TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY

AMALIA MORENO GALERA

DATA PROCESSING TOOLBOX FOR PET SCANNERS

Master of Science Thesis

Examiners: Professor Ulla Ruotsalainen
Professor Jari Viik
M.Sc. Jussi Forma
Examiners and topic approved in the Natural Science Faculty Council meeting on 4 September 2013.

ABSTRACT

TAMPERE UNIVERSITY OF TECHNOLOGY
International Master's Degree Programme in Biomedical Engineering
MORENO GALERA, AMALIA: Data Processing Toolbox for PET Scanners
Master of Science Thesis, 84 pages, 3 Appendix pages
March 2014
Major subject: Medical Physics
Examiner: Professor Ulla Ruotsalainen, Professor Jari Viik and M.Sc. Jussi Forma
Keywords: PET, List Mode, histogramming events, Sinogram, data corrections, image reconstruction.

This thesis presents the design and implementation of a Data Processing Toolbox to process and reconstruct the information acquired by different PET scanners. The Toolbox is divided in three processes: generate realistic list-mode files, histogram list-mode data sets and reconstruct sinograms into 3D images.

The first process produces list-mode data from numerical phantoms. This tool is useful in the early stages of developing a PET scanner because permits to process the Toolbox performance when there is not available data. The list-mode file real appearance is due to the bootstrapping and shuffling tools. In addition, the bootstrapping function, which calculates statistically similar sinograms than the acquired, can be used for results realizations and phantom measurements. The second process histograms events from different PET scanners into sinograms. The adaptability to the different scanners is related to the Radon transform dependence with the scanners geometry and format. The third process corrects, rebins and reconstructs sinograms into 3D images. In order to avoid artefacts, different methods have been used to correct the sinograms based on the quantification of events. A rebinning method was selected to work with 3D data due to the computational time decrease and the use of 2D algorithms for the image reconstruction. This method transforms the 3D data to 2D where the information contained in the oblique planes is added to the transaxial planes. The Maximum Likelihood Expectation Maximization (MLEM) iterative algorithm has been used for the reconstruction of the sinograms into the image domain.

The proper Toolbox functionality was demonstrated by numerical simulations and data from two real PET scanners. The numerical phantom and real data are in the two PET data recording options: list-mode file and sinogram data formats. In addition, the evaluation of the Toolbox efficiency is presented studying the parameter dependency with the executing time of the functions.

As a conclusion, this thesis describes a useful Toolbox for a PET scanner development and use. The proposed Toolbox can be used for different scanners due to its adaptability to the scanner geometry. Moreover, due to the Toolbox modularity, the functions can be used independently.

PREFACE

This thesis work has been developed in the Methods and Models for Biological Signals and Images (M^2oBSI) research group, at the Department of Signal Processing, Tampere University of Technology, Finland.

I would like to express my gratitude to my supervisors Prof. Ulla Ruotsalainen and MSc. Jussi Forma. I am grateful for your guidance and especially for your wise advices and support, professionally and personally. I wish to show my gratitude to my colleagues in the M^2oBSI because working in this group has been easy thanks to them. Especially, to Defne Us because you have shown me that you are more than a colleague, you are a friend.

Finally, I would like to thank to my family, my father Rafael Moreno and my mother Juani Galera. I cannot thank you enough everything that you have done for me. You have been in the difficult moments given me support and in the happiest to enjoy them together, thanks. In general, I am grateful to my friends and family for all the moral support and encouragement.

Thanks to the love of my life, Borja, for all the things that we have done together and for all the things that we have still to live, without you nothing would be possible.

Amalia Moreno Galera
Tampere, March 2014

Insinöörinkatu 60 B 129
33720 Tampere, FINLAND
+358 403 622 600
Amalia.morenogalera@tut.fi

TABLE OF CONTENTS

ACRONYMS

2D/3D	Two Dimensional and Three Dimensional
ART	Algebraic Reconstruction Technique
AX-PET	Axial Positron Emission Tomography
CERN	European Organization for Nuclear Research
COMPET	pre-clinical MRI COMpatible PET
CT	Computed Tomography
EM	Electron Microscopy
ECAT	Emission Computed Axial Tomograph
FBP	Filtered Backprojections
f-MRI	functional Magnetic Resonance Imaging
FoV	Field of View
keV	kilo electron Volts
LM	List-Mode
LoR	Line of Response
M^2BSI	Methods and Models for Biological Signals and Images
ML	Maximum Likelihood
MLEM	Maximum Likelihood Expectation Maximization
MMG	low-energy X-rays MaMmoGraphy
MRI	Magnetic Resonance Imaging
MSE	Mean Squared Error
PET	Positron Emission Tomography
PD	Poisson Distribution
PHA	Pulse Height Analyzer

PM	Photo Multiplier
RT	Radon Transform
SL	Shepp-Logan
SPECT	Single Photon Emission Computed Tomography
SSRB	Single Slice Rebinning Method
TIS	Tomographic Imaging System
ToF	Time Of Flight
US	UltraSonography

1. INTRODUCTION

Since ancient times evolution of the medicine has been tremendous, especially in the last century [1]. This is due to the discovery of the X-rays by Wilhelm Conrad Röntgen in 1895 [2] and a spontaneous reaction with the use of Uranium by Henri Becquerel in 1896 [3]. Marie Curie called this reaction as radioactivity and presented in 1903 [4] a study that describes this phenomenon for additional radioactive elements as thorium, polonium and radium. Note that these radioactive elements were used in radiotherapy for treating some diseases through intravenous injection as it is collected in a Frederick Proescher's study in 1913 [5].

However, it wasn't until 1934 when Irene Curie and Frederic Joliot produced artificially the first radioisotope as well as studied the reaction of an unstable radioisotope that decays by positron emission [6]. These discoveries, added to the invention of the Lawrence's cyclotron in 1932 [7], were the most significant milestones to define the origin of the Nuclear Medicine. Nonetheless, it took another 30 years until the radionuclides were produced for medical use. The use of radionuclides has varied depending on the purpose and medical imaging technique utilized. Currently, the most commonly used are Thallium-201, Fluorine-18, Iodine-131, Oxygen-15, Nitrogen-13, Carbon-11, or Rubidium-87 [8].

Medical imaging techniques were developed to capture images or processes that occur inside the body in the least invasive manner as possible. According to the source and the physical properties of the tissue, the medical imaging techniques can be differentiated in two groups: anatomical/structural or functional images. Some medical devices that provide anatomical images are: Electron Microscopy (EM) discovered in 1931 [9], UltraSonography (US) in 1949 [10], Planar X-Ray Imaging in 1957 [11], Mammography (MMG) in 1969 [12], X-ray Computed Tomography (CT) in 1972 [13] and Magnetic Resonance Imaging (MRI) in 1977 [14]. On the other hand, some functional imaging modalities are: Positron Emission Tomography (PET) discovered during the 1970s [15], Single Photon Emission Computed Tomography (SPECT) in 1980 [16], and functional-MRI (f-MRI) discovered in 1990 [17]. Furthermore, these techniques can be classified based on the use of ionized or non-ionized radiation as well as external or internal sources.

The external sources release radiation through the object to be imaged and detectors situated behind the object collect the information received, which is radiation attenuated by tissues. Some methods that use external ionized radiation are: EM, Planar X-Ray

Imaging, CT or MMG. In contrast with the methods based on external ionized radiation, the methods based on external non-ionized radiation transmit energy through the object and collect the information by reflection. One example of non-ionized radiation method is ultrasonography.

Internal ionized radiation is produced by radionuclides introduced inside the object, intravenously or orally. Then external detectors capture the emitted ionized radiation as in PET or SPECT. This thesis is concentrating in PET technology.

First positron scanning machine developed by Gordon L. Brownell in 1953 [18], evolved from the commercial Emission Computed Axial Tomograph (ECAT) model, designed by Dr. Phelps and Dr. Hoffman in 1978 [19], into current PET scanners. Some examples of PET scanners are: current MRI compatible pre-clinical PET scanner (COMPET) [20] and Inveon PET scanner from Siemens [21]. Both scanners have been used in this Master thesis work. COMPET model is a research project dedicated to build a high resolution, high sensitivity and MRI compatible PET scanner. Inveon PET scanner offers high resolution and sensitivity delivering high-quality images and optimal results. The major differences between scanners are in the design of the scanner, rotation of the multi-crystals detection system, use of a dual-head Anger PET camera, or application of cylinder PET systems [22].

Due to the computational complexity of tomographic techniques, tomographic image acquisition processes need a model to create an image. In this model, the data associated with the cross-sectional scans of an object is reflected. This is fulfilled with the Radon Transform (RT), which was developed in 1917 by Johann Radon [23]. Currently, other methods can be used for the acquisition of the projected data. Analytical or iterative approaches can be used in order to reconstruct the images into the spatial domain.

The objective of this thesis is to process, correct and reconstructs the information acquired by PET scanners into images in the spatial domain. A Toolbox was created with three main sections for this purpose, one per each data manipulation task. A functionality of this Toolbox is to be used for the development of an innovative PET scanner prototype, the Axial PET (AX-PET), based on the new technology developed at the European Organization for Nuclear Research (CERN) [24]. In this novel approach, the scintillator crystals are placed axially along the Field Of View (FOV), instead of placing them radially in the tomograph. This modification increases the axial resolution and sensitivity reducing the number of crystals needed.

In order to test the Toolbox functions numerical simulations and real List-Mode (LM) data, from PET scanners as Inveon or pre-clinical as COMPET, have been used. These tests ensure that any LM data that accomplishes the standard requirements can be processed and reconstructed by this Toolbox.

The thesis is structured as follows: Chapter 2 presents a theoretical background defining and describing the concepts of PET and the methods used in this work for the simulation of the acquisition of projection data, simulation of noise, modelling the projected data into sinograms and the correction, processing and reconstruction from parallel projections of the original image for 2D or 3D. Then, Chapter 3 presents the implementation of the Toolbox for achieving the thesis work objectives. Chapter 4 explains computations based on virtual data and experiments using real data from COMPET and Inveon scanners that have been processed to demonstrate the behaviour of the Toolbox created. The results obtained from the simulations and experiments are given in Chapter 5. The Chapter 6 discusses about the results obtained during the thesis work. Finally, Chapter 7 presents conclusions and future research.

2. POSITRON EMISSION TOMOGRAPHY

Positron Emission Tomography (PET) is a nuclear medical imaging technique based on the visualization of functional processes inside the patient body in cross-sections or planes [25]. These sections combined, ordered and processed produce 3D figures where different functional processes can be detected. This is due to the uptake of radionuclides by organs and tissues.

The numbers of medical applications for this technique are extensive due to the possibility to conduct functional and molecular imaging processes in vivo and the capability to be used in different fields.

The oncology field is the most used and developed for this technique. This imaging modality allows the detection of metabolic alterations in tumour cells, common for neoplastic cells. This detection allows the diagnosis, calculation of the stage and restaging various cancer types as lymphoma, melanoma, colorectal cancer, lung cancer and breast cancer [26].

In neurology, PET allows to estimate the metabolism, quantification of blood flow and receptor binding. These facts permit to improve the diagnosis, calculate the disease progression and the treatment response of some neurological disorders as parkinsonian syndromes, epilepsy, dementia, tumoral cells in the brain or ischaemic stroke [27].

Other medical fields as cardiology or psychiatry use this imaging technique to detect pathologies or diagnose infections [28]. In the future, this modality is intended to be commonly used due to the low radiation dose and the quality of the images. For that reason, the actual state of the investigations is not only focused in improving the sensitivity and quality of the image modifying the hardware of the PET scanner. It is also attempting to reduce one of the main limitations and disadvantage, the high cost for the radiotracer technologies production [29].

2.1. Physics of PET measurement

The detection and measurement of radiation is based on the interaction of radioactive particles with the matter. The objective of a Tomographic Imaging System (TIS) is to detect the amount of ionized radiation emitted by the radionuclides inside the object to be measured.

In order to achieve this aim, first the subject has to be positioned inside the FoV of the PET scanner. An injection of a tracer compound with positron emitting radionuclide has to be injected to the patient. The radionuclide is incorporated into biological compounds used by the body known as radiotracers. The selection of the radionuclide depends on the metabolic activity to be detected. When the nuclide decays, due to the excess of protons, a proton inside the nucleus is converted into a neutron, neutrino and positron. The positron travels among the tissue for a short distance losing the kinetic energy until almost all the energy is dissipated. This distance depends on the isotope and tissue but normally is less than 1mm [30]. At this moment, the positron is able to interact with an electron producing annihilation. As it can be seen in Figure 2.1, each annihilation produces two photons of 511 keV that travel in opposite directions.

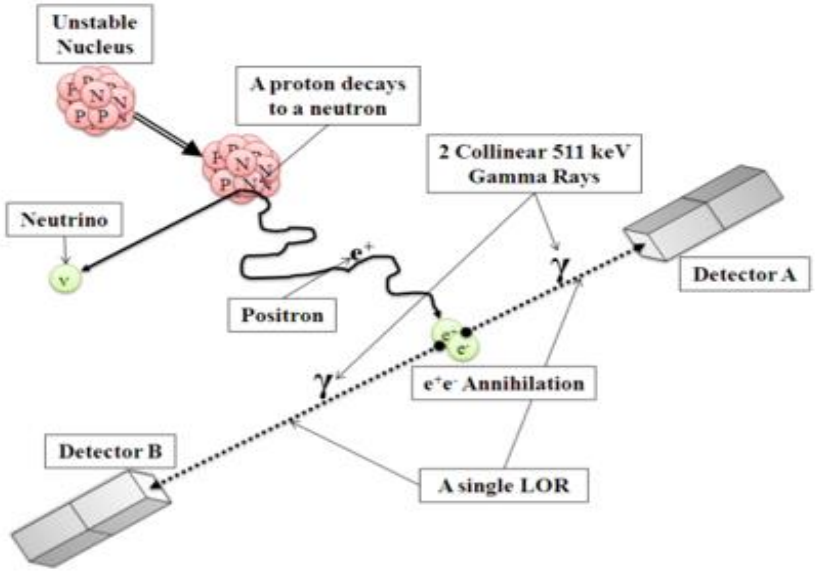


Figure 2.1. *Creation of γ ray process and detection by scintillators around the FoV [31].*

The detectors are placed radially forming a ring around the FoV of the scanner. To generate 3D images more than one ring laid near each other are needed. One of the innovative characteristics of the novel prototype under development is the configuration of the scintillator crystals, instead to be located radially are axially, aligned parallel to the axis of the scanner. This configuration has as major advantage the independence between the sensitivity and spatial resolution.

The γ radiation interacts with the detectors emitting scintillation or flashes of light after absorbing the photons detected at certain Time of Flight (ToF). The ToF correlates two photons detected at certain time difference into the annihilation position in reference with the Field of View (FoV) centre. The detected photons are determined as *'true events'* because they are supposed to come from the same annihilation. The coincidence events are linked by a virtual Line of Response (LoR) at 180° from each photon [32].

The light produced is converted into electrical pulse or signal by a PhotoMultiplier (PM) tube [33]. The pulse is processed to improve the result and reduce the external interferences from the signal. This is done by a linear amplifier and a Pulse Height Analyser (PHA) to convert the signal into counts [34].

The information obtained is converted into the projection domain using the Radon Transform. Noise and imperfections affect the projected data. The acquired data is saved into sinograms, which are processed until are reconstructed in spatial domain images. This process can be seen in the Figure 2.2.

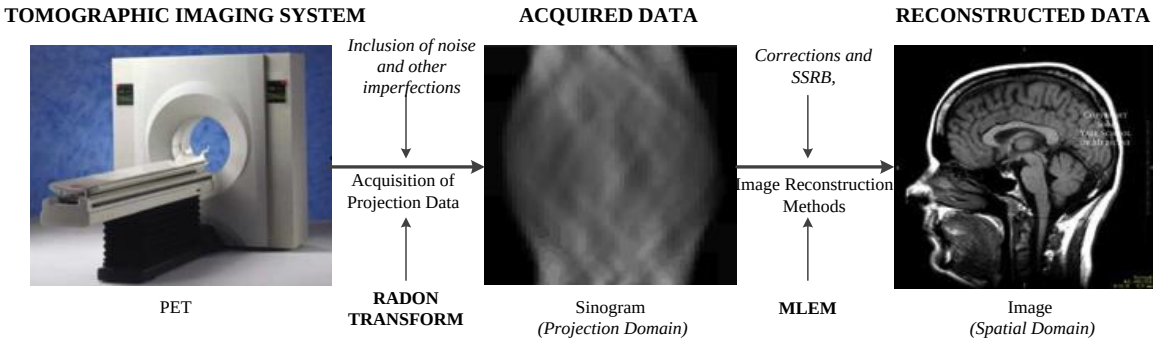


Figure 2.2. *PET imaging technique process [35].*

2.2. Measuring the coincidence annihilation events

The collections of the photon pairs, events between detector pairs, are stored in files. This information has to be processed in order to reconstruct the image scanned. To achieve this purpose, the values stored in the files have to be first converted into sinograms using a parallel projection algorithm.

For the development of this thesis, numerical models have been used to test the functions as Shepp-Logan phantom and images captured with other PET scanner, as Inveon and COMPET.

In this section the models and methods used for obtaining of the sinograms are going to be explained in more detail.

2.2.1. List-Mode file

The acquired data in PET is stored in two main ways: using List-Mode (LM) reconstruction algorithms or by histogramming events into a sinogram during the acquisition process. The second method, histogram-mode data acquisition is a static method that limits some parameters of the sinograms making their modification impossible once the acquisition process is finished. Additionally, this rigidity in the data has an economical effect, each frame will have an unmodifiable information content and cost [36]. For this reason, in the actual PET scanners a LM based reconstruction method is predominantly used [37].

A LM file is a flexible format to store the data. According with the objective of the study, this information will be further processed and reconstructed using different algorithms and statistical methods.

The data is composed by the coordinates of the coincidence events in a sequential data stream or random order. In addition information as the scattered, attenuated or normalized photons or time and energy of each event can be stored for future data corrections.

When the number of coincidence events is low, the resolution of the scanner cannot be appreciated. On the other hand, when the number of events increases, the sensitivity of the scanner can be determined.

2.2.2. Radon Transform

The Radon Transform (RT) is an integral transform used in image processing for the reconstruction of tomographic medical scans. According to Peter E.Valk it can be defined as 'the standard parameterization of 2D and 3D PET data into sinograms' [38]. The RT maps the Cartesian rectangular coordinates x and y to a distance r and an angle ϕ. The value r is the distance between the LOR and the center of the coordinate system, and the angle ϕ determines the orientation of the LOR.

The application of RT to an image I(x,y) can be interpreted as the collection of the projected image along a given set of angles, as for example between 1º to 180º. The result is the sum of intensities for the pixels in each direction, creating a new image R(r, ϕ) where the position of the events is in the sinogram space [39].

The equation (1) represents this transformation, where δ is the Dirac delta function.

$$R(r,\phi) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\delta(r - x\cos\phi - y\sin\phi)dxdy \quad (1)$$

Following Figure 2.3 depicts the RT of I(x,y) based on parallel projections. The representation of the object will be different according with the angle and subsequently the values obtained in this direction. The values represent the number of emissions that occur along these lines for each angle. When the values obtained are represented in the projection domain, the image turns into a sinogram.

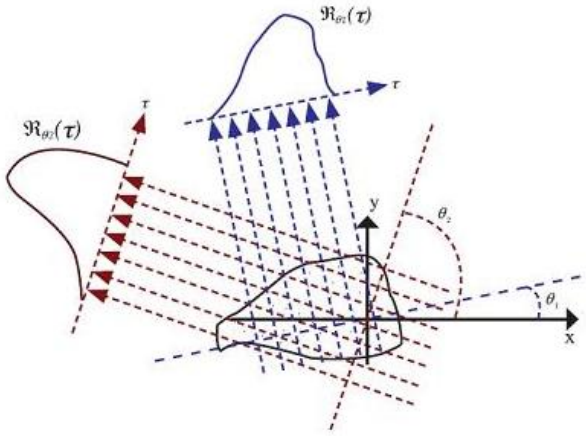


Figure 2.3. *Radon Transform of I(x,y) based on Parallel Projections [40].*

2.2.3. Sinograms

According to Turkington [41] a sinogram is a method of storing and viewing raw PET data. A sinogram is a representation in 2D of the information obtained by the projections applied during the RT. In other words, it is a data array of the angle versus the displacement. The sinograms are composed of sine waves which represent the object scanned during the tomographic process. When the object is more complex, the sine waves can overlap each other containing the data information.

The Figure 2.4 shows a CT data obtained for different angles after the acquisition procedure, displayed as a sinogram. The information contained in the sinogram is converted into an image of the object scanned using reconstruction algorithms. In this case, the result is an abdomen cross-sectional image of the subject submitted to the CT scan.

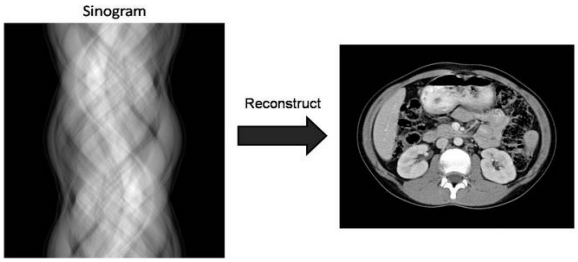


Figure 2.4. *Reconstruction of a CT Sinogram into an object [42].*

2.2.4. Imaging evaluation techniques

The imaging evaluation techniques are based on phantoms created for the study and performance evaluation of methods and algorithms due to the result knowledge. This facilitates the visualization of the behaviour before and after the application of different techniques and image reconstruction algorithms. The phantoms improve quantitatively and qualitatively the evaluation of the techniques applied.

The Shepp-Logan (SL) phantom is a test method based on a numerical model. The SL phantom represents a plane of a human head containing different shapes and sizes inside. This is appropriate to detect the accuracy and resolution of the reconstruction methods. Usually, this method is used for tomography, because it generates an image to test the accuracy of 2D reconstruction algorithms[43].

Another phantom that can be used for the same purposes is the Shepp-Logan Modified. The difference with SL phantom is that the intensities on the phantom have been modified to produce higher contrast. This effect improves the resolution of the phantom resolution.

2.3. Histogramming positioning events

Histogramming is the representation of the detector coordinates within the sinogram. In order to obtain the coordinates of the detected events inside a PET scanner, some trigonometric equations need to be calculated.

Following Figure 2.5 depicts the field of views of the scanner in green, and detector rings position in black. When an event is detected by the rings, a process has to be followed to calculate the scanner coordinates related to the event in the scanner.

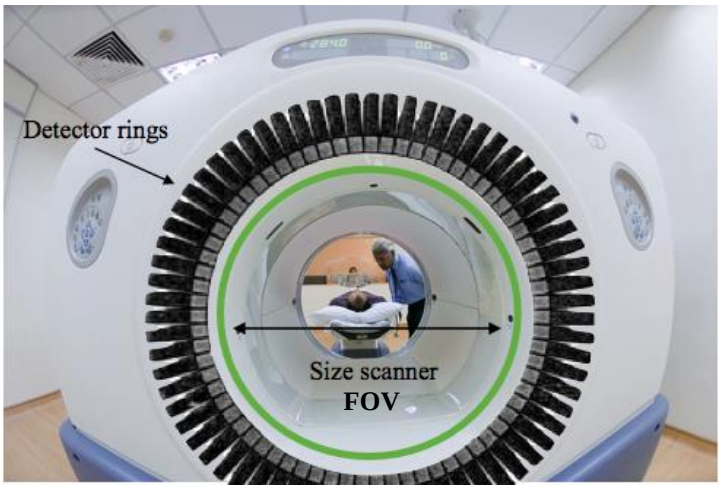


Figure 2.5. *Position of detector rings and scanner FOV for a PET device.*

In the following sections, we are separate the process into two parts: from event coordinates to sinogram, and from the sinogram bins to event coordinates. The second part is needed because because when there is not available data, as in the first stages of the PET development, virtual data is used to test the program performance.

