Investigations into a positron emission imaging algorithm

Matthew Bickell

Dissertation submitted to the University of Cape Town in fulfillment of the degree of Master of Science (Physics)

Department of Physics
University of Cape Town

May 2012

Plagiarism Declaration

I know the meaning of plagiarism and declare that all of the work in the document, save for that which is properly acknowledged, is my own.

Matthew Bickell

Ethical Clearance

Ethical clearance was required for the animal scan which was conducted during this research; this clearance was obtained from the Medical Research Council, Ref. 03/11.

Acknowledgments

I would like to thank all those who assisted me throughout this research, especially my supervisors Prof. Andy Buffler and Dr. Indresan Govender for all their advice and support throughout the project, and for allowing me to steer this research in my own direction. Without them this work would not have been possible. Special thanks go to Mike van Heerden, Lui Cong and Dave Boulton for their generous assistance in preparing for and conducting experiments; to Angus Morrison, Michael Malahe and my other colleagues for the many conversations that helped to smooth over the bumps in the research, especially Dino Giovannoni for help with the initial coding process. I am grateful to Prof. David Parker and the Positron Imaging Centre for the use of their gamma camera for some of the earlier experimental work, and to the National Research Foundation, the University of Cape Town and the PEPT Cape Town research group for financial support over the past two years.

Lastly, I am very grateful to my family and friends for their constant support throughout my studies, and especially to my fiancée, Caitlin, for all the tedious proofreading, the late night coffees, being with me through the working weekends, and providing so much love and support.

Abstract

A positron emission imaging algorithm which makes use of the entire set of lines-of-response in list-mode form is presented. The algorithm parameterises the lines-of-response by a Cartesian mesh over the field-of-view of a Positron Emission Tomography (PET) scanner to find their density distribution throughout the mesh. The algorithm is applied to PET image reconstruction and Positron Emission Particle Tracking (PEPT).

For the PET image reconstruction, a redistribution of the lines-of-response is employed to remove the discrete nature of the data caused by the finite size of the detector cells, and once the density distribution has been determined, it is filtered and corrected for attenuation. The algorithm is applied to static and dynamic systems of hard phantoms, biological specimens and fluid flow through a column. In the dynamic systems, timesteps as low as 1 second are achieved. The results from the algorithm are compared to the standard Radon transform reconstruction algorithm, and the presented algorithm is observed to produce images with superior edge contrast, smoothness and representation of the physical system.

For the application of the algorithm to PEPT, the density distribution is determined for a short time slice of the data, and from this the locations of the particles are identified; this process is repeated for subsequent time slices to track the particles over time. The algorithm is successfully applied to multiple particle tracking, and up to 16 particles are tracked simultaneously. With other groups previously tracking only up to 3 particles [1]; this is the first time a PEPT algorithm has tracked such a high number of particles. The algorithm produces performs well in applications where the particles do not collide, but freely moving particles are also tracked with some success. The algorithm is found to be at least as accurate as the standard PEPT algorithm - the Birmingham algorithm [2] - and a stationary particle is located with an uncertainty of ± 1.4 mm in 3 dimensions.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Positron Emission Tomography (PET) is an imaging technique that is widely used in nuclear medical imaging and has recently also begun to be applied to industrial systems. PET provides a means to study the internal structure of a system *in situ* by imaging a radioactive material introduced into the system. In PET imaging, a β^+, or positron, emitter is used. The isotope decays and emits a positron, which travels through the surrounding material and repeatedly undergoes scattering until it loses enough energy to mutually annihilate with an electron. During this annihilation process two 511 keV γ-rays are emitted. Due to the conservation of linear momentum these two γ-rays must travel in anti-parallel directions in the centre-of-mass frame. In the laboratory frame the positron may have some momentum when it annihilates and thus the resultant γ-ray pair would not be perfectly collinear, however the effect of this on the final image is usually small. If the two γ-rays are simultaneously detected at two points in space then the line connecting the detection points defines a "line-of-response" along which the annihilation event must have occurred (assuming the γ-rays were collinear). By collecting many such lines-of-response at varying angles in a PET scanner, a 3-dimensional image of the radioactive region can be constructed using tomography techniques.

An algorithm for reconstructing images from the raw data was developed by the author during this research and is reported upon in this thesis. The algorithm itself is presented after an overview of positron emission imaging. In this overview the history and development of positron emission imaging since the 1950s through to the latest advances in this field is discussed.

The algorithm is initially presented in its general form, after which its application to Positron Emission Tomography (PET) imaging is discussed, followed by its application to Positron Emission Particle Tracking (PEPT). For PET imaging the Redistribution technique that removes the artefacts associated with the discrete nature of the data is presented, as well as the use of Fourier filters, attenuation correction techniques, and the application of the algorithm to dynamic imaging. The precursor to the PET image reconstruction algorithm, the particle tracking or PEPT algorithm, is then presented with a focus on its application to multiple particle tracking.

Experiments performed to test the algorithms are summarised along with the scanners and radioactive sources used in these experiments. The results from the PEPT and PET experiments are investigated separately, with the PEPT results highlighting the development from a single particle tracking algorithm

to a multiple particle tracking algorithm, and then extending these results to tracking a continuum of radioactivity, illustrating the need for the development of the PET imaging algorithm. Two important context studies are presented: a biological study of the brain of an unconscious vervet monkey, and the imaging of fluid flow through a column of packed ore. In both contexts static and dynamic imaging is conducted. An investigation into the influence of the various algorithmic parameters on the quality of the results is presented, the findings are discussed, and recommendations are made for future analyses.

The thesis is concluded by highlighting the successes and shortcomings of the research and discussing the possible improvements which can be implemented in the future.

In this thesis an algorithm developed entirely by the author for positron imaging is explored which uses the entire set of lines-of-response in list-mode form, thus utilising the full 3-dimensional nature of the data and realising the intrinsic resolution of the scanner. The algorithm is applied to single and multiple particle tracking as well as static and dynamic tomography. Experiments were performed using the ADAC Forte gamma camera at the University of Birmingham, U.K., for the PEPT work, and the ECAT HR++ "EXACT3D" scanner at iThemba LABS, South Africa, for both PEPT and PET experiments. The algorithm is successfully applied to multiple particle tracking and achieves results which surpass those reported upon in literature. When applied to imaging the algorithm produces results which, upon comparison with the 2-dimensional filtered backprojection, have better contrast, signal-to-noise ratio and sharper edges. The algorithm is applied to various contexts, illustrating its viability as an imaging tool.

Chapter 2

Positron Emission Imaging

2.1 PET

2.1.1 History

PET imaging began appearing in journals during the 1950s with major contributions from David Kuhl at the University of Pennsylvania, Gordon Brownell at the Massachusetts Institute of Technology and Michel Ter-Pogossian at the Washington University School of Medicine. Two papers were published independently of one another in 1951 by W. Sweet [7] and F. Wrenn *et al.* [8] that demonstrated the first use of positron annihilation events for medical imaging using NaI(Tl) crystals. Sweet and Brownell published further results in 1953 [9] on the use of opposing NaI(Tl) detectors to locate brain tumours. Brownell and his colleagues continued to contribute significantly to the development of PET imaging in the following years, in addition to building some of the first commercial clinical PET scanners [10].

In the early 1960s D. Kuhl, together with R. Edwards, developed the theory behind emission and transmission tomography [11], which led to various tomographs being built at the University of Pennsylvania. Michel Ter-Pogossian, Michael E. Phelps, and others, at Washington University School of Medicine further developed the tomograph technique, which in those days was referred to as Positron Emission Transaxial Tomography (PETT) [12].

It took two decades before PET started to gain widespread acceptance as a medical imaging tool, with the greatest obstacle to its development being the production of suitable radionuclides. In 1978 a paper was published in the *Journal of Labelled Compounds and Radiopharmaceuticals* by a group under A. P. Wolf and J. S. Fowler on the production of fluorodeoxyglucose, or FDG [13]. The radioisotope ^{18}F can be used in FDG to act as the β^+ emitter. The manner in which FDG is metabolised in the human body is representative of most energy metabolism processes, and the active compound, ^{18}F, has a very convenient half-life (109 minutes) for PET studies. Hence, the development of this compound led to increased viability of PET imaging as a tomography technique [10]. ^{18}F-FDG was tested on two normal human volunteers for the first time in 1979 by M. Reivich *et al.* [14].

The first PET scanner to use a 2-dimensional array of detectors was the PC-I constructed by Brownell

and his group in 1969 and reported in 1972 [15]. In the 1970s it became apparent that the most optimal design for PET imaging would be a ring configuration, but there were two main obstacles; the number of photomultiplier tubes necessary for the camera, and the limited space within the camera geometry. Up to that point, PET scanners had used as many photomultiplier tubes as scintillator crystals, which limited the size of the scintillator crystals considerably. In 1972, however, Burnham and Brownell presented a method which allowed the use of a single photomultiplier tube for multiple scintillator crystals [15]. This then led to the development of higher resolution ring and cylindrical scanners.

The early PET systems acquired solely 2-dimensional data by using "inter-plane septa". These are small, thin, tungsten plates several millimetres thick, placed between adjacent rings or planes in the tomograph and protruding into the field-of-view to act as shields so that only those lines-of-response that are perpendicular to the camera axis are detected and the rest are shielded by the septa. Thus, almost all of the lines-of-response are recorded by detectors belonging to the same transaxial plane [3]. This produces data in the form of a set of 2-dimensional planes. A 3-dimensional image can be formed by stacking consecutive 2-dimensional planes on top of each other. Forming the 3-dimensional image in this manner does produce a fully 3-dimensional image without loss of information, however the signal-to-noise ratio is poor compared to an image formed from fully 3-dimensional data [3, 4], and the image typically requires more smoothing to be interpretable which results in a loss of spatial resolution. Fully 3-dimensional data can be recorded by removing the inter-plane septa, allowing the camera to detect all the lines-of-response, but are much more difficult to handle and requires significantly more storage space.

2.1.2 Scintillator crystals

There is a large array of scintillator crystals used in PET systems, each with their own particular advantages and disadvantages. Many of the first PET scanners (or PETT scanners, as they were referred to in those days) used thallium doped sodium iodide, NaI(Tl), scintillators, such as the PC-I which used 127 NaI(Tl) crystals [15], the ECAT systems designed by Michael Phelps and Edward Hoffman [16], and the Donner 280-crystal ring scanner reported by Donner Laboratories in 1979 [17]. Bismuth germanate (BGO) crystals were developed in the mid 1970s and were used in PET scanners by the end of the 1970s [15, 18, 19]. The BGO crystal is still one of the most common crystals used in PET systems today because of its high stopping power (higher density and linear attenuation coefficient) [20]. Another significant advantage that this crystal has over the NaI(Tl) crystal is its high linear attenuation coefficient. This means it can be much smaller, thus allowing for more crystals to be used in a given geometry, which leads to higher spatial resolution. However, this crystal has a considerably long decay time which decreases its capability to detect large count rates [4].

Most of the various scintillator crystals used in PET scanners today are summarised in Table 2.1.

Lutetium oxyorthosilicate (LSO) has many characteristics that are favourable for PET imaging, namely, high light output, high density, large linear attenuation coefficient and short decay time, but at the expense of lower energy resolution. Furthermore, there is a naturally occurring radioactive isotope of lutetium, namely ^{176}Lu, which emits x-rays in the range 88 to 400 keV. Fortunately this does not produce a significant amount of background, since these photons can be filtered out by a suitable energy window [3]. LSO allows

Property	NaI(Tl)	BGO	LSO	GSO	YSO	BaF_2
Density (g cm^{-3})	3.67	7.13	7.40	6.71	5.37	4.89
Light output (photons per keV)	38	6	29	10	46	2
Decay time (ns)	230	300	40	60	70	0.6
Energy resolution (% at 511 keV)	6.6	20	10	8.5	12.5	11.4
Linear attenuation coefficient (cm^{-1})	0.35	0.96	0.87	0.70	0.39	0.44

NaI(Tl): Sodium iodide doped with thallium, NaI(Tl)
BGO: Bismuth germanate, $Bi_4Ge_3O_{12}$
LSO: Lutetium oxyorthosilicate doped with cerium, $Lu_2SiO(Ce)$
GSO: Gadolinium oxyorthosilicate doped with cerium, $Gd_2SiO_5(Ce)$
YSO: Yttrium oxyorthosilicate doped with cerium, $Y_2SiO_5(Ce)$
BaF_2: Barium fluoride

Table 2.1: Properties of common scintillation crystals used in PET scanners for the detection of 511 keV γ-rays. Based on Table 2.1 in G. B. Saha [3] and Table 4 in M. E. Phelps [4].

for time-of-flight recording whereby each recorded line-of-response also carries the additional information of the time of arrival of each member of the γ-pair. Using this information, one can estimate a region along the line-of-response where the annihilation event occurred, thereby improving the signal-to-noise ratio of the final image and allowing for faster convergence of iterative reconstruction algorithms [21]. Many reconstruction algorithms exist which make use of time-of-flight data, such as the algorithm presented by Defrise *et al.* [22], but numerous technical difficulties arise and so these time-of-flight systems are not common in clinical PET. However, as the technology continues to develop, especially in the production of LSO, more of these systems are appearing [23]. A variant of the LSO crystal, lutetium-yttrium oxyorthosilicate (LYSO), has been used in the commercially available Philips Gemini TF scanner [24]. Another candidate for time-of-flight PET is BaF_2,with a decay time that is sufficiently fast for this purpose; however this crystal performs poorly in other functions.

More recently, Gadolinium oxyorthosilicate (GSO) crystals have become a favourite for new PET systems because of their multiple favourable properties, including an excellent decay time and an energy resolution similar to that of NaI(Tl) [25].

Until the mid 1980s each scintillator crystal was mounted on a single photomultiplier tube, and since they were quite large this imposed a limit on how small the crystals could be made. In 1985 Burnham *et al.* presented an invention which allowed multiple scintillator crystals to be optically connected to each photomultiplier tube via a circular lightguide [26]. It was shown that from the ratio of adjacent photomultiplier tubes one can determine which crystal detected the event. In 1986 Casey and Nutt presented an improvement on this design and introduced the "block detector" [27] whereby 32 crystals were attached to 4 photomultiplier tubes, and by taking the weighted average of the signals from the four photomultiplier tubes the crystal which detected the event could be identified:

$$x = \frac{S_A + S_B - S_C - S_D}{S_A + S_B + S_C + S_D} \tag{2.1}$$

$$y = \frac{S_A + S_C - S_B - S_D}{S_A + S_B + S_C + S_D} \tag{2.2}$$

where S_A, S_B, S_C, and S_D are the signals from the four photomultiplier tubes [4]. Modern block detectors consist of four photomultiplier tubes connected to a single crystal that is sliced in a grid fashion with varying depths which effectively creates multiple crystals. Figure 2.1 shows the design of such a block detector. In this figure there are 16 detector cells per photomultiplier tube, but in some modern detectors there are as many as 144 cells per photomultiplier tube [28]. The invention of the block detector was an important step in the history of PET imaging since it allowed for many detector cells to be used in a scanner at a reasonable cost, and thus made smaller detector cells and high-resolution imaging possible.

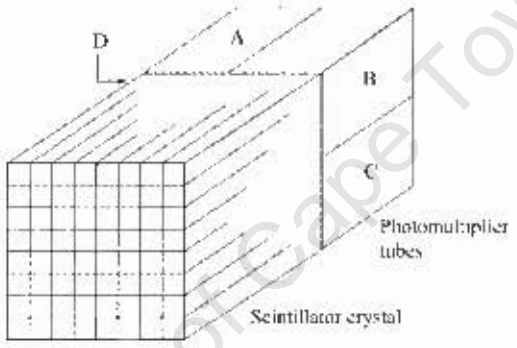


Figure 2.1: An example of a detector block with four photomultiplier tubes attached. Each block is segmented into 8×8 detector cells by slices of variable depth through the crystal, which allow for the determination of the detection event's approximate location on the crystal.

2.1.3 Data acquisition

The events recorded during a PET scan are the detection of γ-rays by the scintillator crystals. Each data point is a pair of detected events representing the coincident detection of the two back-to-back 511 keV γ-rays emitted during an electron-positron annihilation event. When a γ-ray is detected by a detector cell, an electronic window is opened for a short time, on the order of nanoseconds. If another γ-ray is detected during this time, then these two events are considered to have originated from the same annihilation event, provided the second event is within a suitable solid angle of the first event and the energies deposited by both γ-rays in the scintillator crystal fall within a certain pre-set energy window. The line joining the recorded pair of detected events is referred to as a "line-of-response", along which the annihilation event should have occurred. It is possible that the emitted γ-rays are not perfectly collinear in the reference frame of the scanner, since they are only required to be collinear in their centre-of-mass reference frame. This non-collinearity can be a maximum of 0.25° and can cause a spatial deviation from the annihilation event of

approximately 2 mm, depending on the scanner geometry [3]. There are several mechanisms by which lines-of-response can be recorded, which are presented below. Only the first of these carries useful information in reconstruction.

(a) *True* - A true line-of-response is recorded when the two γ-rays emitted during the annihilation process travel uninterrupted towards the detector cells and are detected, paired and recorded as a line-of-response. These form the majority of the recorded data and contribute meaningfully to the reconstruction.

(b) *Scattering* - At least one of the γ-rays undergoes Compton scattering by an electron in the surrounding material and experiences a change in direction of travel. If both γ-rays are detected the line-of-response formed will not pass through the position of the annihilation event, and this line-of-response will be a spurious data point.

(b) *Random coincidence* - Two annihilation events occur within a short time of each other and four γ-rays are emitted. Two particles from different events are detected within the coincidence timing window and paired by the system. The resulting line-of-response does not pass through either annihilation event and is thus a spurious data point.

Other events which can be detected by the scanner but will not result in a line-of-response being recorded are:

(a) *Single* - Of a γ-ray pair from an annihilation event, only a single γ-ray is detected by the scanner while the other is absorbed or exits the field-of-view before being detected. These events are discarded.

(b) *Multiple* - Three or more γ-rays from multiple annihilation events are detected by the scanner within one timing window, making it impossible for the system to decide which detections originated from the same annihilation event and are meant to be paired. These events are also discarded.

Once the lines-of-response have been recorded they are either stored in list-mode form where the detector pairs and the time-stamp of the event are written to file, or they are binned into sinograms, which are discussed in the following section.

