



UNIVERSIDAD NACIONAL DE COLOMBIA

Target volume delimitation with PET-CT in radiotherapy planning: A GDCM and ROOT based software implementation

Helman Alirio Amaya Espinosa

Universidad Nacional de Colombia
Facultad de Ciencias, Departamento de Física
Bogotá D.C., Colombia
2014

Target volume delimitation with PET-CT in radiotherapy planning: A GDCM and ROOT based software implementation

Helman Alirio Amaya Espinosa

Trabajo final presentado como requisito parcial para optar al título de:
Magister en Física Médica

Director:
Dr. sc. hum. Luz Stella Veloza Salcedo

Línea de investigación:
Procesamiento de Imágenes diagnósticas de Medicina Nuclear
Grupo de Investigación:
Grupo de Física Nuclear de la Universidad Nacional de Colombia (gfnun)

Universidad Nacional de Colombia
Facultad de Ciencias, Departamento de Física
Bogotá D.C., Colombia
2014

*To Lucelia, Alirio, John Fredy, Imelda,
Yasmín and Jhovany.*

Acknowledgements

To Wolfram Stiller and Gregor Pahn. Department of Diagnostic & Interventional Radiology, University Hospital Heidelberg, Heidelberg, Germany. For its contribution in the development of the software.

To personal from "Servicio de Medicina Nuclear", and the "Radiofarmacia" from the "Instituto Nacional de Cancerología". For its technical support and equipment of PET-CT, necessary for the acquisition of the Agar Phanthom, and Jaszack Phanthom 18F-FDG PET-CT images.

To Pilar Meléndez. Asociated Professor. Departamento de Farmacia. Universidad Nacional de Colombia. For its collaboration for the preparation of the Agar Phanthom.

To Nathaly Barbosa for everything what she taught me in Nuclear Medicine, and to Stella Veloza for focusing me in an investigation topic much according to my professional profile.

To my family for inspire me the love for the knowledge, and to the source of the spirituality.

Resumen

Un algoritmo computacional basado en detector de bordes de Canny, fue desarrollado para ser utilizado en procesamiento de imágenes de PET-CT y CT. Este algoritmo es un software construido con librerías de ROOT y GDCM. GDCM y ROOT son frameworks desarrollados por el CERN, y están licenciados como software libre.

El software desarrollado mostró una mejor delimitación de una región de hiper-captación simulada con un Phantom de Agar, que el método de thresholding, siendo aplicados ambos en imágenes de 18F-FDG PET-CT. El software permite analizar algunas características anatómicas y metabólicas en los tumores. Este software también puede procesar imágenes de CT, en donde se obtuvo una mejora cualitativa en el contraste entre las regiones anatómicas, con lo que se puede establecer una mejor distinción entre ellas.

La detección de bordes se realizó en las imágenes diagnósticas de CT y 18F-FDG PET-CT, aplicando el algoritmo de Canny por medio de una librería de OpenCV y es posible observar una completa detección de bordes, tanto en regiones anatómicas (CT) como metabólicas (PET-CT), en imágenes de un tumor abdominal.

Palabras claves: Detección de bordes en tumores, planeación de radioterapias con PET-CT, Imágenes diagnósticas, software libre, PET-CT, CT, Imagenología de los tumores.

Abstract

A computational algorithm based on the Canny's edge detector was developed to be used in PET-CT and CT image processing. This algorithm is a software constructed with ROOT and GDCM libraries. GDCM and ROOT are frameworks developed by the CERN, which are licensed as free software.

The developed software showed a better delimitation of a hyper-uptake region simulated with an Agar Phantom, than the thresholding method, and the both methods were applied in 18F-FDG PET-CT images. The software also allows analyze some additional features in the tumors. This software also can process CT images, and a qualitative improvement in the contrast between the anatomical regions was obtained, establishing a better distinction between them.

The edge detection was realized in the CT and 18F-FDG PET-CT diagnostic images using the Canny algorithm present as an OpenCV library, and it is possible to observe a complete edge detection of both, anatomical regions (CT Image), and the tumor (PET-CT images).

Keywords: Tumor edge detection, PET-CT-guided radiation therapy planning, medical imaging, free software, PET-CT Imaging, CT Imaging, tumor imaging.

Table of Contents

Resumen and Abstract	ix
Index of figures	xv
List of Acronyms and Symbols	xvii
1 Introduction	1
2 Images in diagnostic radiology and Nuclear Medicine.	3
2.1 Photographic images	3
2.2 Radiological Images	4
2.2.1 Radiographs.	5
2.2.2 Computed Tomography (CT).	5
2.3 Nuclear Medicine Images.	8
2.3.1 Scintigraphy	8
2.3.2 Single Photon Emission Computed Tomography (SPECT)	12
2.3.3 Positron Emission Tomography (PET).	13
3 The role of CT and PET-CT in Radiotherapy.	17
3.1 Delimitation of tumor volumes by means of computed tomography (CT).	17
3.2 Problems with the CT in Radiotherapy Planning, and the Role of PET-CT.	19

4	Image processing with ROOT and OpenCV.	23
4.1	Image Color extraction and Convolution.	27
4.2	Image Edge detection and enhancement.	33
4.2.1	Gradient Magnitude and Laplace edge enhancement with ROOT.	33
4.2.2	Canny edge detection with OpenCV	38
5	DICOM Image manipulation with GDCM.	41
6	Metodology.	45
6.1	Software Development.	45
6.2	Experimental Measurements.	47
6.3	Agar Phantom Dessign.	48
7	Results and Analysis.	53
7.1	Phantom Images.	53
7.2	Diagnostic Images.	64
8	Conclusions and perspectives	75
	References	79
9	Apendix 1. Filtered back projection algorithm, Mathematical Fundamentals.	81
9.1	Radon Transform [3].	81
9.2	Inverse Radon Transform [43].	83
9.3	Filtered back projection Method.	85
10	Appendix 2. Calculation of the inner (R_i) and outer (R_e) radius.	87

Index of figures

2.1	Chest anteroposterior radiography	5
2.2	CT operation mode	6
2.3	Tomographic transversal slices for head	7
2.4	Collimators and its location in each detector	9
2.5	A SPECT scanner and a Scintigraphy	9
2.6	Positron-electron annihilation	13
2.7	Coincidence associated to an annihilation	13
2.8	Types of coincidence events	14
2.9	Process for obtaining the tomographic imaging using PET	15
2.10	PET and PET-CT Fused Images	16
3.1	Software ECLIPSE TM for planning in Radiation Therapy of head and neck tumors	17
3.2	Target volumes defined by the ICRU	18
3.3	Scheme of the mechanism of localization for ¹⁸ F-FDG, ¹⁸ F-MISO and ¹⁸ F-AZA	20
3.4	CT Image in a patient with a tumor and distal atelectasis in the right lung	21
3.5	Additional tumor found in the planning of radiation therapy with PET-CT	21
4.1	Ailurus fulgens image and its ROOT histogram	24
4.2	3D visualization of a TH2D histogram.	25
4.3	Histograms of a test image for each color, and a grayscale	29
4.4	Original Image in a grayscale, and images with different convolutions.	32

4.5	Ailurus grayscale image and its edge enhancing.	36
4.6	Wheel image and its edge enhancing	37
4.7	Edge detection using the OpenCV Canny function	39
5.1	Test PET-CT DICOM file visualization in ImageJ with inverted LUT	43
6.1	PET-CT Image exported to a ROOT canvas and to a TH2D Histogram	46
6.2	Experimental design for the acquisition of PET-CT images from the phantom Jaszczak	47
6.3	DICOM image of a Phantom Jaszczak/ACRVOI	48
6.4	Results of the Kirby Bauer Test	49
6.5	Process of elaboration of a prototype agar phantom.	50
6.6	Agar phantom, and its positioning in the INC's PET-CT equipment.	51
7.1	Jaszczak Phantom PET-CT histogram, and PET-CT gradient magnitude histogram.	53
7.2	Agar Phantom PET-CT Image, and PET-CT gradient magnitude histogram.	54
7.3	Calculated diameter as a function of the the threshold percentage (Thr)	55
7.4	Percentage difference P_d as a function of Thr	55
7.5	Agar Phantom Thresholding images for some Thr	56
7.6	Calculated diameters for the remaining ring as a function of Thr	57
7.7	Percentage difference P_d for the mean diameter as a function of Thr	57
7.8	Agar Phantom gradient Thresholding.	58
7.9	Jaszczak Phantom PET-CT Image histogram, and its Canny Edge Detection.	59
7.10	Agar Phantom PET-CT Image, and its Canny edge detection.	60
7.11	Original Agar Phantom PET-CT Image, and its Laplacian	61
7.12	Calculated diameter (with Laplacians) as a function of the Thr	62
7.13	Percentage difference P_d for the internal diameter as a function of Thr	63
7.14	Agar Phantom Laplacian Thresholding.	63
7.15	DICOM PET-CT Image exported to ROOT canvas, and TH2D Histogram	64

7.16	DICOM PET-CT Image and exported to a ROOT grayscale TH2D histogram	65
7.17	DICOM CT Image and exported to a ROOT grayscale TH2D histogram	65
7.18	PET-CT Control image and Gaussian Blur convolution.	66
7.19	PET-CT Sharpening convolution and edge enhancing convolution	66
7.20	CT Image Histogram and Gaussian Blur convolution	67
7.21	CT Sharpening convolution and edge enhancing convolution.	67
7.22	PET-CT Image histogram, and PET-CT gradient magnitude histogram	68
7.23	PET-CT gradient magnitude (filtered), and PET-CT laplacian (filtered) histograms.	68
7.24	CT Image histogram, and CT gradient magnitude histogram	69
7.25	CT gradient magnitude (filtered) histogram, and CT laplacian (filtered) histogram.	69
7.26	CT Test Image, and Canny Edge detection with different minimum threshold.	71
7.27	PET-CT Test Image, and Canny Edge detection with different minimum threshold	72
7.28	PET-CT Test Image SUV_{max} Thresholding using different Thr	73
9.1	Domain D for the function $f(x,y)$ and its path of integration in a Radon transform	81
9.2	Projections $p_{\theta}(t)$ for different t	83
9.3	Simple back projection, and the distortion associated with the reconstruction process	85
9.4	Steps of the filtered-projection algorithm.	86

List of Acronyms and Symbols

Symbols with Latin scripts

Symbol	Meaning	Unit	Definition
$n(i, j, t)$	Number of counts per voxel (pixel) i, j and time t		Ec. 2.2
$p_{\theta}(t')$	Proyection per angle θ and position t'		Ec. 9.5
$c(i, j, i)$	Mass concentration per voxel (pixel) i, j	$\text{g}\cdot\text{cm}^{-3}$	Ec. 7.3
Thr	Threshold Percent		Graph. 7.3

Symbols with Greek scripts

Symbol	Meaning	Unit	Definition
∇	Gradient vector	m^{-1}	Ec. 4.3
∇^2	Laplacian operator	m^{-2}	Ec. 7.6
σ	Variance	cm	Tab. 4.1
$\mu(i, j)$	Attenuation Coeficients per voxel (pixel) i, j	cm^{-1}	Ec. 9.5

Acronyms

Acronym	Meaning
ARGB	Alpha Red Green Blue color images
BTV	Biological Target Volume
CERN	European Organization for Nuclear Research
CT	Computed Tomography
DICOM	Digital Imaging and Communications in Medicine
GDCM	Grassroots DICOM
GLUT	Glucose Transporters
GTV	Gross Tumor Volume
TH2D	ROOT 2D Histogram
INC	Instituto Nacional de Cancerología.
OpenCV	Open Source Computer Vision
PET	Positron Emission Tomography
PET-CT	Fused PET and CT imaging
PTV	Planning Target Volume
ROOT	Object-oriented data analysis framework
SPECT	Single Photon Computed Tomography
SUV	Standard Uptake Values

Chapter 1

Introduction

Positron Emission Tomography (PET) images provide information about the registered counts per voxel, i.e., the number of real coincidences associated with two antiparallel photons produced at each electron-positron annihilation occurring in the patient's body after injection of a positron emitter radiopharmaceutical. Since the photons could be absorbed into the patient before reaching the detectors, computed tomography (CT) images provide the necessary information to quantify the degree of attenuation, therefore a fused positron emission and computed tomography (PET-CT) image is directly correlated with the concentration of the radiopharmaceutical which is an indication of the level of metabolism of the radiopharmaceutical at each anatomical region.

Based on the properties of PET-CT images mentioned above, a big variety of tumors can be located using 18F-Fluorodeoxyglucose (18F-FDG). In a conventional way, source-background algorithms are used for delineation of biological tumor volume (BTV) boundaries based on FDG PET-CT images. The algorithms assign a vector of counts per voxel along the length and width of the entire image, after that determine the maximum value of uptake (SUVmax) by the organization from largest to smallest values of the vector and then using a pre-defined threshold value -usually in the order of 30% of the SUVmax- only the voxels that are above the threshold value being part of the tumor.

If this approach is changed and each image from PET-CT is now visualized as a discrete map of the number of counts, which is a continuous function of two spatial coordinates, then directional derivatives and gradients of the function can be computed on the basis of multivariate calculation tools, and the determination of the regions of maximum uptake becomes a typical optimization problem. This principle, which is the basis of the Canny's edge detection tool, is used in this research for processing fused PET-CT images in order to delimit the edges of the hyper-uptake regions of radiopharmaceuticals. The computational algorithm was implemented using the data analysis framework ROOT (CERN, Geneva, Switzerland) and the Grassroots DICOM GDCM libraries. In addition to the edge enhancing and edge detection, other algorithms of image processing, which can be a blur or refocusing of images, among other processing methods can be implemented.

In this work, first, a software that processes DICOM images of PET-CT and CT was developed, and then both diagnostic imaging, such as images of phantoms were processed using this software. The following chapters show the different phases of the processing image software development, and the results obtained when processing diagnostic images, and phantoms that were useful to evaluate the behavior of this developed software.

Chapter 2

Images in diagnostic radiology and Nuclear Medicine.

There are many diagnosis setups for getting images related with anatomy and metabolism from human body's tissues. Setups based on magnetic nuclear resonance and X ray tubes give anatomical information and there are the basis for researching in radiology.

On the other hand, Nuclear Medicine allow the study of metabolic function of tissues, using radioactive substances. Nuclear medicine systems detects gamma radiation from a radiopharmaceutical. The systems described before, produce images in a grayscale intensity, or in color, which allows to make 2D o 3D representation of the anatomy and metabolism of the patient.

In spite of that images are based on intensity acquisition, the information produced for each diagnostic system is different. Next, images from computed tomography (CT), Positron Emission Tomography (PET), and combined images from them (PET-CT) will be described.

2.1 Photographic images

Before treating diagnostic images, we must talk about the way humans perceive the world. Visible light from sun or artificial sources interacts with objects in its way. In that process, some photons are absorbed by the object and others are reflected or transmitted at human eyes, and they are tried at the retina. Retina has three kind of cones which detect photons from several visible wavelengths (red, green and blue), this is the form in which brain construct images colors.

There are other kind of photoreceptors that works in low intensity conditions, called rod cells that detect light from a wavelength. Because of that, human eye cannot distinguish colors in the last condition. Photograph taken by a digital camera, corresponds to an intensity image in which the image is discretized and information about color are stored in little squares called pixels.

There are many ways for storing color components, being the most common the RGB vector, which the first component saves information related with the red color, the second component saves the green one and the third component saves the blue one. Other kind of way for storing information about colors is CMYK (Cyan, Magenta, Yellow and Key). There is also, storing through fourth components vector ARGB, when first component saves information related to opacity in each pixel and the other forms the RGB vector who has been described before. Note that each vector component is called a channel, then an image in format ARGB is an image of four channels.

In an ARGB image, the information stored in each pixel is expressed as hexadecimal number like 0xAARRGGBB (where 0x indicate that the number is expressed in hexadecimal number). Then, the information is an array of four hexadecimal numbers which range from zero to FF that is, zero to two hundred and fifty five. Every hexadecimal number represents a binary number of 8 bits (or 1 byte) which has two possible values 0 or 1, hence there are $2^8=256$ possible combinations [1].

For instance, the decimal number 215 in hexadecimal is expressed like D7, and in binary is 11010111, which is a number of a byte. Therefore, color deeper in an ARGB image is 8 bits per channel, for a whole of 32 bits per pixel (bpp). For images in gray scale, each pixel saves intensity values, between 0 to 256 which corresponds to hexadecimal numbers of 8 bits that is, this kind of images has a whole of 8 bpp.

Additionally, it is possible to convert from images from ARGB to an image in gray scale, and for that there are many ways, for example computing the intensity color, this is the norm of every component of a vector or finding the average [2]:

$$lightness = \frac{(\max(R, G, B) + \min(R, G, B))}{2}$$

$$luminosity = 0.21R + 0.72G + 0.07B$$

Most diagnostic images, are represented in gray scale because of in that way it is possible to represent average value of some physical property which it is measured in a volume element, called voxel, which is analogous to a pixel. Next, some properties from diagnosis image are showed. Then, fundamental and applications of diagnosis of tumors by images systems like computed tomography (CT), Positron emission tomography (PET) and combines images from PET-CT, are showed.

2.2 Radiological Images

Radiological techniques are based on the use of x-rays, to get images in two dimensions (radiographs) or three-dimensional images (CT), among other image modalities. Next, some basic principles of both modalities will be described below.

2.2.1 Radiographs.

X-ray images, are basically obtained by comparison between an initial intensity of x-rays and a final intensity, according to:

$$I = I_0 e^{-\int_L \mu(x,y) ds}$$

In a radiography, each voxel represents the ratio of intensities I/I_0 , so that, each voxel is a average attenuation. In the previous equation the integral can be understood in terms of the average attenuation coefficient along the line L , as $\bar{\mu}L = \int_L \mu(x,y) ds$, where L is the thickness of the patient in each boxel, this is:

$$I = I_0 e^{-\bar{\mu}L}$$

Then, a radiography can be understood as an average attenuation coefficients map for each boxel [3]. In the radiographs, the intensity is major as the x-rays attenuation increases, so the radiographs are images in a grayscale (figure 2.1).

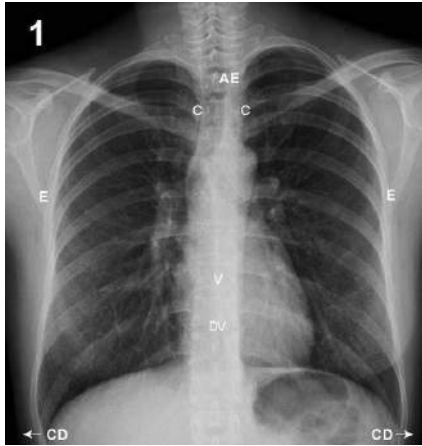


Figure 2.1: Chest anteroposterior radiography. Taken from wikipedia

2.2.2 Computed Tomography (CT).

While the X-ray tube is rotating, a projection $p_\theta(t)$ is obtained, for every angle θ for the tube, such that for every complete rotation, on having weighted the results for each projection an image can be obtained (in the xy plane).

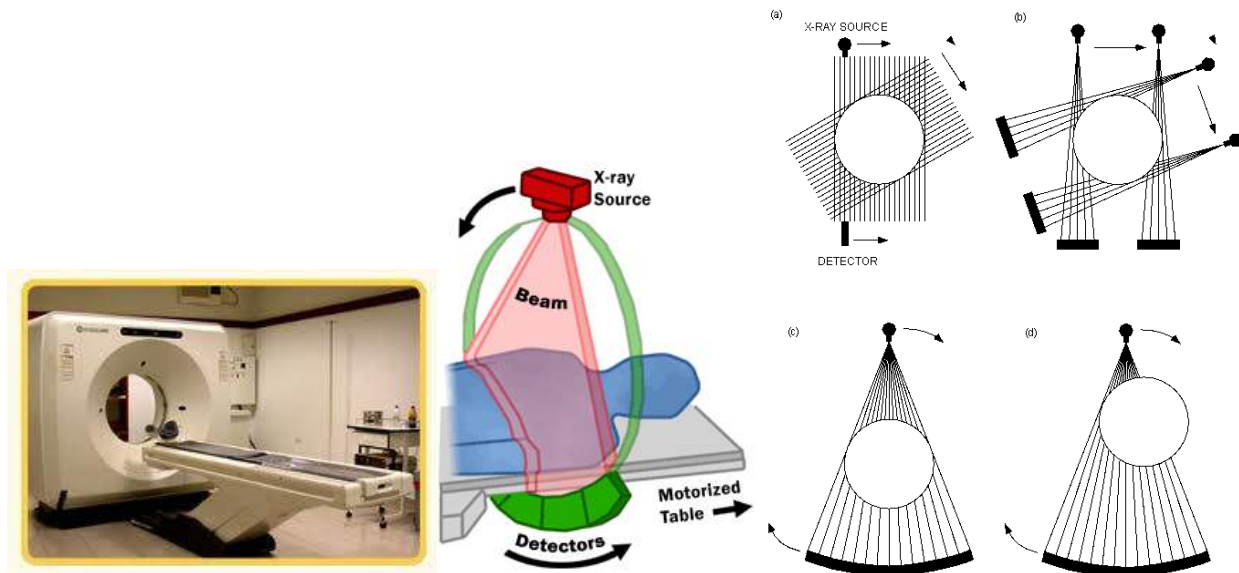


Figure 2.2: CT operation mode. Taken from Wikipedia

The stretcher where the patient is located goes forward, and then images for each position of the stretcher are obtained, this corresponds to a slice in each position of the z axis. Each of these images are known as a tomographic slice, which in this case is an anatomical transversal slice. Not only images in the transversal slice can be obtained for CT, because the projections can be reconstructed to obtain anatomical slices different from the transversal (figure 2.3).

CT Tomographic slices correspond to a 2D map in Hounsfield (HU) units, defined by:

$$HU(x,y) = 1000 \cdot \frac{\mu(x,y) - \mu_{Water}}{\mu_{Water}} \quad (2.1)$$

That is to say that in each tomographic slice obtained, as seen in figure 2.3, represents a distribution (throughout the length and width of the slice) of attenuation coefficients $\mu(x,y)$ corrected with respect to the water attenuation coefficient value μ_{Water} . In that way, the image for each slice can be seen as a function $HU(x,y)$ where x,y are the spatial coordinates. Each of the projections $p_{\theta}(t)$ can be physically understood as:

$$p_{\theta}(t) = -\ln\left(\frac{I}{I_0}\right) = \ln 10 \cdot \left[-\log\left(\frac{I}{I_0}\right)\right] = \ln 10 \cdot A(\theta)$$

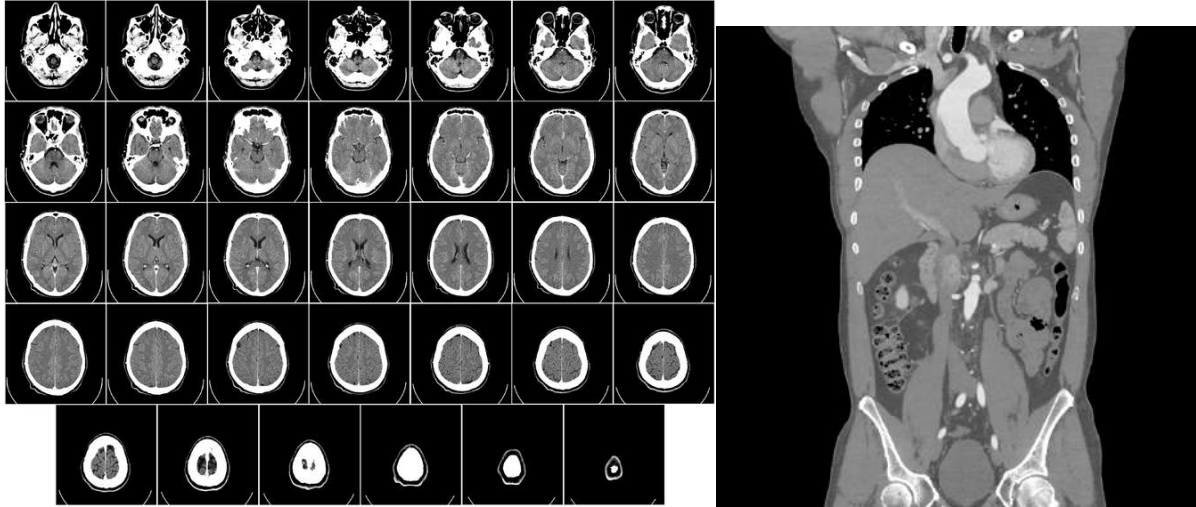


Figure 2.3: Tomographic transversal slices for the head starting at the base of the cranium and finishing in the upper part of the cranium (left). Tomographic coronal slice (right). Taken from Wikipedia

Where $A(\theta)$ is the optical density for each angle θ where the projection is evaluated. According to Lambert-Beer law the logarithm value can be easily found, so each projection can be calculated by:

$$p_{\theta}(t) = -\ln \left[\frac{I}{I_0} \right] = \int_L \mu(x, y) ds$$

The right integral is a path integral along the projection, known as the Radon transform and its trajectory is described by the parameter t . Performing the Fourier transform $P_{\theta}(\omega)$ for each projection, one can obtain:

$$P_{\theta}(\omega) = \int_{\mathbb{R}} p_{\theta}(t) e^{-i\omega t} dt$$

$P_{\theta}(\omega)$ can be seen as a Fourier transform in two variables, and using its associated inverse transform, $\mu(x, y)$ can be calculated:

$$\mu(x, y) = \int_0^{\pi} \left[\int_{-\infty}^{\infty} P_{\theta}(\omega) e^{i\omega t} \omega d\omega \right] d\theta$$

This is known as the inverse Radon transform. These previous steps constitute the filtered backprojection algorithm, from which the distribution of attenuation coefficients $\mu(x, y)$ can be obtained in the studied anatomical slice, which will get compared with that of the water for a better visualization, resulting in the image $HU(x, y)$. In Appendix I, mathematical foundations for the Radon transform and the backprojection algorithm are described.

2.3 Nuclear Medicine Images.

The drugs are often used for the treatment of diseases, and therefore is well known its pharmacological action (mechanism of action), biodistribution, adverse effects, and the time it takes to be eliminated (biological half-life).

If a drug is joined to a radioactive atom, its possible to trace, both the biodistribution of the radiopharmaceutical, such as the degree of localization (uptake) that this one has in a tissue. In some tissues, a great uptake of radiopharmaceutical is an indication of the presence of disease, and for that reason, the radiopharmaceuticals are useful in the diagnosis of diseases.

The dose in the administered radiopharmaceuticals are in the order of nanograms, and a pharmacological action is not obtained, however, this dose is sufficient to obtain diagnostic images. Radiopharmaceuticals are also administered in higher doses for therapeutic purposes, but the therapeutic effect is obtained by means of the energy released by the radionuclide, not because of the pharmacological effect associated with the drug.

If in certain anatomical region a high uptake of radiopharmaceutical was obtained, from this region will be emitted a relatively great amount of photons, which are detected and with tomography equipment that converts this information into a diagnostic image. In the next section will be described the physical principles in which Scintigraphy, SPECT and PET equipment (the most frequently used in Nuclear Medicine) are based.

2.3.1 Scintigraphy .

When a radiopharmaceutical that emits a single photon (gamma radiation) is injected, the photons that are emitted from each anatomical region of the patient (counts) can be registered , by means of two detectors that are located ahead and behind the patient (figure 2.5 left).

In a radionuclide decay, photons are emitted in all directions. To know from which anatomical region the photons were emitted, they pass through a collimator as shown in the figure 2.4. The colimador is a plate of lead that has thousands of holes, and only those photons that pass for the holes (channels), are going to reach the detector. The remaining photons will be absorbed by the septa, which are between the holes [4].

The number of photons that pass through the collimator will depend on both, the diameter of the channels (or the number of channels), and the thickness of the collimator. When the diameter of the channels is large, a major number of photons passes, but it is more difficult to determine the region where from the photons were emitted, i.e. sensitivity can be gained, but spatial resolution will get lost. If the thickness of the collimator is small in size, photons with intermediate and high energies will be able to cross every septum and will be detected, so the resolution of the image will not be good, and it is necessary to increase the thickness of the collimator [4].

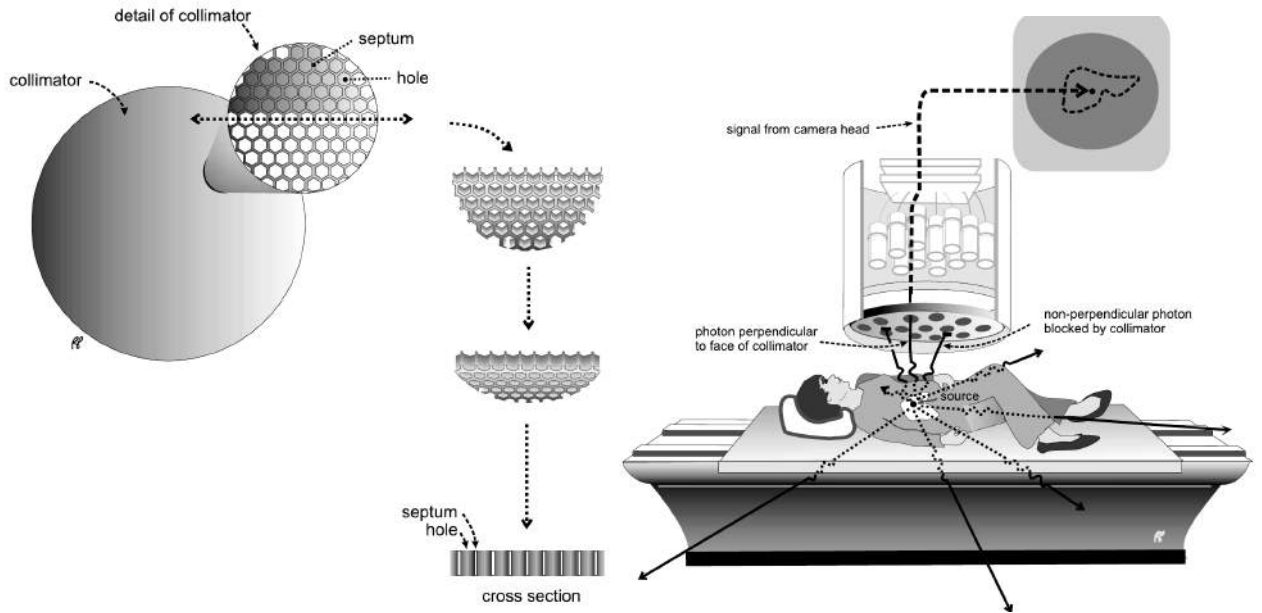


Figure 2.4: Collimators and its location in each detector. Images taken from [4]

The two detectors are in a header that is called gantry. The distance between the detectors can be modified and the gantry can also rotate. For the position of the patient between two detectors, there is a stretcher that moves horizontally and vertically. The gantry with detectors, and the stretcher are part of the SPECT Scanner (figure 2.5 left). The image obtained when the gantry is in a fixed position is called gammagraphy or scintigraphy, which is classified in anteroposterior or posteroanterior, according to the anatomic position in which is acquired (figure 2.5 right).