2.3.1. Histogramming from coordinates to sinogram domain

If an event has been detected by the rings, the LOR will link the coordinates of emitted radiation counts detected at the same time, red points in the Figure 2.6. To transform these values into sinogram bins, the radial and angular positions must be calculated. The *radial position, ro,* is the perpendicular distance between the LOR and the point designed as centre of the scanner. On the other hand, the *angular position, ϕ,* is the angle between the LOR in relation with the coordinates axis. These definitions can be represented with the formulas (2) and (3) correspondingly, see Figure 2.6.

$$r_o = r_a \sin \alpha \tag{2}$$

$$\phi = \tan^{-1}\left(\frac{Y_a - Y_b}{X_a - X_b}\right) \tag{3}$$

For the angular approach (3), all the values are known but not for the radial position (2). We can use the values X_a and Y_a to calculate r_a, being the hypotenuse of the triangle formed by the coordinates which base X_a and height Y_a.

$$r_a = \sqrt{{X_a}^2 + {Y_a}^2} \tag{4}$$

The next unknown value to be calculated is α. This can be solved by estimating the values of the angles belonging to the triangle formed by the LOR, the perpendicular line parallel to y axis and the transection with x axis. The sum of the mentioned angles is 180 degrees. The estimation can be seen in the equation (5):

$$\alpha + \beta + \phi + 90 = 180$$

$$\alpha + \beta = 180 - \phi - 90$$

$$\alpha = 90 - \beta - \phi \tag{5}$$

The angle ϕ is a value known in contraposition with β which correspondence is:

$$\beta = \tan^{-1}\frac{X_a}{Y_a} \tag{6}$$

Substituting (3) and (6) into the equation (5) can be solved the value of α which depends now on the known coordinate values.

$$\alpha = 90 - \tan^{-1}\frac{X_a}{Y_a} - \tan^{-1}\left(\frac{Y_a - Y_b}{X_a - X_b}\right) \quad (7)$$

Replacing the value of α obtained from the equation (7) and r_a from (4) into (2):

$$r_o = \left(\sqrt{{X_a}^2 + {Y_a}^2}\right)\sin\left(90 - \tan^{-1}\frac{X_a}{Y_a} - \tan^{-1}\left(\frac{Y_a - Y_b}{X_a - X_b}\right)\right)$$

Will give as a result:

$$r_o = \left(\sqrt{{X_a}^2 + {Y_a}^2}\right)\cos\left(\tan^{-1}\frac{X_a}{Y_a} + \tan^{-1}\left(\frac{Y_a - Y_b}{X_a - X_b}\right)\right) \quad (8)$$

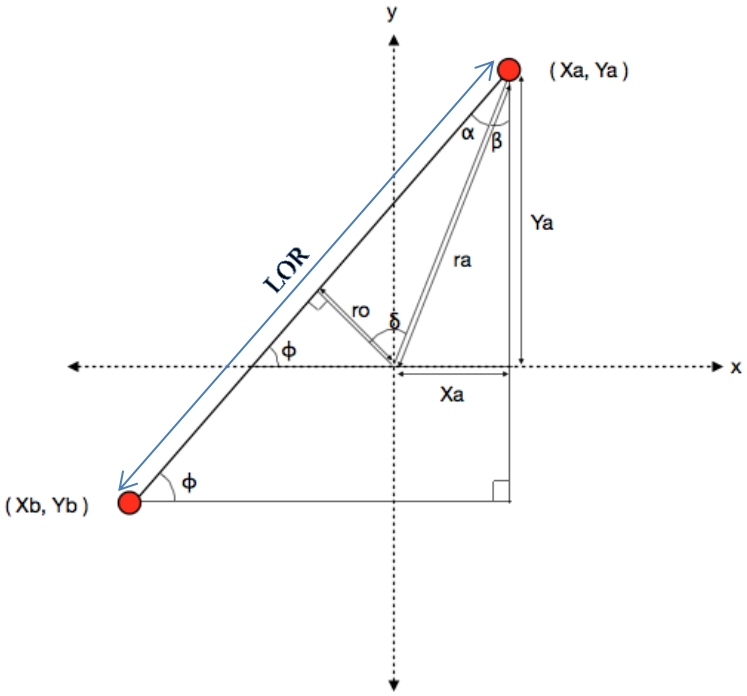


Figure 2.6. *Coordinate system to transform detector coordinates into sinogram bin coordinates.*

This is the trigonometric way to obtain the radial and angular position of each event in the sinogram. However, there is a second manner to calculate these positions: by using the vectorial approach.

The vectorial approach is based on the distance from a point to a line, where the line is the LOR, see Figure 2.6, the point is the centre of coordinates and the distance is the perpendicular from the point to the line, **ro**. The distance of the LOR from a point (9) is the vector **a** (10), that is the difference between a reference point in the line, X_a and Y_a, that can be represented as P_1 (11), and another point which lies on the line, P_2 (12).

$$d(P, LOR) = (\vec{a} - P_3) - \left[(\vec{a} - P_3) \cdot \overrightarrow{u_{LOR}(P)}\right] \cdot \overrightarrow{u_{LOR}(P)} \quad (9)$$

$$\vec{a} = P_1 - P_2 \tag{10}$$

$$P_1 = (X_a, Y_a) \tag{11}$$

$$P_2 = (X_b, Y_b) \tag{12}$$

The unit vector **u** of the LOR for a point (13), is a vector which magnitude is equal to 1. In this specific case, the unit vector can be simplified (14), because the point P_3 with coordinates X_3 and Y_3, is defined to be at the centre of the coordinates.

$$\overrightarrow{u_{LOR}(P)} = \frac{|(X_b - X_a)\cdot X_3 + (Y_b - Y_a)\cdot Y_3 - [(X_b - X_a) + (Y_b - Y_a)]|}{\sqrt{(X_b - X_a)^2 + (Y_b - Y_a)^2}} \tag{13}$$

$$\overrightarrow{u_{LOR}(O)} = \frac{|(X_b - X_a) + (Y_b - Y_a)|}{\sqrt{(X_b - X_a)^2 + (Y_b - Y_a)^2}} \tag{14}$$

The distance vector **ro** (15) is obtained simplifying the equation (9) for the case where P_3 is (0,0) . Substituting the equation (10) at the equation (15) the distance can be divided into the components $\vec{\imath}$ (16) and $\vec{\jmath}$ (17).

$$d(O, LOR) = \vec{a} - \left|\vec{a} \cdot \overrightarrow{u_{LOR}(O)}\right| \cdot \overrightarrow{u_{LOR}(O)} \tag{15}$$

$$dx(O, LOR) = X_a - \left|X_a \cdot \overrightarrow{u_{LOR}(O)}\right| \cdot \overrightarrow{u_{LOR}(O)} \quad \vec{\imath} \tag{16}$$

$$dy(O, LOR) = Y_a - \left|Y_a \cdot \overrightarrow{u_{LOR}(O)}\right| \cdot \overrightarrow{u_{LOR}(O)} \quad \vec{\jmath} \tag{17}$$

Finally, to obtain the value of the distance from the LOR to the origin, **ro** (18), the magnitude has to be calculated applying the modulus to the orthogonal and rectangular vectors components $\vec{\imath}$ and $\vec{\jmath}$.

$$r_o = d(O, LOR) = \sqrt{\left(dx(O, LOR)\right)^2 + \left(dy(O, LOR)\right)^2} \tag{18}$$

To solve this equation, the result of the equation (14) has to be added to the equations (16) and (17) and substitute them into **ro** (18).

In order to calculate the orientation one has to define in which quadrant the event is detected, see Table 2.1. For that purpose, depending on the values obtained in $dx(O, LOR)$ and $dy(O, LOR)$, the position of the event can be calculated with the equation (18).

Table 2.1. *Defining orientation of the radial position magnitude, ro.*

dx(O,LOR)	dy(O,LOR)	Quadrant	Sign
>0	>=0	1st	+
<=0	>0	2on	+
<0	<=0	3rd	-
>=0	<0	4th	-

Notice that these two methods, trigonometric and vectorial approaches have been used in this work. The vectorial approach has been used for the *radial position* calculation. The properties that provide this method, calculating the magnitude, orientation and positioning of each event, are valuable characteristics in order to avoid misplacements or the unwanted mirroring effect. The trigonometric approach has been used in any other case as for example to calculate the *angular position* ϕ.

2.3.2. Histogramming from sinogram domain to coordinates

There are cases when the radial, r_o, and angular value, ϕ, are known and the objective is to obtain the coordinates *X* and *Y*. This occurs when the values are sinogram positions that have to be converted into LM coordinates.

As it can be seen in the Figure 2.7.A, using the *Pythagorean Theorem* to approach the values *X* and *Y*, the value of r_o, stated in the equation (18), is equal to the sum of the areas of the squares whose sides are the two cathetus.

$$r_o = x \cdot \cos\phi + y \cdot \sin\phi \tag{19}$$

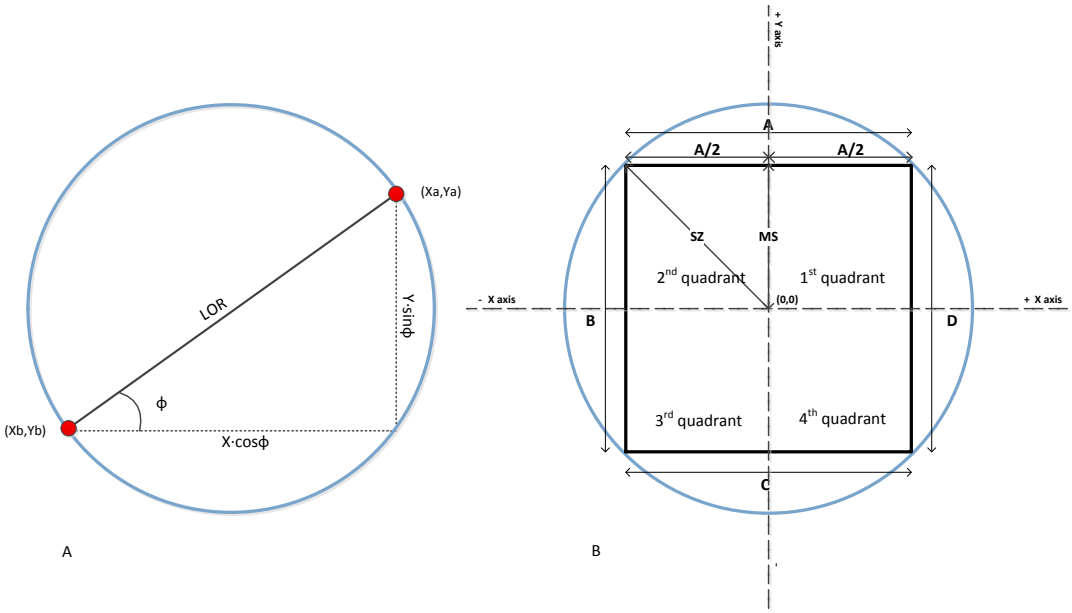


Figure 2.7. *A) Simplified model for detector ring and detectors. B) Cartesian coordinate system whose origin is in the middle of field of view (FOV).*

At this point it is important to determine the reference axis according with the geometry of the scanner. In our case, the origin for the Cartesian coordinate system is in the middle of the scanner as can be seen in the Figure 2.7.B. The coordinates maximum value has to be limited according with the scanner inner space. If the scanner has a FOV where A=B, see Figure 2.7.B: *MS* (20) is the minimum distance from the center, and the maximum value, *SZ*, is the furthest point from the center (21). The value *SZ* corresponds with the radius of the scanner FOV.

$$MS = \frac{A}{2} \tag{20}$$

$$SZ = \frac{FOV \cdot \sqrt{2}}{2} \tag{21}$$

Isolating one of the unknown values, *X* or *Y*, and delimiting the maximum distance possible (21) to the other, the coordinates for each event can be obtained. This asserts a relation between the scanner space and the sinogram to obtain the corresponding LM coordinates. The last detail to take into account is to define the direction of the events before applying the equations (19), (20) and (21). Notice that in the equations (16), (17), (18) and the Table 2.1, the vector model has been applied. In the previous method, to convert the coordinates into bins the magnitude and direction must be defined. In this case, to obtain satisfactory results in concordance with the previous calculations and to avoid misplacements or unwanted effects, the direction of the magnitude has to be calculated. As can be seen in the Table 2.2, for each pair of events some magnitudes of X_a,

X_b, Y_a or Y_b are fixed. These values depend on the scanner FOV limits and the pair activated. For example, if one event has a LOR that reaches the crystals allocated in the segments *B* and *D*, the values known are from *X coordinate axis*, X_a and X_b. The denotation depends on the relation of these points with the coordinate axis, being *B* in the negative part of *X* and *D* in the positive of our coordinates system, see Figure 2.7.B.

Table 2.2. *Detector coordinates direction correspondence.*

Determined values / Pair	Value1	Value2
B - D	$Xa = -SZ$	$Xb = SZ$
A - D	$Ya = SZ$	$Xb = SZ$
A - C	$Ya = SZ$	$Yb = -SZ$
D - C	$Xa = SZ$	$Yb = -SZ$
B - C	$Xa = -SZ$	$Yb = -SZ$
A - B	$Ya = SZ$	$Xb = -SZ$

2.3.3. Histogramming the 3D case

The 3D PET mode represents the object placed inside the scanner FOV in three dimensions. The result has higher sensitivity than the 2D case and lower statistical noise. The difference resides in the planes inclination. In the 2D case, the information acquired is the line integral data for all the planes perpendicular to the scanner. These planes are called direct planes. In a 3D PET scanner, oblique planes are added to the direct planes. The oblique planes represent the data that cross the direct planes with a certain inclination.

The preceding formulas can be used for the 2D or 3D cases, although for the 3D case, now that the angular and radial relation with the coordinates is a known value, the distribution among the *z* axis must be calculated. To calculate the plane position to which corresponds each coordinates X_a, X_b, Y_a and Y_b, it has to be calculated the spatial distribution. The Fig.2.8 indicates the spatial position of each sinograms between Z_a and Z_b, (22).

$$Z_o = \frac{Z_a - Z_b}{2} \tag{22}$$

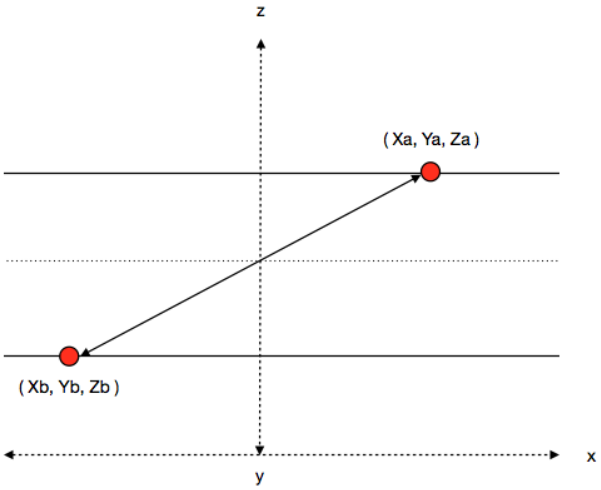


Figure 2.8. *Coordinates in 3D to calculate the position according with spatial distribution.*

The value θ or oblique plane inclination has to be calculated in the 3D case. As it can be seen in the Figure 2.9 is the angle between X_a, Y_a, Z_a with X_b, Y_b, Z_b and the x axis. It can be obtained with the formula below (23):

$$\theta = \frac{|Z_b - Z_a|}{\sqrt{(X_b - X_a)^2 + (Y_b - Y_a)^2 + (Z_b - Z_a)^2}} \tag{23}$$

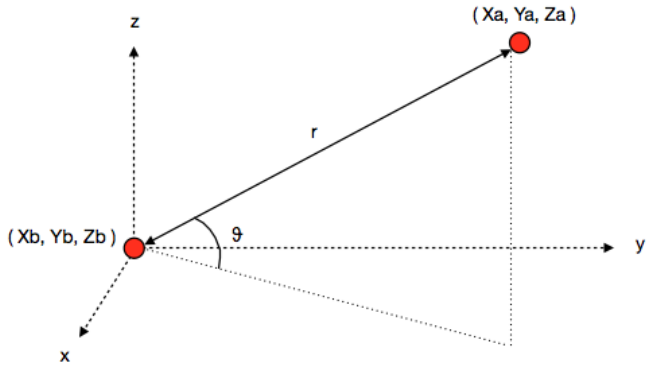


Figure 2.9. *Coordinates in 3D to calculate θ in spatial domain.*

2.4. Further processing of the event information

Due to the use of virtual data to test the functions as SL phantom or SLM phantom, the positions of the events from the sinogram into a LM file are ordered in a sequential manner. In order to create a more realistic result, the events have to be randomly sorted. To achieve that purpose, different techniques can be used as bootstrap or shuffle the events.

The bootstrap approach is a statistical method to determine the reconstruction methods accuracy based on experimental data. The results are as many different replications of the original data as determined [44]. In PET the bootstrap method is used to estimate statistical properties of the PET images. This can be done generating statistically different datasets based on a few data samples [45]. The methods used can be nonparametric and parametric sinogram-based and LM based [46]. In our case, this method has been reinterpreted to realize as many data-sets and modify as many positions of events inside the data set as have been selected by the user.

Due to the performance complexity of the bootstrapping tool and the addition of events during the replication process, we decided to shuffle the events inside the data set. The performance complexity increases the processing time needed to obtain accurate results. Also the increment of events during the process adds unnecessary information to the process. Shuffling the events, the positions are modified in a random manner without repeating events in a unique simulated LM data, recreating a real PET data, in shorter time.

2.5. Sampling and Compression modes

The number of samples corresponds with the sinogram bins. This value can be scaled according with a desired value. Furthermore, the number of sinograms can be reduced without losing too much essential information, decreasing the computational time needed to process the data. This can be done using compression methods.

The rebinning methods are compression modes which functions are to reduce the sinogram data-sets in 3D data collection. To achieve this purpose, the oblique projections are added into the direct sinogram data-sets acquiring as a result a compilation of sinograms in 2D. Subsequently, 2D reconstruction techniques, analytical or iterative, can be used to each sinograms for the image reconstruction in the spatial domain. The selection of the rebinning algorithm method is important because it affects to the resolution and noise of the reconstructed image [47]. In our case, the Single Slice ReBinning (SSRB) method was chosen.

The SSRB algorithm [48] is based on a basic approximation with acceptable accuracy. In this algorithm, an oblique LOR can be projected to the direct plane that is between two points of such LOR. Hence, the events from oblique sinograms are repositioned to the transaxial sinogram situated axially in the middle of the LOR. This process can be followed in the Figure 2.10. When an event is detected in a multiring scanner, it will be situated in a transaxial or an oblique sinogram bin according to θ value. After, the rebinning algorithm is applied to decrease the number of sinograms without losing substantial information. At this moment, the information is ready to be reconstructed into a 3D image.

This approximation is acceptable when the events are near the middle of the scanner axis and for small features [49]. Therefore, this method is satisfactory for the purpose of this thesis.

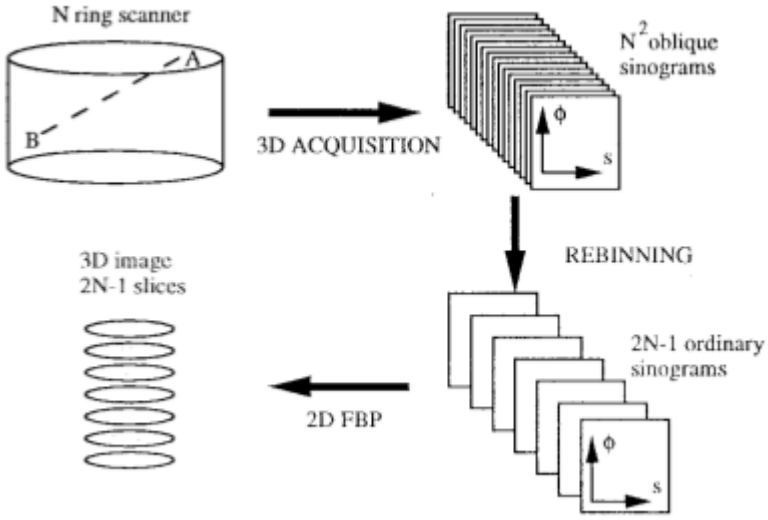


Figure 2.10. *Rebinning algorithm representation for multiring scanners [50].*

2.6. Physical effects

During the acquisition process some external disturbances influence the results. For that reason it is necessary to apply some correction methods. In order to apply these correction methods some physical effects detected during the process are needed: factor of attenuation, normalized data and an estimation of the scattered events.

Firstly, the scattered events are the events that belong to the same annihilation but for different reasons lose energy and the direction is modified. Therefore, they can be linked by the scintillator detectors to be from the same annihilation. However, the LOR is not the correct one becoming as a false 'true event' [51]. On the other hand, the attenuation is an effect that represents the decrease of the gamma ray value. The energy produced in the annihilation is partially lost when the positron travels across the body and interacts with the tissues [52]. Finally, the normalization correction is based in the exposition uniformly of all the detector pairs to a 511 keV photon source without a subject inside the FoV [53].

In our case, the equation used (24) shows that the true events are multiplied by the normalized bins and the scattered photons are subtracted. The result obtained is multiplied by the attenuation factor obtaining a corrected sinogram.

$$Corrected = (Trues \cdot Normalization - Scattered) \cdot Attenuation \qquad (24)$$

The random events are another physical effect which has to be corrected. The random events occur when two gamma rays from different annihilations have a coincidence timing window, range of time. The correction of the random events is already done by the modern scanners using a technique called delayed coincidence timing window. This technique saves the events with a coincidence timing window and a probability of true events zero [54]. The result is subtracted to the true events.

2.7. Image reconstructions from sinograms

The 2D sinograms obtained after the application of the SSRB algorithm can be reconstructed: by analytical reconstruction methods, as Filtered BackProjection (FBP); or using iterative reconstruction methods, as Algebraic Reconstruction Technique (ART) or Maximum Likelihood Expectation Maximization (MLEM).

The MLEM algorithm estimates the Maximum Likelihood (ML) in the presence of missing or hidden data. The ML algorithm can obtain a relation between new parameters and the actual information. This means that after each iteration the data that is most likely to pertain to a sector of the image is going to be calculated. This iteration is done in two steps. The first step is to estimate the hidden parameters given the observed and actual data. Now the missing data is "known" and the ML is applied. The likelihood will increase after each iteration which means that is going to converge [55].

This method is based on stochastic variations in photon counts. For that reason it is used in medical devices that acquire images by photon emissions due to radionuclides as PET or SPET. This algorithm has as objective to calculate the radionuclide image detected by the external sensors [56].

To obtain images with higher quality, the number of iterations has to increase as can be seen in the Figure 2.11. Moreover, if some noise is added to the image when the number of iterations increases, the noise is enhanced and the image obtained is having typical check-board noise.

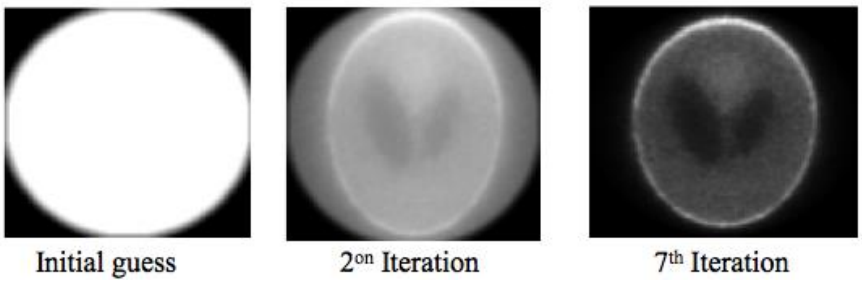


Figure 2.11. *Phantom reconstruction based on MLEM algorithm.*

3. TOOLBOX IMPLEMENTATION

In this section, the functions used by the Toolbox *'PETSimulator'* are going to be explained. This Toolbox has been developed to simulate the behavior of a PET device. The *'PETSimulator'* can be divided in three parts: generation of a LM data from virtual data, histogram LM data-sets, and correction, compression and reconstruction of sinograms. The main property and advantage of the Toolbox is the modularity of the integrated functions. The functions can be used individually or as a block depending of the user needs.

For the generation of LM data the function creates LM data-sets with the same structure as in real cases, based in a SL phantom. Other standard test images can be used to simulate images of an object inside the PET scanner FOV.

In the case of histogram a LM, the function can read 2D and 3D LM data-sets of different formats and calculate the sinogram bins positions: radially, among the x axis; angularly, y axis; and spatially, in the z axis.

The third block of our Toolbox corresponds with the sinogram correction, compression and reconstruction. This program uses different methods to correct, compress and reconstruct the sinogram bins angularly, radially and spatially. The objective is to obtain clearer images for better diagnostic or further investigations.

3.1. Generation of a List-Mode data

The first function of this Toolbox is a function that generates LM data files. The computational process, see Fig. 3.1, starts with the introduction of the parameters by the user, until obtaining of a LM file of events. The LM data set is based on the phantom image for the 2D case and for the 3D case by a three-dimensional image generated by a special function.

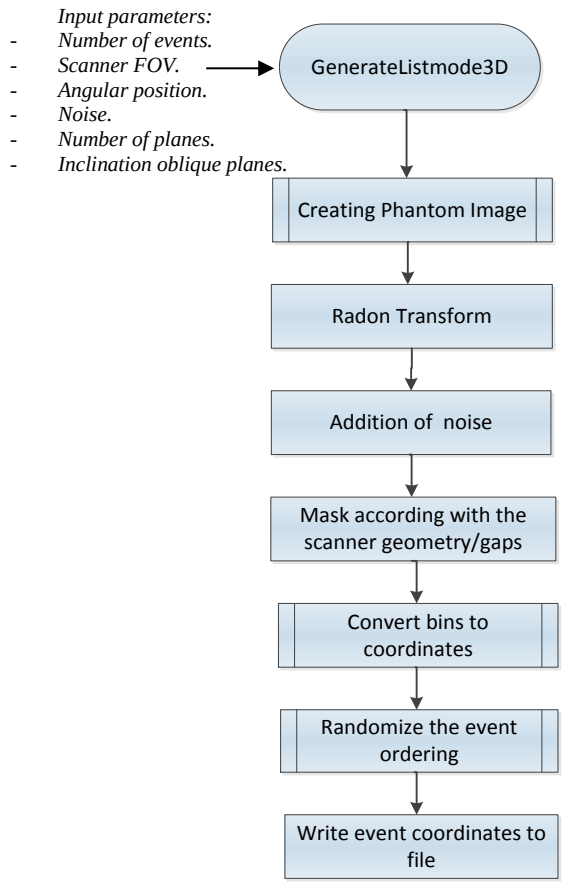


Figure 3.1. *Block Diagram of the function GenerateListmode3D.m.*

In order to obtain these results, the function has to be called using the forthcoming line:

```
[Nangbin,slices]=generateListmode3D( Nevents, Npix, Nang, noisy,
                                planes, incl )
```

The user has to introduce some parameters:

- *Nevents*, the total number of sinogram events.
- *Npix*, the scanner FOV size.
- *Nang*, the angular position of the sinogram.
- *noisy*, the sinogram interferences.
- *planes*, number of planes and mode of activation, 3D or 2D image.
- *incl*, oblique planes inclination.

After the user has given a numerical value to these variables, the remarkable process steps can be seen in the Figure 3.2. To understand these results, the process is going to be explained in detail.

