7-24-2003

Accuracy of activity quantitation of F-18 fluorodeoxyglucose (FDG) Positron Emission Tomography (PET) imaging using simulated malignant tumors

Madhu Durai
Florida International University

FLORIDA INTERNATIONAL UNIVERSITY

Miami, Florida

ACCURACY OF ACTIVITY QUANTITATION OF F-18 FLUORODEOXYGLUCOSE

(FDG) POSITRON EMISSION TOMOGRAPHY (PET) IMAGING USING

SIMULATED MALIGNANT TUMORS

A thesis submitted in partial fulfillment of the

requirements for the degree of

MASTER OF SCIENCE

in

BIOMEDICAL ENGINEERING

by

Madhu Durai

2003

To: Dean Vish Prasad
College of Engineering

This thesis, written by Madhu Durai, and entitled Accuracy of Activity Quantitation OF F-18 Fluorodeoxyglucose (FDG) Positron Emission Tomography (PET) Imaging Using Simulated Malignant Tumors, having been approved in respect to style and intellectual content, is referred to you for judgment.

We have read this thesis and recommend that it be approved.

Eric T. Crumpler

Anthony J. McGoron

Juan Franquiz, Major Professor

Date of Defense: July 24, 2003

The thesis of Madhu Durai is approved.

Dean Vish Prasad
College of Engineering

Dean Douglas Wartzok
University Graduate School

Florida International University, 2003

ACKNOWLEDGMENTS

ABSTRACT OF THE THESIS

ACCURACY OF ACTIVITY QUANTITATION OF F-18 FLUORO-

DEOXYGLUCOSE (FDG) POSITRON EMISSION TOMOGRAPHY (PET) IMAGING

USING SIMULATED TUMORS

by

Madhu Durai

Florida International University, 2003

Miami, Florida

Professor Juan Franquiz, Major Professor

This thesis involves a procedure, which calculated and compared the sum of all the pixel counts, threshold pixel counts sum of a 3D PET image and mean and maximum pixel count of one single transaxial slice (2D) of simulated tumors for a chosen region of interest (ROI). A calibration factor was multiplied by the sum of the pixel counts, threshold pixel counts sum of all the transaxial slices, and the mean, and maximum pixel counts of one single transaxial slice in an ROI to calculate for the activity of the tumor. This activity calculated was compared with the real activity values. The results showed that the sum of all the pixel counts with applied threshold is better to calculate the activity of tumor with greater accuracy.

These findings suggest that a 3D distribution of sum of all the pixel counts was able to calculate the activity of malignant tumors and lung lesions with better accuracy.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. INTRODUCTION

Positron Emission Tomography (PET) is a three-dimensional (3D) imaging technique, which can be used to measure the level of metabolic activity within the cell. The process of measuring begins when a radiopharmaceutical is injected into a vein of the patient, carried to the site of interest, and undergoes radioactive decay. PETs' measurement of this decay is based upon the annihilation reaction between a positron and a tissue electron. Two photons created in the annihilation reaction travel away from each other at a 180-degree angle, and are simultaneously sensed by opposing detectors. This detection reveals their line of origin (Turkington et al., 2001). A computer then constructs a transaxial image according to standard back projection or iterative reconstruction methods. The most commonly used radiotracer for PET oncologic imaging is fluorine-18-labeled fluorodeoxyglucose (Hani et al. 2002).

PET provides the means for imaging the rates of biologic processes in vivo. Imaging is accomplished through the tracer kinetic assay method. The tracer kinetic assay method employs a radiolabeled biologically active compound (tracer) and a mathematical model that describes the kinetics of the tracer as it participates in a biological process. The model permits the calculation of the rate of the biological process. The PET scanner provides the tissue tracer concentration measurement required by the tracer kinetic model, with the final result being a 3D image of the anatomic distribution of the biological process under study (Graham et al., 2000). The tracer technique continues to be one of the most sensitive and widely used methodologies for performing assays of

biological systems. PET allows the transfer of the tracer assay methodology to the living subject, particularly humans. PET builds a bridge of communication and investigation between the basic and clinical sciences, based upon a commonality of methods used and problems studied.

The transfer of tracer methods from the basic biological sciences to humans using PET is made possible by the unique nature of the radioisotopes used in PET to label compounds: 11-C, 13-N, 15-0, and 18-F. These are the only radioactive forms of the natural elements (18-F is used as a substitute for hydrogen) that emit radiation that will pass through the body for external detection. Natural substrates, substrate analogs, and drugs can be labeled with these radioisotopes without altering their chemical or biological properties. This allows the methods, knowledge, and interpretation of results from tracer kinetic assays used in the basic biological sciences to be applied to humans by the quantitative measurement abilities of the PET scanner (Phelps et al., 1992).

1.1 18- FDG

18F-labeled 2-deoxyglucose (FDG) is used in neurology, cardiology and oncology to study glucose metabolism. In cardiology, [18F]-labeled FDG can be used to measure regional myocardial glucose metabolism. Although glucose is not the primary metabolic fuel of the myocardium, glucose utilization has been extensively studied as a metabolic marker in both diseased and normal myocardium. Since [18F]-labeled FDG measures glucose metabolism it is also useful for tumor localization and quantitation. 18-FDG is potentially useful in differentiating benign from malignant lesions because of the high metabolic activity of many types of aggressive tumors (Schulte et al., 1999). The

development of 18-FDG has been the major factor in expanding the clinical role of PET imaging. The development of PET instrumentation, FDA approval of 18-FDG, and its advantages compared to other radiopharmaceutical favors the use of 18-FDG in PET imaging.
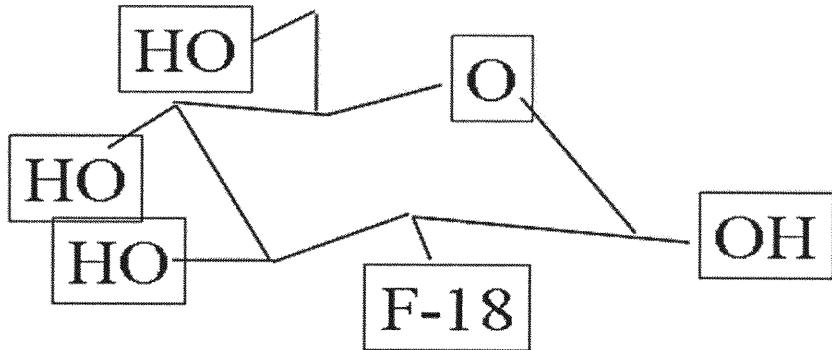
Figure1. Structure of 18-FDG

1.2. PET/CT Hybrid Scanner

A number of computer algorithms are available to register image sets from different modalities, such as CT and PET retrospectively (Marguire et al., 1991). Retrospective image alignment works best for a rigid organ such as the brain. Success in other regions of the body is less certain since, in most cases, the patient must be moved between the two machines and repositioned on different beds, with the two scans perhaps even being performed on different days. This inevitably leads to mis-registration due to differences in patient position, physiological state, scanner bed profile, and the uncontrollable movement of internal organs. Even with the use of reference markers

retrospective alignment procedures can be labor-intensive (Tai et al., 1997) making them less attractive for routine clinical use in high-throughput scenarios.

Figure 2. Combination of GE Discovery LS (GEMS) Light speed plus CT scanner and GE Advance Nxi PET scanner. (Courtesy of GE medical systems).

The advantages with the PET/CT Hybrid Scanner are:

1. Combining a PET and a CT, transmission images are used to construct an attenuation correction map scaled to 511keV.

2. The attenuation map is noise-free, thus a practical solution is obtained for the need of a very rapid, low-noise and quantitatively correct method of PET attenuation correction.

3. The automatic registration of PET and CT images with sub-millimeter accuracy.

4. CT provides the anatomic framework needed for PET images.

5. Clinical diagnostic quality CT images are obtained.

The disadvantages of the PET/CT Hybrid scanner are:

1. The array of detectors surrounding the patient detect two gamma rays having an energy of 511 keV for PET imaging, whereas for CT imaging, the transmission energy is 140 keV (Efficient energy is around 80 keV). Since the attenuation coefficients are energy dependent, coefficients measured at CT energies must be converted to the appropriate values at 511 keV if they are to be used to correct PET emission data.

2. Though the same gantry is used for both CT and PET imaging, involuntary patient motion in the form of respiratory motion or cardiac motion might not produce the desired result after image registration. Other motions during the study can also affect the automatic registration of PET and CT images.

1.3. OBJECTIVE OF THIS RESEARCH

The main objective of this research is to determine the most accurate method for quantitation of activity of 18-FDG in simulated lesion.

The specific aims are:

1. To calculate a voxel calibration factor for activity quantitation of PET images.

2. To compare the accuracy of activity quantitation of PET images by using different methods: mean and maximum voxel activity in a single transaxial slice and sum of voxel activity in a stack of transaxial slices. Measurements in a single transaxial slice were considered as 2D methods and measurements including complete stack of transaxial slices were considered as 3D methods.

3. Since the hybrid PET/CT scanner used in this study is a new instrument (the 3'rd installed in the US and the fifth installed in the world), the results of this research were used to partially validate the performance of the PET-CT scanner.

1.4. HYPOTHESIS

The basic hypothesis of this research is that the 3D quantitation of activity of 18-FDG uptake in simulated tumors provides a more accurate description of the uptake than simple measurements in one plane.

1.5 Significance of the Research

The primary goal of 18-FDG PET imaging is to determine if a tumor is present and to determine if a known tumor is responding to treatment. A quantitative study of 18-FDG PET imaging will determine the uptake of the tumor. One of the most popular and commonly used method for quantitation in clinical 18-FDG PET imaging is Standardized uptake value (SUV) method. SUV, a practical way to quantify glucose metabolism in tumors, is basically the ratio of two specific activities: that of a tumor at the study's end and a temporally constant entire body average. SUV is mathematically shown as:

$$SUV = \frac{\text{voxel concentration } (\mu Ci/ml)}{\text{injected dose } (\mu Ci)/ \text{ body weight } (g)}$$

Its clinical appeal, compared with various other quantitative approaches, lies in its simplicity. SUV is very commonly used as an adjunct to visual interpretation (Joseph et

al., 2000) (SUV = 1 would be obtained if the entire dose distributes uniformly throughout the body).

The SUV explicitly corrects for the variable distribution volume of the tracer, via measurement of the patient's weight, to provide an index that is much more uniform among patients than the earlier measure, with higher SUV's indicating increased 18-FDG uptakes and presumably, increased glucose metabolic rates. Hence there is a direct relationship between tumor growth rate and SUV (Miller et al., 1998). SUV has been calculated by commercial software, by choosing a region of interest (ROI) and calculating the mean or maximum pixel value in that ROI. However only one SUV value, either the average or the maximum value, is not enough to characterize the tumor. This is the main disadvantage of SUV.

The various advantages of SUV- PET imaging are (Schulte et al., 2000):

1. Can monitor cancer therapeutic efficiency.
2. Can differentiate scar lesions from recurrent or new malignant lesions.
3. Can reduce the number of invasive procedures.
4. Improves cancer staging and consequently the therapeutic choices.
5. Increases accuracy of radiation treatment planning (IMRT) treating the most active tumoral cells with the highest radiation dose.

The disadvantages of SUV- PET imaging are:

1. The need to standardize the time between the tracer administration and data acquisition (Kole et al., 1997)
2. SUV is reduced to only one value in one pixel, in one transaxial slice, while the malignant lesion is 3-D and occupies several slices.

3. SUV is affected by various factors like statistical noise, partial volume effect and recovery coefficients.

The method we are proposing is an extension of classical SUV values by including all the pixels, and our study is 3D as we are including all the tumor pixels in all the transaxial slices. SUV is a method which is used to determine the uptake of radiopharmaceutical by the human body and our method is based upon the phantom studies. Here we know the activity of the simulated tumor and we verify the accuracy of this activity in 18-FDG PET imaging by calculating all the pixel count values for a region of interest(ROI) of the simulated tumor and finding the activity of the simulated tumors by using either the sum, mean or maximum pixel counts per minute and multiplying the pixel counts per minute with a calibration factor and determining which method gives better accuracy by comparing the calculated activity with real activity. The calibration factor is found to determine how many pixel counts per minute is equivalent to one micro curie, which is the activity units of 18-FDG. The linear regression analysis is also applied between sum, mean and maximum pixel counts and activities in simulated tumors to determine which among these three pixel counts have better relation with the activity. PET quantitation with 18-FDG is usually done by 2D methods in which we have only one transaxial slice and either the mean or maximum pixel count is used to determine the activity of the tumor. Our method is 3D in which we use all the transaxial slices and draw the same ROI in all the transaxial slices and include all the pixels that corresponds to tumor activity. We are trying to determine the accuracy of the uptake by applying threshold values and eliminating the pixel counts that do not correspond to the activity of the tumor in the chosen ROI.

2. BACKGROUND

Positron Emission tomography (PET) is a diagnostic method that creates high resolution, 3D tomographic images of the distribution of positron emitting radionuclides in the human body. The radiolabeled compounds used include substrates, ligands, drugs, antibodies, neurotransmitters and other biomolecules that are tracers for specific biological processes. Thus the resulting PET images can be considered images of these biochemical or physiological processes (Hoffman et al., 1992). The images produced are functional indexes of blood flow, glucose metabolism, amino acids transport, protein metabolism, neuroreceptor status, oxygen consumption and even cell division (Degrado et al., 1994).

The ability to study biochemical processes in vivo and to quantify and characterize these processes by a functional parameter has become possible in the second half of the 20th century. The conceptual idea of measuring in vivo biochemistry originates from around 1930 when many new discoveries in nuclear physics were made. The first artificial radioisotopes were produced and almost immediately it was realized that the possibility of in vivo biochemistry was within reach. Reality, however, was long in coming because scintillation detectors and their appropriate electronics, the required computers for data acquisition and image reconstruction, were nonexistent at that time. The first scintillation crystals were discovered around 1950, computers came into use somewhat later, so the first positron camera was not constructed until the 1960s (Paans et al., 2002).

In 1978 the first commercial PET scanner based on NaI (TI) crystals became available. A breakthrough in detector technology was accomplished with implementation of the BGO (bismuth germanate) block-structure detector. Compared with NaI (TI), a higher spatial resolution and larger axial length were obtained, as was a much higher sensitivity due to the density and larger atomic number of BGO (Paans et al., 2002). With the development of scatter correction techniques, three-dimensional (3D) data acquisition with its high sensitivity became possible. At present, the whole-body mode is the most frequently used acquisition mode in clinical oncological studies. In the whole-body mode the total body is scanned in steps and afterward these consecutive studies are put together as a total body overview. Especially in the 3D mode, the total body can be scanned and visualized in a reasonable amount of time (Paans et al., 2002).