Figure 2.5: A SPECT scanner (left), taken from [5]. Normal posteroanterior and anteroposterior Scintigraphy (right), taken from [6].

The gammagraphy images have a low resolution, in comparison with a typical X-ray image. This is due to three reasons. First, the number of photons emitted in each tissue is relatively small. Second, part of the photons are scattered by Compton effect, but these are absorbed in the septa that exist in the detectors arrangement. Third, by means of photoelectric effect, some photons are absorbed (attenuated) by the patient.

Each gammagraphy as shown in the figure 2.5, represents a mapping of the number of counts $n(i, j, k)$ in the spatial position (i, j, k) , with a constant acquisition time for a patient. If the gantry is in a fixed position characterized by the coordinate k , the number of counts $n(i, j, k)$ will only depend on the position (i, j) inside the plane of the image (or the detector), so the number of counts can be expressed as $n(i, j)$. If the acquisition time is variable (as in the case of dynamic images), then $n(i, j)$ is also time-dependent, therefore from now, it will be expressed as $n(i, j, t)$.

This number of counts $n(i, j, t)$ in a specific anatomical region depends on the radionuclid concentration, and the acquisition time to get the image. In a first approximation, $n(i, j, t)$ is a function h of the radiopharmaceutical mass concentration per voxel volume $c(i, j, t)$ ¹ in some anatomic region. That is:

$$n(i, j, t) = h[c(i, j, t)]$$

After some considerations, the functional form for h can be determined, having into account that n is the event number registered by a radiation detector. The total number N_0 of emitted particles by the radiopharmaceutical (photon or positrons), in a time Δt is given by [7]:

$$N_0 = A(t)b_p\Delta t$$

Where $A(t)$ is the activity of the radionuclide at time t and b_p is the decay percentage (branch) of interest. When the emission of positrons is considered, emitted positrons have different energies that are distributed in a continuous, however, when the positron loses almost all of its kinetic energy, and form the bound state linked with the electron, only a fraction f_2 of the total number of annihilations produces two photons, so the number of annihilations where two 511 keV photons produced is:

$$N_a = A(t)b_p f_2 \Delta t$$

From now, the last equation will be used, taking into account that when a single gamma is produced $f_2 = 1$. The activity $A(t)$ can be calculated as:

$$A(t) = \lambda N(t) = \frac{\ln 2}{T_{1/2}} N(t)$$

¹The mass concentration $c(i, j, t)$, in general is the radiopharmaceutical mass concentration for the voxel (i, j, k) , whose volume is ΔV . In this case, ΔV can be expressed as $\Delta V = \Delta A \cdot \Delta z$, where ΔA is the area per voxel inside the image and Δz is the voxel thickness, which depends on both, the anatomical region that is being studied, and the gantry position.

λ and $T_{1/2}$ are the disintegration constant, and the time of half-life of the radionuclide, respectively. The $N(t)$ number of radioactive nuclei without decay, can be calculated from the mass of radiopharmaceutical $M(t)$ without decay:

$$N(t) = \frac{N_A M(t)}{M_{RF}}$$

N_A is the Avogadro's number and M_{RF} the molecular weight (in grams) of the radiopharmaceutical. The activity can now be calculated as:

$$A(t) = \frac{\ln 2}{T_{1/2}} N(t) = \frac{\ln 2 \cdot N_A}{T_{1/2} M_{RF}} M(t)$$

Multiplying and dividing by the volume ΔV of each voxel:

$$A(t) = \frac{\ln 2}{T_{1/2}} N(t) = \frac{\ln 2 \cdot N_A}{T_{1/2} M_{RF}} \frac{M(t)}{\Delta V} \Delta V = \frac{\ln 2 \cdot N_A}{T_{1/2} M_{RF}} c(i, j, t) \Delta V$$

The activity is now expressed in terms of the mass concentration of radionuclide in each voxel $c(i, j, t)$. This concentration depends on both of the biodistribution of the radiopharmaceutical, as also of the radioactive decay that present the radionuclide. N_a now takes the form:

$$N_a = A(t) b_p f_2 \Delta t = \frac{\ln 2 \cdot N_A}{T_{1/2} M_{RF}} b_p f_2 \Delta V \Delta t \cdot c(i, j, t)$$

The number $n(i, j, t)$ of counts (or coincidences) that are registered for a voxel (i, j) at the time t , really is not equal to the number N_a , and based on the ratio between the both numbers it is possible to calculate the total efficiency $\varepsilon(i, j)$ for each voxel:

$$\varepsilon(i, j) = \frac{n(i, j, t)}{N_a}$$

In the quality control of the gamma cameras, the uniformity of the obtained images is guaranteed periodically, therefore, the efficiency for each voxel, is practically a constant. Using the last approach, the efficiency $\varepsilon(i, j)$ does not change significantly between the voxels, so it can be expressed in terms of average efficiency for each voxel (ε), i.e., $\varepsilon(i, j) \approx \varepsilon$. $n(i, j, t)$ now can be written as:

$$n(i, j, t) = \varepsilon(i, j) N_a = \varepsilon(i, j) \frac{\ln 2 \cdot N_A}{T_{1/2} M_{RF}} b_p f_2 \Delta V \Delta t \cdot c(i, j, t) = \frac{\ln 2 \cdot N_A}{T_{1/2} M_{RF}} \varepsilon b_p f_2 \Delta V \Delta t \cdot c(i, j, t)$$

For each static gammagraphy study, the acquisition time ($\Delta t = t - t_o$) and the volume of each voxel (ΔV) are constants, also the values b_p and f_2 are constants that depend on the used radionuclide. For each dynamic gammagraphy, the term $a(t) = \frac{\ln 2 \cdot N_A}{T_{1/2} M_{RF}} \varepsilon b_p f_2 \Delta V (t - t_o)$ which accompanies the mass

concentration $c(i, j, t)$ can be considered as a variable that changes linearly in the time, so $n(i, j, t)$ varies with $c(i, j, t)$, in accordance with the linear function 2.2:

$$n(i, j, t) = a(t) \cdot c(i, j, t) \quad (2.2)$$

The latter equation, is also valid for the SPECT and PET-CT dynamic images, which are 3D images, therefore it is necessary to include component k:

$$n(i, j, k, t) = a(t) \cdot c(i, j, k, t) \quad (2.3)$$

2.3.2 Single Photon Emission Computed Tomography (SPECT)

If each scintigraphy is now seen like a projection, making a projection with a constant acquisition time, for different angles of rotation of the gantry, and then performing the reconstruction of images, it is possible to get a three-dimensional representation of the number of counts for each anatomic region. This technique is known as Single Photon Emission Computed Tomography (SPECT).

In SPECT projections are defined as in CT. Each projection is defined as:

$$p_{\theta}(t) = \int_L n'(x, y) ds \quad (2.4)$$

Where $n'(x, y)$ is a function that describes the number of counts for the anatomical regions in each slice. When the inverse radon transform is performed, it is obtained $n'(x, y)$, but also it is necessary to make the attenuation correction.

The probability of a photon (from the decay of a radionuclide in the patient's body) arrives at a detector is:

$$P = e^{-\int_0^{l(x,y)} \mu(x', y') dl}$$

$\mu(x', y')$ is the attenuation coefficient in the position (x', y') . $l(x, y)$ is the distance between the point of emission (x, y) and the detector along the line L. Moreover, as all probability is defined as the number of favorable cases divided into the number of possible cases, $P = n'(x, y)/n(x, y)$, i.e:

$$n'(x, y) = n(x, y) \cdot e^{-\int_0^{l(x,y)} \mu(x', y') dl}$$

By replacing in the equation 2.4:

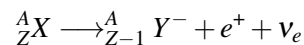
$$p_{\theta}(t) = \int_L n(x, y) \cdot e^{-\int_0^{l(x,y)} \mu(x', y') dl} ds$$

That is the attenuated Radon transform [8]. The line integral goes from the point of emission of the photon, up to the position of the detector. The attenuation coefficients $\mu(x', y')$, are obtained from the CT for each patient's tissue, so the attenuated Radon transform is a basic algorithm for obtaining SPECT-CT image fusion. In the space between the patient and the detector, the attenuation coefficients are practically null, taking into account that it is air. On the other hand, the attenuation coefficients determined by CT for each tissue, must be corrected, because in general, the energy of the photons emitted by the radionuclide is different to the energy of the photons emerging from the X-ray tube [8].

To obtain $n(i, j)$ that is the corrected number of counts for each voxel, there is no analytical method, however, this is possible by means of several approximate methods, such as those presented by Zaidi and Hasegawa [8]. By means of SPECT static images can be obtained, nevertheless, when dynamic images are obtained, $n(i, j)$ must be time-dependent, such as $n(i, j, t)$.

2.3.3 Positron Emission Tomography (PET).

Positron emission tomography (PET), is a technique that is based on the administration of a radio-pharmaceutical that has a proton-rich atom in its structure, and this atom may decay via positron emission:



Once radiopharmaceutical is distributed in the central compartment (blood), it accumulates in the tissue that it is desired to study, and there, it emits positrons [9].

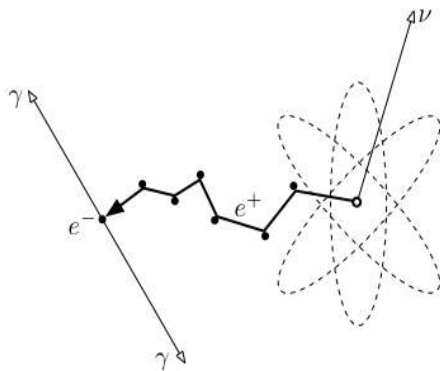


Figure 2.6: Positron-electron annihilation.
Taken from Wikipedia

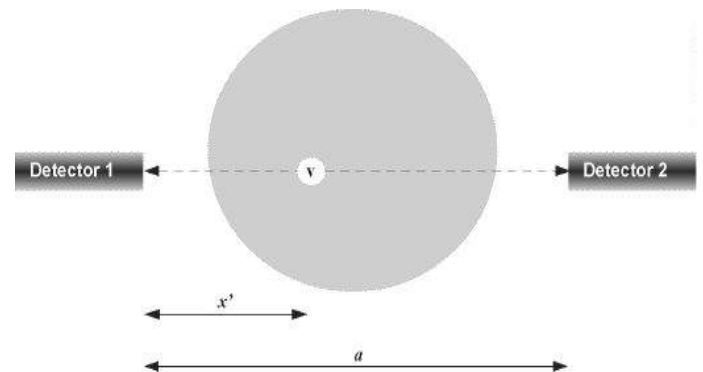


Figure 2.7: Coincidence associated to an annihilation in the point v. Taken from [10]

In the tissue, positron presents Coulomb interactions with the atom electrons (for this reason lose kinetic energy) and prior to annihilate with the electron, it forms a bound state called positronium, that is a hydrogen-like atom. When the electron meets the positron, they annihilate and decay into 2, 3, 4 or 5 photons. Principally two photons of 511 keV, in a practically antiparallel direction [9], as it appears in figure 2.6. Both photons form a LOR (Line of Response), but this is not really a line, because the angle between the two photons is not 180° , really, is minor, because the positron has a little kinetic energy before interaction [9].

Both photons, emitted in a particular anatomical region, are detected in an equipment that has a detectors arrangement to achieve a 3D acquisition. Each LOR gives spatial information that is usually called electronic colimation. The incident photons generate a timed pulse when it interacts with the detector, the pulses are combined. If the combined pulses fall within a short time (time window), they are considered coincident, otherwise the coincidence is a random. An energy window is used to remove the scatter coincidences, however, some of them can pass the energy window, for that reason, periodical calibrations with phantoms are done in the PET-CT equipment [10].

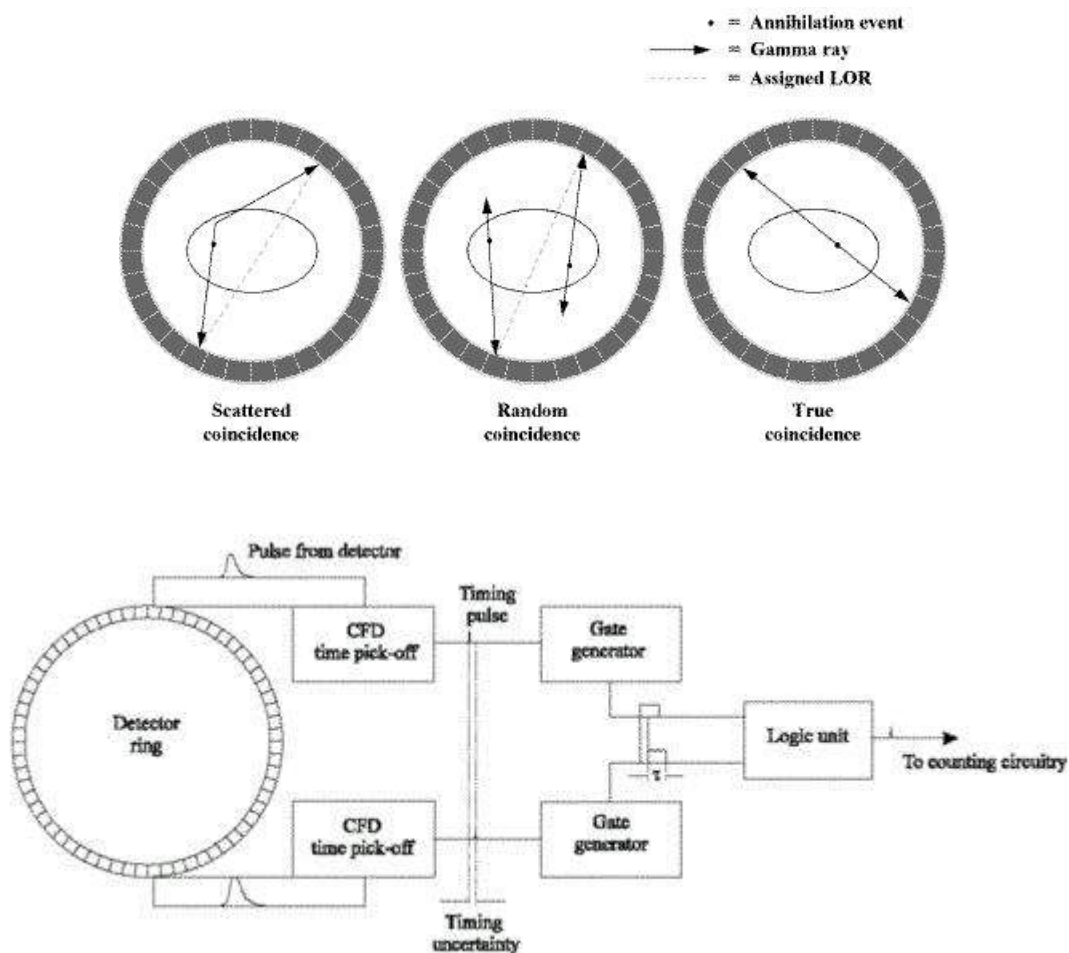


Figure 2.8: Types of coincidence events. Taken from [10]

Each (true) line of response can be correlated with an anatomical region, and by means of image reconstruction algorithms such as the filtered backprojection or the iterative reconstruction, that are implemented in computer equipment (fig 2.9), the number of coincidences (from now, counts) can be calculated for every anatomical region to make a 3D representation that shows the different degree of radiopharmaceutical uptake in each tissue [4].

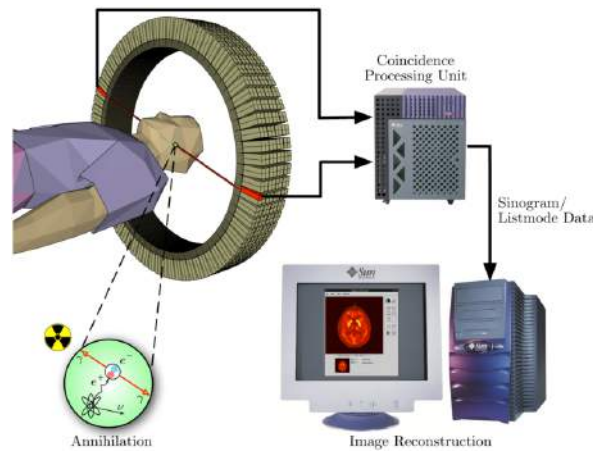


Figure 2.9: Process for obtaining the tomographic imaging using PET. Taken from Wikipedia.

For the obtaining of PET images, a Radon transform can be applied, as is done in SPECT, but in this case, each LOR is taken as a projection. Each "projection" is calculated in a similar way to how it is done in SPECT:

$$p_{\theta}(t) = \int_L n'(x,y) ds$$

As in SPECT, attenuation correction must be done, so in PET the filtered back projection algorithm can be modified under certain considerations. If an annihilation takes place in the point v (2.7), the probability of a photon reaching the detector 1 is:

$$P_1 = e^{-\int_0^{x'} \mu(x,y) ds}$$

And the probability of a photon reaching the detector 2:

$$P_2 = e^{-\int_{x'}^a \mu(x,y) ds}$$

The probability P_C of detecting a coincidence is [10]:

$$P_C = P_1 \cdot P_2 = e^{-\int_0^x \mu(x,y)ds} \cdot e^{-\int_x^a \mu(x,y)ds} = e^{-\int_0^a \mu(x,y)ds} = e^{-\int_L \mu(x,y)ds}$$

As in SPECT, $P_C = n'(x,y)/n(x,y)$ and the number of coincidences n' is:

$$n'(x,y) = n(x,y) \cdot e^{-\int_L \mu(x,y)ds}$$

Using attenuation coefficients obtained from the CT, at the energy of the X rays, and corrected to 511 keV, each "projection" in PET-CT, has the form [8]:

$$p_\theta(t) = \int_L n(x,y) \cdot e^{-\int_L \mu(x',y')dl} ds \quad (2.5)$$

Where the integration line L is the same LOR. By means of computational methods, the inverse of the equation (2.5) can be determined, to get the counts $n(i,j)$ for each voxel (i,j) . In the figure (fig 2.10), 3D images for PET, CT and the PET-CT fusion are shown, from a study realized to the same patient. CT, or PET images, represent the value of $n(i,j)$ for every voxel (i,j) in a grayscale, while the PET-CT fusion is represented in images in two planes of color yellow and red, each one representing PET and CT, respectively.

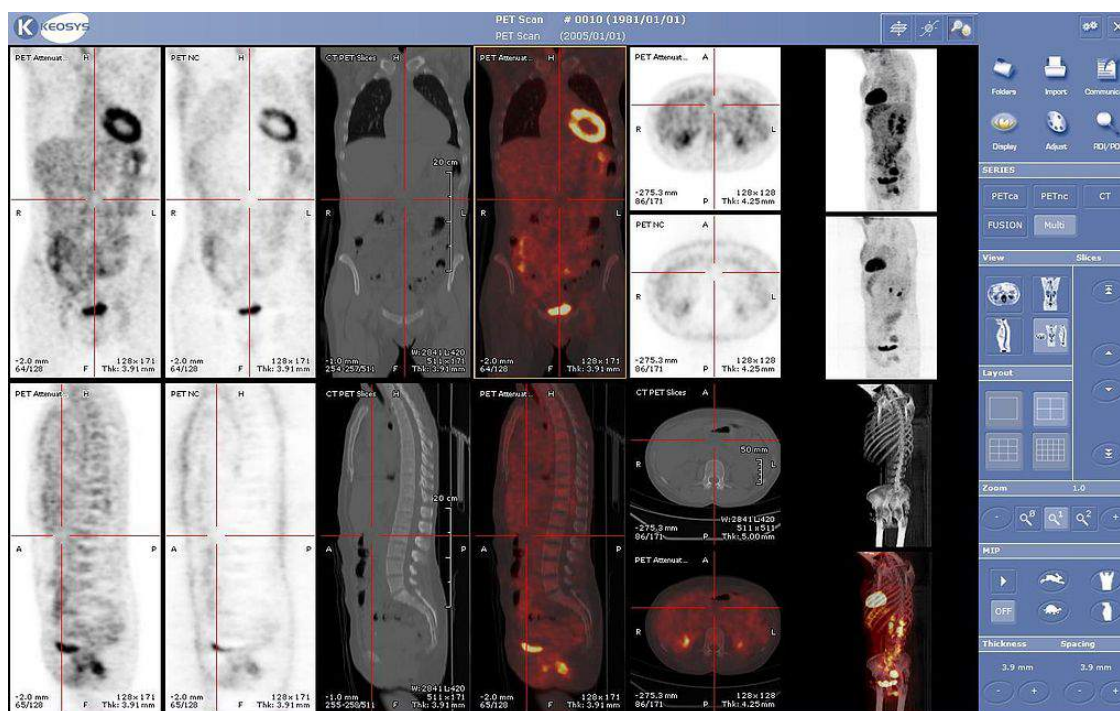


Figure 2.10: PET and PET-CT Fused Images. Taken from wikipedia.

Chapter 3

The role of CT and PET-CT in Radiotherapy.

3.1 Delimitation of tumor volumes by means of computed tomography (CT).

One year after the discovery of the X-rays by Roöntgen, began the use of the X-rays for the treatment of the cancer. With the isolation of the Radio on the part of the husbands Curie, it was begun with in use of the Radio for the treatment of the cancer. From there, several researchers took the trouble to go improving radiotherapy techniques [11]. Nowadays, the X-rays are used, only for diagnostic purposes, and the tumors are radiated with radioactive radioisotopes, and with electron linear accelerators (Linacs).



Figure 3.1: Software ECLIPSETM for planning in Radiation Therapy of head and neck tumors. Taken from [13].

To irradiate tumors, however, you must first locate them (if they exist), and once located the tumor, either by CT, or Positron Emission Tomography (PET), dosimetric planning begins [12], and from the images obtained by these techniques, the target volume is delineated. This target volume is used to perform a computer simulation with the ECLIPSETM software that was developed by Varian company [13]. In the software ECLIPSETM it is simulated what would be a radiotherapy treatment for diverse organs and the software show a profile of the dose radiation that would receive the white organ, and the organs to risk (OAR), which are the organs and adjacent tissues to the tumor, which also would be radiated [13].

The delimitation of target volume in radiation therapy was defined for the first time by the International Commission on Radiation Units & measurements (ICRU) in the ICRU's report 29, but took its final shape in the 50 and 62 reports [14]. At present it is still using the same convention for defining volumes.

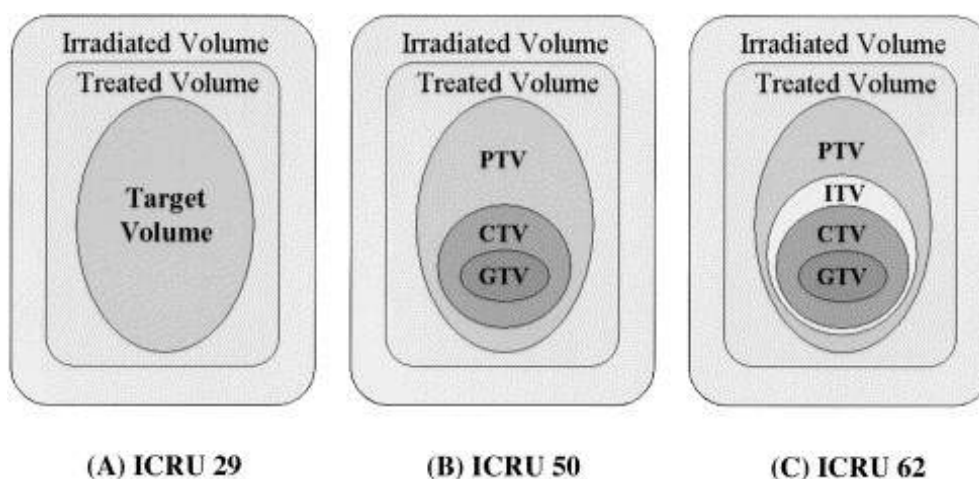


Figure 3.2: Target volumes defined by the ICRU. Taken from [14]

Figure 1, shows the different volumes to consider in a radiation treatment. The first of these is the GTV (gross tumor volume) that is the given volume on the diagnosis, with an additional margin that considers the possibility of tumor cells have spread to lymph nodes or tissue adjacent to the tumor, this is the CTV (Clinical Target Volume) [15]. It is possible that the tissue is in motion for physiological reasons and found that consider another additional margin which is referred to as ITV (Internal tumor volume). It is also possible that uncertainties by patient positioning among other uncertainties, by what is regarded as another additional margin known as PTV (Planning Target Volume) [15].

In the majority of hospitals (IPS) in Colombia where radiotherapies are realized, is usually performed the planning of radiation therapies, first by defining the tumor from Images from CT, and then the Radiation Oncologist, making use of the ECLIPSETM software, in a manually way delimits the target volumes. This delineation of volumes It is taken by the Medical Physicists to Perform the inverse planning, that is to define the type of particles (photons or electrons), the energy of the same ones, and the angles of incidence of the radiation beams of with which it goes to radiate to the patients. Such planning is done in function of the dose of radiation that is necessary to eradicate the tumor, and of the maximum dose that be able to tolerate the OAR in a fraction of the volume of the organ, which is what is known like constraint.

3.2 Problems with the CT in Radiotherapy Planning, and the Role of PET-CT.

CT images, are a map of values in Hounsfield units, that offer morphological information, and with the use of a contrast medium, the tissue density can be seen better, and also the pattern of vascularization in the tissues. However, in some tissues you do not have the sensitivity to gain sufficient morphological information, as you can go to certain tissues such as the brain, where the NMR can provide additional information [16].

The CT images, do not provide functional information, do not provide information about what changes are occurring in the tumor tissue. Although there are no morphological changes in the tissue, metabolic changes may be occurring more quickly, and this can be evaluated by studying the localization of some PET radiopharmaceutical, which allows to predict changes in tumor volume [16].

However, the PET, has very low resolution compared to CT. In addition, the detector array does not reach all of the photons produced, due to the fact that a large part are absorbed in the body of the patient, as such, it is a phenomenon of attenuation. However, this attenuation can be calculated, thanks to information provided by the CT [8]. As it was said previously, CT images are a map of values in Hounsfield units, and this are the attenuation coefficients of tissues, normalized to the attenuation coefficient of water.

Thus, the fusion of images provided by PET and CT (PET-CT), is more than the sum of the parts, the PET defects are corrected, and also it has the functional information which the alone CT lacks, this means that the image fusion provides functional and metabolic information [8]. Although the radiopharmaceuticals not always are localized in cancer tissues, the radiopharmaceutical also can be localized in tissues that are not cancerous; therefore you should have on hand images of CT, or MRI, among other pathological examinations, to compare with the information obtained with PET [17].

What remains is to define which metabolic process is the one you want to study using PET-CT. Taking into account that almost all the tumors have a high consumption of glucose, which is necessary for the maintenance of their metabolic processes, it is more common to find a molecule that is similar to glucose, and that also have within its structure, a positron emitter atom, this is the ^{18}F -FDG (^{18}F -FluoroDeoxyGlucose) [18]. As shown in the figure 3.3 (left), ^{18}F -FDG is entered in the cells using glucose transporters (GLUT), and is then chemically modified by an hexokinase enzyme (Hk II) and this new molecule cannot go out of the cell, remaining retained there [18]. Tumors also have regions where there is a lower amount of oxygen (hypoxia) and this is where the tumor cells are more radio resistant. For that reason, the dose of radiation should be higher there, that in the regions where there is more oxygen.

While ^{18}F -FDG, only allows you to view the metabolism of glucose to the length and breadth of the tumor, there are also other radiopharmaceuticals that allow you to visualize the oxygen metabolism. The most common PET tracers for imaging hypoxia are ^{18}F -MISO, ^{18}F -AZA y ^{64}Cu -ATSM. As shown in figure 3.3 (right), the first two have the particularity that as you enter the cell by diffusion, are reduced; if there is enough oxygen in the cell, are oxidized, returning to their chemical form non-ionized and out of the cell quickly. But if there is not enough oxygen, it is trapped within the cell, in its reduced form. Thus, in conditions of hypoxia, remain trapped in the interior of the cell, so allow to identify the hypoxic cells inside the tumor [19].

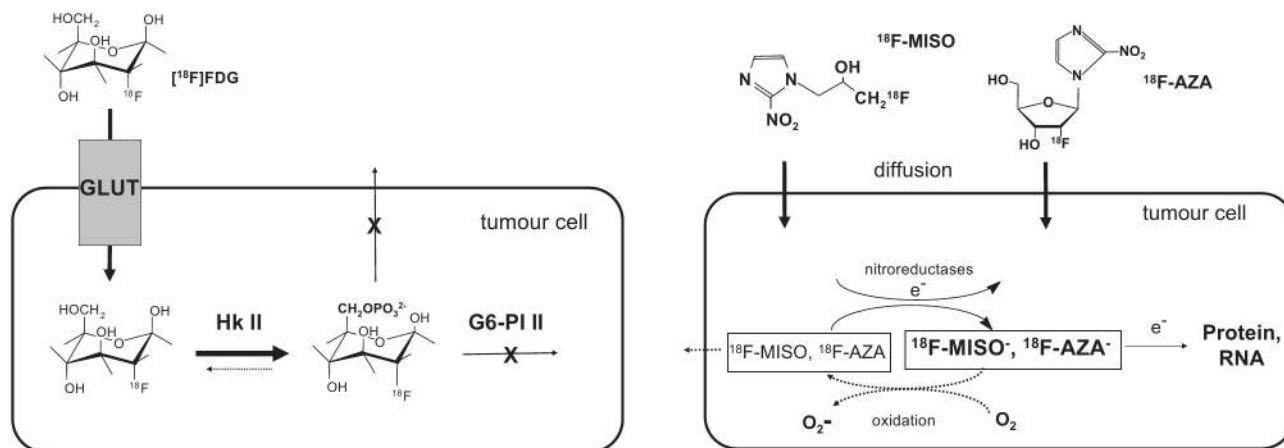


Figure 3.3: Scheme of the mechanism of localization for ¹⁸F-FDG (left), ¹⁸F-MISO and ¹⁸F-AZA (right). Taken from [19].

The radiopharmaceuticals cited above, among other, may be useful to make planning for radiation therapies, with PET-CT. The ¹⁸F-FDG is used to locate tumors and find the regions of greater glucose metabolism within the tumor, while ¹⁸F-MISO, ¹⁸F-AZA y ⁶⁴Cu-ATSM, are used to study the metabolism of oxygen in the tumor, thus enabling us to predict what would be the regions most radioresistant. In the more radioresistant regions, is where higher doses of radiation, must be given to ensure the success of the radiation therapy [19].

In the planning of radiotherapy that is done by CT, the different degrees of radio resistance that is present in the interior of the tumor can not be distinguished, in spite of that, tumors are radiated, by means of Intensity Modulated Radiation Therapy (IMRT), departing from CT images [20]. With the introduction of the planning of radiotherapies with PET-CT using the radiopharmaceuticals ¹⁸F-MISO, ¹⁸F-AZA y ⁶⁴Cu-ATSM, the treatment by means of IMRT, can improve significantly, concentrating the radiation beam, especially in the most radioresistant regions, and administering less radiation dose, in the less radioresistant regions [21, 22, 23].