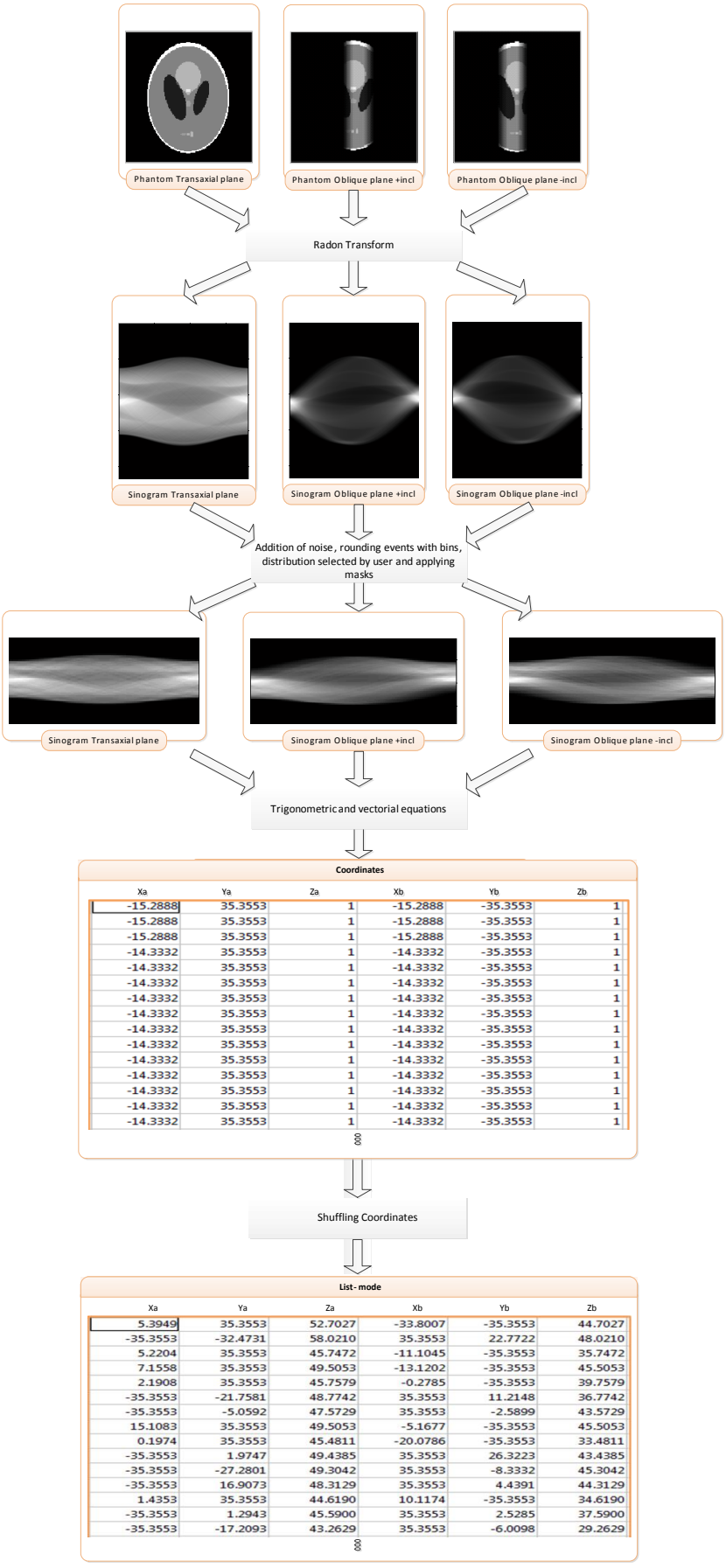


Figure 3.2. *Schematic of the function 'generateListmode3D'.*

One important parameter is the selection of the planes by the variable *planes*. This input value determines if the image is in 2D or in 3D. If the number of planes is bigger than 1 then the user wants to simulate a 3D image, otherwise, the LM obtained is going to be for a 2D simulation.

In the 2D case, the SL phantom is simulated by the Matlab function *phantom* and the RT is applied to obtain the sinogram. Moreover, in the 3D case the process needs some additional steps. First a 3D image has to be created by a subfunction. After this a RT is performed according with the angular position selected by the user. In a real 3D case there are oblique planes to give spatial information to the image. To simulate this case, $incl \neq 0$, a subfunction that calculates oblique planes and simulates different positions of the planes has to be used. Before applying the function that calculates the sinograms, the image planes must be ordered.

When the user selects to simulate the 3D case, it must be known that the total number of planes is going to differ from the one selected. The value corresponds with the result of this equation, *n=(planes+2·planes)*, because there are 2 oblique planes for each transaxial plane. The planes are ordered according to the Eq. (25) is: first, the transaxial sinograms are going to be positioned, from one to *p*. The value of *p*, corresponds with the total number of planes selected by the user. In the second block of planes, the oblique planes with positive inclination are going to be placed in the same order as the transaxial planes. This relation can be seen in the table of correspondences, Table 3.1. The last block of planes is for oblique planes with negative inclination, for correspondence see Fig.3.3. The order depends on the spatial position of the planes.

$$P(z > 0) = z_{transax}(1{:}\,p) + z_{oblique+}((p+1){:}\,2\cdot p) + z_{oblique-}\big((2\cdot p)+1{:}\,end\big) \quad (25)$$

This order has been selected according with real data obtained by real PET devices. It can be said that this is the sinogram standardized distribution for PET. This standardization contributes with a moldable use of our functions for different data provenance.

Table 3.1. *Correspondence of planes.*

Planes / **N**	**Transaxial**	**Oblique + incl**	**Oblique -incl**
1	1	planes + 1	2·planes + 1
2	2	planes + 2	2·planes + 2
3	3	planes + 3	2·planes + 3
N	n	planes +n	2·planes + n
Planes	Planes	2·planes	3·planes

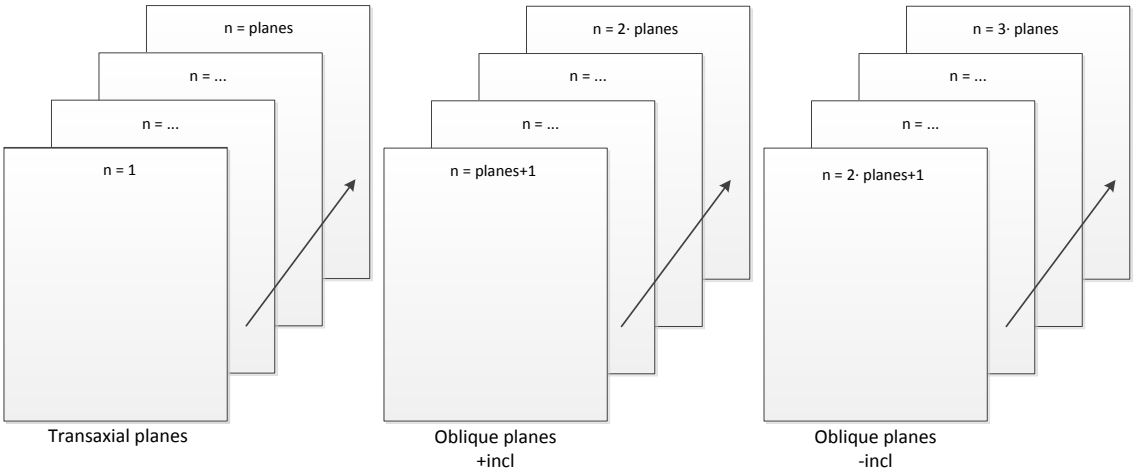


Figure 3.3. *Planes order for the 3D case if there are oblique planes.*

The next step is to calculate the RT. For that a function calculates the image intensity projection for each plane, at certain range of angles along the radial sampling.

In both cases, 2D and 3D, noise is simulated because is a problem that occurs during real data acquisition. If the user activates the variable *noisy*, then noise disturbance is simulated by the Poisson distribution [57]. The function that uses the Poisson distribution generates random numbers that are incorporated to the sinogram as noise. Otherwise, the numbers of events in the sinogram are rounded to ensure that is the same value as the number selected by the user with the variable *Nevents*.

After this process a mask is created, *Smask*. This mask counts the sinogram bins with values different from 0. This mask is saved as well as the sinogram matrix, *Sino*, in a *.mat* and *.txt* file. The mask is used to obtain the sinogram bin positions that need to be converted into LM coordinates. The sinogram matrix is saved to compare the sinograms that originate the LM data with the sinogram histogrammed from the LM. The difference is calculated with the Mean Squared Error (MSE).

The final step of this function is to convert the bins of the sinogram into coordinates. This process is fulfilled by a subfunction which result is a list of coordinates of all the non-zero sinogram elements saved in the mask.

The coordinates of each event are saved in a .mat and .txt file. The name of the file is showed to the user by the Matlab Command window. In the 2D case, the file has as components 4 columns. The first column corresponds to the coordinate X_a, the second to Y_a, the third to X_b and the fourth to Y_b. On the other hand, in the 3D case the file consist of 6 columns as it can be seen in the Fig.3.3. The first column corresponds to the coordinate X_a, the second to Y_a, the third to Z_a, the fourth, fifth and sixth correspond to X_b, Y_b and Z_b respectively. If there is more than one event for the same coordinate, this coordinate is going to be repeated as many times as events.

At this point the coordinates have been calculated but are in order according with the positions of the sinogram bins. The first coordinate will correspond to the first sinogram bin = 1 and so on. To simulate a more realistic data it is necessary to calculate the statistical properties of the PET data as bootstrapping or shuffling the LM. In this function, bootstrapping is used to to simulate the statistical properties of the PET images. On the other hand, the function that shuffles the events is used to move the LM data events in a random manner.

During the computational process, the program uses a debug mode to inform to the user about actual code state. The debug mode shows messages to the user using the Matlab Command window.

3.1.1. Creation of the 3D phantom

This function is used to generate virtual data based on a numerical model. Usually, the SL phantom, explained in the Section 2.2.4, is represented by a cross-sectional plane. In this case, a 3D phantom is needed to evaluate the functions and generate standardized LM data-sets. The image used for default is a 3D SL phantom.

The result consists in 3 ellipses representing the brain and some features inside. The image is saved inside the parameter *p*, which is a grayscale intensity 3D image of a head phantom.

3.1.2. Creation of oblique planes

The sinograms of a 2D image are transaxial planes. In the case of a 3D image, the information to generate 3D images is contained in sinogram oblique planes. A specific function is used to calculate the oblique planes from the 3D phantom.

The oblique planes are obtained by rotating each transaxial plane of the 3D phantom across x, y and z axis. This is done using the variables *AX*, *AY* and *AY*.

In this Toolbox the planes are rotated related to the x axis. The rotations related with the y and z axis are not needed in the creation of PET oblique planes. This is due to the parameter θ, oblique plane inclination, which is the angle between X_a, Y_a, Z_a with X_b, Y_b, Z_b and the x axis, see Fig.2.9.

This function can also translate the planes along x, y and z axis by the variables *TX*, *TY* and *TZ*. Notice that there is this possibility but these rotations are not required at this moment. The default interpolation method is linear interpolation, but other methods can be chosen as nearest neighbor or cubic, etc.

3.1.3. Calculation of the coordinates in sinogram space

This function is used to calculate the sinogram coordinates which are represented by non-zero bins. In the Fig.3.4, the subfunction block diagram can be seen.

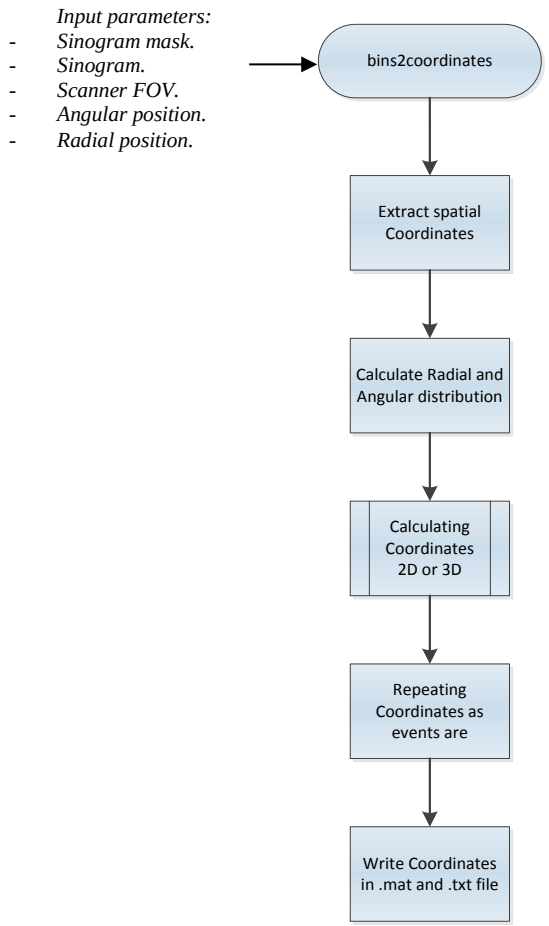


Figure 3.4. *Block diagram of the function bins2coordinates.m.*

First some parameters are given to the function as input values. The *sinogram* and *Smask* are imported and the information is divided in 3 columns: sinogram angle, radial offset and planes number. Once this data is separated, another program calculates and saves the number of events for each sinogram bin.

The next step is to distribute the values angularly, radially and spatially. A relation between the LM coordinates, FOV size and sinogram bins have to be calculated. Once this is fulfilled the coordinates are limited according with the scanner geometry. The result is saved in vectors to be used by the function which calculates the correspondence between the sinogram bins, angle and radial offset, and the LM coordinates, X_a, Y_a, Z_a, X_b, Y_b, and Z_b. This process is done to adapt the sinogram bins of a virtual image into coordinates in a LM format. To convert the sinogram bins into LM coordinates trigonometric equations are used. The equations used can be seen in the section 2.7. The result must be repeated as many times as events are positioned to each coordinate.

The process to create the LM data can be seen in the Fig.3.5. Each sinogram bin has angular, ϕ, and radial, r_o coordinates with different values. In the figure below the value 24 can be seen a depicted. This value corresponds to the number of times that each radial and angular coordinate have to be repeated to represent this bin. Using the equations adapted to the scanner geometry, the coordinates X_a, Y_a, Z_a, X_b, Y_b, and Z_b to the selected bin are obtained. This process is going to be repeated for all non-zero bins.

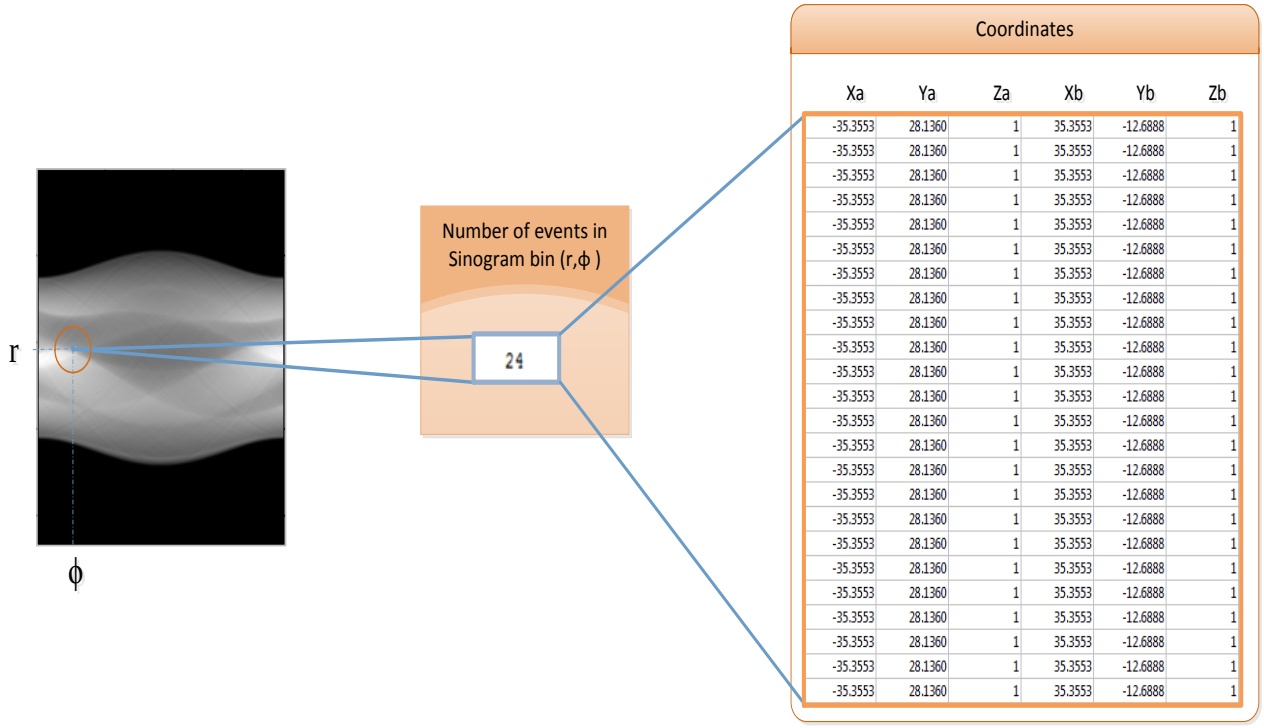


Figure 3.5. *Selection of a sinogram bin and its X_a, Y_a, Z_a, X_b, Y_b,* and *Z_b coordinates.*

The coordinates are saved in a .txt and .mat format file. The name and file extension are showed to the user by the Matlab Command window as an information message.

3.1.4. Calculation of the List Mode coordinates

The conversion of sinogram bins into LM coordinates, for PET LM file generation, is done by a specific function. The process can be seen in the Fig.3.6. In this case, as premise are the sinogram coordinates: angular, ϕ, and radial, r_o, positions of each sinogram bin. Using the equations (3), (19), (21), (22), (23) and the Table 2.2, the sinogram bin positions are converted into LM coordinates X_a, Y_a, Z_a, X_b, Y_b, and Z_b.

There are different possible solutions for each bin. The solutions depend on the detected pair, which is defined in the Table 2.2. The detected pair corresponds with the scanner sides activated by each event. All the possible solutions for X_a, Y_a, X_a and X_b coordinates are calculated. The correct coordinates are the most reliable and logical solutions according with the detected pair.

In the situations when more than one solution is possible, the user is informed with a warning message through the Matlab Command window. Once that the most reliable coordinate values have been selected, the spatial coordinates, Za and Z_b, can be calculated.

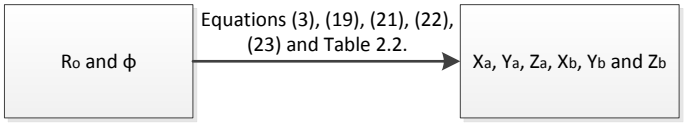


Figure 3.6. *Block diagram of the subfunction 'bins2coord'.*

3.1.5. Randomizing the coordinates positions

The randomization of the coordinate positions can be done bootstrapping or shuffling them. The objective of applying these methods is to get more realistic LM data-sets corresponding to different measurements or realizations.

Due to some modifications during the development of this thesis, it has been decided to use the shuffle function for the LM data randomization. The bootstrapping mode is going to be an optional feature, and it is not going to be used as default mode. Both methods can be used independently of the Toolbox.

The process can be followed in the Fig.3.7. The block diagram shows as first step, to check the extension and import the data. If the bootstrapping tool has been selected, is going to be calculated as many sinograms as selected by the user. The statistical similarity between the original sinogram and the created sinogram is another value that depends of the user selection. The user has to select the percentage of similarity desired. If the user wants a replica of the original sinogram this value will be 0. On the other hand if the desired result is a completely different sinogram, it will be 100. When this process is fulfilled, the original LM file is going to be disordered in a not deterministic manner and saved. This process is done by the shuffling tool. The names of the files are going to be shown to the user in Matlab Command Window by the debug mode.

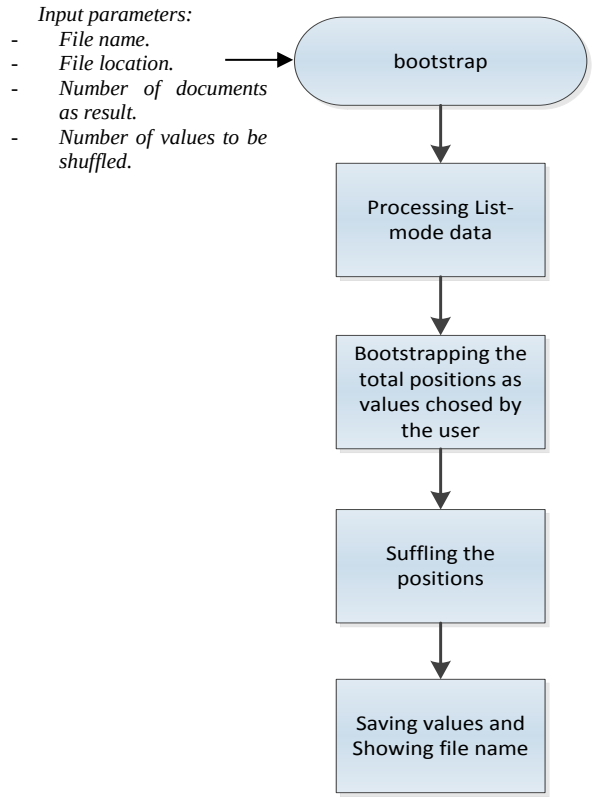


Figure 3.7. *Block diagram of the subfunction 'bootstrap.m'.*

One example of the shuffling tool performance can be seen in the Fig.3.8. After the calculation of the coordinates, the events in the LM file are in a chronological order. This is not a realistic result because the coordinates are repeated sequentially as the bins were collected in the sinogram. In a real PET scanner the events are not going to be detected per sections, neither the photons are going to be emitted in a synchronized order.

With the aim to simulate more realistic results, the LM file has been shuffled to distribute the values randomly. As can be seen in the Fig.3.8, the event selected in the Fig.3.5, which has to be repeated 24 times has been shuffled. After shuffling all the events, the coordinates of the event repeated 24 times are now combined with new events coordinates from different sinogram positions.

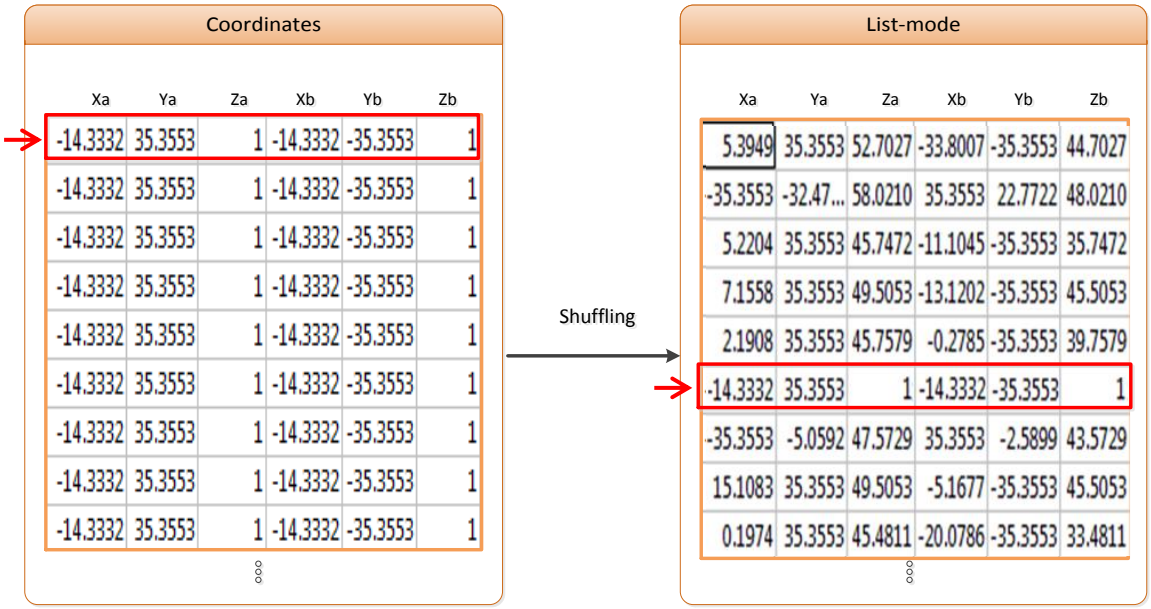


Figure 3.8. *Process of shuffling the coordinates ordered sequentially.*

3.2. Histogramming a List-Mode data

Histogram is a graphical representation of the relative frequency that a phenomenon occurs [58]. In this case, the phenomenon is the number of positrons collected by the scanner detectors. The frequency that an event occurs from one coordinate corresponds with the value of a sinogram bin. For that reason, the process of converting a LM data into sinograms or vice versa is called histogramming. The tool used to achieve this is the main function of the second part of this Toolbox.

The computational process, seen in Fig. 3.9, can be defined as a four step process. First, the user introduces the required input parameters. Then, the function reads the coordinates from a LM data. The typical extensions of the LM files are .txt or .mat. If the user has not information about the FOV, this value can be approximated by the default mode. The second step is to call the subfunction that calculates the range of the sinogram size according with the FOV. Subsequently, a relation between the scanner FOV and the angular and radial position in the sinogram is going to be obtained. Once this is accomplished, the function that calculates the positions of the sinogram bins based on the coordinates is called. Each bin has a spatial component which determines to which sinogram it belongs. This is going to be calculated by a specific subfunction. The final step is to create the sinogram, save the result and print some information through the Matlab Command window informing the user about the location, name and file extension.