Sinograms

Most PET reconstruction algorithms make use of sinograms. A sinogram is effectively a binning of the data, with each element in the sinogram representing a pair of detector cells, and it holds information regarding the number of events detected by that pair. A single sinogram is typically a matrix corresponding to all the pairs of detector cells in a particular plane or neighbouring planes, thus the image reconstructed from a sinogram will be 2-dimensional; a 3-dimensional image is formed by stacking 2-dimensional planes on top of each other. The axes of a sinogram correspond to the "offset" and "projection angle" of the detector pairs. Figure 2.2 shows the offset and projection angle of a particular line-of-response. The offset is the perpendicular distance of the line-of-response from the centre of the field-of-view, and the projection angle is the angle between the y-axis and the line-of-response. Since the detector cells are discrete, the units of

the offset and angle are also discrete and are within the ranges $r \in [-\frac{N}{4}, \frac{N}{4}]$ and $\phi \in [0, \frac{N}{2}]$, where N is the number of detector cells in a ring.

Usually only the direct sinograms (those planes perpendicular to the scanner axis), and cross sinograms (those planes where the ends of the line-of-response are in neighbouring rings) are stored during acquisition. For an N ring scanner there are a possible total of N^2 sinograms, but this can be reduced to $2N - 1$ by using only the direct and cross sinograms.

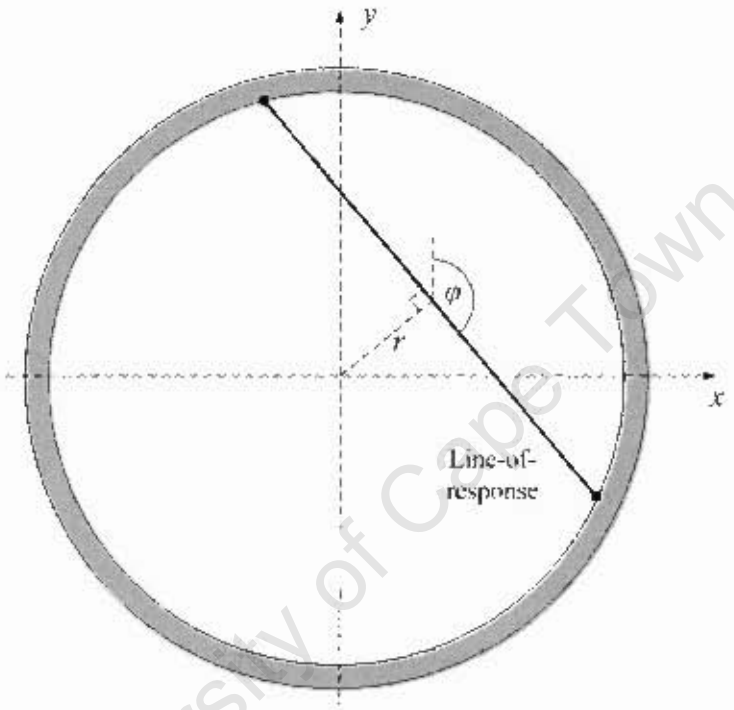


Figure 2.2: A line-of-response is binned into a sinogram according to its offset, r, and projection angle, ϕ. Each pair of detectors has a particular offset and projection angle and thus corresponds to a single sinogram element.

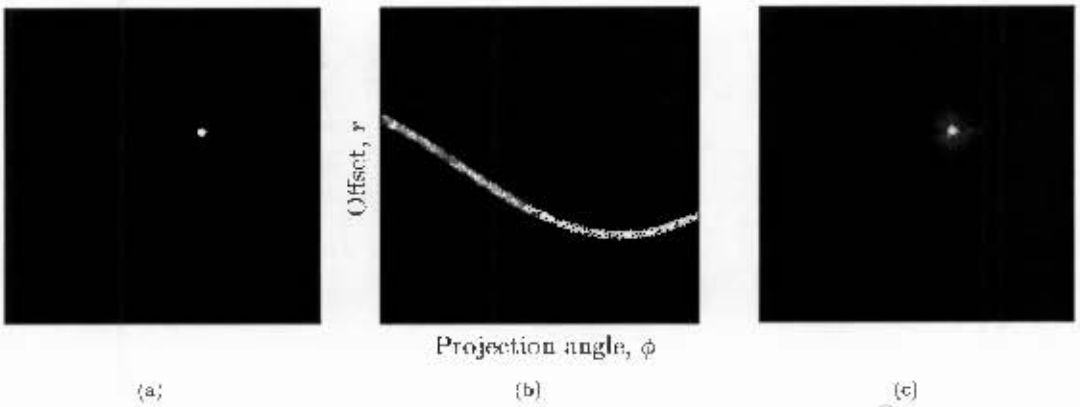


Figure 2.3: A small, circular, radioactive particle (the white dot) is shown in the field-of-view of the scanner in (a). The sinogram representation of the lines-of-response passing through this particle is shown in (b), and the subsequent reconstruction of this sinogram using a standard reconstruction algorithm is shown in (c).

2.1.4 Image reconstruction techniques

Filtered Backprojection

In 1973 David Chesler and his group at the Massachusetts General Hospital (MGH) presented the filtered backprojection technique [29], which was the first of its kind to be applied to PET and CT data [10]. By 1980 the filtered backprojection technique was being used on most PET systems [30].

The filtered backprojection algorithm uses the sinogram data for reconstruction. Each column in each sinogram is transformed into the frequency domain, convolved with a suitable filter to suppress high and low frequency noise, and then transformed back into projection space to produce the filtered sinogram. Some common Fourier filters used are,

$$\begin{aligned}
\text{Ramp:} \quad & H(\omega) = |\omega| \\
\text{Shepp-Logan:} \quad & H(\omega) = \tfrac{2\omega_c}{\pi} \sin\left(\tfrac{\pi\omega}{2\omega_c}\right) \\
\text{Hann:} \quad & H(\omega) = \tfrac{|\omega|}{2}\left(1 + \cos\tfrac{\pi|\omega|}{\omega_c}\right) \\
\text{Hamming:} \quad & H(\omega) = |\omega|\left(0.54 + 0.46\cos\tfrac{\pi|\omega|}{\omega_c}\right)
\end{aligned}$$

where ω_c is the cut-off frequency above which $H(\omega)$ is set to zero. After the filter has been applied each element in the sinogram is "backprojected" through a mesh over the scanner field-of-view along the line connecting the two detector cells to determine which voxels of the mesh the line-of-response passes through. In this way, a distribution of the activity in the plane corresponding to that sinogram is developed [4]. In the original form of the filtered backprojection algorithm, a problem arose due to the scanner's sensitivity being dependent on the location of the source. This was then avoided by using only the transaxial sinograms. Generally, the 3-dimensional image generated by backprojection algorithms were formed by stacking parallel 2-dimensional slices [4].

Further progress towards a fully 3-dimensional reconstruction algorithm was made in 1976 by Orlov [31, 32], and advances continued through the 1980s by Colsher [33], Ra *et al.* [34] and Kinahan *et al.* [35]. In 1989 Kinahan *et al.* [35] presented the technique of forward-projecting (or reprojecting) data into those sections of the field-of-view where no detectors are present in order to generate a spherically complete set of data and remove the dependence of the signal intensity on the position in the field-of-view. After this reprojection the data was then filtered and backprojected to reconstruct the image. This technique was known as the 3D reprojection algorithm (3DRP) and is still in common use today.

The sheer magnitude of fully 3-dimensional data hindered the implementation of 3D reconstruction algorithms. There were several approaches used to handle the data; the first being the improvement of the hardware systems and the optimisation of the 3D reprojection algorithm, the second being the selective sampling of data to reduce the amount of lines-of-response used in the reconstruction, and the third and most successful was the rebinning of the 3-dimensional data into a stack of 2-dimensional, transaxial sinograms. Daube-Witherspoon and Muehllehner presented a technique for rebinning the oblique data into transaxial sinograms in 1987 [36] which became known as the Single-Slice Rebinning algorithm (SSRB). Each oblique line-of-response contributes to the transaxial sinogram that passes through its axial midpoint, and hence the complete set of N^2 possible sinograms is reduced to $2N - 1$ sinograms consisting of only the direct and cross plane sinograms [36]. This rebinning technique was further developed in [37, 38] and proved to be quite successful in keeping the algorithm runtimes low; however it resulted in a loss in the image quality and spatial resolution, especially further (radially) from the centre of the field-of-view [3, 20]. In 1997 Defrise *et al.* presented the Fourier Rebinning (FORE) algorithm, whereby 3-dimensional data are rebinned into 2-dimensional data via a Fourier transform [39]. This retains much of the accuracy of the 3-dimensional data while reducing the size of the data set considerably [40]. Once the data has been rebinned it can then be analysed using any 2-dimensional filtered backprojection or iterative algorithm.

Iterative algorithms

An iterative algorithm follows a very different logic to the traditional backprojection algorithm. An initial guess is made at the distribution of the activity and the projections which would have resulted from such a distribution are generated. These are then compared to the measured projections and the differences between the two are used to correct the guess; this process is then repeated until some convergence criteria is met [3]. The challenge for an iterative algorithm is to ensure that these iterations converge to the desired image, and do so as quickly as possible.

In 1982 Shepp and Vardi introduced the maximum likelihood iterative algorithm based on the expectation maximisation (EM) algorithm presented by Dempster, Laird and Rubin in 1977 [41]. Since then the EM algorithm has played an important role in many iterative reconstruction schemes, with one of the most prominent today being the ordered-subset expectation maximisation (OSEM) algorithm, presented in 1994 by Hudson and Larkin [42]. Because of its accelerated convergence, the OSEM method was the first to make iterative methods feasible in the clinical setting [40]. With the advent of modern computers, iterative methods are becoming more and more common as the reconstruction times become shorter.

2.1.5 Image artefacts due to discrete data

Most PET scanners have detector cells which are not sensitive to the exact spatial position of the detected event and thus assign the position of any event to the coordinates of the centre of the respective detector cell. This causes periodic artefacts to appear in the final reconstructed image with a spacing between the artefacts of approximately half the width of the detector cell. This observation is not very well discussed in the literature, although algorithms which make use of the sinogram data for the reconstruction avoid these artefacts by using an appropriate interpolation.

These artefacts are a significant contributor to the limit on the intrinsic spatial resolution of the scanner, and no information can be extracted regarding structures finer than the spacing between these artefacts. A simple method of handling this discretisation is to smooth the data and/or the final reconstructed image to reduce the effect of these artefacts, or to use a mesh voxel size that is larger than the spacing between these artefacts. In this way, the artefacts are effectively averaged over. However, using such a large mesh voxel size does not allow the full realisation of the intrinsic resolution of the scanner, and smoothing inevitably results in a loss of spatial resolution. A method of handling the discrete nature of the data post-acquisition and before reconstruction has been developed and is presented in this thesis, and its effect of the final image quality is investigated.

2.1.6 Attenuation Correction

In any PET scan, attenuation of the emitted γ-rays will occur to some degree in the material surrounding the radioactive region. The likelihood of attenuation occurring is dependent on the density and Z-number (number of protons in a nucleus) of the surrounding material and is described by the attenuation coefficient.

Attenuation can occur through two main mechanisms: absorption and Compton scattering. Absorption occurs when a γ-ray interacts with a valence electron in the surrounding material and is absorbed into the electron, increasing its energy level. When the corresponding γ-ray is detected by the scanner, no line-of-response is formed since only one of the pair of γ-rays was detected. Compton scattering occurs when a γ-ray interacts with an electron and is scattered. Figure 2.4 illustrates this process. The outgoing γ-ray has a new wavelength given by,

$$\lambda_f = \lambda_i + \frac{h}{m_e c}(1 - \cos\theta), \tag{2.3}$$

where λ_i is the initial wavelength of the γ-ray, m_e is the mass of the electron, h is Plank's Constant, c is the speed of light and θ is the angle between the direction of travel of the ingoing and the outgoing γ-ray.

Any γ-ray that has been attenuated through Compton scattering is undesirable since one cannot tell (without knowing the exact initial conditions of the scattering event) what the scattering angle is. Thus, if a line-of-response were to be formed, it would most likely not pass through the position of the annihilation event. When a γ-ray undergoes Compton scattering it loses some of its energy, which allows for an effective means with which to test whether the detected γ-ray has been attenuated. By enforcing an acceptance energy window on the detected events, it is possible to filter out those that have been attenuated. Some attenuated γ-rays do still fall within this energy window and form spurious lines-of-response, and it is left

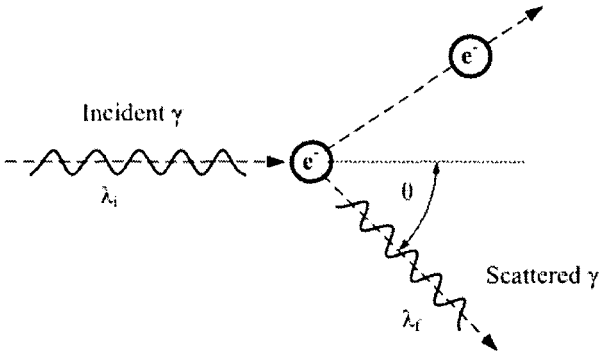


Figure 2.4: Compton scattering: an incident γ-ray scatters off an electron and continues with a new wavelength and at a new direction of travel.

to the image reconstruction algorithm to filter these out.

Attenuation can be corrected in several ways. One of the easiest of these is the "transmission-emission" method, another entails using data from a CT scan, and lastly using calculated attenuation from the known physical characteristics of the object to approximate the attenuation map.

Transmission Scan

A method that is widely used in medical imaging for attenuation correction makes use of the so-called "transmission data" [3]. The first step in this correction method is to acquire a "blank scan". This involves generating a data set using a long, thin rod source that is orientated parallel to the scanner axis and rotated along the entire inside edge of the scanner field-of-view. Sometimes a point source spiralled at the edge of the field-of-view along the entire axial length of the scanner is used. The resulting image over the entire blank scan is a cylinder with a diameter almost that of the field-of-view. The second step is called the "transmission scan". The attenuating material (in medical imaging, this is the patient) is placed inside the camera field-of-view as it would be when it is imaged, and a similar procedure to the blank scan is followed. The resulting image is again a cylinder just inside the camera field-of-view, however the image has now been affected by the attenuating material. The ratio of images from these two scans yields a correction matrix that can be applied to any subsequent image using the same attenuating material to correct for the attenuation. While this method is effective and reasonably easy to implement, it does require two extra scans to be performed.

Computer Tomography scan

A computer tomography (CT), scan uses x-rays to image material and is usually used in the clinical setting. x-rays are emitted from a rotating source and pass through the object to be imaged. During transmission they are absorbed by material inside the objects to varying degrees depending on the density of the material. Thus the resultant image can be thought of as a density map of the object. For PET scans the attenuation of

the γ-rays is also proportional to the density of the material in the object to be imaged and so by using the map from the CT scan, the attenuation coefficients of the material inside the object can be approximated. An attenuation map can be determined and used to correct the PET image. This technique is very effective but does require a CT scan. Magnetic Resonance Imaging (MRI) also has the potential to provide an attenuation map for PET, but requires a much more careful application since MRI maps are not proportional to the density of the material.

Calculated attenuation correction

To avoid the use of extra scans altogether, it is possible to calculate the attenuation correction from the emission data and then correct the data accordingly, as is presented in [43]. A similar but more rudimentary technique is to approximate the attenuation map from the known physical characteristics of the object. The use of this technique is limited to cases where the object being imaged is of a simple geometric shape and has a known density distribution. This technique, however, always requires significant simplifications and approximations, resulting in artefacts and inaccuracies in the reconstructed image.

2.2 Positron Emission Particle Tracking

Positron Emission Particle Tracking (PEPT) is where a single tracer, or particle, from a few hundred micrometres to a few millimetres is labelled with a radioisotope and tracked inside a dynamic system of flow using a modified PET scanner. The first mention of using a PET scanner for particle tracking is in [44], where the so called Birmingham algorithm is presented, and the process is further developed in [2]. Ideally, when using a point particle only two lines-of-response would be necessary to triangulate the position of the tracer, since it would be where the two lines intersect. In reality however, the lines-of-response do not necessarily pass exactly through the centre of the tracer and hence if any two lines-of-response do intersect, it is not necessarily at the tracer location. The Birmingham algorithm uses several hundred lines-of-response for each triangulation and determines the point that minimises the sum of the perpendicular distances from that point to each line-of-response. The number of lines-of-response, N, to be used for each triangulation is specified by the user during initialisation. The algorithm then triangulates the approximate particle location for each set of N lines-of-response in two steps; the first is to find an approximation to the tracer location using all N lines-of-response, and then the furthest lines-of-response from this point are discarded such that a predefined fraction, f (also defined by the user during the initialisation), of the lines-of-response remain. This fraction, f, can be thought of as the proportion of unscattered, or true, events within the data set.

The Birmingham group are based at the Positron Imaging Centre at the University of Birmingham, UK, and have been pioneering PEPT research since 1984 when they acquired their first of many PET cameras [44] constructed by Rutherford Appleton Laboratory. It consisted of a pair of multi-wire proportional chambers, each having a sensitive area of 600×300 mm^2, and operated in coincidence. This camera was replaced in 1999 by a Forte gamma camera constructed by ADAC Laboratories [45]. The Forte camera is made up of two heads each containing a single NaI(Tl) crystals with dimensions $500 \times 400 \times 16$ mm^3.

PEPT triangulation algorithms have been used with great success for many applications; Figure 2.5 shows

the results of tracking a tracer inside a tumbling mill partially filled with particulate material, the path of the tracer over a long time is shown, and Figure 2.6 shows the path of a small tracer in a hydrocyclone, as studied by Chang *et al.* [6]. Other applications of PEPT include floatation cells, flow in fluidised beds, mixing in powder mills, paste flows, and many more. The advantages of PEPT over PET in these applications are discussed in [46] and it is shown that in such systems, PEPT provides more information regarding the parameters of the flow and does so more efficiently than PET imaging. In the past decade many applications of PEPT have been explored. Scanners used for PEPT are almost entirely medical PET scanners that have been adapted for the purposes of PEPT, but with industrial demand several dedicated industrial PEPT scanners have been constructed and a portable PEPT camera was reported by Ingram *et al.* in [47]. The second dedicated PEPT research facility in the world, PEPT Cape Town, is based in Cape Town, South Africa, at iThemba LABS [5]. This group runs the Siemens ECAT HR++ "EXACT3D" scanner which was constructed to be one of the most sensitive PET scanners in the world [48].