2.1. Positrons

Positrons are the anti-particles of electrons. The major difference from electrons is their positive charge. Positrons are formed during decay of nuclides that have an excess of protons in their nucleus compared to the number of neutrons. When decaying, these radionuclides emit a positron and a neutrino. While the neutrino escapes without interacting with the surrounding material, the positron interacts with an electron. During this annihilation process, the masses of the positron and the electron are converted into two photons that travel apart in almost opposite directions. Since the entire (equal) masses are being transformed, each photon obtains 511 keV of energy.
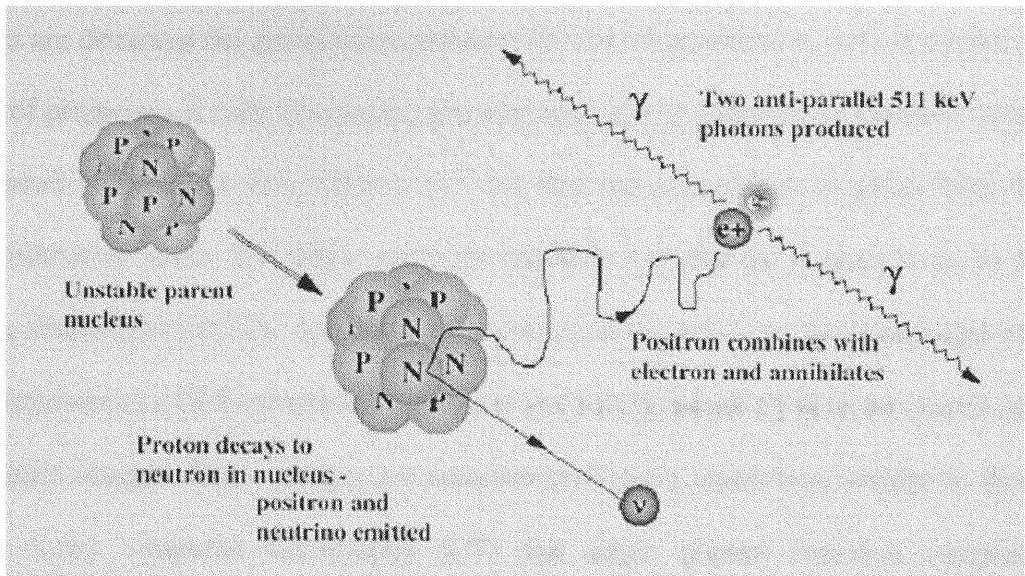
Figure 3. The positron electron annihilation process. Two photons are created, each with energy of 511 keV, traveling in almost exactly opposite directions. (LOP, line of projection). (Courtesy: http://www.jpnm.org).

In the early days of PET, annihilation radiation was considered to be less optimal for imaging because of its relatively high energy. However, later it was demonstrated that the two annihilation photons traveling nearly co-linearly offer substantial advantages in the collimation of this radiation. This property of the annihilation radiation was found particularly desirable in the 3D tomographic imaging of the distribution of positron emitters (Aronow et al., 1967).

2.2. Detection of positrons

Positrons cannot be detected directly. The maximum linear range of a positron is in the order of only a few millimeters. So, in general the positron cannot escape from the human body for external detection. The two annihilation photons of energy 511-KeV having a relative angle of 180°, however, can be detected simultaneously using two or more scintillation detectors in coincidence mode (Paans et al., 2002). When two opposing

detectors are detecting the annihilation photons, the site of annihilation will be a point on the line of projection (LOP), connecting the detectors. If two photons are detected within a very small coincidence time window (15 ns), they are assumed to originate from the same annihilation event. The place where the positron was emitted is close to or on the LOP, the distance to the LOP depends on the energy of the positron. Integrated data of a pair of detectors in a PET camera having two to several thousands of detectors result into tomographic images. These images are reconstructed using algorithms similar to those used in X-ray computed tomography (CT) and single photon emission computed tomography (SPECT) (Blokland et al., 2001).

In contrast with conventional gamma camera imaging, coincidence detection excludes the necessity of collimation with a lead collimator, which significantly increases the sensitivity of PET compared to SPECT.
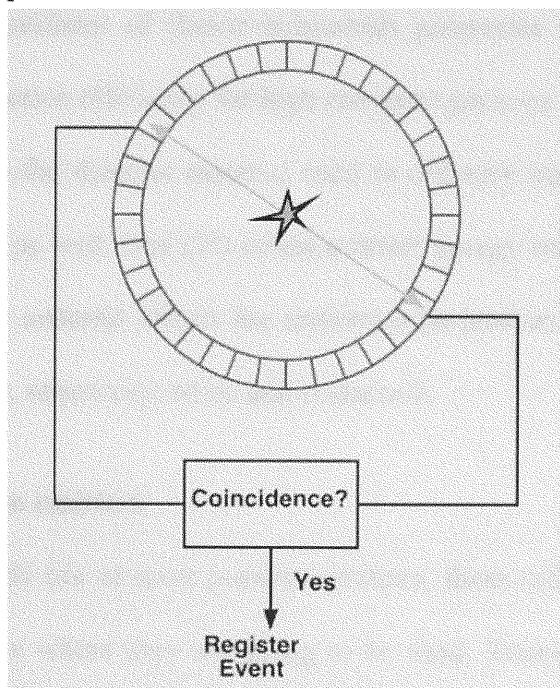
Figure 4. Coincidence Detection (Source: Turkington et al., 2000).

In Figure 4, the location of the annihilation is determined by detection of the two 511 keV photons by a pair of photon detectors using coincidence detection. If the two detectors records two 511 keV photons simultaneously, the position of the annihilation must have occurred somewhere along the line connecting the two detectors, since the annihilation photons are emitted 180° apart. A modern PET system typically has more than 10,000 detector elements arranged in rings surrounding the patient. These detector elements form over 20,000,000 possible coincidence combinations.

2.3. Positron detectors

Radiation detectors for use in PET must be optimized over several physical characteristics of the detectors, including detection efficiency, output signal strength, signal decay time, coincidence timing characteristics, and certain physical properties. In conventional PET, the scintillator of choice is bismuth germanate ($Bi_4Ge_3O_{12}$; BGO), which has a very high detection efficiency for high energetic photons. Thallium-activated sodium iodine (NaI (Tl))—the detector material used in ordinary single photon gamma cameras, has been applied as well. NaI (Tl) offers a better energy resolution than BGO. The choice of the detector material affects the important parameters of a PET scanner, including spatial resolution, sensitivity, noise and count rate.

2.4. Production of positron emitters

Due to the short half-life of most positron emitters, these radionuclides must be produced close to the place where they are going to be used. Transport times must be reduced to a minimum. Two basic approaches for preparing these radionuclides can be used. First, the isotopes can be produced in a nuclear generator—using the

parent/daughter radionuclide approach. Unfortunately, only a few useful short-living radionuclides can be prepared by this method (e.g. ^{82}Rb for cardiac PET imaging) (Blokland et al., 2001).

The second approach is to use cyclotrons to produce artificial radioisotopes, including carbon-11 (11-C), nitrogen-13 (13-N), oxygen-15 (15-O) and fluorine-18 (18-F). Several commercial companies offer small reliable cyclotrons specifically designed to produce radionuclides for PET imaging that are relatively easy to operate (Blokland et al., 2001).

2.5. Positron emitting radio pharmaceuticals

Shortly after the development of the cyclotron, the usefulness of 11-C, 13-N, 15-O, and 18-F as radioactive tracers for biological studies was recognized. Carbon, nitrogen and oxygen are atoms essential to most physiological processes. Fluorine has been found a useful radionuclide to label biologically important molecules. These nuclides most generally exhibit properties that render them particularly desirable in physiological studies. They provide an ideal starting point for the production of a variety of tracers that are either natural substances, such as 11-C-labelled methionine, or analogues, such as 18-FDG. Of course it is necessary to match the half-life of the nuclide to the phenomenon being studied.

One of the challenging problems was the labeling of pharmaceuticals for medical use with positron emitting nuclides. Chemists became interested, which led to the development of fast and automated labeling procedures. In 1980, a list of approximately 30 compounds labeled with positron emitting radionuclides was published (Pogossian et

al., 1980). Since then, the list has increased to several hundreds of compounds (Blokland et al., 2001).

Carbon-11

Carbon-labeled carbon monoxide (11-CO) provides an excellent and simple way to label red blood cells for localizing blood pools by external scanning. 11-CO and 11-CO_2 have been used to study pulmonary function (Blokland et al., 2001).

Nitrogen-13

Nitrogen-13 (13-N) is mainly applied as 13-N-ammonia to study perfusion. It has been used extensively to assess myocardial perfusion. In oncology, studies have been carried out with 13-N-labelled amino acids, as 13-N-methionin, to investigate tumor growth and viability (Blokland et al., 2001).

Oxygen-15

Oxygen-15 (15-O) is a very valuable tracer in biology. Oxidation is a fundamental phenomenon in the life of higher organisms. Probably, the most reliable index of tissue metabolism is its rate of oxidation. Metabolic oxidation is a process, which in most of its phases is comparable in time scale to the half-life of oxygen. Therefore, 15-O can indeed be used to study many phases of metabolism. The combination of the need for a tracer of oxygen and the fact that most phenomena studied that way are short in duration has made 15-O a useful tracer in biological and medical studies. 15-O can be used as a tracer for oxygen, carbon monoxide, and carbon dioxide and as water (Blokland et al., 2001).

Fluorine-18

Among all the positron emitting radiopharmaceuticals Fluorine-18 is the most popular and widely used radiopharmaceutical in PET imaging. Cyclotron-produced

radionuclides are generally produced and used at a single site. Due to the relatively short half-life of the radionuclides, these radiopharmaceuticals are not suitable for distribution over longer distances. The one exception is 18-F, which has a 110 min half-life. Thus, 18-FDG and other 18-F-radiopharmaceuticals can be distributed commercially over regions extending at least several hundred kilometers from the site of production. 18-FDG is especially valuable in detecting primary tumors and metastatic disease (staging). It has been shown to be highly useful clinically especially in oncology (McCready et al., 2000). The ability to perform whole body imaging of cancer patients at high risk of both primary and metastatic recurrence is helpful in diagnosis and staging. In total, PET provides a better selection of patients for specific therapies, whether it is surgery, radiation therapy or chemotherapy. It has also shown to be able to monitor the effect of therapy, which has significant clinical implications (Blokland et al., 2001). The various advantages of fluorine based 18-FDG compared to other positron emitting radiopharmaceutical are listed below.

Advantages of 18-FDG:

1. Relatively long half-life for a positron emitter of 110 minutes.
2. It can be produced in large quantities by small hospital based cyclotrons.
3. Commercial distribution of F-18 has become relatively widespread.
4. Most common cancers can be easily imaged and detected with F-18.
5. The biochemical basis for FDG uptake is firmly linked to glycolysis, and the pharmacology of this substrate is well understood.

2.6. Reconstruction of PET imaging

In the past, filter back projection was used, but a newer technique called iterative reconstruction is favored because it eliminates some of the artifacts generated with filtered back projection. The reconstructed data can be displayed in a three-dimensional rotating volume as well as standard tomographic slices in the transaxial, coronal, and sagittal planes. Iterative reconstruction methods applied to image reconstruction in three-dimensional (3D) positron emission tomography (PET) should result in possibly better images than analytical reconstruction algorithms. However, the long reconstruction time has remained an obstacle to their development and, moreover, their clinical routine use. Together with the constant increase in performances of the computing platforms, recent developments in parallel processing techniques offer practical ways to speed up the calculations and attain clinically viable processing rates (Barrett T et al., 1997).

Various analytical (exact and approximate) and iterative algorithms have been proposed for 3D reconstruction in PET. The reconstruction algorithms already implemented or under development includes: The reprojection algorithm (PROMIS), the Fourier rebinning algorithm (FORE), the maximum likelihood by expectation maximisation (ML-EM) algorithm, the ordered subsets, expectation maximisation (OSEM) algorithm, the maximum a posteriori, expectation maximisation (MAP-EM) algorithm, the least squares (LSQ) algorithm and variants of it including the image space reconstruction algorithm (ISRA) and ordered subsets ISRA (OSISRA), the ordered subsets, Mirror (OS-MIRROR) algorithm, the algebraic reconstruction technique (ART) (labbe et al., 1999).

2.7. Attenuation of PET imaging

Attenuation is the inevitable loss of information in an image due to the interaction of emitted photons with matter, through the photoelectric effect (photon absorption), and the Compton effect (photon scatter). Attenuation, caused by scatter and absorption photons, causes pronounced effects in coincidence imaging. In addition to general loss of counts and quantitative accuracy, nonuniformity and distortions are introduced in reconstructed images when attenuation is not corrected (Turkington et al., 2000).

In photoelectric absorption an atom absorbs a photon and in the process electron is ejected from one of its bound shells. The probability of photoelectric absorption increases rapidly with increasing atomic number of the absorber atom, and decreases rapidly with increasing photon energy (Evans et al., 1955).

In water, the probability of photoelectric absorption decreases with roughly the 3rd power of photon energy and is negligible at 511 keV (Johns et al., 1983). In theory the attenuation effects in PET radionuclide imaging method can be exactly compensated before image reconstruction. If 511 keV annihilation gamma rays were made to travel through a substance with a very high atomic number, such as lead brick, only a few of the photons would pass completely through brick unaltered. Most of the photons would interact with the atoms of lead and may undergo photoelectric effect, which involves an atomic electron and the nucleus of the lead atom. The photon completely disappears in this process. It is totally absorbed by the lead, its energy transferred to the nucleus and a fast moving atomic electron. Other gamma rays passing through the lead brick would interact by a process called scattering.
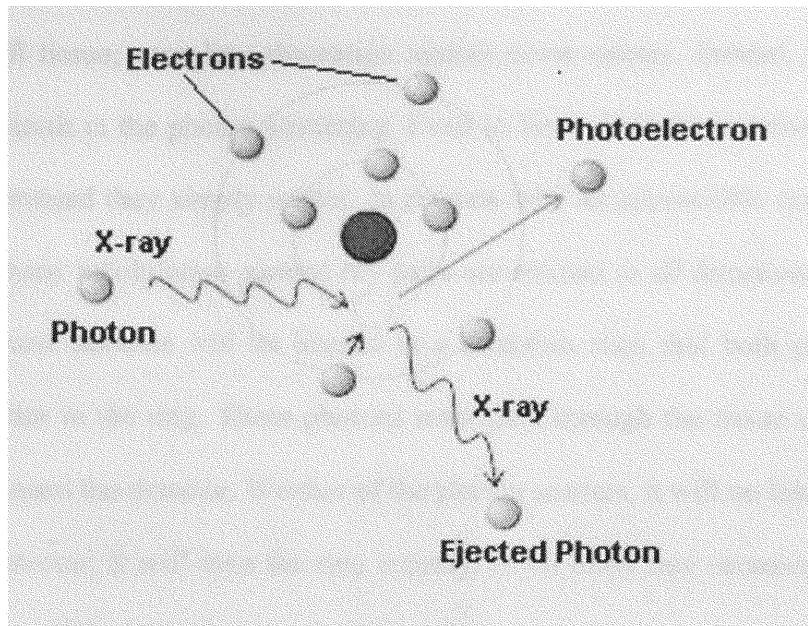
Figure 5. Photoelectric effect. Radiation impinges on the electron within an inner shell and ejects from the atom. (Source: http://www.eee.ntu.ac.uk)
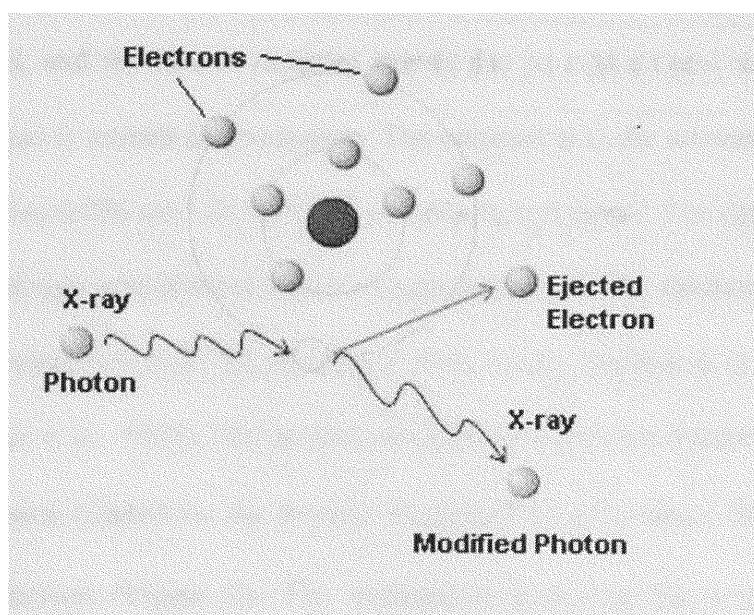
Figure 6. Compton effect. It takes place when high X-ray energy photons collide with an electron. (Source: http://www.eee.ntu.ac.uk).