Respect to ¹⁸F-FDG, since the radiation oncologist began to make planning for radiation therapies with PET-CT, using it as radiopharmaceutical, significant changes in the delimitation of the tumor volumes have been found, with regard to the volumes defined by means of CT [24]. In the tumor volume delimitation with PET-CT, a change in the GTV or in the CTV is obtained, with regard to the delimited volume with CT.

As will be shown below, an increase or a decrease in the GTV or in the CTV can be obtained in the volume delimitation with PET-CT. Although it is necessary to do the proviso, that in PET-CT, the GTV is called BTV (biological tumor volume), due to the metabolic information that is obtained, really is information about the biological behavior of the tumor [25, 26].

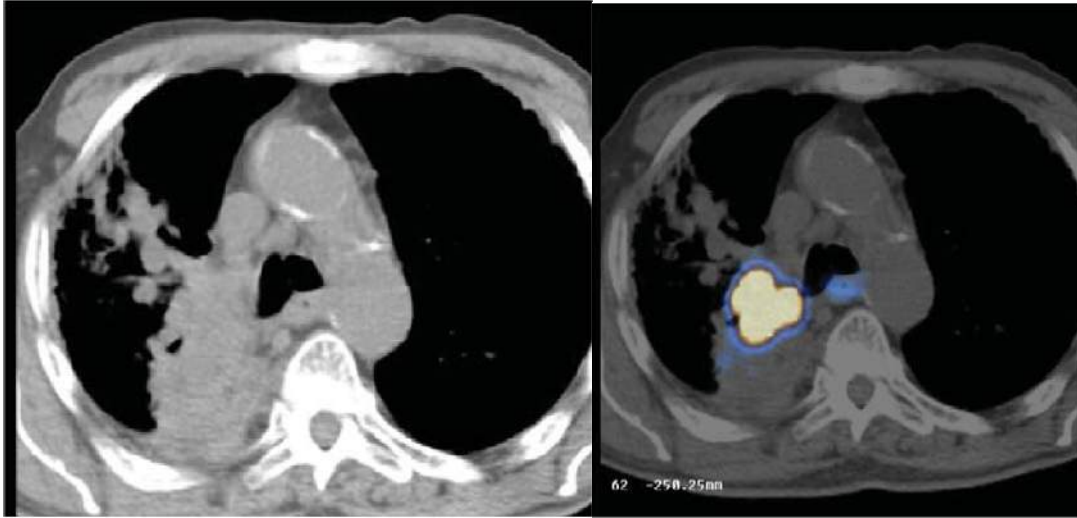


Figure 3.4: CT Image in a patient with a tumor and distal atelectasis in the right lung. Taken from [27].

In figure 3.4, there appear the diagnostic images taken from a patient with a tumor and distal atelectasis in the right lung. According to the CT image (left) cannot see distinction between the tumor and the collapsed lung, whereas in the image of PET-CT (right), the ^{18}F -FDG captation associated with the tumor can be seen in orange. In this case can be seen a significant reduction of GTV [27].

In the planning of radiation therapies with PET-CT, the GTV or the CTV, can also increase, since it can be found unsuspected tumors or metastases that cannot be seen with CT alone. For example, in the PET-CT (^{18}F -FDG) scan of a patient with locoregionally advanced squamous carcinoma of the base of tongue and neck (Figure 3.5), additional nodal disease in the left side of the neck, including a left supraclavicular node was found [12].

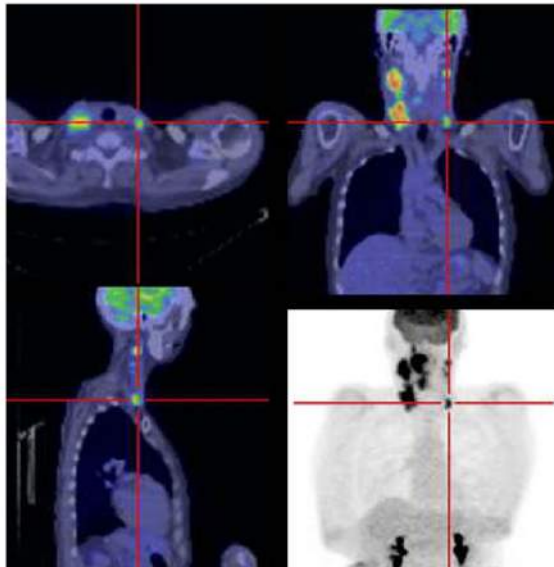


Figure 3.5: Additional tumor found in the planning of radiation therapy with PET-CT, indicated by cross-hairs. Taken from [12]

In summary, the planning of radiation therapies with PET-CT is much more advantageous than the planning with CT alone. The additional metabolic information allows to optimize the delimitation of tumors (^{18}F -FDG), and in addition allows to know the degree of the radiosensitivity of tumor regions (^{18}F -MISO, ^{18}F -AZA y ^{64}Cu -ATSM), thus contributes to the improvement of the radiotherapy treatment of tumors.

This work will focus on the optimization of the process of delimitation of tumor volumes with PET-CT, using ^{18}F -FDG as radiopharmaceutical, because in this moment is the only PET radiopharmaceutical that in Colombia has a wide use. Currently, in Colombia, the radiopharmacy of the INC, is doing clinical studies for the implementation of the use of ^{68}Ga radiopharmaceuticals, and is optimizing processes of synthesis of ^{18}F -FLT, and ^{11}C -methionine. It is expected that the software of delimitation of volumes that will be developed (and calibrated with images obtained for ^{18}F -FDG), will serve as the basis in the delimitation of tumors with other radiopharmaceuticals that begin to have a broad use, and can be used in planning for radiation therapies of tumors.

Chapter 4

Image processing with ROOT and OpenCV.

ROOT is an object-oriented framework designed for data analysis in high-energy physics. As a framework, ROOT has several libraries implemented which basically are classes of c++, and define different objects that are useful for the realization of mathematical calculations, image processing, presentation of results in graphs and histograms, among other functions.

On the ROOT's website (<http://root.cern.ch/root/html/tutorials/image/>) there are tutorials with enough examples. In particular, there are examples of photographic image processing. Below, are described two of these examples, and the way in which they can be used as a base for the implementation of image processing algorithms.

Code 4.1: Execution of ROOT and compilation of a script file.

```
helman@gerineldo:~$root
*****
*
*           W E L C O M E   t o   R O O T           *
*
*   Version      5.34/00           5 June 2012      *
*
*   You are welcome to visit our Web site         *
*           http://root.cern.ch           *
*
*****
ROOT 5.34/00 (branches/v5-34-00-patches@44555, Mar 14 2013,
11:26:00 on linuxx8664gcc)
CINT/ROOT C/C++ Interpreter version 5.18.00, July 2, 2010
Type ? for help. Commands must be C++ statements.
Enclose multiple statements between { }.
root [0] .x image2hist.C
```

Code 4.2: image2hist.C example

```

//Create a 2-D histogram from an image.
//Author: Olivier Couet
void image2hist() {
    TASImage image("rose512.jpg");
    UInt_t yPixels = image.GetHeight();
    UInt_t xPixels = image.GetWidth();
    UInt_t *argb = image.GetArgbArray();
    TH2D* h = new TH2D("h", "Rose_histogram", xPixels, -1, 1, yPixels, -1, 1);
    for (int row=0; row<xPixels; ++row) {
        for (int col=0; col<yPixels; ++col) {
            int index = col*xPixels+row;
            float grey = float(argb[index]&0xff)/256;
            h->SetBinContent(row+1, yPixels - col, grey);
        }
    }
    gStyle->SetPalette(53);
    h->Draw("colz"); }

```

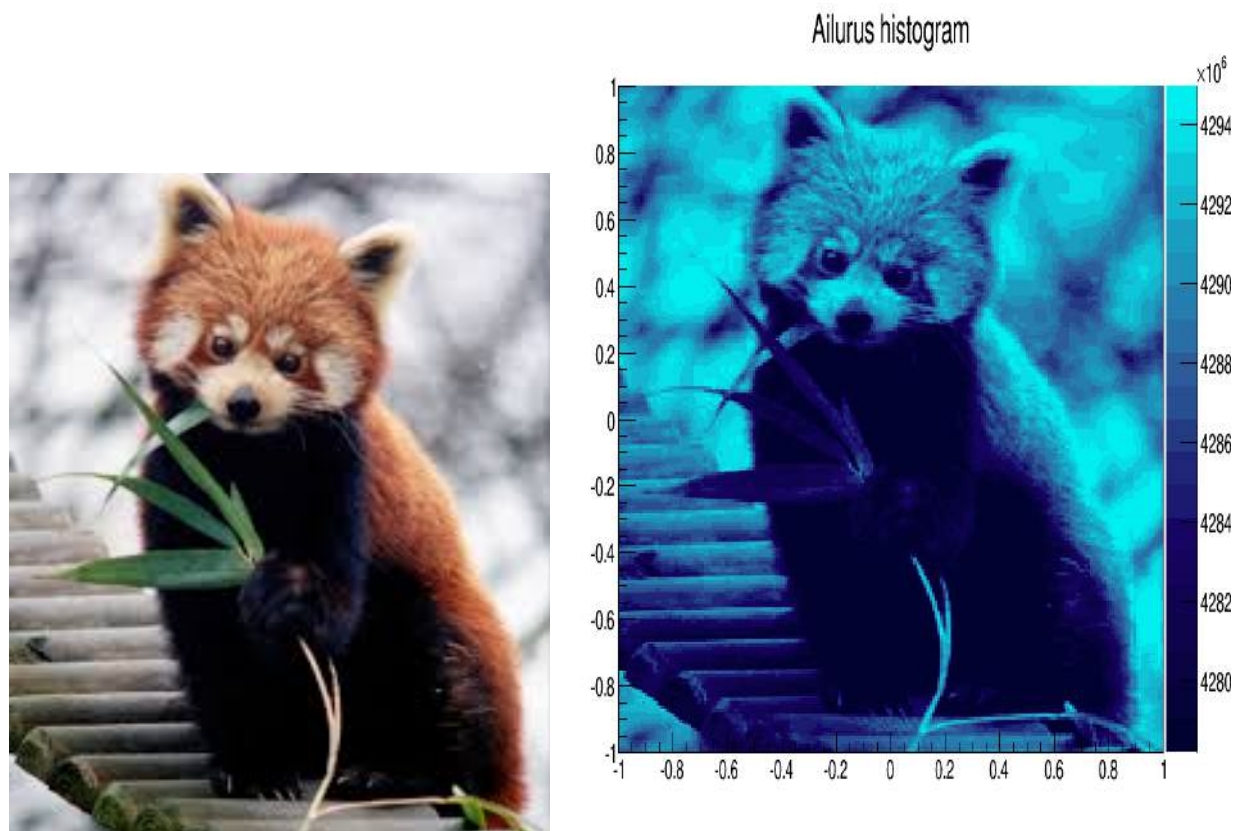


Figure 4.1: Original Image (*Ailurus fulgens*, taken from Wikipedia), and a histogram of the same image, obtained by means of the example image2hist.C

The example `image2hist.C` (Code 4.2) is a code that allows to convert an image file into a ROOT histogram. To compile a script of root, as the code 4.2, it is only necessary to open root from a console and then run `.x image2hist.C`, as shown in the code 4.1

To read an image file, the first step is to define the file image as an object of `TASImage` class and then to obtain both the width and the height of the image, by means of the functions `image.GetHeight()` and `image.GetWidth()` proper of this class. The next step is to convert the image to an `argb` vector, whose components can be read using a for loop. To obtain a histogram from the image, first it is necessary to define a 2D histogram as object of the class `TH2D`, and then by means of a loop for, the histogram is filled with every component of the vector `argb` obtained previously in the phase of reading of the image. In the figure 4.1 an image is shown, together with the generated histogram.

By means of the function `gStyle` (from the class `TStyle`), it is possible to define the colors palette to visualize the histogram with a pseudocolor, that is blue in the showed example. The histogram, is plotted with the option "colz", which is a flat display, but if the histogram is plotted with the option "surf2", a three-dimensional visualization like the showed in the figure 4.2 is obtained. In the same figure it is possible to see that a photographic image, in fact is a flat representation of a function of two spatial coordinates, which is the the color intensity $I(x,y)$.

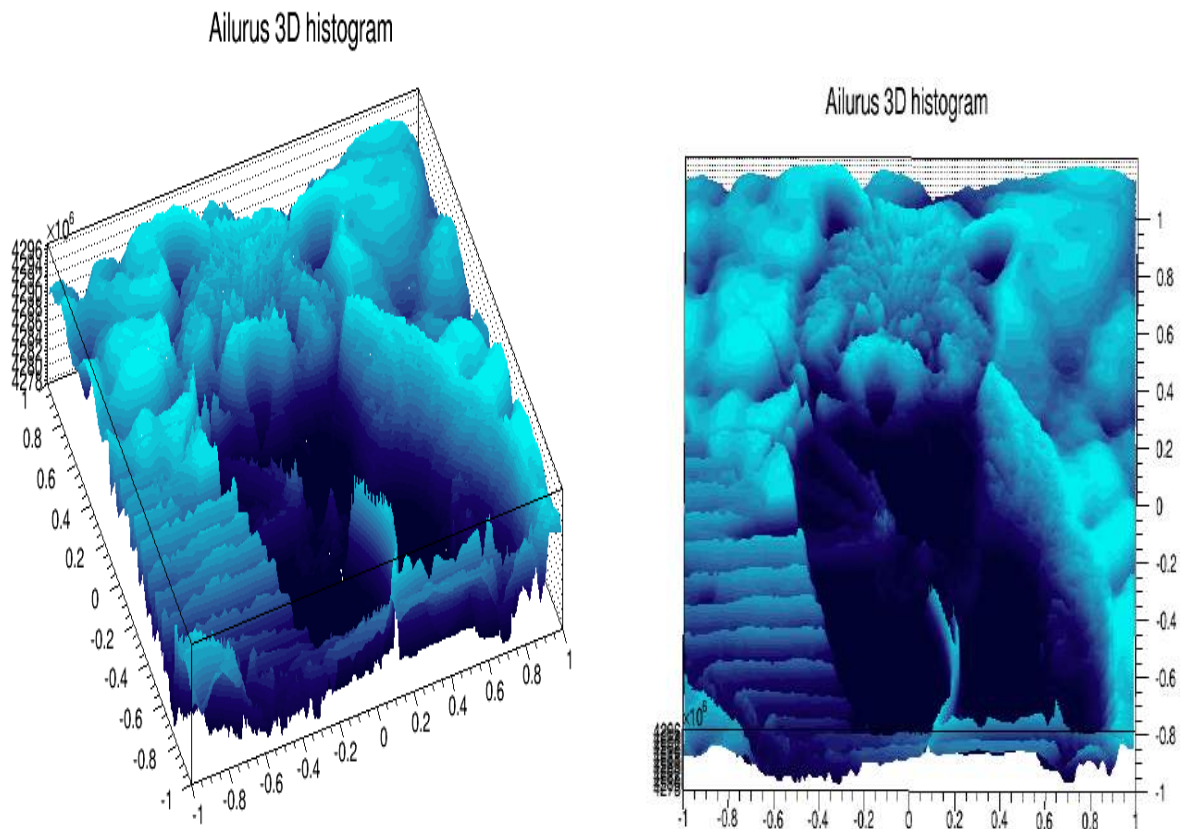


Figure 4.2: 3D visualization of a TH2D histogram.

ROOT also has a discussion forum, where any registered user can ask questions, so that another user answers it. An example that can be found in one of these forums, is how to convert an image into a ARGB matrix (<http://root.cern.ch/phpBB3/viewtopic.php?f=3&t=14070>). This example is showed in the code 4.3.

Code 4.3: TASImage data example

```
//Author: Hobbybastler
TASImage* im = TASImage::Open("test.jpg");
TImagePalette pal = im->GetPalette();
UInt_t w = im->GetWidth();
UInt_t h = im->GetHeight();
    UInt_t* arr = im->GetArgbArray();
    TMatrixD A(w,h); // (this is the transposed image)
TMatrixD R(w,h);
TMatrixD G(w,h);
TMatrixD B(w,h);
for (int i=0;i<w;i++){
    for (int j=0;j<h;j++){
        int t = j*w+i;
        UInt_t val = arr[t];
        A[i][j] = (UInt_t)((val >> 24) & 0xFF);
        R[i][j] = (UInt_t)((val >> 16) & 0xFF);
        G[i][j] = (UInt_t)((val >> 8) & 0xFF);
        B[i][j] = (UInt_t)((val >> 0) & 0xFF);
    }
}
//
// ... apply some operations on the matrices ...
//     for (int i=0;i<w;i++){
for (int j=0;j<h;j++){
    unsigned int val = ((unsigned int)A[i][j])<<24;
    val = val + (((unsigned int)R[i][j])<<16);
    val = val + (((unsigned int)G[i][j])<<8);
    val = val + ((unsigned int)B[i][j]);
    arr[j*w+i] = val;
}
}
im->Draw("xyz");
```

In this example, the image file is converted into an argb vector, and each component of the vector is a hexagesimal number with the form 0xAARRGGBB, and the bits associated with each color can be extracted. For example, the bits of green color can be extracted leaving only the bits 9 to 16, and this is done using the filter `UInt_t)((val >> 8) & 0xFF)`. The information for each color is stored with `TMatrixD`, that is a class that contains functions that performs typical operations between matrices, and operations between matrix elements.

4.1 Image Color extraction and Convolution.

From the codes 4.2 and 4.3, it is possible to make a new code (code 4.3) useful to extract each color matrix of an image, and to plot it in a TH2D histogram (figure 4.3). For the processing of images with the code 3 and generate both the histogram, as the text file, it is necessary to include the libraries `iostream`, `fstream`, `math.h`, `TCanvas.h`, `TFile.h`, `TMath.h`, and `TColor.h`.

Code 4.4: image2TH2D.C code

```
// visualize a 2-D histogram from a processed image.
// Author Helman Amaya
#include <iostream>
#include <fstream>
#include <math.h>
#include <TCanvas.h>
#include "TH2.h"
#include "TFile.h"
#include "TMath.h"
#include "TColor.h"

void image2TH2D(){

    TASIImage* im = TASIImage::Open("ailurus.jpg");
    TImagePalette pal = im->GetPalette();
    UInt_t w = im->GetWidth();
    UInt_t h = im->GetHeight();
    // To obtain ARGB Values
    UInt_t* arr = im->GetArgbArray();

    // Definition of the Intensity root square matrix
    TMatrixD A(w,h);
    for (int i=0;i<w;i++){
        for (int j=0;j<h;j++){
            int t = j*w+i;
            UInt_t val = arr[t];
            // Extracting the colors
            UInt_t alpha = (UInt_t)((val >> 24)&0xFF);
            UInt_t red = (UInt_t)((val >> 16)&0xFF);
            UInt_t green = (UInt_t)((val >> 8)&0xFF);
            UInt_t blue = (UInt_t)((val >> 0)&0xFF);
            // Intensity matrix
            A[i][j]=sqrt(red*red+green*green+blue*blue);
        }
    }
}
```

```

// Exporting Matrix A into a plain text file
ofstream fs("pixel.dat");
    for (int i=0;i<w;i++){
        for (int j=0;j<h;j++){
fs<<i<<"\t"<<j<<"\t"<<A[i][j]<<endl;
        }
    }
fs.close();

//Histogram
TH2D* m = new TH2D("m", "Ailurus_□Blue_□histogram",w,0,w,h,0,h);
for (int i=0;i<w;i++){
    for (int j=0;j<h;j++){
        int t = j*w+i;
        UInt_t val = arr[t];
        UInt_t alpha = (UInt_t)((val >> 24)&0xFF);
        UInt_t red = (UInt_t)((val >> 16)&0xFF);
        UInt_t green = (UInt_t)((val >> 8)&0xFF);
        UInt_t blue = (UInt_t)((val >> 0)&0xFF);
        UInt_t pixel = 1 + blue;
        //UInt_t pixel = 1 + (red+green+blue)/3;
        m->SetBinContent(i+1,h-j,pixel);
    }
}

/*Int_t MyPalette[100];
Double_t r[] = {1.00, 0.84, 0.61, 0.34, 0.00};
Double_t g[] = {1.00, 0.84, 0.61, 0.34, 0.00};
Double_t b[] = {1.00, 0.84, 0.61, 0.34, 0.00};
Double_t stop[] = {0.00, 0.34, 0.61, 0.84, 1.00};
Int_t FI = TColor::CreateGradientColorTable(5, stop, r, g, b, 100);
for (int i=0;i<100;i++) MyPalette[i] = FI+i;
gStyle->SetPalette(100, MyPalette);*/
gStyle->SetPalette(53);
gStyle->SetOptStat(0);
m->Draw("colz");
}

```

In the code 4.3, was introduced a routine that exports the intensity data of the image into a plane text file. The file can be checked, to determine whether the obtained data make sense or not. Inside the definition of the histogram, like a comment was introduced a small algorithm that average the colors, to obtain an image in a grayscale, as shown in the figure 4.3. In a similar way, the averages described in the section 2.1 can be implemented.

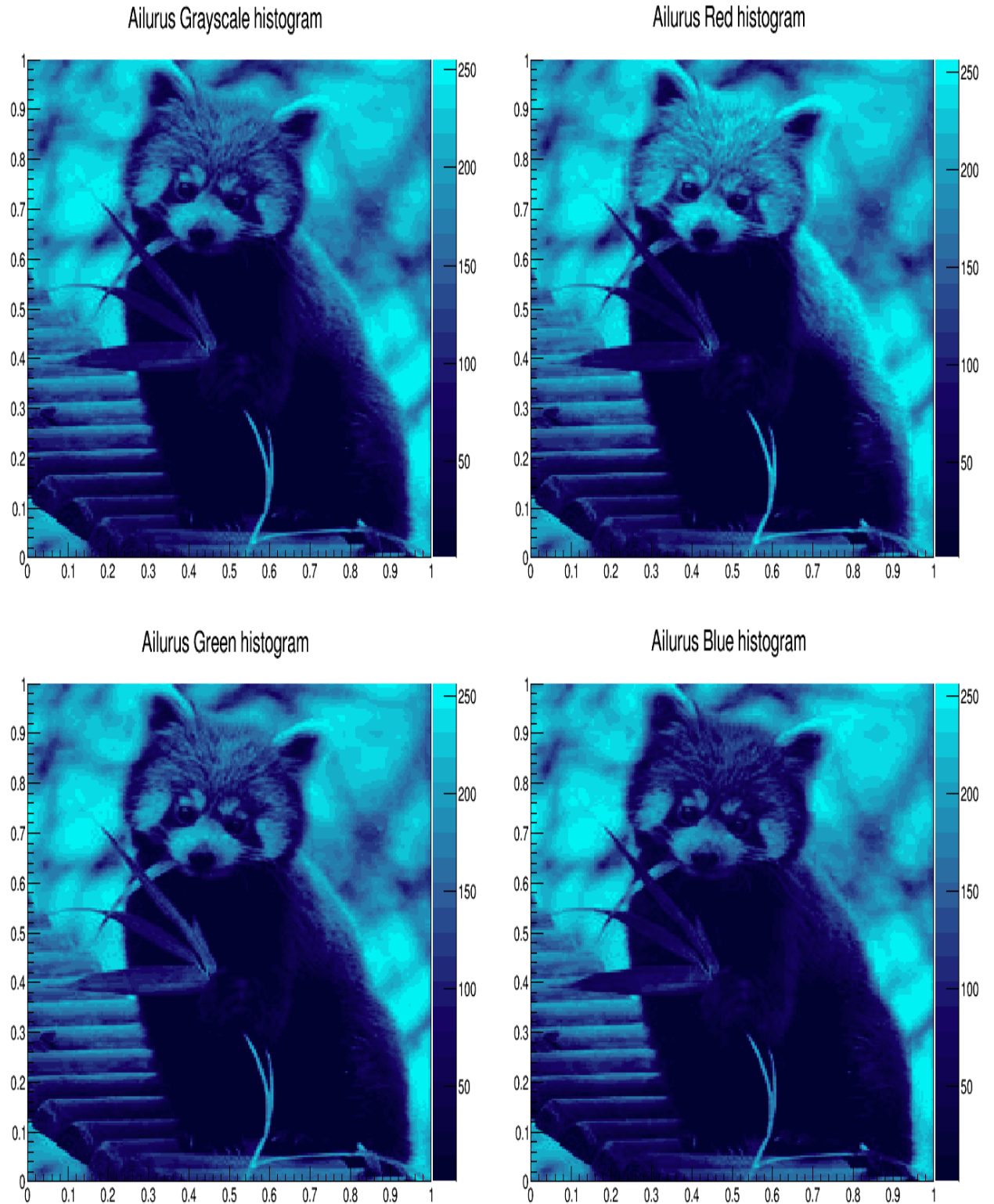


Figure 4.3: Histograms of a test image for each color, and a grayscale

In the figure 4.3, in addition to showing the histogram in a grayscale, are showed histograms of the different colors that compose the test image. All this histograms have values between 0 and 256, and they have blue pseudocolor because the color palette that was used. However, a custom color palette such as the commented in the code 4.3 can be used, and it displays a histogram with a pseudocolor in an inverted grayscale.

At the line where the histogram is defined, you must define a coordinate system, useful to locate each pixel within the canvas. In the images of the figure 4.3, is used a reference system between 0 and 1 in x and y coordinates; nevertheless a reference system can be defined between zero and the length or breadth of the image, to make a more correct location of the position of each pixel (figure 4.4).

In the code 4.4, the histogram can be redefined to obtain the color intensity value of each image pixel, in a grayscale, and store each of these values in a matrix, in which convolutions could be applied. For this reason, the histogram of the code 4.4 is modified, and there, matrix operations typical of every convolution, are introduced.

Code 4.5: Histogram to make a gaussian blur

```
//Histogram
TMatrixD B(w,h); // (Convolutated Intensity image)
TMatrixD C(w,h); // (Intensity image)
TH2D* m = new TH2D("m", "Ailurus_blur_histogram", w, 0, w, h, 0, h);
for (int i=1; i<w-1; i++){
    for (int j=1; j<h-1; j++){
        int t = j*w+i;
        UInt_t val = arr[t];
        UInt_t alpha = (UInt_t)((val >> 24)&0xFF);
        UInt_t red = (UInt_t)((val >> 16)&0xFF);
        UInt_t green = (UInt_t)((val >> 8)&0xFF);
        UInt_t blue = (UInt_t)((val >> 0)&0xFF);
        C[i][j]=(red+green+blue)/3;

        // Convolution
        B[i][j]=(1*C[i-1][j-1]+2*C[i][j-1]+1*C[i+1][j-1]
                +2*C[i-1][j] +4*C[i][j] +2*C[i+1][j]
                +1*C[i-1][j+1]+2*C[i][j+1]+1*C[i+1][j+1])/16;
        //B[i][j]= C[i][j];
        Double_t pixel = 1 + B[i][j];
        m->SetBinContent(i+1, h-j, pixel);
    }
}
```


In the code 4.5, a Gaussian convolution with variance $\sigma = 1.4$ is introduced, however it is only necessary to change the coefficients and the denominator, to obtain other convolutions of the image like that appear in the image 4.4. In the table 4.1, some convolution matrices (kernels) of those shown as an example in the documentation for the gimp software (<http://docs.gimp.org/2.6/C/plugin-in-convmatrix.html>) are shown.

Gaussian Blur	Sharpen	Edge detection	Edge enhance
$\frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$

Table 4.1: Some convolution matrices (kernels).

For the array of neighboring pixels to a certain pixel A_{ij} :

$$\begin{pmatrix} A_{(i-2)(j-2)} & A_{(i-2)(j-1)} & A_{(i-2)j} & A_{(i-2)(j+1)} & A_{(i-2)(j+2)} \\ A_{(i-1)(j-2)} & A_{(i-1)(j-1)} & A_{(i-1)j} & A_{(i-1)(j+1)} & A_{(i-1)(j+2)} \\ A_{i(j-2)} & A_{i(j-1)} & A_{ij} & A_{i(j+1)} & A_{i(j+2)} \\ A_{(i+1)(j-2)} & A_{(i+1)(j-1)} & A_{(i+1)j} & A_{(i+1)(j+1)} & A_{(i+1)(j+2)} \\ A_{(i+2)(j-2)} & A_{(i+2)(j-1)} & A_{(i+2)j} & A_{(i+2)(j+1)} & A_{(i+2)(j+2)} \end{pmatrix}$$

If only the first neighbors are considered, the "neighbors matrix" now is like that:

$$\begin{pmatrix} A_{(i-1)(j-1)} & A_{(i-1)j} & A_{(i-1)(j+1)} \\ A_{i(j-1)} & A_{ij} & A_{i(j+1)} \\ A_{(i+1)(j-1)} & A_{(i+1)j} & A_{(i+1)(j+1)} \end{pmatrix}$$

Each element of the matrix, can be multiplied by a number determined by the convolution matrix. For example, a Gaussian blur can be obtained using the equation:

$$\begin{aligned} 16 \cdot B_{ij} = & A_{(i-1)(j-1)} + 2A_{ij-1} + A_{(i+1)j-1} \\ & + 2A_{(i-1)j} + 4A_{ij} + 2A_{(i+1)j} \\ & + A_{(i-1)j+1} + 2A_{i(j+1)} + A_{(i+1)(j+1)} \end{aligned}$$

That is **not** the matrix multiplication between the first neighbors matrix and the convolution matrix. By plotting the values of B_{ij} , a histogram is obtained as shown in figure 4.4, top right.