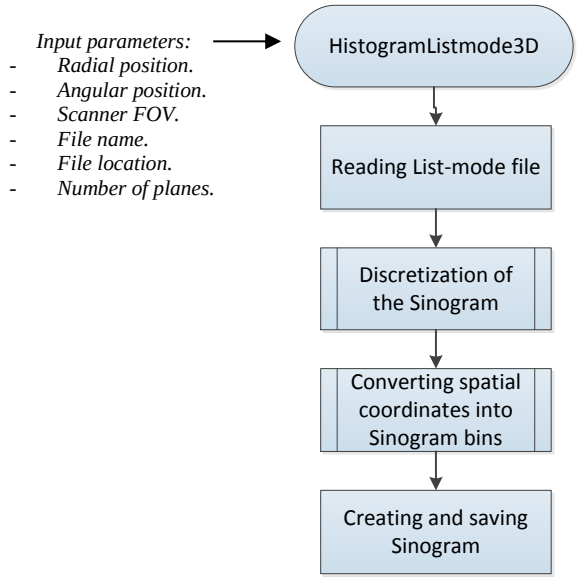


Figure 3.9. *Block Diagram of the function HistogramListmode3D.m.*

In order to start this process the function has to be called using the following line:

```
Sinogram = HistogramListmode3D( Nrad, Nang, NradOld, Nfile, Nlocation
                                planes, obliq )
```

The user has to introduce some parameters as:

- *Nrad*, the radial size of the sinogram to be represented.
- *Nang*, the angular sinogram size.
- *NradOld*, the scanner FOV.
- *Nfile*, the name of the LM file to be read.
- *Nlocation*, the location of this file.
- *planes*, number of planes and mode activation, 3D or 2D image.
- *obliq*, if there are oblique planes.

Note that if the scanner FOV or the total number of planes is unknown information, the user can give the value 0 to these variables and the program will calculate an approximation.

3.2.1. Calculation of sinogram size and distribution vectors

This subfunction has as objective to calculate the range of the sinogram to be histogrammed. To fulfil this purpose, the maximum value of the coordinates based on the FOV is calculated. After this, some parameters are calculated as: a relation between the maximum and minimum values of the LM coordinates, the scanner FOV range, and the sinogram size. Each sinogram, radially and angularly, is going to have the same size, then in this function the total number of planes is not needed.

The maximum and minimum values of the FOV are the result of the equation (21). The results are a positive and a negative value. The positive result corresponds with the maximum value of the scanner FOV. The negative result corresponds with the minimum value of the scanner FOV. This can be seen in more detail in the Fig.2.7, the maximum value corresponds with the positive part of the axis and the minimum with the negative part. The radius of the FOV is the result of the equation (20). Having all these values calculated, the size of each sinogram bin has to be defined. One this has been fulfilled, a relation between the sinogram bins size and the sinogram radial and angular value is stated. The result is saved in vectors which are used as input to a subfunction that calculates the sinogram bins positions. This process is done to adapt the coordinates of any scanner into a sinogram with a size selected by the user.

3.2.2. Calculation of List Mode coordinates from sinogram bins

This function is the core of the program that histograms a LM file. The aim is to transform the coordinates of each event represented in a LM file, into bin positions of a sinogram to be represented. This process can be depicted in the Fig.3.10.

First, the common variables for the 2D and 3D case are going to be calculated. The radial coordinate of the sinogram is calculated by the equation (18), which gives the magnitude, and the table 2.1 to acquire the direction. Moreover, to calculate the angular coordinate, the equation (3) is used and afterwards scaled by $\frac{\pi}{2}$ to obtain the result into $-\pi$ to $+\pi$ range. For the 3D case, besides the above mentioned process, one has to calculate the value of the coordinate z and the azimuthal angle. The z coordinate, z_{mean}, determines the plane position in the set of sinograms and can be calculated using the trigonometric equation (22). The azimuthal angle, θ, is calculated using the equation (23). This value defines the inclination of the oblique planes. The direction is determined positioning the value of theta in the correct quadrant according with the value obtained of z_{mean}.

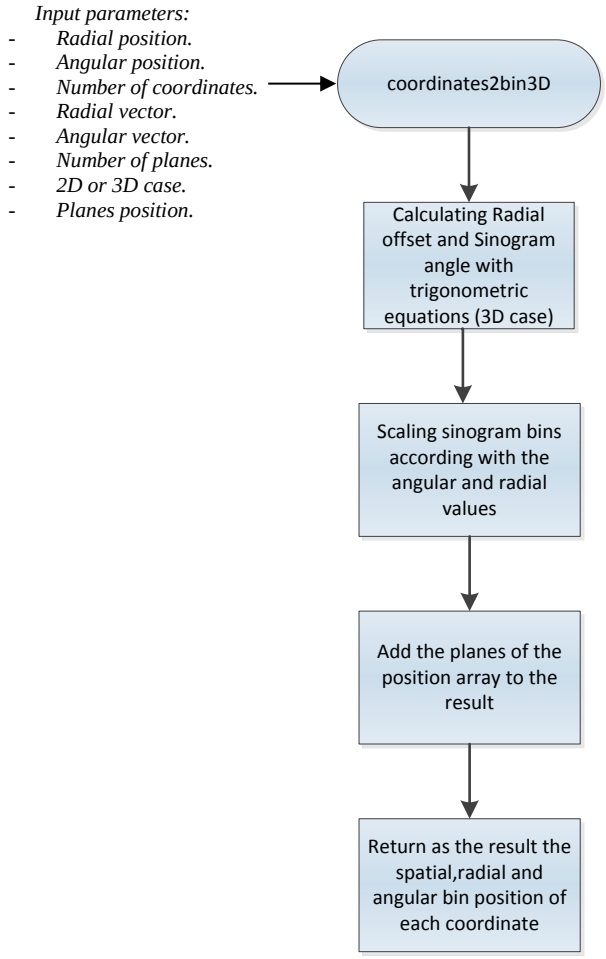


Figure 3.10. *Block Diagram of the subfunction 'coordinates2bin3D.m'.*

Firstly, the events detected in the scanner are converted into radial and angular sinogram values. The next step is to scale the sinogram bins according with the angular and radial values selected by the user with the input parameters *Nrad* and *Nang*. The bin positioning depends on the bin size and the sinogram radial and angular values, which have been calculated previously. The activation of one sinogram bin is decided according to which range of values is closer to the value obtained radially and angularly. In the 3D case, the spatial position of the sinograms is calculated by a specific subfunction.

3.2.3. Calculating the spatial position of the planes

This subfunction has as objective to determine the total number of planes, and position of each sinogram inside this range. For that it has to be differentiated between the case 2D, 3D, and if there are oblique planes.

The process can be seen in the diagram block, see Fig.3.11. First the function calculates the spatial axial position of each plane for the 2D and 3D case. This is differentiated by the value of z_{mean}. If z_{mean} is 0 then the sinograms represents a 2D image. On the other hand, if z_{mean} is superior or inferior to 0 then the information is in 3D. Once this is determined, a relation between the number of planes selected by the user and the values obtained by the equation (22) has to be calculated. This relation is going to determine a new distribution value. This has been done to be able to distribute any LM data according with the user needs. If the user wants to calculate the original size of the planes, it is possible using the default mode. This feature is very useful to deal with large amount of planes or in the 3D case, because the computational time reduces abruptly when the number of planes decreases. Sometimes, in order to have a general idea of the results, it is enough to select one plane. The result is not going to be detailed but the reduction of the computational time is useful in some simulations and experiments. This distribution is done, as in the angular and radial case, calculating the difference between the unitary portions of the spatial distribution defined with respect to z_{mean} and determining which the closest range of values is.

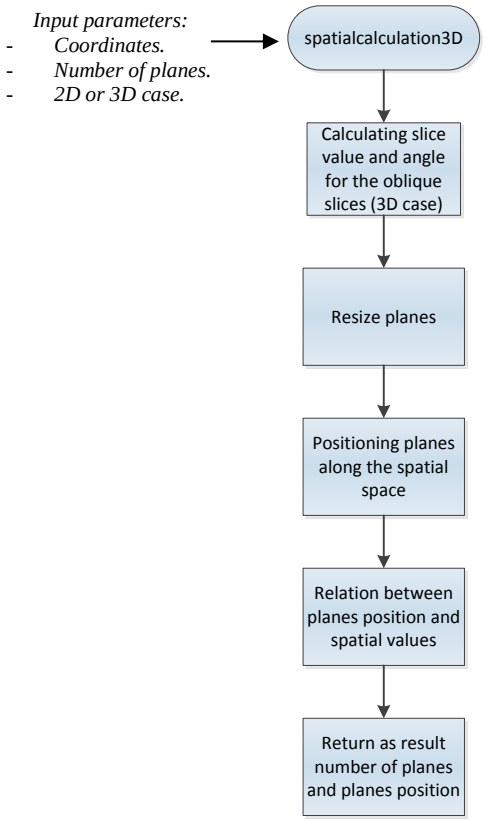


Figure 3.11. *Block Diagram of the subfunction spatialcalculation3D.m.*

After this, in a real 3D case there are oblique planes. To calculate the values of θ the equation (23) is used. One important issue is the allocation of θ values. The value of θ differentiates between transaxial, oblique positive and negative planes. In the case of transaxial planes θ is going to be 0. The oblique planes with positive inclination have a value of θ positive. In contrary with the negative oblique planes, which have a negative value of θ. This value must be checked to ensure the correct placement of the planes. If a plane is misplaced according with the order defined in the Fig.3.3, for example there is a positive value in the transaxial planes section, a new range is going to be calculated with the values of θ and z_{mean}.

During all the processes the user is going to be informed of incidents and new calculations performed by the debug mode of the program. Once a new range of planes is restored, all the information needed has been obtained for the creation of the sinograms.

3.3. Script for Sinogram calculation and representation

This program has been designed to be used as a user script. All the functions explained in the previous sections are applied in this script. This facilitates the use of this Toolbox simplifying the process. Using this function, the user can select if the sinogram is generated from LM file data or if is created by the program itself based on SL phantom. To call the function can be done with the following line:

```
Sinogram = scriptSinogram3D(readfile, createfile, Nfile, Nlocation,
                            varargin)
```

The user has to introduce some parameters as:

- *readfile*, to select if the file is going to be read from an existing file.
- *createfile*, to create a LM file.
- *Nfile*, the name of the LM file to be read.
- *Nlocation*, the location of this file.
- *varargin* are other values needed for the calculation of the functions.

Inside *varargin* are values that can be introduced by the user, or if the user does not have knowledge about them can be used the default values. These values are:

- *Nang*, the angular sinogram size, default = 160.
- *Nrad*, the radial sinogram size, default = 75.
- *NradOld*, the scanner FOV, default = calculated according with the LM coordinates.
- *Nevents*, the total number of sinogram events or resolution, default = 1000.
- *noisy*, the sinogram artifacts, default = 0.
- *incl*, oblique planes inclination, default = calculated according with the LM coordinates.
- *planes*, number of planes and mode activation, 3D or 2D image, default = calculated according with the LM coordinates.

This program is divided in two parts, see Fig.3.12. Firstly, the user selects to read a file or to generate a LM file. Secondly, the user selects to read an existing LM file, the program makes a call to the function that histograms a LM file. The result is a sinogram calculated from LM file data and the parameters selected. On the other hand, if the user

selects to create a sinogram based on virtual data, first a LM data set has to be created based on a phantom model. The coordinates of the sinogram, r_o and ϕ, are converted into Cartesians coordinates X, Y and Z. The second step is to use the LM file generated using previous functions to restore the sinogram. This is done by converting the Cartesian coordinates into sinogram bins, radial, r_o, and angular, ϕ. This process can be seen in the Figure 3.13. The sinograms generated are converted into a LM file and histogrammed into images in the sinogram domain.

The simplicity of this function resides in the use of two Toolbox functions with just one command line. In addition, if some information is unknown, the default mode facilitates these values to enable the code computing.

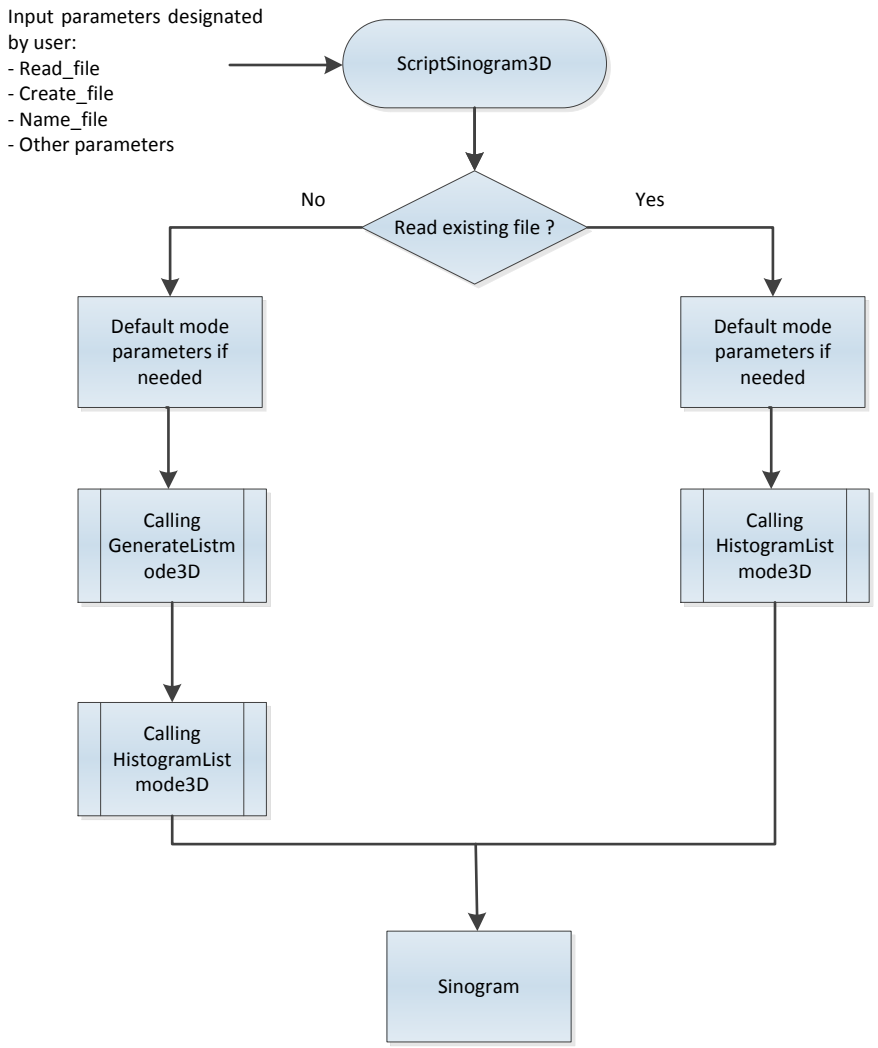


Figure 3.12. *Block Diagram of the script for the creation of the sinogram. On the left side the aim is to generate sinograms based on virtual data. On the right path the objective is to histogram LM files based on real data.*

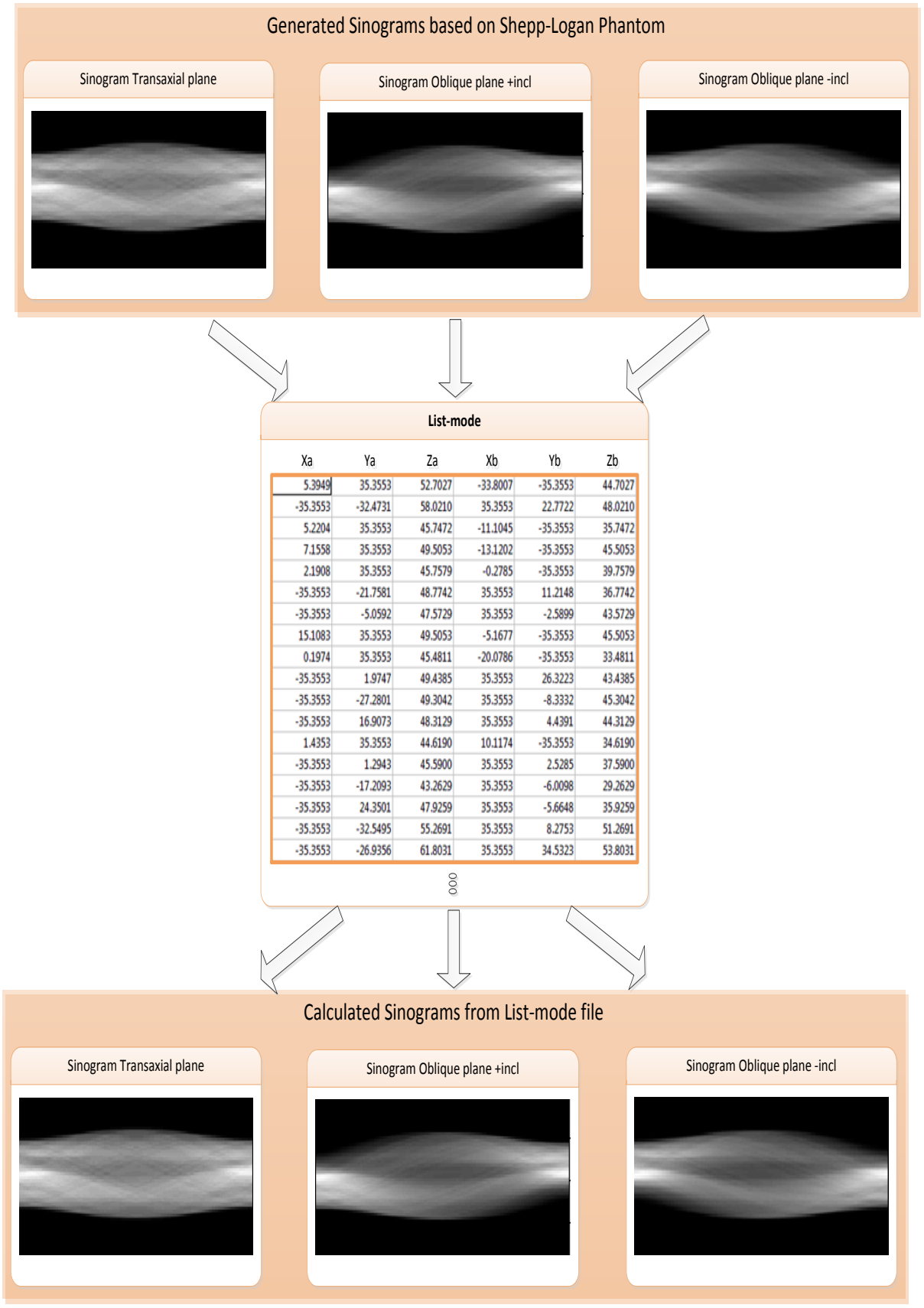


Figure 3.13. *Creation of a LM data from SL phantom and LM histogrammed sinograms for transaxial, oblique $\theta>0$ and $\theta<0$ planes.*

3.4. Correction, compression and reconstruction of the data

This section introduces the functions integrated in the third part of the created Toolbox. The aim of this block of functions is to correct the physical defects occurred during the acquisition process, rebin the sinograms and reconstruct them into the image domain.

As can be seen in the Fig.3.14 and Fig.3.15, the process is divided into three sub-functions. The first function corrects the data contained in the coincidence events sinograms. For that, all the information obtained during the image acquisition process has to be used: attenuation, normalization and scatter correction. The second function uses the SSRB method to reduce the amount of data and rebins the data to 2D for the reconstruction methods. During this process some essential information can be lost. Nonetheless, the need to decrease computational time and the storage reduction compensates this loss of information. The third function reconstructs the image using the MLEM method. The information obtained is going to be saved in a file called *'Final.scn'*, giving this information to the user by the Matlab Command Window.

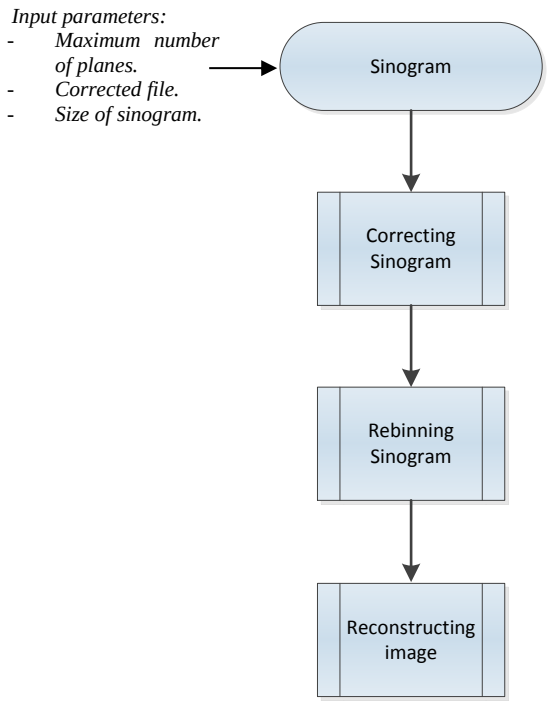


Figure 3.14. *Block Diagram of the function 'Sinogram.m'.*

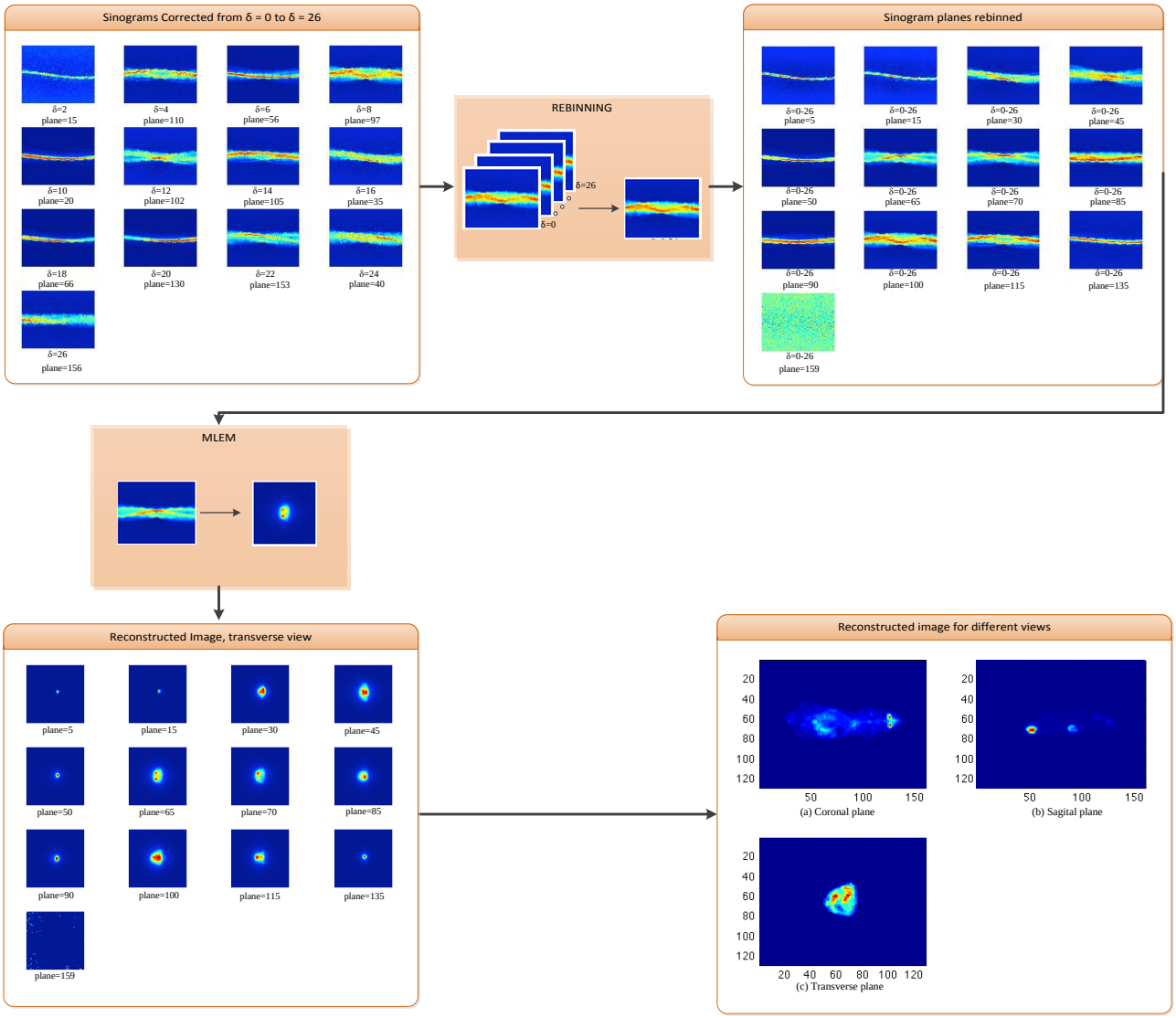


Figure 3.15. *Block diagram of the function 'Sinogram.m'*

3.4.1. Application of correction algorithms

The sinograms have to be corrected to decrease artifacts and unsubstantial information that interferes with the positrons collected. The correction applied can be seen in the equation (24). This correction is based on the substraction and effect of some parameters obtained inside the PET device during the acquisition process.

This information is divided in: scattered, attenuated, normalized and coincidence events sinograms. Another important parameter to process all the information is the value of δ. The variable δ corresponds with the PET span, which means the number of planes merged into a plane, direct and cross planes. According with the range of δ selected by the user and the information previously defined, the correct sinogram is going to be calculated and returned into a matrix named Corrected.

The process can be seen in the Fig.3.16. First a memory-mapped file object has to be created to acquire the values of the sinograms without consuming space and time. According with the value δ and before the correction process, the planes are selected.

