$ sont les quatre valeurs d'énergie les plus élevées pour le $n^{\text{ème}}$ événement, k marque le pixel correspondant et p le numéro de PMT. Pour chaque pixel k, la distribution des valeurs η^k est ajustée en utilisant une distribution normale. Enfin, une carte de $\eta_{\texttt{min}}$ minimums et $\eta_{\texttt{max}}$ maximums en tant que valeurs η^k est construite en utilisant l'intervalle 2σ des distributions ajustées. Connaissant la dispersion maximale et minimale par pixel, le DoI d'un événement n interagissant dans le pixel $k^{\text{ème}}$ peut être estimé linéairement:

$$r_z^n = D_{\text{thkns}} \frac{\eta_n - \eta_{\texttt{max}}^k}{\eta_{\texttt{min}}^k - \eta_{\texttt{max}}^k}, \tag{E.13}$$

où D_{thkns} est la largeur du détecteur. Un ensemble des données générés avec une simulation Monte Carlo de l'expérience a été fait avec le logiciel GATE. La simulation comprenait l'analyse des photons optiques et de tous les éléments réfléchissants de la caméra à scintillation. L'ensemble des données simulées a recueilli à peu près le même nombre d'événements que l'ensemble des données empiriques.

E.5.3 Échantillonnage de la distribution objectif

La distribution objectif est obtenue grâce à une simulation Monte Carlo de la configuration expérimentale dans le logiciel GATE. En utilisant cette simulation, la fonction de distribution cumulative (CDF) est obtenue pour chaque axe. Il est à noter que cette simulation ne nécessite pas le suivi de photons optimaux. Compte tenu des conditions

expérimentales, les interactions sur les axes (X, Y) suivent respectivement les distributions $U(-203.0, 203.0)$ et $U(-152.0, 152.0)$. L'axe Z suit la distribution attendue pour un cristal NaI(TI) [79]. L'axe Z est ensuite échantillonné en utilisant la méthode CDF inverse [108]. Pour y parvenir, le CDF est calculé, puis son inverse est approché à l'aide d'un ajustement polynomial. La fonction polynomiale obtenue est ensuite utilisée pour convertir les échantillons obtenus à partir d'une distribution $U(0, 1)$ en la distribution objectif.

E.5.4 Réseau neuronal

Afin d'ajuster la fonction de localisation des interactions, un NN totalement connecté a été utilisé. L'entrée du NN est le vecteur d'entrée $\overline{e}_n$, composé des 48 valeurs d'énergie. Le NN a une couche d'entrée de 64 neurones, suivie de trois couches cachées de 32, 16 et 8 neurones respectivement, tous *redresseurs* (ReLu). La couche de sortie comporte 3 neurones, avec une activation linéaire et des sorties correspondant à chacun des axes (x, y, z). L'apprentissage du réseau de neurones a deux étapes, d'abord il apprend le positionnement de la méthode de base (comme le CoG) en utilisant une fonction objectif simple, telle que la distance euclidienne:

$$J_{\texttt{base}}(\tilde{\boldsymbol{R}}, \hat{\boldsymbol{R}}) = \frac{1}{N} \sum_{n=1}^{N} \|\tilde{\boldsymbol{r}}_n - \hat{\boldsymbol{r}}_n\|_2^2, \tag{E.14}$$

où N est le nombre d'interactions dans l'ensemble des données et $\tilde{\boldsymbol{r}}_n \in \tilde{\boldsymbol{R}}$ est la position de l'interaction obtenue avec la méthode de base. Cette apprentissage se fait avec l'algorithme Adaptive Moment Estimation (ADAM) [109], avec une taux d'apprentissage de $\epsilon = 0.001$. Le réseau est formé par des 500 époques et avec une taille de lot de $N' = 5 \times 10^3$ interactions. Une fois l'apprentissage terminé, le NN est réappris pour corriger les erreurs de position en incluant une fonction de perte de correspondance, présentée dans E.11:

$$J_{\texttt{match}} = K\, J^*_{\texttt{base}}(\tilde{\boldsymbol{R}}, \hat{\boldsymbol{R}}) + J_{\texttt{dist}}(\boldsymbol{R}, \hat{\boldsymbol{R}}), \tag{E.15}$$

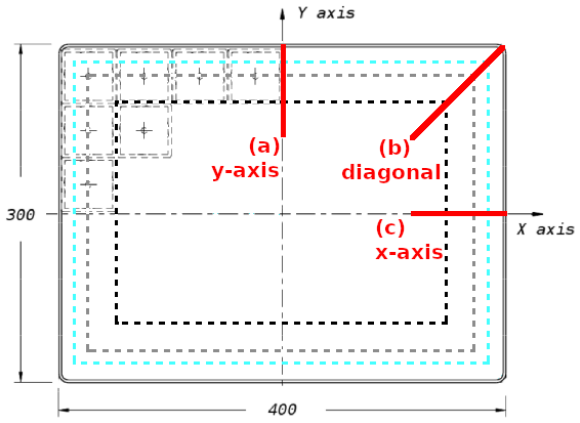
où $K = 0.001$ est une constante de couplage. Encore une fois, le NN est appris à l'aide de l'algorithme ADAM par une période de 500 époques.

E.5.5 Résultats

La méthode proposée a été testée dans deux NN différents. Le premier est pré-formé en utilisant l'ensemble des données expérimentales et l'algorithme CoG comme méthode de base, puis affiné en utilisant la perte par appariement des distributions proposées pour corriger les effets d'accumulation dans l'algorithme CoG, ce NN a été appelé *ANGER_FCN*. Le second suit les lignes directrices de la *ANGER_FCN* mais utilise l'ensemble des données artificielles, ce NN est appelé *Artf_ANGER_FCN* et il a été étudié séparément.

Performance 2D: rayons collimatés perpendiculaires

En utilisation de la technique décrite dans [101], trois séries de rayons collimatés ont été acquises: l'**axe y** (**y axis**) dont l'origine est $(0.0, 72.15)$ `mm` et s'étend jusqu'à $(0.00, 144.38)$ `mm`, la **Diagonale** qui commence en $(123.0, 72.15)$ et s'étend à $(194.25, 144.38)$ et l'**axe x** (**x axis**) dont l'origine est $(123.0, 0.0)$ `mm` et s'étend jusqu'à $(194.25, 0.00)$ `mm`. Toutes les séries ont une étape de 26.2 `mm`. L'emplacement de chaque série peut être vu sur la figure E.9. Les courbes de performance pour la méthode CoG (Anger) et pour les *ANGER_FCN*, pour chaque série, sont présentées dans la figure E.10. La valeur moyenne de FWHM, la tendance moyenne et la tendance maximale pour chaque estimateur de position d'interaction, de séries de rayons collimatés et de zone morte du détecteur sont indiquées dans le tableau E.2.



(a)

Figure E.9: Positions du collimateur dans la surface (x, y) du détecteur. La série "y-axis" (a), la série "Diagonal" (b) et la série "x-axis" (c). Les lignes pointillées noires, grises et cyan représentent une marge d'un, un demi et un quart de PMT a partir du bord du détecteur respectivement.

Performance 2D: test GATE

En utilisant les informations disponibles à partir de la simulation Monte Carlo, l'Artf_ANGER_FCN a été testée par rapport aux valeurs vraies et à l'algorithme d'Anger. Les régions testées étaient les mêmes que la série de rayons collimatés. Les séries simulées ont une étape 2.5 `mm`. Les performances de l'Artf_ANGER_FCN et de la méthode Anger sont illustrées dans la figure E.11. La valeur moyenne de FWHM, la tendance moyenne et la tendance maximale pour chaque estimateur de position d'interaction de séries de rayons collimatés et zone morte du détecteur sont indiquées dans le tableau E.3.

Performances en profondeur d'interaction: rayons collimatés diagonaux

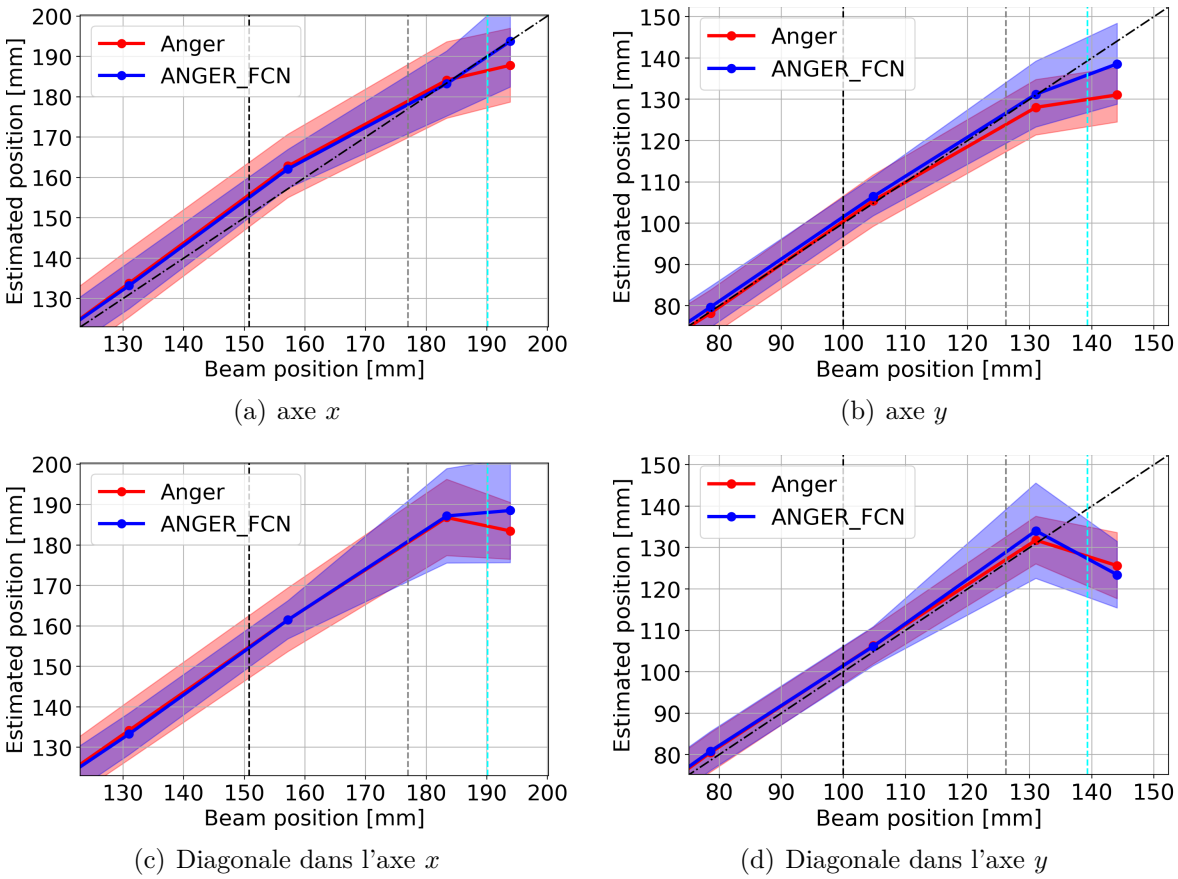
Afin de tester les performances de l'estimation DoI du NN, un faisceau collimaté avec un angle d'incidence de 45° a été mesuré dans la zone centrale de la surface de détection.

Série	Estimateur	Taille de la zone morte								
		52.4 mm			26.2 mm			13.1 mm		
		$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$
	Anger	12.08	2.74	3.72	12.20	3.15	5.63	12.64	2.85	5.63
x axis	ANGER_FCN	10.38	2.73	3.80	10.23	3.11	5.37	10.80	2.78	5.37
	Anger	10.38	1.44	3.22	10.67	1.24	3.22	11.04	1.58	3.22
y axis	ANGER_FCN	10.50	2.10	3.34	10.47	1.98	3.34	11.06	1.73	3.34
	Anger	13.41	2.76	3.21	13.96	3.15	4.30	14.91	3.19	4.30
Diagonal - eje x	ANGER_FCN	10.06	2.39	2.80	9.88	2.75	3.82	11.62	3.05	4.26
	Anger	10.25	1.13	1.89	9.88	1.21	1.89	10.19	1.13	1.89
Diagonal - eje y	ANGER_FCN	9.82	1.48	2.27	9.70	1.22	2.27	12.58	1.81	4.15

Table E.2: Mesure empirique: FWHM moyen ($\overline{\texttt{FWHM}}$), tendance moyen ($\bar{b}$) et tendance maximal ($\max b$), pour chacune des séries de rayons et chaque position estimée. Les métriques ont été testées en utilisant des zones mortes du détecteur 1, 0.5 et 0.25 taille d'un PMT (52.4 mm, 26.2 mm, 13.1 mm respectivement). Toutes les métriques sont exprimées en [mm].

Série	Estimateur	Taille de la zone morte								
		52.4 mm			26.2 mm			13.1 mm		
		$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$	$\overline{\texttt{FWHM}}$	$\bar{b}$	$\max b$
	Anger	6.00	0.88	1.46	6.99	3.32	8.31	7.37	3.32	8.31
x axis	Artf_ANGER_FCN	5.46	0.89	3.17	6.72	2.54	5.96	8.25	2.84	5.96
	Anger	5.81	0.80	1.30	6.64	3.05	7.31	7.02	3.04	7.31
y axis	Artf_ANGER_FCN	5.49	1.25	2.78	6.50	2.05	4.18	9.15	2.87	8.11
	Anger	6.10	0.91	1.93	7.01	3.92	9.87	7.43	3.88	9.87
Diagonal - eje x	Artf_ANGER_FCN	5.16	1.05	3.64	6.25	3.42	7.84	8.63	4.04	7.84
	Anger	5.94	0.89	1.55	6.70	3.65	8.87	7.02	3.58	8.87
Diagonal - eje y	Artf_ANGER_FCN	5.20	0.92	1.55	6.69	2.49	7.73	9.26	3.62	8.68

Table E.3: Simulation dans GATE: FWHM moyen ($\overline{\texttt{FWHM}}$), tendance moyen ($\bar{b}$) et tendance maximal ($\max b$), pour chacune des séries de rayons et chaque position estimée. Les métriques ont été testées en utilisant des zones mortes du détecteur 1, 0.5 et 0.25 taille d'un PMT (52.4 mm, 26.2 mm, 13.1 mm respectivement). Toutes les métriques sont exprimées en [mm].



(a) axe x

(b) axe y

(c) Diagonale dans l'axe x

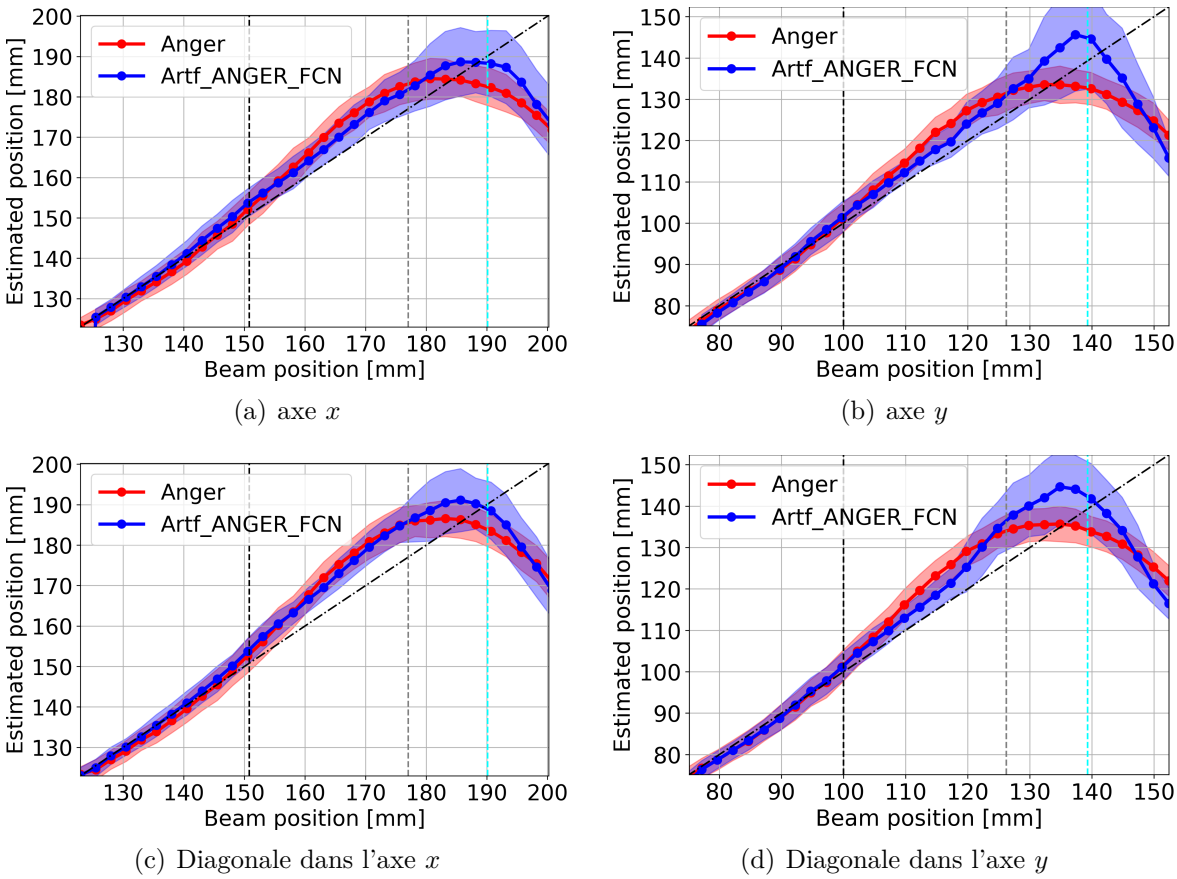
(d) Diagonale dans l'axe y

Figure E.10: Mesure empirique: Performances dans la série "x axis" (a), la série "y axis" (b), la série "Diagonal" sur l'axe x (c) et la série "Diagonal" sur l'axe y (d). La ligne continue représente la position médiane et la zone colorée représente le FWHM. Les lignes pointillées noires, grises et cyan représentent respectivement une marge d'un, un demi et un quart de PMT a partir du bord du détecteur respectivement.

Le rayon a été placé en position $(0.5, 0.5)$ mm et son incidence a suivi l'axe x. Compte tenu de l'angle d'incidence, le DoI suit la déviation de la position sur l'axe x par rapport au point d'impact. C'est-à-dire, $r_n^z = r_n^x - r^x_{\texttt{centredeimpacte}}$. Le DoI a été divisé en 6 sections de 4.23 mm et la FWHM moyenne pour chaque section a été calculée. Le facteur de corrélation entre le DoI estimé et attendu a été calculé, résultant en $\texttt{Corr}^{\texttt{Disp}}_{\texttt{centre}} = 0.28$ et $\texttt{Corr}^{\texttt{ANGER\_FCN}}_{\texttt{centre}} = 0.21$.

Performances en profondeur d'interaction: test GATE

L'Artf_ANGER_FCN et l'estimation DoI par la méthode basée sur la diffusion de la lumière ont été testées à l'aide d'une simulation dans GATE. Deux régions ont été testées: Une région centrale de la surface du détecteur, centrée sur $(0.0, 0.0)$ mm et avec une aire de 50.0×50.0 mm^2 et puis une région de bord centrée sur $(0.0, -133.75)$ mm et avec une aire de 50.0×50.0 mm^2. La valeur de référence du DoI a été obtenue à



(a) axe x (b) axe y

(c) Diagonale dans l'axe x (d) Diagonale dans l'axe y

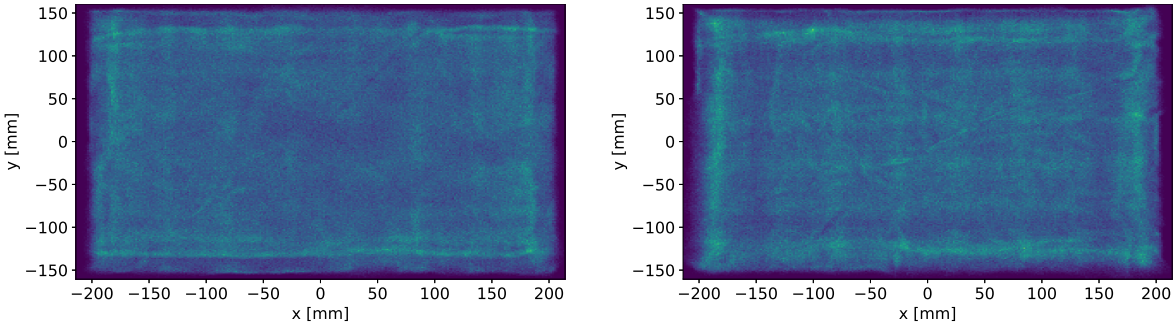
Figure E.11: Simulation dans GATE: Performances dans la série "x axis" (a), la série "y axis" (b), la série "Diagonal" sur l'axe x (c) et la série "Diagonal" sur l'axe y (d). La ligne continue représente la position médiane et la zone colorée représente le FWHM. Les lignes pointillées noires, grises et cyan représentent respectivement une marge 1, 0.5 et 0.25 taille d'un PMT a partir du bord du détecteur respectivement.

partir des données de simulation. Le facteur de corrélation entre le DoI estimé et attendu a été calculé, résultant en $\texttt{Corr}_{\texttt{center}}^{\texttt{Artf\_ANGER\_FCN}} = 0.80$, $\texttt{Corr}_{\texttt{center}}^{\texttt{ANGER GATE}} = 0.79$, $\texttt{Corr}_{\texttt{edge}}^{\texttt{Artf\_ANGER\_FCN}} = 0.53$ y $\texttt{Corr}_{\texttt{edge}}^{\texttt{ANGER GATE}} = 0.50$.