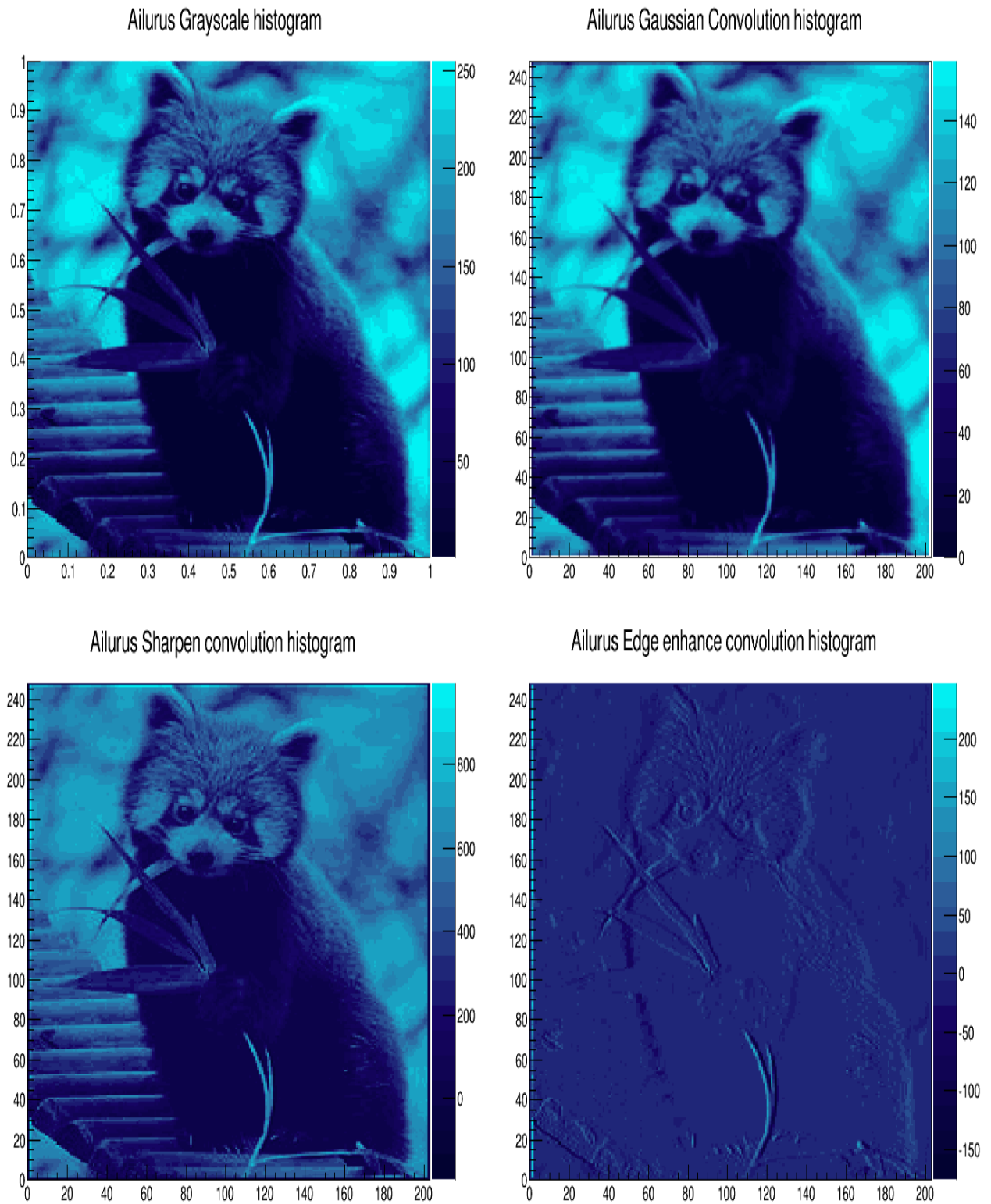


Figure 4.4: Original Image in a grayscale, and images obtained by applying different convolutions.

4.2 Image Edge detection and enhancement.

4.2 Each image that is wanted to process, can be seen as a continuous function of two variables $f(x,y)$, that is a function of the spatial coordinates x , and y , so the A_{ij} matrix (that was described in section 4.1), can be understood as the mapping of the function $f(x,y)$ when continuous variables x , and y take discrete values i and j , i.e. $A_{ij} = f(i, j)$.

In a photographic image, every object has an associated edge, where drastic changes occur in levels of intensity, and hence, the partial derivatives of the intensity of each color, provides important information regarding the presence of edges. In particular, the gradient vector ∇f indicates the direction of maximum variation of the function $f(x,y)$ which, in this case, represents the intensity $I(x,y)$. The Laplacian $\nabla^2 f$ also provides information regarding the presence of edges, so the information provided by the gradient vector or the Laplacian, is useful to detect edges. To enhance edges there are several methods, one of them is the convolution shown in the table 4.1.

4.2.1 Gradient Magnitude and Laplace edge enhancement with ROOT.

Based on the properties of the gradient vector, John Canny developed a method for the edge detection in images [28], which at present is known as the Canny edge detector, and this algorithm can be summarized in the following 5 steps [29]:

1. Smooth the pixels of the image, by applying a filter to reduce the noise.
2. Calculate the partial derivatives of the intensities, by means of the method of approach of derivatives of your preference.
3. Calculate the magnitude of the gradient vector for each pixel.
4. Remove those pixels that are not local maxima of the magnitude in the direction of the gradient.
5. Thresholding with hysteresis:
 - Select the pixels such that $M > T \{h\}$ (high threshold)
 - Collect the pixels such that $M > T \{l\}$ (low threshold), which are neighbors of the pixel selected as edge

If only the first three steps are implemented, the edges of the image will be enhanced, and this is what will be done. The first step was already done, because to make an image smoothing, a Gaussian Blur is applied in the image, as it was done in the previous section. For the second step, the partial derivatives can be calculated by the method of first-order finite difference [30]:

$$\frac{\partial f}{\partial x}(x,y) \approx \frac{f(x+\Delta x,y) - f(x,y)}{\Delta x}, \quad \frac{\partial f}{\partial y}(x,y) \approx \frac{f(x,y+\Delta y) - f(x,y)}{\Delta y}$$

With an error of order Δx and Δy , respectively. In terms of the matrix A_{ij} :

$$\frac{\partial f}{\partial x}(i, j) \approx f(i+1, j) - f(i, j) = A_{(i+1)j} - A_{ij}$$

$$\frac{\partial f}{\partial y}(i, j) \approx f(i, j+1) - f(i, j) = A_{i(j+1)} - A_{ij}$$

The edge enhance kernel showed in the table 4.1, and whose histogram is shown in the lower right image of the figure 4.4, can also be seen as:

$$\frac{\partial f}{\partial x}(i-1, j) \approx f(i, j) - f(i-1, j) = A_{ij} - A_{(i-1)j}$$

Therefore, the calculation of partial derivatives of the pixels intensities, is useful for edge enhancement. On the other hand, the partial derivatives also can be calculated by the method of second-order finite difference [30]:

$$\frac{\partial f}{\partial x}(x, y) \approx \frac{f(x + \Delta x, y) - f(x - \Delta x, y)}{2\Delta x}, \quad \frac{\partial f}{\partial y}(x, y) \approx \frac{f(x, y + \Delta y) - f(x, y - \Delta y)}{2\Delta y}$$

With an error of order Δx^2 and Δy^2 , respectively [30]. In terms of the matrix A_{ij} :

$$f_x(i, j) = \frac{\partial f}{\partial x}(i, j) \approx \frac{f(i+1, j) - f(i-1, j)}{2} = \frac{A_{(i+1)j} - A_{(i-1)j}}{2} \quad (4.1)$$

$$f_y(i, j) = \frac{\partial f}{\partial y}(i, j) \approx \frac{f(i, j+1) - f(i, j-1)}{2} = \frac{A_{i(j+1)} - A_{i(j-1)}}{2} \quad (4.2)$$

Once the partial derivatives were calculated, it is possible to calculate the norm of the gradient vector, in the same way as the norm of any vector is calculated [31]:

$$|\nabla f(i, j)| = \sqrt{\left(\frac{\partial f}{\partial x}(i, j)\right)^2 + \left(\frac{\partial f}{\partial y}(i, j)\right)^2}$$

$$|\nabla I(i, j)| \approx \sqrt{\left(\frac{A_{(i+1)j} - A_{(i-1)j}}{2}\right)^2 + \left(\frac{A_{i(j+1)} - A_{i(j-1)}}{2}\right)^2} \quad (4.3)$$

To determine the direction of the gradient vector, it is only necessary to know the angle α , which is calculated as:

$$\alpha = \arctan\left(\frac{\partial_y}{\partial_x}\right) \approx \arctan\left(\frac{A_{i(j+1)} - A_{i(j-1)}}{A_{(i+1)j} - A_{(i-1)j}}\right)$$

The direction of the gradient vector in a given pixel (i, j) indicates the direction of maximum variation of the intensity $I(i, j)$, and this is what is applied in the next steps of the Canny Edge Detector [28].

The second order derivatives of $f(i, j)$ can be calculated by means of the same second-order approximations for the derivatives:

$$\frac{\partial^2 f}{\partial x^2}(i, j) \approx \frac{f_x(i+1, j) - f_x(i-1, j)}{2} = \frac{B_{(i+1)j} - B_{(i-1)j}}{2}$$

$$\frac{\partial^2 f}{\partial y^2}(i, j) \approx \frac{f_y(i, j+1) - f_y(i, j-1)}{2} = \frac{C_{i(j+1)} - C_{i(j-1)}}{2}$$

With an error of order Δx^2 and y^2 , respectively. To make the calculation of second-order derivatives, it is necessary to store the values of the partial derivatives f_x y f_y as the matrices B_{ij} and C_{ij} . With the values of the second-order partial derivatives, it is possible to calculate the Laplacian [31]:

$$\begin{aligned} \nabla^2 f(i, j) &= \frac{\partial^2 f}{\partial x^2}(i, j) + \frac{\partial^2 f}{\partial y^2}(i, j) \\ \nabla^2 I(i, j) &\approx \frac{B_{(i+1)j} - B_{(i-1)j}}{2} + \frac{C_{i(j+1)} - C_{i(j-1)}}{2} \end{aligned} \quad (4.4)$$

If only the two first steps of the Marr-Hildreth edge detector [29] are made (to make a gaussian smoothing and then apply a Laplacian), an edge enhancement is achieved, as when only the first three steps of the Canny edge detector are performed (figures 4.5 and 4.6).

As the intensity undergoes dramatic changes in the edges, its spatial derivatives change abruptly with the variation of the position, just in the edges. For that reason, either the gradient vector magnitude or the Laplacian are useful to obtain histograms where the edges are enhanced, as shown in the figures 4.5 and 4.6. As can be seen in both figures, the edge enhancing is better when a previous smoothing is applied. For a better visualization of the histograms showed in both figures, the value in each pixel was obtained by subtracting each calculated value (magnitude of the gradient or Laplacian) to the respective maximum value.

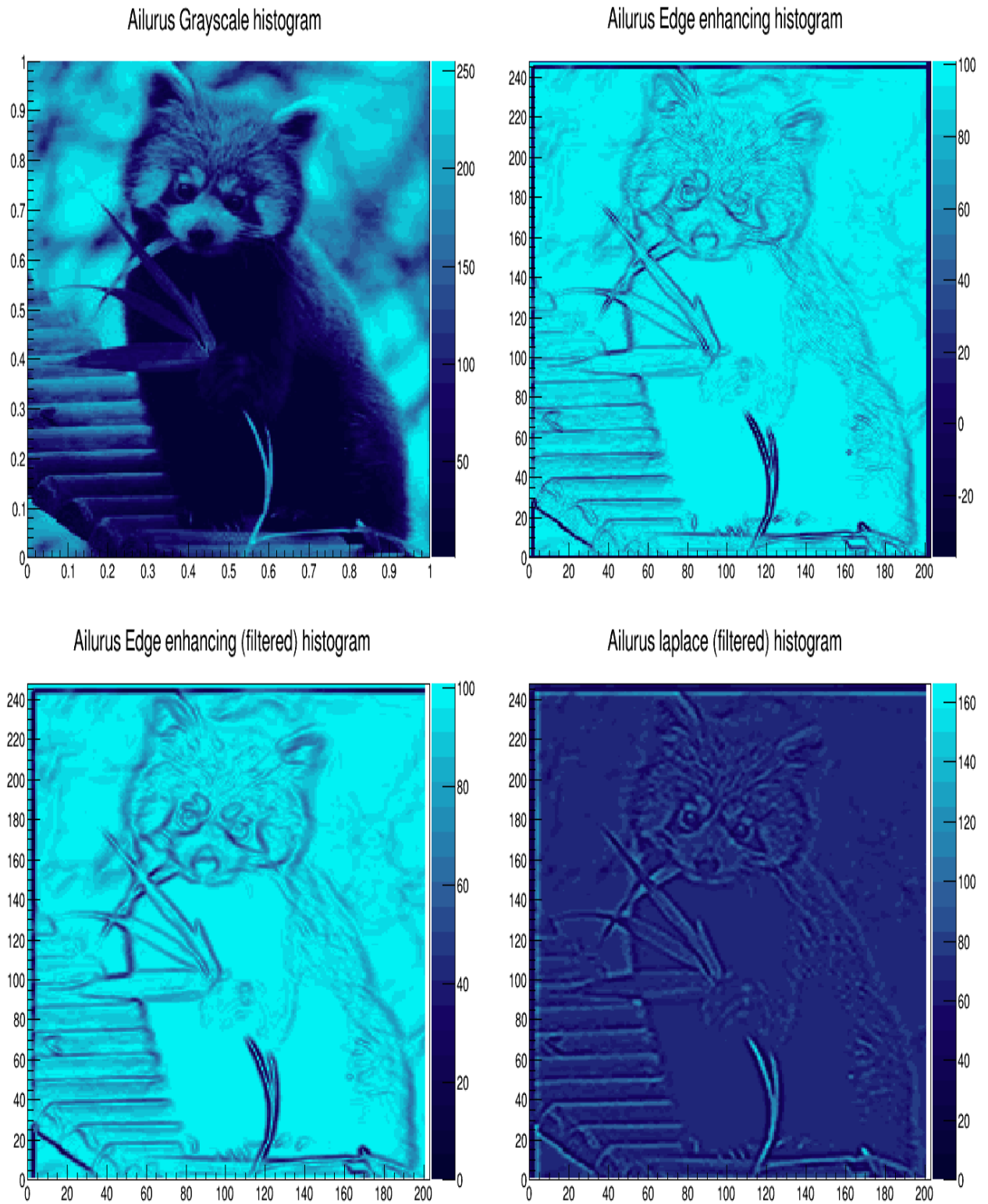


Figure 4.5: Ailurus grayscale image and its edge enhancing.

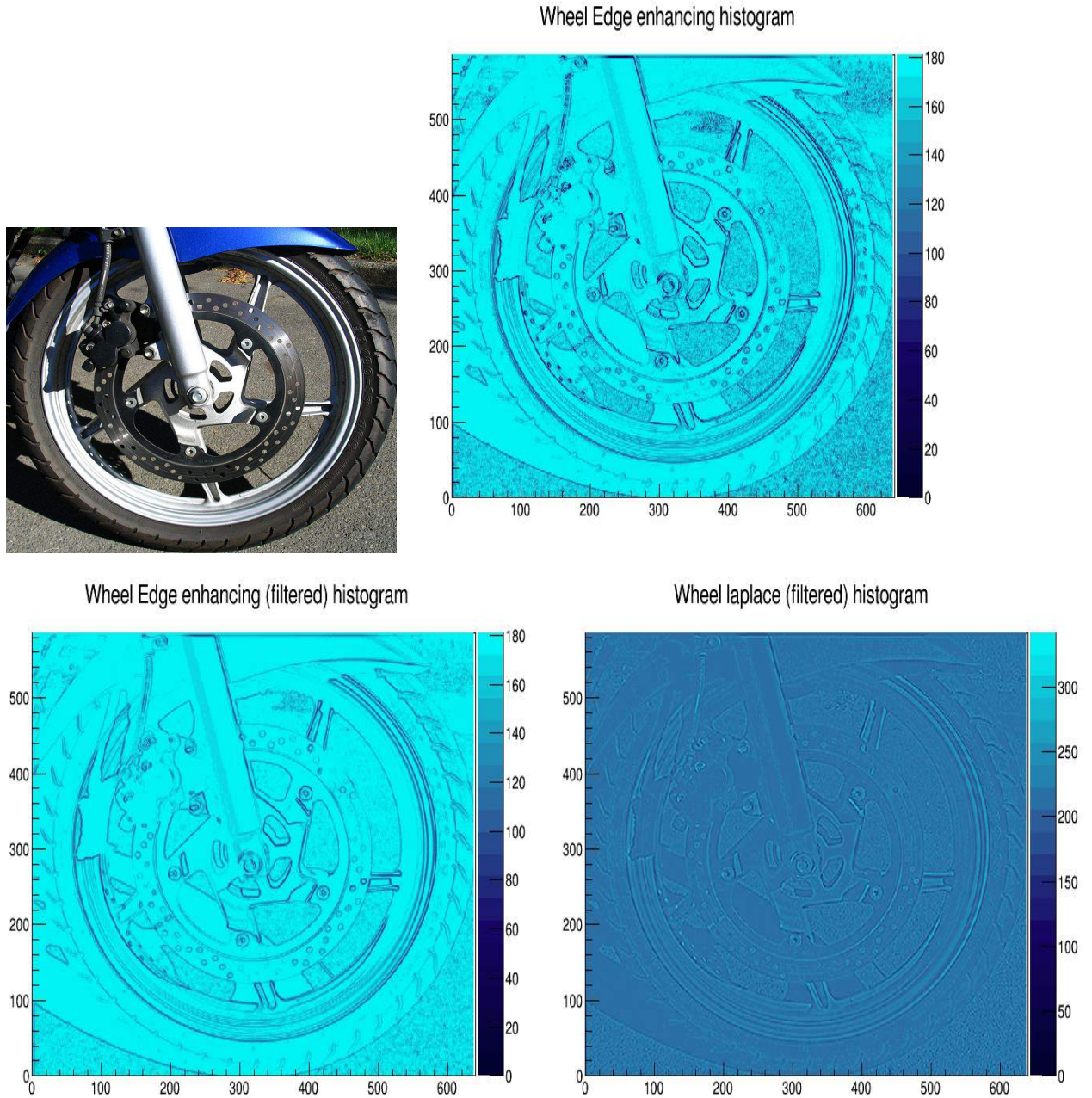


Figure 4.6: Wheel image and its edge enhancing. Original Image taken from Wikipedia

4.2.2 Canny edge detection with OpenCV .

OpenCV (Open Source Computer Vision) is a library of programming functions for the real-time computer vision, that focuses mainly on real-time image processing. Its documentation can be found in <http://docs.opencv.org/index.html>.

Code 4.6: OpenCV Canny tutorial. Taken from http://docs.opencv.org/doc/tutorials/imgproc/imgtrans/canny_detector/canny_detector.html

```
#include "opencv2/imgproc/imgproc.hpp"
#include "opencv2/highgui/highgui.hpp"
#include <stdlib.h>
#include <stdio.h>
using namespace cv;
/// Global variables
Mat src, src_gray; Mat dst, detected_edges;
int edgeThresh = 1;
int lowThreshold;
int const max_lowThreshold = 100;
int ratio = 3; int kernel_size = 3;
char* window_name = "Edge_Map";
/* function CannyThreshold */
void CannyThreshold(int, void*) {
    blur( src_gray, detected_edges, Size(3,3) );
    Canny(detected_edges, detected_edges, lowThreshold,
        lowThreshold*ratio, kernel_size);
    dst = Scalar::all(0);
    src.copyTo( dst, detected_edges);
    imshow( window_name, dst ); }
/*function main*/
int main( int argc, char** argv ) {
    src = imread( argv[1] );
    if( !src.data )
    { return -1; }
    dst.create( src.size(), src.type() );
    cvtColor( src, src_gray, CV_BGR2GRAY );
    namedWindow( window_name, CV_WINDOW_AUTOSIZE );
    createTrackbar( "Min_Threshold:", window_name, &lowThreshold,
        max_lowThreshold, CannyThreshold );
    CannyThreshold(0, 0);
    waitKey(0);
    return 0;
}
```

OpenCV has the Canny function inside its functions, where the Canny edge detection method is implemented, and within the Image Processing examples, a tutorial example explains how the above mentioned function can be used in the edge detection of an image (code 5.1).

In the above mentioned code example, the image is first read, and then it is stored in a matrix. The matrix image is converted to a grayscale and the results are stored in a new matrix where, first a Gaussian Blur is applied and then the Canny function is used. To display the results, the visualization options are defined, one of them is the minimum threshold that can be manually defined by the user during the edge image visualization. Figure 4.7, shows the edge detection in a test image, using the code 5.1.

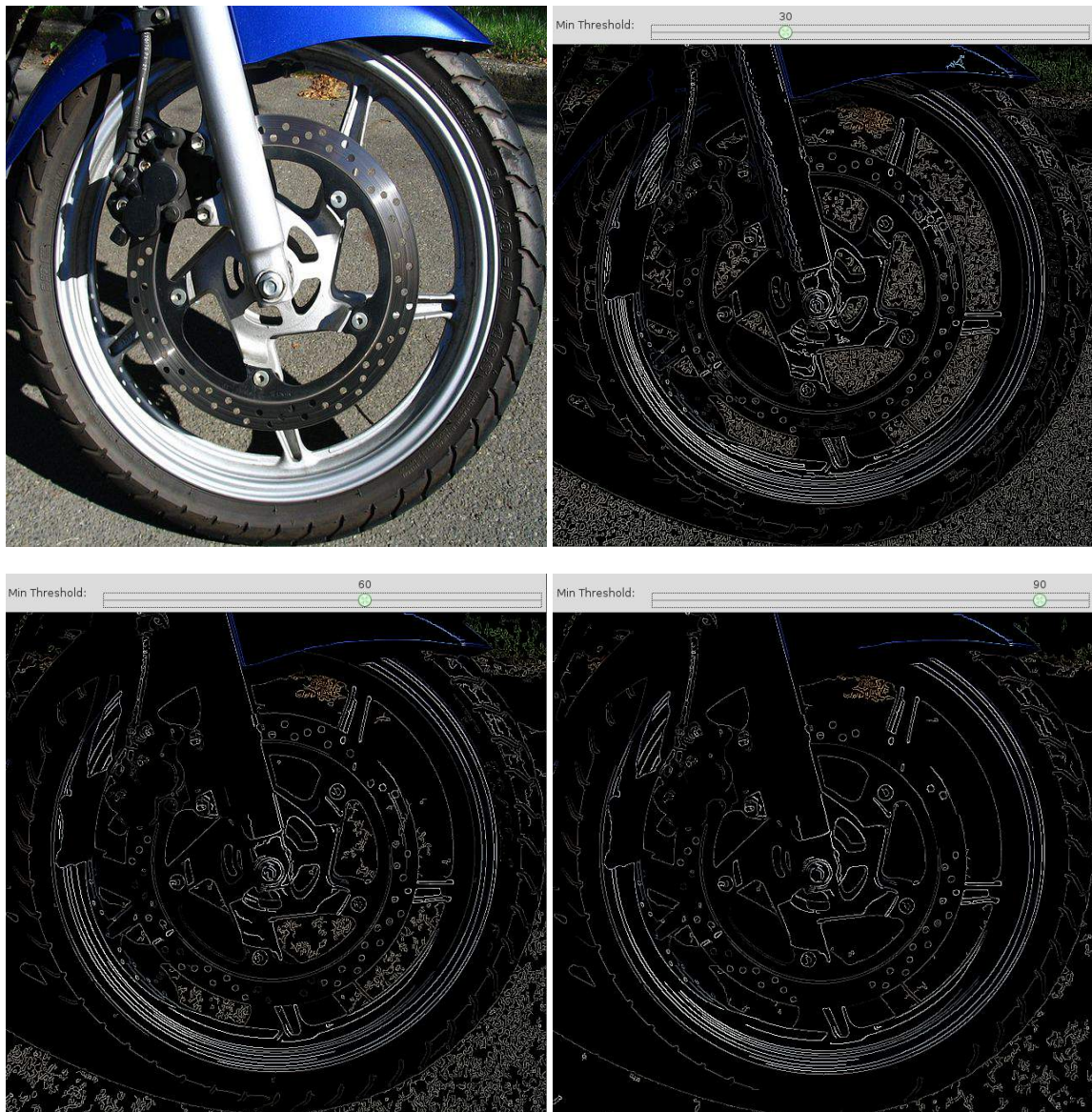


Figure 4.7: Original Image and its edge detection using OpenCV canny function, with a diferent minimum threshold.

Chapter 5

DICOM Image manipulation with GDCM.

Grassroots DICOM (GDCM) is a cross-platform library written in C++ for DICOM medical files, and it also supports RAW, JPEG (lossy/lossless), J2K, JPEG-LS, RLE (and deflated) extensions. GDCM libraries can be downloaded from http://sourceforge.net/projects/gdcm/#What_is_gdcmData_.3F, and for some Linux distributions, are found in the repositories. GDCM is composed of libraries of C++ , which basically are classes with specific functions [32].

With GDCM information contained in the tags can be exported, and the DICOM images can be manipulated. The example HelloVizWorld (code 5.1), shows how to read a DICOM image and write it again. It also exports some image data to the console, such as the number of dimensions, the value of each of these dimensions and the photometric interpretation of the image.

Code 5.1: HelloVizWorld example, taken from <http://gdcm.sourceforge.net/html/examples.html>

```
#include "gdcmImageReader.h"
#include "gdcmImageWriter.h"
#include "gdcmImage.h"
#include "gdcmPhotometricInterpretation.h"
#include <iostream>

int main(int argc, char *argv[])
{
    if( argc < 3 )
    {
        std::cerr << argv[0] << " _input.dcm_output.dcm" << std::endl;
        return 1;
    }
    const char *filename = argv[1];
    const char *outfilename = argv[2];
```

```

// Instanciate the image reader:
gdcM::ImageReader reader;
reader.SetFileName( filename );
if( !reader.Read() )
{
    std::cerr << "Could not read:" << filename << std::endl;
    return 1;
}
// The other output of gdcM::ImageReader is a gdcM::Image
const gdcM::Image &image = reader.GetImage();
// Let's get some property from the image:
unsigned int ndim = image.GetNumberOfDimensions();
// Dimensions of the image:
const unsigned int *dims = image.GetDimensions();
// Origin
const double *origin = image.GetOrigin();
const gdcM::PhotometricInterpretation
&pi = image.GetPhotometricInterpretation();
for(unsigned int i = 0; i < ndim; ++i)
{
    std::cout << "Dim(" << i << "):" << dims[i] << std::endl;
}
for(unsigned int i = 0; i < ndim; ++i)
{
    std::cout << "Origin(" << i << "):" << origin[i] << std::endl;
}
std::cout << "PhotometricInterpretation:" << pi << std::endl;

// Write the modified DataSet back to disk
gdcM::ImageWriter writer;
writer.SetImage( image );
writer.SetFileName( outfileName );
//writer.SetFile( file );

if( !writer.Write() )
{
    std::cerr << "Could not write:" << outfileName << std::endl;
    return 1;
}
return 0;
}

```

In this example, the classes ImageReader and Image are very important. The first is useful to load the DICOM file, and the second has functions that allow to read the contents of the DICOM file.

The first loop of the example program, is related to the syntax of the command that is executed once the program is compiled. It must have three arguments, the first is the name of the executable, the second the name of the input DICOM image and the third is the name of the output image. If the

order has the correct syntax, the program will continue, and if not, an error message will be showed. The second loop, first indicates that the input image is going to be read, and it then reads the image, showing an error message when the image could not be read.

Once the image is read, it is redirected to the class `Image` that is a container from where is possible to request information. The image container is passed as reference, and some of its characteristics (in the example, its number of dimensions, its dimension and the location of origin) can be requested. Also it is possible to request for the photometric interpretation, by means of the class `PhotometricInterpretation`. Finally, the class `ImageWriter`, takes the image container to write a new image, and it shows an error message when could not write. The code 5.2, shows the execution of the example `HelloVizWorld`, to read the file `test.dcm`, whose visualization is showed in the figure 5.1.

Code 5.2: Execution of the `HelloVizWorld` example

```
maverick@Helman:~$ ./HelloVizWorld test.dcm
./HelloVizWorld input.dcm output.dcm
maverick@Helman:~$ ./HelloVizWorld test.dcm test2.dcm
Dim(0): 512
Dim(1): 512
Origin(0): -434.66
Origin(1): -157.12
PhotometricInterpretation: MONOCHROME2
maverick@Helman:~$
```

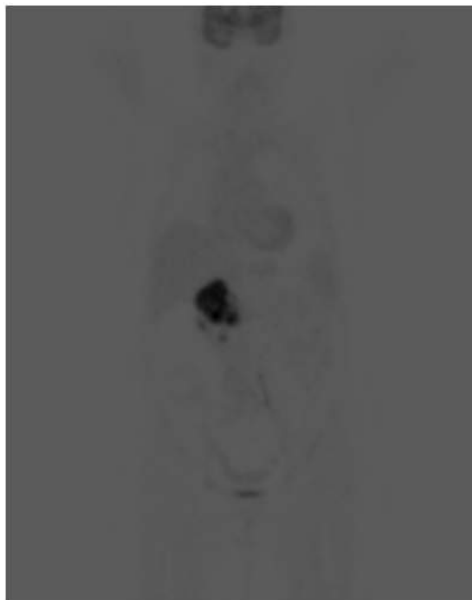


Figure 5.1: Test PET-CT DICOM file visualization in ImageJ with inverted LUT. Image taken from the PETCETIX sample image set, provided by the OsiriX DICOM Viewer.

With GDCM, also it is possible to acquire information about of the tags of DICOM file. This is achieved adding the lines of the code 5.3 into the code 5.1. Some tags can be modified by adding the code 5.4 taken from the example ClinicalTrialAnnotate (http://gdcm.sourceforge.net/html/ClinicalTrialAnnotate_8cxx-example.html) that works on the basis of the class Anonymizer.

Code 5.3: Tag exporting code

```
gdcm::File file = reader.GetFile();
gdcm::DataSet ds = file.GetDataSet();
std::cerr << ds << std::endl;
```

Code 5.4: Tag modify

```
gdcm::Anonymizer ano;
ano.SetFile( reader.GetFile() );
ano.RemoveGroupLength();
ano.RemovePrivateTags();

// PS 3.3 - 2008
// C.7.1.3 Clinical Trial Subject Module
ano.Replace( gdcm::Tag(0x12,0x10), "BigCompany_name" );
ano.Replace( gdcm::Tag(0x12,0x20), "My_Trial_Protocol_ID" );
ano.Replace( gdcm::Tag(0x12,0x21), "My_Trial_Protocol_Name" );
ano.Replace( gdcm::Tag(0x12,0x30), "My_Trial_Site_ID" );
ano.Replace( gdcm::Tag(0x12,0x31), "My_Trial_Site_Name" );
ano.Replace( gdcm::Tag(0x12,0x40), "My_Trial_Subject_ID" );
ano.Replace( gdcm::Tag(0x12,0x42), "My_Trial_Subject_Reading_ID" );
```

Chapter 6

Metodology.

In addition to developing a software to process diagnostic PET-CT or CT images, an experimental methodology was also designed, to process known DICOM images, and evaluate the results obtained in the image processing by means of the developed software. Next, both methodologies are described.

6.1 Software Development.

As reviewed in the previous chapter, GDCM (Grassroots DICOM) is a set of classes of C++ developed by the CERN [32], which allows to perform various processing options to DICOM images. Using `gdcmImageReader.h` [33] and `gdcmStringFilter.h` [34], two of the classes of GDCM, was built a source code, that first of all, read PET DICOM images, with CT attenuation correction, and then converts them into a metafile that can be interpreted by ROOT.

ROOT essentially comprises a set of classes of C++, but has other notable features: The user can interactively define classes, the interpreted code can invoke compiled code, and this again interpreted; additionally, you can process a huge amount of data, processing entries of the order of the terabytes [35]. Inside the developed source code, there were included subprograms that can read the metafile generated by GDCM and process it, by means of functions defined in the classes `TCanvas.h`, `TImage.h` and `TMath.h` of ROOT, and like a control, it exports a visualization of the DICOM image, like that appears in the figure 7.15 (left).

Once the DICOM image is read, it is redirected to the class `Image` that is a container from where is possible to request information. The image container is stored within a buffer, by means of the `GetBuffer` tool and then it is converted into an array, but unlike the processing of photographic images, whose value for each pixel is a hex number, in the case of images of PET-CT and CT, the value for each voxel, corresponds to a decimal number, so each number in the array can be read as a double number.