In this function δ, which value is between 0 and 26, is divided in 52 segments. This means that each value of δ corresponds to two segments. The segments are related with the values of θ, representing the positive and negative oblique planes inclination. The planes are placed according with the δ segments, as can be seen in the Table 3.2. When the segment has a value of 0, this corresponds to the first transaxial plane. The next segment, segment 1, corresponds to the first positive oblique plane. The third segment, the segment with value 2, corresponds with the first oblique negative plane and so on. Notice that the number of planes of each segment decreases when the value of δ increases.

Table 3.2. *Correspondence of planes with δ segments.*

δ	Planes	Segment	Delta	Planes	Segment
0	159	0	**14**	77	27 28
1	155	1 2	**15**	71	29 30
2	149	3 4	**16**	65	31 32
3	143	5 6	**17**	59	33 34
4	137	7 8	**18**	53	35 36
5	131	9 10	**19**	47	37 38
6	125	11 12	**20**	41	39 40
7	119	13 14	**21**	35	41 42
8	113	15 16	**22**	29	43 44
9	107	17 18	**23**	23	45 46
10	101	19 20	**24**	17	47 48
11	95	21 22	**25**	11	49 50
12	89	23 24	**26**	5	51 52
13	83	25 26	**-**	-	-

The user has to select the value of δ. For that, the user has to give a range of values, the initial and final values of the segment selected. This range is limited between 0 and 52, both included. As can be observed in the Table below, for each segment there is the number of planes correspondent with δ value. According with the values selected, the axial compression is going to be performed. If the initial value has been selected to be 0, and as final value 52, that means that the *span* value is 0 and there is not axial compression. If there is not axial compression, all the planes are going to be part of the final sinogram.

Once the sinogram range has been selected and a compression mode has been successfully applied, the correction process for the sinograms can be performed. During the acquisition process, additional information has been collected that interferes with the coincident events detected. The objective is to correct the events collected inside the scanner FOV. For that, the equation (24) is used. In this equation, the corrective parameters used are the scattered, normalized and attenuated information. This equation is going to be applied plane by plane. The objective of this process is to increase the performance and level of details in the sinogram results.

Notice that this function, as the previous ones explained, is modular. The functions can be used alone or as a group according with the needs of the user.

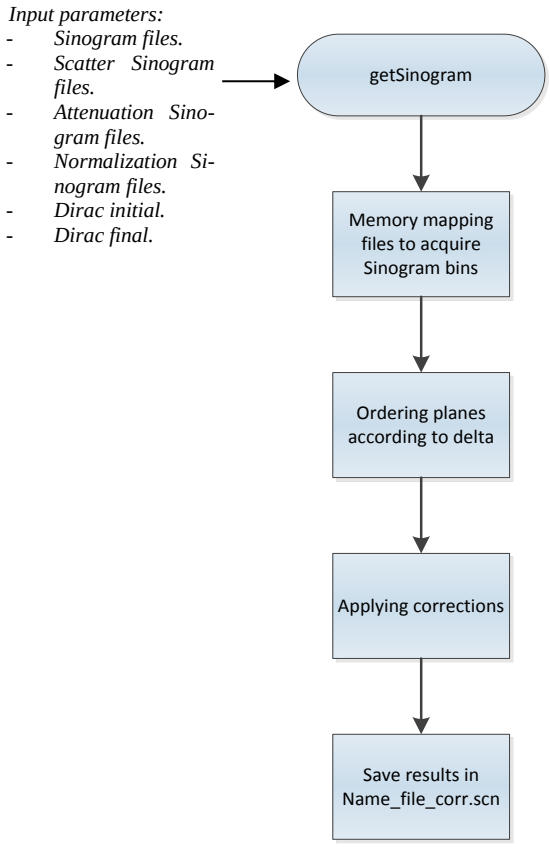


Figure 3.16. *Block Diagram of the subfunction 'getSinogram.m'.*

3.4.2. Rebinning the sinograms

The sinograms have to be rebinned in order to apply 2D reconstruction algorithms to 3D data. In this thesis the Single Slice ReBinning (SSRB) method has been selected to rebin the slices. On example of application of this method can be seen in the Fig.3.17. The function development is divided in two main parts, see Fig.3.18.

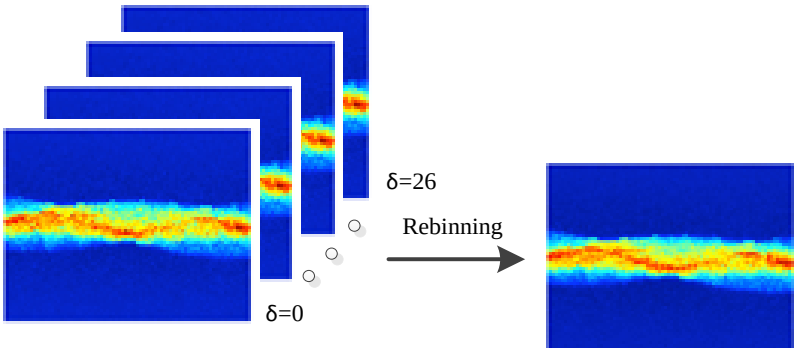


Figure 3.17. *Block Diagram of the subfunction 'rebSSRB.m'.*

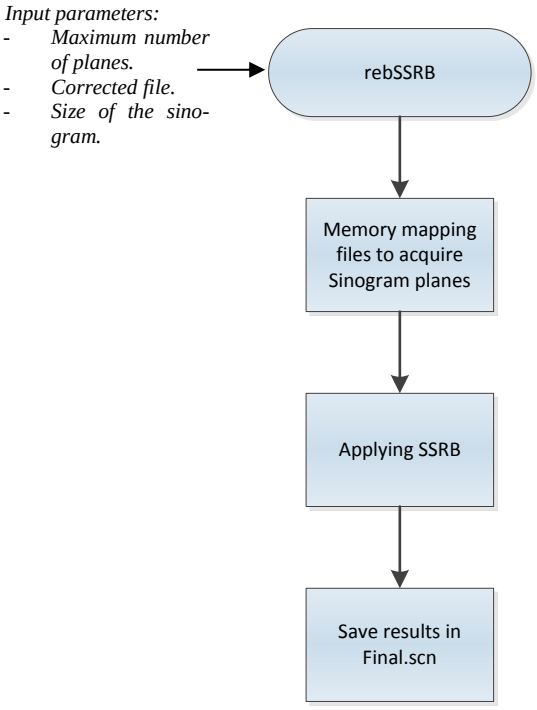


Figure 3.18. *Block Diagram of the subfunction 'rebSSRB.m'.*

The first part is the acquisition part, where the sinograms are going to be mapped to reduce the memory space and computing time. The second part is the core of the func-

tion. The rebinning is applied to the sinograms that are between the minimum and maximum value of δ. This method is going to enhance the quality of the direct images and reduce the information size from the sinograms. The method used for this purpose, as has been explained in the section 2.4, is the SSRB method. The rebinning method adds all the oblique sinograms that intersect with the transaxial field of view of the direct sinograms.

Once all the planes are rebinned according with δ, the information contained in the oblique planes, positive and negative, have been added to the transaxial plane. The order of the planes can be seen in the Fig.3.3. Additionally, the correspondence between transaxial, oblique with positive inclination and oblique with negative inclination can be seen in the Table 3.1. As result, the sinogram bins from different oblique planes and δ range are added to the transaxial planes. This addition of information enhances the final plane obtaining enriched results. This effect can be depicted in the Fig.3.17.

Although the rebinning method increases the counts and decreases the noise, the result loses detail due to the decrease of oblique planes. Notice that, as in the previous cases, this function can be used alone or with the functions of the third Toolbox block.

3.4.3. Reconstruction of the object scanned

This function reconstructs the sinograms measured by a PET scanner. The reconstruction is done independently if the data has been corrected with physical factors or not. The image can be reconstructed using any reconstruction method. For this thesis, the iterative reconstruction method 2D-MLEM has been selected. This method calculates the data that is mostly likely to pertain to the image after each iteration. The likelihood will increase after each repetition which means that is going to converge.This function can be used alone or can be used after the programs explained previously have been processed.

First a system matrix has to be calculated according with the sinograms size. The system matrix defines the final resolution of the image reconstructed. Using this matrix the RT can be performed and modelled according with the scanner geometry. The next step is to load the sinogram to be reconstructed. Also, the number of iterations is needed for the MLEM. The number of iterations is a parameter selected by the user. Notice that as the number of iterations increases, the result is going to be more accurate but the computing time needed is directly proportional to this, and it is going to increase.

The MLEM is performed as many times as iterations have been selected. Also the range of planes to be reconstructed is a parameter that has to be selected. After each iteration the result, a correction factor, is multiplied by the current image estimated and divided by the sensibility. This is done to determine the maximum values that define the

most probable image bins values. The value obtained is the maximum likelihood estimator of the sinogram which represents the original image.

4. SIMULATIONS AND EXPERIMENTS

In this thesis work two types of data were used: data generated by a numerical phantom and data from real scanners. In the first group are the simulations where virtual data was used. The objective of the simulations was to ascertain the proper function development. In the second group are the experiments. The data source in the experiments was real measured data. The aim of the experiments is to demonstrate the usability of the Toolbox with real information.

The first Section is divided in three simulations based on SL numerical phantom explained in Section 3.1. The Section 4.2 is divided in two tests were data from preclinical scanners was used. The Inveon data was supplied by the Turku PET Centre. On the other hand, COMPET data was provided by the University of Oslo, Norway. The section 4.3 explains the tests used to demonstrate the Toolbox performance.

4.1. Simulations

Histogramming sinograms based on a LM data is the core of the Toolbox presented in this thesis. In this section, two simulations are going to be histogramed. The particularity of these simulations resides in the data origin, the LM data files are based on virtual data.

In the first simulation is going to be histogrammed LM data files generated by the function explained in the Section 3.1. In the second simulation is going to be shown that a LM data generated for high sampling values can be histogrammed for lower sampling values.

4.1.1. Histogramming a Phantom List-Mode data

In the first simulation, a LM data is created based on SL phantom. The LM data is then histogrammed with the function explained in detail in the Section 3.2. This simulation is done to ensure the correct behaviour of these functions. To ensure that the result is the same as the generated the MSE is going to be calculated over the created sinogram bins. Another objective of this simulation is to show the compatibility between functions involved in this process.

For the generation of the LM file, the user has to give some input parameters to start the calculations. The input values for this simulation are:

- *Nevents or contrast* = 1000.
- *Npix or scanner FOV* = 50 cm.
- *Nang or angular sinogram size* = from 0 to 180 degrees.
- *noisy or noise interference* = 0, deactivated.
- *planes or number of planes* = 2, 3D case.
- *incl or azimuthal* = 5 degrees.

Notice that the same simulation taking into account the noise is attached in the appendix of this thesis.

The number of events means that for each column of the created sinogram, were generated 1,000 events that, as has been explained in the Section 2.1, are the representation of the true coincidence photons. This value corresponds with the contrast of the image. On the other hand, the number of planes is the number of transaxial planes. In the 3D case, according with the Table 3.1, for each transaxial plane were generated two oblique planes, one with θ=+5° and the second with θ=-5°.

4.1.2. Histogramming a List-Mode data with lower sampling rates

In the second simulation the resilience of the histogramming function is going to be tested. The objective of this simulation is to test if a LM data based on high sampling values can be histogrammed for lower sampling values. Other test is going to be performed to study the dependence of the number of events with the contrast. The objective of the second test is to study when the number of events decreases if the sinogram can be resampled. In both simulations, the LM file created in the first simulation is used. The input values are modified to histogram the LM file. The first simulation for lower sampling values than the generated in the LM file and the second for lower number of events. The parameters selected can be seen in the Table 4.1 and 4.2.

Table 4.1. *Relation between input parameters obtained and used according with the samplings.*

Parameters for high sampling			**Parameters for low sampling**		
Nrad	***Nang***	***Samples***	***Nrad***	***Nang***	***Samples***
75	0-180	1000	30	0-120	1000

In the Table 4.1 can be seen the parameter values for high and low sampling. In the first column are the values obtained from the function that generates the LM file. The values are the radial sinogram size, *Nrad*, and angular sinogram size, *Nang*, and the number of events, *Samples*. These values have been obtained according with the sinograms simulated using as input parameters the values of the first simulation. The sampling values to resize the sinogram are going to be different, lower than in the first simulation. The values selected for that aim can be seen in the Table 4.1. The new radial value is 30 and the new angular value is between 0° and 120°.

In the Table 4.2 can be seen the parameter values for high and low number of events. In the first column are the values obtained from the function that generates the LM file. For this simulation the radial and angular sinogram size must be the same, but the number of events is going to be different, lower than in the first simulation. The values selected for that aim can be seen in the second column of the Table 4.2.

Table 4.2. *Relation between input parameters obtained and used according with the number of events.*

Parameters for high samples			**Parameters for low samples**		
Nrad	***Nang***	***Samples***	***Nrad***	***Nang***	***Samples***
75	0-180	1000	75	0-180	100

4.2. Experiments

The experiments selected are divided according to the PET scanner used for the data acquisition. In the PET scanners, Inveon and COMPET, the data is stored in different formats, explained in the Section 2.2.1.

In the first experiment the data has been acquired by a COMPET scanner. All the events are gathered in a raw LM file from which there is not additional information. The first experiment is to calculate the sinogram related to the COMPET data. To fulfil this objective, the function that converts coordinates into sinograms is used.

In the second experiment the information has been collected using an Inveon scanner. This scanner collects the data in sinograms. To reconstruct the images the third function of the Toolbox has to be used, see Section 3.4.

These experiments were selected to demonstrate the satisfactory behaviour of the functions developed in the Section 3 with data obtained from real acquisitions and saved in the two possible manners of a PET scanner.

4.2.1. Histogramming COMPET List-Mode data

The first experiment is to test if the function that histograms LM files is capable to histogram data obtained from a real PET device. The data in this case has been provided by the Turku PET Centre. The problem with this data is the absence of a header file to clarify the scanner FOV and some other information about the object scanned.

In this case, the scanner FOV has been calculated with the feature that links this value with the maximum and minimum coordinate values of the LM file. Also the total number of planes is unknown, for that reason the option to calculate this value has been selected. The radial and angular values that define the sinogram size were selected, as in previous cases, for a radial size of 75 and angular between 0° and 180°. Two sampling schemes were used to divide the planes according with θ. The first was for all the values of θ obtaining a unique plane. The second was for all the values of θ divided in three planes: one representing the transaxial plane, another for the positive oblique plane and the last one for the negative oblique plane.

4.2.2. Correction, Rebinning and Reconstructing images from Inveon scanner

This section can be divided in the three steps. Each step corresponds with the functions that form the third part of the Toolbox. In this experiment, after the application of the developed functions, the outcomes will be seen until the image is reconstructed.

In this experiment, the Inveon data is in the sinogram domain. The first step is to scan and read the sinograms to use the function that corrects the sinograms, see Section 3.4.1. The sinograms that contain the coincidence true events have to be corrected to obtain detailed results. The corrections have to be done using the attenuated, scattered and normalized events stored in sinograms at the same time that the coincidence events were stored. Firstly, the sinograms have to be selected. This selection depends on the parameter δ. As has been explained in the section 3.4.1, when δ=0 corresponds with the first transaxial plane, δ=1 and segment=1 with the first oblique plane for θ>0 and δ=1 and segment=2 with the first oblique plane for θ<0, successively.

In this experiment the value of δ selected is between 0 and 52 to calculate all the sinograms. The result, the corrected sinograms, is going to be a matrix with 4319 planes. The next step is to compress the number of planes and increase the detail in each plane using the SSRB method. The result is going to be a sinogram with a decreased number of planes from the original size.

The final step is to reconstruct the images acquired in the PET scanner from the sinograms that have been corrected and rebinned. To achieve this, the function that recon-

structs images explained in the Section 3.4.3 was used. This function uses the MLEM algorithm to reconstruct images examining planes of the object to be recreated. The number of iterations done was 2, 20 and 50. This selection of values demonstrates the dependence of the image with the number of iterations.

Notice that a comparison between the results obtained after each step is going to be shown to demonstrate the effect of the correction methods performed.

4.3. Evaluation of the implementation performance

One important part of a program, besides the correct behavior, is fast performance, robust structure, the capability to handle large files and perform a huge amount of processes inside a function without causing an error or running out of memory. Using the Profiler Window the functions of the three programs that forms the Toolbox can be tested. This is a reliable method to improve the functions and learn the weaknesses of the program.

The first group of simulations is done to test the performance of the function that generates a LM file. To test this function 5 simulations were performed. The differences between them are the input parameters. These parameters are modified to see how these values affect the computing time and where the function expends the major time in percentages. The input values are: the number of events, *Nevents*, if noise is added to the simulation, *noisy*, the number of planes, *planes*, and the inclination, θ.

The second group of simulations was be done to test the performance of the function that histograms a LM file. The same simulations tested in the previous functions were used. In that way, the time needed to generate a LM file with the time needed to histogram the obtained LM file can be compared.

The third group of simulations was used to test the functions contained in the third block of the Toolbox. The simulations that were performed are three, one for each function. The input parameters for the first two functions are based on the range span and in the number of iterations for the third function.

5. RESULTS

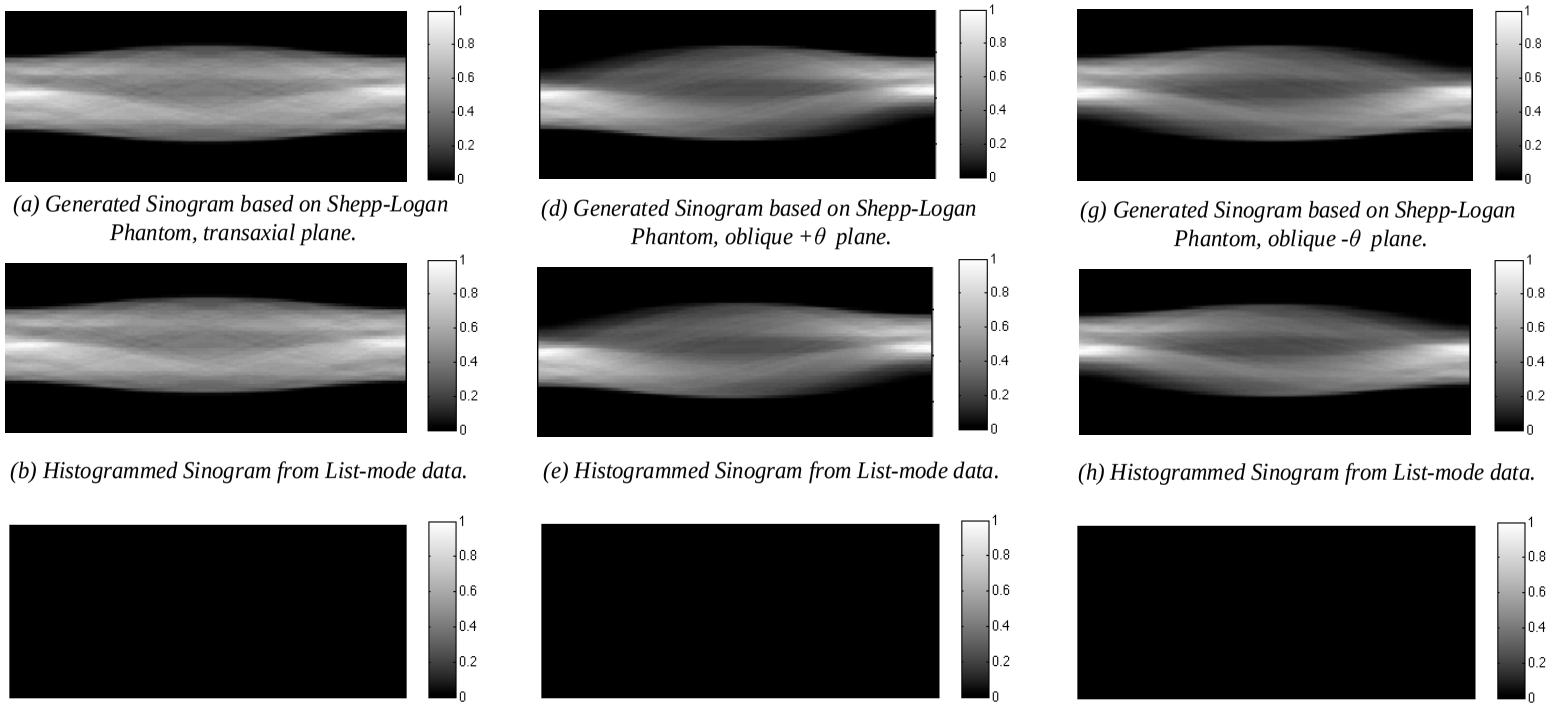
In this chapter the results obtained for the Section 4 are going to be shown. In the previous section, the tests have been divided in simulations and experiments, according with the data source. The simulations are the tests that use virtual data. Otherwise, the tests that use real data are called experiments.

5.1. Results from the Phantom tests

5.1.1. Histogramming a Phantom List-Mode data

The results obtained from the simulations explained in the Section 4.1.1 can be seen in the Figure 5.1. The Figure 5.1 compares the sinograms based on the SL phantom with the histogrammed output sinograms. The sinograms based on the SL phantom are divided in Fig.5.1 (a) for the transaxial plane, Fig.5.1 (b) oblique plane with $\theta = +5°$ and Fig.5.1 (g) oblique plane with $\theta = -5°$. These sinograms were saved as different planes of the sinogram with *Sino.mat*. This was done to be able to compare this result with the histogramed at posteriori. The slices have been ordered according with the Figure 3.3 and the Table 3.1. After this process, a LM file has been created from the sinograms generated. Using the function that histograms LM files the sinograms are obtained. The results can be seen in the Figure 5.1 (b), (e) and (h).

Once the visual comparison is done, it is time to compare the results numerically. It can be depicted in the Figure 5.1 (c), (f) and (i) according with the trasaxial, oblique positive and oblique negative planes, respectively, that there exist no difference between the simulated and the histogrammed sinograms.



(a) Generated Sinogram based on Shepp-Logan Phantom, transaxial plane.

(d) Generated Sinogram based on Shepp-Logan Phantom, oblique +θ plane.

(g) Generated Sinogram based on Shepp-Logan Phantom, oblique -θ plane.

(b) Histogrammed Sinogram from List-mode data.

(e) Histogrammed Sinogram from List-mode data.

(h) Histogrammed Sinogram from List-mode data.

(c) Differences between (a) and (b).

(f) Differences between (d) and (e).

(i) Differences between (g) and (h).

Figure 5.1. *Comparison between the sinograms generated without noise (a),(d),(g), Sino.mat, and the result obtained (b),(e),(h), sinogram.mat and the difference between them for transaxial, oblique +θ=5 and –θ=5 planes (c),(f) and (i).*

This difference has been checked numerically applying the MSE to the generated and histogrammed sinograms. As it can be seen in the Figure 5.2, the MSE is zero. This result is satisfactory because it shows that the phantom sinograms are the same as the sinograms histogrammed from a LM file.

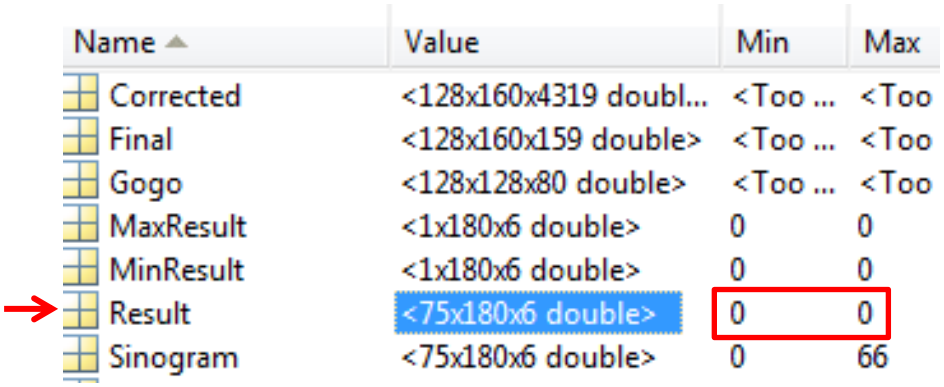


Figure 5.2. *MSE between Sino.mat, and Sinogram.mat.*

Also the number of events has been checked numerically, ensuring that in each column of the sinogram the range of the intensity values is the value selected by the user. Remember that for this simulation the number of events introduced was 1000.

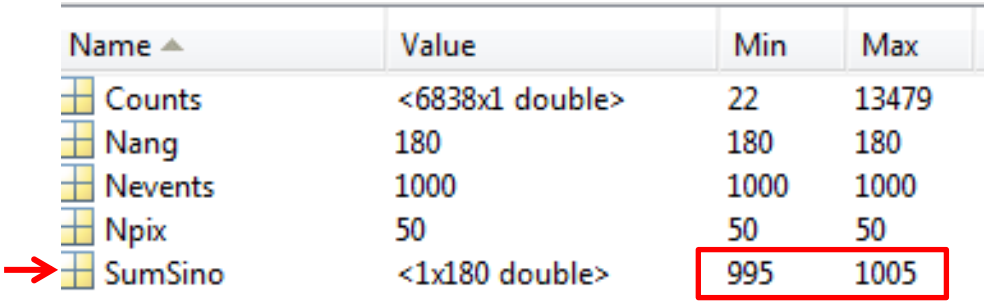


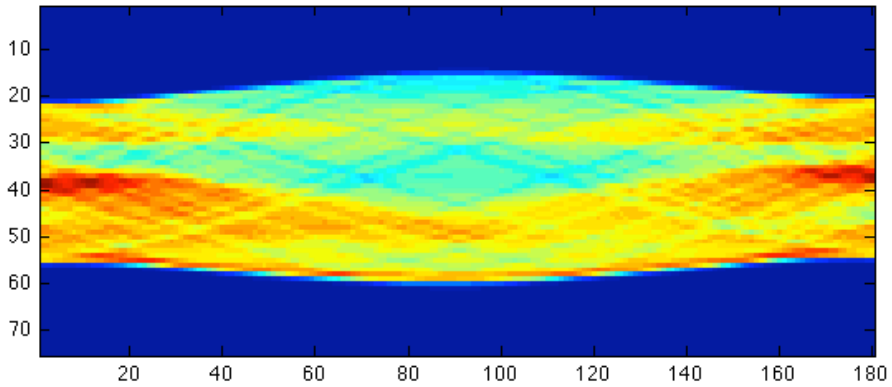
Figure 5.3. *Number of events in the obtained sinogram.*

The same simulation adding noise to the SL phantom has been added in the appendix of this thesis.