Analyse de la distribution

L'empilement des distributions résultantes est analysé dans le plan (x, y) à l'aide d'un histogramme 2D des événements dans l'ensemble des données. Pour chaque méthode de localisation, un histogramme 2D avec une taille de boîte de 1.56×1.56 $\texttt{mm}^2$ a été calculé. Les histogrammes résultants (x, y) peuvent être vus sur la figure E.12.

Pour chaque méthode analysée, la densité empirique le long des axes x et y a été calculée en utilisant l'histogramme sur chaque axe. L'irrégularité de la distribution a été mesurée à l'aide de la différence de comptes normalisée $\Delta C = (C_{\texttt{max}} - C_{\texttt{min}})/C_{\texttt{max}}$, où $C_{\texttt{max}}$ et $C_{\texttt{min}}$ sont les valeurs de comptage maximum et minimum dans les cases de



(a) COG_NN (x, y) histogram (b) Artf_COG_NN (x, y) histogram

Figure E.12: Histogrammes 2D de l'ANGER_FCN (a) et l'Artf_ANGER_FCN (b).

l'histogramme. La métrique a été appliquée dans l'estimation de la densité empirique sur les axes x et y. Pour chaque axe, trois mesures ont été calculées, une avec une marge de 52.4 mm, une outre avec une marge de 26.2 mm et enfin un avec une marge de 13.1 mm (respectivement 1, 0.5 et 0.25 taille d'un PMT). Ces résultats ont été résumés dans le tableau E.4

zone morte	52.4 mm		26.2 mm		13.1 mm	
	axe x	axe y	axe x	axe y	axe x	axe y
Anger	0.17	0.13	0.38	0.58	0.47	0.83
ANGER_FCN	0.17	0.12	0.24	0.24	0.29	0.39
Artf_Anger (GATE)	0.34	0.34	0.54	0.67	0.73	0.89
Artf_ANGER_FCN	0.29	0.27	0.31	0.34	0.42	0.35

Table E.4: Différence normalisée de comptage pour chaque axe de la densité empirique. Chaque métrique a été testée en utilisant 1, 0.5 et 0.25 taille de un PMT (52.4 mm, 26.2 mm, 13.1 mm respectivement), en représentant trois limites de zone morte différentes sur le détecteur. Une valeur baise indique une plus grande uniformité.

Discussion

Dans ce travail, une technique de positionnement des interactions photons gamma basée sur les NN a été présentée. La méthode ne nécessite qu'une acquisition de champ et une connaissance de la distribution attendue des interactions dans le volume du scintillateur. Les performances 2D de la technique proposée ont été testées à l'aide d'une série de rayons collimatés, montrant une amélioration des valeurs de tendance et de la moyenne FWHM par rapport à l'algorithme CoG (Anger). Cette amélioration est évaluée lorsque la zone morte est réduite à une moitié de PMT (26.2 mm). Une amélioration par rapport à la FWHM est observée dans la série de rayons collimatés "x-axis" et "Diagonal", représentés dans les tableaux E.2 et E.3.

La capacité de la technique proposée à estimer la DoI a été testée à l'aide d'un faisceau collimaté avec un angle d'incidence de 45°. Le NN a atteint un facteur de corrélation de $\texttt{Corr}_{\texttt{centro}}^{\texttt{ANGER\_FCN}} = 0.21$ contre $\texttt{Corr}_{\texttt{centro}}^{\texttt{Disp}} = 0.28$ pour la méthode de base. Dans le cas de la simulation GATE, la méthode NN atteint $\texttt{Corr}_{\texttt{center}}^{\texttt{Artf\_ANGER\_FCN}} = 0.80$ contre $\texttt{Corr}_{\texttt{center}}^{\texttt{ANGER GATE}} = 0.79$ de la méthode de base. Une amélioration directe de l'estimation du DoI n'a pas été observée, cependant la taille relativement petite du NN a pu reproduire la méthode d'estimation du DoI de base. Ceci est important car la méthode de base nécessite l'utilisation de LuT pour produire le DoI, alors que le NN est composé uniquement de multiplications et d'additions. La mise en œuvre de meilleurs algorithmes pour estimer la DoI permettra aux NN de réduire la FWHM observée dans la mesure expérimentale, puisque la capacité du réseau est suffisante pour produire une meilleure estimation de la DoI.

L'effet d'empilement de l'algorithme CoG était très réduit. Comme on le voit dans les histogrammes présentés dans les figures E.12(a) et E.12(b). La surface présente moins d'effets de bord par rapport à ses références sur les figures E.8(a) et E.8(b). La méthode proposée a repositionné les interactions "empilées" à leur emplacement le plus proche, suivant un schéma de transport optimal. Une mesure de différence de compte a été calculée pour refléter cet effet et est présentée dans le tableau E.4. Le tableau montre une forte réduction de la variabilité des comptes dans l'acquisition du champ.

L'amélioration de le forme et la réduction de la différence de comptage permettent de réduire la zone morte du détecteur de 52.4 $\texttt{mm}$ a 26.2 $\texttt{mm}$, en augmentant la surface effective des détecteurs de 48.7% à 72.1% de la surface du cristal de scintillation si un maximum ΔC de 25% est toléré. Cela se traduit en une augmentation de 50% de la détection d'événements simples, ça veut dire $0.721^2/0.487^2 = 2.19$ fois plus de probabilité de détecter une correspondance dans un système PET.

La méthode proposée a été testée en utilisant une technique de positionnement de base simple telle que le CoG, mais elle peut être utilisée avec toute autre méthode de positionnement de base. L'amélioration de le forme de la méthode de base se traduit par une amélioration de le forme finale du NN. tant que le NN a une capacité suffisante. La méthode présentée nécessite la mise en œuvre d'une seule acquisition, contrairement à d'autres méthodes qui nécessitent des ensembles des données soigneusement acquis pour former leurs NN. D'autre part, la méthode peut être utilisée pour corriger ou adapter des méthodes basées sur des modèles, telles que celles présentées dans [37] et [102]. Cette adaptation est nécessaire lorsque le modèle théorique ne correspond pas parfaitement au détecteur réel. L'Apprentissage NN peut être amélioré en incluant des limitations sur la fonction objectif. De cette manière, les effets de la statistique faible dans l'ensemble des données d'apprentissage (un faible ratio des $\texttt{impacts}/\texttt{mm}^3$) peuvent être atténués. La taille du NN est également un facteur crucial pour le forme ultime du réseau. Cependant, la mise en œuvre de NN dans les détecteurs PET nécessite des temps de traitement réduits.

E.6 Discussion et commentaires

Dans ce chapitre, l'AR-PET et ses capacités techniques ont été présentés. Les exigences techniques de l'AR-PET ont inspiré les méthodes développées dans ce chapitre. Tout d'abord, à l'aide d'algorithmes génétiques, une méthode de sélection PMT a été développée. Elle est plus efficace que la sélection manuelle de PMT entre eux. De plus, la méthode s'est avérée appropriée pour trouver des configurations de caméras lorsque le nombre de PMT disponibles est très limité. Le deuxième travail s'est concentré sur la localisation des interactions de photons gamma dans le cristal de scintillation. Cette tâche joue un rôle central dans la précision finale du scanner. Étant donné que les caméras du scanner AR-PET sont grandes avec des cristaux solides et de grands PMT, la tâche de localisation n'est pas anodine et de nouvelles techniques ont dû être étudiées. La méthode proposée a tiré parti de la capacité de traitement du signal de l'AR-PET pour mettre en œuvre des algorithmes de positionnement non linéaire. Ces algorithmes sont basés sur des réseaux de neurones et il est possible de les intégrer complètement dans les FPGA. Les caméras AR-PET ne fournissant aucune autre information que les valeurs des PMT situées à la surface du cristal scintillateur, une nouvelle façon de former les réseaux a été conçue. Cette procédure d'apprentissage utilise uniquement les informations d'une seule acquisition de champ pour obtenir toutes les données d'apprentissage et essaye ensuite de faire correspondre la distribution des événements dans le volume du cristal à la distribution prévue des événements. La méthode a été testée dans les caméras AR-PET, transformant la distribution CoG (Anger) en la distribution attendue, améliorant la zone sensible de la caméra et lui donnant la possibilité d'estimer la DoI à un faible coût de calcul. Cependant, la méthode est valide pour correspondre à toute autre méthode de positionnement initial dans tout autre modèle de caméra, tant que la position des PMT/SiPM, leurs énergies, la distribution cible et une méthode de positionnement initial, sont connues.

Les chapitres suivants se concentrent sur les méthodes d'imagerie PET. Dans ceux-ci, des algorithmes de reconstruction et des méthodes d'analyse d'image sont mis en œuvre pour permettre une étude PET sans avoir besoin d'études supplémentaires pour obtenir l'image d'atténuation, en empruntant un chemin plus direct vers la réduction de la dose dans une étude PET.

F Représentation d'images tomographiques au moyen de maillages

Dans ce chapitre, le problème de la reconstruction d'images au moyen de maillage par éléments finis (FE) est étudié. Tout d'abord, le concept de FE est rapidement introduit et puis la chapitre se concentré sur la reconstruction d'images de tomographie PET sur un maillage.

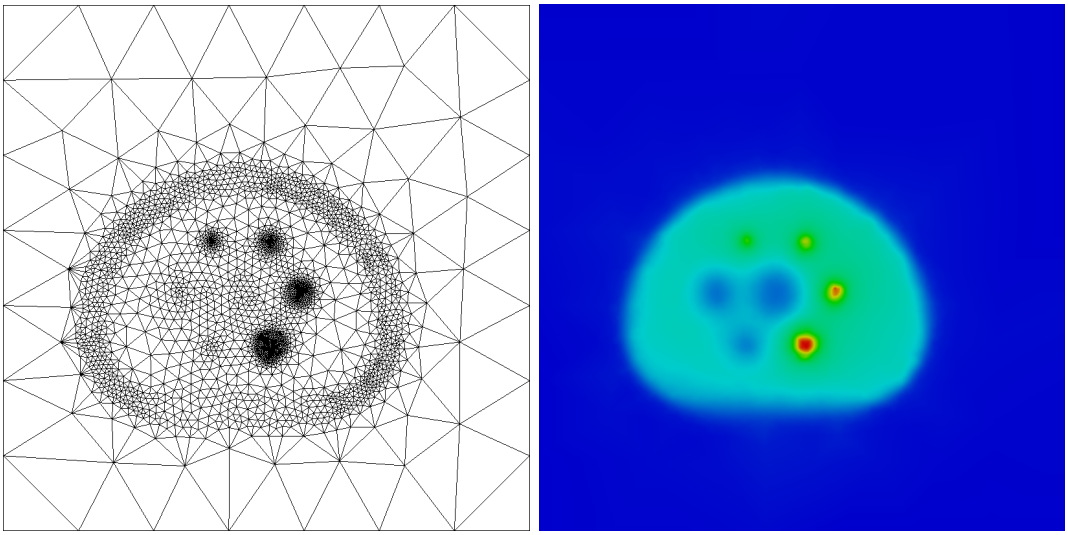
F.1 Introduction aux maillages

Le concept d'utilisation de maillages pour le calcul de FE provient de la discrétisation d'un domaine fermé en des éléments simples. Les FE cherchent à faire une approximation discrète d'un domaine complexe à l'aide de polynômes élémentaires. Ces polynômes ne sont définis que dans de petites régions (éléments) du domaine, où leur approximation est valide, et sont exprimés en termes de valeurs aux nœuds (ou sommets) de ces régions (ou éléments). L'ensemble de tous ces éléments définit le maillage. Contrairement aux représentations pixellisées ou voxélisées, où tous les éléments sont de la même taille et classés par position, un maillage n'a pas de telles restrictions. Un maillage se définit par ses deux caractéristiques principales:

- Géométrie: La position des nœuds (ou sommets) du maillage dans l'espace du domaine.
- Topologie: La liste des connexions des sommets, qui décrivent chaque élément.

Avec ces caractéristiques (géométrie et topologie), l'échantillonnage d'une image peut être adapté pour décrire plus efficacement son support. La figure F.1 montre la représentation maillée d'un fantôme de qualité d'image de la National Electrical Manufacturers Association (NEMA). On peut voir que le maillage est optimisé pour générer des éléments plus grands dans des sections de l'image où leur activité varie très peu ou de manière linéaire, et des éléments plus petits dans les sections avec des variations plus grandes. De cette manière, la représentation maillée intègre également une partie de l'activité dans sa géométrie.

L'idée d'utiliser un support maillé pour l'imagerie médicale n'est pas nouvelle. Il a été largement étudié pour les simulations hémodynamiques [113], la simulation des systèmes respiratoires [114] et l'enregistrement d'images médicales [115, 116]. Dans le



(a) (b)

Figure F.1: Représentation maillée (a) du fantôme NEMA et de son activité reconstruite (b).

contexte de la reconstruction d'images, les méthodes de maillage n'ont pas encore été adoptées, cependant des applications pour PET et SPECT existent.

F.2 Reconstruction de l'image d'activité PET sur des maillages 2D

F.3 introduction

L'utilisation d'un support maillé pour les processus de reconstruction d'images médicales a été étudiée dans le passé pour la PET [128–131] et SPECT [132]. Ces algorithmes de reconstruction sont basés sur MLEM ou OSEM. Ces études montrent que la projection à travers un support réduit peut être plus rapide que sur un support pixélisé pour la reconstruction 2D [132]. Les travaux de Sitek, Pereira et Boutchko [128, 130, 131] ont également rapporté une qualité d'image similaire ou meilleure lors de l'utilisation d'un support de maillage ou de nuage de points. Dans ces études, la qualité du support est mise en évidence comme un aspect important de la reconstruction, cependant seules des techniques simples de mise à jour de maillage sont appliquées, compte tenu du coût de calcul d'un support optimal. La représentation utilisée dans [132] est fixe et basée sur une image voxélisée initiale, obtenue par rétroprojection filtrée (FBP). Cette image initiale est maillée à l'aide d'un modèle de maillage adapté au contexte [134] basé sur un échantillonnage non uniforme d'une carte de caractéristiques obtenue à partir de la distribution spatiale de la deuxième dérivée directionnelle de l'image initiale. La méthode de maillage contrôle ensuite l'erreur en

contrôlant le nombre de nœuds placés, qu'elle contrôle ensuite en utilisant le principe de longueur de description minimale (MDL) [135]. Dans [129], une représentation de maillage 2D est implémentée qui s'adapte dynamiquement à la distribution du radiotraceur au moyen d'un remaillage. La représentation initiale est un maillage grossier aléatoire qui s'adapte à la distribution du radiotraceur à chaque étape de la reconstruction. Le maillage est guidé par une carte caractéristique basée sur le Hessien de l'image reconstruite et les nœuds sont positionnés à l'aide de l'algorithme de diffusion d'erreur Floyd-Steinberg [136]. Sa méthode de reconstruction et de représentation est faite en 2D, une extension n'est pas donnée pour la reconstruction 3D. La génération de maillages présentée dans [128] est dynamique et basée sur un simple déplacement de points où les nœuds sont agrégés au sein des tétraèdres qui présentent la plus grande variation entre leurs quatre nœuds, c'est une technique simple à faible coût de calcul mais le nombre final de nœuds et la qualité du maillage ne sont pas contrôlés. Dans un travail ultérieur [130], ils proposent de commencer par une grille dense qui est ensuite réduit pour obtenir un support de maillage qui ne perd aucun détail. Dans ce dernier travail, il souligne la nécessité d'améliorer la qualité du maillage de support pour éviter les artefacts en forme de coin et l'instabilité numérique due aux éléments de maillage dégénérés. La méthode présentée ici diffère des précédentes sur deux aspects principaux: premièrement, le support de maillage est basé sur une collection de maillages 2D qui représentent un volume 3D. Cette représentation permet de mettre en œuvre une reconstruction 3D tout en utilisant des techniques de remaillage 2D plus simples et réalisables en parallèle. Deuxièmement, le support de maillage s'adapte dynamiquement à la distribution, en commençant par un maillage grossier sans configuration particulière. L'emplacement et le nombre de nœuds dans le maillage sont déterminés à l'aide d'une stratégie qui contrôle l'erreur d'interpolation et la qualité du maillage. La combinaison de ces techniques produit une représentation compacte et de haute qualité du processus de reconstruction. La précision est régie par l'erreur d'interpolation du support. On montre qu'une telle combinaison permet de réduire le nombre de paramètres de plus d'un ordre de grandeur.

La méthode présentée utilise une représentation en mode liste. Ce mode a été choisi en raison des besoins de l'AR-PET.

F.4 Matériaux, méthodes et théorie

Cette section décrit les opérations de projection et de rétroprojection sur le maillage, suivies des méthodes d'optimisation du maillage et de l'activité. Enfin, l'algorithme de reconstruction complet est présenté ainsi que la configuration expérimentale.

F.4.1 Trace de la ligne de réponse sur un support multi-mailles

F.4.1.1 Opération de projection

Étant donné que l'activité dans un volume inconnu est de distribuée manière continue, elle peut être représentée par une fonction spatiale 3D continue Λ. Cette fonction peut être utilisée pour obtenir le nombre d'événements détectés dans une ligne de réponse

(LoR) en modélisant le nombre d'événements détectés dans une LoR à l'aide d'une distribution de Poisson. Le paramètre de cette distribution est l'activité à travers la trajectoire LoR. Ensuite, le nombre d'événements attendus dans la trajectoire de chaque LoR peut être calculé comme l'intégration de la trajectoire du LoR dans l'estimation actuelle $\Lambda^{(t)}$, comme présenté dans l'équation F.1, où s_l est la différentielle linéaire sur le $l^{ème}$ LoR.

$$\hat{x}_l = \int \Lambda^{(t)}(s_l) ds_l. \tag{F.1}$$

Nous proposons ici l'utilisation d'un modèle de maillage 2D multiple, comme le montre la figure F.2, pour décrire la distribution $\Lambda^{(t)}$. L'activité est échantillonnée en utilisant un total de coupes axiales N_s où les maillages sont générés. L'activité le long de l'axe z de l'une des tranches est considérée comme constante (comme dans la représentation voxélisée), ensuite l'activité dans une tranche ne dépend que des coordonnées (x, y). Dans ce schéma de représentation, l'élément résultant est un prisme triangulaire (ou *grand* triangle de largeur Δz) qui peut être traité comme un élément triangulaire pour l'interpolation dans une tranche.

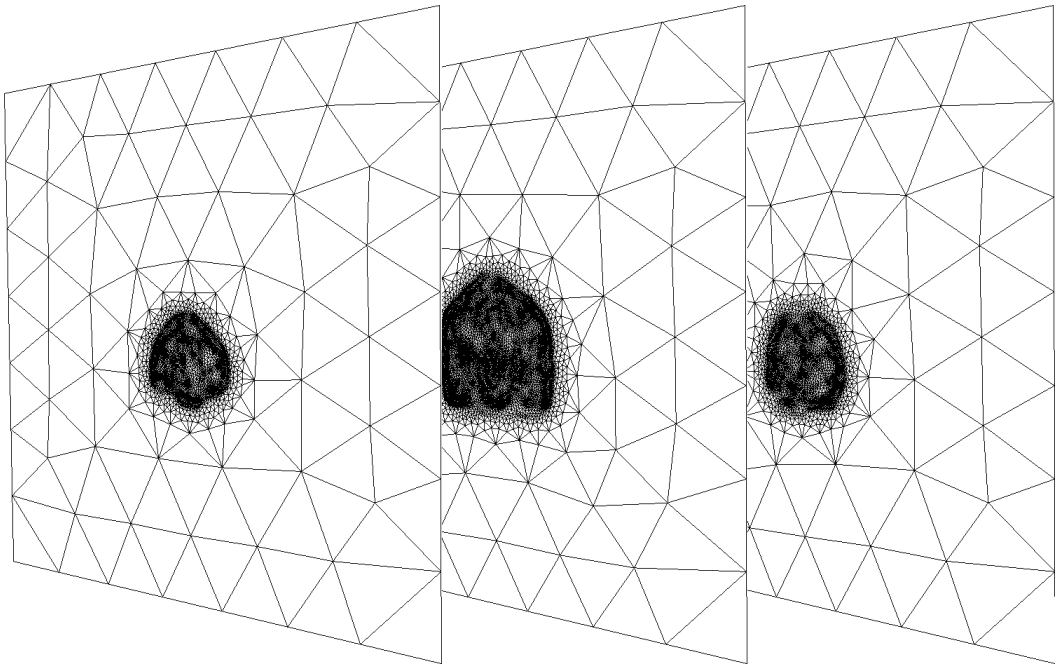


Figure F.2: Série de maillages 2D représentant un volume. Chaque maillage représente une partie de l'axe axial.