For PEPT studies small radioactive tracers need to be manufactured. These tracers are usually between a few hundred micrometres and a few millimetres in diameter, but there is a drive to produce sub 60 μm tracers for use in applications such as floatation cells where the particles that the tracer is imitating are of this scale. Large tracers ($\gtrsim$ 1 mm) can be produced by placing a drop of radioactive fluid inside a casing and sealing it off with a suitable epoxy, or by bombarding a target tracer with a charged particle beam such as ^{3}He. For smaller particles, ion-exchange methods are usually used. This is where weak- or strong-based anion exchange resins are immersed in a solution containing radioactive ions, and these ions are then absorbed by the resin [49]. A technique to improve the uptake of these ions into the resin is presented by Fan *et al.* in [50], whereby metallic ions are introduced onto the surface of the tracer to make it more susceptible to ion exchange.

Figure 2.5: Trajectory of a single 2 mm tracer in a rotating drum, or tumbling mill, as detected by the ADAC Forte gamma camera and analysed using the Birmingham algorithm [5].

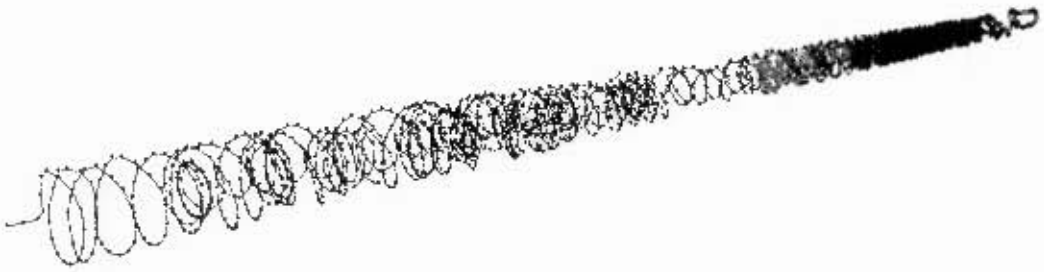


Figure 2.6: Trajectory of a single tracer in a hydrocyclone, each dot represents a tracer location [6].

2.2.1 Multiple Particle Tracking

The Positron Imaging Centre research group worked on an algorithm for tracking multiple particles based on the Birmingham single particle tracking algorithm, and the result was reported in 2006 [31]. The method they developed requires that the various particles be activated to different activities. This allows for the unique identification of the separate particles. The triangulation algorithm is applied to the data and the particle with the highest count rate will be identified by the algorithm. The lines-of-response associated with this particle are then removed from the data set and the procedure is repeated, identifying the second most active particle next, and so on. In this manner, all particles are located in order of decreasing count rate.

In 2007 Yang *et al.* published another paper that improved upon this algorithm [1]. In this paper they address the issue in the previous algorithm of the particles needing to be kept at a fixed distance from each other. They present an improved algorithm that can track multiple freely moving particles. This new algorithm accounts for the fact that the count rate detected by the camera varies with the particle position within the field of view of the camera. Hence the strongest particle (as found in the previous algorithm) might not have the highest count rate if it were to be located near to the edge of the camera, and the particles would not be able to be uniquely identified and tracked. This is handled by normalising the count rate over the entire field of view, using control experiments of single stationary particles at specific positions in the field of view. Thus, by multiplying the detected count rates by a suitable factor, depending on the particle's position, one can ensure that the multiple particles will retain their relative activity once detected. However, this algorithm allows for only three particles to be tracked simultaneously, since the spacing in the activities of the particles needs to be sufficiently large for the particles to be distinguishable; and there is a limit to the total activity that can be present inside the scanner before the detector cells become flooded. In the case of rigid rotation and some other applications it is sufficient to track only three particles, but there exist many applications where the tracking of more than three particles is necessary.

Chapter 3

The Line Density Algorithm

The Line Density algorithm utilises the statistical nature of the data to build up a distribution of the likelihood of an annihilation event occurring at some location. This algorithm is similar to a family of standard PET reconstruction algorithms known as "filtered backprojection" methods. While many standard backprojection algorithms use the sinogram data for the reconstruction, the Line Density algorithm analyses the data in list-mode form. The only information that is stored in the list-mode data is which two detector cells observed the γ-ray pair, and the time stamp of that detection event. These two detector cells define a line-of-response connecting the centres of the two detectors, along which the annihilation event must have occurred. The Line Density algorithm as applied to particle tracking has been presented in a paper by the author and others, in the journal of Nuclear Instrumentation and Methods in Physics Research A [52].

The algorithm was written entirely in MATLAB 7.11.0 (R2010b), which makes use of parallel processing on multiple cores to decrease the runtime. This parallelisation is performed automatically by MATLAB on certain predefined functions; for example any vectorised operations, matrix multiplication, sorting or searching, etc.

3.1 The Generalised Algorithm

3.1.1 Algorithm Parameters

A strength of the Line Density algorithm is in how few imposed parameters are required. The concept of an "imposed" or "arbitrary" parameter is important to be aware of when working with a physical system. An imposed parameter is one which does not represent a naturally occurring characteristic of the physical system. For example, when modelling a swinging pendulum, the mass of the pendulum is a "natural" parameter, while the time-step between iterations in the computational model is an "imposed" parameter. Imposed parameters can have a substantial effect on the outcome of the algorithm (as can natural parameters), and the extent of this effect can be difficult to judge. While natural parameters are set based on the corresponding physical characteristic in the system, imposed parameters need to be set during the initialisation of the analysis. Therefore it is usually necessary to optimise the imposed parameters, and if there are several

such parameters present then this optimisation can be difficult, time-consuming and computationally costly - usually the number of iterations necessary to optimise multiple parameters grows exponentially as the number of parameters increases. Thus, a good algorithm is one with minimal imposed parameters.

The Line Density algorithm has two imposed parameters: the size of the mesh over the field-of-view of the camera, and the number of lines-of-response used to generate an image. Both of these parameters are easily optimisable, and this optimisation is usually transferable from one analysis to the next. The mesh size has an important influence over the final resolution of the algorithm, but modern computational hardware has made it possible to use a mesh size that is finer than the intrinsic resolution of the scanner (approximately 4.5 mm), which does not make its influence negligible but it does reduce its effect considerably. The mesh size has a large dependence on the available storage capacity of the computational system being used, since the memory usage of the algorithm is proportional to the cube of the mesh size. The number of lines-of-response used directly affects the quality of the resultant image, and the general rule is that the more lines-of-response used, the better the image quality will be, since this improves the signal-to-noise ratio and statistics of the final image. Since the Line Density algorithm is a "one-pass" algorithm; meaning that it only passes through the list-mode data once; the runtime of an analysis is linearly proportional to the number of lines-of-response used for the analysis.

The only other parameter of the algorithm is the size of the field-of-view of the scanner (which is always constant), and while this does also have an effect on the memory usage of the algorithm it cannot be optimised.

3.1.2 Initialisation

During the initialisation of the algorithm the user can set the initial parameters, particularly the mesh size and the number of lines-of-response to use, and declare which data file is to be used. An example of the list-mode file in ASCII format is shown in Figure 3.1. The coordinates (x_A, y_A, z_A) and (x_B, y_B, z_B) are the coordinates of the two detectors cells that detected the γ-ray pair, and the time-stamp in the last column is the millisecond time-stamp assigned to each line-of-response. Because many lines-of-response are recorded every millisecond there can be many within an individual time-stamp, but these are all recorded chronologically and can be treated as such (which can occasionally be important in PEPT analysis).

The user has the option of defining a "region-of-interest" where the algorithm will only set up a mesh over this region and not over the entire field-of-view. This is employed to reduce the memory usage of the algorithm, which can be quite substantial when the mesh size is fine. This region-of-interest is defined in real spatial coordinates with the centre of the camera field-of-view being the origin of the coordinate system. In fact, the centre of the field-of-view of the camera is set at (0.0, 0.0, -2.4) mm relative to the coordinate axes of the lines-of-response in the ASCII data files. This is due to the fact that the detector cells are discrete and that there is an even number (48) of transaxial planes, so the actual centre of the field-of-view will be in between two planes; namely, planes 24 and 25. However, the lines-of-response are centred on the "zeroth" plane, which is half a plane thickness (i.e. 2.4 mm) from the actual centre of field-of-view. This could be remedied by shifting all the lines-of-response in the axial direction by half of the thickness of a plane, or to simply define the centre of the field-of-view to be (0.0, 0.0, -2.4) mm relative to the coordinate axes of

```
This file contains line generated from att01 _1.lm
for Start time = 0 msec to Stop time = 30000 msec.

      xA        yA        zA        xB        yB        zB      time
  193.94   -372.54    -33.95   -324.67    266.43    106.70         0
  315.78   -276.92    -43.65   -411.01     86.41    101.85         0
  407.96    -99.82   -116.40   -139.32   -396.22   -116.40         0
  384.43    169.16    -14.55   -411.93    -81.95     82.45         0
  401.91    121.92    -58.20   -303.38   -290.44     82.45         0
  407.96    -99.82     67.90   -411.93     81.92    -19.40         0
  407.96    -99.82     77.60   -396.21   -139.35    -53.35         0
  411.93    -81.93    106.70   -413.62    -72.95    -63.05         0
  181.64    378.69    -63.05   -104.26   -406.85     38.80         0
  210.01   -363.73     92.15   -341.40    244.63    -77.60         0
  412.80    -77.43    -63.05   -349.21   -233.35     24.25         0
  413.62     72.94     29.10   -419.98      4.57     24.25         0
  413.62     72.94     29.10   -156.50    389.75     43.65         0
  404.48    113.13      0.00   -380.64   -177.51     43.65         0
  147.95   -393.08     63.05   -177.52    380.64    -29.10         0
  378.69   -181.64     58.20   -419.97     -4.60    -53.35         0
  269.98   -321.73      0.00   -354.23    225.65     29.10         0
  406.85   -104.27     82.45   -393.09    147.93    -29.10         0
```

Figure 3.1: An example of the list mode file in ASCII format, as output by the HR++ scanner, which is input into the Line Density algorithm.

the lines-of-response. The latter approach was utilised in the analysis of the data during this research. At this point in the initialisation, the geometry of the physical system is defined, using the region-of-interest as discussed above.

Finally, the lines-of-response are imported into the MATLAB workspace. If the data files are not too large (up to 4 or 5 million lines-of-response is manageable) then the entire data can be read in, but as the experiment times get longer and the data files get larger, it may be prudent to import only subsets of the data at a time, which is permitted and handled by the code. The imported data is sorted into a structure field to allow for efficient use of the information. The redistribution technique, as discussed in section 3.3.1 is applied to the raw data at this point in the initialisation.

During the initialisation, it is possible to generate lines-of-response instead of using real data, which is a useful tool to test the algorithm in ideal situations. The manner with which the data is generated will be discussed in the next section.

Towards the end of the initialisation, all the information pertaining to the current analysis regarding the data file, parameters, etc., is stored in a structure to be saved in memory (along with the final result at the end of the analysis) for future reference.

3.1.3 Generating Data

In order to test the algorithm it is often useful to have ideal data under a controlled situation. However, this is very difficult to achieve with a physical system, and thus it is sometimes necessary to generate data. During the initialisation, the user has the option to generate data rather than use real data. The algorithm that was developed to generate data uses a simplified Monte-Carlo technique. Generally, Monte-Carlo algorithms aim to replicate the physical situation as closely as possible, thus a true Monte-Carlo algorithm for PET data

would model the emission of a positron from the radioisotope, its path through the surrounding material until annihilation with an electron, and the flight of the resulting back-to-back γ-rays with possible attenuation as they move away from the point of annihilation. This would be repeated for millions of events to generate a full set of list-mode data. For the purpose of testing the Line Density algorithm, however, there is no need to model this entire process. The process was simplified by only considering where the annihilation events occur and assuming that these are uniformly distributed throughout the volume of interest. Attenuation and scatter of the γ-rays as they move through the surrounding material was not taken into account.

Usually, a spherical volume is generated. The locations of the annihilation events in polar coordinates were generated in such a way that these locations were uniformly distributed throughout the volume, according to the following formulae:

$$r = \sqrt[3]{\rho_1} R \tag{3.1}$$

$$\theta = \arccos \rho_2 \tag{3.2}$$

$$\phi = \pi \rho_3, \tag{3.3}$$

where R is the radius of the sphere, and

$$\rho_1 \in [0, 1] \tag{3.4}$$

$$\rho_2 \in [-1, 1] \tag{3.5}$$

$$\rho_3 \in [-1, 1]. \tag{3.6}$$

The Cartesian coordinates of these locations is then given by:

$$x = x_0 + r \sin\theta \cos\phi \tag{3.7}$$

$$y = y_0 + r \sin\theta \sin\phi \tag{3.8}$$

$$z = z_0 + r \cos\theta, \tag{3.9}$$

where the point (x_0, y_0, z_0) is the centre of the generated volume. The direction of flight of the resulting γ-rays is determined by using the same formulae for θ and ϕ as in 3.2 and 3.3 to generate a set of uniformly distributed points on a shell that is centred on the centre of the field-of-view at a fixed radius, and by pairing each of these up with an annihilation point inside the volume. Each line is then extended to find its second intersection with the shell and hence the other endpoint of the line-of-response. Thus a set of uniformly and spherically distributed lines are generated emanating from within a spherical volume.

To record the lines-of-response one must determine if and where the γ-rays intersect the scanner detectors. This is done by calculating the value of each line's z-coordinate at the point where the line intersect the cylinder of detectors. If this z-value lies outside of the axial extent of the scanner then this line is discarded. All remaining lines are recorded as lines-of-response. One more simplification is made here: it is assumed that the detecting surface of the camera is continuous and the direction of the lines-of-response is exactly the direction generated during the Monte-Carlo process. In reality, the PET scanner's detector cells are finite

in size and the endpoints of any line-of-response is recorded as being the centre of the detector cells which detected it, introducing the discreteness to the data which has already been discussed. Thus, the result of the Monte-Carlo process produces data which is similar to the real data after the redistribution technique is applied.

Once the lines-of-response have been recorded they are assigned an arbitrary time-stamp and are stored in a suitable structure so that they can be parsed on to the rest of the code and treated as a usual data set.

3.1.4 The Algorithm

After the initialisation process, the line-of-response data structure is parsed to another function to perform the image reconstruction. This function is responsible for almost all of the computations and runtime of the code. The first step in this function is to generate the 3-dimensional mesh over the field-of-view of the camera. This mesh is always Cartesian and usually cubic, and it is represented in the algorithm by a 3-dimensional matrix where each element represents a voxel in the mesh. This matrix is referred to as the density matrix, since it will hold the information regarding the distribution of the density of the lines-of-response throughout the field-of-view.

Each line-of-response is treated as a vector of the form,

$$\vec{r} = \vec{a} + \mu\vec{d}, \tag{3.10}$$

where $\vec{r}$ is the position vector for a point along the line-of-response, $\vec{a}$ is the starting point (i.e. either of the end points defining the line-of-response), μ is an arbitrary parameter, and $\vec{d}$ is the direction vector which is defined as the vector pointing from the starting point $\vec{a}$ to the other end point of the line-of-response. The voxels inside which each of the endpoints of the line-of-response lie are determined using the quotient of the coordinates of the endpoints and the voxel size in the corresponding dimension.

The algorithm now defines a counting system corresponding to each direction (x, y and z) for each line-of-response. The unit of the counting system is related to the length of the line-of-response through the mesh voxels in each respective direction. Each member of the counting system then represents an intersection of the line-of-response with the mesh, allowing for the identification of the voxels through which the line-of-response passes. The sets of intersections in each direction determined by this counting system are given by:

$$\mathbf{S}_x = \left\{ \vec{p}_x \in \mathbf{F} : \vec{x} = \vec{a} + \mu\hat{d}_x,\ \hat{d}_x = \frac{v_x}{b_x - a_x}(\vec{b} - \vec{a}),\ \mu \in \mathbb{N} \right\} \tag{3.11}$$

$$\mathbf{S}_y = \left\{ \vec{p}_y \in \mathbf{F} : \vec{y} = \vec{a} + \mu\hat{d}_y,\ \hat{d}_y = \frac{v_y}{b_y - a_y}(\vec{b} - \vec{a}),\ \mu \in \mathbb{N} \right\} \tag{3.12}$$

$$\mathbf{S}_z = \left\{ \vec{p}_z \in \mathbf{F} : \vec{z} = \vec{a} + \mu\hat{d}_z,\ \hat{d}_z = \frac{v_z}{b_z - a_z}(\vec{b} - \vec{a}),\ \mu \in \mathbb{N} \right\} \tag{3.13}$$

where $\mathbf{F}$ is the set of coordinates within the field-of-view of the camera and v_i, $i \in \{x, y, z\}$, is the size of the mesh voxel in each direction. The points $\vec{a}$ and $\vec{b}$ are the two points inside the field-of-view where the line first intersects the mesh. Points $\vec{a}$ and $\vec{b}$ can thus be thought of as the zero- and end-points of the counting

system, respectively. This process is illustrated in Figure 3.2. The direction vector $\hat{d}_i$ is in the direction from $\vec{a}$ to $\vec{b}$, but with magnitude such that its component in the i direction is equal to the width of the voxel in that direction.