In soft tissue, complete absorption almost never occurs. Instead, essentially all interactions result in the photon scattering. Even in bone, 511keV photons are absorbed only rarely. Instead they simply scatter. In regions with an appreciable concentration of radiotracer, many annihilation gamma ray pairs are emitted in all directions. Some small fraction of these photons will be headed in a direction such that both photons would strike a detector in the ring. These photons must pass through the tissue of the body as they travel toward the detector. If either of the photon scatters, it will no longer be headed toward the detector. It will miss the ring entirely, or on those rare occasions that it does not, its energy will be too reduced to be detected.

A coincidence event that would have occurred in the absence of intervening tissue now does not occur. The photons emanating from this small section of the body have been attenuated, and the loss of detected events due to interactions with atoms of the intervening tissue is termed as attenuation. The constant μ is the attenuation coefficient, which has a value 0.096 cm^{-1} for 511 keV photons in soft tissue. The number of photons that make it through unscathed decrease exponentially with the thickness of interposed tissue. Lower energy photons are attenuated more easily, because μ is higher at lower energies (Botker et al., 1998). The number of photons reaching a detector is equal to the number of photons headed for the detector multiplied by $e^{-\mu a}$, where 'a' is the distance traveled by photons (Figure 7). The attenuation occurring for a pair of photons (coincidences) can be calculated by the equation:

Coincidences = Number of photons headed in the right direction $\bullet$ $e^{-\mu a}$

This can be demonstrated by considering:

Number of photons hitting detector 1 = $e^{-\mu x}$

Number of photons hitting detector 2 = $e^{-\mu (a-x)}$

Number of photons hitting both (coincidences) = $(e^{-\mu x}) \bullet (e^{-\mu (a-x)}) = e^{-\mu a}$
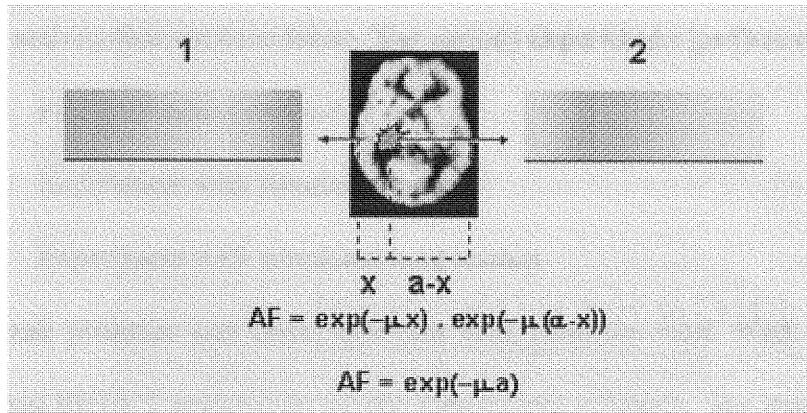
Figure 7. Calculation of Attenuation Coefficient.

Photons, which scatter lead to a loss in, detected events along the LOR's where the events would otherwise have been recorded. The attenuation can lead to several detrimental effects like overall lose of counts, leading to higher image noise; image nonuniformity, due to the differential attenuation of photons from some body regions than others; and distortions, due to the differential attenuation of photons from a particular source location, as a function of angle. Overall losses of counts in the body due to attenuation can be as high as 85% for a moderately large body, but the loss varies substantially with body size, resulting in varying noise levels.

3. MATERIALS AND METHODS

The accuracy of activity quantitation of 18-FDG in simulated tumors and simulated lung lesions was performed by calculating the 3D pixel counts of the simulated tumors and lung lesions and comparing the calculated activities of sum of pixel count of all the transaxial slices, mean and maximum pixel count of one single transaxial slice with the real activity that we have taken. The various experiments performed wee,

1. PET calibration method

2. Experiment with simulated tumors

3. Experiment with simulated lung lesions

Jaszczak phantom and Data Spectrum lung phantom wass used for processing with the hybrid PET/CT scanner for all the above three methods. The experiments were conducted at the Baptist Hospital of Miami, using the Hybrid PET/CT system (Discovery LS, GE Medical Systems). The images were acquired in the DICOM format. DICOM is a standard file format used in storing and transferring medical data. The initial goal in developing a standard for the transmission of digital images was to enable users to retrieve images and associated information from various digital imaging equipment.

The DICOM standard is extremely adaptable, a planned feature that has led to the adoption of DICOM by other specialties that generate images. The fact that many of the medical imaging equipment manufacturers are global corporations has sparked considerable international interest in DICOM (DICOM, 2000). The DICOM standard has become the predominant standard for the communication of medical images. DICOM also provides a means by which users of imaging equipment may assess whether two

devices claiming conformance will be able to exchange meaningful information. The software for processing the images and calculating the pixel count was developed using MATLAB (Mathworks Inc., Natick MA).

3.1. PET-CT Scanner

A Hybrid PET/CT with a whole-body positron emission tomograph with 2 ring detectors was used to perform positron emission tomography. It has two detector rings and each ring has 4096 detectors (8192 total detectors) with bismuth germanate (BGO) crystal. The LightSpeed (LS) portion of the system provides cubic CT data sets with the smallest practical volume, delivering superior image quality in 3D and multi-planar reformatting. The reconstruction of the PET image is done by a filter back projection method by built in software on the PET-CT scanner computer.

Attenuation Correction of PET Emission Scans

CT images have been used to calculate attenuation values for radiotherapy treatment and for PET values (Blankespoor et al., 1996). In order to use the CT values for attenuation correction, an attenuation map was constructed by converting the CT values into attenuation coefficients at the required energy of 511 keV for coincidence imaging. The attenuation correction was done automatically by the built in software on the PET-CT scanner.

3.2. Radioisotope Calibrator

A radioisotope Calibrator was used to determine the activity of radionuclide 18-FDG directly. It provides fast, accurate radionuclide activity measurements with performance that easily surpasses the most stringent regulatory requirements. The radioisotope calibrator used in our experiment is CRC-15R calibrator (Radiation Products Design, Inc. Albertville, MN). It has various features like,

1. On screen display of Nuclide, Activity, Unit of Measure and Calibration Number.

2. Large character, high visibility display with automatic backlighting.

3. Over 80 Nuclides with half-lives in memory.

4. Built-in dose calibration, quality control and self-diagnostics

5. Resolution of 0.01 μCi.
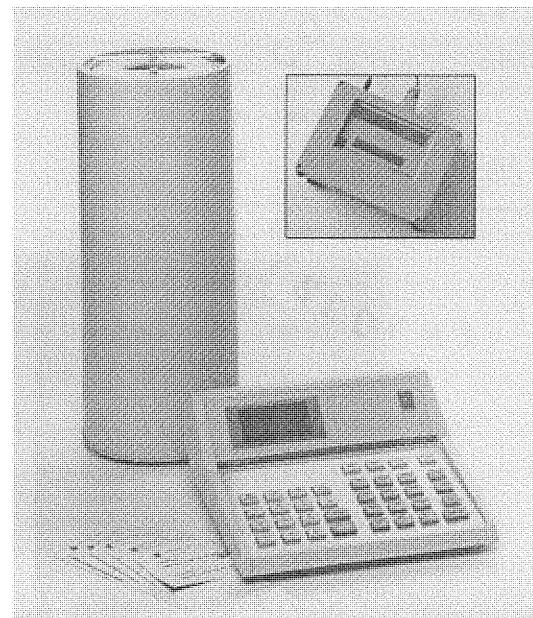
Figure 8. Radioisotope calibrator (Source: www.biodex.com).

To the right of the Activity Display were the units of measure, e.g., micro curies (μCi) or Becquerel (Bq). We can select the required units of activity. Then an isotope is selected, and the atom lab dose calibrator was automatically calibrated to display the activity of that isotope. Press the 18-FDG as Isotope Selection key and observe the LED on the key to come on. Then the 18-FDG source is placed into chamber well and the corresponding activity is found directly from the LED display unit.

3.3. Radio Pharmaceutical

The radioisotope we used was 18-FDG. The isotope 18-F is a positron emitter with a half-life period of 110 minute, bound to deoxyglucose.

3.4. PET Calibration

The phantom we used for the PET calibration is the Jaszczak Phantom (Figure 9), which is cylindrical in shape with interior dimensions 21.6cm * 18.6cm, with the rods and spheres removed.
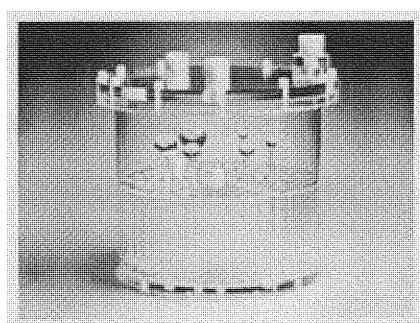
Figure 9. Jaszczak phantom with the rods and spheres installed (Source: http://www.biodex.com).

The specifications of the Jaszczak phantom are:

1. Cylinder Interior Dimensions: 8.5" dia x 7.32" h (21.6 x 18.6 cm)

2. Cylinder Wall Thickness: 0.125" (3.2 mm)

3. Volume: 6.75 L

4. Volume With Inserts: 6.1 L

(All the above information is provided, courtesy www.biodex.com).

In the PET calibration method, we were calibrating for the activity of the injected dose (µCi/ml) in terms of (counts/min/pixel). The concentration values we obtain from the PET images are in terms of counts/pixel. But we need activity values in terms of µCi/ml, which is the unit for activity. Here, in PET calibration method, we are trying to find how many counts per minute are equivalent to 1µCi. A typical PET calibration experiment was performed with the phantom filled with water, and a specific dose of a known amount of activity of 18-FDG is injected into the phantom. The activity of FDG (2.487mCi) was measured, before injecting into the phantom, using the radioisotope dose calibrator. Then the Jaszczak phantom is mounted on the patient table of the Hybrid PET-CT and a projection set is taken for 30 min. Transaxial slices (2D PET images) were obtained by Filter back projection (FBP) reconstruction of the projections. Totally, there were 35 PET transaxial slices obtained and each slice is a 128 * 128 matrix. Attenuation correction of the PET images wass derived following CT imaging.

In our PET calibration method, we were including a spread sheet which has all the important parameters, like name of the phantom, volume of the phantom in 'L', activity of Phantom in 'mCi', average scanning time, decay correction time, decay corrected activity. A typical spreadsheet is shown in chapter 4.

From the transaxial slices obtained, we can choose region of interest (ROI) and the mean pixel values for each ROI are calculated with the help of suitable software. Matlab software was used in our experiment for calculating minimum, mean and maximum pixel counts in PET calibration method and description of calculation is added in the appendix.

3.5 Decay correction of PET calibration method

In our PET calibration method there is a time difference between the radiopharmaceutical injected and PET image acquisition and the radiopharmaceutical will undergo radioactive decay and we need to correct the activity for this decay. The formula used to correct the radioactive decay is,

Decay corrected activity of scan, $A = A_o * exp(-0.693 * t/T_{1/2})$.

where,

A_o = initial activity of phantom at the time of injection, in μCi.

t = decay correction time, which is the difference between average scanning time and the
    time at the activity of injecting the radiopharmaceutical in the phantom, in min.

$T_{1/2}$ = half life time of 18-FDG which is 110 min.

The decay corrected activity of scan is shown in chapter 4.

3.6. Uniformity calculation:

The purpose of calculating uniformity is to determine, how good is the distribution of the pixel counts uniformly in the image so that we can determine whether

the mean pixel counts would be appropriate for calculating the calibration factor. The formula for calculating uniformity is,

$$\text{Uniformity}(\%) = \frac{(\text{maximum pixel counts} - \text{minimum pixel counts}) \; * 100}{(\text{maximum pixel counts} + \text{minimum pixel counts})}$$

A non-uniformity of less than 15 % will determine that the pixel counts are distributed uniformly in the image and mean pixel counts would be better for calculating calibration factor. A graph between slice number and uniformity was drawn to determine the average uniformity and its standard deviation. Also a graph between slice number and counts per minute (cpm) per pixel is drawn to determine the average cpm per pixel and its standard deviation.

3.7 Experiment with simulated tumors

The Experiment with simulated tumors was performed using the Jaszczak phantom filled with water containing 6 spheres, each sphere containing different volumes and different activities of 18-FDG. Here each sphere corresponds to a tumor, and the activity of each sphere corresponds to tumor activity. The volume of the spheres is 15 ml, 8 ml, 4 ml, 2 ml, 1 ml, and 0.5 ml respectively.

Then, 18-FDG is added to the spheres as follows; 2.9 mCi in 15 ml sphere, 2.69 mCi in 8 ml sphere, 2.36 mCi in 4 ml sphere, 1.84 mCi in 2 ml sphere, 2.59 mCi in 1 ml sphere, 2.04 mCi in 0.5 ml sphere. The range of activities used in this study is approximately same level of activity found in clinical studies. There is no specific activity characterizing tumors and activity in tumor depends on size of tumor, metabolic activity, kind of tumor, image time and various other factors. So we used an approximate range of activity/ml in this experiment which is usually found in clinical studies. The

phantom containing simulated tumors was then mounted on the patient table of the PET-CT scanner and projection set was taken for 5 min. Transaxial slices (2D PET images) were obtained by FBP reconstruction of the projections. Totally there are 60 transaxial slices in which we have seven transaxial slices containing simulated tumors. Attenuation correction of PET imaging is derived following CT imaging. A typical table showing all the values of the simulated tumor experiment and the images containing simulated tumors are shown in chapter 4.

3.8 Experiment with simulated lung lesions

A lung lesion is something in the lung that can be inflammatory, a benign tumor or a malignant tumor. Simulated lung lesion can be approximated by a sphere with higher uptake of the radiotracer 18-FDG. The experiment with simulated lung lesion was performed with Data Spectrum lung phantom with elliptical cylinder. The Data spectrum lung phantom consists of two chambers that are shaped to simulate the lungs. The chambers can be filled with material that mimics the lung tissue. For example, when packed with styrofoam beads and filled with a radioactive solution, lung chambers simulate lung tissue with density of ~0.3 gm/cm^3 and with any desirable radioactivity concentration.
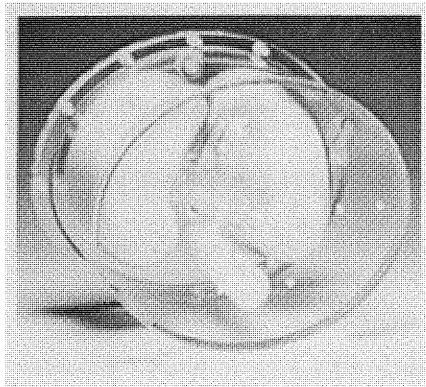
Figure 10. Data Spectrum lung phantom with elliptical cylinder

The specifications of Data Spectrum lung phantom are :

Inside Diameter Elliptical Shape:

1.   Diameter along major axis: 12.2" (30.5 cm)

2.   Diameter along minor axis: 8.7" (22.1 cm)

3.   Inside Height: 7.3" (18.6 cm)

Volume:

1.   Empty cylinder: ~ 9.4 L

2.   Right Lung (w/o Styrofoam beads): ~ 1.1 L

3.   Left Lung (w/ Styrofoam beads): ~ 0.36 L

4.   Right Lung (w/ Styrofoam beads): ~ 0.44 L

Two small lung lesions containing ^{18}F were simulated, one in the right lung (0.34 ml and 0.8478 μCi) and the second one in the left lung (0.02 ml and 0.0981 μCi), in a Data spectrum lung phantom. We mentioned left lung has lung 1 and right lung has lung 2.