L'activité dans une tranche peut être présentée comme l'interpolation bilinéaire des valeurs d'activité sur chacun des nœuds de l'élément courant. Chaque maillage applique alors une fonction de forme Φ pour approcher la valeur de $\Lambda^{(t)}$ à n'importe quelle position. La forme de la fonction peut être décrite à partir de la valeur aux coordonnées d'un point sur un plan (un plan décrit par trois points coplanaires). Les coordonnées inconnues d'un LoR 3D sont définies à l'aide de la notation par ordre

d'origine $\boldsymbol{r}(a) = \boldsymbol{r}^P.a + \boldsymbol{r}^O$. Les points $\boldsymbol{r}^O = [r_x^O, r_y^O, r_z^O]$ et $\boldsymbol{r}^P = [r_x^P, r_y^P, r_z^P]$ sont les coordonnées tridimensionnelles de l'ordonnée à l'origine et à la pente, respectivement, de la LoR. Cependant, dans un maillage 2D, la coordonnée de l'axe z est ignorée car l'activité le long de cet axe est constante dans le maillage. Ainsi, le LoR dans un maillage est défini par $\boldsymbol{r}^O = [r_x^O, r_y^O]$ et $\boldsymbol{r}^P = [r_x^P, r_y^P]$. Chaque sommet $\boldsymbol{r}^{v,e}$ (avec $v \in [1, 2, 3]$) pour chaque élément e du maillage est décrit par sa position et l'estimation de son activité $\boldsymbol{r}^v = [r_x^{v,e}, r_y^{v,e}, \lambda_{v,e}]$. L'activité à la position du nœud est déterminée par la fonction d'activité volumétrique $\lambda_v^e = \Lambda^{(t)}(r_x^{v,e}, r_y^{v,e}, z_s)$, où z_s est la position de la coupe dans l'axe z du FoV. En utilisant cette notation, la fonction de forme est obtenue par dérivation du point et représentation normale d'un plan décrit par trois points: $\boldsymbol{r}^{0,e}$, $\boldsymbol{r}^{1,e}$ et $\boldsymbol{r}^{2,e}$. Cette fonction se rapproche de $\Lambda^{(t)}$ à n'importe quelle position dans un élément arbitraire e et peut s'écrire:

$$\Phi^e(\Lambda^{(t)}, e, \boldsymbol{r}^P, \boldsymbol{r}^O, a) = \begin{cases} \frac{d^e - n_x^e(r_x^P a + r_x^O) - n_y^e(r_y^P a + r_y^O)}{n_\lambda^e}, & \text{inside } e \\ 0, & \text{outside } e \end{cases} \tag{F.2}$$

où $\bar{n^e} = [n_x^e, n_y^e, n_\lambda^e]$ est le vecteur normal au plan sur la surface d'activité définie par le $e^{\text{ème}}$ élément et la distance $d^e = \bar{n^e} \cdot \boldsymbol{r}^{v,e} = (r_y^{3,e} r_x^{2,e} - r_x^{3,e} r_y^{2,e})\ \lambda_1^e + (r_y^{1,e} r_x^{3,e} - r_x^{1,e} r_y^{3,e})\ \lambda_2^e + (r_y^{1,e} r_x^{2,e} - r_x^{1,e} r_y^{2,e})\ \lambda_3^e$ fait partie de l'équation linéaire du plan : $n_x^e\ x + n_y^e\ y + n_\lambda^e\ \lambda - d^e = 0$. Ainsi, l'opération de projection sur le maillage est donnée par:

$$\hat{x}_l = x_l^\mu x_l^{\texttt{eff.}} \sum_{e=1}^{E} \int \Phi^e(\Lambda^{(t)}, e, \boldsymbol{r}_l^P, \boldsymbol{r}_l^O, a) da. \tag{F.3}$$

Les termes x_l^μ et $x_l^{\texttt{eff.}}$ sont, respectivement, le facteur de correction d'atténuation et le facteur de normalisation pour l'efficacité de détection de la $l^{ème}$ LoR. Opérant dans F.2 et en remplaçant dans l'équation F.3, l'expression peut être réduite à une forme multiplicative:

$$\hat{x}_l = x_l^\mu x_l^{\texttt{eff.}} \sum_{e=1}^{E} \sum_{v=1}^{2} \frac{\mathcal{K}_{l,e,v} \lambda_v}{n_\lambda^k}, \tag{F.4}$$

avec

$$\begin{aligned} \mathcal{K}_{l,e,1} = [(r_y^{2,e} - r_y^{3,e}).r_x^O + (r_x^{3,e} - r_x^{2,e}).r_y^O + (r_y^{3,e} r_x^{2,e} - r_x^{3,e} r_y^{2,e})](a_{n,e}^{\texttt{out}} - a_{n,e}^{\texttt{in}}) \\ + [(r_y^{2,e} - r_y^{3,e}).r_x^P + (r_x^{3,e} - r_x^{2,e}).r_y^P] \left[\frac{(a_{n,e}^{\texttt{out}})^2 - (a_{n,e}^{\texttt{in}})^2}{2} \right], \end{aligned} \tag{F.5}$$

$$\begin{aligned} \mathcal{K}_{n,e,2} = [(r_y^{3,e} - r_y^{1,e}).r_x^O + (r_x^{1,e} - r_x^{3,e}).r_y^O + (r_y^{1,e} r_x^{3,e} - r_x^{1,e} r_y^{3,e})](a_{n,e}^{\texttt{out}} - a_{n,e}^{\texttt{in}}) \\ + [(r_y^{3,e} - r_y^{1,e}).r_x^P + (r_x^{1,e} - r_x^{3,e}).r_y^P] \left[\frac{(a_{n,e}^{\texttt{out}})^2 - (a_{n,e}^{\texttt{in}})^2}{2} \right], \end{aligned} \tag{F.6}$$

$$\mathcal{K}_{n,e,3} = [(r_y^{1,e} - r_y^{2,e}).r_x^O + (r_x^{2,e} - r_x^{1,e}).r_y^O + (r_y^{1,e}r_x^{2,e} - r_x^{1,e}r_y^{2,e})](a_{n,e}^{\mathtt{out}} - a_{n,e}^{\mathtt{in}}) + [(r_y^{1,e} - r_y^{2,e}).r_x^P + (r_x^{2,e} - r_x^{1,e}).r_y^P] \left[\frac{(a_{n,e}^{\mathtt{out}})^2 - (a_{n,e}^{\mathtt{in}})^2}{2}\right]. \quad \text{(F.7)}$$

Les éléments $a_{n,e}^{\mathtt{in}}$ et $a_{n,e}^{\mathtt{out}}$ des équations F.5, F.6 et F.7 sont les valeurs d'entrée (*in*put) et de sortie (*out*put) du scalaire a qui décrit le point de départ et de fin du $l^{\text{ème}}$ LoR dans le $e^{\text{ème}}$ élément du maillage. Ces valeurs sont obtenues en traçant le LoR à l'intérieur du $e^{\text{ème}}$ élément. Les éléments $\mathcal{K}_{l,e,v}$, de l'équation F.4, présentés dans les équations F.5, F.6 et F.7 sont la contribution du point échantillonné à la probabilité de détecter un événement originaire de la position $\boldsymbol{r}(a)$, en direction de la $l^{\text{ème}}$ LoR. Cette valeur est dérivée de l'estimation de la valeur d'un point dans un plan défini par trois points. L'indice v représente les sommets qui composent l'élément triangulaire e du maillage. Le total des éléments E est composé des nœuds I du maillage. Pour plus de simplicité, la notation basée sur les éléments est laissée de côté et l'équation F.4 peut être réécrite comme une sommation sur les nœuds I du maillage, où la matrice système peut être obtenue directement:

$$\hat{x}_l = x_l^{\mu} x_l^{\mathtt{eff.}} \sum_{i=1}^{I} \mathcal{K}'_{l,i}\lambda_i, \quad \text{(F.8)}$$

avec

$$K'_{l,i} = \sum_{e=1,v=i}^{E} \frac{K_{l,e,v}}{n_{\lambda}^{e}}. \quad \text{(F.9)}$$

F.4.1.2 Opération de rétroprojection

L'opération inverse de la projection, requise pour compléter l'algorithme de maximisation des espérances (EM), est construite comme une somme du facteur d'influence des LoR L dans la liste. Cette opération représente l'activité λ pour chacun des nœuds définis dans la distribution $\Lambda^{(t)}$. L'opération de rétroprojection (BP) est définie comme suit:

$$\lambda_i = \sum_{l=1}^{L} \frac{x_l^{\mu} x_l^{\mathtt{eff.}} \mathcal{K}'_{l,i}}{\mathtt{SC}_i} x_l. \quad \text{(F.10)}$$

L'opération BP inclut un nouveau facteur de normalisation d'échantillonnage $\mathtt{SC}_i$ qui est nécessaire pour corriger l'échantillonnage non uniforme de la représentation avec maillages. Cet effet est illustré sur la figure où les différents points d'échantillonnage aux positions $\boldsymbol{r}^A$ et $\boldsymbol{r}^B$, contribuent à la définition de l'activité en différents volumes, respectivement Vol_A et Vol_B. L'activité à chaque nœud i est utilisée pour interpoler l'activité à chaque coordonnée dans tous les éléments qui composent ce volume. La surface combinée de chacun de ces éléments et la distance entre les coupes créent un volume où le nœud i a une influence. Étant donné que les éléments du maillage ont

des tailles différentes, le volume d'influence de chaque nœud est différent. La relation de la zone d'influence du nœud i avec la zone plus grande définit la valeur $\mathtt{SC}_i$.

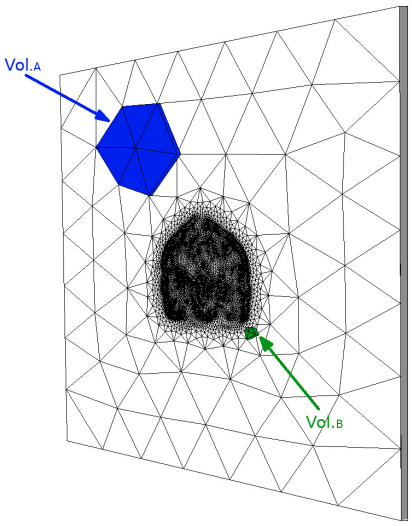


Figure F.3: Volume d'influence de deux nœuds différents. En haut de l'image, le volume autour du nœud central "A" (Vol_A) et près du centre de l'image le volume autour du nœud central "B" (Vol_B).

F.4.1.3 Opération dans LoR obliques

L'activité totale d'un LoR 3D traversant l'intérieur d'un prisme triangulaire dépend de: La valeur d'activité sur chacun des six nœuds de l'élément, trois du triangle avant et trois du triangle arrière; La coordonnée d'entrée (in); La coordonnée de sortie (out). Sur la figure F.4, le croisement d'un LoR dans l'élément 3D est représenté. L'activité d'un point aléatoire $\boldsymbol{r}(a)$ dans l'élément 3D peut être calculée à l'aide d'une interpolation tri-linéaire. Ceci est réalisé grâce au calcul initial de l'activité sur chacun des bords du plan triangulaire ($\boldsymbol{r}^0$, $\boldsymbol{r}^1$, $\boldsymbol{r}^2$) qui contient le point à interpoler à l'aide d'une interpolation linéaire:

$$\lambda_v = (\frac{\lambda_v^b - \lambda_v^f}{\Delta z})(r_z^P a + r_z^O) + \lambda_v^f, \tag{F.11}$$

puis appliquer l'équation F.2 en utilisant les valeurs des nœuds $\boldsymbol{r}^{n,e}$. Cependant, comme il a été choisi que l'activité le long de l'axe z d'une coupe est constante, l'activité dans le triangle avant et arrière est la même ($\lambda_v^f = \lambda_v = \lambda_v^b$). Ensuite, l'interpolation trilinéaire est simplifiée à l'interpolation bi-linéaire présentée dans l'équation F.2.

Tous les LoR qui ne rentrent pas dans un seul maillage 2D sont des LoR obliques multi-tranches. Ces LoR sont segmentés dans chaque coupe traversée et chaque segment est projeté ou rétro-projeté en utilisant les équations F.8 et F.10, respectivement. La segmentation d'un LoR est effectuée en calculant le paramètre d'entrée et de sortie a pour chaque coupe n_s: $z_s^{\mathtt{in}} = z_s - \Delta z/2$ et $z_s^{\mathtt{out}} = z_s + \Delta z/2$. Résultant en deux

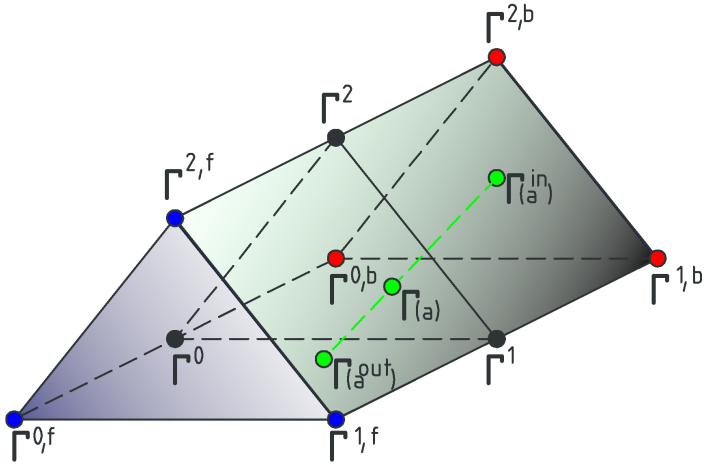


Figure F.4: Prisme triangulaire représentant l'élément 3D du maillage. Les points bleues ($\boldsymbol{r}^{0,f}$, $\boldsymbol{r}^{1,f}$, $\boldsymbol{r}^{2,f}$) et les points rouges ($\boldsymbol{r}^{0,b}$, $\boldsymbol{r}^{1,b}$, $\boldsymbol{r}^{2,b}$) représentent les limites avant et arrière du prisme. La ligne pointillée verte avec des limites $\boldsymbol{r}(a^{\texttt{in}})$ et $\boldsymbol{r}(a^{\texttt{out}})$ représente la trajectoire de la LoR dans l'élément. Le point vert $\boldsymbol{r}(a)$ est un point coplanaire inclus dans un triangle arbitraire à l'intérieur du prisme, représenté par les points noirs ($\boldsymbol{r}^0$, $\boldsymbol{r}^1$, $\boldsymbol{r}^2$).

points, $\boldsymbol{r}(a_s^{\texttt{in}})$ et $\boldsymbol{r}(a_s^{\texttt{out}})$. Puis les éléments $K_{l,e,v}$ sont calculés sur ces valeurs dans le segment $\overline{\boldsymbol{r}(a_s^{\texttt{out}}) - \boldsymbol{r}(a_s^{\texttt{in}})}$. Enfin, le LoR est modulé par la longueur de celui-ci dans la coupe.

F.4.2 Maximisation de la vraisemblance

Pour optimiser la distribution d'activité proposée $\Lambda^{(t)}$ à la mesure x_l, la vraisemblance logarithmique est maximisée:

$$\ell(X|\Lambda^t) = \sum_{l=1}^{L} -\hat{x}_l + x_l \ln(\hat{x}_l). \tag{F.12}$$

L'expression du mode liste de l'algorithme MLEM [146] est alors exprimée par:

$$\lambda_i^{(t+1)} = \frac{\lambda_i^{(t)}}{\sum_{l=1}^{L^*} \frac{x_l^{\mu} x_l^{\texttt{eff}} \cdot \mathcal{K}'_{l,i}}{\texttt{SC}_i}} \sum_{l=1}^{L} \frac{1.0}{\hat{x}_l} \frac{x_l^{\mu} x_l^{\texttt{eff}} \cdot \mathcal{K}'_{l,i}}{\texttt{SC}_i}, \tag{F.13}$$

où $\sum_{l=1}^{L^*} \frac{AC_i DC_i \mathcal{K}'_{l,i}}{\texttt{SC}_i}$ est la somme sur une liste densément échantillonnée de la distribution de tous les LoR mesurables. Chaque étape de la fonction F.13 augmente de façon monotone la probabilité de la distribution $\Lambda^{(t)}$ à l'ensemble des LoR détectés $\mathcal{X}$.

F.4.3 Schéma de remaillage

Comme le scanner ne peut capturer que des correspondances d'inclinaison perpendiculaires ou limitées par rapport à l'axe z, il a été choisi de diviser l'image en utilisant une série de maillages 2D le long de l'axe axial du scanner (l'axe z). Par rapport à un maillage 3D, la représentation proposée nécessite moins de calculs, en contre partie d'une erreur d'interpolation plus importante sur l'axe z. Chacun des maillages dans la représentation proposée est calculé comme un maillage 2D, le processus de remaillage est donc simplifié. Le volume d'une tranche est défini par la taille du maillage 2D et l'espacement avec le maillage suivant. Puisque l'activité au sein d'une tranche est supposée constante le long de l'axe z, la représentation 3D est considéré comme 2D pour chaque tranche. En utilisant cette représentation, les nœuds peuvent être stratégiquement positionnés dans des régions où une représentation plus compacte est nécessaire (pour les coins ou les textures complexes). La possibilité de contrôler les positions des nœuds permet la description de l'image avec potentiellement moins d'éléments que dans la représentation voxélisée. Une stratégie de positionnement des nœuds consiste à utiliser un processus de remaillage adaptatif [142, 143]. Pour produire un maillage avec une erreur d'interpolation contrôlée, un processus d'adaptation est nécessaire. Ce processus améliore le maillage en utilisant des métriques sur un maillage précédent avec unne estimation de l'erreur. La distribution réelle du radiotraceur étant inconnue, l'estimation de l'erreur est réalisée a posteriori, à partir de la reconstruction sur le maillage actuel. L'erreur est confinée localement, au voisinage du nœud, à l'aide d'une simple approximation basée sur l'écart local de la surface de la solution [125]. Cette métrique, basée sur le Hessien discret, résulte en une carte de taille qui est utilisée pour produire le nouveau maillage adapté. Chaque élément de la carte peut être écrit comme suit:

$$h(i) = \frac{\delta}{\eta(i, \lambda_i)}, \tag{F.14}$$

où $\eta(i, \lambda_i)$ est l'écart maximal entre les nœuds adjacents et le plan tangent à ce nœud. Le processus de remaillage est décrit comme suit:

Algorithm 10: Processus de remaillage

```
while e_s > δ do
    Calculer la carte des tailles F^(t)(Λ), basé sur la solution discrète du Hessien
     de Λ.
    Mettre à jour F_s^(t+1)(Λ) étant donné la carte de taille.
    Évaluer l'erreur "e" de F_s^(t)(Λ), en utilisant une interpolation linéaire.
end
```

L'opération de remaillage basée sur une carte de taille garantit que l'erreur d'interpolation entre les nœuds est limitée par une taille minimale et maximale ($h_{min} < h(i) < h_{max}$), en sélectionnant soigneusement h_{min}, il est possible d'éviter de créer des structures artificielles à haute fréquence (structurelles) et des artefacts de bruit. Dans les nœuds où $h = h_{min}$, l'erreur ne sera pas confinée, cependant, en choisissant la valeur de $h = h_{min}$

suffisamment faible, ces différences sont inférieures à la résolution du tomographe.

F.4.4 Itérations de reconstruction

En utilisant la procédure de reconstruction présentée comme le modèle physique dans le processus de remaillage, une opération de reconstruction avec une erreur d'interpolation confinée est réalisée. L'ampleur de cette erreur est directement liée à la précision de l'opération de projection (voir l'équation F.8). Le maillage ainsi obtenu est une représentation plus compacte, qui ne perd pas du précision et nécessite le calcul de moins de nœuds à mesure que le maillage converge. Le processus de reconstruction est effectué comme suit:

Algorithm 11: Processus de reconstruction

```
1 Créer N_s maillages initiaux épaisses F_s^(t=0)(Λ^(t=0))
2 for S_L étapes do
3 |   for S_R étapes do
4 |   |   Calculer l'intensité de chacun F_s^(t)(Λ^(t)), comme un volume (étape de
  |   |    reconstruction) .
5 |   end
6 |   for tous les F dans N_s do
7 |   |   Obtenir F_s^(t+1)(Λ^(t+1)) en appliquant un remaillage pendant S_remesh
  |   |    étapes (Voir l'algorithme 10).
8 |   end
9 end
```

Chacune des étapes de reconstruction de l'algorithme 11 est une série d'étapes de reconstruction S_R MLEM. Le nombre d'étapes à appliquer est arbitraire. L'estimation d'activité n'est pas nécessaire pour atteindre une valeur optimale avant de mettre à jour le maillage du support. Le processus de remaillage étant régi par la limite σ, les positions spatiales des nœuds sont adaptées pour minimiser l'erreur d'interpolation. Il en résulte un changement contrôlé de la valeur de projection $\hat{x}_l$ qui à son tour contrôle le changement de la fonction objectif. Ainsi, chaque itération permet d'obtenir une meilleure qualité de représentation de l'activité inconnue, avec un impact confiné sur la fonction objectif.