In the case of the PET-CT and CT images, the array where all the information of the image will remain stored, it is not obtained by means of the use of the `GetArgbArray()` tool, as it was done in the photographic images, now the vector is obtained using the tool `GetVecArray()`, that allows to obtain a vector of numbers decimal from the buffer where all the read DICOM file information is contained.

The obtained array can be converted into an arrangement 2D, as it was done in the code 4.3 that is not another thing than to obtain a A matrix, that is the base for the implementation of the image processing algorithms. Every element A_{ij} of from the matrix A , for the case of PET-CT, represents the counts for the boxel ij , while for the CT case, it represents the Hunsfield units for the boxel ij .

The matrix A_{ij} was defined by means of the TmatrixD class of ROOT. To make a visualization of the obtained matrix, within the program was defined a 2D Histogram, by means of the tool TH2D ROOT, which essentially represents the A_{ij} value for each boxel ij , as shown in Figure 7.15 (right). For DICOM images processing, algorithms were defined in the same way, as shown in the sections 4.1 and 4.2, and the results also were visualized by means of a TH2D histogram.

As a control, to visualize the array obtained from a reading of the DICOM image, a routine which exported vector to a grayscale image, was defined. as the DICOM image was becoming a photographic image in grayscale (figure 7.15 (left)). The routine was built by means of the creation of an image file from the array, using the class TImage, that converts the DICOM image into a photographic image in a grayscale, but to make this it is necessary to define a pseudo-color palette in a grayscale.

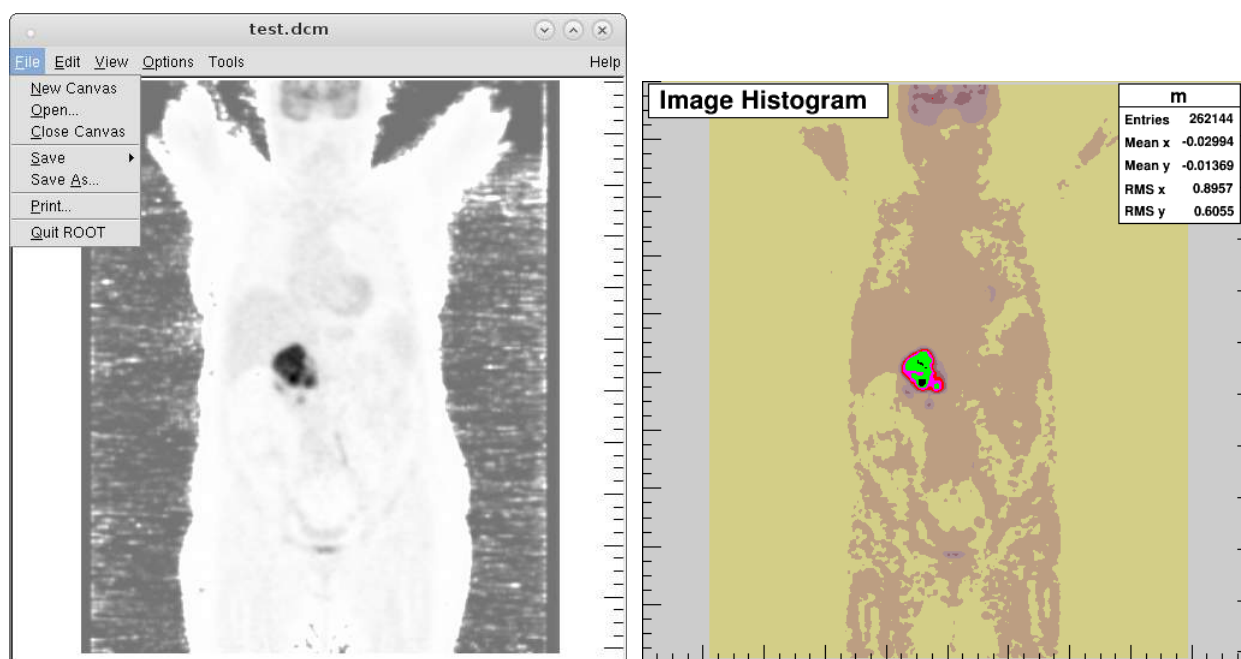


Figure 6.1: DICOM PET-CT Image exported to a ROOT canvas (left) and to a TH2D Histogram with Palette color 50 (right). Image taken from the PETCETIX sample image set, provided by the OsiriX DICOM Viewer.

6.2 Experimental Measurements.

Measurements were taken on a Siemens tomograph Biograph True Point PET-CT 64, which is a hybrid equipment that has a rotating X-ray tube for CT and a ring of detectors for PET. This equipment is located within the Department of Nuclear Medicine of the Instituto Nacional de cancerología (INC). PET images were obtained using ^{18}F -FDG as radiopharmaceutical and this is usually synthesized in a automated synthesis module Coupled to a cyclotron, which is located In the INC's Radiopharmacy.

For the image acquisition, a non-diagnostic CT is made, usually using 80 mAs and 90 KV, and then the PET acquisition scan is started, for what the computer performs the record of the number of true matches between each pair of anti-parallel detectors. The image acquisition was done in a total time of 5 to 15 minutes for a phantom. For the reconstruction PET and CT image reconstruction, the computer performs an iterative reconstruction method. The computer finally generated images in a DICOM format.

To obtain images of PET-CT is performed an experimental setup as that shown in the figure 6.2. The Jaszczak phantom was prepared by filling each cylinder with a solution of ^{18}F -FDG, whose activity concentration is higher than the concentration of activity that will have the inside of the phantom, this last can to simulate a radioactive background. The activity concentrations should be defined in accordance with the activity concentration that the tumors usually present for every patient, these values are highly variable, but are tabulated, depending on the patient injected dose [36].



Figure 6.2: Experimental design for the acquisition of PET-CT images from the phantom Jaszczak (central image). Taken from [37].

Once reconstructed, the images of PET-CT for the Phantom Jaszczak/ACRVOI are obtained as shown in figure [38]. The images shown in the figure were obtained by filling each cylinder with a ^{18}F -FDG solution of 6 mCi/ml, and the interior of the phantom with 1 mCi/mL solution of the same radiopharmaceutical [38]. The sequence of PET-CT and CT images from the Jaszczak Phantom was read with the developed software, to make the respective image processing.

On the other hand, as it will be shown, the phantom Jaszczak has several limitations, for that reason an Agar phantom was designed. For the Agar Phantom, PET-CT images will be acquired, performing the same procedure for positioning and image acquisition, which was just described for the phantom Jaszczak. The Agar phantom, was designed to simulate the biological behavior of the ^{18}F -FDG, more accurately than the phantom Jaszczak, and the images obtained from it, also allow to evaluate the results obtained in the processing of images using the software of delimitation of volumes.

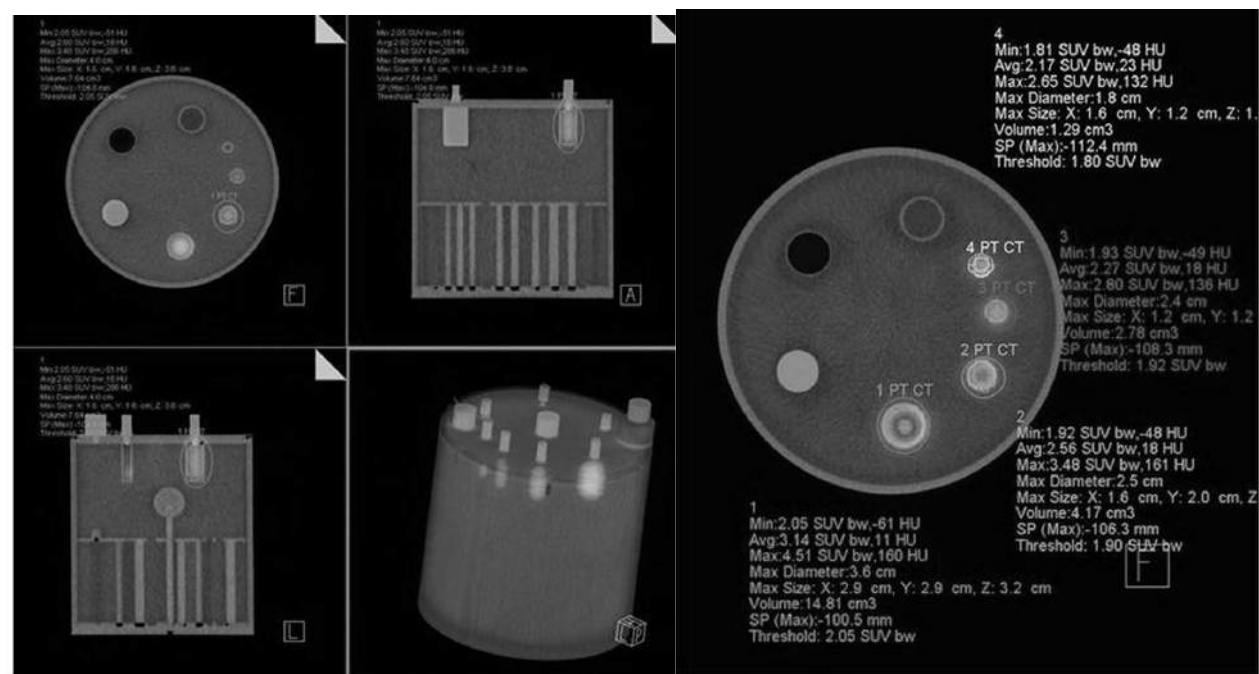


Figure 6.3: DICOM image of a Phantom Jaszczak/ACRVOI. Taken from [38].

6.3 Agar Phantom Design.

The design of the phantom of agar, is inspired by the method of Kirby-Bauer (KB) for sensitivity to antibiotics. This method is used to quantify the content of antibiotics present in a sample. The test consists of, first, preparing the Petri dishes, which have holes where the samples can be located, and second, measure the radius of inhibition of bacterial growth. For the preparation of the Petri dishes, agar with an enriched environment (which favors the growth of a bacterial culture) is dissolved in boiling water. The hot mixture is added to the disc Petri dish, which has a few molds in a cylindrical shape. After, a bacterial culture is added over the agar, so when the the Petri dish is cool, the agar will be in the form of gel, with cylindrical holes, where also there will be a bacterial growth[39].

The samples of antibiotic are introduced in cylindrical holes, and the antibiotic will spread in radial direction, creating a region around the cylindrical hole, where there will be no growth of bacteria. The antibiotic concentration throughout the region in which spreads, will decrease as a function of distance, up to the point where it no longer can inhibit the growth of bacteria; that way allow determine the minimum inhibitory concentration of the antibiotic, determining the maximum radius up to where it will be able to inhibit bacterial growth. In the figure 6.4 the results of a KB test are shown, there are a Petri dish, and in its interior, the clear zone, where the antibiotic from the sample inhibited bacterial growth [39].



Figure 6.4: Results of the Kirby Bauer Test. Taken from Wikipedia

Thinking in this principle, it is necessary to design a phantom in where the ^{18}F -FDG can submit diffusive phenomena. An agar gel was prepared in a Petri dish with a diameter of 150 mm, by placing a 250 mL beaker as mold, and a Phantom with a cylindrical hole 68 mm in diameter was obtained. For the elaboration of the phantom, were weighed 4 g of agar, then was it completed to 200 mL with distilled water in a 500 mL Erlenmeyer flask. Erlenmeyer Flask was left in an oven until boiling, and this dissolution after was introduced in the 15 mm diameter Petri dish, in that was introduced the 250 mL beaker. The Petri dish was cool, until is a cylindrical phantom with an intermediate hole. This hole was filled with ^{18}F -FDG, to simulate a cylindrical tumoral uptake of this radiopharmaceutical. The figure 6.5, shows a summary of the process of preparing the phantom, and the figure 6.6 shows the prepared phantom, and its positioning in a PET-CT equipment.

Although the phantom Jaszack, contains several cylinders of fixed volumes, which can be filled with ^{18}F -FDG, to simulate a uptake of the radiopharmaceutical, several aspects must to be taken into consideration:

1. The cylinders of the phantom Jackzack contain a known volume of dissolution of ^{18}F -FDG in Normal Saline Solucion. However, in a biological medium, the ^{18}F -FDG, will not be dissolved in a liquid, but that will be dispersed in a medium, which viewed in a macroscopic point of view, is a colloidal system.
2. The cylinders of the phantom Jackzack have rigid and impermeable walls, but in a biological medium the ^{18}F -FDG is spreading in a volume that don't have fixed borders.
3. It is necessary to evaluate the delimitation of volumes in different geometries, in particular in geometries that do not have a regular shape, since tumors generally are amorphous. However, the phantom Jackzack only have compartments with cylindrical shape.
4. The concentration of ^{18}F -FDG is homogeneous in the compartments of the phantom Jackzack, and this is very different to what happens in a biological medium, where the concentration of radiopharmaceutical is not homogeneous in a tumor.

Bearing in mind the previous considerations, an Agar phantom will be designed, since this one material is of colloidal type, for what the ^{18}F -FDG will spread in a colloidal way, and the volume occupied by the radiopharmaceutical is not going to have fixed borders. With the phantom of Agar, radioactive volumes with diverse volumes and forms, and non homogeneous concentration can be simulated. This radioactive volumes can be delimited with the software that was developed. The phantom of Agar can be used to simulate biodistribution of radiopharmaceuticals, and for future studies, the Agar phantom's composition can be changed, in order to make more complex biodistribution studies.



Figure 6.5: Process of elaboration of a prototype agar phantom.

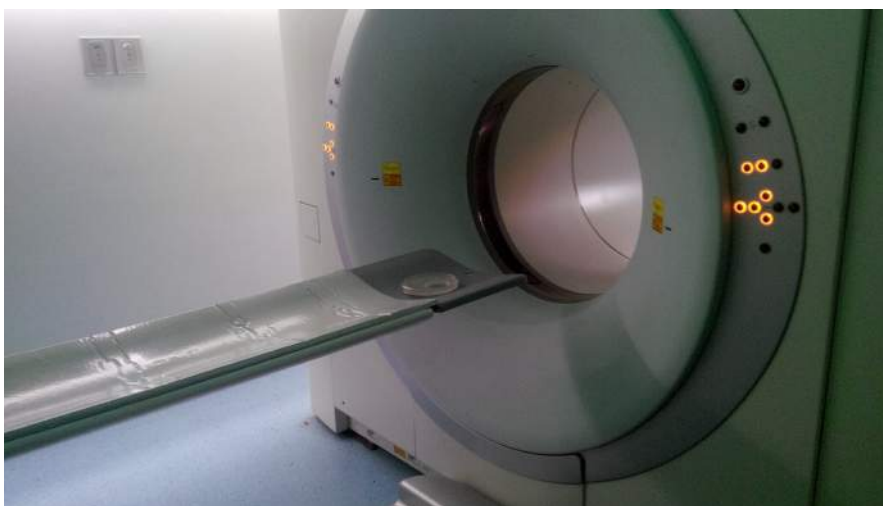


Figure 6.6: Agar phantom, and its positioning in the INC's PET-CT equipment.

For the filling of the Agar phantom, 620 ml of a ^{18}F -FDG water dissolution with an activity concentration of $1,54 \mu\text{Ci}/\text{mL}$ were prepared, and then 6 mL were added to the Agar phantom. An image acquisition was realized an hour later, with a duration of 15 minutes. Also were added 4 dots over the agar surface, whose total volume were 2 mL.

Chapter 7

Results and Analysis.

The results obtained in the phantoms image processing are showed, and then are showed the results obtained when diagnostic images are processed.

7.1 Phantom Images.

On having realized the PET-CT phantom images, the results showed in the figures 7.1 and 7.2, were obtained for the Jaszczak phantom and for the Agar phantom, in addition, the gradient of counts histograms were also plotted.

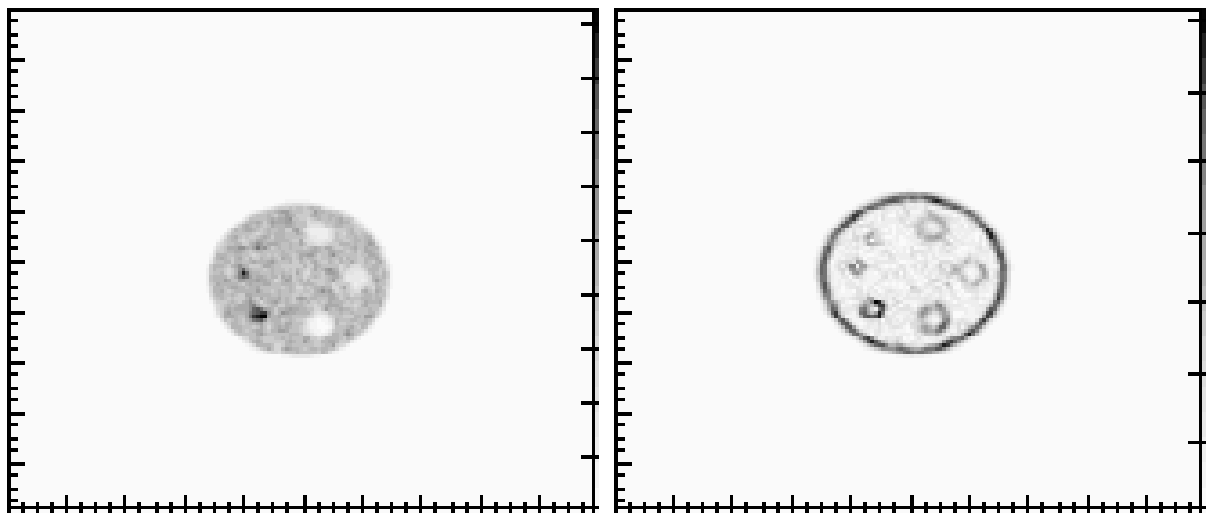


Figure 7.1: Jaszczak Phantom PET-CT Image histogram (left), and PET-CT gradient magnitude histogram (right).

In the gradient magnitude images the edges are enhanced, but that edges are so thick, both for the phantom Jaszczak, as for the Agar phantom. This is because the difference between the activity concentrations outside the hot zone and inside the same, is very large. Regarding the Jaszczak phantom,

images were obtained in a transversal plane, while the Agar phantom images were obtained in a coronal plane. The gradient of counts histogram of the Jaszczak phantom (figure 7.1, right) shows the hot spots edges, but that edges do not correspond to a circular contour, so this images will not be used in subsequent analysis. In contrast, the gradient of counts histogram for the Agar phantom (figure 7.2, right), shows a well-defined circular contour.

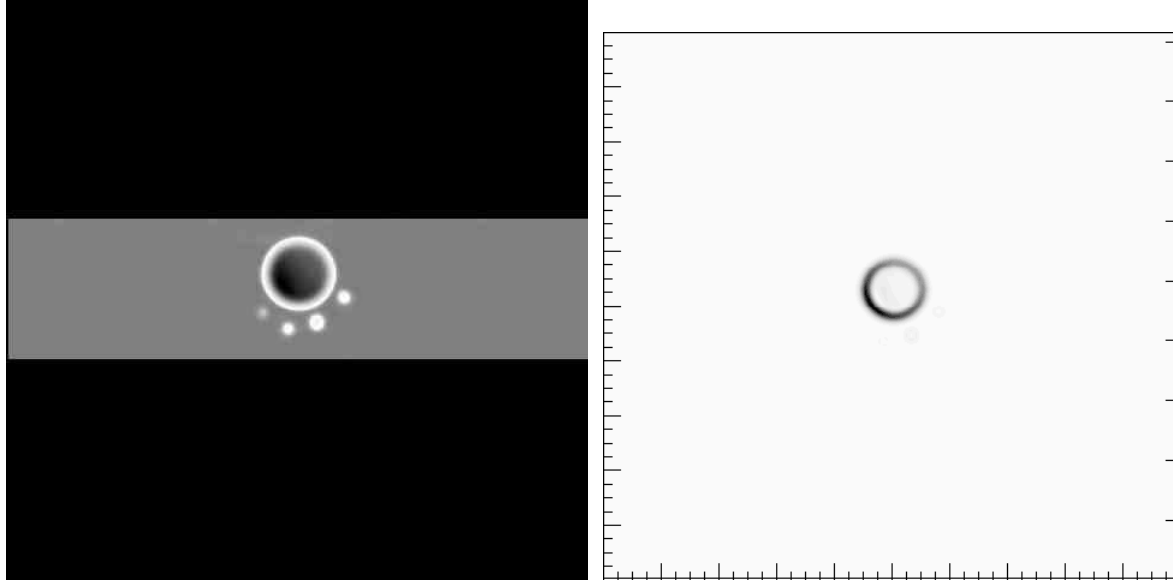


Figure 7.2: Agar Phantom PET-CT Image (left), and PET-CT gradient magnitude histogram (right).

To analyze the delimitation made with the SUV_{max} thresholding, the diameter of the disc remaining was calculated. This calculation was done, making a software routine that counts the number N of pixels exceeding the threshold percent value. Because the area for each pixel is 1 pixel^2 , the total area of the disk (A_d) will be:

$$A_d(\text{mm}^2) = N \text{ pixels}^2 \cdot \frac{468184.38 \text{ mm}^2}{234256 \text{ pixels}^2} \quad (7.1)$$

Where the ratio $468184.38 \text{ mm}^2 / 234256 \text{ pixels}^2$ is calculated with the DICOM file information about the image and object size. The area of the disk is $A_d = \pi D^2 / 4$, so the calculated diameter of the disc is $D_c = 2 \cdot \sqrt{\frac{A_d}{\pi}}$. Plotting D_c as a function of the threshold percentage (Thr), the figure 7.3 is obtained.

As shown in the figure 7.3, the calculated diameters generally are different from the measured value (69 mm) represented in green. From $Thr = 10\%$, D_c decreases approximately linearly with respect to the Thr . From $Thr = 60\%$, the curve presents an inflection and the D_c begins to decrease more quickly. For a better visualization, the percentage difference P_d was calculated with eq 7.2, and plotted as a function of Thr as shown in the figure 7.4.

$$P_d = \left| \frac{D_c - 69}{69} \right| \cdot 100 \quad (7.2)$$

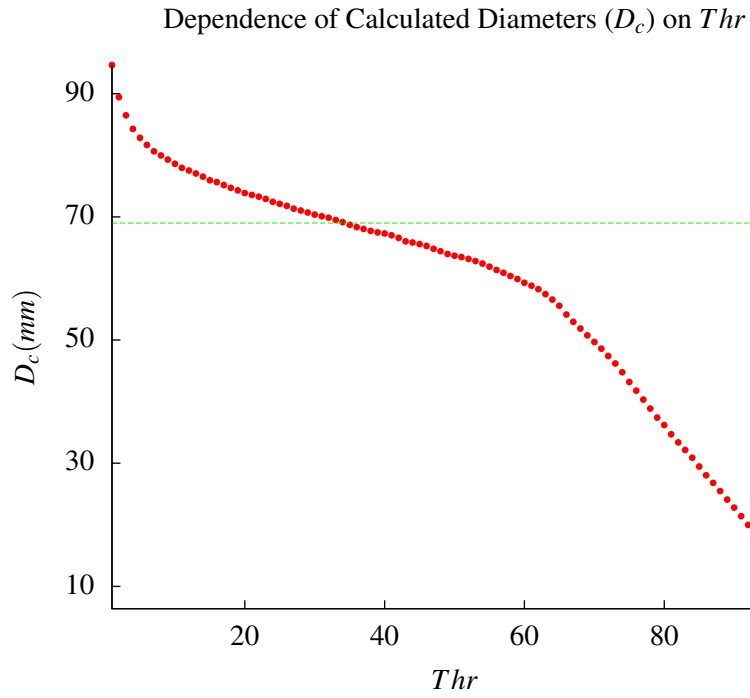


Figure 7.3: Calculated diameter as a function of the the threshold percentage (Thr)

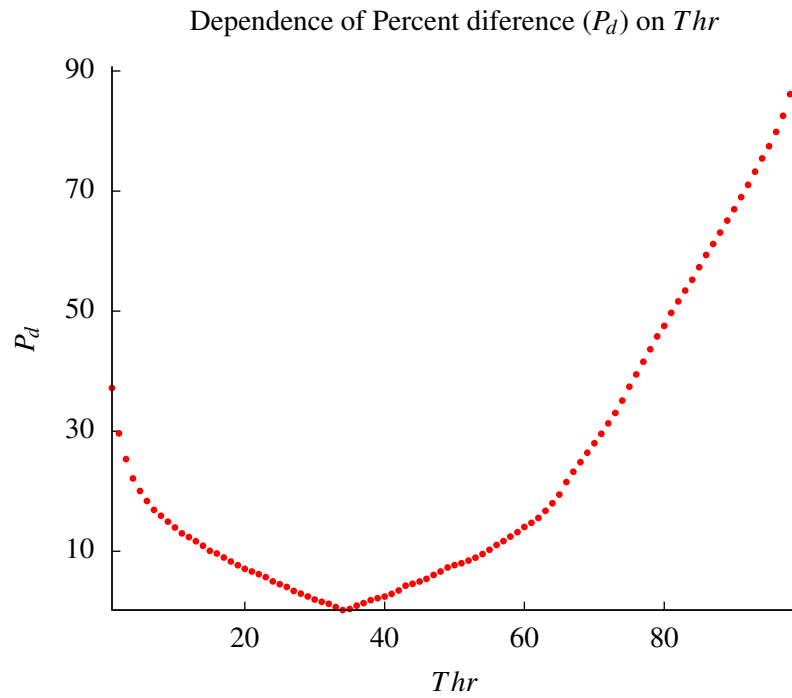


Figure 7.4: Percentage difference P_d as a function of Thr

In the figure 7.4, before the $Thr = 15\%$, the P_d is greater than 10%, since there it remains lower than 10%, but when $Thr = 55\%$ it starts to increase, and presents a inflection at $Thr \approx 65\%$, and its increase continues. In the computational algorithm used to calculate the diameters, all pixels in the image were considered. For this reason, when the Thr is lower than 10%, the additional drops that were added to the phantom, interfere with the calculation. From $Thr \approx 10\%$, the drops no longer interfere and the diameter can be calculated accurately, until the $Thr = 55\%$. As it is possible to see in the figure 7.5, from $Thr \approx 60\%$, the delimited image is no longer a disc, so the diameter values begin to deviate significantly from the expected value.

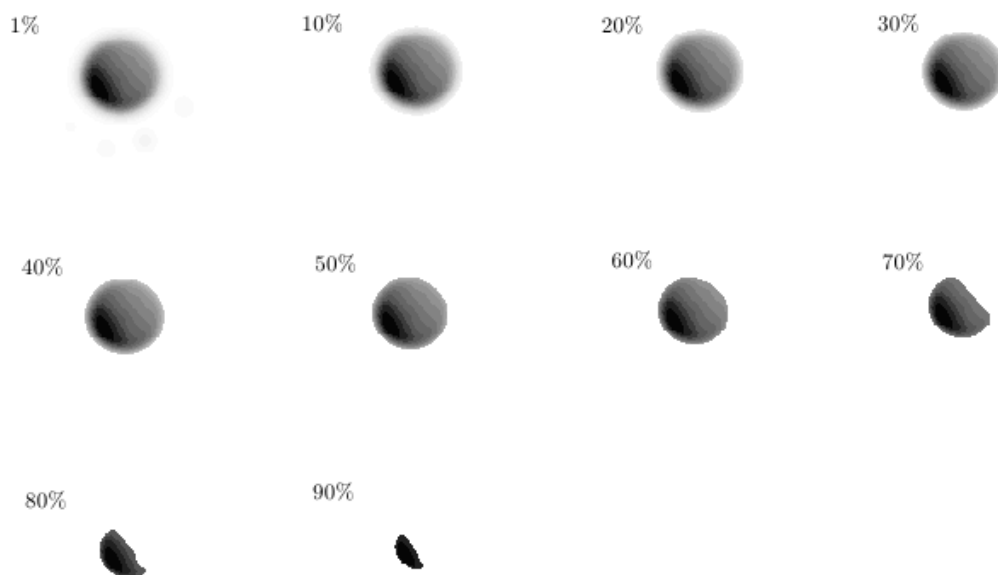


Figure 7.5: Agar Phantom Thresholding images for some Thr .

Now, to analyze the delimitation made with the gradient magnitudes, the diameters of the remaining ring were calculated (Appendix 2) and plotted as a function of Thr , as shown in the figure 7.6. In the figure 7.6, the mean diameters generally are very close to the measured value (69 mm) represented in color magenta. From $Thr \approx 55$, all the diameters decreases with respect to Thr . External and inner diameters have almost the same behavior that the mean diameter, and as shown in the figure 7.6, their values are reliable when $Thr < 15$ and $Thr > 55$. In the computational algorithm used to calculate all the diameters, all pixels in the image were considered. As it is possible to see in the figure 7.8, from $Thr \approx 60\%$, the delimited image is no longer a ring, so the diameter values begin to decrease quickly, and deviate significantly from the expected value.

These results show that the mean diameter allows a better definition than the other two diameters, however, for a better visualization, its percentage difference P_d was calculated with eq 7.2, and plotted as a function of Thr (figure 7.7).