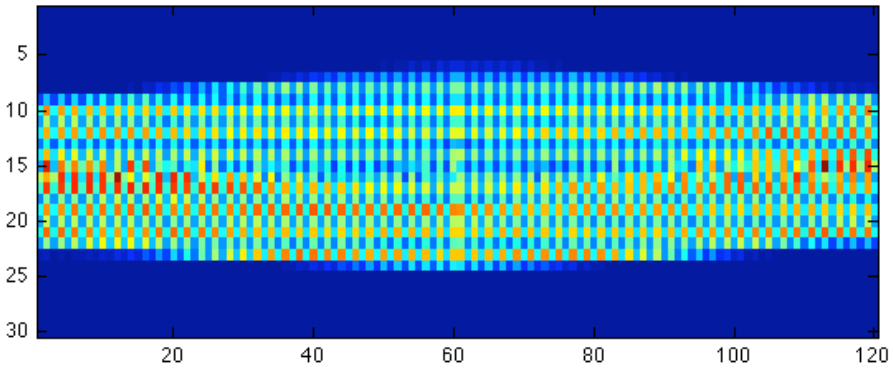
5.1.2. Histogramming a List-Mode data with lower sampling rates

In the second simulation, the objective was to demonstrate how LM data can be histogrammed for more sparse sampling values than in the first simulation.

The histogrammed sinogram from the LM generated in the Simulation 1 can be visualized in the Figure 5.4 (a). In the Figure 5.4 (b) can be seen the result of histogramming the LM generated in the first simulation for lower sampling rates.



(a) Sinogram with ro=75 and Φ=180°



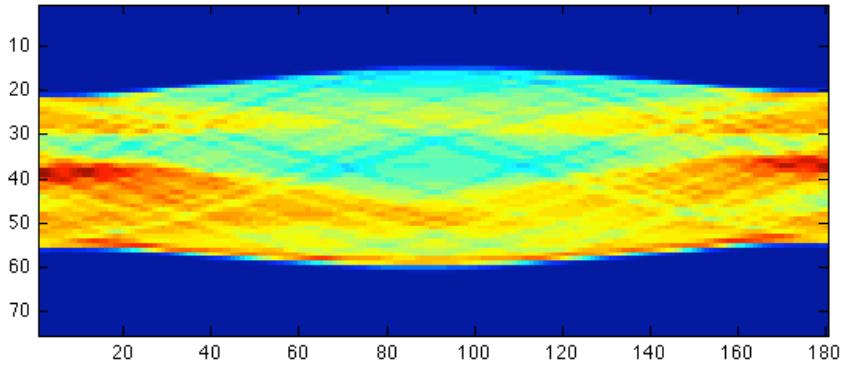
(b) Sinogram with ro=30 and Φ=120°

Figure 5.4. *(a) Sinogram for high samplings, (b) Sinogram for low samplings.*

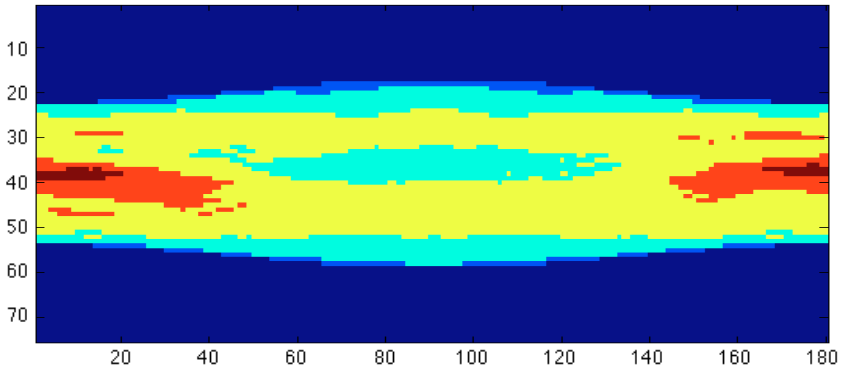
As can be seen in the Figure 5.4, the LM can be histogrammed using the original values or lower sampling. The radial and angular sampling decreases which affects to the bin size. The information is distributed according to the new bin size, which is placed in adjacent bins. This affects to the result obtaining a pixelated sinogram for the increased information at some positions. If a detailed result is required the selection of high sampling rate is recommended.

The result of the second test, see Table 4.2, can be seen in the Figure 5.5. As result, the sinogram losses constrast when the LM file is histogrammed for lower number of events than the generated. The result is less detailed than the original losing image quality.

A rule can be stated, the LM data can be histogrammed for lower sampling rate and lower number of events than the original values but the result will lose precision.



(a) Sinogram for high number of events= 1000.



(b) Sinogram for low number of events = 100.

Figure 5.5. *Sinogram for high (a) and low (b) number of events.*

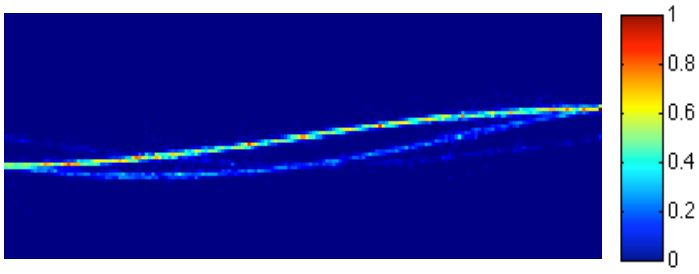
5.2. Result of the experiments based on COMPET and Inveon scanners

In this Section are going to be shown the results obtained for each experiment explained in the Section 4.2. The experiments are divided according with the data source. If the COMPET scanner is used the data is stored in LM data-sets. On the other hand, the Inveon scanners store the information in sinogram data-sets.

5.2.1. Histogramming COMPET List-Mode data

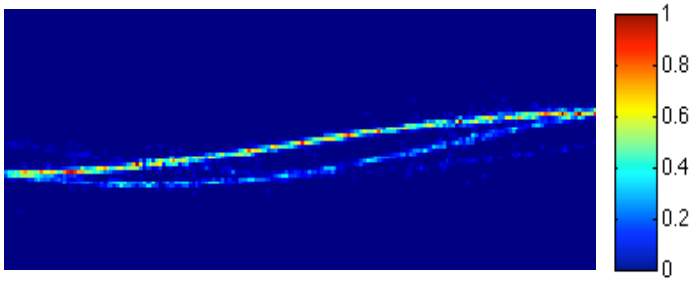
The Figure 5.6 shows the representation of the coordinates contained in the COMPET data into a sinogram for the values selected in the Section 4.2.1. The radial sampling value was 75 and the angular sampling value was between 0° and 180°. The scanner FOV was an unknown value. It was calculated by the function with a reasonable approximation of 98.113 mm. The number of planes was 1 in the first experiment and three in the second. In the second experiment the sinograms are divided into transaxial when $\theta = 0$, positive oblique plane for value of $\theta > 0$ and negative oblique plane when $\theta < 0$.

Figure 5.6 shows that the results obtained for the COMPET LM. The sinogram represents sinusoidal patterns, which are expected from rod-source measurements. The number of events was 8,003,877 and the time needed for the computing the LM file was 3 minutes and 48 seconds.

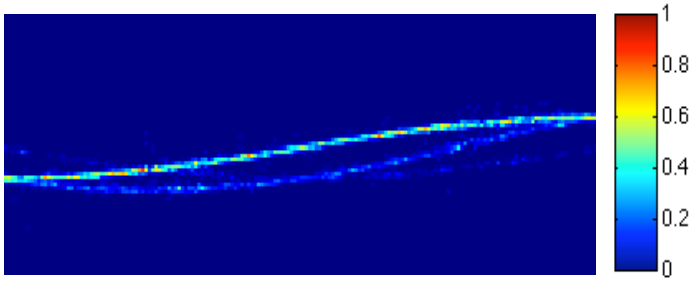


(a) COMPET Sinogram for all the values of θ .

(b) COMPET Sinogram Transaxial slice $\theta = 0$.



(c) COMPET Sinogram Oblique slice $\theta > 0$.



(d) COMPET Sinogram Oblique slice $\theta < 0$.

Figure 5.6. *Result obtained for COMPET LM file.*

5.2.2. Correction, Rebinning and Reconstructing Images from Inveon scanner

This experiment is divided in 3 parts. The first has as objective to scan and correct the sinograms using information stored during the acquisition process.

In the Figure 5.7 different planes for the same span value δ can be seen. This is done to observe the sinogram dependence with the projection domain. The parameter δ represents the span, a discriminative value that distributes the sinograms along the z axis. The δ value depends on the structure and scanner rings number. For the scanners that are going to be tested in this thesis, the range of δ is between 0 and 26. In the Table 3.2 are the correspondences between δ and the planes. In this case all the planes have been selected. The result for this experiment is a corrected sinogram matrix with size 128x128x4319.

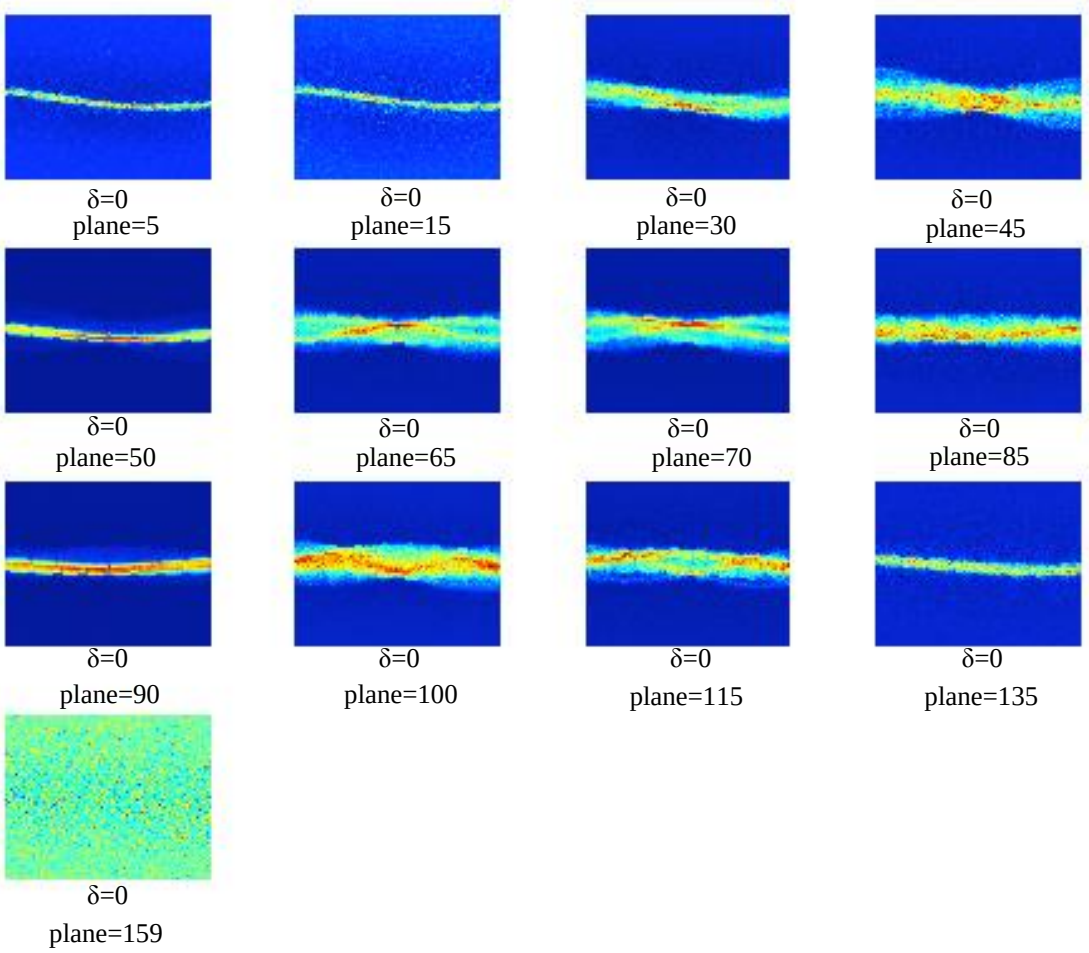


Figure 5.7. *Transaxial Sinograms planes for δ = 0.*

The second part is to compress the number of planes and increase the detail in each plane using the SSRB method. In the Fig.5.8 can be seen the result, a sinogram matrix

rebinned. Instead of being a matrix with size 128x128x4319 as in the first figure, it has been reduced into a matrix of dimensions 128x128x159.

As can be seen in the Fig.5.7 and 5.8, the addition of the planes with different δ increases the information contained in each sinogram. In the Fig.5.7 the planes are all from the same δ value, meanwhile in the Fig.5.8 each plane has information from $\delta = 0$ until $\delta = 26$. If the planes are compared between the planes with the same position, can be seen how the definition of the sinograms increases when the range of δ increases.

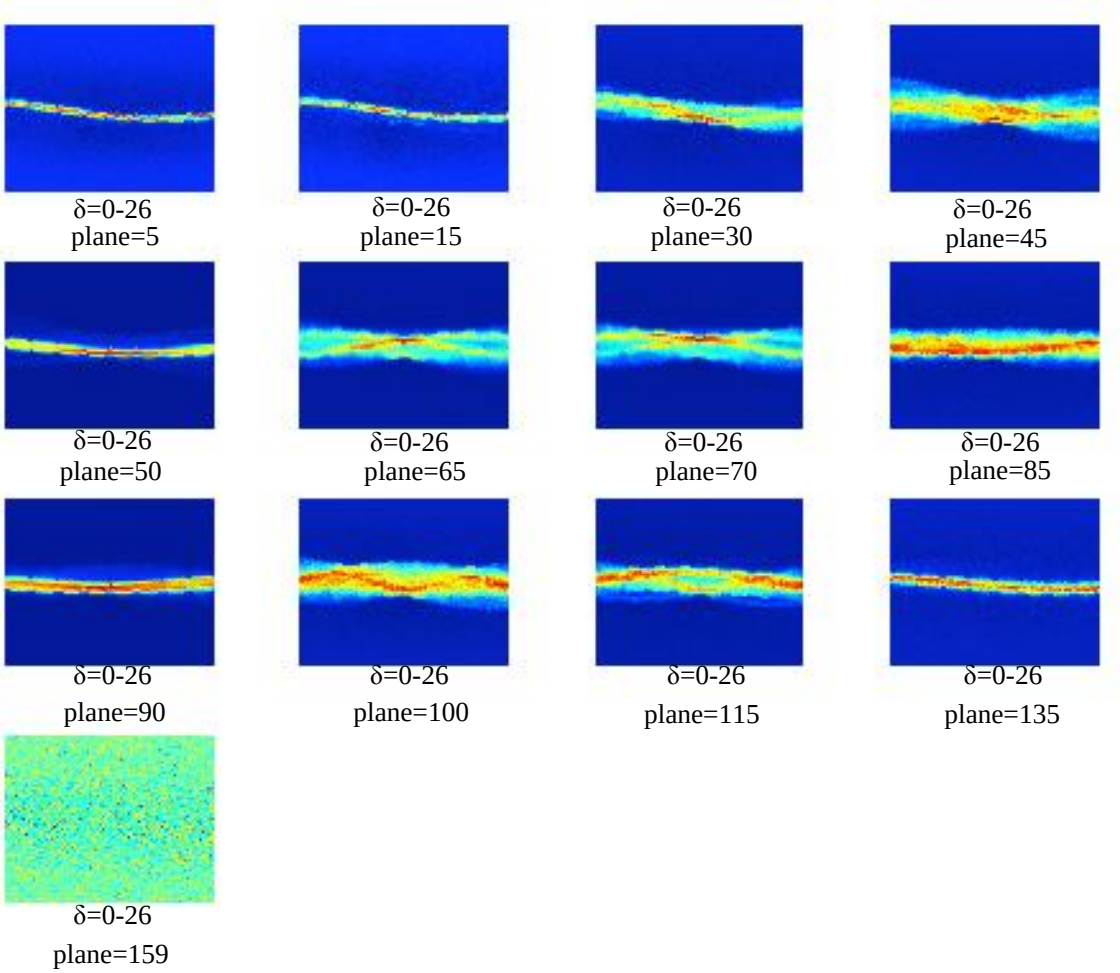


Figure 5.8. *Different rebinned sinograms planes from δ 0 to 26.*

The final step is to reconstruct the image from the sinograms. The Fig.5.9 shows the images reconstructed from the sinograms in axial planes. The number of iterations performed in this approach is two. In this Figure, some features cylindrical or spherical can be detected, where the events are concentrated. These zones enhanced in red are the areas where the radiotracers injected to the animal have highest uptake.

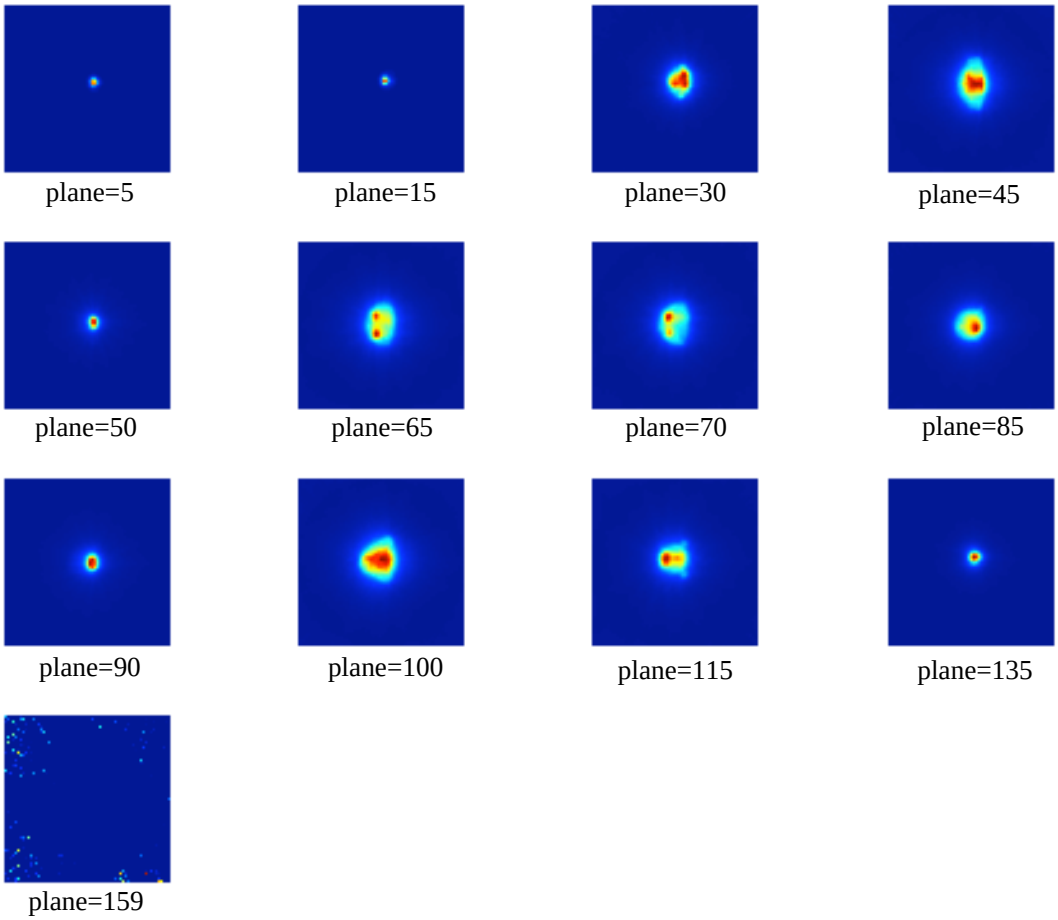
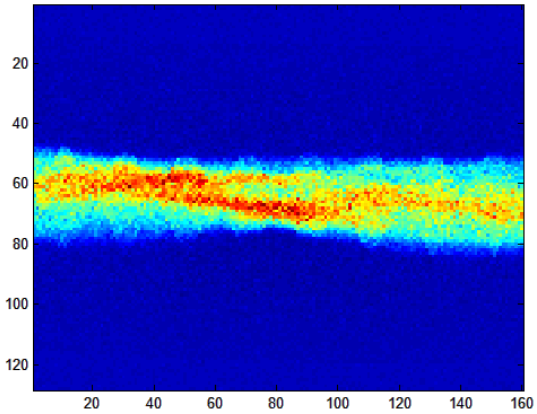
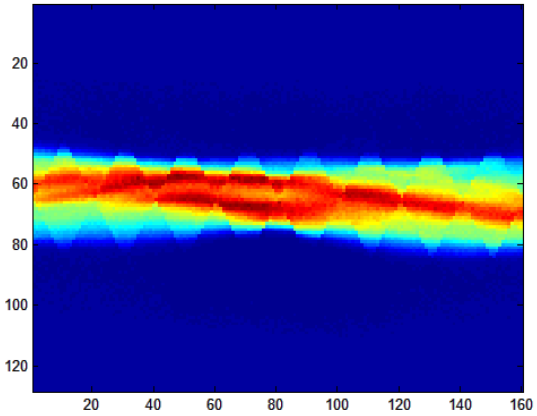


Figure 5.9. *Cross-sectional image planes after 2 iterations.*

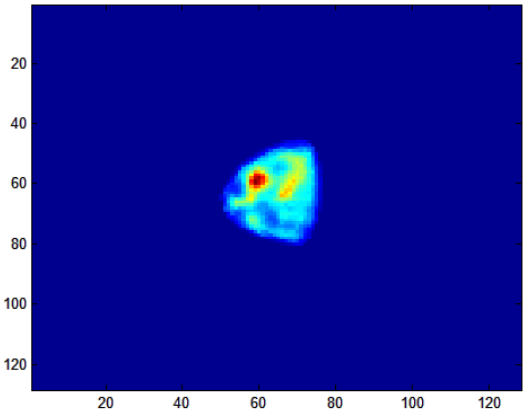
This process can be seen in more detail in the Figure 5.10. The same plane has been corrected, rebbined and reconstructed showing different results in each stage. In the Fig.5.10 (a) can be depicted the appearance of the sinogram 75 after has been corrected. The Fig.5.10 (b) shows the same plane rebinned. Notice that after applying the rebinning method, all the planes which correspond with the plane 75 between δ 0 and 26 have been added. As can be seen the size and level of detail increases after the sinograms have been rebinned. In the last step, the Fig.5.10 (c) shows the cross-sectional plane as reconstructed image.



(a) Corrected plane 75 of the sinogram.



(b) Rebinned plane 75 of the sinogram.

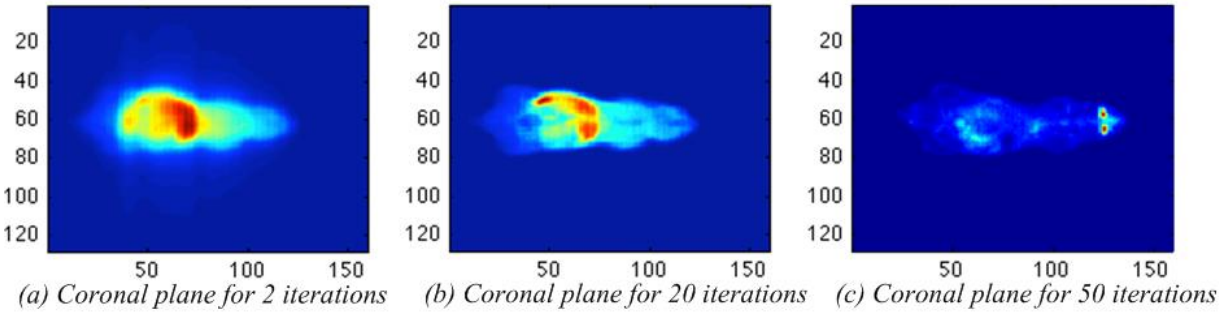


(c) Reconstructed plane 75 of the sinogram for 50 iterations.