F.4.5 Expériences

La stratégie de reconstruction proposée est testée à l'aide de deux fantômes: une simulation Monte Carlo du cerveau 3D de Hoffman et une acquisition du fantôme de qualité d'image NEMA [112]. La simulation a été réalisée à l'aide du logiciel GATE [107], où la géométrie de l'AR-PET a été simulée. L'acquisition du fantôme NEMA a été réalisée à l'aide du prototype préclinique AR-PET. La méthode de reconstruction proposée est comparée à la reconstruction MLEM sur support voxélisé présentée dans [146].

Toutes les reconstructions partagent les mêmes données (Liste des événements), le même modèle mathématique de la tomographie et des techniques de correction.

Simulation du fantôme cérébral de Hoffman

Le fantôme est situé au centre du champ de vision (FoV). La simulation est exempte d'effets d'atténuation et d'erreurs de positionnement afin de mieux refléter la capacité de compression du maillage. Pendant la simulation, un total de 1.74×10^8 correspondances sont collectées (y compris réel et aléatoire). Afin d'évaluer le comportement de la technique de reconstruction proposée pour différents niveaux de statistique, la simulation du cerveau 3D Hoffman a été reconstruite en utilisant quatre niveaux de statistiques différents 1×10^5, 1×10^6, 1×10^7 et 1.7×10^8 LoRs. La reconstruction du support de maillage utilise des coupes de 2 mm, $\delta = 0.01$ et $h_{min} = 1.0$ mm. Le maillage initial est un maillage quasi cartésien avec une distance entre les nœuds de 4 mm. Le processus est appliqué pendant 10 itérations avec $S_R = 10$ étapes de reconstruction et $S_{\texttt{remesh}} = 5$ étapes de remaillage, résultant en 50 étapes de remaillage et 100 étapes de reconstruction. La reconstruction sur support voxélisé se fait avec $S_R = 100$ itérations et utilise la même liste d'entrée que la méthode présentée. La reconstruction sur voxels est filtrée à l'aide d'un filtre gaussien avec un écart type de $\sigma = 1$ mm, pour lisser tout bruit haute fréquence du calcul de reconstruction. Ce filtrage n'est pas nécessaire pour le support de maillage puisqu'il évite en ré-échantillonnant le support là où il ne respecte pas la limite σ. Deux configurations voxélisées ont été testées, l'une avec une taille de voxel de $2 \times 2 \times 2$ mm^3, résultant en $256 \times 256 \times 256 = 16777216$ voxels et l'autre avec une taille de voxel de $4 \times 4 \times 4$ mm^3, résultant en $128 \times 128 \times 128 = 2097152$ voxels. Afin de comparer les reconstructions, la méthode présentée est projetée sur un volume de taille voxel de $2 \times 2 \times 2$ mm^3 en utilisant une interpolation bilinéaire dans chaque tranche. La représentation voxel de faible qualité est interpolée linéairement à une taille de voxel de $2 \times 2 \times 2$ mm^3 pour être comparée.

Pour comparer les images reconstruites, le rapport signal sur bruit (SNR), l'erreur quadratique moyenne (MSE) et le contraste de la matière grise ($\mathtt{C}_{\texttt{grey}}$) ont été utilisés. La métrique de contraste est calculée comme suit:

$$\mathtt{C}_{\texttt{grey}} = \frac{\overline{\mathcal{V}_{\texttt{grey}}} - \overline{\mathcal{V}_{\texttt{white}}}}{\overline{\mathcal{V}_{\texttt{white}}}}, \tag{F.15}$$

où $\overline{\mathcal{V}_{\texttt{grey}}}$ et $\overline{\mathcal{V}_{\texttt{white}}}$ sont la valeur moyenne des voxels identifiés respectivement comme matière grise et blanche. La segmentation du voxel a été réalisée en utilisant un seuil dans l'image vraie. Étant donné que l'image occupe une petite fraction du FoV total, une boîte englobante a été construite autour du fantôme et les mesures ont été appliquées uniquement dans cette boîte. La taille de la boîte englobante pour le fantôme Hoffman est de $102 \times 102 \times 102 = 1.06 \times 10^6$ voxels.

Mesure du fantôme NEMA

Le fantôme est construit selon les directives proposées par NEMA et situé au centre du FoV. L'acquisition est statique (une seule position de la civière) et les détecteurs tour-

nent à 2 rpm. L'acquisition dure 47 minutes et 2.0×10^7 coïncidences sont enregistrés. La dose injectée est 1.85 `mCi`, les lésions sévères sont remplies avec une concentration 4 fois supérieure à celle du fond. Les images d'atténuation sont acquises à l'aide d'un scanner hélicoïdal *Aquilion Lightning, Canon Medical Systems* de 16 lignes et 32 tranches. Les images sont co-enregistrées manuellement en utilisant une reconstruction préliminaire sans correction d'atténuation. Le fantôme NEMA a également été testé en utilisant différents niveaux de statistiques, dans ce cas les ensembles comprennent 1×10^5, 1×10^6 et 2×10^7 LoRs. Les paramètres de reconstruction sur le support de maillage étaient les mêmes que dans la reconstruction de la simulation cérébrale de Hoffman. La reconstruction sur le support de maillage est appliquée sur 20 itérations avec $S_R = 5$ et $S_{\texttt{remesh}} = 5$, résultant en 100 itérations de remaillage et 100 itérations de reconstruction. La reconstruction du voxel a été réalisée avec les mêmes paramètres que dans la simulation. Comme précédemment, pour comparer les reconstructions, la méthode proposée est projetée dans un volume de résolution équivalent en utilisant une interpolation bilinéaire dans chaque tranche.

L'acquisition fantôme NEMA a été testée à l'aide des métriques décrites dans le document NEMA 2012 [112].

F.5 Résultats

F.5.1 Simulation du fantôme Hoffman

L'activité réelle et la reconstruction sur le support de maillage et le support voxélisé haute résolution, pour l'ensemble complet de LoR, peuvent être vues sur la figure F.5, où les coupes le long de chaque axe sont affichées.

Les métriques résultantes pour chaque ensemble de LoR et le type de reconstruction sont données dans le tableau F.1. L'évolution des métriques en fonction des statistiques, pour la dernière itération, est représentée sur la figure F.6. En utilisant ces métriques, la convergence et la reconstruction peuvent être observées. Dans la figure F.7, les métriques en fonction de l'itération, pour l'ensemble total de statistiques, sont affichées.

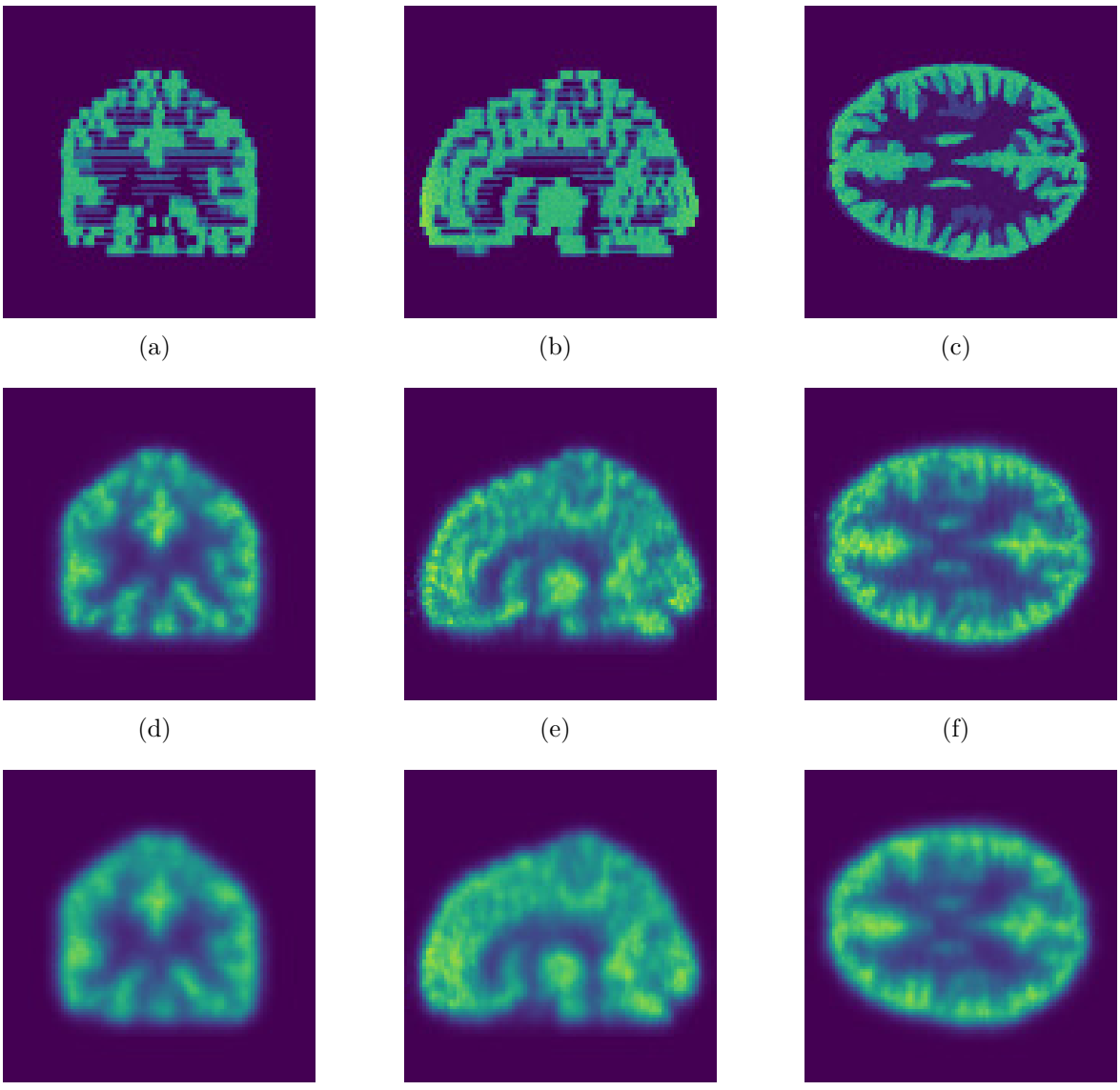


Figure F.5: Comparaison entre la distribution réelle de l'activité (a, b, c), l'interpolation de la reconstruction sur des maillages sur un support pixélisé de 2 mm (d, e, f) et la reconstruction sur support voxélisé de 2 mm (g, h, i) de la simulation du fantôme cérébrale de Hoffman. Les images présentées ont été reconstruites avec l'ensemble total de LoR.

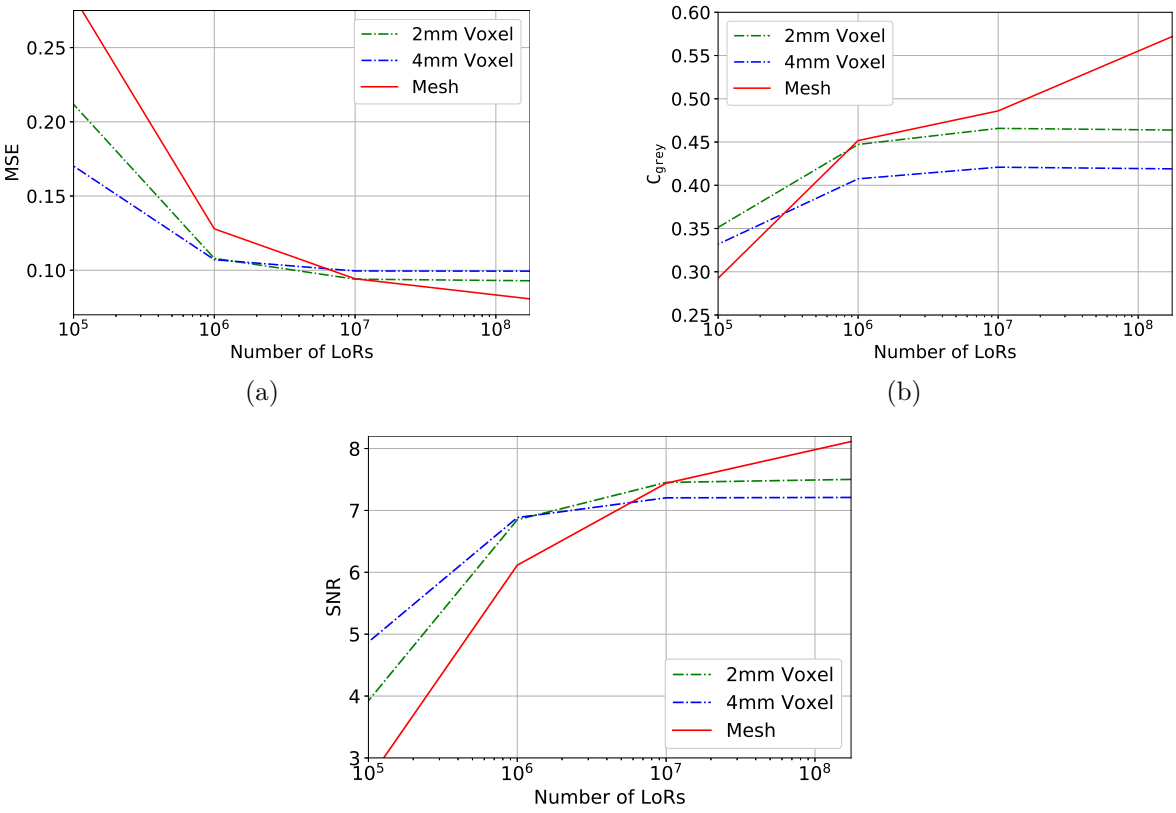


Figure F.6: Évolution du MSE (a), $\mathtt{C_{grey}}$ (b) et SNR (c) en fonction de la quantité de statistiques collectées pour la simulation du cerveau de Hoffman. La ligne pointillée verte représente la reconstruction sur le support voxélisé de taille $2 \times 2 \times 2$ mm^3, la ligne pointillée bleue représente la reconstruction sur le support voxélisé de $4 \times 4 \times 4$ mm^3 et la ligne rouge continue représente la reconstruction sur le maillage.

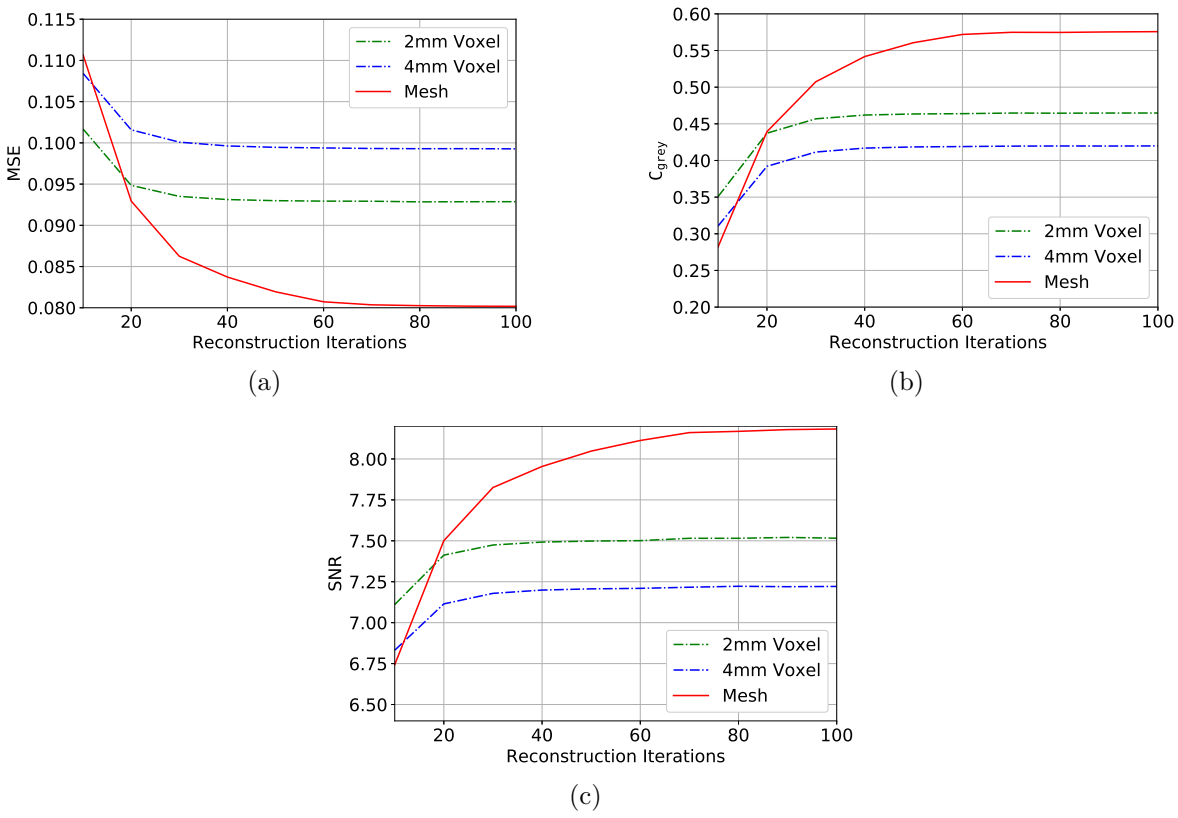


Figure F.7: Évolution du MSE (a), $\mathtt{C}_{\mathtt{grey}}$ (b) et SNR (c) en fonction de l'itération pour le plus grand ensemble statistique de la simulation du cerveau de Hoffman. La ligne pointillée verte représente la reconstruction sur le support voxélisé de taille $2\times2\times2$ mm^3, la ligne pointillée bleue représente la reconstruction sur le support voxélisé de $4\times4\times4$ mm^3 et la ligne rouge continue représente la reconstruction sur le maillage.

Table F.1: Métriques calculées pour la simulation du fantôme cérébral de Hoffman dans chaque ensemble de statistiques.

LoRs	Reconstruction		Métrique		
	Support	Résolution	MSE	SNR	C_{grey}
1×10^5	Voxel	4 mm	0.17	4.88	0.33
		2 mm	0.21	3.93	0.35
	Maillage	-	0.28	2.66	0.29
1×10^6	Voxel	4 mm	0.11	6.90	0.40
		2 mm	0.11	6.86	0.44
	Maillage	-	0.13	6.15	0.45
1×10^7	Voxel	4 mm	0.099	7.21	0.42
		2 mm	0.094	7.46	0.46
	Maillage	-	0.093	7.51	0.49
1.7×10^8	Voxel	4 mm	0.099	7.22	0.41
		2 mm	0.092	7.51	0.46
	Maillage	-	0.080	8.18	0.57

F.5.2 Mesure du fantôme NEMA

La forme fantôme, la reconstruction sur le maillage et la reconstruction sur le support voxélisé haute résolution sont visibles sur la figure F.8, où sont représentées les coupes le long de chaque axe.

La précision des corrections d'atténuation et de dispersion ($C_{\texttt{lung}}$, le plus base correspondent a la plus grand efficacité), contraste dans les sphères froides ($Q_{\texttt{cold}}$, le plus élevée correspondent a la plus grand efficacité), contraste dans les sphères chaudes ($Q_{\texttt{hot}}$, le plus élevée correspondent a la plus grand efficacité) et variation moyenne du fonds ($N_{\texttt{mean}}$, le plus base correspondent a la plus grand efficacité) sont résumées, pour chaque ensemble de statistiques et de résolution, dans le tableau F.2. Dans la figure F.2, les métriques en fonction des itérations, pour le plus grand ensemble de statistiques, sont présentées.

F.5.3 Convergence du support de maillage

Le processus de reconstruction se termine lorsque la reconstruction et le processus de maillage convergent. Le processus de remaillage produit une configuration de nœuds spaciaux qui garantit que l'erreur d'interpolation est inférieure à un seuil δ. La convergence de la représentation du maillage peut être observée en termes de nombre de nœuds utilisés pour représenter le volume. Sur la figure F.10, l'évolution du nombre total de nœuds et des nœuds dans une section centrale, pour chaque fantôme, est présentée. La coupe choisie pour chaque reconstruction est présentée sur la figure F.11.