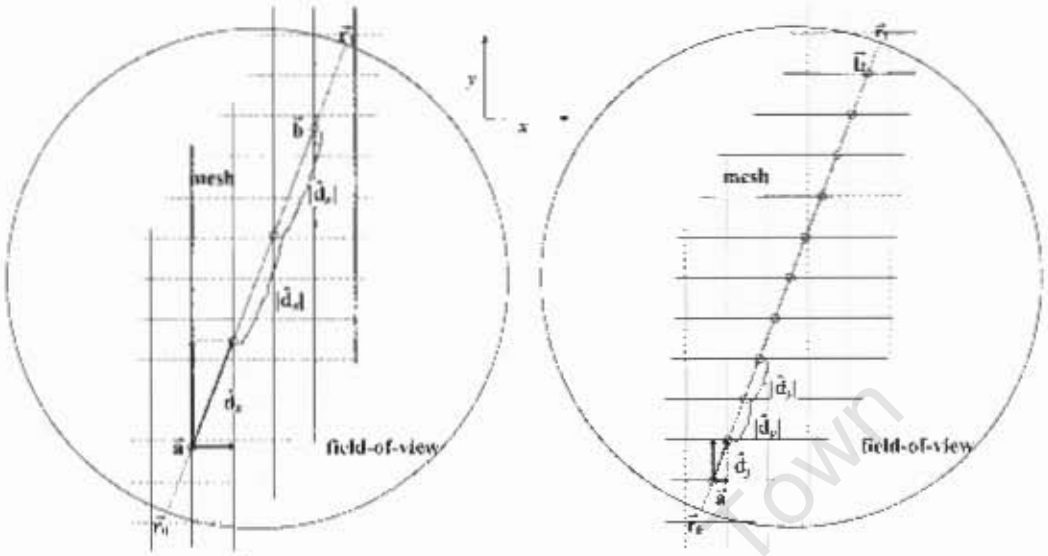
For a particular line-of-response, the sets defined above contain all the points along that line-of-response which lie on the mesh. It is then a simple procedure of dividing these points by the voxel size in the respective coordinates to determine the matrix indices of the voxels through which that line-of-response passes.

In practice the algorithm does not need to follow the above procedure for all 3 directions. If one carefully chooses the appropriate direction then all the mesh intersections can be determined along that direction and the other 2 directions need not be considered. Figure 3.2 illustrates why this is the case. By considering only the y-direction all the mesh intersections would be accounted for and the x-direction can be ignored. To determine which direction should be used for a particular line-of-response, the algorithm considers the lengths of the components of $\hat{d}_i$. The direction where the component in that direction is the longest should be used to determine the mesh intersections.

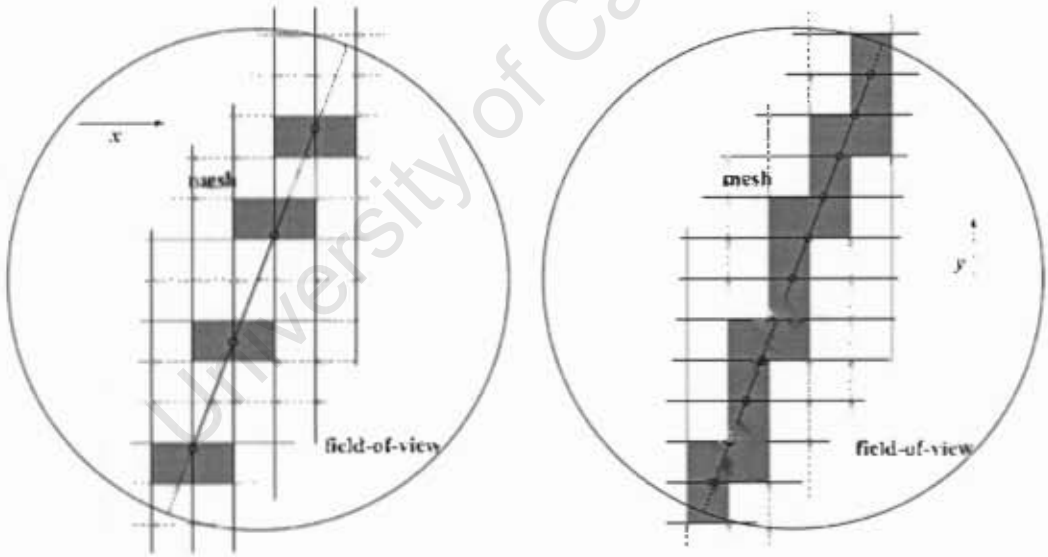
After this procedure, a set of element indices of the mesh is produced. Duplicates in this list do occur, and these are removed. Finally, the density matrix is then incremented at these indices, and this density matrix represents the reconstructed image of the data. This matrix is usually normalised to its maximum to make comparison of different images easier; this option is set by the user during the initialisation.

The resulting matrix is then returned to the parent function for further analysis. All that is left to do is to apply attenuation correction techniques if necessary, filter the image to remove noise and artefacts, and export it to a suitable file format for viewing. The filtering and attenuation correction are described in the following sections of this chapter.

Most of the code has been vectorised, and those sections which have not been vectorised employ only a single loop over the lines-of-response. This vectorisation allows for efficient execution, especially on multiple processors.



(a) Demonstrating how $\vec{d}_i$ is defined in the x-direction case (left) and y-direction case (right). In the left panel notice that the length of the component of $\vec{d}_x$ in the x-direction is one voxel width. Thus, each integer multiple of $\vec{d}_n$ will have its endpoint on a mesh intersection at those planes perpendicular to the x-axis. In this diagram $\vec{r}_0$ and $\vec{r}_1$ are the end-points of the line-of-response.



(b) Indicating where the planes perpendicular to the x-axes (left) and y-axes (right) are pierced, as specified by the red circles.

Figure 3.2: Illustrating how the algorithm determines which voxels a line-of-response passes through. The shaded boxes are the voxels that are ticked. Only the x- and y-directions are shown as the z-direction is similar to these.

3.1.5 Psuedo-code

The pseudo-code for the generalised form of the Line Density algorithm is given below.

Main

- Initialise parameters: camera geometry, mesh size, data file, length of run, region-of-interest
- Generate data if applicable
- Store run information for future reference
- Apply **Meshed Line Density** algorithm to lines-of-response; get density matrix

Meshed Line Density

- Generate the 3D mesh over the field-of-view, or the region-of-interest if appropriate
- Determine $\vec{a}$ and $\vec{b}$ from the endpoints of the lines-of-response
- Determine the scaled direction vector, $\hat{d}_i$, for each line-of-response in each direction
- Determine which direction should be considered to determine mesh intersections
- **loop** over all lines-of-response
  - Determine the points where this line-of-response pierces each plane in this direction
  - Determine the matrix index corresponding to each of those points and remove any duplicates
- Concatenate the lists of indices from all lines
- Increment the elements in the 3D mesh which correspond to these indices, thus forming a "density matrix"
- Normalise this density matrix, if necessary
- Return the density matrix

3.2 Application to PEPT

The Line Density algorithm can be applied to the tracking of single or multiple tracers moving within the field-of-view of the scanner, this technique is known as PEPT. Since the location of the tracer is in that region of space through which most of the lines-of-response pass, the Line Density algorithm can be used to determine that region of highest density. Considering a particular time slice of the data (usually a few milliseconds and thus only including several hundred lines-of-response), the algorithm determines the density matrix for those lines-of-response. This approach is illustrated in figure 3.3 for 50 lines-of-response over a coarse mesh. The voxel containing the highest count, and thus the source of the γ-rays, is clearly identified.

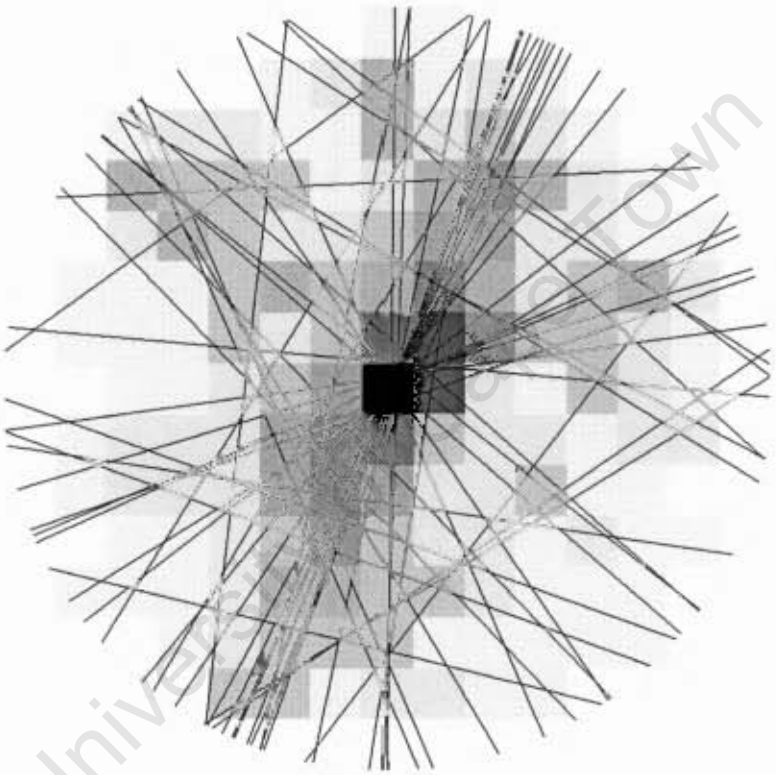


Figure 3.3: Illustrating the analysis of approximately 50 lines-of-response using a course mesh. The region of highest density is indicated by the darkest colour, and clearly encompasses the location of the tracer.

The peak of the density distribution is identified and three slices (one voxel width thick) through this peak along each of the three dimensions are considered. Figure 3.4 shows the slices of the distribution of counts for 4 ms of a typical data set. A Gaussian fit is then applied to these distributions, which is found to be reasonable provided the size of the tracer is smaller than the size of the mesh voxels. The best approximation of the location of the tracer is given by the centroids of these three Gaussian fits, while the uncertainty in the location is proportional to the full-width-at-half-maximum (FWHM) of the Gaussians. By repeatedly applying this method to subsequent subsets of the data, the chronological locations of the tracer may be

realised.

Gaussian distributions are used as a fit to the data without rigorous theory to justify their use. Through the observation of many data sets and the corresponding fit parameters, it has been found that the Gaussian distribution provides a technique that is quite appropriate for the data. This is especially true for data where the tracer was smaller than the mesh voxel size and could thus be approximated to a point particle relative to the mesh. The fit became worse if the tracer was of a size that is comparable to or larger than the mesh voxels. Monte Carlo simulations were also used to test this assumption since ideal data for a point particle could be created and the Gaussian fit tested. Figure 3.4 shows an example of real data with their Gaussian fits to illustrate the goodness of the fit.

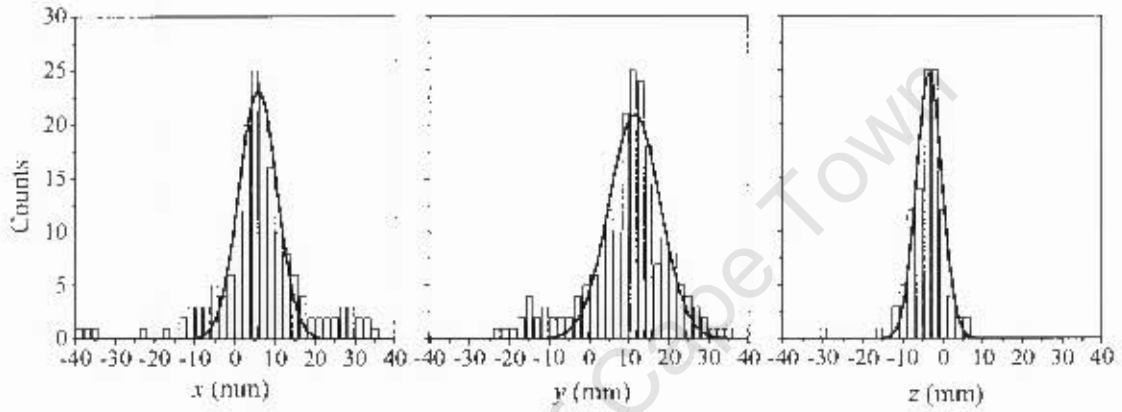


Figure 3.4: Distributions of voxel counts for about 500 lines of response from a typical data set, with their Gaussian fits. The centroids of the fitted Gaussians provide best estimates of the (x, y, z) coordinates of the tracer.

3.2.1 Multiple Particle Tracking

The Line Density algorithm was developed to allow for the simultaneous tracking of multiple particles. When more than one tracer is being tracked the algorithm requires that the approximate initial positions of the tracers be specified. The simplest way of providing this information is by defining regions within the field-of-view of the camera where only one tracer is present at the start of the run, which is readily achieved by a visual inspection of the first time slice of the data. The algorithm then isolates all lines-of-response in each of these regions and determines the location of the tracer in each region, using the Line Density algorithm for single particles. For subsequent iterations the algorithm uses the previous locations of the tracer to estimate which lines-of-response should be used to calculate the next location. This is achieved through extrapolation: the previous two locations of the tracer are used to approximate the velocity of the tracer and hence an approximate next location. A cubic region of a predefined size is created about this extrapolated location and all lines-of-response passing through this cube, or region-of-interest, are considered for the next iteration of the tracer's location. In order for the cube to encompass the tracer location at the next time

step, the dimensions of the cube must satisfy

$$d > |a|(\partial t)^2, \tag{3.14}$$

where d is the length of a side of the cube, $|a|$ is the magnitude of the acceleration of the particle, and ∂t is the time step used. For example, if a cube of dimension 50 mm is used with a time step of 50 ms, then the acceleration must be less than 20 ms^{-2}. If the tracers are moving too erratically then a shorter time step or a larger cube can be used, taking care not to enclose another tracer within the cube. The acceleration is usually observed to be such that this constraint is easily satisfied. By using this method of extrapolation to obtain the next position of the tracer, each tracer is effectively identified at each time step, avoiding the need to rely on physical attributes of the tracers to distinguish between them.

Any multiple particle tracking algorithm is susceptible to the fact that if two particles are very close to each other, the lines-of-response cannot be uniquely associated with either particle. The Line Density algorithm is no exception; therefore the applications for which this algorithm is used must be carefully chosen to avoid particle collisions. This excludes applications with multiple freely moving particles. It is possible to handle such systems by carefully analysing each collision to observe how the particles move away from each other after the collision has occurred; however this has not been implemented in the algorithm presented here since we are largely concerned with applications where collisions between particles are avoided. Collision management will be implemented in a future development of this code.

3.2.2 Pseudo-Code

The pseudo-code for the Line Density algorithm as applied to PEPT with multiple particle tracking is given below.

Main

- Initialise parameters: camera geometry, mesh size, data file, length of run, number of particles, time step, number of time steps
- **loop** over location counter, i
  - **loop** over number of number of particles
    * extrapolate particle's next location using previous locations
    * take subset of LORs passing near to the extrapolated location
    * **if** $i == 1$,

      consider LORs passing near to the initial location as defined by the user
    * **else** $(i > 1)$,

      extrapolate particle's next location using average velocity (initial velocity set to zero)

      consider LORs passing near to the extrapolated location
    * apply **Meshed Line Density** algorithm (as in the generalised case, section 3.1.5) to the subset of LORs
    * locate the peak of the density matrix and consider three 1D slices through this peak parallel to each coordinate axis
    * fit a Gaussian to each of these slices
    * return the fitted Gaussians' means and widths to describe the location of the particle with its algorithmic uncertainty
    * check validity of results by examining whether the particle has moved unrealistically far
    * store valid results
- Display results

3.3 Application to PET

3.3.1 The Redistribution Technique

During the electron-positron annihilation process γ-rays are emitted isotropically and could therefore strike any point on the face of a particular detector. Since the detector cells are not sensitive to the exact spatial position of the detected event on the face of the cell, each event is assigned the coordinates of the centre of the respective detector cell. This imposes a discrete nature on the data with the lines-of-response being discretely distributed about the edge of the scanner ring. This discrete nature produces periodic artefacts in the final reconstructed image, particularly if the resolution of the image is finer than the period of these artefacts. Near the centre of the field-of-view, the lines-of-response cluster in periodic regions with a peak-to-peak separation of half the detector cell width. Figure 3.5 shows a representative example of this discrete nature of the lines-of-response in the axial and transaxial directions.

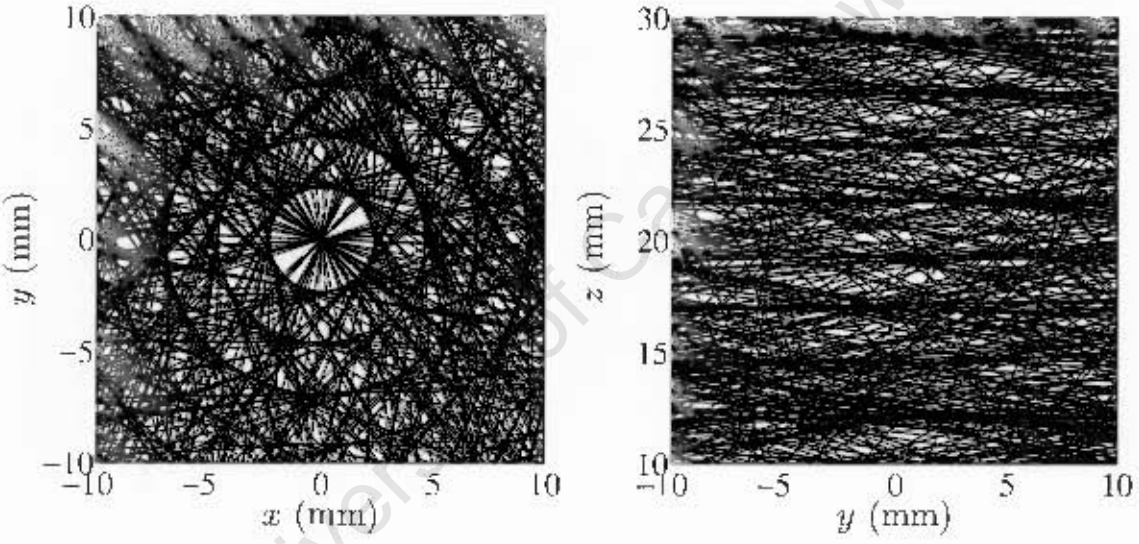


Figure 3.5: The detector cells, and thus the lines-of-response, are discrete in nature, which appears in the periodic clustering of the lines-of-response. On the left is an axial projection of the lines of response near the centre of the field of view showing concentric clustering, and on the right is a transaxial projection showing planar clustering. This data was recorded by the HR++ scanner using an off-centre point-like source.

The redistribution technique performs a small random shift to the ends of each line-of-response, so that the lines-of-response are approximately uniformly spread over the face of the detector cell. The shift is

defined by:

$$x_f = x_i + \mu_1 \frac{y_i}{\sqrt{x_i^2 + y_i^2}} \tag{3.15}$$

$$y_f = y_i - \mu_1 \frac{x_i}{\sqrt{x_i^2 + y_i^2}} \tag{3.16}$$

$$z_f = z_i + \mu_2 \tag{3.17}$$

where $\mu_{1,2}$ is a uniformly distributed random number in the range $[-2.425, 2.425]$. The shift in (x_i, y_i, z_i) is only 2-dimensional since it is over the face of the detector and hence only two random values are required for the shift. Figure 3.6 illustrates this shift, and Figure 3.7 shows the result of this redistribution of the lines-of-response.