Figure 5.10. *Reconstruction of the plane 75 after each function is processed.*

In the Figure 5.11 can be seen the reconstruction of the same image for the coronal view, after 2, 20 and 50 iterations. In this case, one can see how when the number of iterations increases, the quality of the image also increases giving more accurate results. In the Figure 5.11 (a) the shape of the scanned object can be seen, but more iterations are needed to define precisely the object contour. In the Figure 5.11 (b) the feature of a rat can be seen clearly. Also the region were the radionuclides are mostly concentrated can be depicted. These results show that increasing the number of iterations the images

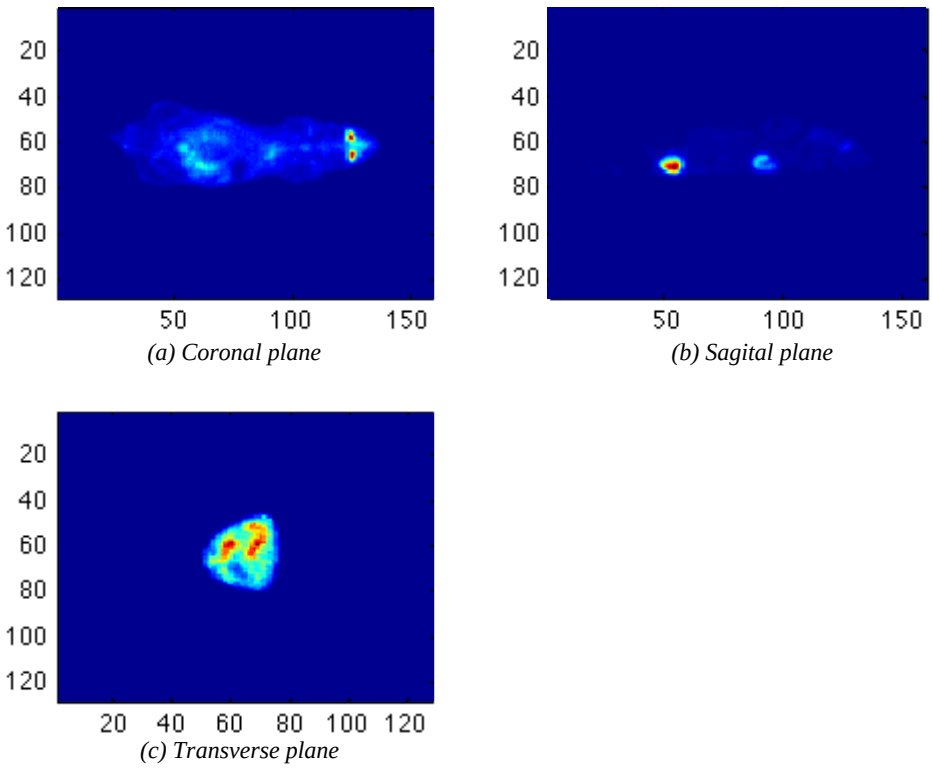
improve. The result after 50 iterations, see Figure 5.11 (c), is a highly defined image with an elevated level of detail.



(a) Coronal plane for 2 iterations *(b) Coronal plane for 20 iterations* *(c) Coronal plane for 50 iterations*

Figure 5.11. *Different coronal planes of the reconstructed image for 2 (a), 20(b) and 50 (c) iterations.*

In the Figure 5.12, the reconstruction for the coronal, sagittal and transverse view can be seen. The result is not blurred as in previous cases, the events are contained inside the contour of the object reconstructed and the uptake concentrations of events are localized into the regions were the radionuclide was absorbed. This absorption indicates regions with higher metabolic activity, which has to be analysed by a specialist for a diagnostic.

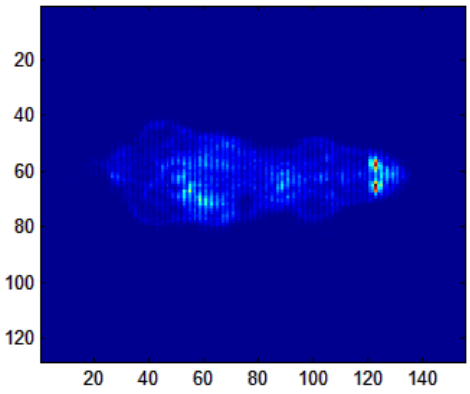


(a) Coronal plane *(b) Sagital plane*

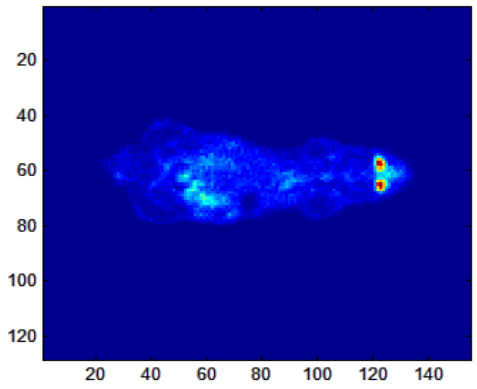
(c) Transverse plane

Figure 5.12. *Different perspective views of the reconstructed image for 50 iterations.*

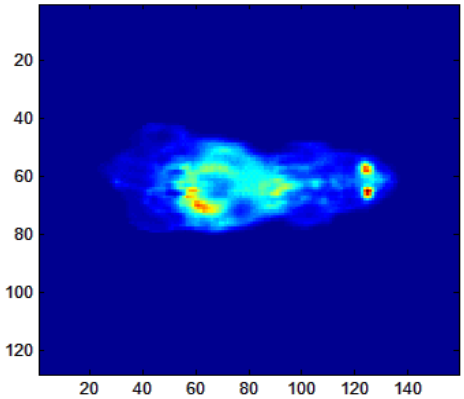
The Figure 5.13 shows the images reconstructed before using the functions that integrate the third Toolbox program. This experiment has been selected to show the improvements in the reconstructed images after applying the correction algorithm and the SSRB method.



(a) Reconstructed data without applying corrections and SSRB methods for 50 iterations.



(b) Reconstructed data without applying SSRB Method for 50 iterations.

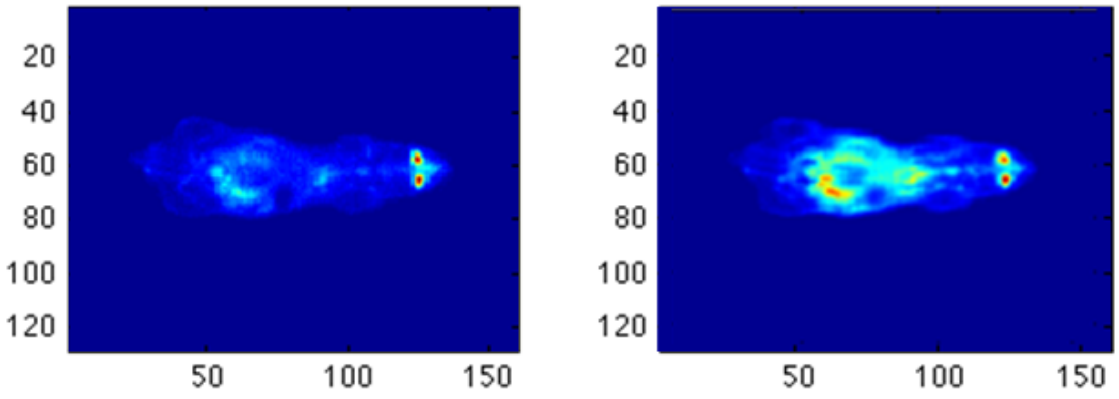


(c) Reconstructed data applying corrections and SSRB methods for 50 iterations.

Figure 5.13. *Reconstructed image for 50 iterations after functions 'getSinogram.m' (a), 'rebSSRB.m' (b) and 'My_MLEM.m' (c) have been applied.*

As can be seen in the Figure 5.13 (a), before applying the corrections the image is not complete. This is the reconstruction of the coincidence events sinograms. It can be seen how during the acquisition process some information is missed appearing as stripe artifacts in the image. The result is improved when the correction is applied as can be visualized in the Figure 5.13 (b). The last image, Fig. 5.13 (c), shows how the result can be enhanced by adding the oblique planes to the transaxial. This is due to the increase of information per plane.

Another comparison can be tested: the contrast between reconstructed images for different δ span. In the Figure 5.14 (a), can be shown the image reconstructed from the matrix $\delta = 1$. On the other hand, the Fig.5.14 (b) shows a reconstructed image where the entire δ sets have been added to the final matrix. Comparing these images can be seen how the quality of the image improves when more information is added before the reconstruction. This addition of information makes it easier to detect sensitive areas.

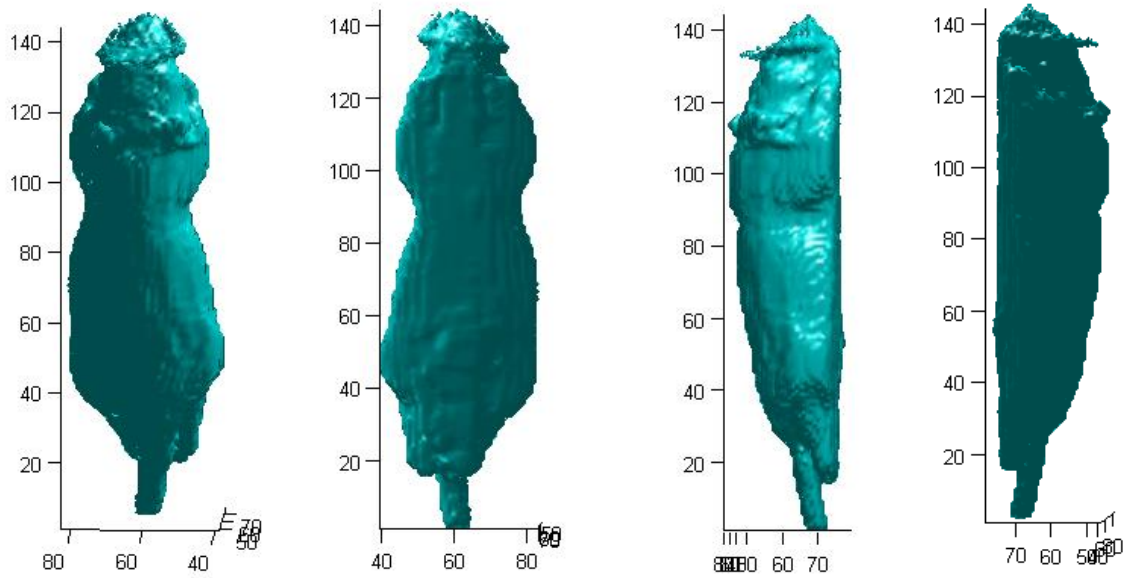


(a) Reconstruction for Rebinning d=1

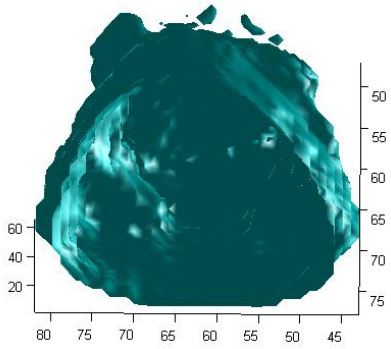
(b) Reconstruction for Rebinning d=0-23

Figure 5.14. *Reconstructed image for 50 iterations for different values of delta in the rebinning process. (a)Rebining for δ=1 (b) Rebinning for δ between 0 and 23.*

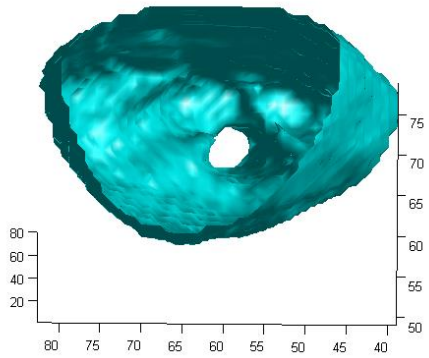
Finally, with all the information obtained in this experiment, the object scanned inside the FOV can be reconstructed. This is done to recognise the object and demonstrate that the information is processed for 3D. In the Figure 5.15 can be seen the 3D image of the object inside the Inveon scanner for different views.



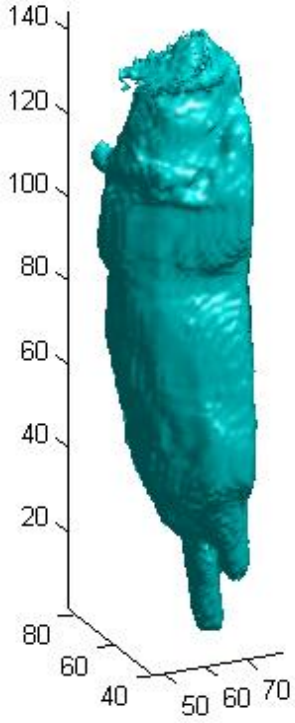
(a) Coronal plane, upper view *(b) Coronal plane, lower view* *(c) Sagital plane, right view* *(d) Sagital plane, left view*



(e) Transverse plane, slices 1 to 80



(f) Transverse plane, slices 80 to 159



(g) General view

Figure 5.15. *Reconstructed image for 50 iterations in 3D for different views.*

5.3. Results of the implementation performance evaluation

In this section are going to be shown the computational time and the maximum processing time of the functions from the tests. The computer characteristics are relevant information because the computer efficiency affects directly to the implementation performance and execution time. In this case, the processor is Intel(R) Core(TM) 2Duo on a CPU of 3.00GHz with a RAM of 4GB and the OS is Windows 7.

Each test to each data set has been done 5 times. The time shown is the mean obtained for these 5 simulations. In the first set of simulations the function that generates a LM file is tested.

Creation of a LM file

The first test is based in the number of events. This simulation has been selected to test if the number of events has influence in the function execution time. The results obtained can be seen in the Table 5.1.

Table 5.1. *Performance simulation 1 for function that generates a LM file.*

Nevents	**Npix** (cm)	**Nang** (°)	**Noisy**	**Planes**	**Incl** (°)	**Time** (s)	**Max relative processing time** (%)
100	50	180	0	1	0	1.981	Bins to coordinates, 67 Shuffle coordinates, 32.3
1000	50	180	0	1	0	63.488	Bins to coordinates, 75.7 Shuffle coordinates, 24.3
10000	50	180	0	1	0	21220.999	Bins to coordinates, 0.71 Shuffle coordinates, 99.24

As can be seen, the time needed for the execution increases exponentially with the number of events. The enormous increase of time is directly associated with the time expended by the function responsible to shuffle the coordinates.

In the second simulation the number of events has been combined with the addition of noise. In this manner it can be detected if the addition of noise affects to the performance time. The results obtained can be seen in the Table 5.2.

Table 5.2. *Performance simulation 2 for function 'generateListmode3D.m'.*

Nevents	**Npix** (cm)	**Nang** (°)	**Noisy**	**Planes**	**Incl** (°)	**Time** (s)	**Max relative processing time** (%)
100	50	180	0	1	0	1.981	Bins to coordinates, 67 Shuffle coordinates, 32.3
100	50	180	1	1	0	0.520	Bins to coordinates, 79.3 Shuffle coordinates, 10.1
1000	50	180	0	1	0	63.488	Bins to coordinates, 24.3 Shuffle coordinates, 75.7
1000	50	180	1	1	0	0.187	Bins to coordinates, 83.4 Shuffle coordinates, 8.6

These results show an interesting behavior, when the noise is added the time needed to process the information is lower than when there are not external interferences in the data.

The results of the third simulation can be seen in the Table 5.3. This simulation has as objective to compare the computing time between one and two planes for 100 and 1000 events.

Table 5.3. *Performance simulation 3 for function 'generateListmode3D.m'.*

Nevents	**Npix** (cm)	**Nang** (°)	**Noisy**	**Planes**	**Incl** (°)	**Time** (s)	**Max relative processing time** (%)
100	50	180	0	1	0	1.981	Bins to coordinates, 67 Shuffle coordinates, 32.3
100	50	180	0	2	0	10.844	Bins to coordinates, 70.1 Shuffle coordinates, 28.7
1000	50	180	0	1	0	63.488	Bins to coordinates, 24.3 Shuffle coordinates, 75.7
1000	50	180	0	2	0	1211.879	Bins to coordinates, 7 Shuffle coordinates, 93

In this case, when the number of events is constant, it can be seen the time dependence with the number of planes. The time increases by a factor of 5.47 in the first case and by 19.08 in the second case, being 3.4 times higher. This table demonstrates the difference in the processing time needed by the functions when the number of events changes from 100 to 1000. When the number of events is 100 the maximum processing time is used by the function that converts the bins to coordinates. Meanwhile, when the number of events is 1000, the maximum processing time is used by the function that shuffles the coordinates.

LM file in 3D

The fourth simulation is based in the relation between the number of events and the value of θ. The aim of this performance test is to detect if there is an increase of time when θ increases. Notice that for the 3D case θ is different to 0.

Table 5.4. *Performance simulation 4 for function 'generateListmode3D.m'.*

Nevents	**Npix** (cm)	**Nang** (°)	**Noisy**	**Planes**	**Incl** (°)	**Time** (s)	**Max relative processing time** (%)
100	50	180	0	2	15	87.600	Bins to coordinates, 69.3 Shuffle coordinates, 29.7
100	50	180	0	2	50	81.412	Bins to coordinates, 68.1 Shuffle coordinates, 31,7
1000	50	180	0	2	15	12226.833	Bins to coordinates, 5.5 Shuffle coordinates, 94.1
1000	50	180	0	2	50	12214.662	Bins to coordinates, 5.5 Shuffle coordinates, 94.5

The Table 5.4 shows how the selection of θ doesn't affect to the execution time, neither to the relation of time spent by the different functions.

The last test is for the function that generates LM files is for the 3D case. In that case we compared the time needed when the value of θ is 0 and higher than 0. The number of events tested is 100 and 1000, as in previous cases.

Table 5.5. *Performance simulation 5 for function 'generateListmode3D.m'.*

Nevents	**Npix** (cm)	**Nang** (°)	**Noisy**	**Planes**	**Incl** (°)	**Time** (s)	**Max relative processing time** (%)
100	50	180	0	2	0	10.844	Bins to coordinates, 70.1 Shuffle coordinates, 28.7
100	50	180	0	2	15	87.600	Bins to coordinates, 69.3 Shuffle coordinates, 29.7
1000	50	180	0	2	0	1211.879	Bins to coordinates, 7 Shuffle coordinates, 93
1000	50	180	0	2	15	12226.833	Bins to coordinates, 5.5 Shuffle coordinates, 94.1

The Table 5.5 demonstrates that even having the same number of planes, if θ is 0, only the transaxial planes have to be calculated. This affects to the execution time needed being lower if θ is 0. Notice that when θ is different than 0, which corresponds with the 3D case, the oblique planes are added to the transaxial planes.

LM to sinogram

In this set of tests, the efficiency of the function that histograms LM files is proved. This function has been tested for the same simulations as the previous set. This is due because when the LM is generated, in each simulation, by the previous function is going to be used before the function that histograms LM files. Hence, the parameters that define the tests are the same for both performances.

The first test, see Table 5.6, compares the time needed for each performance if the number of events increases, starting in 100, raising until 1000 and finishing at 10000.

Table 5.6. *Performance simulation 1 for function 'histogramListmode3D.m'.*

Nevents	**Npix** (cm)	**Nang** (°)	**Noisy**	**Planes**	**Incl** (°)	**Time** (s)	**Max relative processing time** (%)
100	50	180	0	1	0	0.525	Reading file, 48.2 Coordinates to bin, 43.7
1000	50	180	0	1	0	4.825	Reading file, 47.7 Coordinates to bin, 44.6
10000	50	180	0	1	0	45.137	Reading file, 48.9 Coordinates to bin, 43.3

The results contained in the Table 5.6 show that the execution time is directly proportional with the number of events. The increase of time is related with the increase of number of events, with a factor approximately of 10. In this case, in contrast with the first simulation the time needed for histogramming the LM data set is negligible.

In the next performance test, see Table 5.7, the parameters selected are used to check if the time oscillates with the selection of noise or not. For that purpose we com-

pared the option with or without noise. This has been done for different number of events to show that the result is not fortuity.

Table 5.7. *Performance simulation 2 for function 'histogramListmode3D.m'.*

Nevents	**Npix** (cm)	**Nang** (°)	**Noisy**	**Planes**	**Incl** (°)	**Time** (s)	**Max relative processing time** (%)
100	50	180	0	1	0	0.525	Reading file, 48.2 Coordinates to bin, 43.7
100	50	180	1	1	0	0.071	Add path, 75.8 Coordinates to bin, 24.2
1000	50	180	0	1	0	4.825	Reading file, 47.7 Coordinates to bin, 44.6
1000	50	180	1	1	0	0.161	Reading file, 50 Coordinates to bin, 9.6

In this simulation can be seen, as in the second simulation of the function that generates LM files, how the execution time decreases when the noise is added to the image.

The third simulation shows the time difference when the number of planes increases in the 2D case. The results of this simulation can be seen in the Table 5.8.

Table 5.8. *Performance simulation 3 for function 'histogramListmode3D.m'.*

Nevents	**Npix** (cm)	**Nang** (°)	**Noisy**	**Planes**	**Incl** (°)	**Time** (s)	**Max relative processing time** (%)
100	50	180	0	1	0	0.525	Reading file, 48.2 Coordinates to bin, 43.7
100	50	180	0	2	0	1.231	Reading file, 55.7 Coordinates to bin, 35.4
1000	50	180	0	1	0	4.825	Reading file, 47.7 Coordinates to bin, 44.6
1000	50	180	0	2	0	11.059	Reading file, 56.8 Coordinates to bin, 37.4

In the Table 5.8 can be detected that the time needed to histogram a LM increases proportionally with the number of planes. For 100 events the factor of increment between results is 2.3, being 2.29 when the number of events increases until 1000.

The next simulations are focused in the 3D case. In the fourth simulation, see Table 5.9, the performance time for different θ values and different number of events is compared. The objective of this test is to observe the dependence of θ with the computing time.

Table 5.9. *Performance simulation 4 for function 'histogramListmode3D.m'.*

Nevents	**Npix** (cm)	**Nang** (°)	**Noisy**	**Planes**	**Incl** (°)	**Time** (s)	**Max relative processing time** (%)
100	50	180	0	2	15	3.411	Reading file, 56 Coordinates to bin, 37.2
100	50	180	0	2	50	3.418	Reading file, 56.6 Coordinates to bin, 36.5
1000	50	180	0	2	15	33.290	Reading file, 56 Coordinates to bin, 38.3
1000	50	180	0	2	50	33.129	Reading file, 56.9 Coordinates to bin, 37.4

In this simulation it has been shown how the value of θ when is different to 0, does not affect to the time expended. Also, the relative percentage of computing time spent by the functions is similar in all the cases.

The last simulation using the function that histograms LM files, intends to demonstrate the relation between the 2D and 3D case. For that the number of events increases and the value of θ is 0, 2D case, and 15, 3D case. The selection of these values is to test the computing time needed to histogram each generated LM. The results can be seen in the Table 5.10.

Table 5.10. *Performance simulation 5 for function 'histogramListmode3D.m'.*

Nevents	**Npix** (cm)	**Nang** (°)	**Noisy**	**Planes**	**Incl** (°)	**Time** (s)	**Max relative processing time** (%)
100	50	180	0	2	0	1.231	Reading file, 55.7 Coordinates to bin, 35.4
100	50	180	0	2	15	3.411	Reading file, 56 Coordinates to bin, 37.2
1000	50	180	0	2	0	11.059	Reading file, 56.8 Coordinates to bin, 37.4
1000	50	180	0	2	15	33.290	Reading file, 56 Coordinates to bin, 38.3

The results demonstrate that when θ is different to 0 the time needed to histogram the data-sets is higher than when θ is 0. Besides, it can be seen how the relative percentage of computing time is independent of the input parameters, being similar in all the cases.

Image reconstruction

The last set of simulations is used to assess the performance of the functions contained in the third block of the Toolboox. The first simulations are to test the relation of time with the number of span. In the first test the set of sinograms selected are from δ=0 to δ=26, meanwhile in the second simulation δ=1. The results obtained can be seen in Table 5.11.

Table 5.11. *Performance simulation for function 'getSinogram.m'.*

range	**Time** (s)	**Max relative processing time** (%)
0-52	46.887	Eq. Correction, 85.8
1-1	1.666	Eq. Correction, 82

The result of these tests is that when the range of delta increases, the computational time increases. Moreover, the relative percentage of the computing time is similar.

The simulations used to test the function that rebins the sinograms are based on the results obtained from the previous simulation. The results after rebinning both cases can be seen in the Table 5.12.

Table 5.12. *Performance simulation for function 'rebSSRB.m'.*

range	**Time** (s)	**Max relative processing time** (%)
0-52	8.486	Save Result, 67
1-1	0.132	Save Result, 50.4

As in the previous simulations, it is demonstrated that the execution time is directly related with the range of δ. Although, the function that spends the major time percentage is the same, that in this case is the function *Results*.

The last set of simulations is to test the function that reconstructs the images. The difference between simulations is based on the number of iterations performed. As has been demonstrated in the Fig.5.11, when the number of iterations increases the definition and quality of the image increases. But this has an effect on the execution time. Due to that effect this simulation has been selected. The results obtained in this case can be seen in the Table 5.13.

Table 5.13. *Performance simulation for function 'myMLEM.m'.*

Niterations	**Time** (s)	**Max relative processing time** (%)
2	11.238	Muliply Phantom Matrix, 45 Multiply Ratios, 42.9
20	101.328	Muliply Phantom Matrix, 49.3 Multiply Ratios, 46.9
50	256.701	Muliply Phantom Matrix, 49.4 Multiply Ratios, 47
100	505.047	Muliply Phantom Matrix, 49.5 Multiply Ratios, 47.3

As can be easily detected, the time needed for the information execution has the same relation as the increment of the number of iterations. In other words, the time expended in each iteration is the same, approximately 0.55 seconds. On the other hand, the percentage of time needed by the functions is also similar for all the simulations. This result demonstrates that the percentage is independent of the input values.

6. DISCUSSION

In this thesis, the methods and algorithms used by the data processing Toolbox have been developed and implemented. The usability of the Toolbox is to reconstruct data from PET scanners or design and develop a PET scanner. The Toolbox is divided in three main programs. The first part simulates data in LM format based on a numerical phantom. The second part of the Toolbox is a histogramming instrument, able to histogram any standardized LM data into a sinogram in the projection domain. The last part of the Toolbox was to create an image reconstruction tool. Each part of the Toolbox is going to be discussed separately.

The first part of the Toolbox is the generator of a LM file. This generator is a useful tool to test all the functions with numerical simulations. In addition, using a numerical phantom the two standarized PET storage methods can be generated: LM files and sinograms. To achieve that purpose, the Radon transform equations were adapted to different scanner geometry and format of real PET scanner. The SL phantom is the numerical phantom selected for the development of this function. At this point, each image bin has been converted into a LM event. The problem is that the events are positioned in a sequential manner, as they are calculated from the image. In order to solve this problem a bootstrapping and shuffling tools are used to obtain more realistic results. Different physical effects are not included, so the LM files are not realistic in that manner.