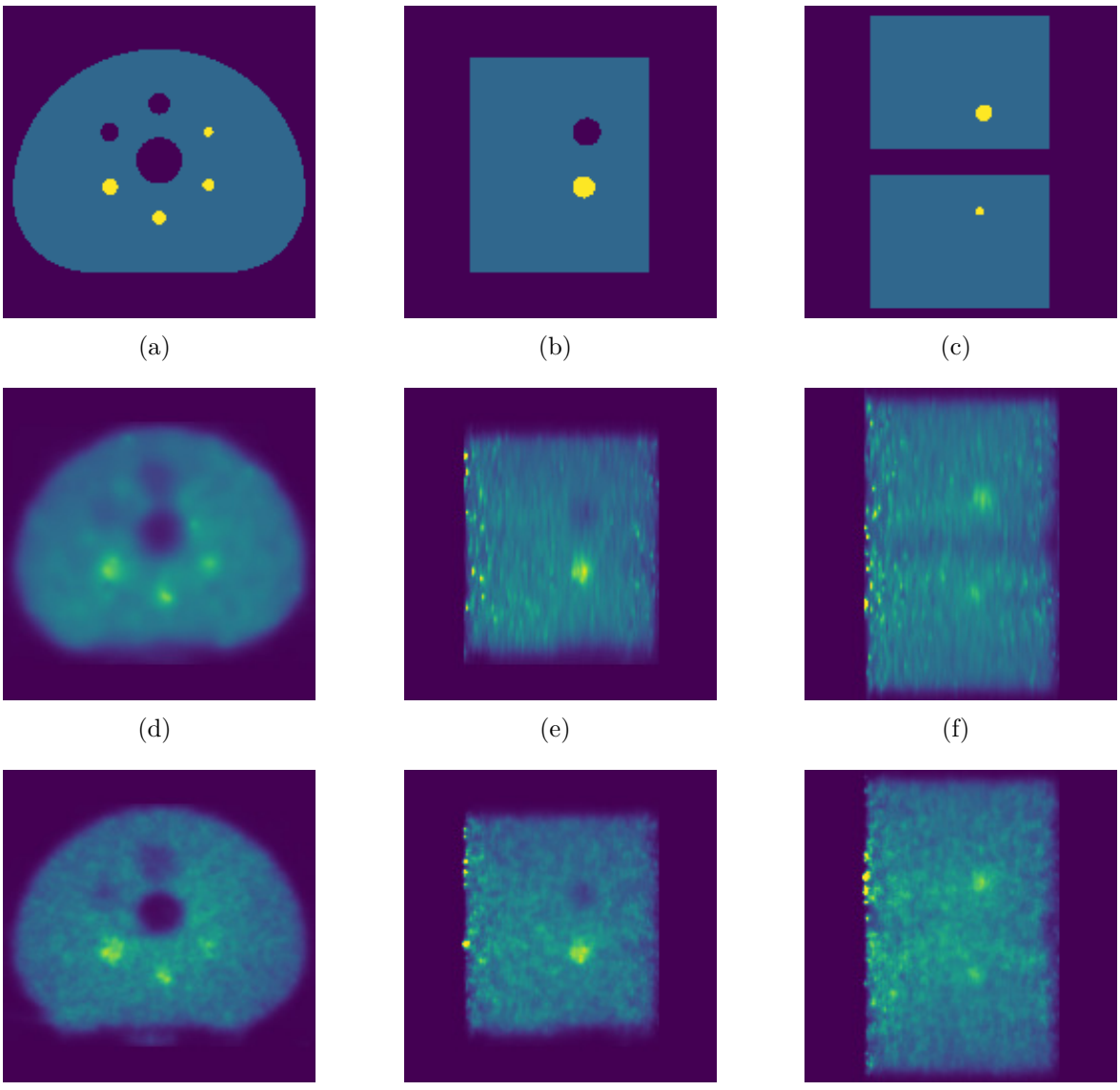


Figure F.8: Comparaison entre la répartition attendue de l'activité (a, b, c), l'interpolation de la reconstruction sur des maillages sur un support pixélisé de 2 mm (d, e, f) et la reconstruction en support voxélisé de 2 mm (g, h, i) de la mesure du fantôme NEMA. Les images présentées ont été reconstruites avec l'ensemble total de LoR.

Table F.2: Métriques pour la mesure du fantôme NEMA 2012.

LoRs	Reconstruction		Métrique			
	Support	Résolution	$C_{\texttt{lung}}$	$Q_{\texttt{cold}}$	$Q_{\texttt{hot}}$	$N_{\texttt{mean}}$
1×10^5	Voxel	4 mm	0.52	0.17 ± 0.06	0.53 ± 0.14	0.61 ± 0.17
		2 mm	0.43	0.15 ± 0.07	0.31 ± 0.05	0.87 ± 0.21
	Maillage	-	0.52	0.23 ± 0.32	0.21 ± 0.29	0.60 ± 0.29
1×10^6	Voxel	4 mm	0.39	0.24 ± 0.09	0.38 ± 0.07	0.23 ± 0.07
		2 mm	0.36	0.23 ± 0.09	0.33 ± 0.06	0.29 ± 0.08
	Maillage	-	0.42	0.18 ± 0.06	0.18 ± 0.06	0.26 ± 0.11
2×10^7	Voxel	4 mm	0.30	0.26 ± 0.00	0.41 ± 0.11	0.065 ± 0.025
		2 mm	0.29	0.26 ± 0.01	0.42 ± 0.11	0.082 ± 0.025
	Maillage	-	0.37	0.33 ± 0.14	0.33 ± 0.14	0.043 ± 0.023

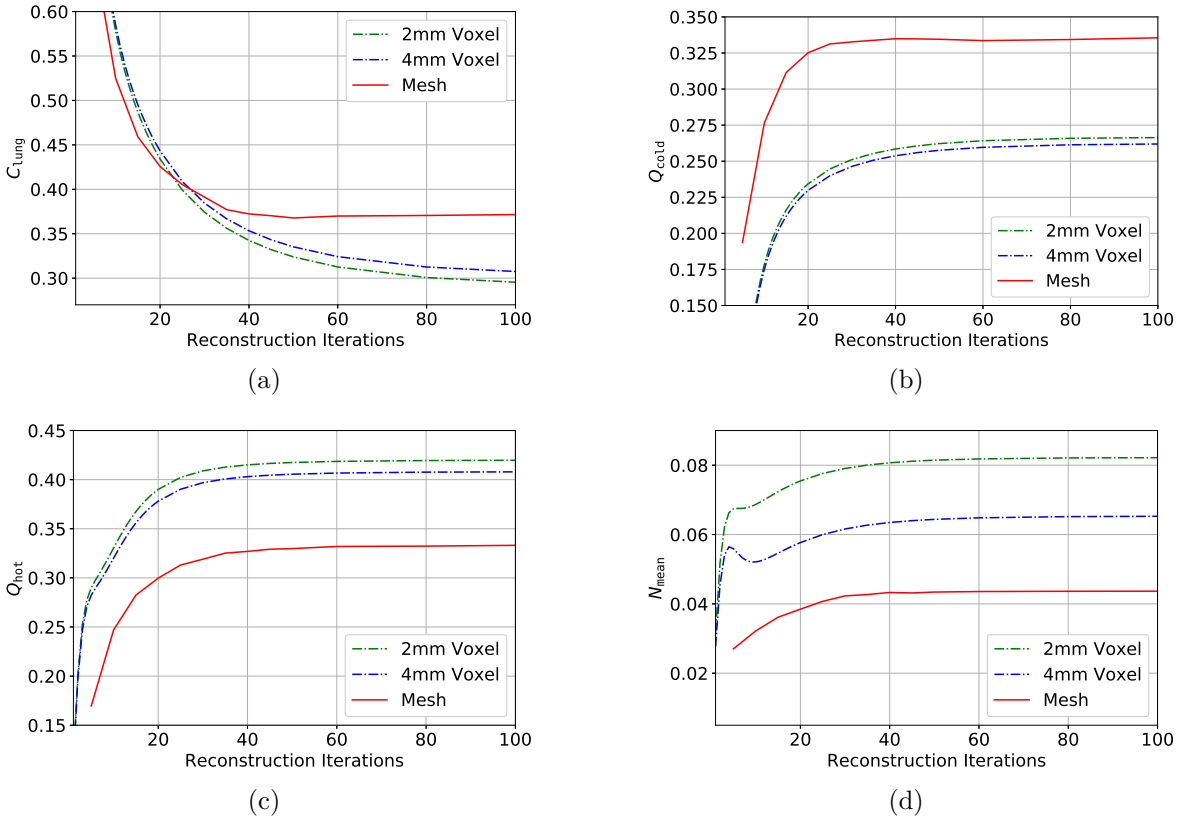
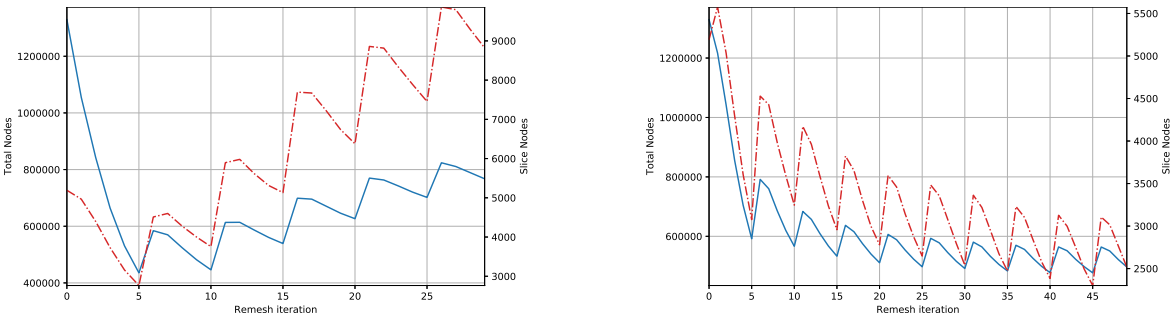


Figure F.9: Evolution de la correction d'atténuation et de dispersion $C_{\texttt{lung}}$ (a), contraste dans les sphères froides $Q_{\texttt{cold}}$ (b), contraste dans les sphères chaudes $Q_{\texttt{hot}}$ (c) et variabilité moyenne du fond $N_{\texttt{mean}}$ (d) en fonction de l'itération pour l'ensemble de statistiques de l'acquisition fantôme NEMA. La ligne pointillée verte représente la reconstruction sur le support voxélisé de taille $2 \times 2 \times 2$ mm^3, la ligne pointillée bleue représente la reconstruction sur le support voxélisé de $4 \times 4 \times 4$ mm^3 et la ligne rouge continue représente la reconstruction sur le maillage.



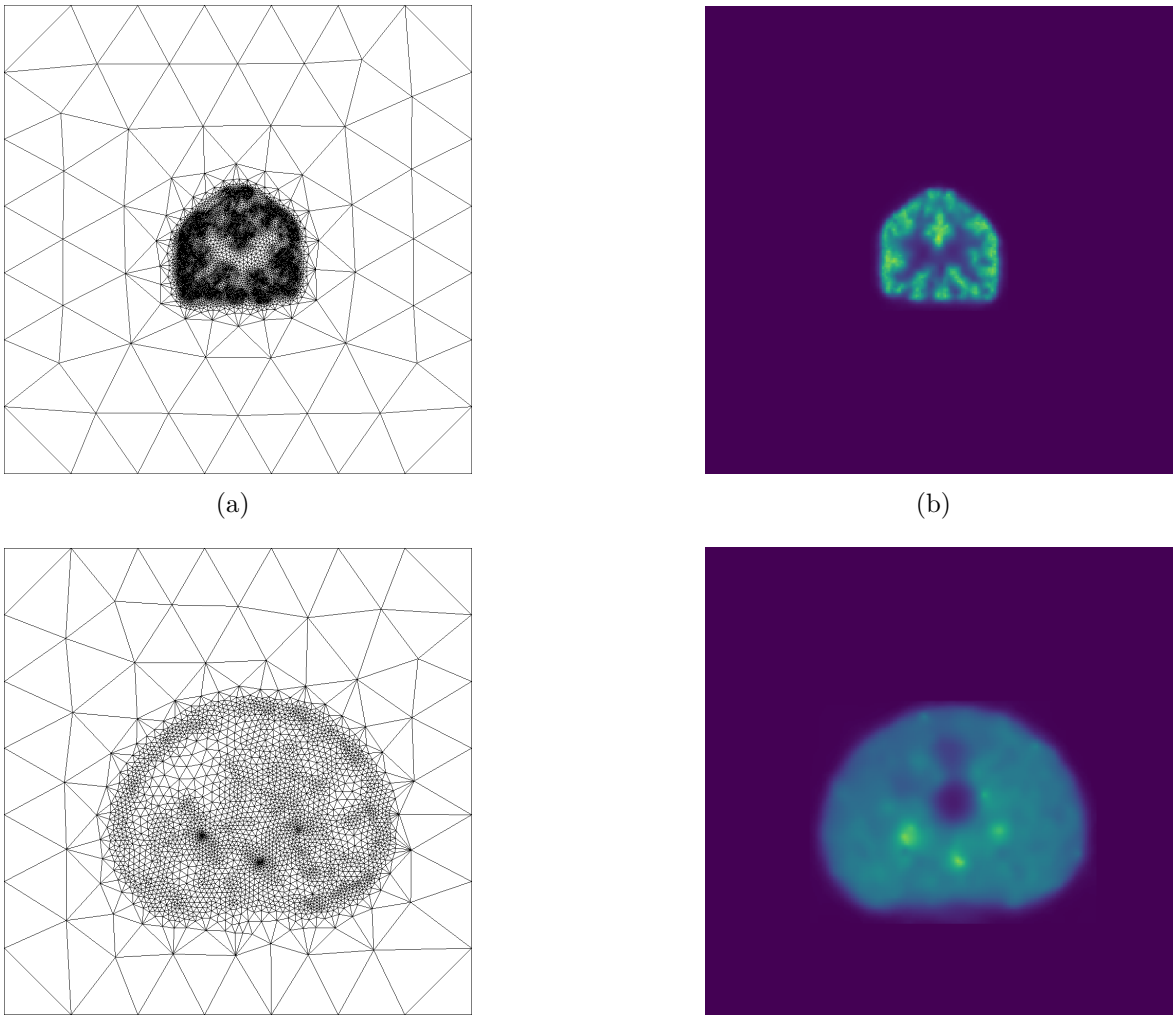
(a) (b)

Figure F.10: Evolution du nombre de nœuds utilisés pour échantillonner la simulation du cerveau de Hoffman (a) et l'acquisition du fantôme NEMA (b). La ligne pleine représente le nombre total de nœuds (échelle gauche) et la ligne en pointillé représente l'évolution des nœuds dans une section centrale du fantôme (échelle droite). L'axe x représente les itérations du processus de remaillage, toutes les cinq itérations, un processus de reconstruction a été appliqué.

F.6 Discussion

La méthode présentée permet d'obtenir une qualité de reconstruction supérieure dans le fantôme cérébral de Hoffman, améliorant le MSE de 0.092 à 0.080, le SNR de 7.51 `db` à 8.18 `db` et le contraste de la matière grise de 0.46 à 0.57 par rapport à la reconstruction voxélisée. Pour la reconstruction du fantôme NEMA, le maillage a partiellement réussi, améliorant le contraste dans les sphères froides de 0.26 à 0.33 et la variabilité de fond de 0.082 à 0.043 mais dégradant la correction d'atténuation de 0.29 à 0.37 et le contraste des sphères. de 0.42 à 0.33.

On peut voir que les métriques de simulation de fantôme cérébral de Hoffman (table F.1) pour le support de maillage et pour le support voxélisé haute résolution sont très similaires pour les ensembles de statistiques de 10×10^6 et 10×10^7 LoRs. Lorsque le nombre de LoR dépasse le 1×10^7, le support de maillage est supérieur aux support voxélisé. Cependant, la qualité de la reconstruction sur le maillage diminue plus vite avec le nombre de LoR que la reconstruction sur le support voxélisé. Cet effet est visible sur la figure F.6, où le MSE et le SNR de la dernière itération de reconstruction dans le support de maillage sont inférieurs à celui du support voxélisé pour des statistiques 1×10^7. Il convient de noter que cela se produit également lors de la comparaison des reconstructions voxélisées, le support 4 mm atteint une qualité similaire ou meilleure que le support 2 mm pour les ensembles de statistiques de 1×10^6 LoR. La même chose est observée dans le fantôme NEMA dans le tableau F.2 où la reconstruction sur les maillages pour 1×10^6 LoRs atteint des métriques inférieures. Dans le cas du fantôme NEMA, le maillage a un meilleur contraste des sphères froides et une meilleure variabilité du fond, cependant il a une aptitude plus faible à la correction d'atténuation et au contraste des sphères chaudes. Ce comportement peut être lié à des failles dans la représentation du maillage, qui est plus sensible aux données plus éparses ou plus



(a) (b)

(c) (d)

Figure F.11: Représentation du maillage (a, c) et de la distribution d'activité correspondante (b, d) d'une coupe détaillée des reconstruction du fantôme de cerveau de Hoffman (a, b) et du fantôme NEMA (c, d).

bruités.

L'application de l'algorithme MLEM sur les maillages peut améliorer la qualité des ensembles avec de grandes statistiques et réduire le nombre de paramètres nécessaires pour représenter la distribution de l'activité. Le cerveau Hoffman nécessite 7.67×10^5 nœuds pour être représentés et le fantôme NEMA 5.22×10^5 nœuds, la reconstruction de voxel haute résolution nécessite 1.7×10^7 voxels et la basse résolution 2.1×10^6. C'est plus d'un ordre de grandeur dans le nombre d'éléments pour représenter la même information. Pour réduire l'effet des grandes zones vides dans le FoV, on peut analyser une tranche détaillée de la reconstruction. Cette coupure nécessite 8.82×10^3 nœuds pour le cerveau Hoffman et 2.55×10^3 nœuds pour le fantôme NEMA. La même coupe pour le support voxélisé nécessite $256 \times 256 = 6.55 \times 10^4$ et $128 \times 128 = 1.64 \times 10^4$ voxels pour un rendu haute et basse résolution respectivement. Même dans ce cas, la représentation maillée utilise au plus la moitié des paramètres dans le pire des cas. Cette amélioration peut être utilisée pour mettre en œuvre des algorithmes plus complexes et des systèmes PET dynamiques qui bénéficient d'une matrice système réduite.

Comme dans les études précédentes [128–132] la mise en place d'un support au moyen de maillages a conduit à une qualité de reconstruction similaire ou meilleure et à une réduction de la taille du problème. Ces études rapportent également une réduction du bruit de fond lors de la reconstruction d'acquisitions réelles [129,131], ceci a également été observé dans nos expériences, avec la réduction de la variation de fond moyenne dans le fantôme NEMA. En plus de la nouvelle approche de la reconstruction de l'activité sur les maillages, les effets des niveaux d'activité sur la reconstruction ont été explorés. Ceci était inexploré dans la littérature et nous montrons que cela affecte la reconstruction sur le support de maillage.

Pour éviter les problèmes de représentation de l'image d'atténuation dus au fait que le support est adapté à l'activité, il a été choisi d'utiliser un maillage secondaire pour représenter la carte d'atténuation. Bien que cette approche nécessite de tracer la LoR sur un nouveau domaine, l'atténuation peut être précalculée, puisqu'elle ne varie pas. L'optimisation d'un support unique pour l'atténuation et l'activité, pour utiliser dans des algorithmes tels que la maximisation de la probabilité d'activité et l'atténuation, sont les perspectives futurs.

F.7 Discussions et commentaires

Dans ce chapitre, des applications de maillage pour la reconstruction d'images PET ont été présentées. Les bases de la reconstruction d'images PET sur un maillage avec une nouvelle méthode de reconstruction sont présentées. La méthode montre qu'une série de maillages 2D peut représenter des distributions 3D complexes. Cette représentation n'est pas limitée aux LoR 2D et elle est capable de représenter toutes les projections 3D d'un tomographe conventionnel (n'étant limité que par les LoR totalement transversales, qui ne sont pas détectables dans les scanners actuels). De plus, le support de maillage s'adapte dynamiquement à la géométrie du radiotraceur, en partant d'un maillage grossier et en s'adaptant à mesure que les détails de l'image émergent.

La représentation résultante est plus compacte et nécessite potentiellement moins de calculs qu'un FoV voxélisé pour représenter les mêmes données. Cette représentation signifie que moins de paramètres sont nécessaires et rend les projections des LoR moins coûteuses dans les domaines où les détails ne sont pas nécessaires.

Ceci termine le présentation des techniques de maillage incluses dans cette thèse. Il a été montré qu'il est possible d'exprimer le problème de reconstruction d'image PET en utilisant moins de paramètres sans perte de qualité d'image. La représentation sur les maillages, couplée à la possibilité d'appliquer l'algorithme Simulation Simple de la Dispersion, est un pas vers la reconstruction mutuelle de l'activité et de la carte d'atténuation avec des coûts de calcul réduits. Le dernier chapitre de cette thèse se concentre sur la réduction supplémentaire de la complexité de la reconstruction mutuelle de l'activité et de l'atténuation grâce à la génération automatique d'une image d'atténuation a priori.