We were doing our data analysis for the lung lesions and compare the results of activity of sum of pixel counts of all the transaxial slices, mean and maximum pixel count of one single transaxial slice with the real activity. A typical table showing all the values of the simulated lung lesion and a PET image of a simulated lung lesion are shown in chapter 4.

3.9 Region of Interest (ROI) and Threshold method

The software used for drawing ROIs is Matlab. The matlab program written for this is included in the appendix. The ROI was drawn for a particular sphere in one of the transaxial slice and this ROI is going to be the same for all the transaxial slices of that particular sphere for the calculation of pixel counts. The ROI is drawn in such a way so that we can eliminate the pixel counts having zero values using a command called 'sparse'. We can also calculate the sum, mean and maximum pixel counts of each transaxial slices and also the overall pixel counts sum. Threshold values like 10%-50% of the maximum pixel counts are found and the pixel counts which are less than this threshold value are eliminated from the pixel count calculation. The basic purpose of applying threshold values is to eliminate the pixel counts that are not really contributing to the activity of the tumors and to increase the accuracy of the calculation of uptake. Choosing the threshold value is arbitrary and from the threshold value pixel count we can identify which threshold gives better accuracy for calculation of activity for particular sphere.

The mean, maximum pixel counts of individual transaxial slices and overall sum of all the transaxial slices for the threshold values were also determined. ROI was also

chosen for other spheres using the same program. Images containing spheres are shown in chapter 4. By using this matlab program code we can obtain all the pixel counts of all the transaxial slices for one particular ROI. In our experiment we assume that there is no background activity for the tumors which will not be the case in clinical studies with patients. So we need to include the background correction accordingly if we are going to use this matlab program for calculating pixel counts for patient studies. The ROI on this work has arbitrary size in which same ROI for all the transaxial slices will introduce an error. Since we don't have background activity this error is negligible. But in clinical studies, real background assists the ROI and the ROI must fit the image of the lesions in each transaxial slice.

3.10. Linear Regression Analysis

The goal of linear regression analysis is to find the "best fit" straight line through a set of y vs. x data.

The linear regression analysis was applied for sum of all the pixel counts, mean and maximum pixel counts of individual transaxial slice for all the six spheres and from the correlation coefficient values we are trying to determine which one among sum, mean and maximum pixel count is better to calculate the activity of tumor. Calculation of pixel counts is discussed in chapter 4. We have chosen threshold values like 10%-50% of maximum pixel counts and the pixel counts, which are lesser than those threshold values, are excluded from linear regression analysis. Threshold values are applied for all the six spheres and their corresponding sum of all the pixel counts and mean pixel counts of individual transaxial slice are found.

3.11. Drawing Histograms

In our Experiment with simulated tumors we draw histograms with horizontal (x) axis corresponding to pixel counts, while the vertical axis (y) axis represents the frequency of each class or category. Frequency is determined assuming 10 intervals (bins) of pixel counts values. Histograms were drawn for all the six spheres and two simulated lung lesions and is shown in chaper 4. Its distribution is discussed in chapter 5. A table named histogram analysis is added in chapter 4 with all parameters like mean (µCi), median (µCi), kurtosis and skewness calculated for all the histograms. We are trying to verify whether the distribution follows a normal gaussian distribution.

3.12 Comparison of activity

The activity of each sphere was calculated by dividing the sum of all pixel counts by scanning time so that we can express pixel counts in terms of counts per minute (CPM). Pixel counts per minute multiplied by the calibration factor will give the activity in terms of µCi, which is the calculated activity of the tumor. This activity is compared to the real activity. Comparison of activity is done for all the pixel counts and also for threshold value pixel counts. We can similarly calculate the activity of mean and maximum pixel counts of one particular transaxial slice. Comparison of activities will determine which calculated activity among threshold sum of all pixel counts, mean and maximum pixel counts of one single transaxial slice will give better accuracy with respect to the real activity.

3.13 Errors in the calculation

The error percentage calculation with sum of pixel counts was calculated by considering all the pixel counts in all the transaxial slices whereas for mean and maximum it is calculated from one single transaxial slice. The error percentage is calculated as follows:

$$\text{Error Percentage (\%)} = \frac{(\text{real activity} - \text{calculated activity}) \ * \ 100}{\text{real activity}}$$

The error percentage values will determine which among various threshold sum of all pixel count values will give better accuracy of calculated activity to real activity and also the mean and maximum pixel counts of one single transaxial slice.

3.14 t-test analysis:

The strength of the relation between pixel counts and activity is indicated by the correlation coefficient r and is actually measured by coefficient of determination r^2. The significance of the relation is expressed in probability levels and in our t-test we assume a significance level of 0.05. The t-test analysis is done with a common assumption that there is NO relationship between X and Y in the population. Under this common null hypothesis in correlational analysis: $r = 0.0$. The formula for computing the appropriate t value to test significance of a correlation coefficient employs the t distribution:

$$t = \ r \ \sqrt{(n\text{-}2)/ \ (1\text{-}r^2)}$$

where,

 n-2 = degrees of freedom

r = regression coefficient.

The critical 'r' for a 0.05 significance level is 2.132 and the calculated 'r' is compared with this value to determine the significance of the relation. A table of t-test analysis of linear regression graphs is shown in chapter 4.

4. RESULTS

4.1. Data Analysis of PET calibration method

The spread sheet used for PET calibration method with all the parameters is shown in Table 1 and the images that we obtained in PET calibration method is shown in Figure 10. Twelve images among 35 transaxial slices are shown in Figure 10. With the help of the Matlab software we can choose a ROI in all 35 transaxial slices and calculate the minimum, mean and maximum pixel counts in the ROI. All these data are shown in Table 2. The average counts per minute (cpm)/pixel in all the transaxial slices is found to be 303.96 cpm/pixel. The volume of one pixel is 0.064 cm^3 and dividing cpm/pixel by this amount will give the cpm/ml, which is 4745.94 cpm/ml. We already know the concentration of 18-FDG in the phantom and the volume of the phantom, so we can calculate the concentration in terms of the unit μCi/ml. The ratio of μCi /ml to that cpm/ml gave the calibration factor. The calibration factor in our PET calibration method is found to be $4.84*10^{-5}$ μCi/CPM. We can use this value in finding the real activity of the simulated tumors in our tumor experiment. The sequence of calculations used in calculating calibration factor is shown in Table 3.

The uniformity in terms of percentage was calculated to determine the significance of using the mean pixel counts per minute for calculating the calibration factor. The uniformity percentage shows that the uniformity is less than 15% and the distribution of pixel counts is even in the image. Hence we assume mean pixel counts per minute for calculating the calibration factor instead of minimum or maximum pixel counts in the chosen ROI. While doing uniformity calculations, the pixel counts of first

two and last two transaxial slices were not considered since those values would not be exact due to the artifacts that could be accounted at the two sides of the phantom.

PET Calibration							
Date	10/29/2002						
PET Scanner	GE LS Discovery						
Phantom	Jaszczak						
Phantom Values							
Volume of Phantom (L)	Activity of Full Syringe(mci)	Time at the Activity of full Syringe(min)	Activity of Empty Syringe(mCi)	Time at the Activity of Empty Syringe(min)	Activity of Phantom (mCi)	Concentration (µCi/ml)	
6.75	3.13mci	6:46	0.643mci	6:47	2.487mci	0.368	
PET Scanner Values							
Starting time (min)		End Scanning time(min)		Scanning duration(min)		Decay Correction time(min)	
7:44		8:14		30		72 min	
Decay corrected Activity during scan, A :		2.487 exp(-.693*72/110)					
		= 1.579 mCi					
		= 1579 µCi					

Table 1. PET Calibration Data
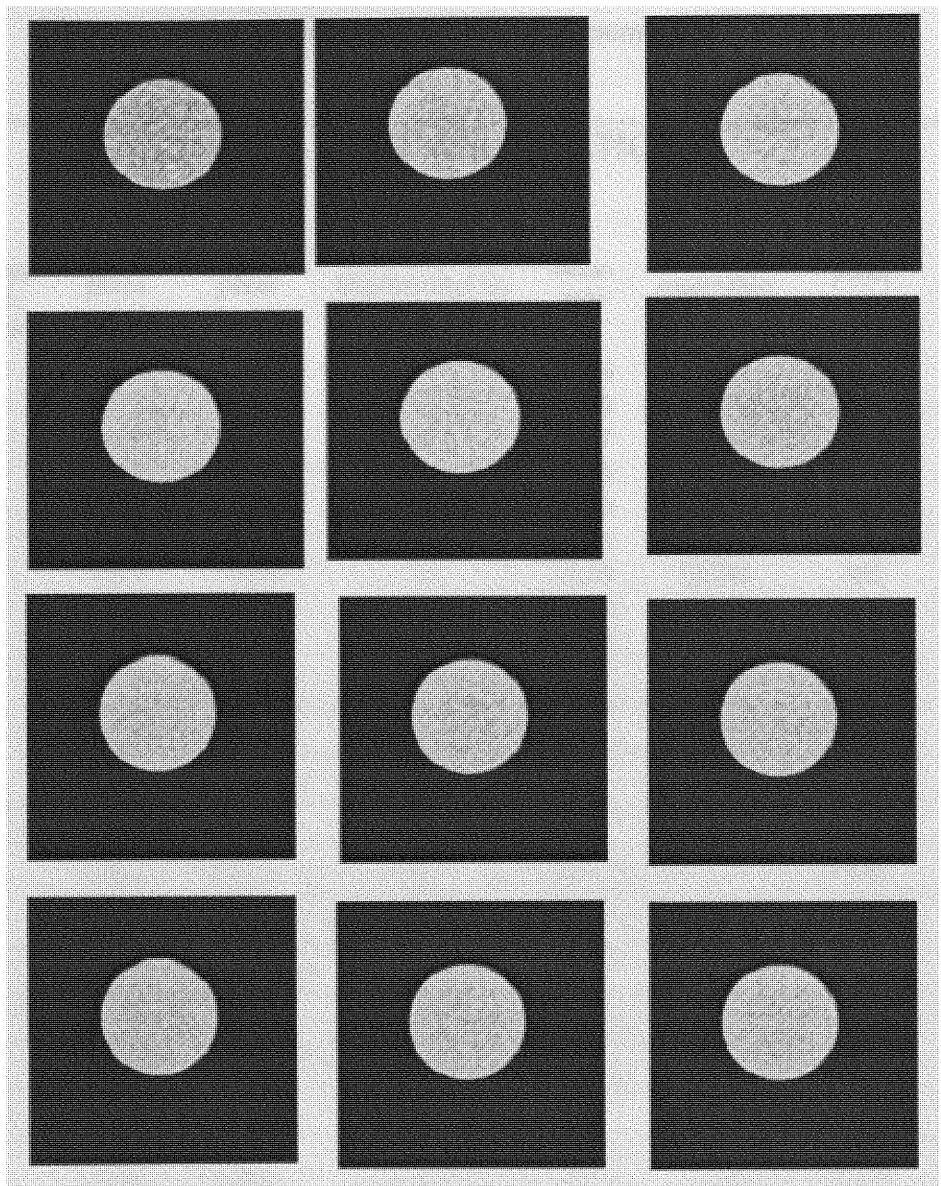
Figure 11. PET Calibration images

Table 2. PET Calibration pixel count values

PET CALIBRATION							
Minimum pixel counts	Mean pixel counts	Max pixel counts	Uniformity(%)	max-min	max+min	Mean pixel counts per min	Mean Error (%)
6788	8819	10974	24	4186	17761	294	1.23
7486	9072	10592	17	3106	18078	302	1.20
7770	9158	10537	15	2767	18307	305	1.19
7590	9048	10560	16	2969	18150	302	1.20
7304	8994	10672	19	3368	17977	300	1.21
7868	8968	10602	15	2734	18470	299	1.21
7607	9069	10436	16	2829	18043	302	1.20
8256	9299	10896	14	2639	19152	310	1.17
7619	9066	11084	19	3465	18703	302	1.20
7765	9053	10585	15	2820	18349	302	1.20
7958	9106	10503	14	2545	18461	304	1.19
7505	9097	10540	17	3035	18045	303	1.19
7825	9276	10753	16	2929	18578	309	1.17
7849	9058	10166	13	2317	18015	302	1.20
7325	9063	10676	19	3350	18001	302	1.20
7924	9234	10438	14	2514	18361	308	1.18
8160	9213	11101	15	2941	19261	307	1.18
7904	9066	10674	15	2770	18578	302	1.20
7771	8975	10341	14	2570	18112	299	1.21
8098	9142	11021	15	2923	19119	305	1.19
8148	9196	10360	12	2212	18507	307	1.18
7697	8901	10122	14	2424	17819	297	1.22
7838	8938	10421	14	2583	18260	298	1.21
7735	9274	10711	16	2976	18445	309	1.17
7734	9164	10856	17	3122	18590	305	1.18
8080	9102	10835	15	2755	18915	303	1.19
7900	9051	10406	14	2505	18306	302	1.20
8206	9203	10441	12	2235	18647	307	1.18
7827	9213	10354	14	2528	18181	307	1.18
8136	9116	10227	11	2091	18363	304	1.19
8065	9166	10539	13	2474	18604	306	1.18
8000	9350	10540	14	2540	18541	312	1.16
7822	9128	10579	15	2757	18401	304	1.19
7883	9096	11228	18	3345	19110	303	1.19
6915	9257	12647	29	5732	19562	309	1.17