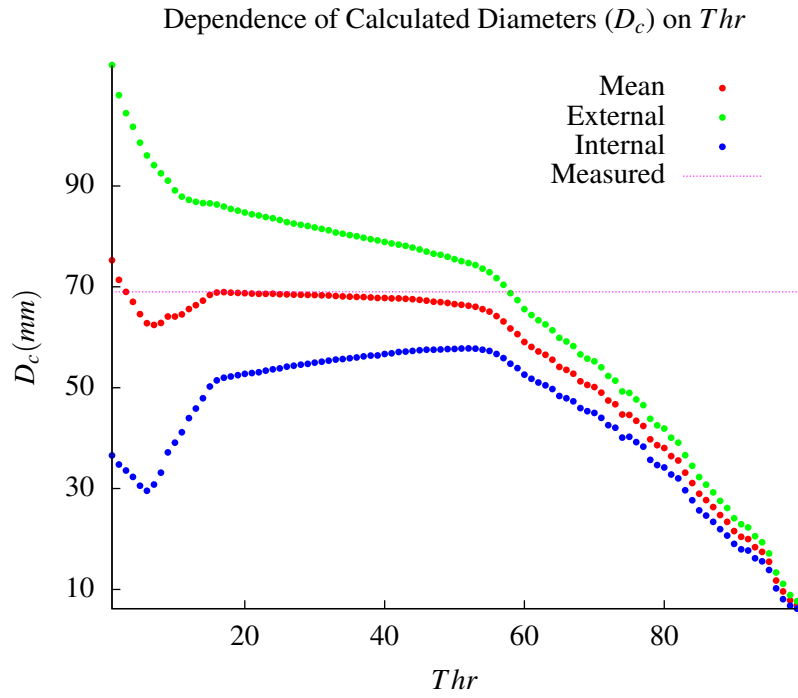


Figure 7.6: Calculated diameters for the remaining ring as a function of Thr

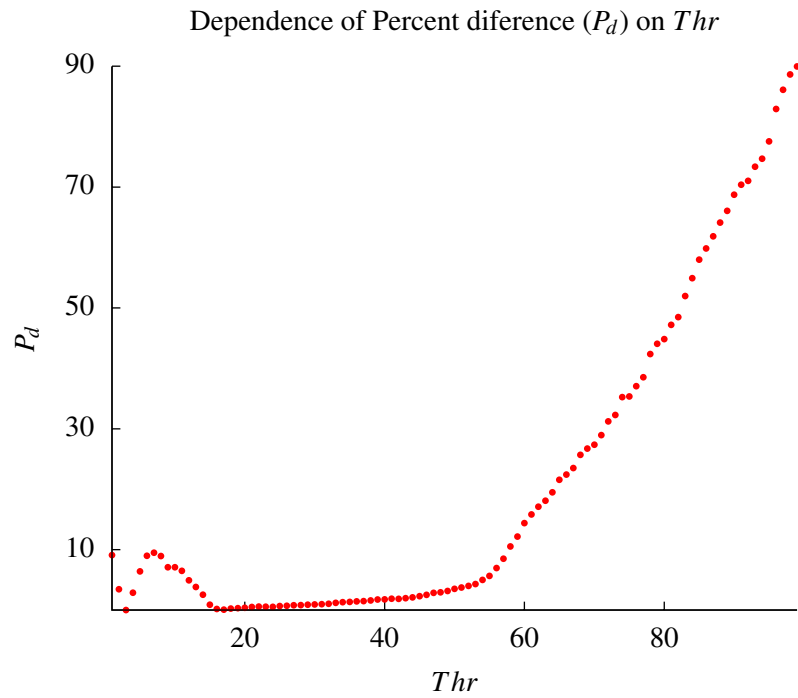


Figure 7.7: Percentage difference P_d for the mean diameter as a function of Thr

For the mean diameters, all the P_d values are lower than 10 % when the Thr is lower than 55 (even in the presence of the additional points), as can be seen in the figure 7.7. For the mean diameter, the P_d values for $15 \% \leq Thr \leq 50 \%$ are smaller than 5 %. In general, better results are obtained when delimitation is done using counts gradient thresholds, than the counts thresholding, because the calculated average diameters are more reliable, and the outer diameters provide additional information that should be taken into account.

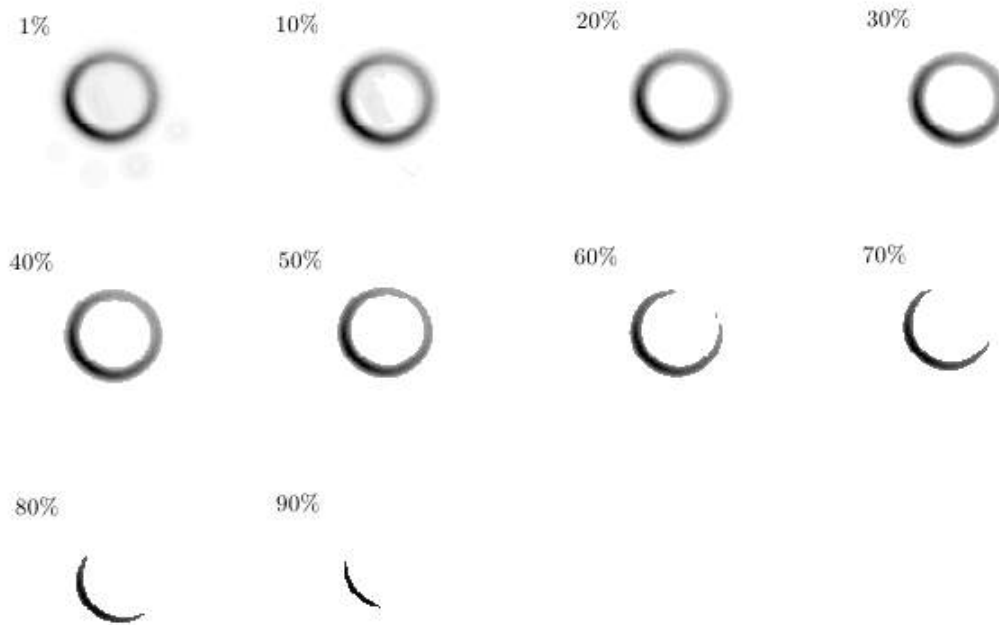


Figure 7.8: Agar Phantom gradient Thresholding.

The gradient of counts obtained for PET-CT, has a physical interpretation. As demonstrated in the section 2.3.1, the gradient of counts is proportional to mass concentration:

$$n(i, j) = a(t) \cdot c(i, j) \quad (7.3)$$

If the gradient operator is applied:

$$\nabla n(i, j) = a(t) \cdot \nabla c(i, j) \quad (7.4)$$

Remembering the Fick's first law [40], $\mathbf{j} = -D \nabla c$ (with c the mass concentration), the gradient of the number of counts, can be seen as:

$$\nabla n(i, j) = -\frac{a(t)}{D_{ij}} \cdot \mathbf{j}(i, j) \quad (7.5)$$

According to the equation 7.4, the rings (figure 7.8) indicates a different gradient of mass concentration for the ^{18}F -FDG around the wall of the hole in the Agar Phantom. According to this, the results obtained for the Agar Phantom, show that average diameter ring defines very well the hole wall, and both R_e and R_i define a region where the ^{18}F -FDG mass concentration changes, because the wall of the hole in the Agar phantom, in this case acts as membrane.

The results obtained with the gradient thresholding in the Agar Phantom, are useful to understand that the BTV, is a region where the ^{18}F -FDG present an hyperuptake and this region also includes a external region where the radiopharmaceutical simply presents diffusion, so the BTV includes an additional margin that only represents the ^{18}F -FDG diffusion from the hyper-uptake region, towards the tissues that are neighbors to the region of interest and this margin not necessarily represents an external hyper-uptake region.

The edges in the image of the phantoms can be detected using the Canny method, and this was done by means of OpenCV, similarly as shown in the section 4.2.2. The result image is showed in the figures 7.9 and 7.10.

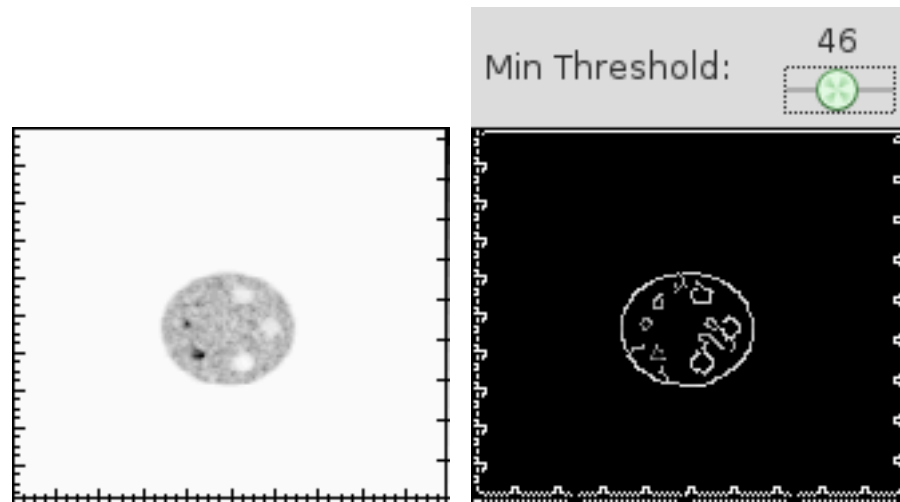


Figure 7.9: Jaszczak Phantom PET-CT Image histogram (left), and its Canny Edge Detection (right).

In the Canny edge detection for the Jaszczak Phantom (figure 7.9), not circular edges were obtained, as expected, while the Canny Edge detection for the Agar Phantom (figure 7.10) shows two circles, whose diameters were measured directly from the image and averaged, obtaining the values $66.2 \pm 0.9 \text{ mm}$ and $89.5 \pm 1.0 \text{ mm}$. According to these values, the first corresponds to the edge of the hole, while the second corresponds with the end of the diffusive region.

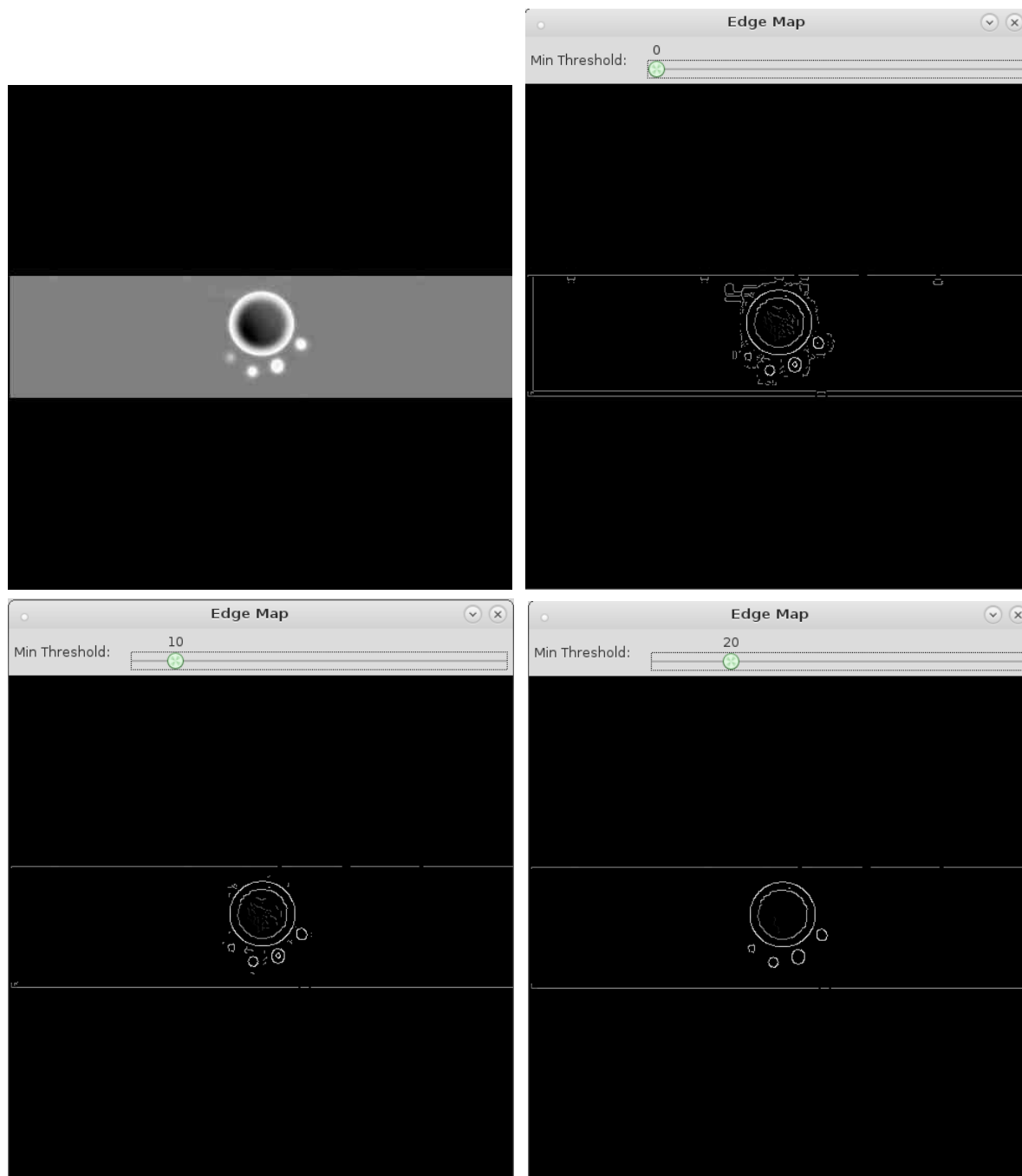


Figure 7.10: Agar Phantom PET-CT Image (left), and its Canny edge detection (right) using 0, 10 and 20 percent as the min threshold.

Both diameters keep constant, even if the min threshold varies, in addition, the regions formed by the additional drops are well delimited. The other two straight lines, corresponds with the region selected previously in the PET-CT equipment before image acquisition start. The both lines were discarded in the images 7.2 (right) and 7.8. According to the above results, the best delimitation, is done when the Canny edge detector is used. The diameter calculated by means of this method, fits with high accuracy to the measured diameter and does not depend on the min threshold.

On the other hand, when a Gaussian blur and then a Laplacian are applied for the image 7.2, figure 7.11 is got, where there are, both positive values such as negative, unlike the gradient magnitude histogram where only positive values are got.

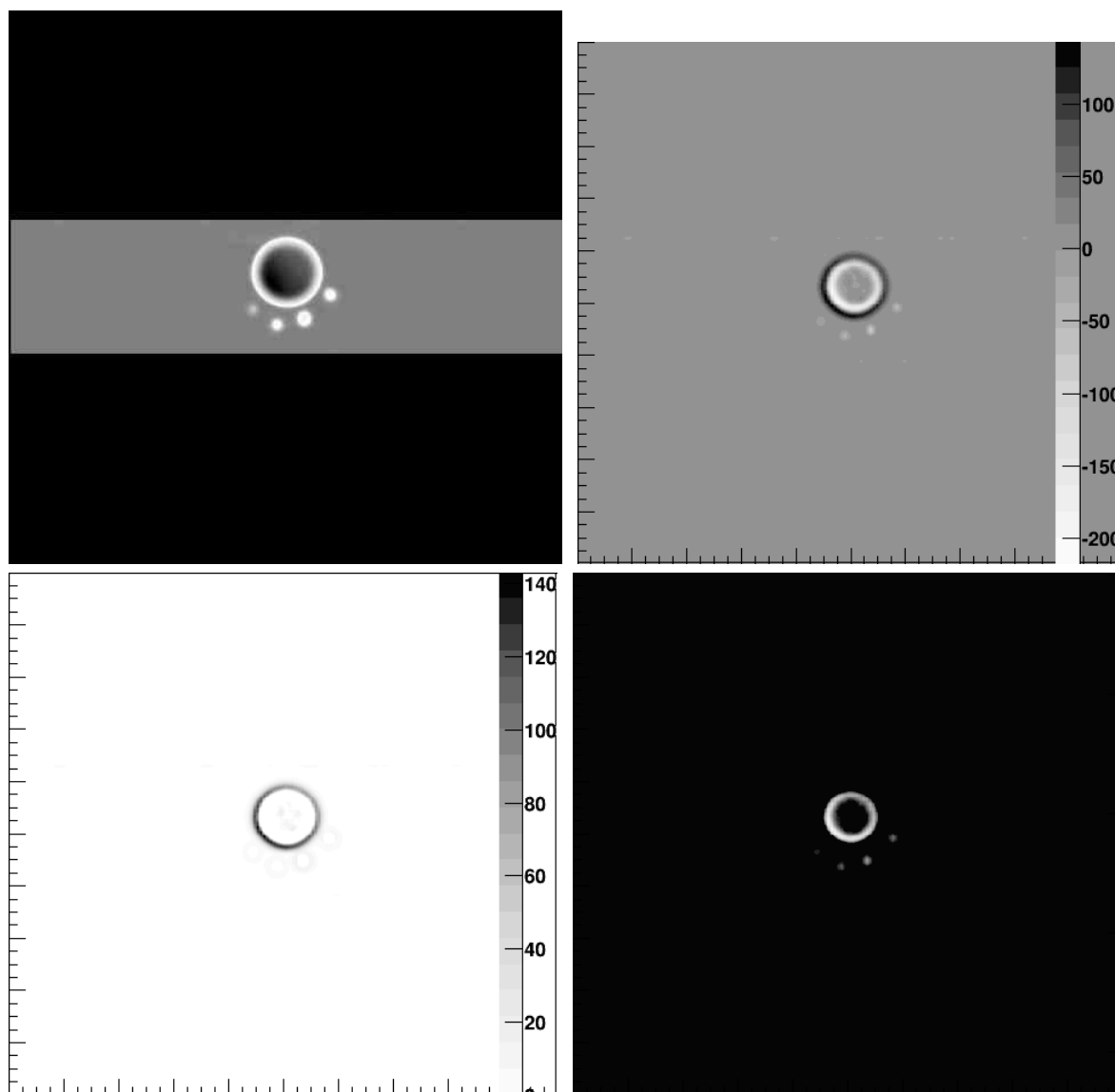


Figure 7.11: Original Agar Phantom PET-CT Image (upper left), and its Laplacian after a Gaussian blur (upper right), Laplacian positive values (bottom left) and Laplacian negative values (bottom right).

In the image 7.5 (upper right), two rings can be seen, and in accordance with the images 7.5 (bottom left) and 7.5 (bottom right), the outer ring corresponds with the positive values of the Laplacian of counts, and the internal corresponds with the negative values of the Laplacian. The additional points in the four pictures above can also be observed. In the histogram of positive laplacians, the internal, promedio and external diameters were calculated (Apendix 2) and the results are shown in the figure 7.12.

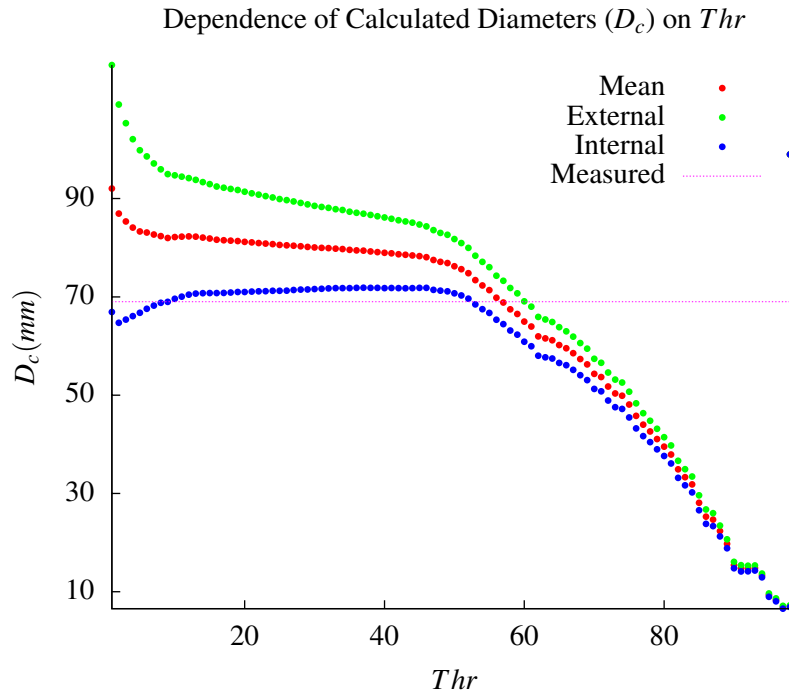


Figure 7.12: Calculated diameter (with Laplacians) as a function of the Thr

These results show that the internal diameter allows a better delimitation than the other two diameters, and the difference P_d was calculated with eq 7.2, and plotted as a function of Thr (figure 7.13). On the other hand, when the Laplacian is applied to equation 7.3:

$$\nabla^2 n(i, j) = a(t) \cdot \nabla^2 c(i, j) \quad (7.6)$$

In accordance with the Fick's second law [40], $\frac{\partial c}{\partial t} = D \nabla^2 c$, and when this is replaced in the previous equation:

$$\frac{\partial c}{\partial t} = \underbrace{\frac{D}{a(t)}}_{B(t)} \nabla^2 n(i, j) = B(t) \nabla^2 n(i, j) \quad (7.7)$$

According to the Fick's second law, in the points where positive Laplacians are obtained, the mass concentration is increasing in time. When the Laplacians value is negative, the opposite occurs. In the 7.5 (bottom left) image, the black ring (positive values of the Laplacian) shows those points where the concentration is increasing, while the image 7.5 (bottom right) shows the white ring (negative values of the Laplacian), that represents those points where the concentration is decreasing. In the figure 7.5 (upper right), the region between the white ring, and the black ring is the region where the Laplacian is zero, i.e., where the concentration changes from negative, to positive, and is there where the edge of the hole is located in the Agar.

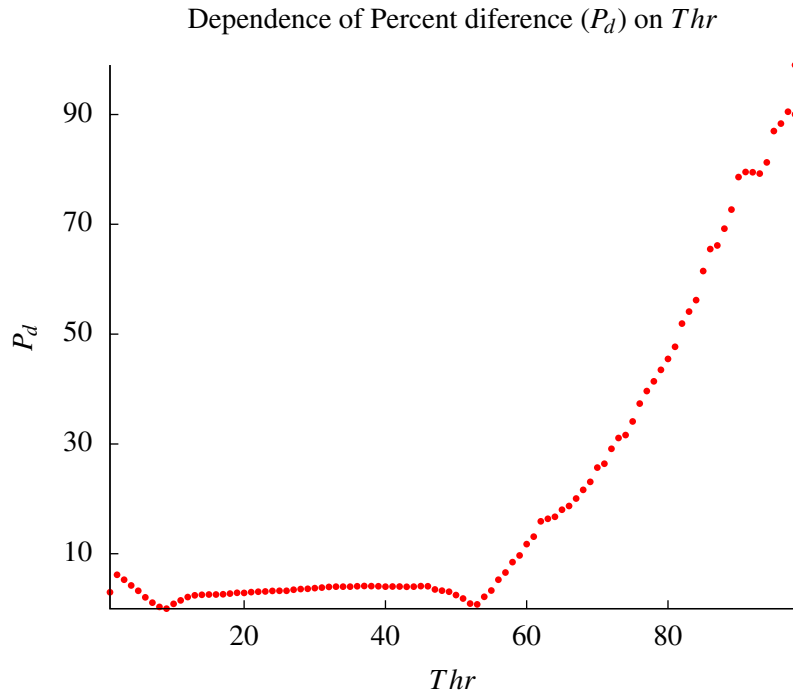


Figure 7.13: Percentage difference P_d for the internal diameter as a function of Thr

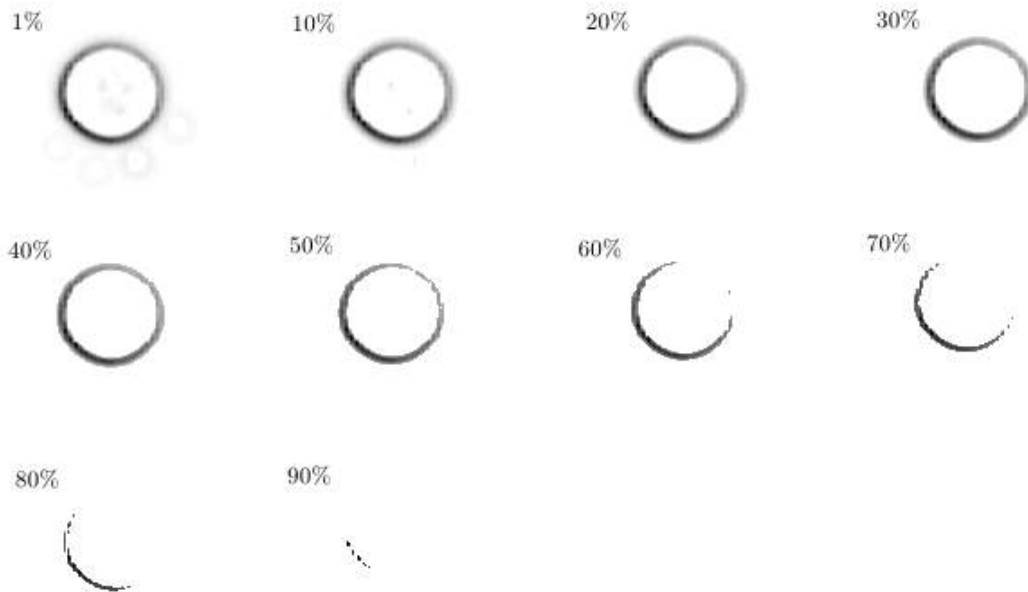


Figure 7.14: Agar Phantom Laplacian Thresholding.

Those pixels where the Laplacian begins to be positive, show where the Laplacian changes its sign, and that is why, in the image 7.5 (bottom left) image, where only positive Laplacians were considered, the internal diameter is very close to the measured directly diameter as can be seen in the figure 7.13 where the P_d is around 5 percent, when Thr lower than 53 is used. When Thr upper than 50 percent, the rings began to deform (figure 7.14), as happened with the gradient magnitude threshold images.

In the images obtained for the Agar Phantom, the radioactive material was not distributed in a uniform way as can be seen in the figure 7.5, since in the image the counts are higher in the lower left region. This is because the Phantom was not positioned in a completely horizontal position, so the radioactive material was occupying a volume in the shape of cylindrical wedge, with a calculated 3 mm of height. As the volume occupied by the radioactive material in the Agar Phantom, has a very small thickness, laplacians and gradients in two dimensions can be used, and them have a physical interpretation. For larger thicknesses, the whole image sequence should be considered for the calculation of gradients and Laplacians. In the next section, the results obtained in diagnostic images processing are shown.

7.2 Diagnostic Images.

Figure 7.15 shows a PET-CT image of a patient with abdominal lymphoma after reading the DICOM data with the developed software. On the left is displayed the grayscale control image where the darkest areas correspond to the areas of the major uptake of ^{18}F -FDG and the right image is a histogram where the color scale represents the counts (level of ^{18}F -FDG uptake).

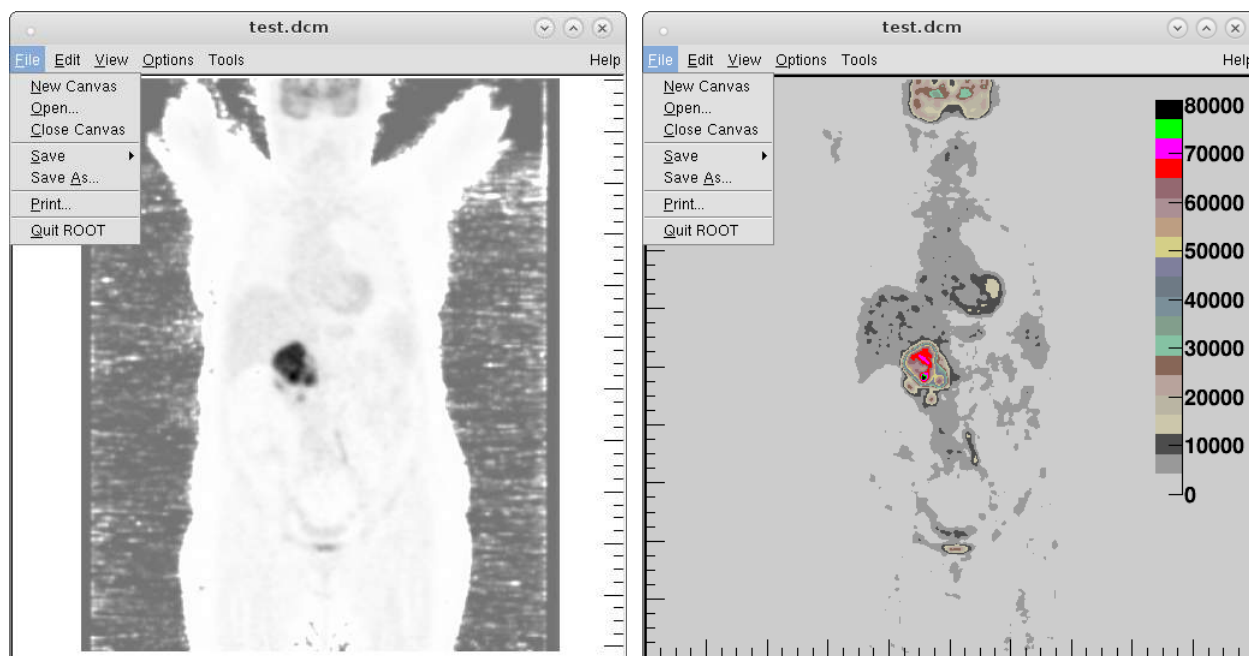


Figure 7.15: DICOM PET-CT Image exported to ROOT canvas (left) and exported to a TH2D Histogram with Palette color 50 (right). Image taken from the PETCETIX sample image set, provided by the OsiriX DICOM Viewer.

Figures 7.16 and 7.17 show a common PET-CT and CT images, together with the representation of the same image in a ROOT TH2D grayscale histogram (right). And in these two figures can be seen that the DICOM test images, were read correctly, although the CT image is much more clear, and that is because the grayscale color palette of colors defined in the software, is in an inverted grayscale, which is useful for PET-CT, but not in CT, so the palette color must be inverted for the visualization of the CT DICOM images.

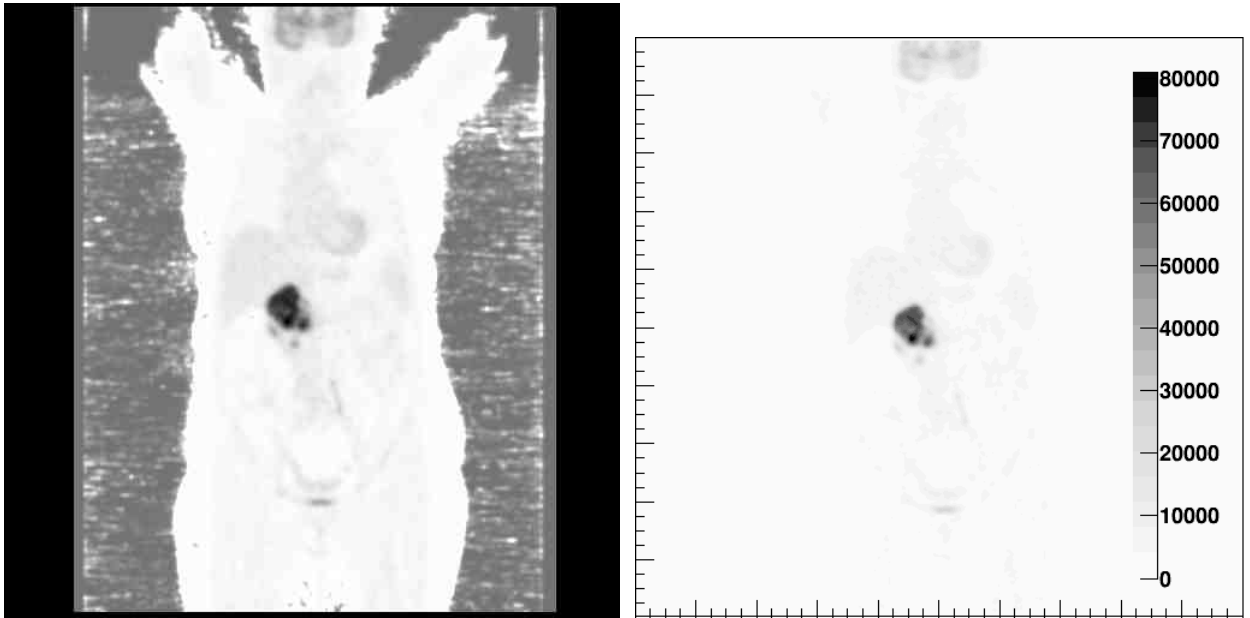


Figure 7.16: DICOM PET-CT Image (left) and exported to a ROOT grayscale TH2D histogram (right).