G Génération de cartes d'atténuation à l'aide de réseaux de neurones profonds

G.1 Introduction

L'utilisation de méthodes d'apprentissage profond dans la segmentation, la classification et la génération d'images est une tendance forte dans différents domaines. Avec l'apparition d'AlexNet [147] en 2012, de nombreux problèmes ont commencé à être repensés avec une certaine influence des réseaux de neurones profonds (DNN). Cela se traduit par l'augmentation exponentielle des publications, à travers une grande variété de domaines, y compris des mots-clés tels que l'apprentissage de la machine ou d'un réseau de neurones. Le domaine de l'imagerie médicale a augmenté l'utilisation de ces nouveaux outils. Dans ce domaine, la tendance à construire des descripteurs personnalisés pour des problèmes spécifiques est passée à l'utilisation de représentations apprises, utilisant la puissance des modèles profonds [148]. Le réseau de neurones convolutifs (CNN) a commencé à être la technique prédominante dans le domaine de l'imagerie médicale en 2015, appliquant une variété de modèles pour différents problèmes [149]. Récemment, l'imagerie médicale a commencé à faire partie intégrante de la tendance ML. En 2018, les travaux de synthèse d'images médicales ont atteint 49% des publications dans le domaine [150]. Les algorithmes les plus modernes de synthèse d'images médicales sont basés sur l'apprentisage antagoniste des topologies neuronales profondes, une technique introduite en 2014 [151]. Ces algorithmes génératifs ont obtenu des résultats impressionnants dans des domaines où de grandes bases des données existent, comme la génération de visages humains [152]. Bien que les données ne soient pas aussi abondantes dans ce domaine, les représentations possibles dans le domaine des images médicales sont également réduites par rapport à d'autres domaines de l'image comme les photographies de visages. Ensuite, en exploitant cette régularité dans les images médicales, les modèles antagonistes peuvent être appliqués à différentes modalités d'imagerie [153].

Ce chapitre aborde le thème du DNN dans les images médicales. Il se concentre sur la génération de cartes d'atténuation (AM) à utiliser dans la correction des images PET. Il s'agit d'une étape fondamentale vers un étude PET sans images de transmission. Les images générées avec ces méthodes sont destinées à être utilisées comme une estimation approximative de l'AM pour corriger directement l'image PET ou pour servir d'image a priori spécifique au patient à utiliser dans la reconstruction mutuelle de l'activité et

de l'AM. En plus, il sera montré que le schéma présenté fonctionne sur les données de différents patients, lésions et scanners.

G.2 Synthèse de la carte d'atténuation pour la tomographie par émission de positrons avec $[^{18}]$F FDG à l'aide de réseaux génératifs antagonistes 3D

L'estimation correcte de la correction des cartes d'atténuation des images PET est essentielle pour leur bonne reconstruction, mais une mesure directe de cela signifie un rayonnement ionisant supplémentaire pour le patient. Une autre approche pour obtenir ces informations consiste à utiliser des méthodes d'analyse d'images. Ces méthodes créent une structure d'atténuation à partir d'une autre modalité d'imagerie, telle que l'imagerie par résonance magnétique (MRI) ou l'imagerie PET sans correction d'atténuation (NAC-PET). Cette traduction de la modalité d'imagerie est particulièrement difficile dans les images NAC-PET du corps entier, car les informations qu'elle présente sont incomplètes. Dans ce scénario, où la traduction d'images doit également combler les lacunes d'information, les réseaux antagonistes génératifs (GAN) sont particulièrement puissants. Les GAN dans les tâches de traduction de modalité d'imagerie ont été appliqués avec succès dans de nombreux domaines de l'imagerie médicale, y compris la synthèse de cartes d'atténuation à partir d'images PET. Cependant, la plupart des méthodes analysent la génération de cartes d'atténuation de l'MRI au scanner à l'aide de CNN [49] et GAN [184], en nécessitant une image MRI co-enregistrée contenant des informations non présentes dans l'image NAC-PET. La traduction de l'image PET (et NAC-PET) en CT continue d'être l'un des domaines les moins explorés, en particulier en tomographie du corps entier. Les études dans ce domaine particulier se concentrent sur la traduction PET-CT dans des images corrigées du crâne. Liu [13] propose l'utilisation d'une architecture 2D U-Net pour traduire les tomodensitogrammes du crâne de NAC-PET à CT, montrant des résultats prometteurs pour les régions crâniennes. Armanious [153] propose une application GAN généralisée composée de générateurs de type U-Net 2D sous la forme d'une cascade et d'un discriminateur qui évalue la perte perceptive et le style de l'image générée. Ils montrent la capacité de la topologie à traduire les études PET en CT, en utilisant uniquement des coupes axiales et, encore une fois, uniquement pour les régions crâniennes. Aucune des deux méthodes ne fournit d'informations sur leur capacité à traduire des images de corps entier, ce qui est un problème plus difficile à résoudre, car il y a plus de modes possibles dans les structures d'atténuation. La traduction d'images du corps entier a été étudiée par Dong [14] en utilisant un GAN enseigné avec une cohérence cyclique et par Armanious [15] en utilisant un GAN 2D basé sur un U-Net en forme de cascade, mais ils ont formé et testé leurs modèles uniquement sur des scanners PET de l'état de l'art avec capacité à mesurer le temps de vol (ToF). La reconstruction de l'AM a également été étudiée à l'aide de la reconstruction de la probabilité maximale

d'activité et de la carte d'atténuation (MLAA) [51], qui se révèle prometteuse dans les tomographies PET avec la capacité de mesurer le ToF uniquement, lorsqu'il est combiné avec des réseaux de neurones [186]. Cependant, sans utiliser le post-traitement, la sortie MLAA est bruitée, avec ou sans informations ToF [187]. La résolution du problème de la génération d'une carte d'atténuation directement à partir d'une image peut permettre aux tomographes sans capacité ToF d'utiliser des cartes d'atténuation synthétiques.

Nous présentons ici une topologie GAN 3D avec une perte mixte pour générer des sinogrammes d'atténuation du corps entier au moyen d'images CT artificielles à partir d'images NAC-PET. Puisque la dimensionnalité des volumes 3D est comparable aux images 2D haute résolution. Un apprentissage en deux parties est appliqué, en commençant par un apprentissage étiqueté supervisé, puis un bloc de perte de l'antagoniste est ajouté pour améliorer la résolution de l'image. Le modèle est enseigné avec des ensembles des données publiques disponibles dans les archives d'images de cancer [188], l'ensemble des données contient des séries d'images du corps entier CT, PET et NAC-PET co-enregistrées de Carcinomes Épidermoïdes de Tête et Cou (HNSCC) [189], acquis avec le tomographe Discovery ST/STE/RX General Electric. Huit différents ensembles des données sont utilisés pour le processus de test, contenant cinq types de carcinomes et plusieurs modèles de scanners: Discovery ST/STE/RX/LS/IQ/610/690, de General Electric y Biograph de Siemens. Seul le Discovery 690 a la capacité de mesurer ToF.

G.3 Matériaux, méthodes et théorie

G.3.1 Description de la topologie

Deux architectures sont testées dans ce travail. Tout d'abord un U-Net 3D de base, enseigné de manière entièrement supervisée. Deuxièmement, un GAN dont le générateur est composé du 3D U-Net de base avec des couches supplémentaires enseignées de manière antagoniste avec un critique convolutif (ou discriminateur). Pour réduire l'instabilité de l'apprentissage antagoniste, le gradient antagoniste ne s'écoule pas dans le U-Net 3D de base.

Générateur:

La représentation du modèle est visible sur la figure G.1. La section initiale du générateur est une topologie de type U-Net 3D [165] (le bloc *3D U-Net*). Le réseau U-Net a 5 niveaux de résolution, chacun composé de deux couches convolutives avec des filtres de taille $3 \times 3 \times 3$ et des unités de redresseurs (ReLU) comme activation. Chaque niveau de résolution a une connexion de saut entre le chemin de réduction de résolution et le chemin d'augmentation de résolution. Au lieu d'utiliser le rééchantillonnage convolutif, les changements de résolution sont effectués avec un rééchantillonnage tri-linéaire. Après chaque couche de convolution, une normalisation de voxel est appliquée sur les cartes d'entités, en divisant chaque voxel par $\sqrt{\frac{1}{p_c}\sum_{i=1}^{p_c}(\mathtt{v}_i)^2 + e}$, où p_c est le nombre de

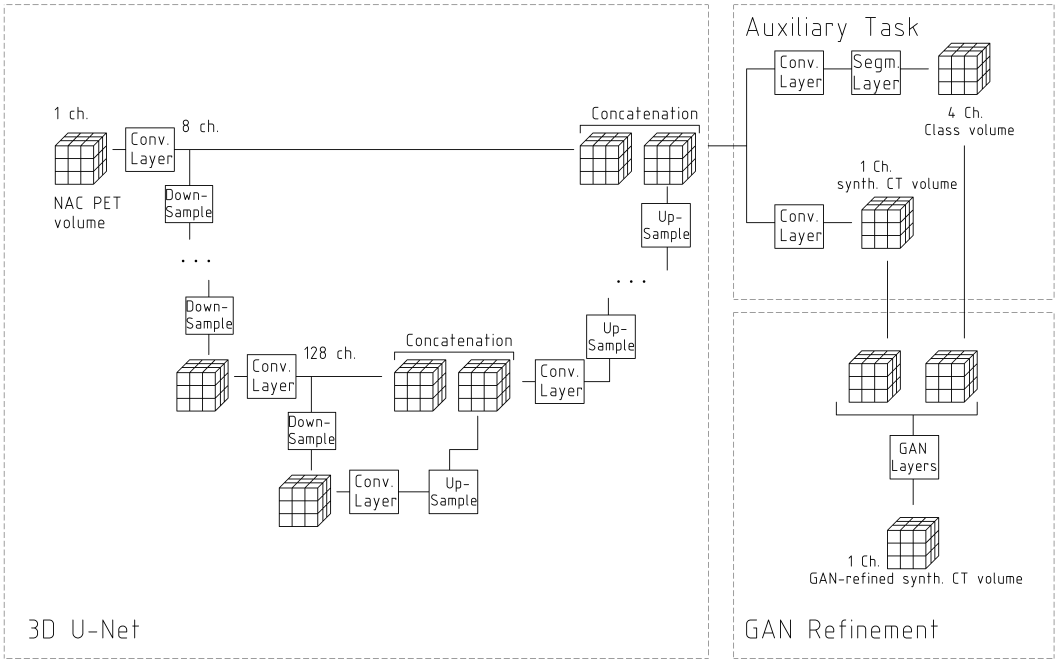


Figure G.1: Topologie du réseau générateur basé sur un U-Net 3D. La branche supérieure du bloc *tâche auxiliaire (auxiliary task)* est utilisée pour la segmentation et la branche inférieure pour la génération de CT artificiel. La sortie du bloc *raffinement par GAN (GAN Refinement)* est un CT synthétique raffiné par des couches GAN et construit à partir de sorties de bloc *tâche auxiliaire (auxiliary task)*.

canaux sur la carte des caractéristiques, $\mathtt{v}_i$ est la valeur du $i^{ème}$ voxel et $e = 1.0 \times 10^{-8}$. Un facteur de mise à l'échelle est également appliqué à chaque noyau de convolution en fonction de la mise à l'échelle d'initialisation des poids He [207]. Après le bloc 3D U-Net, le réseau est divisé en deux branches dans le bloc *Auxiliary Task*. La première branche est utilisée pour segmenter et est composée de trois couches convolutives et se termine par une couche de fonction exponentielle normalisée (softmax). La deuxième branche est chargée de générer un CT synthétique (sCT), il est composé d'une couche convolutive avec une activation tangente hyperbolique. Enfin, les sorties 3D U-Net sont fusionnées et traitées par les couches GAN dans le bloc *GAN Refinement*. Ce dernier bloc avec 8 filtres dans chaque couche, est utilisé lors de l'apprentissage de l'antagoniste. Toutes les opérations de convolution utilisent un filtre de taille $3 \times 3 \times 3$ à l'exception de la couche de sortie, qui a un filtre de taille $1 \times 1 \times 1$.

Critique:

Le réseau critique ou discriminateur est un réseau convolutif normal avec une activation de type ReLU dans toutes ses couches, seule la dernière couche n'a pas d'activation. L'entrée de ce réseau est un volume à deux canaux: l'image NAC-PET et l'image CT réelle ou sCT. La sortie réseau est une valeur proportionnelle à la qualité de

l'image générée. Le réseau est composé de 4 niveaux de résolution avec deux couches convolutives par niveau. Chaque filtre de convolution a une taille de $3 \times 3 \times 3$ et a une activation de type ReLU. Aucune normalisation de lot ou de pixel n'est appliquée. Les deux dernières couches de la réseau critique sont une opération d'aplatissement suivie d'une seule couche dense avec une sortie linéaire.

G.3.2 Schème de apprentissage

L'apprentissage du réseau est divisée en deux étapes. En premier lieu, le générateur est enseigné de manière supervisée à l'aide d'une fonction objectif composite. La branche de segmentation du réseau applique une perte de type DICE 3D comme indiqué dans l'équation G.1

$$\mathcal{L}_D = \frac{1}{p_c} \sum_{i=1}^{p_c} \frac{2 \sum_{j=1}^{V} \hat{\mathtt{v}}_{j,i}^{\mathtt{TL}} \mathtt{v}_{j,i}^{\mathtt{TL}}}{\sum_{j=1}^{V} (\hat{\mathtt{v}}_{j,i}^{\mathtt{TL}})^2 + \sum_{j}^{V} (\mathtt{v}_{j,i}^{\mathtt{TL}})^2}, \tag{G.1}$$

où p_c est le nombre de classes objectif, V est le nombre de voxels dans le volume, $\mathtt{v}_{j,i}^{\mathtt{TL}}$ sont les valeurs des voxels objectif et $\hat{\mathtt{v}}_{j,i}^{\mathtt{TL}}$ les valeurs de la sortie réseau. Le DICE 3D varie entre 0 et 1. Il produit un maximum lorsque tous les voxels objectifs ($\mathtt{v}_{j,i}^{\mathtt{TL}}$) ont la même valeur que la sortie réseau ($\hat{\mathtt{v}}_{j,i}^{\mathtt{TL}}$). Sur la sortie du réseau a été appliquée l'opération softmax, le dénominateur de l'équation est toujours supérieur au numérateur sauf lorsque $\mathtt{v}_{j,i}^{\mathtt{TL}}$ et $\hat{\mathtt{v}}_{j,i}^{\mathtt{TL}}$ sont identiques. Dans le cas d'un problème multi-classes ($p_c > 1$), la valeur finale est divisée par le nombre de classes.

La branche de synthèse CT, jusqu'aux couches GAN, est enseignée en utilisant la distance euclidienne entre la sCT et l'image CT objectif, $\mathcal{L}_T = ||\boldsymbol{V}^{\mathtt{sCT}}, \boldsymbol{V}^{\mathtt{CT}}||^2$, où $\boldsymbol{V}^{\mathtt{sCT}}$ est le CT synthétique et $\boldsymbol{V}^{\mathtt{CT}}$ le CT réel. La fonction de perte pour l'apprentissage supervisée est montré dans l'équation G.2

$$\mathcal{L}^{\mathtt{U-Net}} = (1 - \mathcal{L}_D) + k_e \mathcal{L}_T, \tag{G.2}$$

où k_e est une constante de couplage.

En deuxieme lieu, après l'apprentissage initial, l'apprentissage antagoniste commence. L'apprentissage antagoniste utilise la stratégie du Wassersterin-GAN (W-GAN) [177], résultant en une perte de générateur comme indiqué dans l'équation G.3,

$$\mathcal{L}_W = -\mathtt{Crit}_W(\mathtt{Gen}_W(\boldsymbol{V}^{\mathtt{NPT}}), \boldsymbol{V}^{\mathtt{NPT}}), \tag{G.3}$$

où $\mathtt{Crit}_W()$ est la fonction du réseau critique, $\mathtt{Gen}_W()$ est la fonction du réseau génératrice et $\boldsymbol{V}^{\mathtt{NPT}}$ est l'image NAC-PET d'entrée. Pendant l'apprentissage antagoniste, les couches GAN sont activées et enseignées à l'aide de la perte W-GAN. Le gradient GAN ne coule pas dans les couches 3D U-Net. Le réseaux critique est formé à l'aide d'une paire d'images NAC-PET et CT couplées réelles ou générées. Il est formé en utilisant

la perte de wasserstein indiquée dans l'équation G.4,

$$J_{crit.}^{GAN} = \frac{1}{N_m^{true}} \sum_{n=1}^{N_m^{true}} \texttt{Crit}_W(\boldsymbol{V}_n^{\texttt{NPT}}, \boldsymbol{V}_n^{\texttt{CT}}) - \frac{1}{N_m^{fake}} \sum_{n=1}^{N_m^{fake}} \texttt{Crit}_W(\boldsymbol{V}_n^{\texttt{NPT}}, \texttt{Gen}_W(\boldsymbol{V}_n^{\texttt{NPT}})) + k_{\texttt{gp}} \frac{1}{N_m^{true}} \sum_{n=1}^{N_m^{true}} \nabla_{\texttt{GP}}(\texttt{Gen}_W(\boldsymbol{V}_n^{\texttt{NPT}}), \boldsymbol{V}_n^{\texttt{CT}}), \quad \text{(G.4)}$$

où $\boldsymbol{V}_n^{\texttt{CT}}$ est le CT real du $n^{\text{ème}}$ échantillon , $\boldsymbol{V}_n^{\texttt{NPT}}$ es le $n^{\text{ème}}$ entrée NAC PET, $\nabla_{\texttt{GP}}$ est la pénalité sur le gradient [179] et $k_{\texttt{gp}} = 10.0$. Le réseau critique est enseigné sur 5 étapes pour chaque étape du générateur. Dans l'étape initiale de l'apprentissage GAN, le critique est enseigné à son optimum avant d'initialiser la boucle d'apprentissage GAN. Les générateurs sont enseignés à l'aide de l'optimiseur Adaptive Moment Estimation (ADAM) avec les paramètres $\beta_1 = 0.0$, $\beta_2 = 0.99$ et $\epsilon = 1.0 \times 10^8$ et le taux d'apprentissage $lr = 0.0001$. Le discriminateur utilise l'optimiseur RMSprop avec un taux d'apprentissage de $lr = 0.0005$.

G.3.3 Description de l'ensemble des données d'apprentissage

L'ensemble des données HNSCC comprend une série d'études CT et NAC-PET co-enregistrées sur les carcinomes épidermoïdes de la tête et du cou. Les échantillons sélectionnés à partir de l'ensemble des données consistent en des études sur le corps entier. L'ensemble des données a été initialement dépouillé de tous les études qui ne contenaient pas d'échantillons NAC-PET et CT co-enregistrés. Ensuite, tous les échantillons ont été testés par chevauchement et rognés sur les coupes axiales contenant des informations provenant des deux modes d'image. Une fois l'ensemble des données nettoyé, il est normalisé à une taille FoV de $128 \times 128 \times 256$ voxels avec une taille de voxel de $5.46 \times 5.46 \times 5.08$ $\texttt{mm}^3$. L'ensemble final des données contient 118 images de 71 patients différents, dont 7 patients (10 images) ont été séparés comme données de validation. Avant d'alimenter chaque échantillon vers le réseau, le volume est découpé au aléatoirement en un volume de $128 \times 128 \times 32$ voxels et tous les voxels du NAC-PET se voient ajouter une valeur de $\pm 10\%$ la valeur pleine échelle et sont re-normalisés.

Normalisation CT cible:

Les images initiales font apparaître le lit sur lequel repose le patient. Pour éliminer cette structure une méthode basée sur la variance des voxels le long de l'axe axial [199] peut être utilisée. Ensuite, la plage dynamique de l'image est rognée entre -125 et 1300 unités Housfield (HU) et normalisée entre 0 et 1, pour maximiser la distance entre les tissus mous et les os.

Génération d'étiquettes:

Quatre classes d'étiquettes sont extraites des images CT non normalisées, sans la structure du lit, en utilisant des seuils dans les valeurs des voxels. Le masque *Air-*

Poumon est situé entre -1000 HU et -125 HU, Le masque *Fluides-Graisse* est situé entre -125 HU et 10 HU, Le masque *Tissu Mou* est situé entre 10 HU et 90 HU et Le masque *Os* est situé entre 90 HU et 1300 HU.