Table 3. Calculation of calibration factor

Calculation of Calibration Factor:		
1	**Average counts/pixel** :	9112.22 counts/pixel
2	**Volume** :	3.90* 3.90* 4.25 mm3 = 0.064 cm3
3	**Counts/pixel. Volume** :	9112.22/0.064 = 142378.43 Counts/ml
4	**Counts/min/ml** :	142378.43/30 = 4745.94 (counts/min)/ml
5	**Activity/Volume of Phantom :**	1579µCi/6750ml = 0.23uci/ml
6	**Calibration Factor** :	$\dfrac{\mu Ci/ml}{(counts/min)/ml}$ = 0.23/4745.94 = 4.84 e-05 µCi/(counts/min)
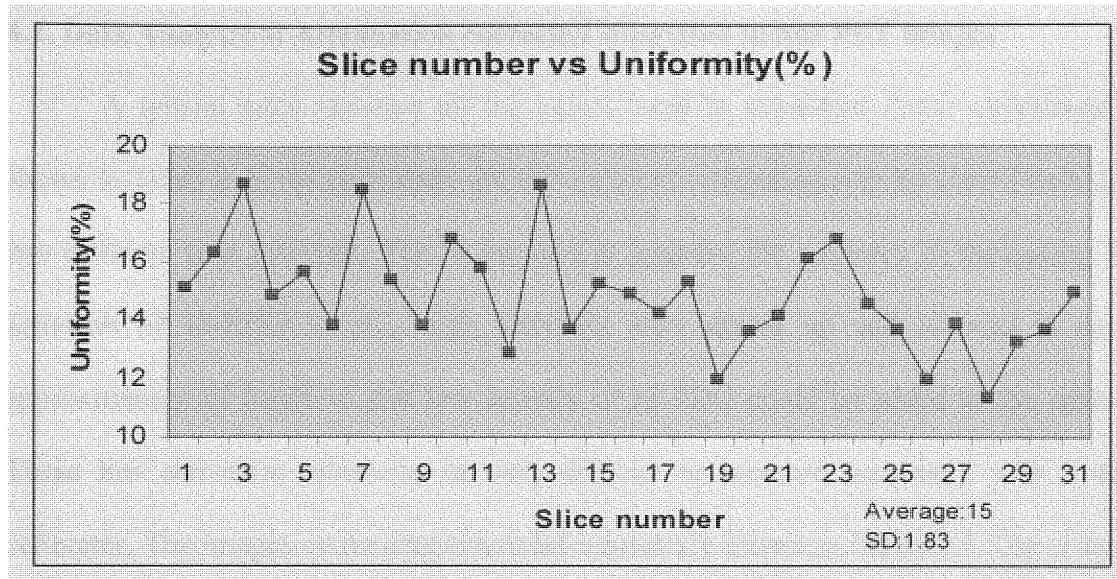
Figure12. Slice number vs Uniformity(%)
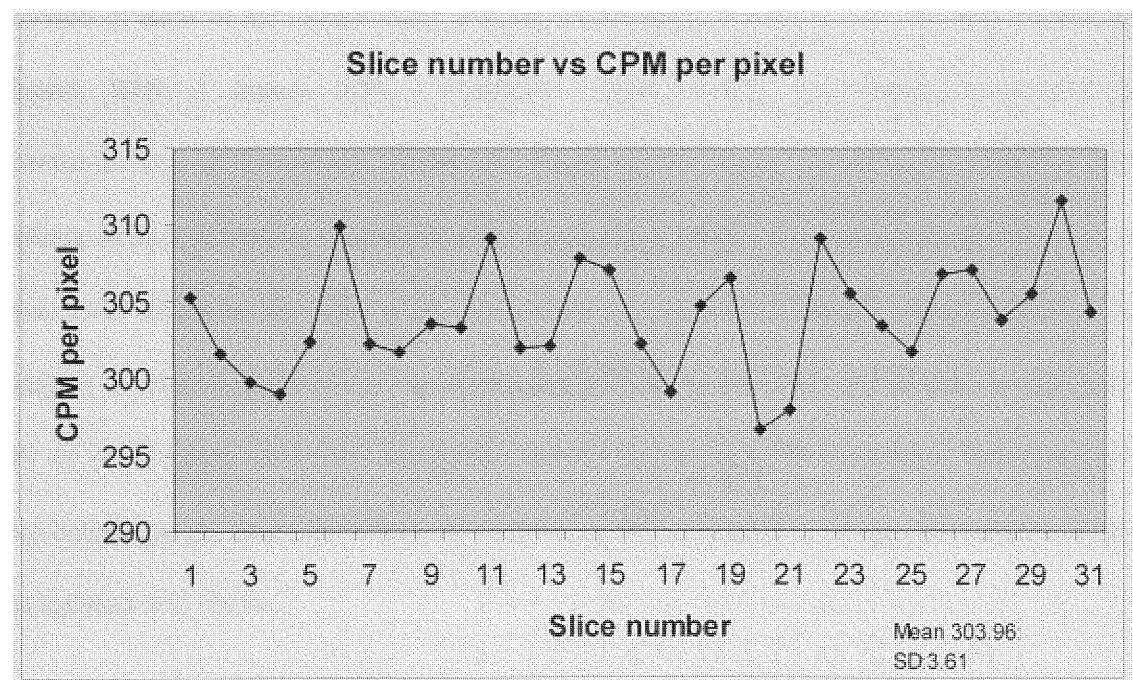
Figure 13. Slice number vs CPM per pixel

4.2. Data Analysis of Attenuation corrected simulated tumor PET images

A typical table showing all the values used in simulated tumor experiment is shown in Table 4. In the table there are two columns containing activity corresponding to two different time intervals. This is because to improve the accuracy of measuring low activities of 18-FDG in μCi, high activities were drawn up and allowed to decay approximately 22 hours so that low activity of 18-FDG could be added to the spheres. These low activation otherwise be very difficult to measure accurately with a dose calibrator. The images containing simulated tumors are shown in Figure 14. The order of spheres in each image is in the counterclockwise direction. The ROI for each sphere is chosen in each of the seven slices and the entire pixel counts for all the ROI chosen are found. ROI is drawn with a computer mouse along line segments around the circumference of the sphere of interest. Opening the image, choosing the ROI for each sphere in that particular transaxial slice and finding the pixel values of each sphere in all transaxial slices is all done with the help of the software, Matlab. The program code written for this is included in the Appendix. The pixel counts for all of the six spheres in all of the 7 slices and their threshold value pixel counts are tabulated. The mean and maximum pixel counts for all the six spheres and the over all sum for each sphere is also calculated and shown.

Sphere No	Volume (ml)	Date - time	Activity injected (mCi)	Scanning time	Scan time (min)	Activity at the time of experiment (μCi)	Act. (μCi/ml)
1	15	3/26/03 11:04	2.9	3/27/03 9:01	5	0.7228	0.0482
2	8	3/26/03 11:05	2.69	3/27/03 9:01	5	0.6747	0.0843
3	4	3/26/03 11:06	2.36	3/27/03 9:01	5	0.5957	0.1489
4	2	3/26/03 11:06	1.84	3/27/03 9:01	5	0.4644	0.2322
5	1	3/26/03 11:07	2.59	3/27/03 9:01	5	0.6578	0.6578
6	0.5	3/26/03 11:07	2.04	3/27/03 9:01	5	0.5181	1.0363

Table 4. Simulated tumor experiment data
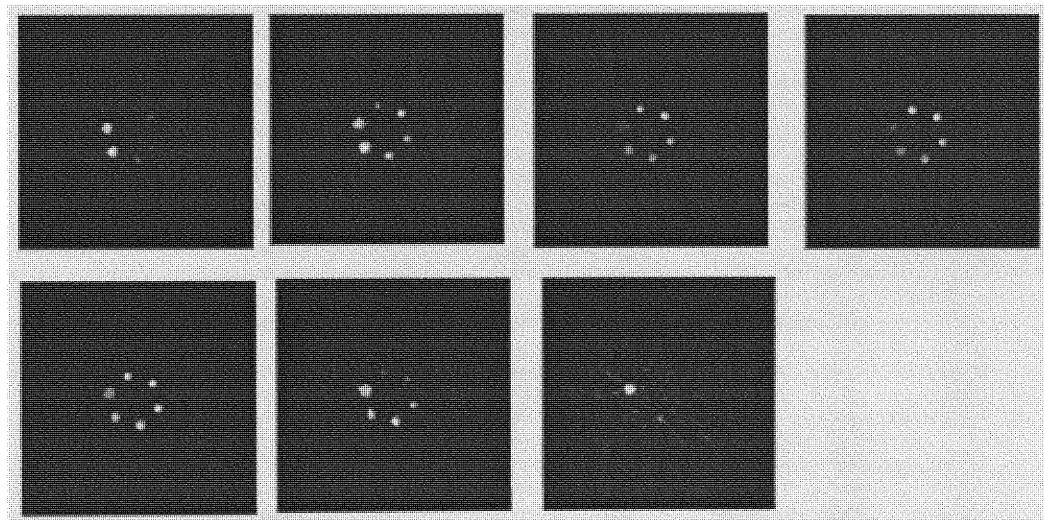
Figure 14 . Transaxial slices 28-34 of attenuation corrected PET image.

Table 5. 3D pixel count values of sphere 1 of attenuation corrected images

SPHERE 1:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
311	235	63	103	149	207	193
136	619	222	149	503	611	172
14	451	229	175	395	426	287
495	108	117	153	237	251	131
819	451	218	179	440	558	664
1043	722	252	198	599	725	783
825	826	302	236	627	884	699
381	773	292	216	550	759	629
164	458	174	152	412	589	241
119	160	104	70	147	387	425
287	20	75	230	340	110	1168
710	436	180	259	538	814	1573
1207	712	262	285	603	917	1598
1559	880	260	297	577	907	1381
1309	884	249	237	578	988	709
650	883	242	151	513	1044	226
227	669	216	92	238	933	208
127	359	188	43	23	501	566
334	192	85	198	435	76	1271
761	726	212	268	591	647	1788
1237	1071	269	294	611	975	1966
1580	1140	258	279	577	921	1735
1372	1025	253	226	527	892	1062
730	904	271	164	469	1026	403
261	678	292	143	296	989	998
648	461	287	257	560	422	1529
1061	1044	280	288	607	892	1665
1348	1111	276	225	603	947	1356
1171	1008	289	182	474	916	834
585	860	295	141	405	1037	742
295	622	281	194	421	883	1195
620	583	213	214	495	636	1160
919	710	245	162	499	842	780
765	697	243	156	389	883	418
252	635	202	109	330	905	541
239	451	158	82	250	636	472
420	301	134	83	242	532	244
279	328	140	112	228	561	30
173	319	82	28	15	526	
	69	63			229	
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
652	614	212	180	423	700	838
Max:	Max:	Max:	Max:	Max:	Max:	Max:
1580	1140	302	297	627	1044	1966
SUMALL:						
141831						

44

Table 6. 10 % Threshold value pixel count values of sphere 1

10 % Threshold:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
311	235	222	198	503	207	287
495	619	229	236	395	611	664
819	451	218	216	237	426	783
1043	451	252	230	440	251	699
825	722	302	259	599	558	629
381	826	292	285	627	725	241
287	773	262	297	550	884	425
710	458	260	237	412	759	1168
1207	436	249	198	340	589	1573
1559	712	242	268	538	387	1598
1309	880	216	294	603	814	1381
650	884	212	279	577	917	709
227	883	269	226	578	907	226
334	669	258	257	513	988	208
761	359	253	288	238	1044	566
1237	726	271	225	435	933	1271
1580	1071	292	214	591	501	1788
1372	1140	287		611	647	1966
730	1025	280		577	975	1735
261	904	276		527	921	1062
648	678	289		469	892	403
1061	461	295		296	1026	998
1348	1044	281		560	989	1529
1171	1111	213		607	422	1665
585	1008	245		603	892	1356
295	860	243		474	947	834
620	622	202		405	916	742
919	583			421	1037	1195
765	710			495	883	1160
252	697			499	636	780
239	635			389	842	418
420	451			330	883	541
279	301			250	905	472
	328			242	636	244
	319			228	532	
					561	
					526	
					229	
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
748	687	256	247	462	732	921
SUMALL:						
135116						

45

Table 7. 20 % Threshold value pixel count values of sphere 1

20 % Threshold:				
Slice 28:	Slice 29:	Slice 32:	Slice 33:	Slice 34:
495	619	503	611	664
819	451	395	426	783
1043	451	440	558	699
825	722	599	725	629
710	826	627	884	425
1207	773	550	759	1168
1559	458	412	589	1573
1309	436	538	814	1598
650	712	603	917	1381
761	880	577	907	709
1237	884	578	988	566
1580	883	513	1044	1271
1372	669	435	933	1788
730	726	591	501	1966
648	1071	611	647	1735
1061	1140	577	975	1062
1348	1025	527	921	403
1171	904	469	892	998
585	678	560	1026	1529
620	461	607	989	1665
919	1044	603	422	1356
765	1111	474	892	834
420	1008	405	947	742
	860	421	916	1195
	622	495	1037	1160
	583	499	883	780
	710		636	418
	697		842	541
	635		883	472
	451		905	
			636	
			532	
			561	
			526	
Mean:	Mean:	Mean:	Mean:	Mean:
808	750	523	786	1038
SUMALL:				
114764				

Table 8. 30 % Threshold value pixel count values of sphere 1

30 % Threshold:				
Slice 28:	Slice 29:	Slice 32:	Slice 33:	Slice 34:
819	619	599	611	664
1043	722	627	725	783
825	826	603	884	699
710	773	591	759	629
1207	712	611	814	1168
1559	880	607	917	1573
1309	884	603	907	1598
650	883		988	1381
761	669		1044	709
1237	726		933	1271
1580	1071		647	1788
1372	1140		975	1966
730	1025		921	1735
648	904		892	1062
1061	678		1026	998
1348	1044		989	1529
1171	1111		892	1665
620	1008		947	1356
919	860		916	834
765	622		1037	742
	710		883	1195
	697		636	1160
	635		842	780
			883	
			905	
			636	
Mean:	Mean:	Mean:	Mean:	Mean:
1017	835	606	870	1186
SUMALL:				
93665				

47

Table 9. 40 % and 50 % Threshold pixel counts value of sphere 1

40 % Threshold:					50 % Threshold:			
Slice 28:	Slice 29:	Slice 33:	Slice 34:		Slice 28:	Slice 29:	Slice 33:	Slice 34:
819	826	884	1168					
1043	880	814	1573		1043	1071	988	1168
825	884	917	1598		1207	1140	1044	1573
1207	883	907	1381		1559	1025	1026	1598
1559	1071	988	1271		1309	1044	989	1381
1309	1140	1044	1788		1237	1111	1037	1271
1237	1025	933	1966		1580	1008		1788
1580	904	975	1735		1372			1966
1372	1044	921	1062		1061			1735
1061	1111	892	998		1348			1062
1348	1008	1026	1529		1171			998
1171	860	989	1665					1529
919		892	1356					1665
		947	834					1356
		916	1195					1195
		1037	1160					1160
		883						
		842			Mean:	Mean:	Mean:	Mean:
		883			1289	1066	1017	1430
		905						
					SUMALL:			
Mean:	Mean:	Mean:	Mean:		45815			
1188	970	930	1392					
SUMALL:								
67959								

Table 10. 3D pixel count values of sphere 2 of attenuation corrected images

SPHERE 2:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
224	96	44	8	71	29	122
516	363	58	95	272	245	53
460	542	107	152	302	383	59
192	558	253	153	96	239	65
62	514	247	101	53	6	168
111	387	88	13	347	114	212
71	120	7	82	615	355	115
274	199	34	256	660	781	280
888	703	163	365	458	814	159
1346	1187	291	338	179	427	50
1108	1296	370	282	32	114	134
507	1197	459	208	15	62	60
178	872	411	92	136	84	37
65	349	261	37	545	377	213
308	61	171	138	833	1014	216
890	302	102	339	910	1240	219
1664	995	172	480	783	804	334
1966	1644	363	479	445	305	246
1498	1711	463	443	140	92	18
615	1563	537	393	5	12	13
100	1176	529	241	138	383	56
480	518	399	88	541	1071	9
1160	194	212	136	795	1368	111
1860	239	50	322	876	952	87
1965	1027	77	474	800	354	82
1430	1714	289	471	509	230	108
595	1738	436	406	205	829	93
59	1569	557	350	31	1125	109
280	1144	599	227	85	825	100
207	400	432	87	378	313	63
659	39	147	32	613	105	58
1295	115	46	121	709	475	164
1458	871	251	271	597	678	152
1126	1572	433	410	348	545	28
590	1676	556	417	165	266	178
189	1451	571	329	95	14	133
24	969	371	252	28	42	3
28	266	77	155	197	211	
103	18	39	59	417	319	
542	578	217	4	486	262	
744	1145	358	50	358	145	
650	1226	401	117	144	35	

Table 10. Contd…

399	964	357	224	19	46	
208	616	209	293	24	153	
105	254	29	266	21	116	
106	280	45	202	96	54	
195	574	156	116	237	121	
288	496	201	31	238	104	
238	212	157	27	174	51	
121	114	88	85	49	49	
39	127	36	88	2	55	
186	161	1	77	79		
220	250	94	62	14		
87	161	56	13	29		
128	195	11	21	28		
165		4	32	65		
67		10		7		
3		33		105		
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
535	704	226	197	286	369	116
Max:	Max:	Max:	Max:	Max:	Max:	Max:
1966	1738	599	480	910	1368	334
SUMALL:						
131812						