The study of the function behaviour and its correct performance has been tested. The results obtained can be consulted in the Section 5.1. In the work we have demonstrated that the sinograms generated and the sinograms histogrammed from the phantom are the same. Also, the number of counts obtained from the Toolbox has been tested. The result is the same than the value selected by the user.

The second part of the Toolbox is the histogramming instrument. This function uses the modified Radon transform equations to histogram any standarized LM file, from different scanners, into sinograms in the projection domain. The equations are adapted to the PET scanners format and geometry. If the user does not give all the critical values to process the function, the default mode approximates these values. This approximation is based on the information contained in the LM file.

The correct behaviour of this function has been asserted in the Section 5.2. Firstly, the function has been tested with simulated LM data from the first part of the Toolbox. Once the result ensures the correct performance of the function, the code can be tested

with real data. For this thesis, real LM data is from a COMPET scanner. The COMPET LM file does not have any basic information about the acquisition process, scanner size or expected result. For that reason, the result is only a demonstration.

The third part of the Toolbox is the function that processes and reconstructs sinograms from different sources. Firstly, the sinograms are corrected to reduce artifacts of the collection method and improve image quality. During the acquisition process parameters as the attenuation, normalization and scatter events were collected. The corrections are based on the effects of these parameters on the coincidence events. Afterwards, the 3D data is compressed into 2D data. This process is done to reconstruct the images using 2D reconstruction algorithms. Dealing with data in 3D is not a trivial issue and it has to be managed carefully for avoiding unintentional loss of information or selecting inconvenient methods for processing the data. The SSRB algorithm is the compression method selected to reduce the size of sinograms. The oblique planes are added to the transaxial planes obtaining more accurate results. Once the sinograms have been processed, they can be reconstructed using a 2D algorithm as MLEM. The result is an image of the object positioned in the PET scanner either in cross-sectional planes or in 3D.

This function has been tested in the Chapter 5.3. The results prove that the function works satisfactory. The real data, sinograms, have been collected using Inveon and COMPET scanners. The results look reasonable in both cases. Due to the lack of information for the COMPET LM file, the result obtained is only a demonstration.

Additionally, the interoperability between functions has been proven. All the functions are able to exchange information between each other sequentially or using external information to work independently.

During the development of the functions mentioned previously, we found out some difficulties. In the first function of the Toolbox was learned that histogramming very large datasets can be a time-consuming task. This is a natural finding, since the sampling of the modern sinograms is always quite dense. However, the speed was not a priority in this work. One solution to solve this problem can be to histogram a LM file with higher sampling to lower sampling values. This has been shown in the second simulation, see Figure 5.4. Despite the resolution decreases, losing quality and detail, the performance time improves significantly. Also, the use of rebinning techniques decreases the reconstruction time by reducing the size of sinogram data-sets and the result quality is satisfactory.

Moreover, during the development of the equations used to histogram the events some problems were found. When real LM data-sets were used, some bins were positioned in improper locations of the sinogram. The misplaced bins occur for the angular positions, ϕ, between 0° and 45°, in a mirrored position from the correct bin location. This problem was due to the angle sign in the trigonometric equations. The development

of vectorial equations was proposed to avoid this problem, obtaining a sturdier result avoiding the mirrored effect. Notice that this effect was not possible to be seen with phantom data due to the symmetry of the SL phantom.

The bootstrap tool is used to estimate statistical properties of PET images based on a few data samples. The objective of the first function is to give a realistic appearance to the LM data-sets, which can be done in a more efficient way by shuffling the events inside the LM randomly instead of using the bootstrapping tool. This solution decreases the computational time which was a collateral effect of the bootstrap used to create realistic LM data. On the other hand, the bootstrapping tool is independent, can be used alone to create statistically similar sinograms by resampling data samples. This is a useful tool during the PET prototype acquisition process. Using the bootstrapping tool sinograms based on a single sinogram collected can be generated. This avoids multiple scans of the object decreasesing the physicist radiation exposure. Another advantage is that the physical phantom measurement can be resampled.

The Matlab Profiler was used to study the dependence of the function with the input parameters. During the execution of the functions some interesting behaviours were found. The execution time is related with the number of events, planes, noise and if the oblique planes inclination, θ, is different to 0. The number of events and planes are directly proportional to the performance time. The same can be seen with the parameter θ because when θ is different to 0 the execution time is 6.5 times higher than when this value is 0. This is due to the creation of oblique planes which increases the number of planes. An interesting behavior detected is the dramatic decrease of time to generate and histogram LM data-sets with noise. Meanwhile the time needed to histogram a LM data with noise is 7.32 times lower than to generate the LM data set, the average is 470.14 times lower. This fact occurs because when the noise is added the images lose valuable information decreasing the content of the LM data. The performance time can be increased exponentially by the function responsible to shuffle the coordinates. Notice that this function does not have any effect in the processing time to histogram the LM data in contrast with the bootstrapping function, because no information is added to the LM.

Another parameter that affects the computational time is the range of δ. This parameter defines the number of planes merged into a plane, direct and crosses. Although the functions with higher usability remain with similar values, the time to process the data is proportional to δ. When δ increases the time needed to process the code is higher.

Further developments can be focused in the calculation of the coincidence, scattered, normalized and attenuated events. The prototype scanner coincidence events have been calculated already. At this point, the images can be reconstructed but the result would be similar to the Fig.5.13 (a). Incomplete images composed by stripes with and without information. In order to improve the result, as in the Fig. 5.13 (c), the corre-

spondent corrections must be applied to perform a real data reconstruction from the prototype scanner.

7. CONCLUSION

A modular data processing Toolbox has been created composed by three main functions. A realistic LM data based on a phantom has been generated by the first function. The second function histograms any standard LM file into sinograms in the projection domain. The third function performs the necessary data corrections and image reconstruction of the original image scanned. The Toolbox performance satisfies the expectations and hence it can be used for the development of the PET scanner prototype. The Toolbox functions are able to work independently but also being supplied by the outcome of the previous function. This characteristic shows the interoperation capability between functions.

REFERENCES

[1] Stearns, S.C. and Ebert, D. *Evolution in health and disease: work in progress.* The Quarterly Review of Biology vol.76 (4), Pp:417-32. Doi: 10.1086/420539. 2001.

[2] Glasser, Otto. Wilhelm Conrad Röntgen and the Early History of the *Roentgen Rays*. London: John Bale, Sons and Danielsson, Ltd. p. 305. (1933) OCLC 220696336.

[3] Becquerel, Antoine H.. *On radioactivity, a new property of the matter*. Nobel Lecture, December 11, 1903. http://www.nobelprize.org/nobel_prizes/physics/laureates/1903/becquerel-lecture.pdf

[4] Curie, Marie Sklodowska. *Radio-Active Substances*. In: The Chemical News, Vol. 88, no. 2282 - Vol. 88, no. 2297. London: Edwin John Davey, 1903.

[5] Frederick Proescher, M. D., *The Intravenous Injection of Soluble Radium Salts in Man.* Radium Journal. Vol 1, no. 5, Pp:9-10. The Radium Publishing Company. Pittsburgh,PA.1913. http://radiation-hormesis.com/bathtub%20experiments%20pitchblende%201913-1914_v1-2.pdf

[6] Joliot-Curie, Irène and Joliot, Frédéric. *Artificial Production of Radioactive Elements*. Nobel Lecture. December,12,1935. http://www.nobelprize.org/nobel_prizes/chemistry/laureates/1935/joliot-curie-lecture.html

[7] Lawrence, Ernest O. and Livingstone, M.S.. *The Production of High Speed Light Ions Without the Use of High Voltages*. Published in Physical Review Journal. Vol.40, 19-35. April,1, 1932. University of California. http://prola.aps.org/abstract/PR/v40/i1/p19_1

[8] European Comission. *Radionuclides for Medical Use*. Joint Research Centre and Institute of Transuranium Elements. June, 25, 2004. http://www.jrc.es.europa.eu

[9] Knoll, Max & Kügler, J.. *Subjective Light Pattern Spectroscopy in the Electroencephalic Range*. Nature (London) 184: 1823-1824. 1959.

[10] Watts, G. *The History of Ultrasound: A collection of recollections, articles, interviews and images*. BM Journal. Vol. 339. November, 15, 2005.

[11] Webb, S. *The Physics of Medical Imaging*. P 577-578. Taylor & Francis Group,LLC. Grat Britain.1988.

[12] Helvie MA. *Imaging analysis: Mammography*. In: Harris JR, Lippman ME, Morrow M, Osborne CK, eds. Diseases of the Breast. 4th ed. Philadelphia, Pa: Lippincott Williams & Wilkins; 2010:116-130.

[13] Hounsfield GN. *Computerised transverse axial scanning (tomography): Part 1. Description of system*. Br J Radiol. Vol.46 :1016–22. 1973.

[14] Hollingworth, W; Todd, C. J.; Bell, M. I.; Arafat, Q.; Girling, S.; Karia, K. R.; Dixon, A. K. *The Diagnostic and Therapeutic Impact of MRI: an Observational Multi-centre Study*. *Clinical Radiology* 55: 825–831. 2000.

[15] Ahluwalia, B.D.; Hales C.A.; Brownell, G.L. and H. Kazemi. *Study of regional lung function using cyclotron produced, RSNA and AAPM*. Chicago. November 1973.

[16] Jaszczak RJ, Coleman RE, Lim CB.. S*PECT: Single photon emission computed tomography*. IEEE Tran Nuclear Science. Vol.27, Pp: 1137–1153. 1980.

[17] Ogawa, S.; Lee, T. M.; Nayak, A. S.; Glynn, P.. *Oxygenation-sensitive contrast in magnetic resonance image of rodent brain at high magnetic fields*. Magnetic Resonance in Medicine. Vol. 14, Pp: 68–78.1990.

[18] Sweet, W.H. and Brownell, G.T. *Localization of brain tumors with positron emitters. Journal Nucleonics* Vol. *11*, Pp: 40-45*.1953.*

[19] Phelps, M.E.; Hoffman, E.J.; Huang, S.C.; Kuhi, D.E.. *ECAT:A new computerized tomographic imaging system for positren-emitting radiopharmaceuticals*. Journal NucIear Medicine. Vol.19, Pp: 635-647. 1978.

[20] Hines, K.E.; Rissi, M.; Volgyes, D.; Bolle, E.; Stapnes, S.; Rohne, O.; Dorholt, O.; Bjaalie, J.G.; Skretting, A.. *COMPET: High Resolution High Sensitivity MRI Compatible Pre-Clinical PET Scanner*. 13th Vienna Conference on Instrumentation VCI2013. Austria. February, 11-15, 2013.http://indico.cern.ch/contributionDisplay.py?contribId=167&sessionId=15&confId=186337

[21] Siemens. *Inveon: No Limits on Discovery*. Germany. 2008. http://www.healthcare.siemens.com/molecular-imaging/preclinical-imaging

[22] Daghighian, F.; Sumida, R. and Phelps, M.E.. *PET imaging: an overview and instrumentation*. Journal Nuclear Medicine Today.Vol.18 Pp:5. 1990.

[23] Beltrame, P.; Bolle, E.; Braem, A.; Casella, C.; Chesi, E.; Clinthorne, N.; De Leo, R.;Dissertori, G.; Djambazov, L.; Fanti, V.; Heller, M.; Joram, C.; Kagan, H.; Lustermann, W.; Meddi, F.; Nappi, E.; Nessi-Tedaldi, F.; Oliver, J.F.; Pauss, F.; Rafecas, M.; Renker, D.; Rudge, A.; Schinzel, D.; Schneider, T.; S´eguinot, J.; Solevi, P.; Stapnes, S.; and Weilhammer, P.. *The AX-PET demonstrator -Design, construction and characterization.* Nuclear Instruments and Methods in Physics Research, Accelerators, Spectrometers, Detectors and Associated Equipment. Vol. 654, Pp:546 – 559. 2011. http://indico.cern.ch/getFile.py/access?resId=0&materialId=slides&confId=175430

[24] Radon, Johann. *On the determination of functions by their integral values along certain manifolds*. IEEE Transactions on Medical Imaging. Vol. 5, Pp: 170-176. (1986).

[25] Bailey, D.L.; Townsend, D.W.; Valk, P.E.; Maisey, M.N. *Positron Emission Tomography.* Pp: 2-4. Springer-Verlag Gmbh. June, 1, 2005. ISBN-10: 1852337982.

[26] Czernin, J and Phelps, M.E.. *Positron emission tomography scanning: current and future applications*. Annu. Rev. Med. Vol. 53 Pp: 89-112. Los Angeles, California. 2002.

[27] Tai, Y.F.; Piccini, P.. *Applications of positron emission tomography (PET) in neurology.* Neuroscience for Neurologists. J Neurol Neurosurg Psychiatry. Vol. 75, Pp: 669–676. http://www.ncbi.nlm.nih.gov/pmc/articles/PMC1763584/pdf/v075p00669.pdf

[28] Keng, F.Y.J.. *Clinical Applications of Positron Emission Tomography in Cardiology : A Review.* Annals Academy of Medicine. Vol. 33 no.2. March, 2004. http://annals.edu.sg/pdf200403/v33n2p175.pdf

[29] Center for Drug Evaluation and Research (CDER). *Investigational New Drug Applications for Positron Emission Tomography (PET) Drugs GUIDANCE*. U.S. Department of Health and Human Services . Food and Drug Administration. Center for Drug Evaluation and Research (CDER). December 2012. http://www.fda.gov/Drugs/GuidanceComplianceRegulatoryInformation/Guidances/default.htm

[30] Michael E. Phelps. *PET : Physics, Instrumentation and Scanners*. Pp.8-10. Springer. 2006. ISBN 0-387-34946-4.

[31] *Uygar Tuna. Cosine domain Gap-Filling for PET Sinograms.* M. Sc. Thesis Tampere University of Technology. 2008.

[32] Kim, E. Edmund; Myung-Chul Lee; Tomio Inoue; and Wai Hoi Wong. Clinical PET: Principles and Applications. Pp.5-6. Springer. 2n ed. 2010 ISBN: 0387408541 / 0-387-40854-1.

[33] Hamamatsu Photonics. *Photomultiplier Tubes: Basics and Applications*. Chapter 14. Pp.270-273. 2006. http://psec.uchicago.edu/links/pmt_handbook_complete.pdf

[34] Gopal B. Saha. *Physics and Radiobiology of Nuclear Medicine*. Pp: 97-98. Springer-Verlag Gmbh, 4th ed. October, 1, 2012. ISBN-10: 1461440114.

[35] Sakari Alenius. The Big Picture of Tomographic Image Reconstruction. Introduction to Medical Image Processing. Tampere University of Technology. Spring 2013.

[36] Cheng, J.C;Rahmim, A. Blinder, S.; Camborde, M-L.; Sossi, V.. *Implementation of scatter correction List-Mode OP-EM reconstruction algorithm and a dual (Histogram/List-Mode) reconstruction scheme for dynamic PET imaging*. 2005 IEEE Nuclear Science Symposium Conference Record. http://www.jhu.edu/rahmim/research_work/dual_histogram_list_mode_Cheng_MIC05.pdf

[37] Popescu, L. M.; Matej, S.; Lewitt, R. M.. *Iterative image reconstruction using geometrically ordered subsets with list-mode data*. National Institutes of Health, USA http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.112.7882&rep=rep1&type=pdf

[38] Peter E. Valk, Michael N. Maisey, Dale L Bailey and David W.Townsend.. Positron Emission Tomography:Basic Science. Ed. Springer-Verlag London. Pp: 63-65. 2005. ISBN 1852337982 2005

[39] Stanley Roderick Deans, *The Radon transform and some of its applications*. Published by John Wiley and Sons, Inc.. New York. 1983. ISBN 10:0-486-46241-2.

[40] A.C. Kak, and M. Slaney. *Principles of Computerized Tomographic Imaging*. IEEE Press, 1988. http://www.slaney.org/pct/pct-toc.html

[41] Turkington TG. Introduction to PET Instrumentation. J Nucl Med. Technol. Vol. 29, Pp: 4-11. 2001.

[42] Vision and Image Engineering. Stanford University. 2008.

[43] Gash, H. Michael. *2D&3D Shepp-Logan Phantom Standards for MRI*. 19th International Conference on System Engineering. Las Vegas, Nevada. Aug, 19-21, 2008.

[44] Efron, B. and Tibshirani, R.J.. *An introduction to the bootstrap*. New York, London. Chapman & Hall, 1993.

[45] Lartizien, C.; Aubin, J.-B. and Buvat, I.. *Comparison of Bootstrap Resampling Methods for 3-D PET Imaging*. IEEE Trans. Med. Imag., vol. 29. 2010.

[46] Haynor, D. R. and Woods, S. D.. *Resampling estimates of precision in emission tomography*. IEEE Transactions on Medical Imaging, vol. 8, pp. 337-343, 1990.

[47] Lewitt, R.M.; Muehllehner, G. and Karp, J.S. *Three-dimensional reconstruction for PET by multi-slice rebinning and axial image filtering*. Phys Med Biol. Vol. 39, Pp: 321-340. 1994.

[48] Daube-Witherspoon, M. E. and Muehllehner, G.. *Treatment of axial data in 3D PET*. Journal Nucl Med. Vol. 28, Pp: 1717-1724. 1987.

[49] Defrise, M. and Kinahan, P. E.. *Data acquisition and image reconstruction for 3D PET*. Theory and Practice of 3D PET. Vol. 32, Pp. 11-54. Developments in Nuclear Medicine, D. W. Townsend and B. Bendriem, Eds. Dordrecht: Kluwer Academic Publishers. 1998.

[50] Defrise, M.; Kinahan, P. E.; Townsend, D. W.; Michel, C.; Sibomana, M. and Newport, D. F.. *Exact and Approximate Rebinning Algorithms for 3D PET Data*. IEEE Trans Med Imaging. Vol. 16, Pp. 145-158. 1997.

[51] Watson, C. C.; Newport, D. and Casey, M. E.. *A single scatter simulation technique for scatter correction in 3-D PET*. Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine. Pp. 255–268. 1996.

[52] Lalush, D. and Tsui, B.. *Performance of ordered-subset reconstruction algorithms under conditions of extreme attenuation and truncation in myocardial SPECT*. Journal Nucl Med. Vol. 41, Pp. 737-744. 2000.

[53] Yavuz, M. and Fessler, J. A.. *Statistical Tomographic Recon methods for randoms- precorrected PET scans.* Medical Imaging Analysis. Vol. 2, Pp. 369-378. December, 1998.

[54] P. E. Valk, D. L. Bailey, D. W. Townsend, and M. N. Maisey. Positron Emission Tomography: Basic Science and Clinical Practice. Springer, London, UK, 2003.

[55] Shepp L.A. and Vardi Y. *Maximum Likelihood Reconstruction for Emission Tomography*. IEEE Trans Med Imag. 1:2:113-122. 1982.

[56] Vermeulen F. and Lemahieu I. *An Evaluation of Maximum-Likelihood Reconstruction of PET Images in the Presence of Low Quality Attenuation Correction Measurements*. Vth Mediterranean Conference on Medical and Biological Engineering, Medicon 89. 336-337. Greece. 1989.

[57] Johnson, N.L.; Kotz, S. and Kemp, A.W.. *Univariate Discrete distributions* (3rd edition). John Wiley & Sons. Pp: 157. August, 2005. ISBN-10: 0471272469.

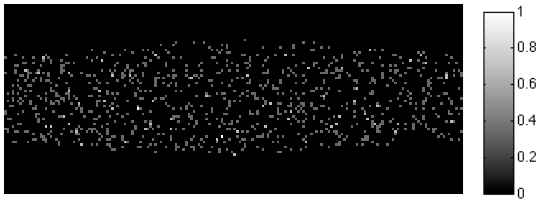
[58] Pearson, K. *Contributions to the Mathemathical Theory of Evolution*. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **186**: 343–414. Bidcode:1895RSPTA.186..343P.

APPENDIX 1: HISTOGRAMMING A PHANTOM LIST-MODE DATA WITH NOISE

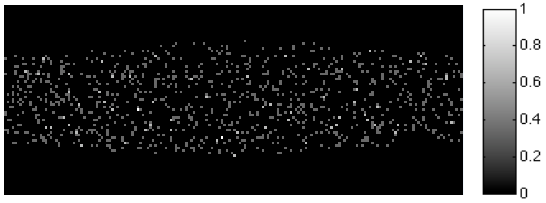
In the Section 5.1.1 has been seen the histogramming of a LM file based on a SL phantom. The data showed is without noise, for that reason it has been considerate appropriated to add the same simulation adding noise to the virtual data. The input parameters are:

- *Nevents* = 1000.
- *Npix* = 50 cm.
- *Nang* = between 0 to 180 degrees.
- *noisy* = 1.
- *planes* = 2.
- *incl* = 5 degrees.

The result for this simulation can be seen in the Figure 8.1. As in the Fig.5.1 the difference between the sinograms based on virtual data and histogrammed from the generated LM data is 0. This difference has been checked numerically applying the MSE to the generated and histogrammed sinograms. This results means that even adding interferences to the phantom, the function can generate a LM file that histograms the same sinogram without errors.

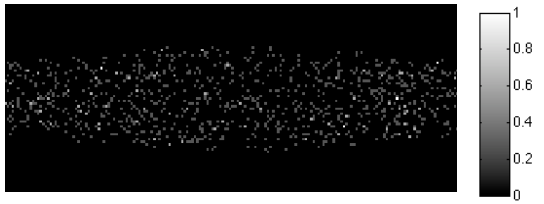


(a) Generated Sinogram based on Shepp-Logan Phantom with noise, transaxial slice.

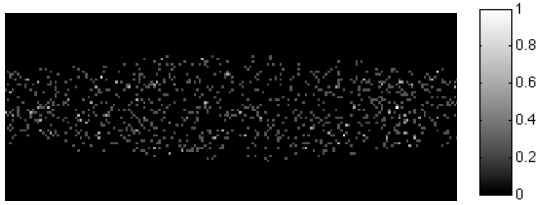


(b) Histogrammed Sinogram from List-mode data.

(c) Differences between (a) and (b).



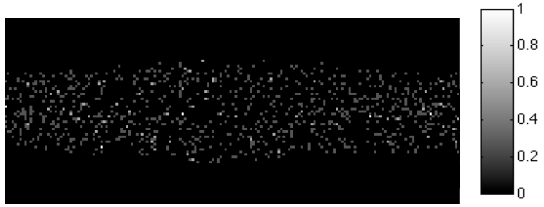
(d) Generated Sinogram based on Shepp-Logan Phantom with noise, oblique $+\theta$ slice.



(e) Histogrammed Sinogram from List-mode data.

(f) Differences between (d) and (e).

(g) Generated Sinogram based on Shepp-Logan Phantom with noise, oblique $-\theta$ slice.

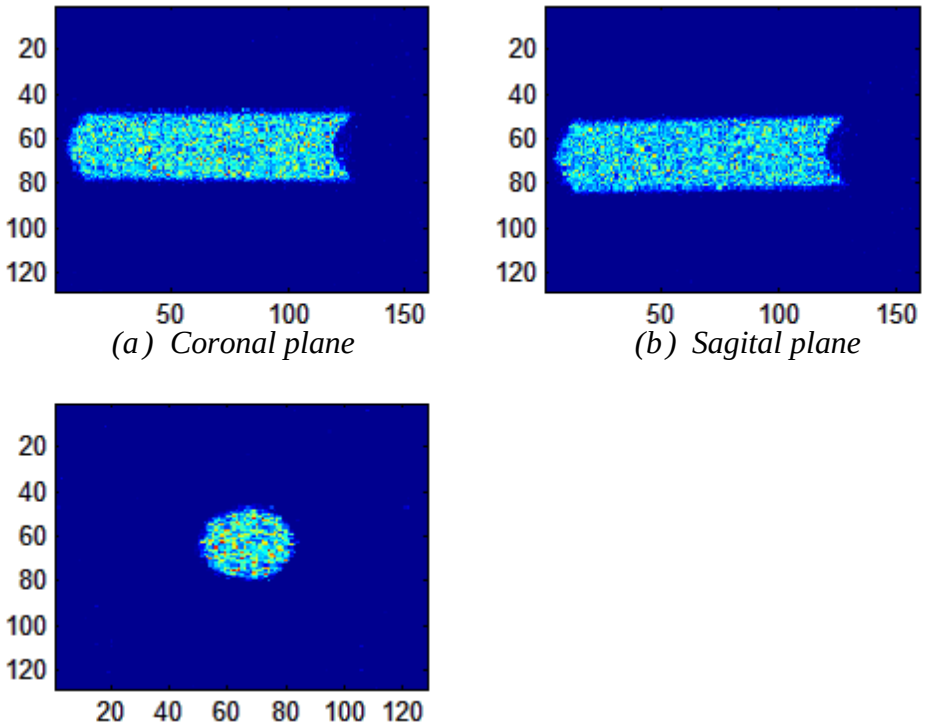


(h) Histogrammed Sinogram from List-mode data.

(i) Differences between (g) and (h).

APPENDIX 2: RECONSTRUCTING PHANTOM OF INVEON DATA

The Inveon data, on the contrary than the COMPET data, has several slices with additional information to experiment different acquisition tests. In the figure below can be seen the phantom measured with Inveon scanner. This image was used in our experiments to test the Toolbox functions. This image was selected due to the simplicity of the object which leads to reduce the processing time.



(a) Coronal plane *(b) Sagital plane*

(c) Transverse plane

Reconstruction of the Inveon phantom stored in sinograms files. The image is showed for the coronal, sagittal and axial planes.