G.3.4 Description de l'ensemble des données de test

Les ensembles des données de test sont une série d'ensembles des données publiques, venant également du TCIA, qui incluent différents types de lésions, de patients et de technologies de tomographie. L'ensemble des données est composé de 133 échantillons de test: 73 à partir de l'ensemble des données de Carcinome Bronchique à Cellules Non Pétites (NSCLC) [191], 25 à partir de l'ensemble des données d'Atlas du génome du cancer - Carcinomes Épidermoïdes de Tête et Cou (TCGA-HNSC) [192], 20 à partir de l'ensemble des données d'Atlas du génome du cancer - Adénocarcinome Pulmonaire (TCGA-LUAD) [193], 1 à partir de l'ensemble des données d'Atlas du génome du cancer - Cancer de la Thyroïde (TCGA-THCA) [194], 4 à partir de l'ensemble des données de Consortium pour l'Analyse Préotonique Clinique des Tumeurs - Adénocarcinome Pulmonaire (CPTAC-LUAD) [195], une à partir de l'ensemble des données de Consortium pour l'Analyse Préotonique Clinique des Tumeurs - Adénocarcinome Canalaire Pancréatique (CPTAC-PDA) [196], 3 à partir de l'ensemble des données de Consortium pour l'Analyse Préotonique Clinique des Tumeurs - Cancer de l'Endomètre Utérin (CPTAC-UCEC) [197] et 6 à partir de l'ensemble des données de Consortium pour l'Analyse Préotonique Clinique des Tumeurs - Carcinomes Épidermoïdes Pulmonaire (CPTAC-LSCC) [198]. Ces ensembles des données ont été nettoyés des échantillons non enregistrés et normalisés pour s'adapter à l'entrée réseau. Pendant le processus de test, le NAC-PET est donné au réseau neuronal en tranches consécutives de $128 \times 128 \times 32$ voxels. L'intensité est remise à $[0, 1]$ pour chaque coupe (aucune autre normalisation n'est appliquée). Le sCT résultant est composé d'un seul volume en utilisant une opération de moyenne pondérée.

G.3.5 Mesures de correction d'atténuation

Les sinogrammes des ensembles des données n'étant pas disponibles, une reconstruction directe de l'activité PET n'est pas possible en utilisant le modèle proposé. En tant qu'évaluation quantitative de la performance de la correction d'atténuation, l'atténuation sur une ligne de réponse (LoR) a été comparée en utilisant le CT fourni et le sCT généré. L'atténuation dans les LoR a été construite en utilisant des sinogrammes d'atténuation de taille (180×180) et avec un pas axial de 5 mm. Chaque CT et sCT a été initialement converti des unités HU en atténuation linéaire $[1/cm^2]$, en 120 keV. Ensuite, l'atténuation linéaire a été convertie en énergie PET 511 keV. En utilisant le CT réel et le sCT, une paire de sinogrammes d'atténuation a été créée. Pour chacun des LoR dans les sinogrammes, le coefficient d'atténuation a été calculé comme $x_l^\mu = e^{-\sum_{u=1}^{U} h_{l,u}^\mu \mu_u}$. En utilisant le CT fourni comme objectif, la différence entre les sinogrammes d'atténuation a été mesurée en utilisant la différence moyenne et l'écart type des valeurs du sinogramme. Cela a été appliqué à quatre échantillons, HNSCC-

01-0148 de l'ensemble des données de validation, échantillon AMC-009 de NSCLC Radiogenomics, C3N-00957 de CPTAC-PDA et TCGA-BB-7863 de TCGA-HNSC.

G.3.6 Mesures de qualité d'image

La qualité d'image des sCT générés a été testée par rapport à la CT réelle en utilisant trois métriques: le rapport signal au bruit de crête (PSNR), l'erreur absolue moyenne (MAE) et la corrélation croisée normalisée (NCC).

G.4 Résultats

G.4.1 Correction d'atténuation

Les métriques résultant des sinogrammes d'atténuation sont résumées dans le tableau G.1. Un anneau central du sinogramme d'atténuation du CT de référence et du sCT est représenté sur la figure G.2 pour l'échantillon de validation HNSCC-01-0148 et sur la figure G.3 pour l'échantillon AMC-009 de l'ensemble des données NSCLC Radiogenomics. Les sinogrammes sont affichés avec les histogrammes de différence de valeur pour tous les sinogrammes FoV, montrant la forme de la distribution d'erreur présentée dans le tableau G.1.

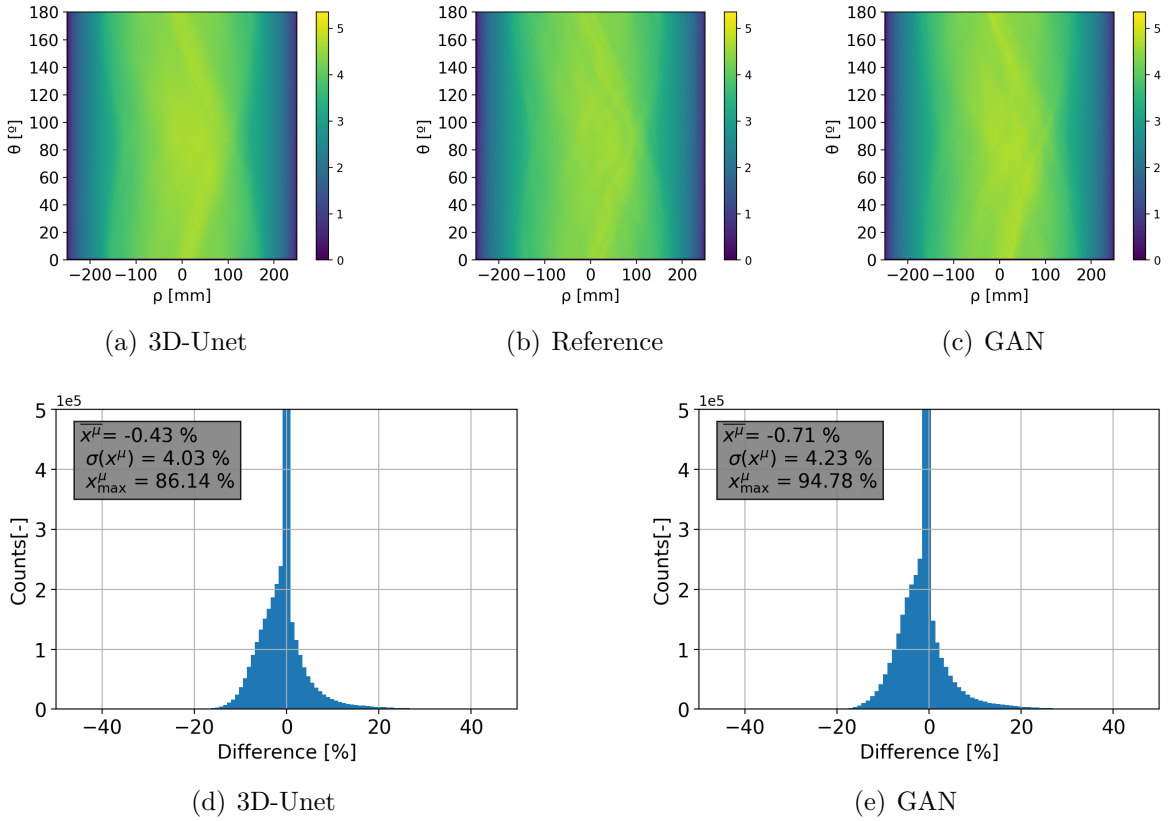
Topologie	HNSCC-01-0148	AMC-009	C3N-00957	TCGA-BB-7863
U-Net	$-0.43 \pm 4.0\%$	$-0.53 \pm 5.5\%$	$0.34 \pm 6.3\%$	$-0.49 \pm 3.6\%$
GAN	$-0.71 \pm 4.2\%$	$-0.65 \pm 5.8\%$	$0.41 \pm 7.7\%$	$-0.66 \pm 3.9\%$

Table G.1: Écart moyen et standard de la différence d'atténuation entre le CT réel et le sCT généré, en pourcentage de la valeur de référence, pour un échantillon de validation et trois échantillons de test. Une bonne métrique sera proche de zéro et aura un faible écart type.

G.4.2 Qualité d'image

Les métriques PSNR, MAE et NCC du sCT généré par rapport au CT cible sont présentées dans la figure G.4, chacun des diagrammes correspond aux métriques 3D U-Net et GAN dans différents ensembles des données. Les ensembles des données sont présentés par leur source car le nombre d'échantillons dans les ensembles individuels est très faible dans certains cas. Les valeurs de l'ensemble de test et de validation sont résumées dans le tableau G.2.

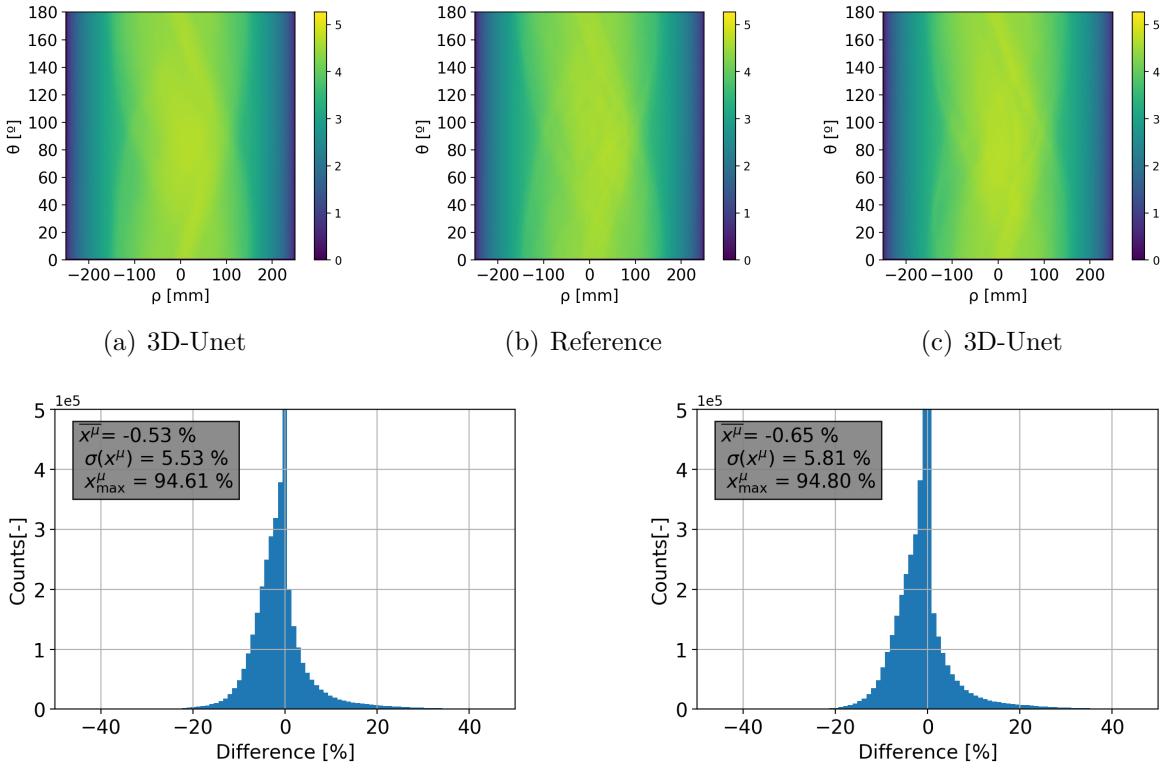
Trois échantillons de trois ensembles des données peuvent être vus dans les figures G.5, G.6, et G.7, deux avec des patients avec leurs bras levés au-dessus de la tête (bras vers le haut) et un autre avec leurs bras positionnés à côté du corps (bras vers le bas). Ces images correspondent respectivement aux ensembles NSCLC Radiogenomics, CPTAC-PDA et TCGA-HNSC. Sur ces figures, les images sCT générées



(a) 3D-Unet (b) Reference (c) GAN

(d) 3D-Unet (e) GAN

Figure G.2: Anneau médian des sinogrammes d'atténuation pour le 3D U-Net sCT (a), le CT de référence (b) et le GAN sCT (c) de l'échantillon HNSCC-01-0148 de l'ensemble des données de validation. Les histogrammes (d, e) de la différence des valeurs d'atténuation, respectivement la sCT du 3D U-Net et la sCT du GAN 3D.



(a) 3D-Unet (b) Reference (c) 3D-Unet

(d) 3D-Unet (e) GAN

Figure G.3: Anneau médian des sinogrammes d'atténuation pour le 3D U-Net sCT (a), le CT de référence (b) et le GAN sCT (c) de l'échantillon AMC-009 de l'ensemble des données NSCLC Radiogenomics. Les histogrammes (d, e) de la différence des valeurs d'atténuation, respectivement la sCT du 3D U-Net et la sCT du GAN 3D.

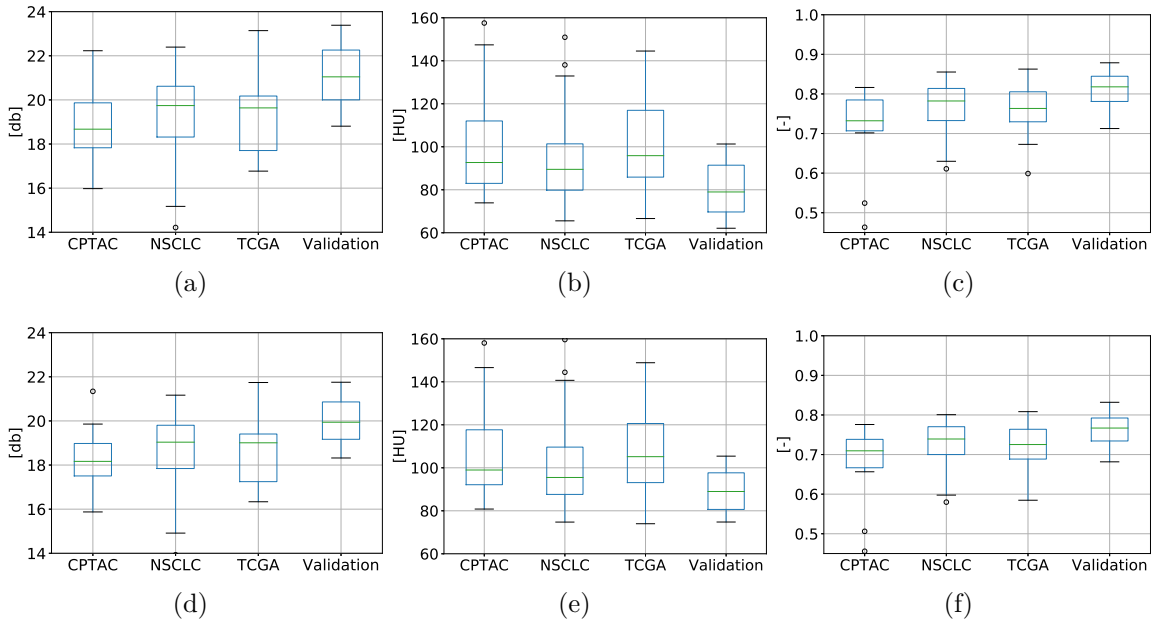


Figure G.4: Les intervalles de mesures du PSNR (a, d), MAE (b, e) et NCC (c, f) pour chaque source de l'ensemble des données de test et de validation. Les figures (a, b, c) et (d, e, f) correspondent respectivement au U-Net 3D et au réseau avec apprentissage antagoniste.

uniquement avec la fonction de perte supervisée indiquée dans l'équation G.2, et les sCT générées avec la perte antagoniste indiquée dans l'équation G.3 sont comparés par rapport au CT de référence. Egalement sur ces figures, une vue 3D du squelette généré par chaque topologie et la structure de référence est présentée.

G.5 Discussion

La capacité de correction d'atténuation du réseau a été testée en utilisant une série de sinogrammes d'atténuation et en comparant les sinogrammes des sCT aux sinogrammes du CT réel. Cette métrique a atteint une erreur moyenne inférieure à 1% et un écart type de 8% dans le pire des cas, comme le montre le tableau G.1. Cette métrique peut être améliorée en utilisant des ensembles des données plus grands et plus uniformes (en termes de couverture anatomique). Alors que l'ajout de couches GAN améliore l'aspect visuel des sCT, une amélioration de la correction d'atténuation n'est pas perçue. Contrairement au résultat attendu, le modèle entièrement supervisé semble être plus adapté. Notre technique s'avère robuste pour plusieurs types de technologies de reconstruction et de tomographie lorsqu'elle fonctionne à la résolution sélectionnée, comme le montrent les mesures de test de la figure G.4. La topologie U-Net 3D de base génère des images d'atténuation synthétiques avec un PSNR de 19.3 ± 1.7 `dB`, un MAE de 97 ± 20 `HU` et un NCC de 0.760 ± 0.064. L'inclusion des couches GAN permet d'obtenir un PSNR de 18.6 ± 1.4 `dB`, un MAE de 103 ± 18 `HU` et un NCC de

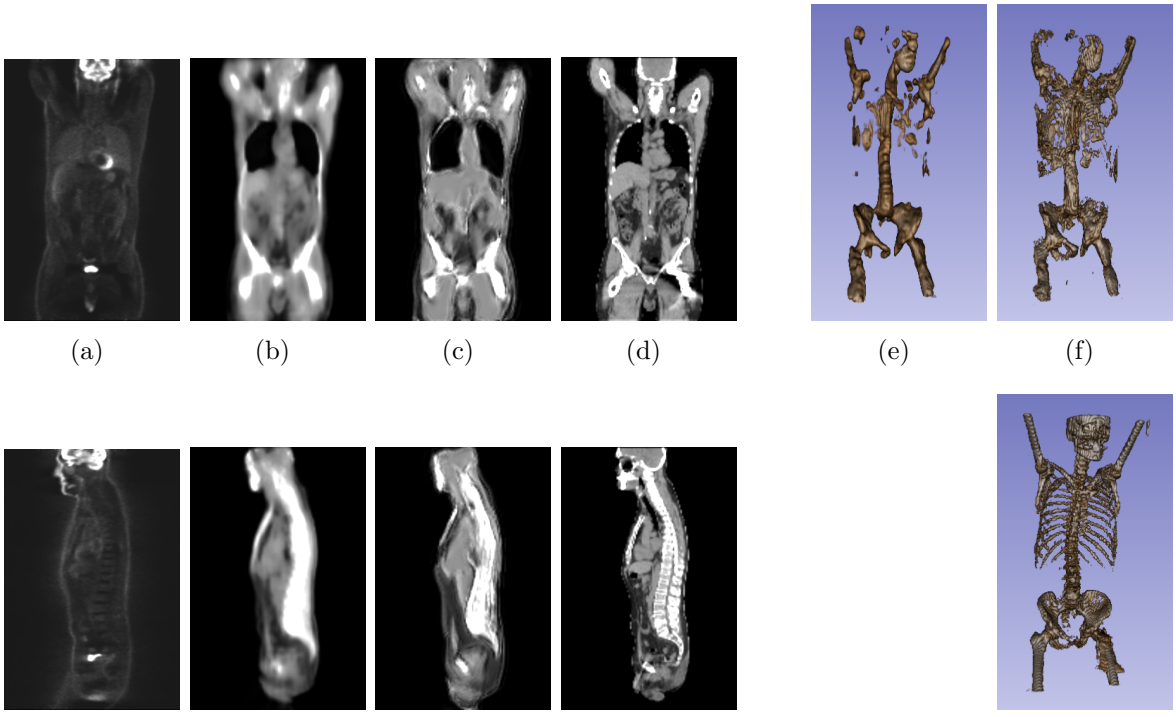


Figure G.5: Entrée NAC-PET (a, g), 3D-Unet sCT (b, e, h), 3D GAN sCT (c, f, i) et CT de référence (d, j, k) coronale (a, b, c, d), sagittale (g, h, i, j) et perspective 3D du squelette (e, f, k) de l'échantillon AMC-009 de l'ensemble NSCLC Radiogenomics. L'amélioration du GAN se traduit par la diminution du flou entre les images (b, c) et (h, i). En perspective 3D, le réseau GAN (f) est capable de générer plus de détails que U-Net 3D (e), par rapport à la référence (k).

	Fuente	PSNR [dB]	MAE [HU]	NCC [−]
U-Net	Validation	21.00 ± 1.4	80 ± 13	0.80 ± 0.050
	Test	19.3 ± 1.7	96 ± 20	0.760 ± 0.060
GAN	Validation	19.90 ± 1.3	89 ± 10	0.76 ± 0.050
	Test	18.6 ± 1.5	103 ± 18	0.720 ± 0.060

Table G.2: Valeurs des ensembles des données de validation (10 échantillons) et de test (133 échantillons) pour U-Net 3D et topologie GAN proposée.