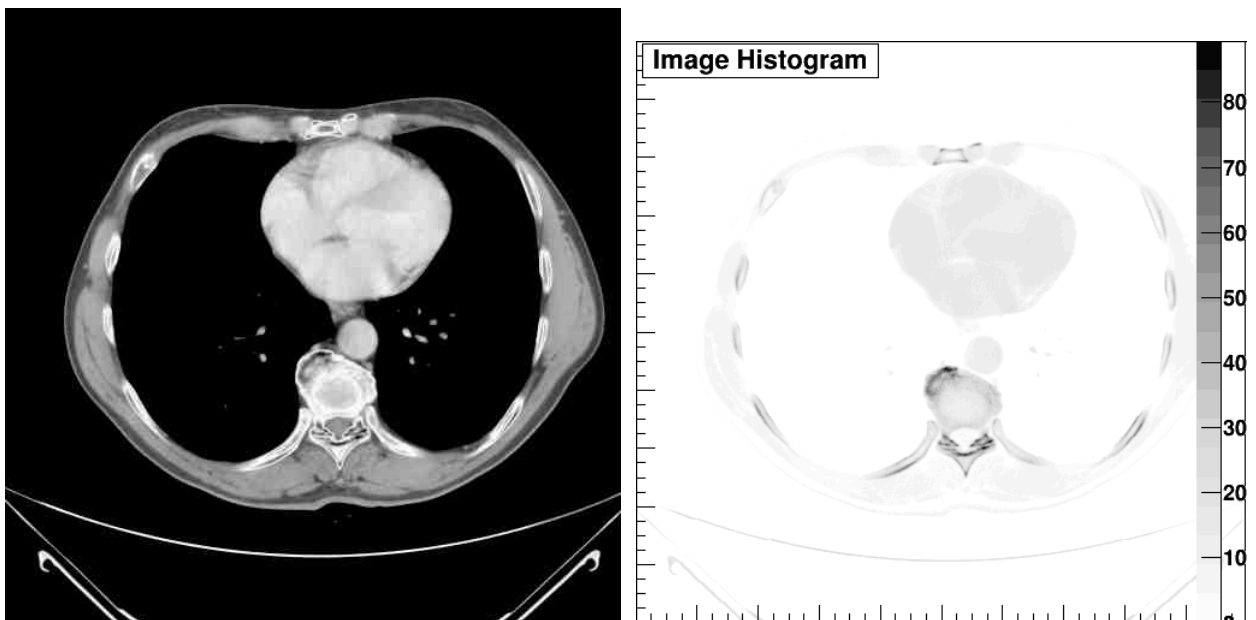


Figure 7.17: DICOM CT Image (left) and exported to a ROOT grayscale TH2D histogram (right). Image taken from the DICOM example of Geant4.

By applying convolutions into the PET-CT DICOM images, the images shown in the figures 7.18 and 7.19 were obtained. Is possible to see there, the changes in the shape of the original image, showing in each case the desired effect, either the blur, the sharpen, or the enhanced edges, and this results indicates that the image processing was successful in each case. Something similar can be seen when convolutions are applied in the CT images (figures 7.20 and 7.21).

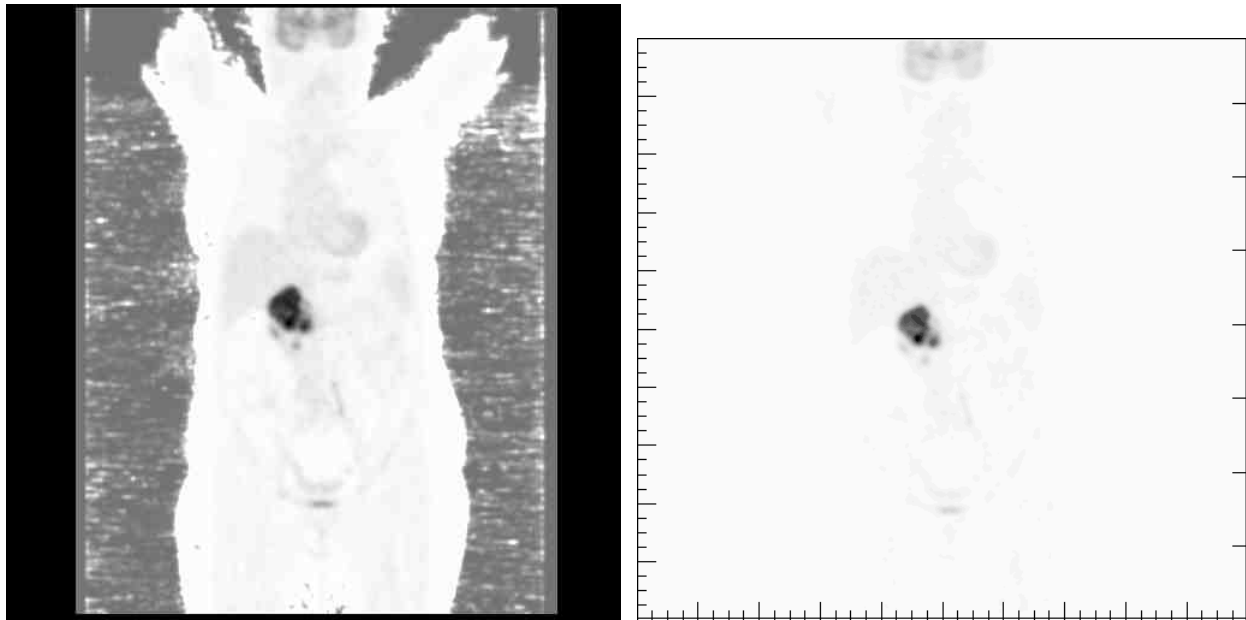


Figure 7.18: PET-CT Control image (left) and Gaussian Blur convolution (right).

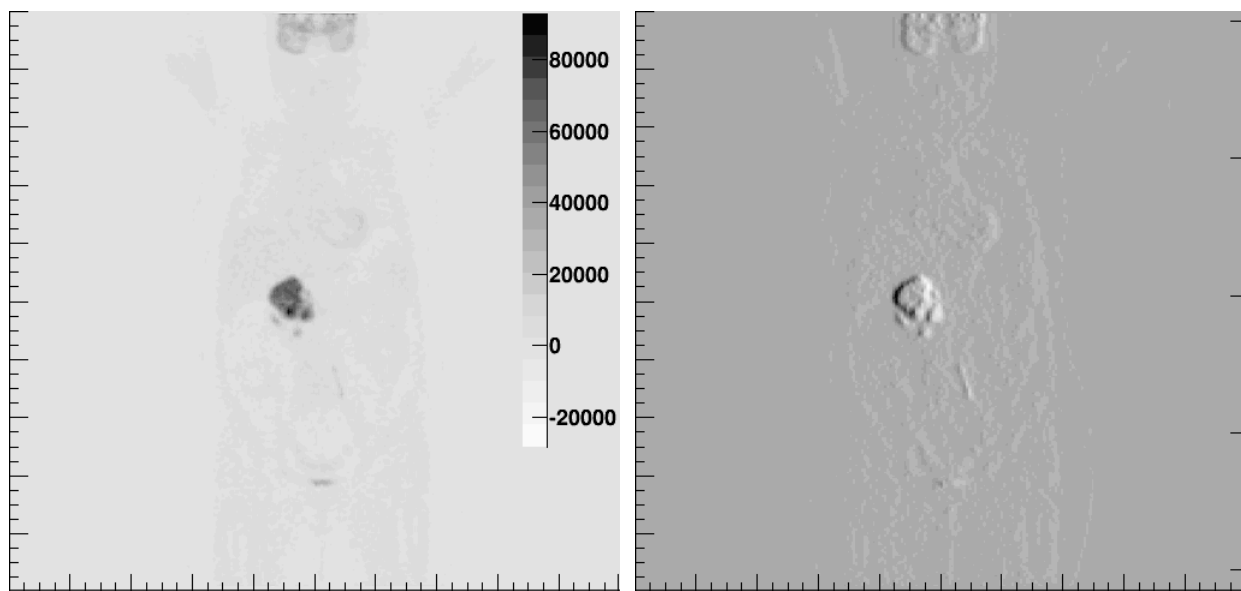


Figure 7.19: PET-CT Sharpening convolution (left) and edge enhancing convolution (right).

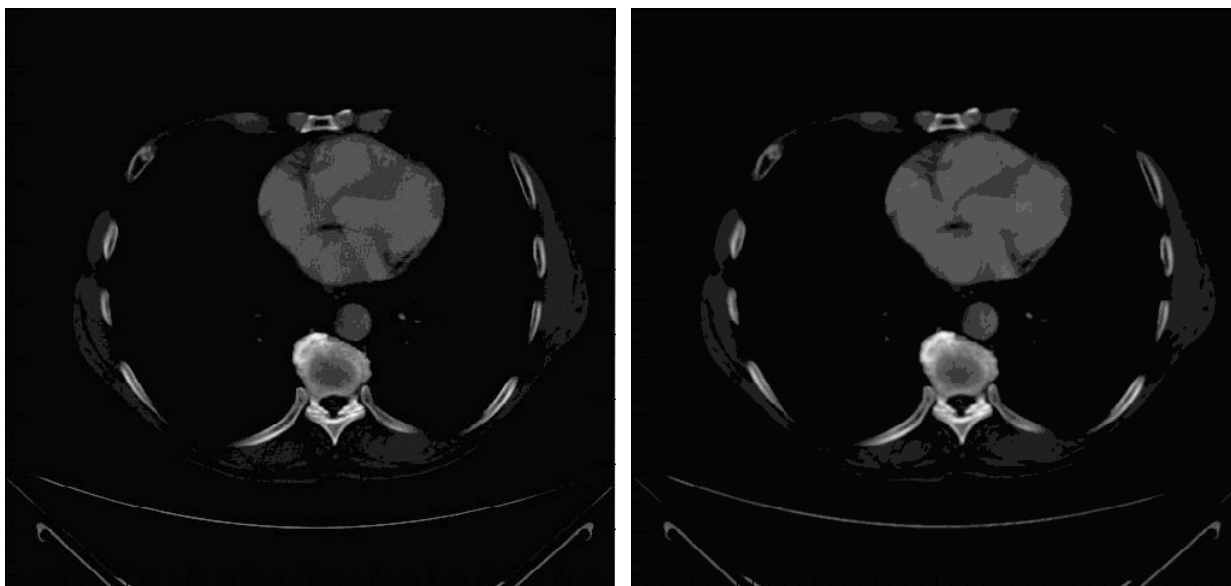


Figure 7.20: CT Image Histogram (left) and Gaussian Blur convolution (right).

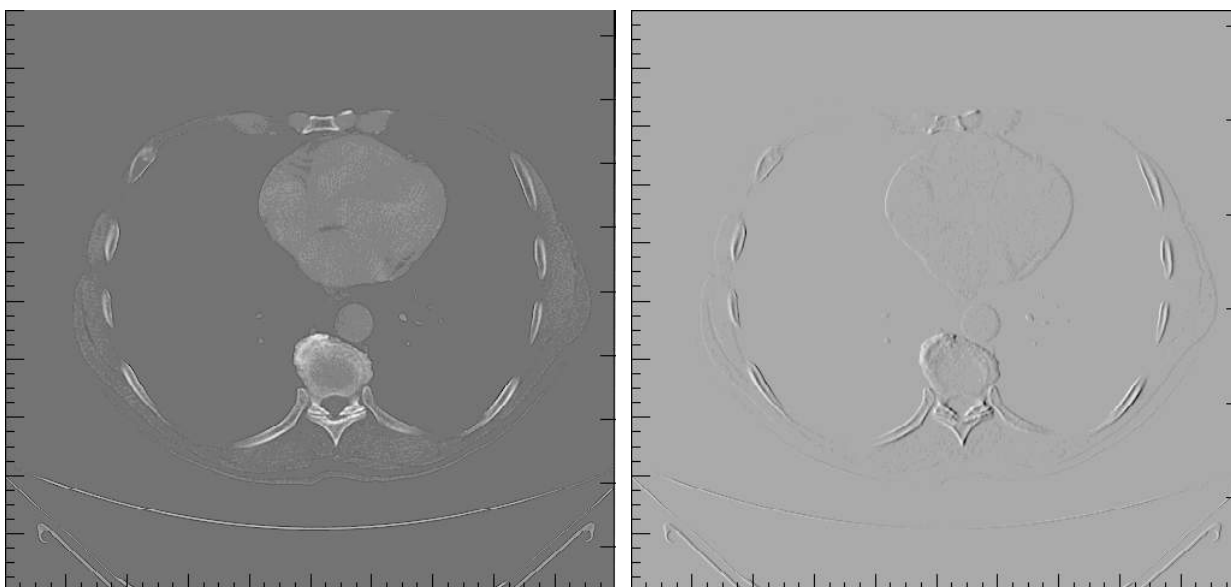


Figure 7.21: CT Sharpening convolution (left) and edge enhancing convolution (right).

According to the results shown in the displayed figures, the developed software is useful, both to read the DICOM images of PET-CT and CT, as to implement convolutions on the same images. For the delimitation of tumors, in particular the application of convolutions that enhance the edges is useful, and as revised in the section 4.1, the convolution of enhanced edges, is not more that plotting the histogram of the partial derivative with respect to the x-coordinate.

Once the equations 4.1, 4.2 and 4.3 are applied to calculate the partial derivatives, and the magnitude of the gradient vector, histograms of the magnitude of the gradient vector for each voxel can be plotted. Figures 7.22, 7.23 and 7.25, 7.24 are gradient magnitude histograms for PET-CT and CT images, respectively.

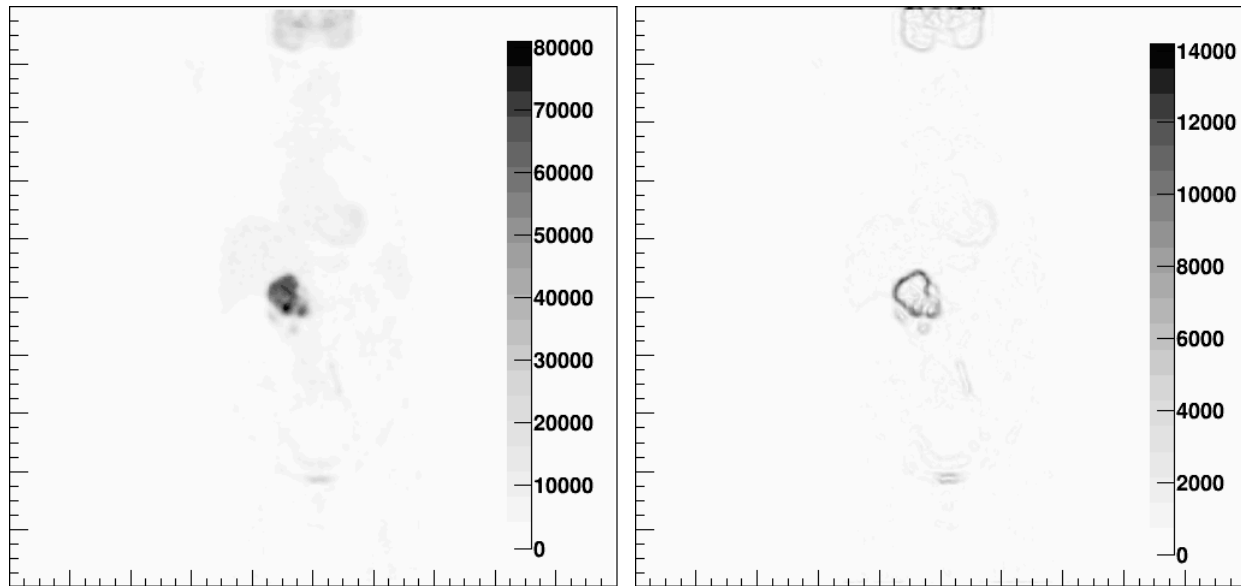


Figure 7.22: PET-CT Image histogram (left), and PET-CT gradient magnitude histogram (right).

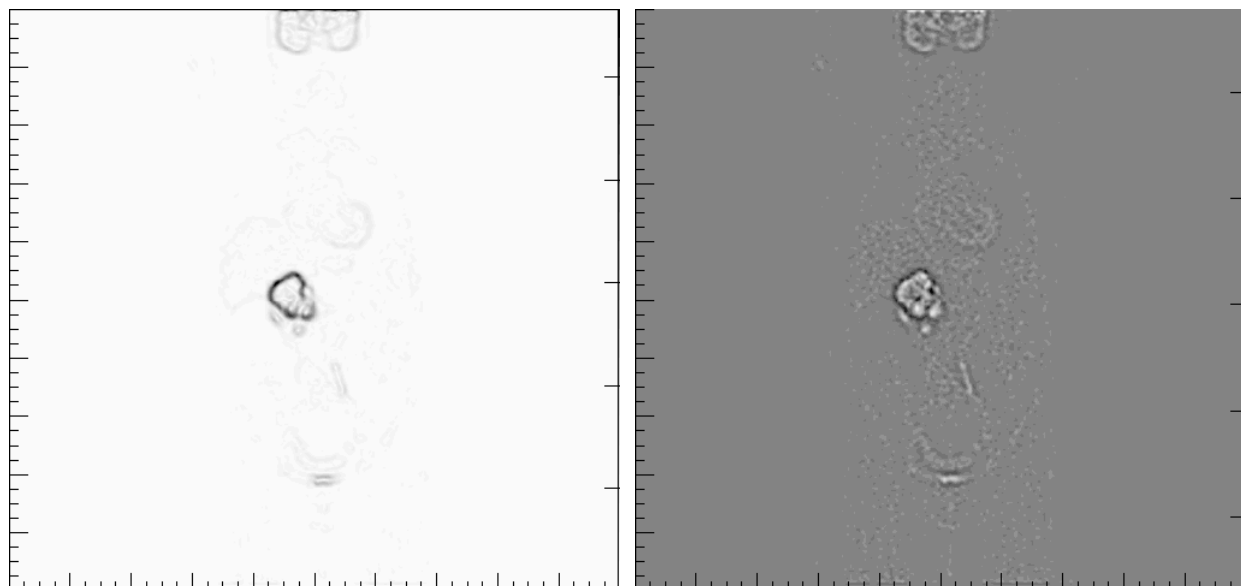


Figure 7.23: PET-CT gradient magnitude (filtered) histogram (right), and PET-CT laplacian (filtered) histogram (right).

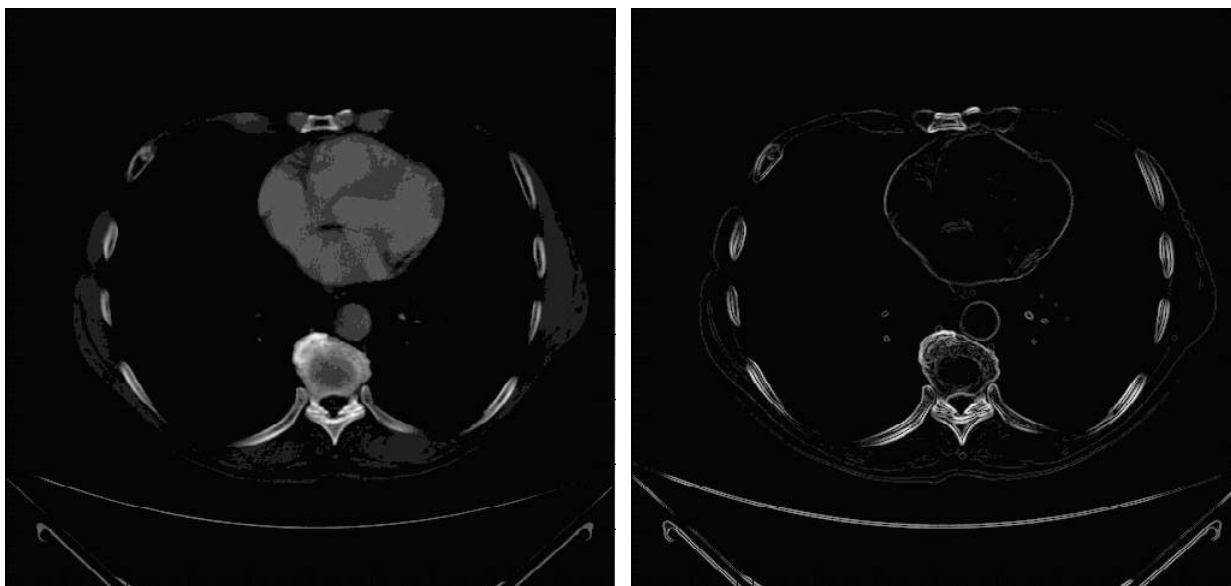


Figure 7.24: CT Image histogram (left), and CT gradient magnitude histogram (right).

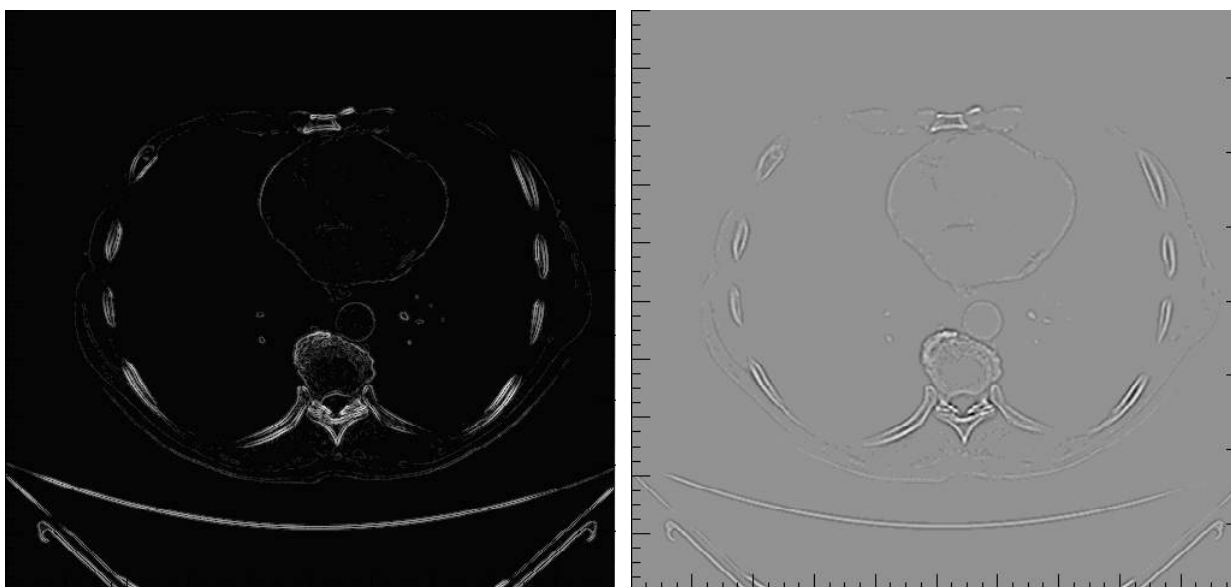


Figure 7.25: CT gradient magnitude (filtered) histogram (right), and CT laplacian (filtered) histogram (right).

In these images, the edges are enhanced, especially, in case of the CT images, and is clear that, the gradient histogram obtained from CT also provides a better representation of the anatomical region edges. This information could be used either as supplementary information in the delimitation of tumors based on PET-CT images or in the diagnosis of other diseases.

Both in the processed image of PET-CT, as in the CT image, the edge enhancing is better when is first applied a Gaussian filter, and then is calculated the gradient, than the image processing when only a gradient is applied. This is because when a Gaussian filter is applied, every voxels are averaged with its neighbors, and for this reason, the variation of the gradient is attenuated in the edges [28].

According to the equation 7.3, when the counts gradient magnitude is calculated, a representation of the mass concentration gradient is done. This concept allows a new stratification of the uptake regions inside the tumor, different to conventional concept of image representing only counts per voxel. This new stratification also provides a better representation of the edges of hyper-uptake regions and this could be useful for the delimitation of tumors and thus for the optimization of the radiation dose in radiation therapy planning.

The mass concentration gradient gives a clearer idea about of the behavior of the radiopharmaceutical in the tissues, but that gradient is not only caused by passive transport, as is described by the Fick's first law [41]. That gradient is also caused in a big part, by the Glucose Transporters (GLUT) that are over-expressed in tumors. However, the results obtained in the Agar Phantom, indicate that the BTV delimited using PET-CT images, includes not only those cells who uptake ^{18}F -FDG, but it also represents those regions of the tissue where the ^{18}F -FDG diffuses, once is chemically modified (defosforilation).

This same principle can be applied in other forms of nuclear medicine, as for example, SPECT, for the realization of dynamic studies, especially when it is in a single organ, for example, the pulmonary perfusion or the myocardial perfusion studies. It is necessary to clarify that in this work, the gradients were calculated for planar images, and therefore, the z component of the gradient vector was not taken into account, so is necessary to process a whole sequence of images, but that is left for subsequent studies.

With regard to the histograms obtained when the Laplacian of counts is calculated, a similar interpretation can be made. The Laplacian of the number of counts represents the Laplacian of mass concentration according to the equation (7.6), and is associated in a temporary change of the mass concentration in certain tissue, but is also associated with transport of glucose via GLUT transporters by facilitated diffusion, for this reason the equation 7.7 can not be used in the analisis of the laplacian histograms. The histograms that represent the Laplacian and the gradient magnitude of the number of counts, could be used in Nuclear Medicine dynamic studies, combined with biokinetic models, to obtain more metabolic information about the images studied.

The edge detection was also realized in the CT and PET-CT diagnostic images using the Canny algorithm. For that, each diagnostic image was converted into a .png image by means of the developed software, and the new image can be processed by means of OpenCV, like was shown in the section 4.2.2. and the results are shown in the figures 7.26 and 7.27. In these images it is possible to observe a complete edge detection of both, anatomical regions (CT Image), and the tumor (PET-CT images). Respect to the developed software, it is pending the implementation of the two missing steps to complete the Canny Edge detector to don't depend of OpenCV.



Figure 7.26: CT Test Image (upper left), and Canny Edge detection with different minimum threshold.

On the other hand, with the developed software, a computational algorithm to delimit the tumor with the SUV_{max} thresholding in the PET-CT images was also implemented. It can be done by assigning a zero value of counts, to those voxels that have a lower value than some percentage of the SUV_{max} . Results of the Test Image SUV_{max} thresholding made in the image 7.22 (left) are shown in the figure 7.28.

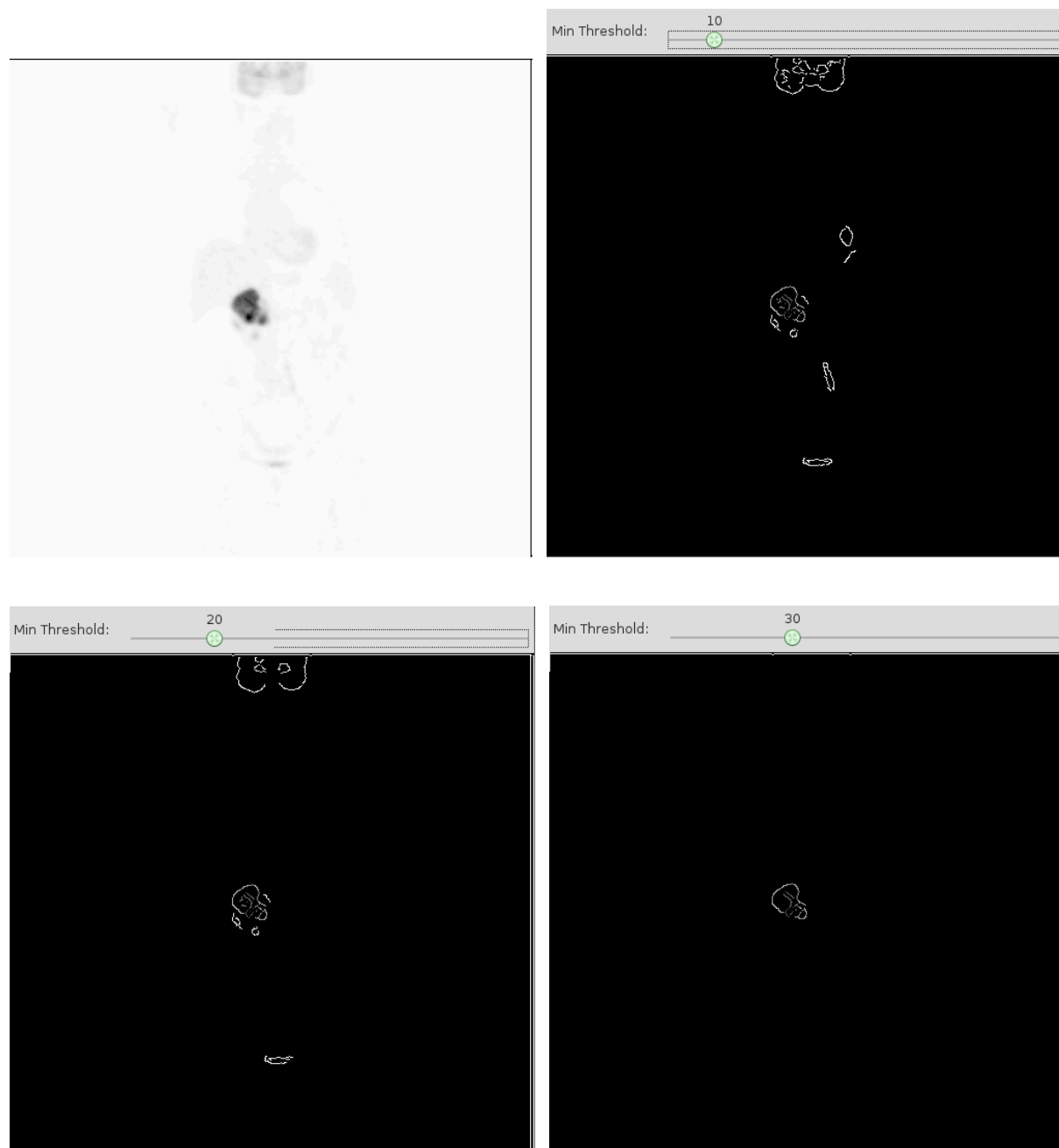


Figure 7.27: PET-CT Test Image (upper left), and Canny Edge detection with different minimum threshold.

In figure 7.28, a delimitation of the tumor can be seen, but some edges inside the tumor cannot be seen as it happens in the gradient magnitude and laplacian histograms (figure 7.23). Only the results in where the tumor is more precisely defined were presented in figure 7.28. The results obtained processing the images obtained by means the Agar Phantom, validate the different methods of delimitation implemented, so it is possible to affirm that highlighted and detected in diagnostic images, edges correspond with accurate approximations for the BTVs.