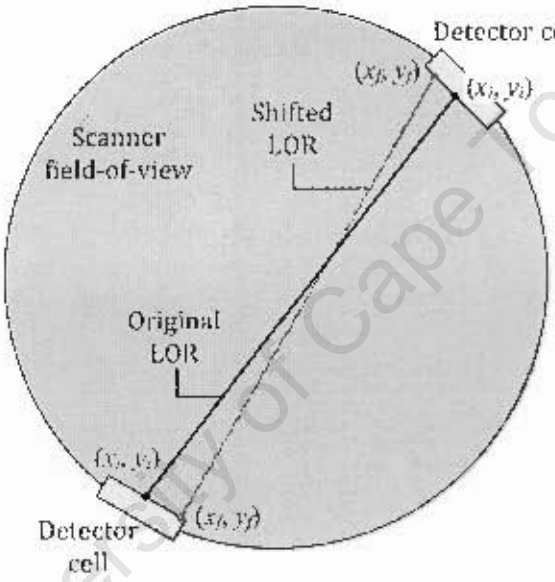


Figure 3.6: The redistribution technique removes the discrete nature of the data by applying a small random shift of the ends of the lines-of-response over their respective detector cell face.

It can be assumed that all lines-of-response pass through the "patient port" of the scanner, which, in the case of the HR++ scanner, limits the field-of-view to a diameter of 570 mm. For this field-of-view, the domain of the angular shift of the lines-of-response during this redistribution process is $\theta \in [0.33°, 0.49°]$. This redistribution of the lines-of-response does not result in a loss of resolution since it only affects details finer than the period of the artefacts, and since the artefacts themselves have already limited the possible resolution to that period. Conversely, the redistribution technique also does not improve the resolution of the final image, but without it the artefacts that appear in the images at fine mesh resolutions degrade the image quality to such an extent that it becomes very difficult to extract information from the image.

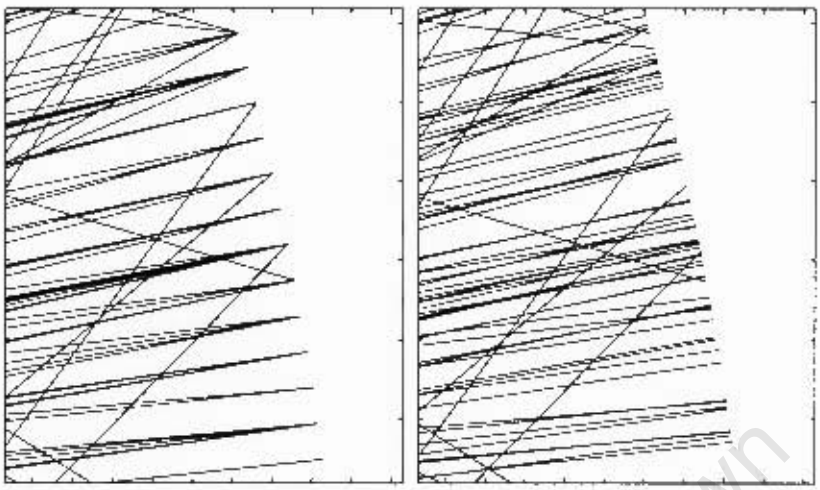


Figure 3.7: The result of the redistribution technique is that the lines-of-response are spread uniformly over the detector faces. The original discrete nature of the data can be observed in the left frame and the same data after applying the redistribution technique is shown in the right frame.

3.3.2 Fourier Filters

The reconstruction algorithm under consideration is susceptible to noise, since any line-of-response contributes to all voxels in the mesh through which it passes. Ideally each line-of-response would contribute only to that voxel in which the annihilation event occurred. This is unfortunately impossible, since no information regarding the origin of the line-of-response is carried by the line-of-response itself. The noise can be partially removed by applying a threshold to the density distribution, but this is not very effective for various reasons - one of them being that it does not remove the noise that happens to be inside the source itself. A better way to remove this noise is by using a Fourier filter.

The Fourier filtering technique is a common technique in image processing to remove noise and thus improve the image quality. This technique involves transforming the processed data into the frequency domain, applying a filter and then transforming it back to the spatial domain; it is described thoroughly in R. W. Hamming's book *Digital Filters* [53].

The Fourier transform

The Fourier transform of a function produces an infinite sum of sines and cosines of varying frequency and amplitude which form an exact (in the infinite sum case) representation of the original function, and in the more realistic truncated and finite sum, an approximation of the function. The transform of a function $f(x)$ is given by

$$F(\omega) = \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx \tag{3.18}$$

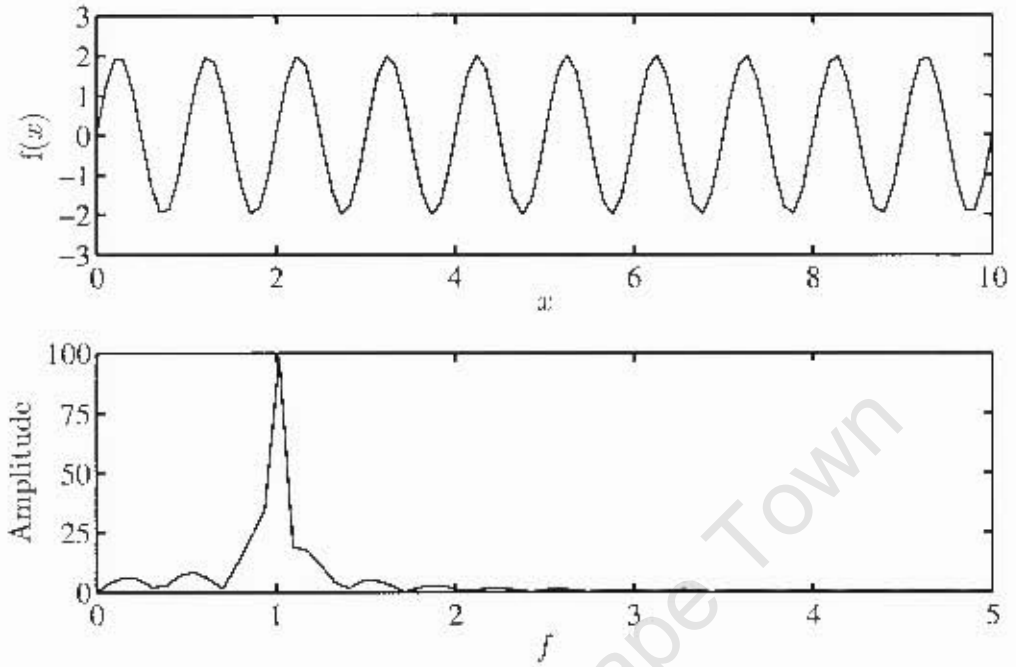


Figure 3.8: A simple example of the Fourier transform (bottom frame) of a sinusoidal function (top frame). The top frame shows the sinusoidal function as it varies with the x-coordinate, and the bottom frame shows the amplitude of the Fourier transform as it varies with frequency, f.

and the inverse transform is given by

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{\infty} F(\omega) e^{i\omega x} d\omega. \tag{3.19}$$

A simple example of a Fourier transform is shown in Figure 3.8. The frequency of the sinusoidal function is 1, and the peak in the frequency domain is at approximately 1. Since the integral in equation 3.18 had to be truncated during the computation the transform is only an approximation, if it were exact the frequency domain would have consisted of only a single spike at $f = 1$.

Figure 3.9 illustrates how the Fourier transform can be used to filter out noise from a signal. A sinusoidal function is generated and random noise is added to it. In the transform of this function the peak around 1 is the only desirable section of the frequency domain: all other frequencies are due to the noise in the function. Thus, if we eliminate high and low frequencies in the frequency domain we can remove that noise. Performing an inverse Fourier transform on the filtered transform gives us a function which has much less noise than the original. This is shown in the last panel of Figure 3.9.

Very noisy and noise-less data were simulated to observe the effect of noise on the frequency domain. The normalised transforms are shown in Figure 3.10. Notice that for the noisy data, the spectrum is nearly flat

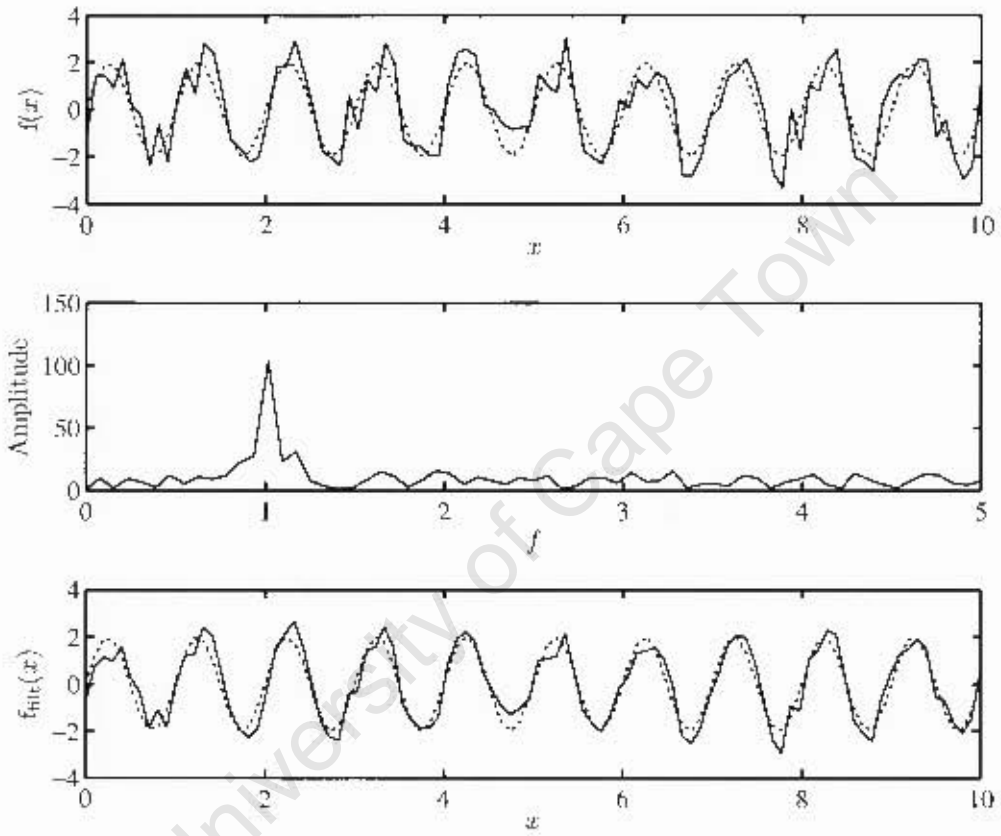


Figure 3.9: An example of filtering using the Fourier transform of a noisy sinusoidal function. The dotted line in the first and last panes is the original sinusoid, and the solid line is the sinusoid with random noise added. The middle pane shows the Fourier transform of the noisy sinusoid.

on average; this demonstrates that the noise occupies all frequencies approximately uniformly (except for a slight rise at the very low frequencies), while the noise-free data clearly has a stronger dependence on the low to mid frequencies than on the higher frequencies. Thus a filter should eliminate the high frequencies, but also reduce the effect of the very low frequencies.

A traditional approach

In the traditional filtered backprojection algorithm, as discussed in section 2.1.4, the Fourier transform of each projection in a sinogram is found. The Central Slice Theorem states that this is equal to the 2 dimensional Fourier transform of the image parallel to the line along which the projection was taken.

Since backprojection places counts along the entire length of each line-of-response, streaks appear to emanate from all line-of-response sources. For a point source this distribution falls off as $1/r$ where r is the distance from the point source, and is known as the Point Spread Function (PSF). To reduce the effect of this blurring a "ramp" filter is employed in the frequency domain. This filter is shown Figure 3.11. The ramp filter is known as a "high-pass" filter as it amplifies the high frequencies relative to the low frequencies.

To preserve finer image details which are governed by higher frequencies, the ramp filter is usually used in combination with a filter which reduces its effect at the high frequencies. Such a filter is referred to as a "low-pass" filter. Various low-pass filters are illustrated in Figure 3.12, along with the combinations of these with the ramp filter. The analytical expressions for these low-pass filters are as follows:

$$\begin{aligned}
\text{Shepp-Logan:} \quad & H(\omega) = \tfrac{2\omega_c}{\pi\omega} \sin\left(\tfrac{\pi\omega}{2\omega_c}\right) \\
\text{Cosine:} \quad & H(\omega) = \cos\left(\tfrac{\pi\omega}{2\omega_c}\right) \\
\text{Hann:} \quad & H(\omega) = \tfrac{\omega_c}{2}\left(1 + \cos\tfrac{\pi\omega}{\omega_c}\right) \\
\text{Hamming:} \quad & H(\omega) = 0.54 + 0.46\cos\tfrac{\pi\omega}{\omega_c}
\end{aligned}$$

where ω_c is the cut-off frequency above which $H(\omega > \omega_c) = 0$. The product of a low- and high-pass filter is applied to the frequency domain of the data before being transformed into the image.

It is not usually possible to know *a priori* which filter to use since the spectrum of the object being imaged could have more dependence on the lower frequencies or on the higher frequencies. Usually various filters are applied and the best quality image is chosen.

A variation on the Fourier filtering technique

In the Fourier filtering technique it is important to note that the filters and transforms are all 1-dimensional, while the data itself is made up of a stack of 2-dimensional sinograms. It is however possible to transform 2-dimensional data into a 2-dimensional frequency domain, and so one might produce better quality images by using a 2-dimensional filtering technique rather than the usual 1-dimensional technique.

In this research, the list-mode data is used rather than the more traditional sinogram data, thus when the Fourier transform is found, it is the transform of the real space domain and not the projection space of the sinograms.

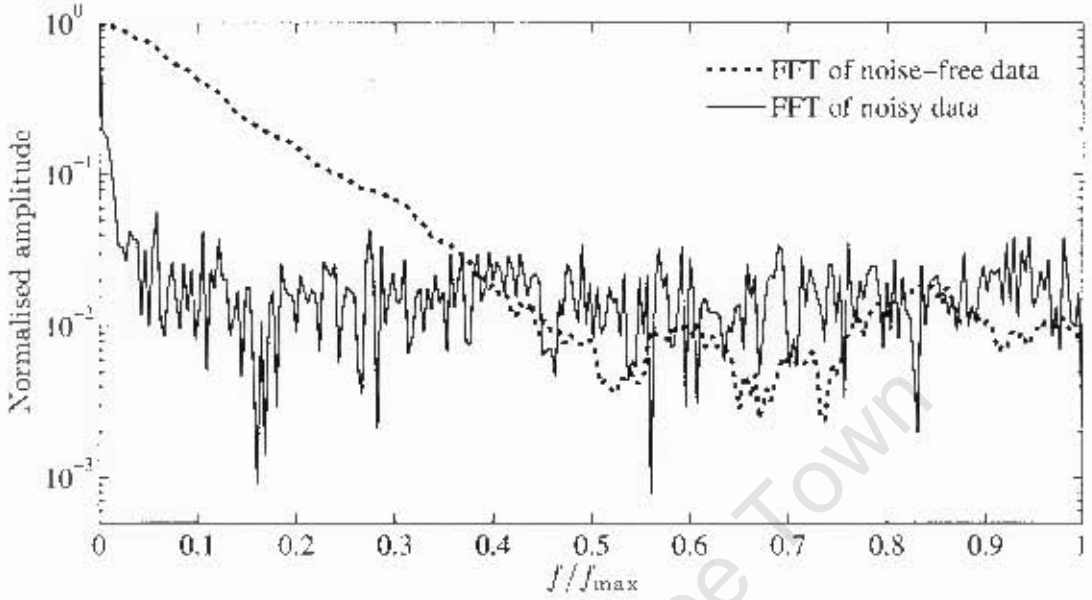


Figure 3.10: A comparison of the Fourier transforms of noisy and noise-free data. For the noisy data the frequency is almost flat on average while the noise-free data has a strong representation of low to mid frequency contributions.

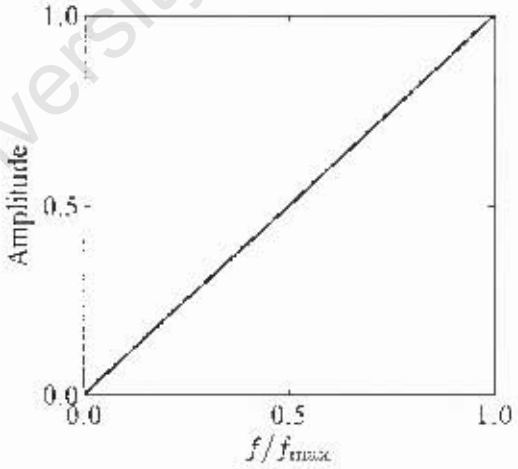
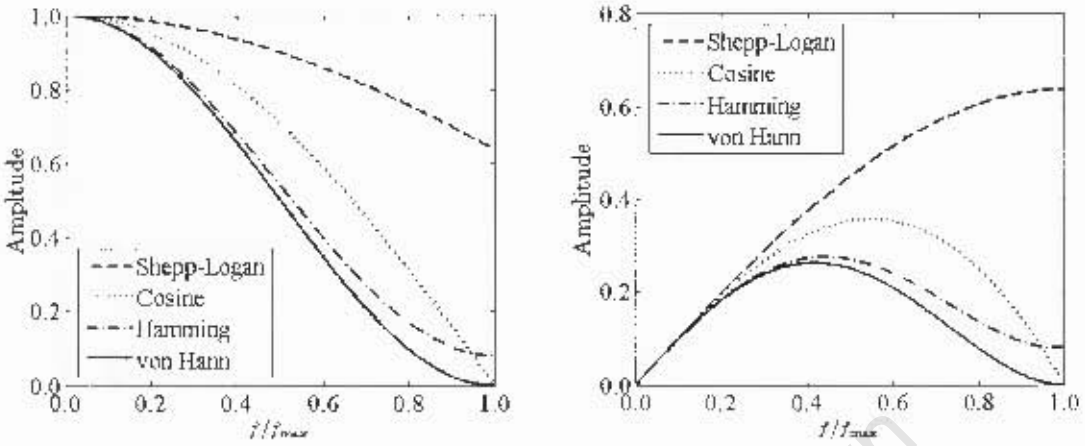


Figure 3.11: The ramp filter, or a high-pass filter.