50

Table 11. 10 % Threshold value pixel count values of sphere 2

10 % Threshold:							
Slice 28:	Slice 29:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
224	363	964	253	256	272	245	212
516	542	616	247	365	302	383	280
460	558	254	291	338	347	239	213
274	514	280	370	282	615	355	216
888	387	574	459	208	660	781	219
1346	199	496	411	339	458	814	334
1108	703	212	261	480	545	427	246
507	1187	250	363	479	833	377	
308	1296		463	443	910	1014	
890	1197		537	393	783	1240	
1664	872		529	241	445	804	
1966	349		399	322	541	305	
1498	302		212	474	795	383	
615	995		289	471	876	1071	
480	1644		436	406	800	1368	
1160	1711		557	350	509	952	
1860	1563		599	227	205	354	
1965	1176		432	271	378	230	
1430	518		251	410	613	829	
595	239		433	417	709	1125	
280	1027		556	329	597	825	
207	1714		571	252	348	313	
659	1738		371	224	197	475	
1295	1569		217	293	417	678	
1458	1144		358	266	486	545	
1126	400		401	202	358	266	
590	871		357		237	211	
542	1572		209		238	319	
744	1676		201			262	
650	1451						
399	969						
208	266						
288	578						
238	1145						
220	1226						
Mean:	Mean:		Mean:	Mean:	Mean:	Mean:	Mean:
819	868		380	336	517	593	246
SUMALL:							
119111							

51

Table 12. 20 % and 30% Threshold value pixel count values of sphere 2

20 % Threshold:						30 % Threshold:				
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 28:	Slice 29:	Slice 30:	Slice 32:	Slice 33:
516	542	459	480	615	781	888	703	599	615	781
460	558	411	479	660	814	1346	1187		660	814
888	514	463	443	458	427	1108	1296		833	1014
1346	703	537	474	545	1014	890	1197		910	1240
1108	1187	529	471	833	1240	1664	872		783	804
507	1296	399	406	910	804	1966	995		795	1071
890	1197	436	410	783	1071	1498	1644		876	1368
1664	872	557	417	445	1368	615	1711		800	952
1966	995	599		541	952	1160	1563		613	829
1498	1644	432		795	829	1860	1176		709	1125
615	1711	433		876	1125	1965	1027		597	825
480	1563	556		800	825	1430	1714			678
1160	1176	571		509	475	595	1738			
1860	518	401		613	678	659	1569			
1965	1027			709	545	1295	1144			
1430	1714			597		1458	871			
595	1738			417		1126	1572			
659	1569			486		744	1676			
1295	1144					650	1451			
1458	400						969			
1126	871						1145			
590	1572						1226			
542	1676						964			
744	1451						616			
650	969									
399	578					Mean:	Mean:	Mean:	Mean:	Mean:
	1145					1206	1251	599	744	958
	1226									
	964					SUMALL:				
	616					73228				
	574									
	496									
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:					
1016	1069	485	448	644	863					
SUMALL:										
95516										

Table 13. 40 % and 50% Threshold value pixel count values of sphere 2

40 % Threshold:					50 % Threshold:		
Slice 28:	Slice 29:	Slice 32:	Slice 33:		Slice 28:	Slice 29:	Slice 33:
888	1187	833	814		1346	1187	1014
1346	1296	910	1014		1108	1296	1240
1108	1197	795	1240		1664	1197	1071
890	872	876	804		1966	995	1368
1664	995	800	1071		1498	1644	1125
1966	1644		1368		1160	1711	
1498	1711		952		1860	1563	
1160	1563		829		1965	1176	
1860	1176		1125		1430	1027	
1965	1027		825		1295	1714	
1430	1714				1458	1738	
1295	1738				1126	1569	
1458	1569					1144	
1126	1144					1572	
	871					1676	
	1572					1451	
	1676					1145	
	1451					1226	
	969						
	1145				Mean:	Mean:	Mean:
	1226				1490	1391	1164
	964						
					SUMALL:		
Mean:	Mean:	Mean:	Mean:		48723		
1404	1305	843	1004				
SUMALL:							
62614							

Table 14. 3D pixel count values of sphere 3 of attenuation corrected images

SPHERE 3:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
6	114	39	18	31	126	44
54	139	10	83	30	338	55
53	73	92	188	59	312	4
71	183	108	113	106	7	72
33	135	76	1	192	48	31
63	9	20	35	177	321	1
48	75	71	66	133	567	102
103	216	159	235	106	595	72
132	190	274	392	130	328	173
71	75	308	322	232	28	323
79	71	254	134	447	188	335
120	120	190	33	628	473	168
97	20	114	41	549	1035	102
112	307	5	42	284	1379	379
11	563	33	123	69	1164	545
111	775	206	241	94	548	458
192	716	411	427	415	144	132
249	375	504	567	887	137	132
167	110	553	562	1086	39	180
38	40	479	383	930	234	366
23	32	218	139	549	554	498
86	384	228	37	142	1374	385
281	1030	507	29	59	1966	153
394	1477	594	103	506	1680	67
394	1253	635	297	1131	796	2
205	632	518	523	1319	197	91
181	193	208	670	1146	117	174
377	24	197	716	773	15	103
436	35	499	542	289	117	131
313	309	601	211	29	78	277
104	855	543	38	39	205	296
5	1263	354	32	49	940	259
164	1075	113	3	399	1669	247
265	524	1	180	979	1571	72
180	100	96	441	1229	776	42
103	82	305	648	1141	148	79
69	159	414	686	757	42	17
18	358	322	486	231	30	44
52	551	133	167	37	334	86
80	532	4	17	50	828	165
45	308	22	27	26	876	255

Table 14. Contd…

26	35	13	108	4	445	155
53	17	17	290	161	72	198
44	57	114	427	526	16	136
27	70	187	398	756	82	74
9	100	97	229	792	228	173
38	195	14	47	507	276	12
51	259	36	16	44	227	240
45	92	40	114	26	98	221
3	75	45	177	16	24	116
18	75	95	170	150	11	123
41	66	1	97	236	14	189
21	244	19	30	326	29	58
35	138	76	47	259	81	
91	96	23	83	29	200	
70	154	22	79	25	198	
34	173	8	54	101	13	
49	214	40	14	152	59	
	28		4	55	184	
	187		43	42	209	
	151		2	96	49	
			9	96	136	
			35	37	22	
			44	100		
			43	72		
			61			
Mean:	**Mean:**	**Mean:**	**Mean:**	**Mean:**	**Mean:**	**Mean:**
108	294	194	191	340	397	166
Max:	**Max:**	**Max:**	**Max:**	**Max:**	**Max:**	**Max:**
436	1477	635	716	1319	1966	545
SUMALL:						
103914						

Table 15. 10 % Threshold value pixel count values of sphere 3

10 % Threshold:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
249	216	274	235	232	338	323
281	307	308	392	447	312	335
394	563	254	322	628	321	379
394	775	206	241	549	567	545
205	716	411	427	284	595	458
377	375	504	567	415	328	366
436	384	553	562	887	473	498
313	1030	479	383	1086	1035	385
265	1477	218	297	930	1379	277
	1253	228	523	549	1164	296
	632	507	670	506	548	259
	309	594	716	1131	234	247
	855	635	542	1319	554	255
	1263	518	211	1146	1374	198
	1075	208	441	773	1966	240
	524	197	648	289	1680	221
	358	499	686	399	796	
	551	601	486	979	197	
	532	543	290	1229	205	
	308	354	427	1141	940	
	259	305	398	757	1669	
	244	414	229	231	1571	
	214	322		526	776	
				756	334	
				792	828	
				507	876	
				236	445	
				326	228	
				259	276	
					227	
					200	
					198	
					209	
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
324	618	397	441	666	692	330
SUMALL:						
83391						

Table 16. 20 % Threshold value pixel count values of sphere 3

20 % Threshold:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
394	563	411	427	447	567	545
394	775	504	567	628	595	458
436	716	553	562	549	473	498
	1030	479	523	415	1035	
	1477	507	670	887	1379	
	1253	594	716	1086	1164	
	632	635	542	930	548	
	855	518	441	549	554	
	1263	499	648	506	1374	
	1075	601	686	1131	1966	
	524	543	486	1319	1680	
	551	414	427	1146	796	
	532		398	773	940	
				399	1669	
				979	1571	
				1229	776	
				1141	828	
				757	876	
				526	445	
				756		
				792		
				507		
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
408	865	521	546	793	1012	500
SUMALL:						
64009						

Table 17. 30 % Threshold value pixel count values of sphere 3

30 % Threshold:						40 % Threshold:		
Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:				
775	594	670	628	595		Slice 29:	Slice 32:	Slice 33:
716	635	716	887	1035		1030	887	1035
1030	601	648	1086	1379		1477	1086	1379
1477		686	930	1164		1253	930	1164
1253			1131	1374		855	1131	1374
632			1319	1966		1263	1319	1966
855			1146	1680		1075	1146	1680
1263			773	796			979	796
1075			979	940			1229	940
			1229	1669			1141	1669
			1141	1571			792	1571
			757	776				828
			756	828				876
			792	876				
Mean:	Mean:	Mean:	Mean:	Mean:		Mean:	Mean:	Mean:
1008	610	680	968	1189		1159	1064	1273
SUMALL:								
43828						SUMALL:		
						32870		

50 % Threshold:								
Slice 29:	Slice 32:	Slice 33:						
1030	1086	1035						
1477	1131	1379						
1253	1319	1164						
1263	1146	1374						
1075	1229	1966						
	1141	1680						
		1669						
		1571						
Mean:	Mean:	Mean:						
1219	1175	1480						
SUMALL:								
24987								

58

Table 18. 40 % and 50% Threshold value pixel count values of sphere 3

40 % Threshold:		
Slice 29:	Slice 32:	Slice 33:
1030	887	1035
1477	1086	1379
1253	930	1164
855	1131	1374
1263	1319	1966
1075	1146	1680
	979	796
	1229	940
	1141	1669
	792	1571
		828
		876
Mean:	Mean:	Mean:
1159	1064	1273
SUMALL:		
32870		
50 % Threshold:		
Slice 29:	Slice 32:	Slice 33:
1030	1086	1035
1477	1131	1379
1253	1319	1164
1263	1146	1374
1075	1229	1966
	1141	1680
		1669
		1571
Mean:	Mean:	Mean:
1219	1175	1480
SUMALL:		
24987		

Table 19. 3D pixel count values of attenuation corrected images of sphere 4

SPHERE 4:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
55	97	50	41	46	140	29
11	221	115	57	19	61	81
15	45	59	33	41	8	79
76	138	41	74	46	88	52
12	247	145	76	39	85	153
134	209	123	65	64	47	117
92	42	20	81	96	49	165
152	83	64	69	30	16	202
15	184	38	44	49	24	79
77	163	71	31	63	48	16
255	48	275	103	1	143	50
108	5	343	320	149	145	38
47	397	196	375	416	19	89
39	594	59	278	505	65	81
6	373	31	169	260	109	98
163	65	45	91	34	131	226
321	85	20	30	463	243	263
144	24	15	259	1156	473	161
10	193	223	727	1376	548	98
43	725	573	912	910	325	67
102	1014	742	683	325	163	10
224	712	549	323	14	95	79
128	308	218	98	127	92	142
52	123	15	27	692	48	130
108	50	48	8	1568	96	131
113	77	323	347	1880	298	113
5	89	748	919	1318	748	25
81	219	963	1167	570	990	64
128	621	735	859	149	702	49
101	866	309	353	24	394	24
110	637	29	56	184	161	41
76	320	19	15	506	2	44
99	187	53	23	1090	2	36
155	79	226	248	1336	128	127
201	80	561	678	938	587	180
140	82	734	864	384	822	36
124	123	554	607	105	506	227
186	272	241	206	55	188	
137	351	46	29	107	46	
56	237	17	55	107	95	

Table 19. Contd…

144	104	10	4	170	16	
220	75	37	59	398	223	
258	44	209	265	453	355	
190	118	304	378	274	109	
103	43	226	258	75	34	
120	4	111	95	4	80	
79	74	55	17	100	64	
119		55	55	37	84	
110		62	36	151	39	
6		30	91	59	40	
		27	66	116	31	
		54	51	39		
		51	81	44		
		52		4		
Mean:	**Mean:**	**Mean:**	**Mean:**	**Mean:**	**Mean:**	**Mean:**
109	231	202	243	355	196	97
Max:	**Max:**	**Max:**	**Max:**	**Max:**	**Max:**	**Max:**
321	1014	963	1167	1880	990	263
SUMALL:						
72850						

61

Table 20. 10% Threshold value pixel count values of sphere 4

10 % Threshold:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
255	221	275	320	416	243	202
321	247	343	375	505	473	226
224	209	196	278	260	548	263
201	397	223	259	463	325	227
220	594	573	727	1156	298	
258	373	742	912	1376	748	
190	193	549	683	910	990	
	725	218	323	325	702	
	1014	323	347	692	394	
	712	748	919	1568	587	
	308	963	1167	1880	822	
	219	735	859	1318	506	
	621	309	353	570	223	
	866	226	248	506	355	
	637	561	678	1090		
	320	734	864	1336		
	272	554	607	938		
	351	241	206	384		
	237	209	265	398		
		304	378	453		
		226	258	274		
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
238	448	963	525	801	515	230
SUMALL:						
55414						

Table 21. 20% and 30% Threshold value pixel count values of sphere 4

20 % Threshold:				
Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:
397	573	727	416	473
594	742	912	505	548
725	549	683	463	748
1014	748	919	1156	990
712	963	1167	1376	702
621	735	859	910	394
866	561	678	692	587
637	734	864	1568	822
	554	607	1880	506
		378	1318	
			570	
			506	
			1090	
			1336	
			938	
			384	
			398	
			453	
Mean:	Mean:	Mean:	Mean:	Mean:
696	684	779	887	641
SUMALL:				
41248				
30 % Threshold:				
Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:
594	573	727	1156	748
725	742	912	1376	990
1014	748	683	910	702
712	963	919	692	587
621	735	1167	1568	822
866	734	859	1880	
637		678	1318	
		864	570	
		607	1090	
			1336	
			938	
Mean:	Mean:	Mean:	Mean:	Mean:
738	749	824	1167	770
SUMALL:				
33763				

Table 22. 40% and 50% Threshold value pixel count values of sphere 4

40 % Threshold:				
Slice 29:	**Slice 30:**	**Slice 31:**	**Slice 32:**	**Slice 33:**
1014	963	912	1156	990
866		919	1376	822
		1167	910	
		859	1568	
		864	1880	
			1318	
			1090	
			1336	
			938	
Mean:	**Mean:**	**Mean:**	**Mean:**	**Mean:**
940	963	944	1286	906
SUMALL:				
20949				
50 % Threshold:				
Slice 29:	**Slice 30:**	**Slice 31:**	**Slice 32:**	**Slice 33:**
1014	963	1167	1156	990
			1376	
			1568	
			1880	
			1318	
			1090	
			1336	
Mean:	**Mean:**	**Mean:**	**Mean:**	**Mean:**
1014	963	1167	1389	990
SUMALL:				
13859				

Table 23. 3D pixel count values of attenuation corrected images of sphere 5

SPHERE 5:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
130	168	245	263	193	84	201
61	161	244	385	201	15	169
58	17	120	201	97	312	102
11	647	49	772	42	287	67
176	914	814	1198	2	175	139
222	577	1122	833	605	38	121
219	252	772	224	831	131	42
234	113	274	384	588	440	132
111	98	34	1290	269	514	55
61	461	442	1966	78	315	121
250	1382	1362	1453	4	82	80
398	1966	1966	497	324	35	12
371	1393	1437	35	1127	128	182
245	553	506	318	1641	291	57
66	103	40	1148	1221	363	124
45	477	338	1741	506	223	130
313	1245	1146	1276	103	41	46
426	1787	1772	445	263	1	73
230	1243	1338	30	1024	44	17
28	431	448	110	1554	105	229
49	71	27	545	1128	134	203
245	98	1	836	395	81	60
278	139	49	601	4	94	101
29	431	471	212	137	41	41
117	728	845	10	506	210	56
78	463	655	131	754	155	174
3	108	224	213	535	50	53
47	70	48	165	199	108	14
45	114	49	56	37	94	43
26	16	131	32	154	53	120
75	41	220	45	203	44	99
	127	143	43	180	82	135
	27	44	19	129	156	29
	20	41	15	98	91	87
	5	61	19	45	79	98
	74	84		36	164	
	120	33		45	127	
	77	30		84	54	
	3	7		55	51	
	117	39		1	82	
	2				43	
	6				15	
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
150	401	442	500	385	134	97
Max:	Max:	Max:	Max:	Max:	Max:	Max:
426	1966	1966	1966	1641	514	229
SUMALL:						
81119						