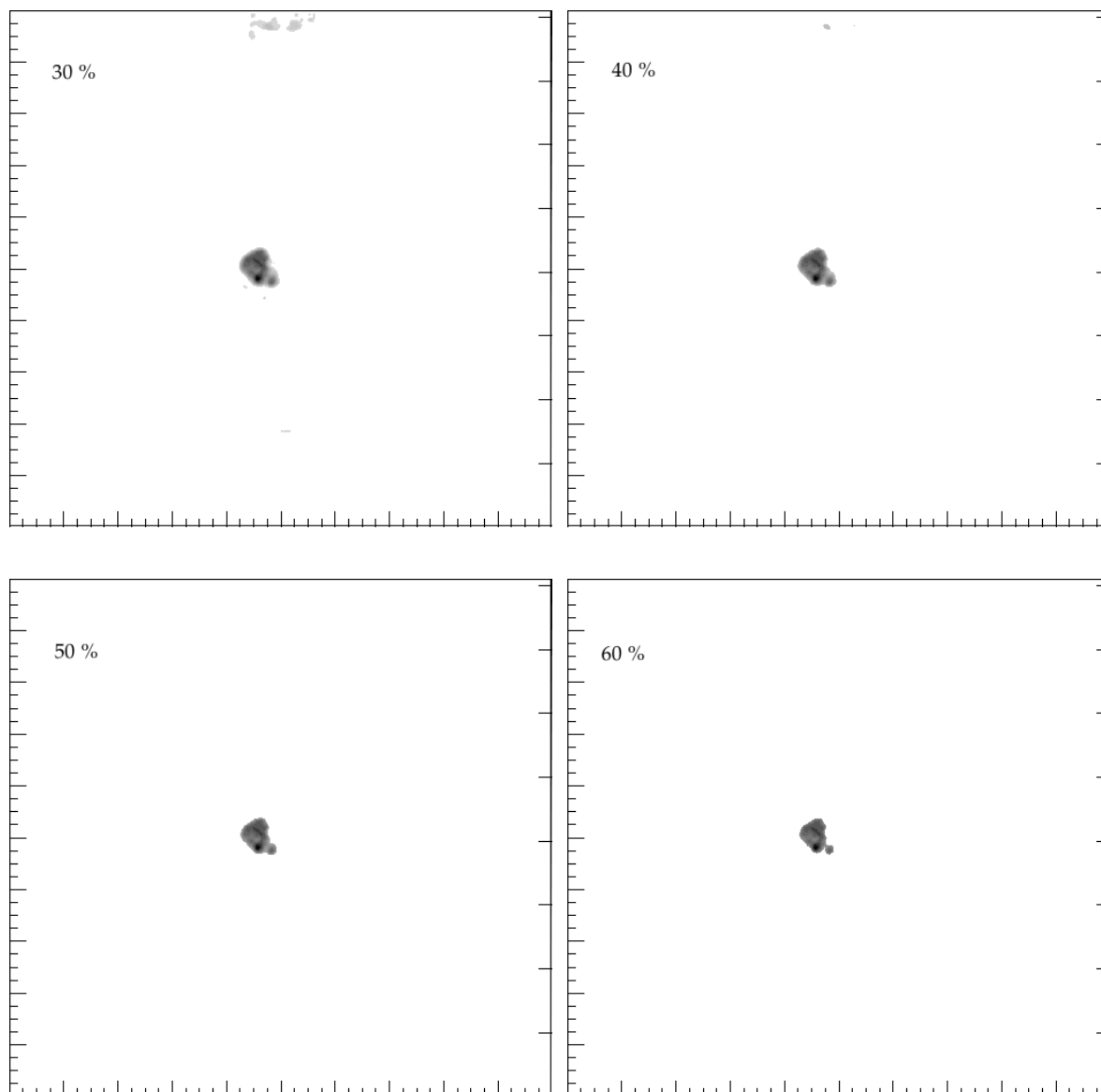


Figure 7.28: PET-CT Test Image SUV_{max} Thresholding using different Thr

The principal objective, to develop a software for the tumor delimitation, was fulfilled, and the designed Agar Phantom, in addition to allow the software validation, allowed to understand in a practical way, that in areas where there is a high number of counts in addition to presenting hyperuptake of radiopharmaceuticals, it is also possible to submit some diffusive processes.

Although, the goal of developing software to enhance the edges of tumors was achieved, remains pending to complete the implementation of the Canny Edge Detector in ROOT. The developed software, can also be modified, to implement other methods of image processing, useful in Medical Physics, as for example, in the calculation of Internal Dosimetry.

Chapter 8

Conclusions and perspectives

The developed software is a useful tool to read diagnostic images of PET-CT and CT, and also to implement image processing algorithms. A Gaussian blur and a sharpening convolution were applied to the diagnostic images, edges were enhanced, and the hyperuptake regions in the PET-CT were observed better. With the edge enhancing, satisfactory results were obtained, but for later works, the implementation of the Canny Edge Detection, using ROOT should be done to complete the edge detection with the developed software, so reach similar results to those obtained with the Canny function present in OpenCV. The implemented computation algorithms could be taken as a basis for the implementation of other algorithms useful for calculations in Nuclear Medicine, or implemented to develop algorithms that facilitate the morphological studies of tumors by means of CT, that may be useful in the emerging field of radiomics.

With regard to the physical meaning of the counts gradient histograms and the Laplacian of counts for the Agar Phantom with ^{18}F -FDG PET-CT, they were associated with diffusion processes explained by the Fick laws and also was possible to make quantitative calculations for the diameter of the hole in the agar, obtaining percent differences (P_d) lower than 5 percent, that allowed to validate and to interpretate the methodologies of delimitation. In the case of diagnostic images, the same interpretation was not done, because in addition to passive diffusion, the ^{18}F -FDG enters in the cell also by processes of facilitated diffusion mediated by GLUT. The results suggest that for low-resolution ^{18}F -FDG PET-CT images, the proposed automated edge-enhancing and edge-detection method yields reliable results and should be used in the tumor delimitation instead of conventional *SUVmax* Thresholding, because they have more physical meaning.

It is possible to apply several biokinetic models, where in addition to considering the Fick's laws, facilitated diffusion processes are considered and studied experimentally with the Agar Phantom by adding cells with a high glucose (or another tracer) consumption, for example tumor cells. These models could be useful to getting more metabolic and dynamic information about the tumors, based on the its image processing and, to develop new methodologies for the dosimetry calculations by means of tumor imaging and software algorithms using the developed software as a basis.

The Jaszczak Phantom was not a good experimental setup to test the scope of the developed software, in contrast with the Agar Phantom that was very useful to understand the relationship between the gradients and Laplacians of counts with the diffusive process in an experimental way, using ^{18}F -FDG. Nothing can be said, respect the dynamic behavior of PET-CT images, because only one slice was analyzed, and for a single acquisition time. However it is possible to make the dynamical analysis by means of Agar Phantom, but that is left to subsequent works, either in PET-CT, or in SPECT-CT.

References

- [1] Hall. D. Microprocessors and digital systems. McGraw-Hill, 1983
- [2] The Gimp Documentation. Desaturate. Online. Mayo 20, 2014.
Available in: <http://docs.gimp.org/2.6/en/gimp-tool-desaturate.html>
- [3] Deans. S. The Radon Transform and Some of its applications. John Wiley and Sons. Canada. 1983. Chapter 1.
- [4] Powsner R. Essential Nuclear Medicine Physics. Second edition. Blackwell Publishing. 2006.
- [5] What is Nuclear Medicine. Spetect Scan. Online. June, 2014.
Available in: <http://www.whatisnuclearmedicine.com/upload/SPECT%20scanner.jpg>
- [6] Infobuddy. Scintigraphy. Online. June, 2014.
Available in: http://www.infobuddy.nl/images/beeldvormende_technieken_4_400.jpg
- [7] Leo. William R. Techniques for Nuclear and Particle Physics Experiments: A How-To Approach. Springer Science & Business Media. Germany. 1987. Chapter 5.
- [8] Zaidi H. Quantitative Analysis in Nuclear Medicine Imaging. Springer Science. Singapore. 2006. Chapters 2, 6.
- [9] Saha G. Physics and Radiobiology of Nuclear Medicine. Third Edition. Springer Science. 2006.
- [10] University of Washington - Division of Nuclear Medicine. Introduction to PET Physics. Online. June, 2014. Available in: http://depts.washington.edu/nucmed/IRL/pet_intro/toc.html
- [11] Vujosević B, Bokorov B. *Radiotherapy: past and present. Arch Oncol 2010;18(4):140-2.*
- [12] MacManus M, Nestle U, Rosenzweig K, Carrio I, Messa C, Belohlavek O, Danna M, Inoue T, Deniaud E, Schipani S, Watanabe N, Dondi M, Jeremic B. *Use of PET and PET/CT for Radiation Therapy Planning: IAEA expert report 2006–2007. Radiotherapy and Oncology 91 (2009) 85–94.*
- [13] Varian. ECLIPSETM overview. Planning Technology. Online. September, 2013.
Available in: http://www.varian.com/us/oncology/radiation_oncology/eclipse/#.UZqih6pKgZ4

- [14] Radiotherapy Physics. Conformal Radiotherapy. Online. June, 2014.
Available in: <http://www.radiotherapyphysics.com/RadiotherapyPhysics/Pathway>
 - [15] Burnet N, Thomas S, Burton K, Jefferies S. *Defining the tumour and target volumes for radiotherapy. cáncer Imaging (2004) 4, 153–161*
 - [16] Chiti A, Kirienko M, Grégoire V. Clinical use of PET-CT data for radiotherapy planning: What are we looking for?. *Radiotherapy and Oncology*. 96. 2010. 277–279.
 - [17] Thorwarth D. Geets X. Paiusco M. Physical radiotherapy treatment planning based on functional PET/CT data. *Radiotherapy and Oncology*. 96. 2010. 317–324.
 - [18] Páez D. Reyes A. Llamas A. Positron Emission Tomography: A New Paradigm in Cancer Management. *Rev Colomb Cancerol* 2007. 11(1). 5-12.
 - [19] Haubner. R. PET radiopharmaceuticals in radiation treatment planning – Synthesis and biological characteristics. *Radiotherapy and Oncology*. 96. 2010. 280–287.
 - [20] Bortfeld T, Schmidt R, Neve W, Wazer D. *Image-Guided IMRT*. Springer Verlag. Berlin. 2006. Chap 2,8.
 - [21] Lee N, Mechalakos J, Nehmeh S, Lin Z, Squire O, Cai S, Chan K, Zanzonico P, Greco C, Ling C, Humm J, Schöder H. Fluorine-18-labeled fluoromisonidazole positron emission and computed tomography-guided intensity-modulated radiotherapy for head and neck cancer: a feasibility study. *Int J Radiat Oncol Biol Phys*. 2008.70(1). pp 2-13.
 - [22] Grosu A, Souvatzoglou M, Röper B, Dobritz M, Wiedenmann N, Jacob V, Wester HJ, Reischl G, Machulla HJ, Schwaiger M, Molls M, Piert M. Hypoxia imaging with FAZA-PET and theoretical considerations with regard to dose painting for individualization of radiotherapy in patients with head and neck cancer. *Int J Radiat Oncol Biol Phys*. 2007 Oct 1. 69(2). pp 541-51.
 - [23] Chao K, Bosch W, Mutic S, Lewis J, Dehdashti F, Mintun M, Dempsey J, Perez C, Purdy J, Welch M. A novel approach to overcome hypoxic tumor resistance: Cu-ATSM-guided intensity-modulated radiation therapy. *Int J Radiat Oncol Biol Phys*. 2001. 49(4). pp 1171-82.
 - [24] IAEA. *The Role of PET/CT in Radiation Treatment Planning for cáncer Patient Treatment*. IAEA-TECDOC-1603. 2008.
 - [25] Price P, Jones T. The role of PET scanning in radiotherapy. *The British Journal of Radiology*. Supplement 28(1). pp. 2–4.
 - [26] IAEA. Use of PET and PET/CT for Radiation Therapy Planning: IAEA expert report 2006–2007 . *Radiotherapy and Oncology*. 91. 2009. 85–94 .
 - [27] Eakin R, Foster J, Hanna, G, Hounsell A, Lynch T, McAleese J, McNally O, Page W, Stewart D, Summers Y. *The Role of PET-CT in Radiotherapy Planning*. *Oncology News*. Volume 1. Issue 6. April-May 2007.
 - [28] Canny. J. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. PAMI-8, 6. Nov. 1986. pp 679 - 698.
 - [29] Nadernejad. E, Sharifzadeh. S, Hassanpour. H. Edge Detection Techniques: Evaluations and Comparison. *Applied Mathematical Sciences*. Vol. 2. No. 31. 2008, pp. 1507-1520.
-

- [30] Lynch, Daniel R. Numerical Partial Differential Equations for Environmental Scientists and Engineers. Springer Verlag. New York. 2005. Chapter 2.
- [31] Marsden J. Tromba A. Vectorial Calculus. W H Freeman & Co (Sd); 3rd edition. 1988. Chapters 2-3.
- [32] Malaterre M. Grassroots DICOM Cross-platform DICOM implementation. Online. October 4, 2013.
Available in: <http://sourceforge.net/projects/gdcm/>. Consulted on
- [33] Cern. GdcmStringFilter C++ class. Online. October 4, 2013.
Available in: http://gdcm.sourceforge.net/html/gdcmStringFilter_8h_source.html
- [34] Cern. gdcmImageReader C++ class. October 4, 2013.
Available in: http://gdcm.sourceforge.net/html/gdcmImageReader_8h_source.html.
- [35] Cern. ROOT Data Analysis Framework. Online. October 2, 2013.
Available in: <http://root.cern.ch/drupal/>.
- [36] PET Phantom Instructions for Evaluation of PET Image Quality. American College of Radiology. ACR Nuclear Medicine Accreditation Program PET Module. 2013.
Available in <http://www.acr.org/~media/ACR/Documents/Accreditation/Nuclear%20Medicine%20PET/PETPhantomInstructions.pdf>
- [37] Health Physics Norwest. Equipment evaluation. Online. September, 2013 .
Available in: <http://www.hpnw.com/equipment-evaluation-accreditation1/>
- [38] Berenice Pérez Lopez, Iván Fabricio Vega-Gonzalez, Enrique Estrada-Lobato, J. Jesús Pérez Molina, Blanca Miriam Torres-Mendoza, Luis Alberto Medina. *Volume-of-Interest Assessment of Oncologic Response Using 18F-FDG PET-CT: A Phantom Study. Journal of NUCLEAR MEDICINE TECHNOLOGY. Vol. 39. No. 1. 2011.*
- [39] Parija S. Textbook of Microbiology & Immunology. Elsevier. India. 2009. pp 70-72.
- [40] J. Crank. The Mathematics of Diffusion. Oxford University Press. Second Edition. 1975. Chapter 1.
- [41] Sperelakis. N. Cell Physiology Sourcebook Essentials of Membrane Biophysics. Fourth Edition. Elsevier Academic Press. Canada. 2012. Chapters 8, 11.
- [42] Kontaxakis G, Vaquero J, Santos A. Reconstrucción de imagen en Tomografía por Emisión de Positrones. Rev.R.Acad.Cienc.Exact.Fis.Nat. Vol. 96, N.os 1-2. España. 2002. pp 45-57, 2002.
- [43] Kak,Slaney. Principles of Computerized Tomographic Imaging. IEEE Press. 1988. Chapter 3.
- [44] Steven W. Smith. The Scientist and Engineer's Guide to Digital Signal Processing. Online. May, 2014.
Available in: <http://www.dspguide.com/ch25/5.htm>
- [45] Ramirez J. Arboleda C. McCollough C. Tomografía computarizada por rayos X: fundamentos y actualidad. Revista Ingeniería Biomédica. ISSN 1909-9762. Vol 2 (4). Medellín, Colombia. 2008.
-

Chapter 9

Appendix 1. Filtered back projection algorithm, Mathematical Fundamentals.

The filtered back projection, is an algorithm that is implemented in the main imaging techniques such as computed tomography (CT) and positron emission tomography (PET). In this chapter, the mathematical principles of the algorithm will be shown.

9.1 Radon Transform [3].

$f(x,y)$ is a function whose domain is a region D in the xy plane as shown in the figure 9.2.

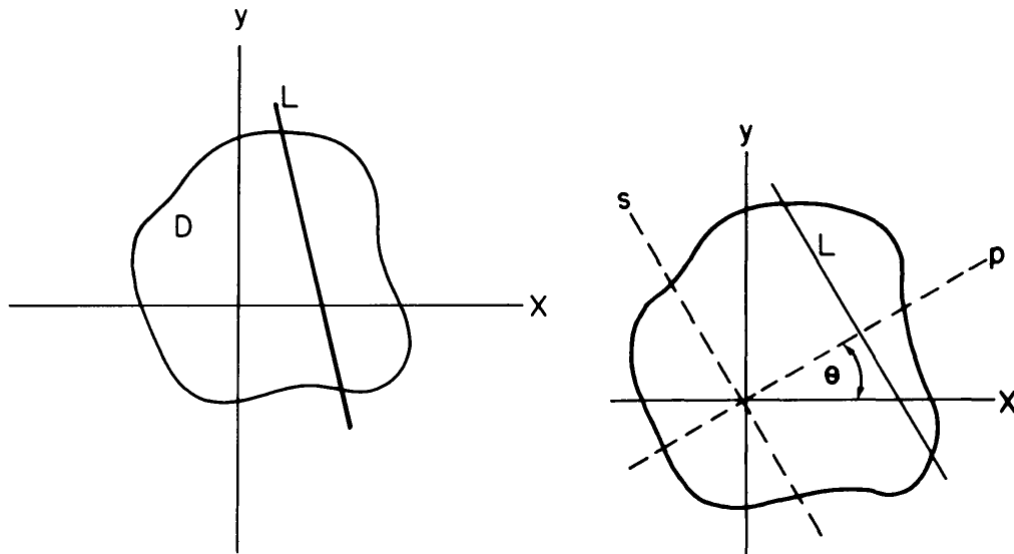


Figure 9.1: Domain D for the function $f(x,y)$ (left). Path of integration view from a coordinate system ps (Right). Taken from [3].

To evaluate the integral through the line L , which in the system coordinate ps (obtained by rotating the xy system at an angle θ) can be seen as a constant function with the form $p = t = \text{constant}$. The coordinates of any point in the domain D , can be found from the xy -coordinates, using the rotation matrix:

$$\begin{pmatrix} p \\ s \end{pmatrix} = \begin{pmatrix} \cos\theta & \text{sen}\theta \\ -\text{sen}\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (9.1)$$

The matrix product has the form:

$$\begin{aligned} p &= x \cdot \cos\theta + y \cdot \text{sen}\theta \\ s &= -x \cdot \text{sen}\theta + y \cdot \cos\theta \end{aligned}$$

If the equation (9.1) is multiplied by the inverse of the rotation matrix:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos\theta & -\text{sen}\theta \\ \text{sen}\theta & \cos\theta \end{pmatrix} \begin{pmatrix} p \\ s \end{pmatrix}$$

That is:

$$\begin{aligned} x &= p \cdot \cos\theta - s \cdot \text{sen}\theta \\ y &= p \cdot \text{sen}\theta + s \cdot \cos\theta \end{aligned}$$

For $p = t = \text{constant}$:

$$\begin{aligned} x &= t \cdot \cos\theta - s \cdot \text{sen}\theta \\ y &= t \cdot \text{sen}\theta + s \cdot \cos\theta \end{aligned} \quad (9.2)$$

$$\begin{aligned} t &= x \cdot \cos\theta + y \cdot \text{sen}\theta \\ s &= -x \cdot \text{sen}\theta + y \cdot \cos\theta \end{aligned} \quad (9.3)$$

In the rotated coordinated system, the projection $p_\theta(t)$ is defined like:

$$p_\theta(t) = \int_L f(x,y) ds \quad (9.4)$$

That is the line integral of the function $f(x,y)$, along the line $p = t$, with length element ds . This projection is what is known as Radon transform. With the equation 9.2, $f(x,y)$ can be evaluated on the line L , and the Radon transform has the form:

$$p_{\theta}(t) = \int_{\mathbb{R}} f(t \cos\theta - s \sin\theta, t \sin\theta + s \cos\theta) ds \quad (9.5)$$

Along the integration path, t is a constant. However, when the path integral is evaluated, it will depend on the value of s , so the projection is depending on the values of t and θ . In the figure 9.2, for the same θ angle, can be seen that the value of the projection $p_{\theta}(t)$ is different for each t .

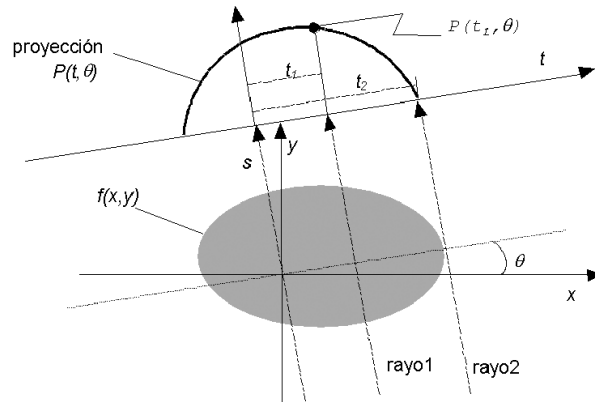


Figure 9.2: Projections $p_{\theta}(t)$ for different t . Taken from [42].

9.2 Inverse Radon Transform [43].

The Fourier transform for a function of one variable is defined as:

$$F(\omega) = \int_{\mathbb{R}} f(x) e^{-i\omega x} dx \quad (9.6)$$

In a similar way, the Fourier transform for a function of two variables, is defined like:

$$F(\vec{\omega}) = \iint_{\mathbb{R}} f(\vec{r}) e^{-i\vec{\omega} \cdot \vec{r}} d\vec{r} \quad (9.7)$$

Then, the Fourier Transform $P_{\theta}(\omega)$ of the projection $p_{\theta}(t)$ is:

$$P_{\theta}(\omega) = \int_{\mathbb{R}} p_{\theta}(t) e^{-i\omega t} dt = \int_{\mathbb{R}} \left[\int_{\mathbb{R}} f(t \cos\theta - s \sin\theta, t \sin\theta + s \cos\theta) ds \right] e^{-i\omega t} dt \quad (9.8)$$

That is:

$$P_{\theta}(\omega) = \iint_{\mathbb{R}} f(t \cos\theta - s \sin\theta, t \sin\theta + s \cos\theta) e^{-i\omega t} ds dt \quad (9.9)$$

The values x, y and t can be replaced by the use of the equations 9.2 and 9.3: ¹

$$P_{\theta}(\omega) = \iint_{\mathbb{R}} f(x, y) e^{-i\omega(x \cos\theta + y \sin\theta)} dx dy \quad (9.10)$$

And this equation can be rewritten as:

$$P_{\theta}(\omega) = F(\omega \cos\theta, \omega \sin\theta) \quad (9.11)$$

In accordance with the equation (9.11), as the projection $P_{\theta}(\omega)$ is a kind of a Fourier Transform from the function $f(x, y)$, it is possible to find $f(x, y)$ from transformed inverse of $P_{\theta}(\omega)$. For the Fourier Transform in two variables, (equation 9.7), the inverse associated is:

$$f(\vec{r}) = \iint_{\mathbb{R}} F(\vec{\omega}) e^{i\vec{\omega} \cdot \vec{r}} d\vec{\omega} \quad (9.12)$$

Bearing in mind that $\vec{r} = (x, y)$ and $\vec{\omega} = (\omega_x, \omega_y)$, the previous equation can be written as:

$$f(x, y) = \iint_{\mathbb{R}} F(\omega_x, \omega_y) e^{i(x\omega_x + y\omega_y)} d\omega_x d\omega_y \quad (9.13)$$

In polar coordinates, in the plane of the frequencies ω_x and ω_y , $\omega = \|\vec{\omega}\|$, $\omega_x = \omega \cos\theta$ and $\omega_y = \omega \sin\theta$, also $F(\omega_x, \omega_y) = F(\omega \cos\theta, \omega \sin\theta)$. The area element is no longer $d\omega_x d\omega_y$, now is $\omega d\omega d\theta$. In this way, the inverse Fourier transformed in polar coordinates is written as:

$$f(x, y) = \int_0^{2\pi} \int_0^{\infty} F(\omega \cos\theta, \omega \sin\theta) e^{i(x\omega \cos\theta + y\omega \sin\theta)} \omega d\omega d\theta$$

The limits of integration in the above equation can be changed by symmetry:

$$f(x, y) = \int_0^{\pi} \int_{-\infty}^{\infty} F(\omega \cos\theta, \omega \sin\theta) e^{i\omega(x \cos\theta + y \sin\theta)} \omega d\omega d\theta$$

¹Where the area element is $ds dt = J dx dy$, but the Jacobian of the transformation is $J = 1$, because the transformation matrix is unitary.

The equation (9.11) can be replaced in the previous equation and, because $t = x\cos\theta + y\sin\theta$:

$$f(x, y) = \int_0^\pi \left[\int_{-\infty}^{\infty} P_\theta(\omega) e^{i\omega t} \omega d\omega \right] d\theta = \int_0^\pi Q_\theta(t) d\theta \quad (9.14)$$

The integral in the square brackets $Q_\theta(t) = \int_{-\infty}^{\infty} P_\theta(\omega) e^{i\omega t} \omega d\omega$ represents a filtering operation, where ω is the frequency of the filter, and the integral on the angle θ represents the back projection, which is the sum of each of the filtered projections $Q_\theta(t)$ for each angle θ . The equation (9.14) can be written as:

$$f(x, y) = \int_0^\pi Q_\theta(x\cos\theta + y\sin\theta) d\theta \quad (9.15)$$

The equation (9.14) is the mathematical basis for a method of reconstruction of images called filtered back projection by means of that, objects images can be obtained from each projection $p_\theta(t)$.

9.3 Filtered back projection Method.

In the practice, when projections of an object, are made for different angles θ , a distorted image of the object is obtained, as shown in the figure 9.3. For that reason, once the projection has been realized, and there had been obtained the Fourier transform of the same one, it is convoluted with a filter [44].

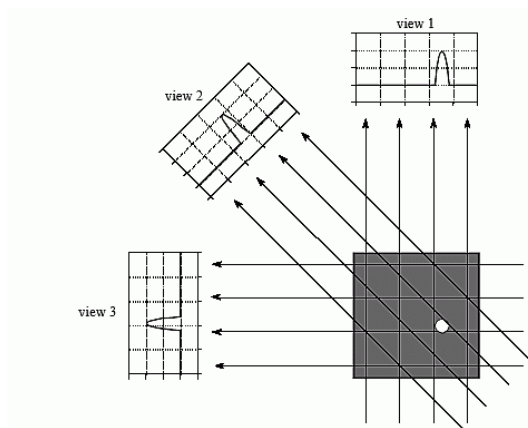


FIGURE 25-15
CT views. Computed tomography acquires a set of views and then reconstructs the corresponding image. Each sample in a view is equal to the sum of the image values along the ray that points to that sample. In this example, the image is a small pillbox surrounded by zeros. While only three views are shown here, a typical CT scan uses hundreds of views at slightly different angles.

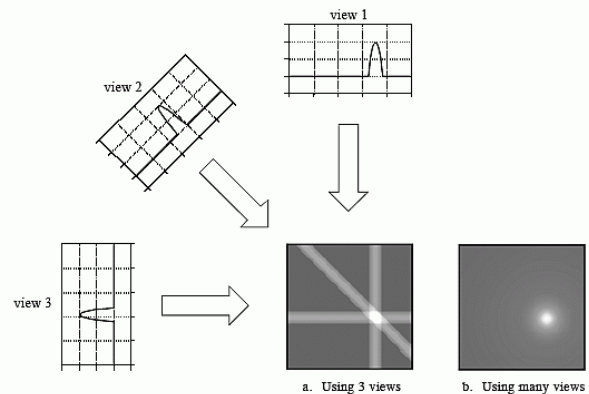


FIGURE 25-16
Backprojection. Backprojection reconstructs an image by taking each view and *smearing* it along the path it was originally acquired. The resulting image is a blurry version of the correct image.

Figure 9.3: Simple back projection, where the distortion associated with the reconstruction process can be seen. Simple back projection (left). Obtaining of projections of an object (Right). Taken from [44]

For the filtered back projection method, it is possible to generalize the equation (9.14) by means of the change of the filter ω , for a filter $g(\omega)$ that is a more general kernel [45]:

$$f(x,y) = \int_0^{2\pi} \left[\int_{-\infty}^{\infty} P_{\theta}(\omega) e^{i\omega t} g(\omega) d\omega \right] d\theta \quad (9.16)$$

In the integral over the angle θ , this time θ increases up to 2π . As shown in the previous subsection, the first thing that is done is to obtain a projection $p_{\theta}(t)$. After, one Fourier transform in one variable is applied to the projection, then, a convolution is applied, and finally $f(x,y)$ is obtained. Figure 9.4 shows a diagram of the steps to perform a filtered back projection.

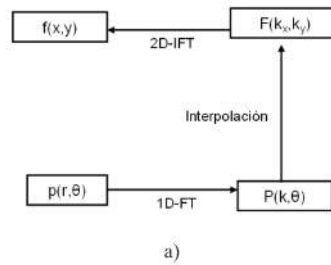


Figure 9.4: Steps of the filtered-projection algorithm. Taken from [45]

Chapter 10

Appendix 2. Calculation of the inner (R_i) and outer (R_e) radius.

The area of the ring (A_r) can be expressed as:

$$A_r = \pi(R_e^2 - R_i^2) = \pi \underbrace{(R_e - R_i)}_{\varepsilon} \cdot \underbrace{(R_e + R_i)}_{2\langle R \rangle} = 2\pi\varepsilon \langle R \rangle \quad (10.1)$$

A_r can be calculated also from the count of the pixels exceeding the threshold. And the mean diameter $\langle D \rangle$ can be also calculated. The point in the center of the ring corresponds to the center of mass of the points that are present in the ring, so its x_{cm} coordinates can be calculated as:

$$x_{cm} = \frac{\sum_{i=1}^r x_i m_i}{\sum_{i=1}^r m_i} = \frac{\sum_{i=1}^r x_i m}{\sum_{i=1}^r m} = \frac{\mathcal{M} \sum_{i=1}^r x_i}{r\mathcal{M}} = \frac{\sum_{i=1}^r x_i}{r}, \quad y_{cm} = \frac{\sum_{i=1}^s y_i}{s}$$

Because all the pixels have the same area, its "weights" are the same, ie $m_i = m$. r and s are the those pixels who along the x and the y axis that do not have a zero counts value. This calculation can be made very easily, storing the values of x and y per each voxel into independent ROOT TH2D matrices (for example X_{cm} and Y_{cm}), and then calculating the ratio $X_{cm}.Sum()/X_{cm}.NonZeros()$. To calculate the average radius $\langle R \rangle$, for those pixels in the ring that exceed the threshold value, in accordance with the mathematical definition of a circle in \mathbb{R}^2 :

$$R_{ij} = \sqrt{(i - x_{cm})^2 + (j - y_{cm})^2}$$

R_{ij} values are stored in a TH2D matrix, and $\langle R \rangle$ is the mean for the values of this matrix. ε is calculated from equation 10.1, and using the calculated values for A_r and $\langle R \rangle$:

$$\varepsilon = \frac{A_r}{2\pi \langle R \rangle}$$

In the equation 10.1, two equations are defined:

$$\begin{aligned}\varepsilon &= R_e - R_i \\ \langle D \rangle &= R_e + R_i = 2 \langle R \rangle\end{aligned}$$

And from these equations:

$$R_e = \frac{\langle D \rangle + \varepsilon}{2}$$

$$R_i = \frac{\langle D \rangle - \varepsilon}{2}$$