(a) Low-pass filters used to suppress high frequency noise.

(b) Low-pass filters in combination with the ramp filter.

Figure 3.12: Common filters used to reduce the effect of high and low frequency noise in PET data.

The 2-dimensional Fourier transform of a matrix is given by:

$$F(u,v) = \frac{1}{\sqrt{MN}} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n) e^{-2\pi i \left(\frac{mu}{M} + \frac{nv}{N}\right)} \tag{3.20}$$

and the inverse by,

$$f(m,n) = \frac{1}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F(u,v) e^{2\pi i \left(\frac{mu}{M} + \frac{nv}{N}\right)} \tag{3.21}$$

where $(0 \leq m,u \leq M-1, 0 \leq n,v \leq N-1)$, and M and N are dimensions of the matrix. By utilising the separability of exponentials and the commutability of scalar multiplication, it can be shown that in the computation of the Fourier transform of a matrix, one can calculate the 1-dimensional Fourier transform of the rows and then the 1-dimensional transform of the columns of the resulting matrix, or one can perform the operation on the columns first and then the rows. Either way, the resulting matrix in the frequency domain is the same; the 2-dimensional transform of the original matrix. One can then multiply this frequency domain by an appropriate 2-dimensional filter to eliminate high and low frequencies.

To perform a 1-dimensional Fourier transform, the number of calculations required is $O(N^2)$, where N is the number of terms in the summation. This can be greatly improved by using a technique known as "fast Fourier transform," or FFT, which reduces the number of calculations to $O(N \log N)$. The number of calculations required for a 2-dimensional transform can be reduced to $O(N^2 \log N)$ using FFT. Thus, one must decide whether the increase in image quality is worth the increase in runtime for 2D Fourier filtering. This will be discussed when examining the results of the Fourier filtering. This same issue must be taken into account when one considers extending the filtering to 3-dimensions, which involves yet another increase

in the runtime.

The choice of which filter to use is based on the 1-dimensional case. The 1-dimensional filters can be extended to the 2-dimensional case. The Shepp-Logan and Hamming filters are given as examples below, with $\omega_c = 1$,

$$\text{Shepp-Logan:} \quad H(u,v) = \frac{1}{\pi u}\sin\left(\frac{\pi u}{2}\right) + \frac{1}{\pi v}\sin\left(\frac{\pi v}{2}\right)$$

$$\text{Hamming:} \quad H(u,v) = 0.54 + 0.23(\cos \pi u + \cos \pi v)$$

where u and v are the angular frequencies corresponding to the x and y directions, respectively. The 2-dimensional Hamming filter is shown in Figure 3.13. The filter is applied to the 2-dimensional transform of each plane in the density distribution as determined by the Line Density algorithm, and the result is then transformed back to spatial domain.

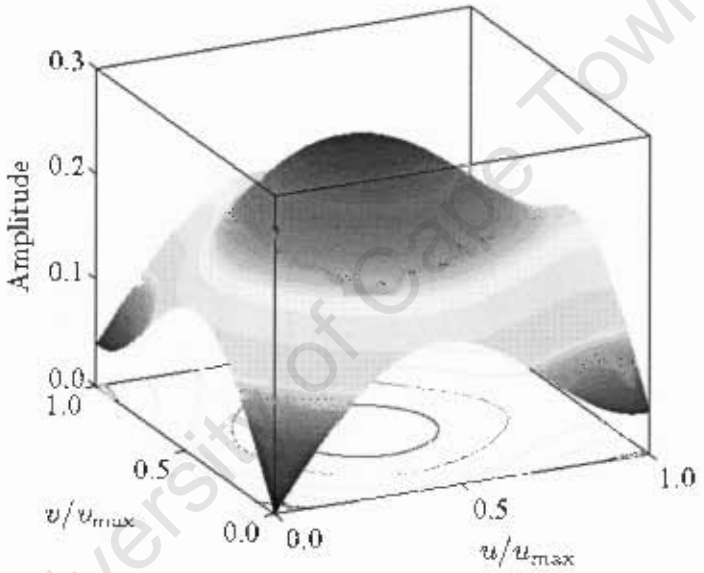


Figure 3.13: The 2-dimensional Hamming filter, in combination with the ramp filter.

3.3.3 Attenuation Correction

As discussed in chapter 2, attenuation of the γ-rays in the surrounding material leads to regions of lower-than-expected intensity in the reconstructed image. Since this effect is known and reproducible, there are techniques that one can employ to correct for this deviation. The most common is known as the "transmission-emission" correction technique, on which the attenuation correction technique used in this research was based.

The "transmission-emission" method

A method that is widely used in medical imaging for attenuation correction makes use of the so-called "transmission data" [3]. The first step in this correction method is to acquire a "blank run". This involves generating a data set using a long, thin rod source that is rotated along the entire inside edge of the camera field-of-view. The resulting image over the entire blank run is a cylinder with a diameter almost that of the field-of-view. The second step is called the "transmission run". The attenuating material (in medical imaging, this is the patient) is placed inside the camera field-of-view as it would be when it is imaged, and a similar procedure to the blank run is followed. The resulting image is again a cylinder just inside the scanner's field-of-view; however the image has now been affected by the attenuating material, which is evident in the image. In PET, along a particular line-of-response through attenuating material, the sum of the paths of both members of the γ-pair will always pass through the same volume of attenuating material, regardless of the origin of that pair. Thus, the effect of attenuation is the same for transmission data (where the source is outside the object) and emission data (where the source is inside the object). This fact allows for the correction for attenuation of each line-of-response by considering the difference between the transmission and emission data. The expected intensity, $E(\ell)$, of a particular line-of-response, ℓ, from an emission scan is given by,

$$E(\ell) = S(\ell)\tau_E \exp\left(-\int \mu(x)dx\right) \int \lambda(x)dx, \tag{3.22}$$

where x parameterises the position along ℓ, $\lambda(x)$ is the activity per unit length along ℓ, $\mu(x)$ is the attenuation coefficient at x, τ_p is the duration of the scan, and $S(\ell)$ is the sensitivity of the detector for ℓ. The expected value for the transmission scan is given by,

$$T(\ell) = A(\ell)\tau_T \exp\left(-\int \mu(x)dx\right), \tag{3.23}$$

where $A(\ell)$ is the combination of the sensitivity of ℓ as well as the activity of the source and τ_T is the duration of the transmission scan. If a blank scan is taken with the same source as the transmission scan, the expected intensity of ℓ should be,

$$B(\ell) = A(\ell)\tau_B, \tag{3.24}$$

where $A(\ell$ is the same as before and τ_B is the duration of the blank scan. Multiplying the recorded emission scan by the ratio of the blank and transmission scan then yields the expected intensity for ℓ without attenuation,

$$E_c(\ell) = E(\ell)\frac{\tau_T}{\tau_B}\frac{B(\ell)}{T(\ell)} = \int \lambda(x)dx. \tag{3.25}$$

For this research a simplification of this method was applied whereby the ratio of the images of the blank and transmission scan was used to correct the image of the emission scan, rather than applying this correction to the line-of-response data. This results in some approximations, but was observed to produce reasonable results and sufficed for the purpose of this research.

The ratio of the blank scan image and the transmission scan image yields a correction matrix that can be applied to any subsequent image using the same attenuating material to correct for the attenuation. In its most basic form the matrix is given by,

$$C(i,j,k) = \frac{B(i,j,k)}{T(i,j,k)}, \tag{3.26}$$

where B is the matrix representing the 3-dimensional blank run image, T is the similar matrix representing the transmission run image, and C is the correction matrix. Any subsequent data can then be multiplied, elementwise, by this correction matrix to correct for the attenuation. It has been observed however that, more often than not, this expression for obtaining the correction matrix does not make a sufficient correction to the attenuated data. Thus, an expression which more strongly emphasises the differences between the blank and transmission run is needed, such as,

$$C(i,j,k) = \left(\frac{B(i,j,k)}{T(i,j,k)}\right)^n, \tag{3.27}$$

where $n > 1$ and serves the purpose of emphasising the differences between B and T while not affecting the similarities between these matrices very much. This factor needs to be set empirically using control tests. After investigation it was found that $n = 5$ produces the best results in most cases.

A second issue with the expression given in equation 3.26 is that, if there are any zeros in either B or T, these will cause either zeros or infinities in C, respectively, and thus an overcorrection of the image. One method to avoid this issue is to perform a significant smoothing of the B and T matrices, which will result in very few, if any, zeros in these matrices. Another method is to use the ratio

$$C(i,j,k) = \frac{B(i,j,k)+1}{T(i,j,k)+1}. \tag{3.28}$$

This form of C will affect the attenuated data less than equation 3.26, since each element in this correction matrix is closer to 1 than the C in equation 3.26. It does however eliminate the problem of having zeros in the numerator or denominator (note that no elements in B and T are negative). The best results have been observed when combining this expression with that in equation 3.27. Hence, in most reconstructions

the following correction matrix was used,

$$C(i,j,k) = \left(\frac{B(i,j,k)+1}{T(i,j,k)+1}\right)^5. \tag{3.29}$$

This correction matrix becomes more effective the better the statistics are that are used for B and T; and as a result of using better statistics, the matrices are smoother with less noise. Thus, after determining the matrices B and T, a 3-dimensional Gaussian smoothing algorithm is applied to these to reduce the effect of the noise. Smoothing generally reduces the resolution of an image, but fine structure is usually not important for the attenuation map. The advantage gained in reducing the noise outweighs the loss of resolution.

Alternative attenuation correction techniques

The attenuation correction technique described in the previous section is one of the more common approaches to attenuation handling. Another useful approach is to make use of Monte Carlo simulations to predict the macroscopic behaviour of the interactions of the γ-rays with the attenuating material. Once this behaviour is quantified, a similar correction matrix can be generated and applied to the data. An advantage of the Monte Carlo technique is that the attenuation correction matrix can be generated without having to take extra readings at the time of the experiment. For the attenuation correction technique described in the previous section, one is required to perform additional scans, namely the blank and transmission scans, which need to be performed at the time of the experiments to ensure accurate correction. The Monte Carlo approach can be applied quite successfully, but to do so one must accurately discern the geometry of the attenuating material and how γ-rays interact with that material. Since millions of microscopic events are simulated, small errors or inconsistencies in the set up can lead to significant macroscopic anomalies in the result. Thus, the more complex the attenuating material, the more difficult and time-consuming these simulations become. The transmission-emission technique is robust, simple and produced good results, and was the preferred technique used for attenuation correction during this research.

3.3.4 Dynamic Imaging

Dynamic imaging, or the imaging of a time dependent region of activity within the scanner field-of-view, can be achieved with the Line Density algorithm, because it makes use of the list mode data for reconstruction. The list mode data consists of a chronological list of the lines-of-response recorded during the acquisition, and by considering only those lines-of-response in a specific time range, one can reconstruct an image for that time slice. This process can then be repeated for subsequent sets of lines-of-response to construct multiple frames of images of the activity as it changes in time. One has control of the time range over which each frame is integrated; however if this range is too long and the activity region changes considerably within that time, then significant blurring will be observed. To avoid this blurring effect, the length of this range must be carefully chosen depending on the rate of change of the active region. Many lines-of-response are needed to reconstruct an image, and the signal-to-noise ratio generally becomes better as more lines-of-response are used. Usually at least several million lines-of-response are necessary; but this number increases as the active region increases in spatial extent, since the activity concentration will decrease proportionately. In

static imaging, the acquisition times can be increased in order to obtain enough statistics as will ensure a reasonable signal-to-noise ratio; however in the case of dynamic imaging, there is a limit to the length of time available for each frame. Therefore in order to ensure that enough lines-of-response are detected, it is necessary to increase the activity concentration inside the region of interest. This must be taken into account during the experimental set up.

In many traditional reconstruction algorithms, sinograms are used to perform the reconstruction. Sinograms hold no time information and so make dynamic imaging impossible; they offer only a single integration of the data over the entire acquisition. Some reconstruction algorithms which use sinogram data start with the list mode data, split it into appropriate subsets, bin these subsets into separate sinograms, and then perform the reconstruction to achieve dynamic imaging. In order to do this, the list mode data must be recorded during acquisition. Reconstruction algorithms which perform the reconstructions directly from the list-mode data, such as the Line Density algorithm, avoid this problem by not binning the data into sinograms. An example of another such algorithm is presented in [54].

3.3.5 Pseudo-Code

The pseudo-code for the Line Density algorithm as applied to dynamic PET imaging is given below. For static imaging the same procedure is followed but no loop is necessary and the block of code inside the loop below is performed once only.

Main

- Initialise parameters: camera geometry, mesh size, data file, length of run, time step, number of time steps
- Apply the **Redistribution technique** to the LORs: shift each end of the LORs randomly over their respective detector cell face
- **loop** over the image counter (for dynamic imaging)
  - Determine the subset of LORs to be used for this image
  - Apply **Meshed Line Density** algorithm (as in the generalised case, section 3.1.5) to the LORs
  - If applicable, multiply density distribution by attenuation correction matrix
  - Apply a Fourier filter to the resulting density distribution
  - Store the filtered distribution
- Display the result

Chapter 4

Experimental Work

Experimental work was carried out with the primary intention of demonstrating in principle that the Line Density algorithm can be used as a static and dynamic PET image reconstruction technique, and to track multiple particles in PEPT studies. Once this had been established, more context-based experiments such as flow through a packed-bed column and biological imaging were conducted to demonstrate the usefulness of the algorithm. The experiments were conducted using the Siemens ECAT HR++ "EXACT3D" PET scanner at the laboratories of PEPT Cape Town at iThemba LABS in Cape Town, South Africa, and the ADAC Forte gamma camera at the Positron Imaging Centre at the University of Birmingham in Birmingham, U.K.

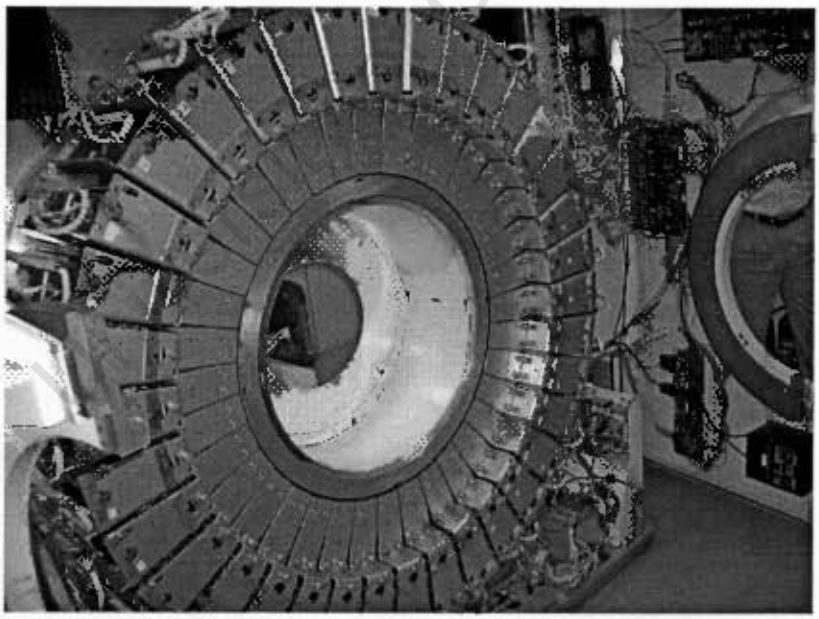


Figure 4.1: The HR++ scanner at iThemba LABS in Cape Town. The scintillator housings are exposed in this picture, and one can see the ring configuration of the camera.

4.1 Scanners

4.1.1 The ECAT HR++ "EXACT3D" scanner

The Siemens ECAT (model 966, CTI, Knoxville, TN) HR++ "EXACT3D" scanner, shown in Figure 4.1, was developed with the intention of achieving unprecedented sensitivity and spatial and temporal resolution [48]. It uses the same Bismuth Germanate (BGO) crystals as the commercial ECAT HR+ "EXACT" system. The HR++ consists of 48 rings each with 72 crystal blocks around its circumference. Each detector block is sectioned into 8×8 detector cells with each cell measuring $4.39 \times 4.05 \times 30$ mm^3 (centre-to-centre spacing of 4.85×4.51 mm^2) yielding 576 detector cells around each ring of the detector, with a total of 27648 cells in the system [48]. The cells are segmented by slices through the crystal of variable depth, allowing for the determination of where on the crystal face the detection event occurred since the different sections of the crystal will yield unique scintillation signals. Each 8×8 detector block is mounted on four photomultiplier tubes (PMT) and the spatial position of the detected event is determined by a weighted average of the detected scintillations.

The HR++ scanner has an axial depth of 234 mm and a ring diameter of 830 mm, although the patient port restricts this to a field-of-view of 600 mm. The mean intrinsic spatial resolution of the scanner is 4.8 ± 0.2 mm FWHM (transaxial, 1 cm off-axis) and 5.6 ± 0.5 mm (axial, on-axis) [48]. The scanner is a dedicated 3D scanner as it was designed without the septa needed for 2D acquisition. This, along with other factors, makes the gantry design very simple and efficient.

The HR++ system was designed to acquire data in both sinogram form and list mode form to allow for greater flexibility and scope in its research capacity. Since the HR++ has 48 axial rings, the total number of sinograms needed to record all events is (48×48) 2304 sinograms, which would require a large storage capacity. Due to this, the data is usually rebinned into 95 sinograms, corresponding to 48 direct sinograms where the ends of each line-of-response lie in the same plane, and 47 cross sinograms where the ends of each line-of-response lie in neighbouring planes. This rebinning is achieved using a maximum ring difference ($rd_{\max}$) of 40 and a "span" of 9, meaning that every alternate four and five sinograms are averaged over except for those near the axial edge of the field-of-view [48]. While still largely maintaining the high signal-to-noise ratio associated with using all lines-of-response, rebinning the data can result in blurring and a loss of spatial resolution away from the centre of the field-of-view of the scanner.

The HR++ can handle a data transfer rate from the detectors to memory of about 17 MB s^{-1}, which equates to approximately 4×10^6 coincidence events per second. These are then written to a 34 GB RAID hard disk (which will soon be upgraded to a 1 TB system).