Table 24. 10% Threshold value pixel count values of sphere 5

10 % Threshold:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
222	647	245	263	201	312	201
219	914	244	385	605	287	229
234	577	814	201	831	440	203
250	252	1122	772	588	514	
398	461	772	1198	269	315	
371	1382	274	833	324	291	
245	1966	442	224	1127	363	
313	1393	1362	384	1641	223	
426	553	1966	1290	1221	210	
230	477	1437	1966	506		
245	1245	506	1453	263		
278	1787	338	497	1024		
	1243	1146	318	1554		
	431	1772	1148	1128		
	431	1338	1741	395		
	728	448	1276	506		
	463	471	445	754		
		845	545	535		
		655	836	199		
		224	601	203		
		220	212			
			213			
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
286	879	792	764	694	328	211
SUMALL:						
69291						

66

Table 25. 20% and 30% Threshold value pixel count values of sphere 5

20 % Threshold:					
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:
398	647	814	772	605	440
426	914	1122	1198	831	514
	577	772	833	588	
	461	442	1290	1127	
	1382	1362	1966	1641	
	1966	1966	1453	1221	
	1393	1437	497	506	
	553	506	1148	1024	
	477	1146	1741	1554	
	1245	1772	1276	1128	
	1787	1338	445	395	
	1243	448	545	506	
	431	471	836	754	
	431	845	601	535	
	728	655			
	463				
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
412	919	1007	1043	887	477
SUMALL:					
58591					
30 % Threshold:					
Slice 29:	Slice 30:	Slice 31:	Slice 32:		
647	814	772	605		
914	1122	1198	831		
1382	772	833	1127		
1966	1362	1290	1641		
1393	1966	1966	1221		
1245	1437	1453	1024		
1787	1146	1148	1554		
1243	1772	1741	1128		
728	1338	1276	754		
	845	836			
	655	601			
Mean:	Mean:	Mean:	Mean:		
1256	1203	1192	1098		
SUMALL:					
47534					

Table 26. 40% and 50% Threshold value pixel count values of sphere 5

40 % Threshold:			
Slice 29:	Slice 30:	Slice 31:	Slice 32:
914	814	1198	831
1382	1122	833	1127
1966	1362	1290	1641
1393	1966	1966	1221
1245	1437	1453	1024
1787	1146	1148	1554
1243	1772	1741	1128
	1338	1276	
	845	836	
Mean:	Mean:	Mean:	Mean:
1419	1311	1305	1218
SUMALL:			
42000			
50 % Threshold:			
Slice 29:	Slice 30:	Slice 31:	Slice 32:
1382	1122	1198	1127
1966	1362	1290	1641
1393	1966	1966	1221
1245	1437	1453	1024
1787	1146	1148	1554
1243	1772	1741	1128
	1338	1276	
Mean:	Mean:	Mean:	Mean:
1503	1449	1439	1282
SUMALL:			
36926			

Table 27. 3D pixel count values of attenuation corrected images of sphere 6

SPHERE 6:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
63	51	35	59	33	126	92
45	56	52	37	110	67	148
111	68	42	28	94	15	118
143	51	3	59	51	56	16
93	74	29	118	54	188	47
21	108	3	69	93	139	59
134	51	2	18	161	54	4
194	5	15	23	100	48	50
74	116	3	21	57	24	2
34	329	20	295	89	132	130
126	279	14	744	151	139	290
142	97	34	795	621	186	197
36	39	9	379	834	260	68
139	349	138	61	482	225	40
40	582	410	30	81	66	109
63	466	441	75	20	55	283
139	171	190	40	505	230	233
94	6	45	143	1624	423	121
54	131	18	755	1943	425	94
17	417	2	1756	1007	195	44
41	523	11	1905	147	33	85
82	405	371	1006	27	41	76
83	172	1039	232	624	146	69
86	9	1177	22	1710	382	36
47	37	604	64	1966	404	41
87	137	123	111	1013	166	60
61	250	28	711	228	31	26
47	228	60	1777	26	92	74
56	183	407	1945	21	223	13
2	84	1077	1034	99	194	34
13	44	1210	235	354	23	4
44	3	631	2	794	23	65
12	37	134	232	906	75	62
24	36	6	797	489	83	23
27	11	32	862	173	45	46
113	32	56	395	48	79	118
169	51	216	60	70	69	
227	53	478	170	108	115	

69

Table 27 Contd…

43	21	483	100	72	28	
168	84	222	96	105	57	
184	127	70	8	155	51	
58	20	57	26	121	167	
92	19	10	55	58	150	
117	17	69	142	63	48	
	93	56	61	37	51	
		1	11	1	73	
		47	50	17	32	
		26		12	15	
		1		75	23	
		52		91	121	
		2		66		
		22		98		
		38		12		
		55				
		23				
		41				
		36				
		14				
		32				
		64				
		32				
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
83	136	174	375	338	122	83
Max:	Max:	Max:	Max:	Max:	Max:	Max:
227	582	1210	1945	1966	425	290
SUMALL:						
64955						

Table 28. 10% Threshold value pixel count values of sphere 6

10 % Threshold:						
Slice 28:	Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:
227	329	410	295	621	260	290
	279	441	744	834	225	197
	349	371	795	482	230	283
	582	1039	379	505	423	233
	466	1177	755	1624	425	
	417	604	1756	1943	382	
	523	407	1905	1007	404	
	405	1077	1006	624	223	
	250	1210	232	1710		
	228	631	711	1966		
		216	1777	1013		
		478	1945	228		
		483	1034	354		
		222	235	794		
			232	906		
			797	489		
			862			
			395			
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
227	383	626	881	944	321	251
SUMALL:						
47348						

71

Table 29. 20% and 30% Threshold value pixel count values of sphere 6

20 % Threshold:				
Slice 29:	Slice 30:	Slice 31:	Slice 32:	Slice 33:
582	410	744	621	423
466	441	795	834	425
417	1039	755	482	404
523	1177	1756	505	
405	604	1905	1624	
	407	1006	1943	
	1077	711	1007	
	1210	1777	624	
	631	1945	1710	
	478	1034	1966	
	483	797	1013	
		862	794	
		395	906	
			489	
Mean:	Mean:	Mean:	Mean:	Mean:
478	723	1114	1037	417
SUMALL:				
40599				
30 % Threshold:				
Slice 30:	Slice 31:	Slice 32:		
1039	744	621		
1177	795	834		
604	755	1624		
1077	1756	1943		
1210	1905	1007		
631	1006	624		
	711	1710		
	1777	1966		
	1945	1013		
	1034	794		
	797	906		
	862			
Mean:	Mean:	Mean:		
956	1174	1186		
SUMALL:				
32865				

Table 30. 40% and 50% Threshold value pixel count values of sphere 6

40 % Threshold:		
Slice 30:	Slice 31:	Slice 32:
1039	795	834
1177	1756	1624
1077	1905	1943
1210	1006	1007
	1777	1710
	1945	1966
	1034	1013
	797	794
	862	906
Mean:	Mean:	Mean:
1126	1320	1311
SUMALL:		
28177		
50 % Threshold:		
Slice 30:	Slice 31:	Slice 32:
1039	1756	1624
1177	1905	1943
1077	1006	1007
1210	1777	1710
	1945	1966
	1034	1013
Mean:	Mean:	Mean:
1126	1570	1544
SUMALL:		
23189		

Now we know all of the pixel counts of all the spheres in each of the seven slices, and also their sum of all the pixel counts in all the transaxial slices, mean and maximum pixel counts of each single transaxial slice. Also we know their corresponding 10-50% sum of all the pixel counts in all the transaxial slices, mean and maximum values of one each single transaxial slice. A linear regression analysis is applied between the activity on x-axis and either the sum of pixel counts or mean or maximum pixel counts of one single transaxial slice on y-axis. The Various linear regression analysis graph drawn are.

a. Spheres Activity (vs.) Mean Pixel Values (Figure 15).

b. Spheres Activity (vs.) Maximum Pixel Values (Figure 16).

c. Spheres Activity (vs.) Sum Pixel Values (Figure 17).

d. Spheres Activity (vs.) 10% threshold Sum Pixel Values (Figure 18).

e. Spheres Activity (vs.) 20% threshold Sum Pixel Values (Figure 19).

f. Spheres Activity (vs.) 30% threshold Sum of Pixel Values (Figure 20).

g. Spheres Activity (vs.) 40% threshold Sum of Pixel Values (Figure 21).

h. Spheres Activity (vs.) 50% threshold Sum Pixel Values (Figure 22).

.

Real activity (μCi)	Pixel counts mean
0.7228	1430
0.6747	1490
0.5957	1480
0.4644	1389
0.6578	1503
0.5181	1570
Slope m :	20.55094589
Y-intercept, B :	1464.55469
Correlation Coefficienct r:	0.032589685

Table 31. Linear regression analysis of sphere activity(μCi) vs pixel counts mean
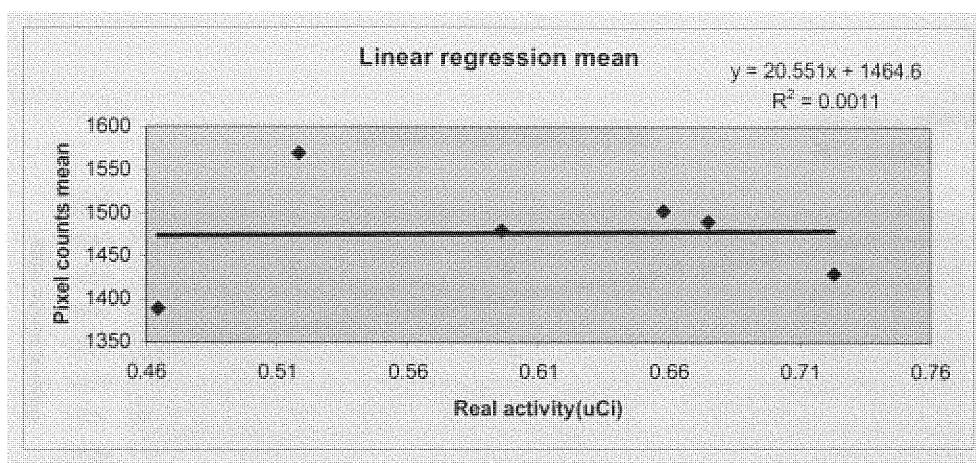
Figure 15. Linear regression analysis of sphere activity(μCi) vs pixel counts mean.

Real activity (µCi)	Pixel counts max
0.7228	1966
0.6747	1966
0.5957	1966
0.4644	1880
0.6578	1966
0.5181	1966
Slope m :	248.1599103
Y-intercept, B :	1801.385161
Correlation Coefficienct r:	0.699195983

Table 32. Linear regression analysis of sphere activity(µCi) vs pixel counts max
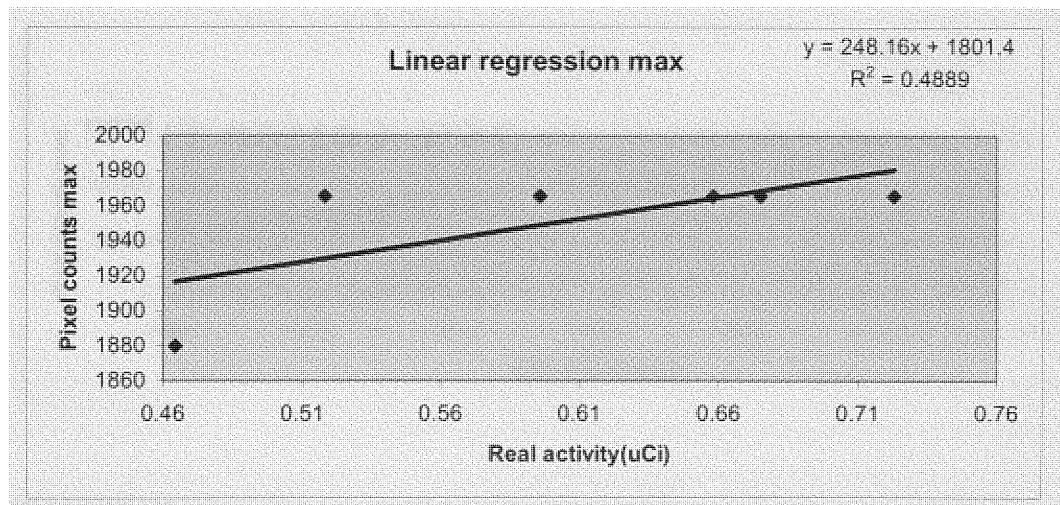
Figure 16. Linear regression analysis of sphere activity (µCi) vs pixel counts maximum

Real activity (µCi)	Pixel counts sum
0.7228	141831
0.6747	133580
0.5957	103914
0.4644	72850
0.6578	81119
0.5181	64955
Slope m :	267716.6451
Y-intercept, B :	-62416.57169
Correlation Coefficienct r:	0.820094669

Table 33. Linear regression analysis of sphere activity(µCi) vs pixel counts sum
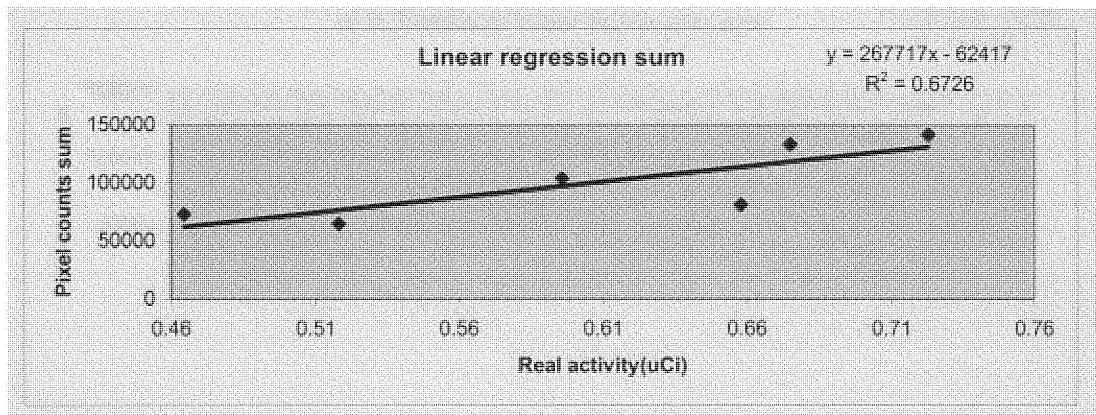
Figure 17. Linear regression analysis of sphere activity (µCi) vs pixel counts sum

Real activity (µCi)	Pixel counts sum
0.7228	135116
0.6747	119111
0.5957	83391
0.4644	55414
0.6578	69291
0.5181	47348
Slope m :	304507.02
Y-intercept, B :	-99459.20953
Correlation Coefficienct r:	0.854208287