0.720 ± 0.059. Ces valeurs sont obtenues dans des échantillons provenant de différents tomographes, patients et lésions, montrant que la technique peut être utilisée avec différentes sources. On peut voir que le réseau surveillé génère des images floues tandis que le GAN montre des détails de fréquence plus élevée comme le montre la figure G.5. Cependant, ces améliorations ne sont pas reflétées dans les métriques, ce qui entraîne des résultats inférieurs mais pas significativement différents de ceux obtenus par la base U-Net 3D. Le U-Net 3D et le réseau antagoniste ne parviennent pas à se généraliser au haut du corps où moins des données d'apprentissage étaient disponibles. Cela se reflète également dans les bras, qui apparaissent à différentes positions dans l'ensemble des données. Bien que cela puisse être résolu avec plus des données d'apprentissage, il peut également être atténué en faisant correspondre anatomiquement les données d'apprentissage et les réseaux spécifiques à la région d'apprentissage. Une amélioration supplémentaire dans cette direction peut être obtenue en apprissant un réseau dans un espace intermédiaire de taille réelle pour cartographier chaque section du corps à un espace spécifique, comme le font les représentations intermédiaires présentées dans [211] et [212]. La génération actuelle de sCT peut également être utilisée comme image a priori pour la reconstruction de l'atténuation à l'aide de techniques telles que MLAA et la modélisation de dispersion simple [56], réduisant son coût de calcul élevé. Ces techniques peuvent potentiellement supprimer des artefacts des cartes d'atténuation générées, comme le contraste d'estomac CT vu sur la figure G.7(d) qui n'est pas présent sur la figure G.7(b) ou la figure G.7(c).

Les métriques obtenues sont comparables à la méthode proposée par Dong [213], qui atteint un MAE de 108 ± 19 HU dans la reconstruction d'une image sCT. Cependant, le modèle présenté est plus compact que l'architecture GAN cyclique proposée par eux, nécessitant moins de paramètres et contrôlant également le gradient, ce qui se traduit par un apprentissage plus stable. Notre travail est également testé sur plusieurs tomographes PET. En outre, la plupart des TC testés n'ont pas de capacités de mesure ToF, de sorte que des méthodes telles que MLAA rendre des cartes d'atténuation du corps entier de qualité inférieure. Enfin, pour pouvoir faire des évaluations directes entre les différents modèles et évaluer leur complexité, il est nécessaire de disposer d'un ensemble des données commun. Pour cette raison, l'ensemble des données utilisé dans ce travail est publié [1].

[1] https://github.com/RawthiL/PET_DeepAttCorrection

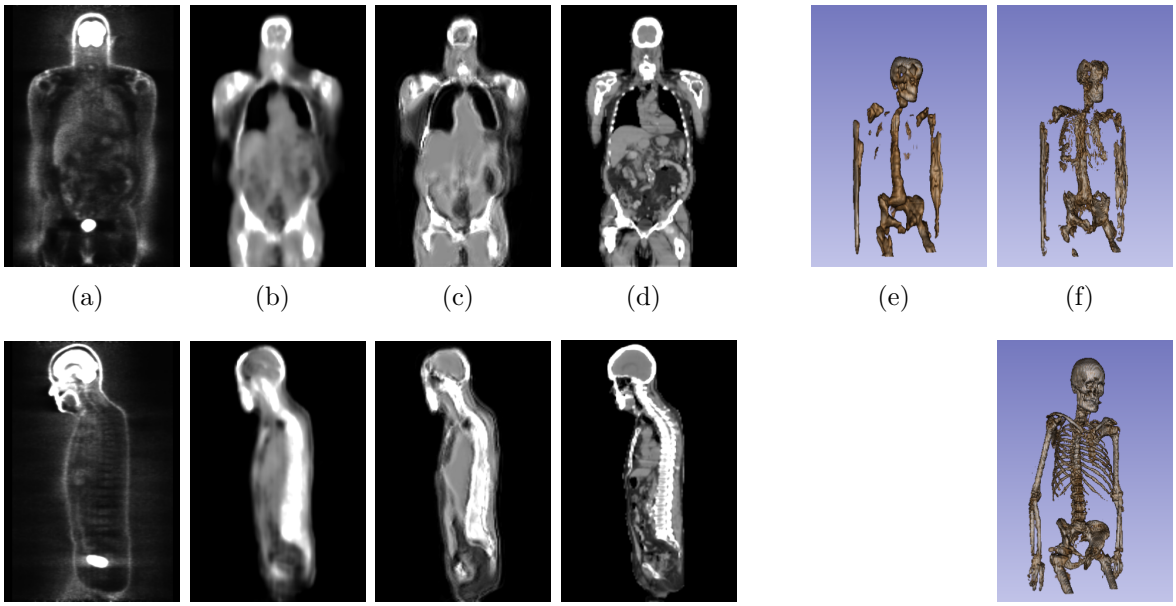


Figure G.6: Entrée NAC-PET (a, g), 3D-Unet sCT (b, e, h), 3D GAN sCT (c, f, i) et référence CT (d, j, k) coronale (a, b, c, d), sagittale (g, h, i, j) et perspective 3D du squelette (e, f, k) de l'échantillon C3N-00957 de l'ensemble CPTAC-PDA.

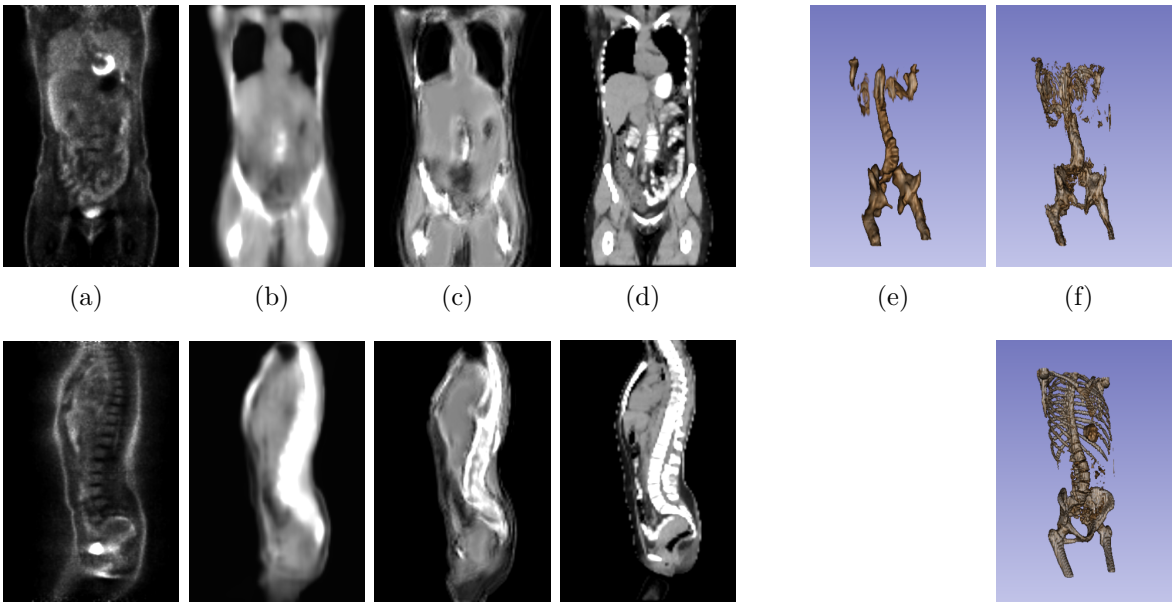


Figure G.7: Entrée NAC-PET (a, g), 3D-Unet sCT (b, e, h), 3D GAN sCT (c, f, i) et CT de référence (d, j, k) coronale (a, b, c, d), sagittale (g, h, i, j) et perspective 3D du squelette (e, f, k) de l'échantillon TCGA-BB-7863 de l'ensemble TCGA-HNSC.

H Conclusions finales et perspectives

H.1 Discussion principale

Tout au long de cette thèse, les différents aspects de la tomographie PET ont été étudiés, en se concentrant sur l'amélioration de la tomographie à faible dose. Le problème de la réduction des doses chez le patient lors d'une PET a été analysé sur trois aspects.

Le prototype AR-PET

Dans une première partie, l'optimisation du scanner AR-PET a été étudiée. Ce prototype, conçu pour permettre l'accès à cette technologie aux communautés à faible revenu, se compose de gros cristaux solides et ne dispose pas d'équipement de mesure d'atténuation. Les caractéristiques de l'AR-PET ont joué un rôle central dans les chemins de recherche empruntés au cours de cette thèse. Plus précisément, deux problèmes ont été étudiés pour ce scanner: d'abord la correspondance des gains des PMT avec les caméras a été analysée et ensuite la localisation des événements dans le cristal scintillateur. Dans la section E.3 l'appariement des PMT avec les caméras de scintillation a été analysé et a abouti à une optimisation appliquée lors de la construction de l'AR-PET. Dans cette étude, l'optimisation de la dispersion du gain des PMT dans différentes conditions de travail a été étudiée et un modèle simplifié de la chambre AR-PET a été construit. Ce modèle, et une procédure d'optimisation basée sur des algorithmes génétiques, ont permis la sélection de groupes de PMT pour la construction des chambres de scintillation AR-PET sous contraintes. L'application de cette technique a permis d'améliorer l'efficacité de base des caméras. Bien que cette étude ait été testée dans la construction du tomographe AR-PET, le concept de caractérisation PMT et de modélisation de la chambre de scintillation peut être appliqué à tout type de chambre de scintillation à cristal solide.

La localisation des photons gamma dans le cristal solide a été étudiée dans la section E.5, où un nouvel algorithme de localisation basé sur l'appariement des distributions avec les réseaux de neurones a été développé. La principale contribution de ce travail a été le développement de l'algorithme avec la capacité d'estimer la profondeur d'interaction, qui ne nécessite pas un ensemble des données complexes pour la formation. Alors que dans les méthodes actuelles de réseaux de neurones, l'ensemble des données doit contenir l'emplacement exact de chaque événement (ce qui est normalement difficile à obtenir), la méthode développée n'a besoin que de connaître la distribution spatiale de ces événements, en ignorant l'emplacement particulier de chacun d'eux. Ainsi la méthode proposée peut améliorer les défauts d'une méthode de

localisation de base en faisant correspondre la distribution obtenue par la méthode de base et la distribution attendue lors de l'acquisition. La méthode a été appliquée pour corriger le positionnement des événements dans les caméras AR-PET, améliorant le positionnement par centre de gravité. La zone sensible des caméras a été améliorée de plus de 23% et la capacité d'estimer la profondeur d'interaction a été incluse.

Images PET sur un maillage support

Le deuxième problème abordé était la réduction du nombre de paramètres utilisés pour décrire la distribution du radio-traceur dans une image PET. Plus précisément, la représentation volumétrique des images PET avec maillages a été étudiée. Cette réduction des paramètres est importante pour deux raisons: d'une part, elle peut se traduire par une amélioration de la vitesse de reconstruction et d'autre part elle réduit la dimensionnalité des données, qui peut être utilisée dans algorithmes de reconstruction basés sur des techniques d'optimisation. Pour tester ces hypothèses dans la section F.2, une reconstruction améliorée basée sur le maillage a été développée. Au lieu d'utiliser des maillages 3D polygonaux pour décrire le volume, une série de maillages 2D représentant des tranches du volume a été utilisée. Cela a été possible car les informations utilisées pour reconstruire les images PET, les LoR, sont limitées à certains angles, donc le LoR ne peut jamais traverser une tranche à un angle perpendiculaire à celle-ci. Cette façon de décrire le volume permet d'affiner les maillages 2D, qui ont moins de nœuds que les maillages 3D. En outre, l'amélioration des maillages peut être entièrement parallélisé, ce qui se traduit par des temps de traitement plus courts. La technique développée a été mise en œuvre à l'aide d'un algorithme de traçage LoR personnalisé, utilisé pour modéliser l'acquisition du tomographe AR-PET et du logiciel de remaillage BL2D-V2. La méthode développée a été testée en utilisant une simulation sans erreur d'un fantôme complexe du cerveau humain et une acquisition d'un fantôme de qualité d'image NEMA à l'aide du prototype préclinique AR-PET. Ces fantômes ont été reconstruits à l'aide d'un algorithme MLEM basé sur un maillage. Grâce à ces expériences, il a été possible de montrer que l'application d'une série de maillages 2D pour décrire une dose volumétrique n'entraîne pas de perte de qualité d'image tout en réduisant la quantité d'éléments nécessaires pour décrire l'image volumétrique de plus d'un ordre de grandeur quand on compare à un support voxélisé. Cette réduction est encore plus notable lorsque des images voxel haute résolution sont utilisées. Cependant, en utilisant un algorithme de reconstruction simple tel que MLEM, la qualité des images décline plus rapidement dans le support de maillage que dans le support voxélisé lorsque le nombre de LoR utilisé dans le processus de reconstruction diminue. Cependant, la diminution du nombre de nœuds peut être utilisée pour améliorer la qualité du modèle de reconstruction, à des coûts inférieurs à ceux d'un support voxélisé, et pour atténuer l'effet des faibles statistiques.

Génération de cartes d'atténuation à l'aide de méthodes d'apprentissage automatique

Le troisième et dernier sujet abordé dans cette thèse était la génération de cartes d'atténuation pour les images PET au moyen de méthodes ML. Ce chapitre s'est concentré sur l'analyse des différentes stratégies pour générer une carte d'atténuation (AM) en utilisant uniquement l'image NAC-PET et en appliquant des outils générateurs de ML d'État de l'art. Pour explorer les méthodes ML, un ensemble d'images NAC-PET co-enregistrées avec des études CT a été généré. L'ensemble des données a été créé avec des images publiques de la base des données TCIA. La topologie choisie pour implémenter la génération AM était un réseau neuronal profond basé sur une version 3D de l'U-Net. Deux variantes ont été étudiées dans la section F.2, chacune enseigné d'une manière différente: un apprentissage entièrement supervisé et un apprentissage antagoniste. La formation entièrement supervisée était capable de générer une AM avec une qualité relativement bonne et était également capable de segmenter différents types de tissus. Cependant, les images générées présentaient une perte d'information dans les hautes fréquences. Il fallait s'y attendre puisque la formation utilisait une métrique de type L2 comme fonction objectif. Pour améliorer la qualité des images, une série de couches d'apprentissage antagoniste ont été ajoutées au réseau, ce qui a constitué une formation hybride. Ce réseau combine un apprentissage supervisé avec un réseaux antagoniste utilisé pour améliorer les détails à haute fréquence. En conséquence, les images présentent des caractéristiques plus réalistes sans perdre la spécificité du réseau surveillé. L'utilisation du réseau hybride permit de reproduire des métriques d'état de l'art. En outre, le réseau a été testé dans différents modèles de tomographie PET, de procédures d'administration de dose et de processus de reconstruction d'image, montrant toujours de bons résultats. L'AM générés sont de qualité suffisante pour être appliqués directement à la correction d'images PET selon une étude récente. Cependant, il a été observé que l'existence de structures étrangères (telles que des prothèses, des contrastes ou des calcifications) peut être supprimée par erreur des images AM générées. Pour cette raison, il est conseillé d'affiner l'AM. Ce raffinement peut être effectué en utilisant une reconstruction mutuelle de l'activité et de l'atténuation et en prenant le AM généré comme image a priori.

H.2 Perspectives et travaux futurs

Le chemin vers un système PET à faible coût sans rayonnement supplémentaire pour les études de transmission est long et cette thèse a fait quelques pas vers cet objectif. Cependant, il reste encore beaucoup à faire. Les principes de base d'une reconstruction d'activité et d'atténuation ont été développés lors de ce travail, une représentation et une reconstruction optimisées pour les images PET et une méthode de génération AM a priori basée sur des images de haute qualité. S'il n'a pas été possible de développer davantage le potentiel de la représentation de maillage pour la reconstruction des cartes d'activité et d'atténuation (en raison de contraintes de temps), la base de la reconstruction de l'activité sur les maillages a été développée en même temps que

la génération de AM avec des réseaux de neurones profonds. Ces travaux sont des pièces fondamentales pour les algorithmes de reconstruction mutuelle d'activité et d'atténuation. L'extension de ces travaux comprend l'extension des modèles de reconstruction développés pour la reconstruction mutuelle de l'activité et de l'AM, mettant en œuvre un MLAA à maillage, suivant [11, 51, 52]. Une autre voie prometteuse est l'optimisation de la méthode de simulation de dispersion simple (SSS) associée aux correspondances à faible énergie (diffusées) et l'utilisation d'une image AM a priori de haute qualité pour réduire les problèmes de convergence [54, 55]. De plus, en utilisant la dimensionnalité réduite de la représentation de maillage, le modèle de reconstruction amélioré la qualité des images PET à faible dose [19].

La génération de AM par les méthodes ML peut être améliorée sous plusieurs aspects. Les modèles de génération bénéficient grandement des données cohérentes. Cela fait partie du succès des grands réseaux génératifs comme [216]. L'appariement histologique des échantillons de jeux des données peut améliorer l'efficacité de la formation et donc la qualité des AM générés. Une autre amélioration du modèle peut être l'utilisation d'un espace latent intermédiaire à haute résolution à partir duquel le modèle génératif peut échantillonner. Ceci est proposé en [212] où ils mettent en œuvre cet espace latent pour améliorer la représentation 3D des images. Le développement de nouveaux outils d'analyse des données 3D, tels que [172]m doivent être testés. Ces outils permettent la mise en œuvre rapide de modèles 3D qui analysent les données voxélisées ou les nuages de points. L'analyse de nuages de points, telle que [167–170], peut être directement implémenté sur les nœuds d'un maillage, réduisant la dimensionnalité et évitant les erreurs de projection.

Enfin, on s'attend également à ce que la génération de AM puisse être testée dans les données des patients à partir de l'AR-PET. Les travaux présentés dans cette thèse ont aidé à la construction et aux tests de l'AR-PET. Le prototype du tomographe est actuellement en cours de test et attend d'être accepté par l'Administration Nationale des Médicaments, de l'Alimentation et de la Technologie Médicale (ANMAT) en Argentine pour commencer les tests sur des patients.

Ramiro Germán RODRÍGUEZ COLMEIRO

Doctorat : Optimisation et Sûreté des Systèmes

Année 2021

Tomographie par Emission de Positron à faible dose par échantillonnage d'images et apprentissage automatique

Cette thèse étudie le problème de dose de radiation dans les études de Tomographie par Emission de Positrons (PET). Trois aspects du PET-scan sont analysés.

La première partie de cette thèse est dédiée à la technologie PET-scan. Deux techniques sont développées pour le PET-scan à faible dose : l'AR-PET. Une première stratégie de sélection et de placement de photomultiplicateurs est proposée, augmentant la résolution énergétique. Une technique de localisation d'impacts des photons gamma dans les cristaux solides de scintillation est développé. Cette technique est basée sur des réseaux de neurones artificiels et sur une acquisition unique de champ. Nous montrons qu'une augmentation de la sensibilité du détecteur est obtenue.

Dans la deuxième partie de cette thèse, la reconstruction de l'image PET avec l'aide de maillages est étudiée. Un algorithme de reconstruction qui utilise une série de maillages 2D pour décrire la distribution 3D du radiotraceur est proposé, résultant en une diminution du nombre de points d'échantillonnage et rendant possible l'optimisation et la parallélisation des maillages.

Enfin, la génération de l'image d'atténuation au moyen de réseaux de neurones artificiels profonds est explorée. L'apprentissage du réseau de neurones se fait à travers une transformation d'images PET FDG sans correction d'atténuation pour produire une image de tomodensitométrie (CT) synthétique.

La conclusion des travaux de cette thèse pose la base pour l'usage de PET-scan à bas coût et à faible dose, via l'usage d'une image d'atténuation artificielle.

Mots clés : imagerie médicale – tomographie par émission – apprentissage automatique – calcul adaptatif – réseaux adverses génératifs.

Towards Reduced Dose Positron Emission Tomography Imaging Using Sparse Sampling and Machine Learning

This thesis explores the reduction of the patient radiation dose in screening Positron Emission Tomography (PET) studies. It analyses three aspects of PET imaging, which can reduce the patient dose: the data acquisition, the image reconstruction and the attenuation map generation.

The first part of the thesis is dedicated to the PET scanner technology. Two optimization techniques are developed for a novel low-cost and low-dose scanner, the AR-PET scanner. First a photomultiplier selection and placement strategy is created, improving the energy resolution. The second work focuses on the localization of gamma events on solid scintillation crystals. The method is based on neural networks and a single flood acquisition, resulting in an increased detector's sensitivity.

In the second part, the PET image reconstruction on mesh support is studied. A mesh-based reconstruction algorithm is proposed which uses a series of 2D meshes to describe the 3D radiotracer distribution. It is shown that with this reconstruction strategy the number of sample points can be reduced without loosing accuracy and enabling parallel mesh optimization.

Finally the attenuation map generation using deep neural networks is explored. A neural network is trained to learn the mapping from non attenuation corrected FDG PET images to a synthetic Computerized Tomography.

With these approaches, this thesis lays a base for a low-cost and low-dose PET screening system, dispensing the need of a computed tomography image in exchange of an artificial attenuation map.

Keywords: imaging systems in medicine – tomography, emission – machine learning – adaptive meshing – generative adversarial network.

Thèse réalisée en partenariat entre :

UTN.BA Universidad Tecnológica Nacional Facultad Regional Buenos Aires — CONICET — utt Université de Technologie Troyes

Ecole Doctorale "Sciences pour l'Ingénieur"