4.1.2 The ADAC Forte camera

The Forte gamma camera, constructed by ADAC Laboratory, at the Positron Imaging Centre at the University of Birmingham consists of two parallel plates each containing a single NaI(Tl) crystal with dimensions 500 mm $\times$ 400 mm $\times$ 16 mm. Optically coupled to each of these two crystals are 55 photomultiplier tubes. This camera can operate at a maximum of about 2 million single events per second, with a deadtime of less than 170 ns [45]. When a γ-ray is detected on one camera plate a window is opened for about 20 ns, and if

another γ-ray is detected on the other camera plate within this time window these two events are associated with one another and a line-of-response is formed.

The energy resolution of NaI(Tl) crystals is sufficient to detect discrepancies of more than 15% of the 511 keV photopeak's FWHM, which equates to γ-rays that have deviated by more than $\theta = 30°$ from their original flight path. The spatial resolution of the camera near the centre of the field-of-view (the FWHM of a backprojected image of a point source) is approximately 6 mm [45]. To achieve the optimal resolution the detecting plates of the camera should be as close to each other as possible, allowing for a maximal solid angle in which lines-of-response can be detected. For this reason, the heads housing the detector plates can be moved along the transaxial axis of the camera over the range 250 mm - 800 mm.

Although the ADAC Forte camera was manufactured to only produce projection data in the form of sinograms, it has been adapted to also store the list mode data for use in PEPT. List mode form has the advantage of making efficient use of storage space, having high temporal resolution and allowing for post-acquisition data slicing and/or rebinning [18]. For PEPT imaging fully 3D images can be produced by rotating the plates of the camera around the field-of-view; however, this requires long acquisition times (more than an hour) and requires that the subject remain static throughout the run.

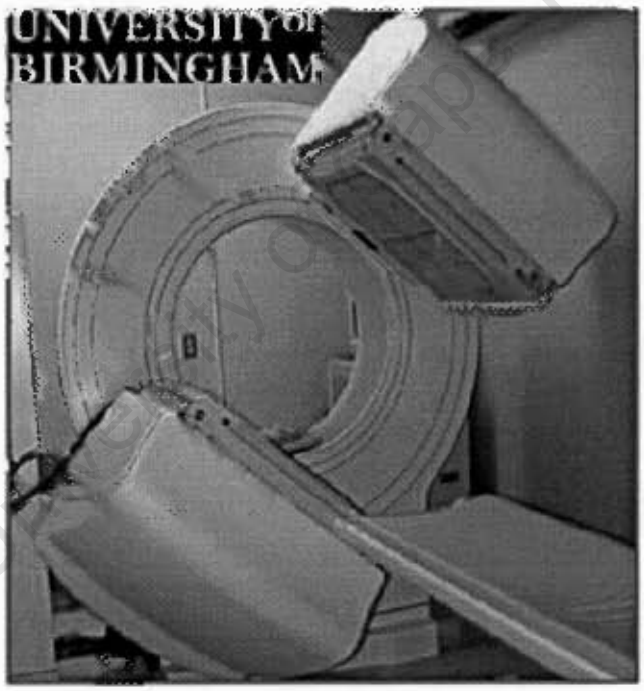


Figure 4.2: The ADAC Forte camera at the University of Birmingham in the UK. The camera has a parallel plate configuration; here the camera has been rotated slightly around its axis.

4.2 Radioisotopes

Three radioisotopes were used in the experiments: ^{68}Ga, ^{18}F, and ^{22}Na. ^{68}Ga and ^{18}F have convenient half-lives of 68 minutes and 109 minutes respectively, and are thus well suited for PET and PEPT studies since their activity will not change significantly during an experiment, but they will decay fast enough to be discarded shortly after use. ^{22}Na on the other hand has a long half-life of 2.6 years and can thus be stored and reused in many experiments.

Experiments performed at the University of Birmingham used ^{18}F tracers. The ^{18}F was produced by direct activation on resin particles using a 35 MeV ^{3}He beam from a MC40 cyclotron. The sources used for the experiments performed at iThemba LABS were produced using various techniques. ^{22}Na is routinely produced at iThemba LABS by direct activation using a proton beam, as is ^{18}F, and ^{68}Ga is produced via a ^{68}Ge/^{68}Ga generator, also at iThemba LABS.

For many PEPT experiments at iThemba LABS "button" sources of either ^{68}Ga or ^{22}Na were used as tracers. These "button" sources were small Perspex cylinders 11.5 mm in length and 6 mm in diameter. A small hole was drilled into the centre of the cylinder and a drop of ^{68}Ga or ^{22}Na was inserted into the hole and the cylinder was then sealed off with a suitable epoxy. The spot size of the active region was less than 1 mm^3 in volume.

Some tracers were labelled using ion exchange. Ion exchange techniques involve immersing the tracer in an active fluid and allowing the active ions to replace the inactive ions on the tracer surface. Both weak-base and strong-base anion exchange techniques were used to label tracers smaller than approximately 1000 μm. This process can be enhanced by introducing metallic ions such as Fe^{3+} onto the tracer surface to improve absorption of the active ions [50].

For a PEPT algorithm to successfully track the particle in each iteration, approximately 100 to 300 lines-of-response are needed. After accounting for the attenuating material in the field-of-view, this places a lower bound on the activity of the particle. Usually each particle needs to be activated to between 100 μCi and 500 μCi when no attenuating material is present. Much lower activities can be used but the results are sometimes not as accurate as experiments with higher activities.

For the PET imaging experiments most of the sources were in liquid form, and ^{68}Ga was the most commonly used isotope since it has a half-life (68 minutes) which is long enough to be used for runs of several minutes in length and short enough so that within a day or two only an insignificant amount of activity would remain. For the vervet monkey brain scans, ^{18}F was used in the form of FDG, or fluorodeoxyglucose which is known to be a good indicator of metabolic processes, especially in the brain [14].

4.3 Experimental apparatus

4.3.1 PEPT

The device shown in Figure 4.3 was designed and manufactured to carry out PEPT experiments with the ADAC Forte camera at the University of Birmingham. This device allowed particles to be moved in a 1- or 2-dimensional oscillatory fashion perpendicular to the faces of the camera, with a maximum of 8 particles

moving independently of one another. Since the particles move in a predictable and simple fashion, the results of the particle tracking could be compared to the known physical motion to check their validity.

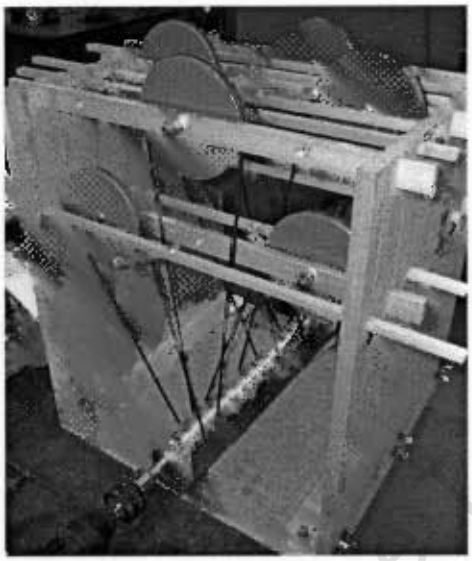


Figure 4.3: Mechanical device used to conduct multiple particle tracking experiments with the ADAC Forte camera.

PEPT experiments were also performed on the HR++ scanner in Cape Town. For many of these, a rotating disk was used with particles attached to the disk at varying radii. The disk was rotated by means of a hand drill positioned within the field-of-view of the scanner. Other experiments were conducted with multiple freely moving particles tumbling inside a container of volume 2 dm^3.

4.3.2 List of experiments performed

Table 4.1 summarises many of the experiments performed to test the application of the **Line Density** algorithm to PEPT and multiple particle tracking.

Scanner	Num. of tracers	Radio-isotope	Approx. activity per tracer (μCi)	Approx. total activity (μCi)	Approx. prompt data rate (ms^{-1})	Length of run (s)	Comments
Forte	1 - 8	^{18}F	350 - 500	350 - 3200	3 - 85	60 - 120	Fig. 4.3
HR++	1 - 16	^{22}Na	50	50 - 800	30 - 1 200	30	On a rotating disk
HR++	4 - 16	^{22}Na	50	50 - 800	190 - 620	20	Shaken randomly in a container

Table 4.1. A summary of the experiments performed for the PEPT studies.

4.3.3 PET

Many experiments were performed with a line source consisting of a tube with inner diameter 1.5 mm or 2.5 mm and filled with a ^{68}Ga in solution. Figure 4.4 shows several experimental setups of line sources, which were used to test various aspects of the reconstruction algorithm. The setup shown in Figure 4.4(a) is of two nearly parallel line sources which was used to test the resolving power of the reconstruction algorithm and the scanner: the point where the two lines can be resolved in the final image is the resolving power of the scanner. Figure 4.4(b) shows a coiled tube filled with an active fluid that was used to test the general performance of the reconstruction technique, and Figure 4.4(d) shows the same coil with a lead plate encompassing it to provide a test for the attenuation correction technique. The rig shown in Figure 4.4(c) was also used to test the attenuation correction technique by surrounding a line source by a column filled with ceramic beads; this rig was also used to test the dynamic imaging capabilities of the algorithm.

A variety of fillable objects were used to create phantoms with which to test the reconstruction algorithm. A small guitar-shaped bottle provided a means to test the imaging of complex shapes with intricate geometry. A hexagonal cylinder was also used in these experiments to test whether the hexagonal shape could be recognised after the reconstruction. The guitar-shaped bottle was also used for dynamic imaging by moving it inside the field-of-view. These objects are shown in Figure 4.5.

PET imaging was conducted on an anaesthetised vervet monkey injected with ^{18}F in compound form to study the uptake of fluorodeoxyglucose (FDG), a glucose, in the brain. Figure 4.6 shows the monkey in the scanner during the experiment. These experiments also allowed for the validation of dynamic imaging in soft tissue by studying the progressive uptake of FDG in the brain over time.

Fluid flow through columns of packed ore or other material is an important area of study in many fields, with applications for the mining industry in particular. This context could be well suited to being studied by PET, since the fluid could be labelled before letting it pass through the column inside the PET scanner. A column to hold ore was manufactured and ^{68}Ga was used to label a fluid to be passed through the column. The column is shown inside the PET scanner in Figure 4.7.

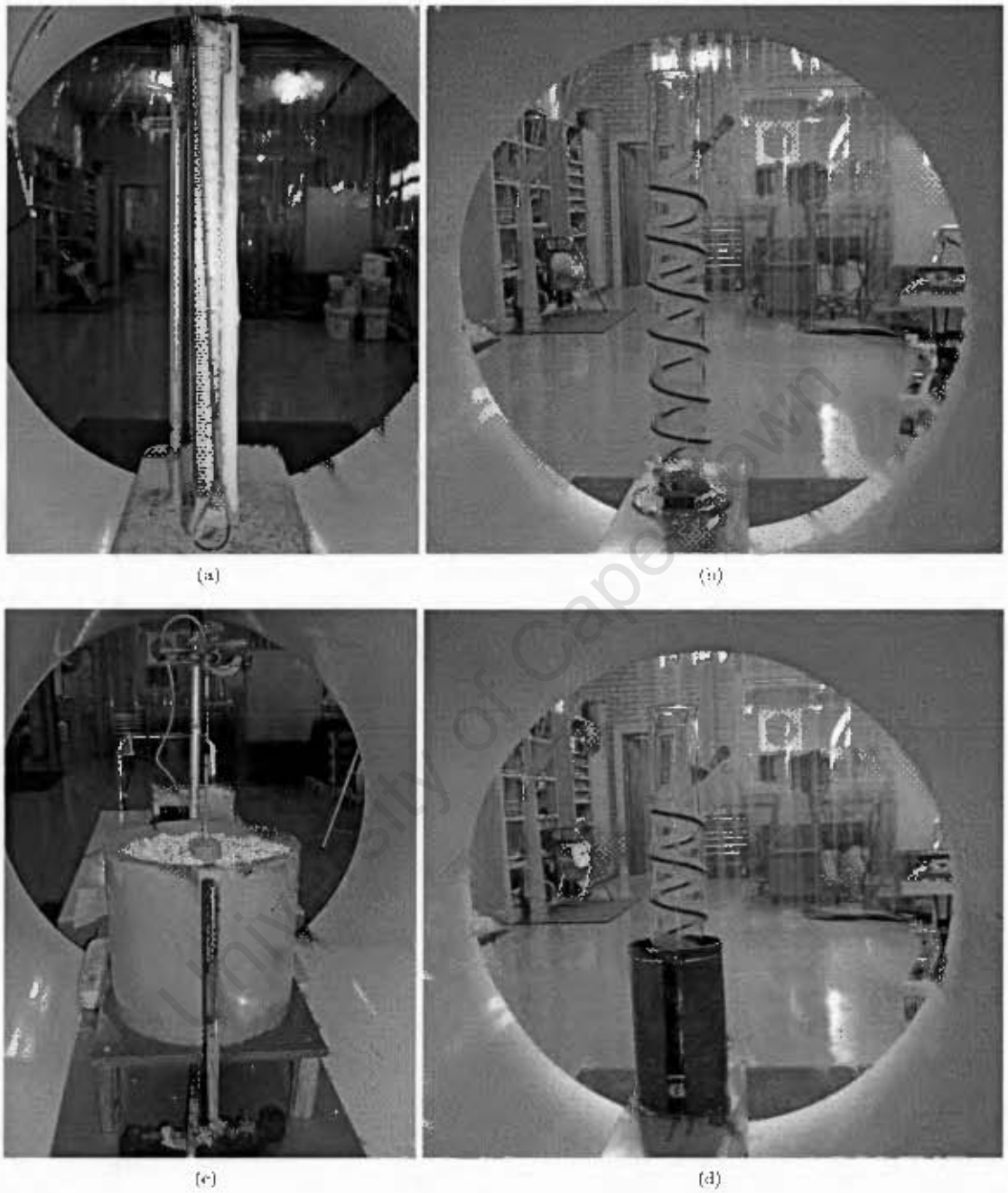


Figure 4.4: Experimental setup of various line sources used to test the reconstruction algorithm; (a) was used to test the resolving power, (b) to test the general performance of the reconstruction technique, and (c) and (d) to test the attenuation correction technique as well as the viability of dynamic imaging.

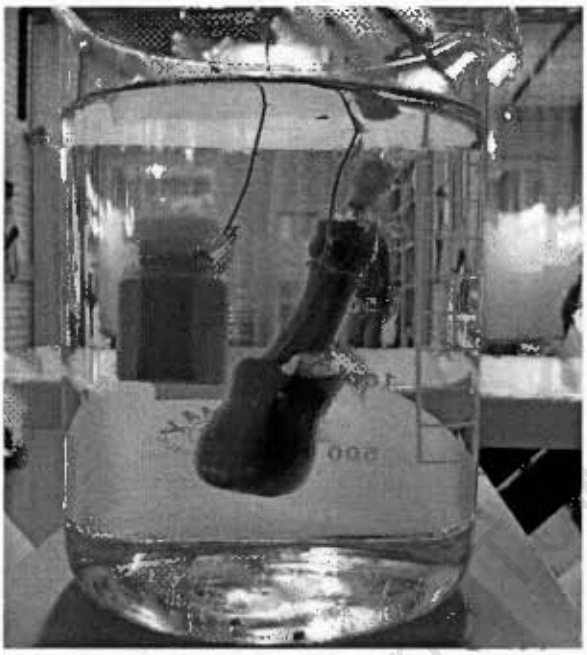


Figure 4.5: A guitar-shaped bottle and a hexagonal cylinder were used to test the reconstruction algorithm on extended objects.

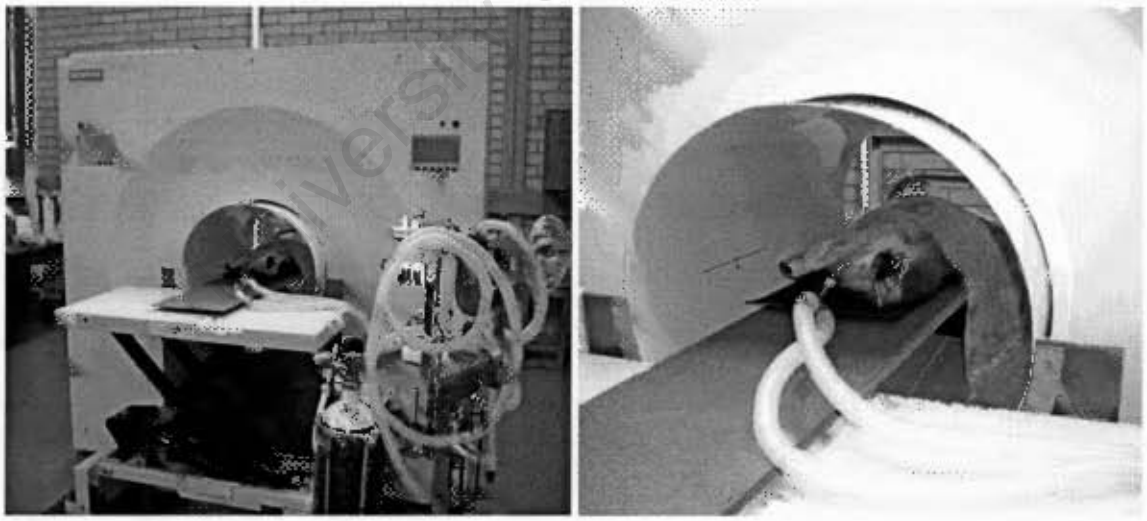


Figure 4.6: Soft tissue scans were performed on an anaesthetised vervet monkey, injected with FDG labelled with ^{18}F, to image the uptake of the FDG in the brain.

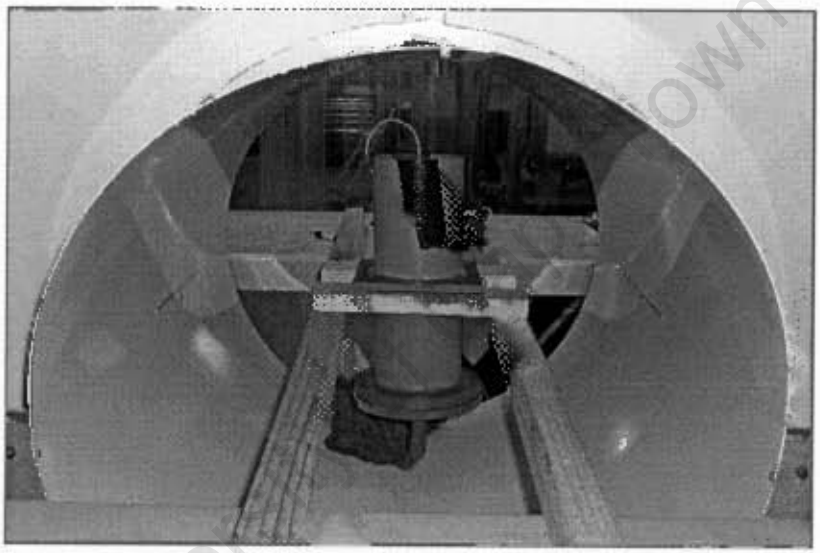


Figure 4.7: A column packed with a sample of mineral ore to study the flow of a fluid through the ore.

4.3.4 List of experiments performed

Table 4.2 summarises many of the experiments performed to test the application of the Line Density algorithm to PET imaging.

Scanner	Radio-isotope	Volume (mℓ)	Approx. total activity (μCi)	Approx. prompt data rate (ms^{-1})	Length of run (s)	Comments
HR++	^{68}Ga	125	300		30	Cubic phantom
HR++	^{68}Ga				1200	Fluid flow through column of ballatini
HR++	^{22}Na			175 - 400	30	Rod source
HR++	^{22}Na			130 - 300	30	Rod source with attenuation
HR++	^{68}Ga	5	1000	80	30	Line source; dynamic and static
HR++	^{68}Ga	5	1000	40	30	Line source through column of ceramic beads; dynamic and static; Fig. 4.4(c)
HR++	^{68}Ga	5	1000	220	30	Tube coiled around cylinder; Fig. 4.4(b)
HR++	^{68}Ga	5	1000	160	30	Tube coiled around cylinder with lead plate; Fig 4.4(d)
HR++	^{68}Ga	2	800	180 - 360	30	Two nearly parallel tubes; Fig. 4.4(a)
HR++	^{68}Ga	200	160, 250	100 - 300	30	2 phantoms: hexagonal cylinder, guitar-shaped bottle
HR++	^{68}Ga	200	160, 250	100	30	2 phantoms in water; Fig. 4.5
HR++	^{68}Ga	100	80	40	30	Dynamic guitar-shaped bottle
HR++	^{18}F	0.15	1350		2700	Anaesthetised vervet monkey injected with FDG; Fig. 4.6
HR++	^{68}Ga	20	1115	60 - 110	300 - 600	Fluid flow through packed ore; Fig. 4.7

Table 4.2: A summary of the experiments performed for the PET imaging studies.

Chapter 5

Results: PEPT

An example of the raw PEPT data is shown in figures 5.1 and 5.2. These subsets of the data each contain several hundred lines-of-response and the approximate location of the tracer(s) can clearly be identified by eye.

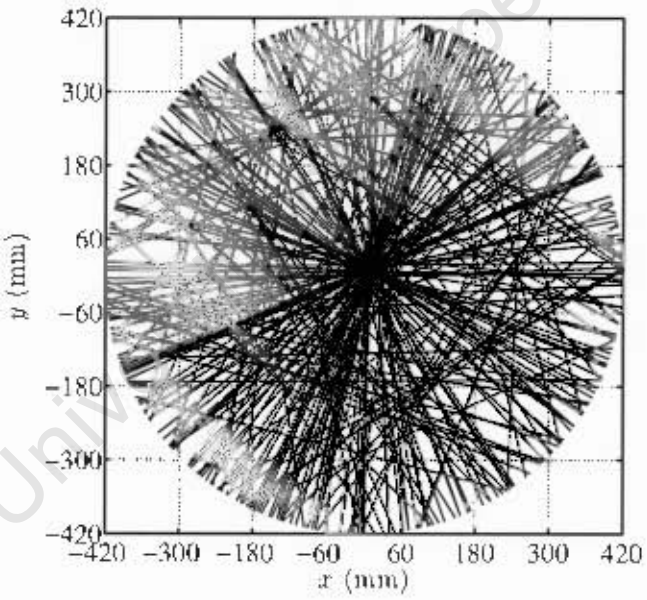


Figure 5.1: Examples of raw data in the form of lines-of-response for a single particle, from the ECAT HR++ "EXACT3D" scanner.

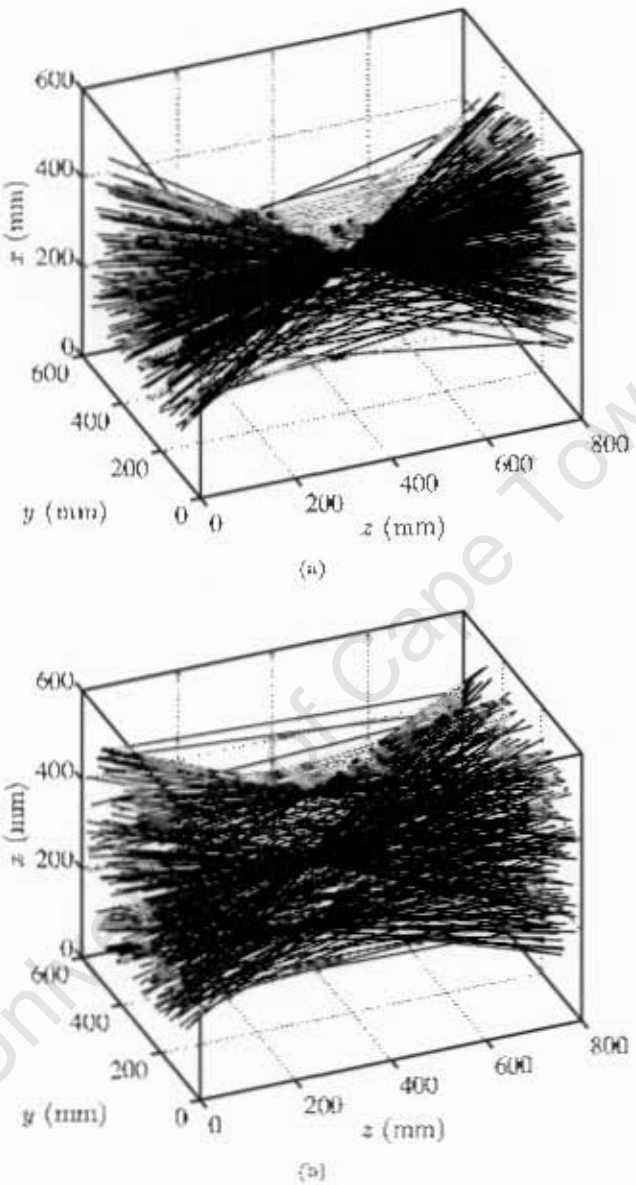


Figure 5.2: Examples of raw data in the form of lines-of-response for a single and multiple particle system, from the ADAC Forte camera. The parallel plates of the camera are situated at $z = 0$ mm and $z = 800$ mm. The approximate locations of the tracers can easily be identified by the human eye in both the (a) single and (b) three particle case.

5.1 Single Particle Tracking

Figure 5.3 shows the result of the analysis using the Line Density algorithm of a single particle moving in a circular motion. The mesh size used for this analysis was $2 \times 2 \times 2$ mm^3 and the time step was 25 ms. The data for this run was acquired using the ADAC Forte scanner. The motion is clearly sinusoidal in nature as expected, and constant in the x-direction. A sinusoidal function, given by $y = a \sin(b(t-c)) + d$, was fit to this result using a χ^2 fit. The constants of the fit were found to be:

	y	z
a (mm)	53.98 ± 0.08	51.9 ± 0.3
b (s^{-1})	2.912 ± 0.001	2.913 ± 0.003
c (ms)	784.0 ± 0.8	1323 ± 2
d (mm)	312.2 ± 0.1	403.6 ± 0.2
R^2	0.9999	0.9990

Table 5.1: The results of fitting a sinusoidal function to the determined locations over time for a single tracer undergoing oscillatory motion in 2 dimensions using the Line Density algorithm.

During the experiment the frequency was set at 0.5±0.1 Hz; the observed frequency was $f = b/2\pi = 0.46 \pm 0.0002$ Hz, which is within the expected value. The particle was positioned as close as possible to the centre of the field-of-view of the camera, and thus its equilibrium position was $(x, y, z) = (300, 300, 400) \pm (20, 20, 20)$ mm. The observed values of $(x, y, z) = (311.3, 312.2, 403.6) \pm (0.7, 0.1, 0.2)$ mm closely match the expected values (since the tracer was stationary in the x-direction, the x-value was calculated as its mean x-position). Finally, the amplitude of the oscillation was expected to be 54 ± 3 mm, which also corresponds to the observed value. The R^2 values for these fits are very close to unity and it can therefore be concluded that the sinusoidal function provides a good fit to the data. A plot of the fits is shown in Figure 5.4.

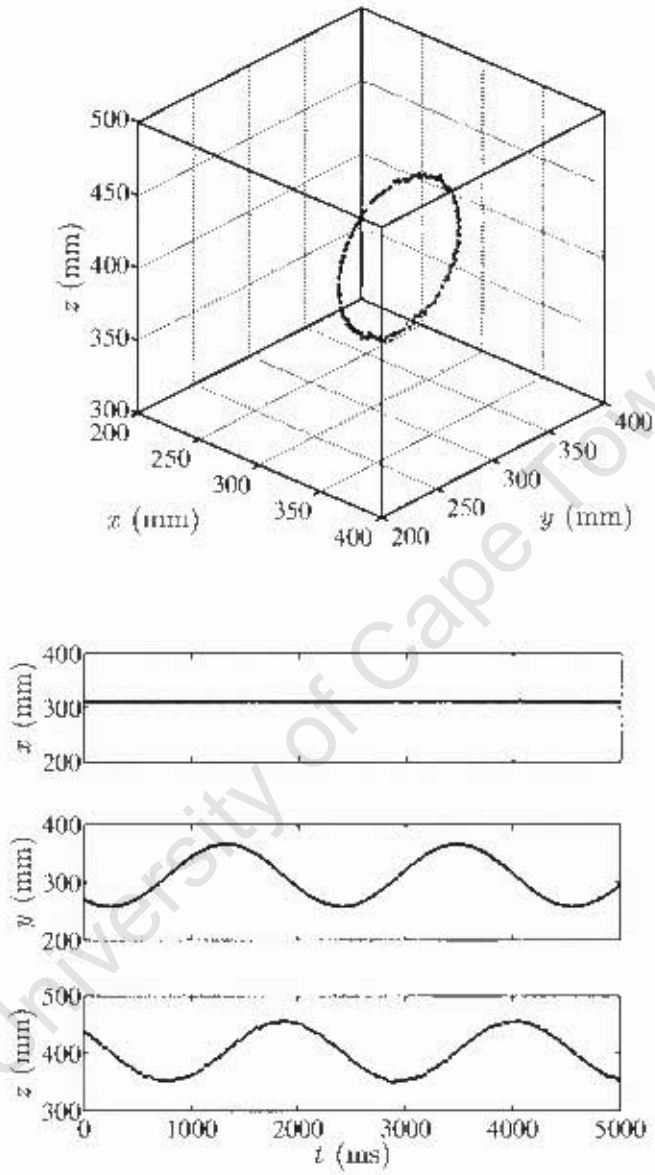


Figure 5.3: Measured (x, y, z) coordinates for a single tracer (initial activity 600 μCi) tracked within the field of view of an ADAC Forte camera. The Line Density algorithm is applied with a time step of 25 ms and mesh dimensions of $2 \times 2 \times 2$ mm^3.

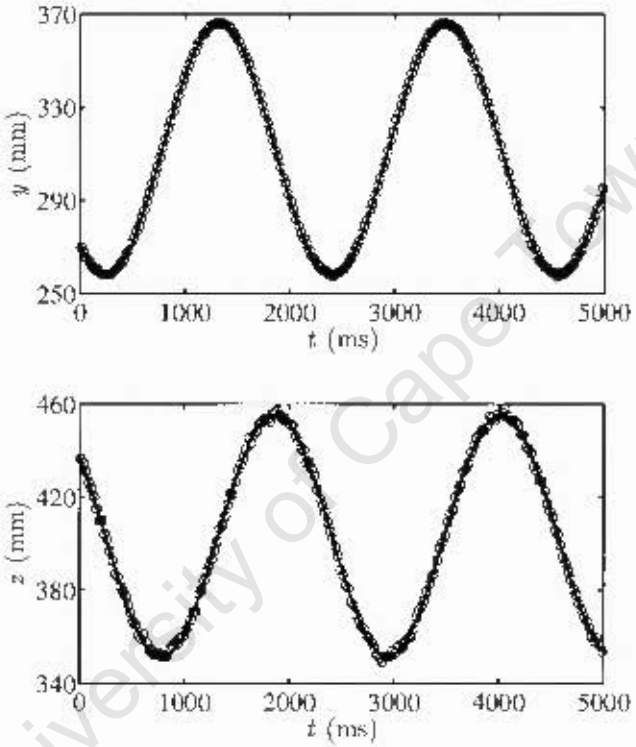


Figure 5.4: A sinusoidal function was fitted to the y and z data shown in Figure 5.3 since the tracers are expected to be moving in a circular manner. The open circles are the data points and the solid line is the fit.

5.1.1 Comparison with the Birmingham algorithm

The Birmingham algorithm is the standard approach to PEPT analysis for single particles, which was presented and discussed in [2, 46, 55] and was introduced here in section 2.2. A comparison of the tracking of a single particle using the Line Density algorithm and the Birmingham algorithm was conducted, using the same data as was used in the previous section. Figure 5.5 shows both results plotted on the same axes. The results appear to be in good agreement.

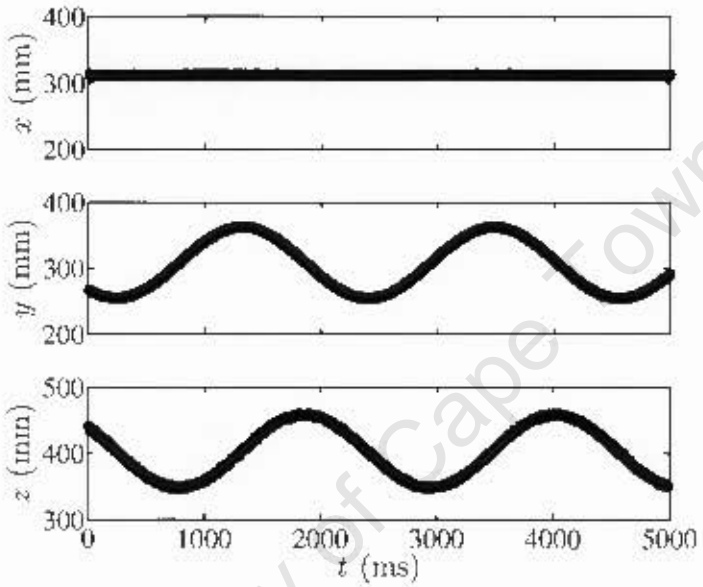


Figure 5.5: Measured (x, y, z) coordinates for a single tracer (initial activity 600 μCi) tracked within the field-of-view of the ADAC Forte camera, using both the Line Density algorithm (applied with a time step of 25 ms and a mesh dimension of $2 \times 2 \times 2$ mm^3) and the Birmingham algorithm (applied with f = 0.40 and N = 320). Both results are plotted and lie on top of each other; the large black dots are the results from the line density algorithm and the smaller grey dots are from the Birmingham algorithm.

As with the results from the Line Density algorithm, a sinusoidal fit was applied to the results from the Birmingham analysis. The determined fit parameters are shown in Table 5.2.

	y	z
a (mm)	54.01 ± 0.04	55.2 ± 0.1
b (s^{-1})	2.912 ± 0.001	2.913 ± 0.001
c (ms)	784.4 ± 0.4	1323 ± 1
d (mm)	308.2 ± 0.1	404.0 ± 0.1
R^2	0.9999	0.9994

Table 5.2: The results of fitting a sinusoidal function to the determined locations over time for a single tracer undergoing oscillatory motion in 2 dimensions using the Birmingham algorithm.

The fits to the results of both algorithms agree with almost all the fit parameters coinciding within their confidence intervals. The small discrepancies in d for the y-coordinate and in a for the z-coordinate may be due to a systematic error in the decoding of the binary list-mode data into ASCII files, since these discrepancies have been observed elsewhere.

The Line Density and the Birmingham algorithms were both used to analyse data from a single stationary particle located at the centre of the field-of-view. The Line Density analysis used a time step of 1.0 ms and a mesh size of $2 \times 2 \times 2$ mm^3, while the Birmingham algorithm used 210 lines-of-response with $f = 0.40$ per triangulation resulting in an average time step of 1.0 ms. The mean and standard deviation of the measured locations over 5 seconds of data were $(8.69, 10.70, -4.42) \pm (1.40, 1.13, 0.72)$ mm using the Birmingham algorithm and $(6.66, 11.29, -4.02) \pm (1.37, 1.40, 0.70)$ mm using the Line Density algorithm. The mean locations fall within one standard deviation of each other except in the x-direction, where the difference between the two algorithms is within two standard deviations.

5.2 Multiple Particle Tracking

Figure 5.6 shows the analysis of a system of eight particles moving in a linearly oscillating manner. This data was acquired using the ADAC Forte camera. These particles were tracked simultaneously using the Line Density algorithm with a mesh dimension of $2 \times 2 \times 2$ mm^3 and a time step of 150 ms. The coordinates with respect to time for three of the particles are also shown.

Sixteen particles were attached at fixed radii to a disk rotating at a frequency of 0.95 Hz, and tracked using the HR++ scanner. Figure 5.7 shows the result of the analysis of this data, where the particles were tracked simultaneously using the Line Density algorithm. The mesh dimensions were $2 \times 2 \times 2$ mm^3 and 2 ms of data was used for each iteration. This is the first time that such a large number of particles have been tracked simultaneously using PEPT.

Figure 5.8 shows the results of tracking 4 freely moving particles, tumbling randomly within a closed container of volume 2 dm^3. The data was acquired using the HR++ scanner. The particles were moving independently and could thus collide with or pass very near to each other, after which point the algorithm failed to separate and identify each particle involved in the collision. A future development of the algorithm will be to effectively manage collisions between particles.