Table 34. 10% Threshold linear regression analysis of sphere activity (µCi) vs pixel counts sum
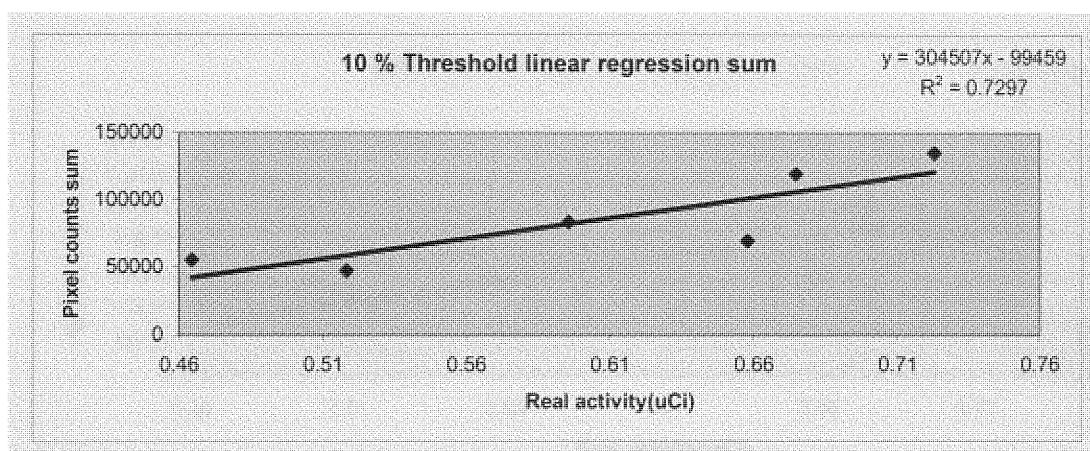


Figure 18. 10% Threshold linear regression analysis of sphere activity (µCi) vs pixel counts sum

Real activity (µCi)	Pixel counts sum
0.7228	114764
0.6747	95516
0.5957	64009
0.4644	41248
0.6578	58591
0.5181	40599
Slope m :	267858.0144
Y-intercept, B :	-93089.18255
Correlation Coefficienct r:	0.882503935

Table 35. 20% Threshold linear regression analysis of sphere activity (µCi) vs pixel counts sum
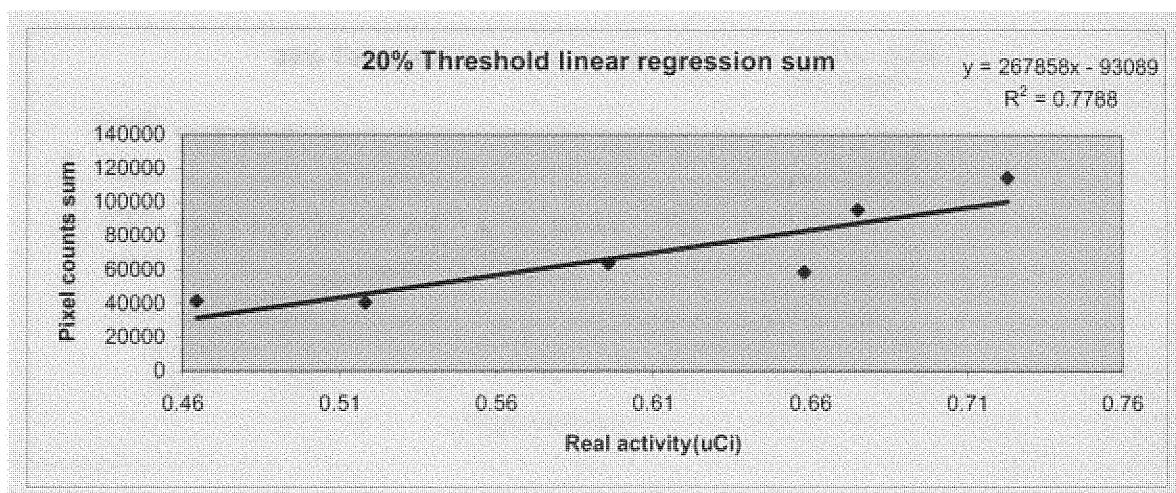
Figure 19. 20 % Threshold linear regression analysis pf sphere activity(µCi) vs pixel counts sum.

Real activity (µCi)	Pixel counts sum
0.7228	93665
0.6747	73228
0.5957	43828
0.4644	20949
0.6578	42000
0.5181	32865
Slope m :	244598.5274
Y-intercept, B :	-97035.62489
Correlation Coefficienct r:	0.892032717

Table 36. 30% Threshold linear regression analysis of sphere activity (µCi) vs pixel counts sum
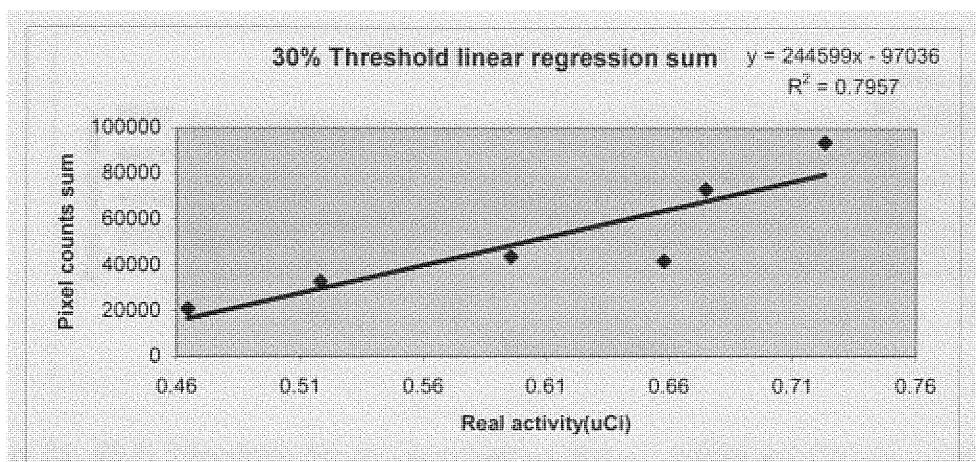
Figure 20. 30 % Threshold linear regression analysis of sphere activity (µCi) vs pixel counts sum

Real activity (µCi)	Pixel counts sum
0.7228	67959
0.6747	62614
0.5957	32870
0.4644	20949
0.6578	42000
0.5181	28177
Slope m :	178615.7352
Y-intercept, B :	-65738.54566
Correlation Coefficienct r:	0.927305017

Table 37. 40% Threshold linear regression analysis of sphere activity (µCi) vs pixel counts sum
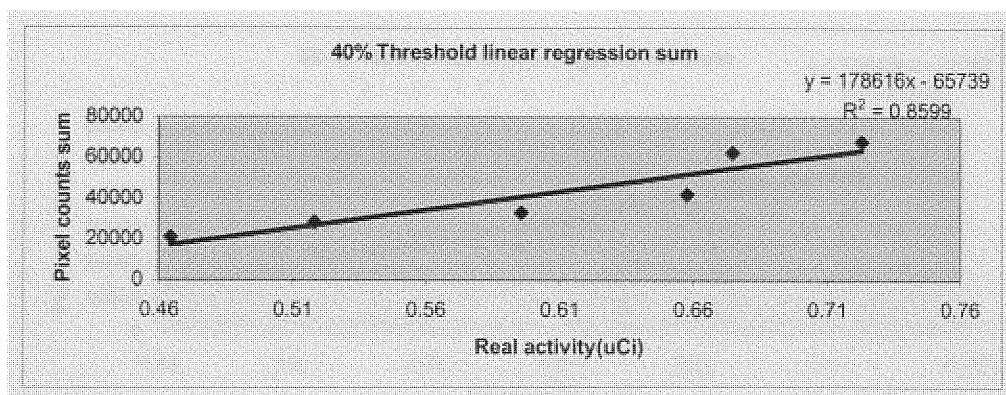
Figure 21. 40 % Threshold linear regression analysis of sphere activity (µCi) vs pixel counts sum

Real activity (µCi)	Pixel counts sum
0.7228	45815
0.6747	48723
0.5957	24987
0.4644	13859
0.6578	36926
0.5181	23189
Slope m :	131496.2111
Y-intercept, B :	-47382.08048
Correlation Coefficienct r:	0.943527183

Table 38. 50% Threshold linear regression analysis of sphere activity (µCi) vs pixel counts sum
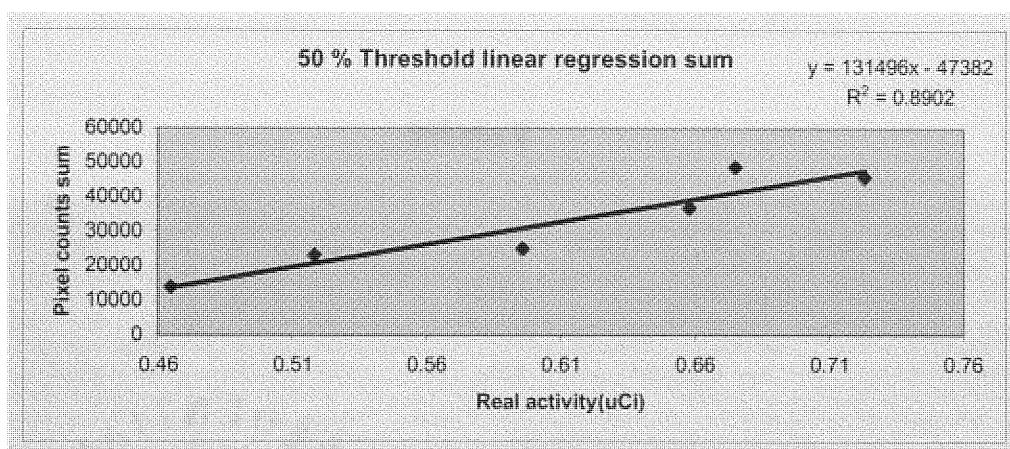
Figure 22. 50 % Threshold linear regression analysis of sphere activity (µCi) vs pixel counts sum
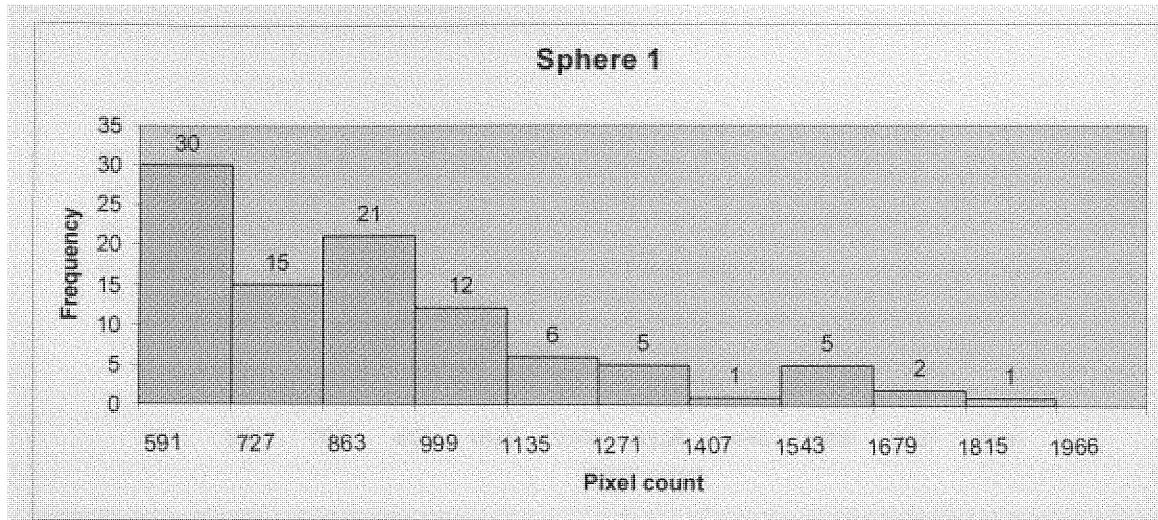
Figure 23. Histogram of pixel counts vs frequency of sphere 1
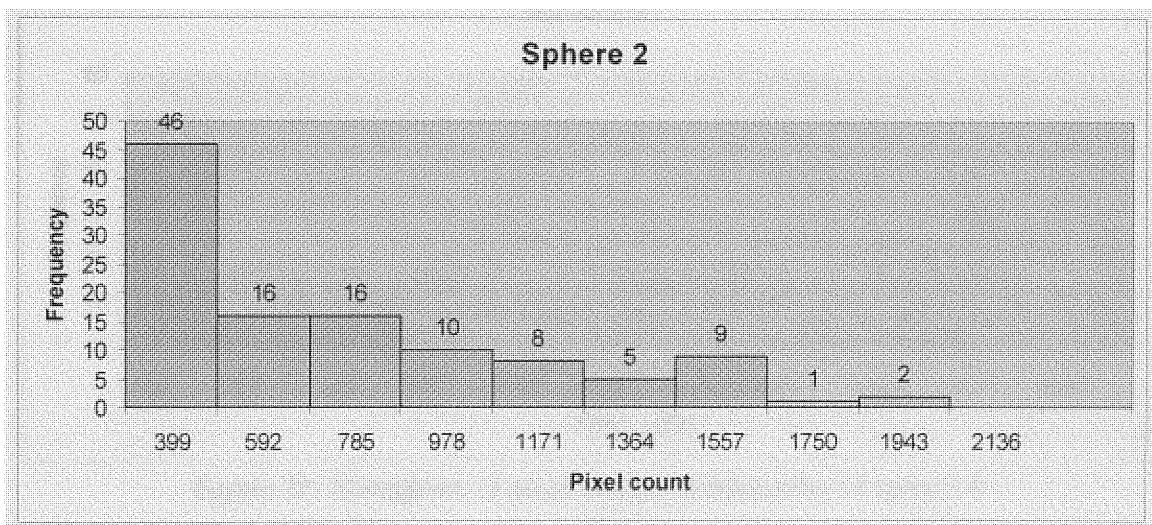
Figure 24. Histogram of pixel counts vs frequency of sphere 2
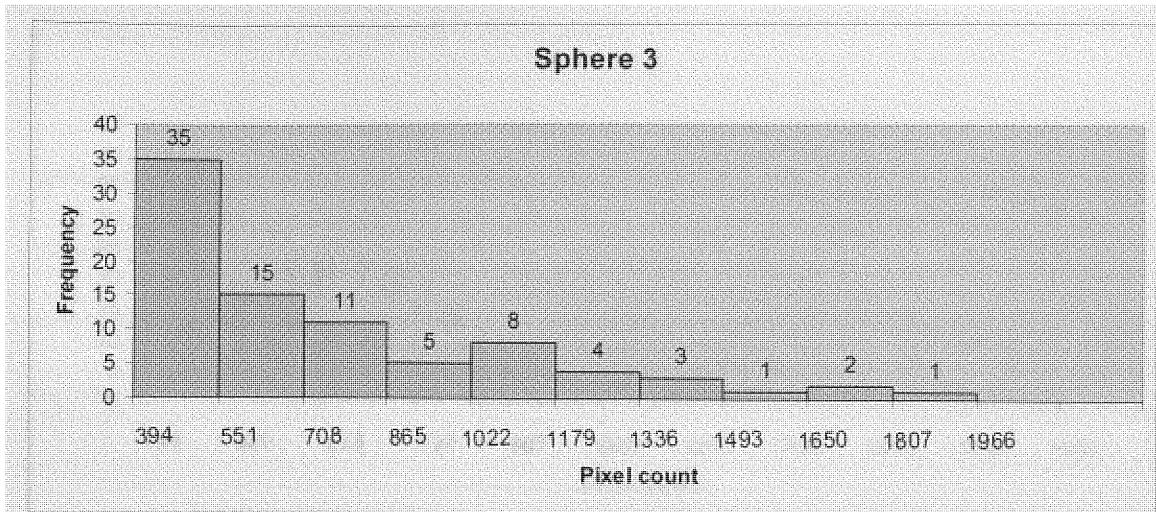
Figure 25. Histogram of pixel counts vs frequency of sphere 3
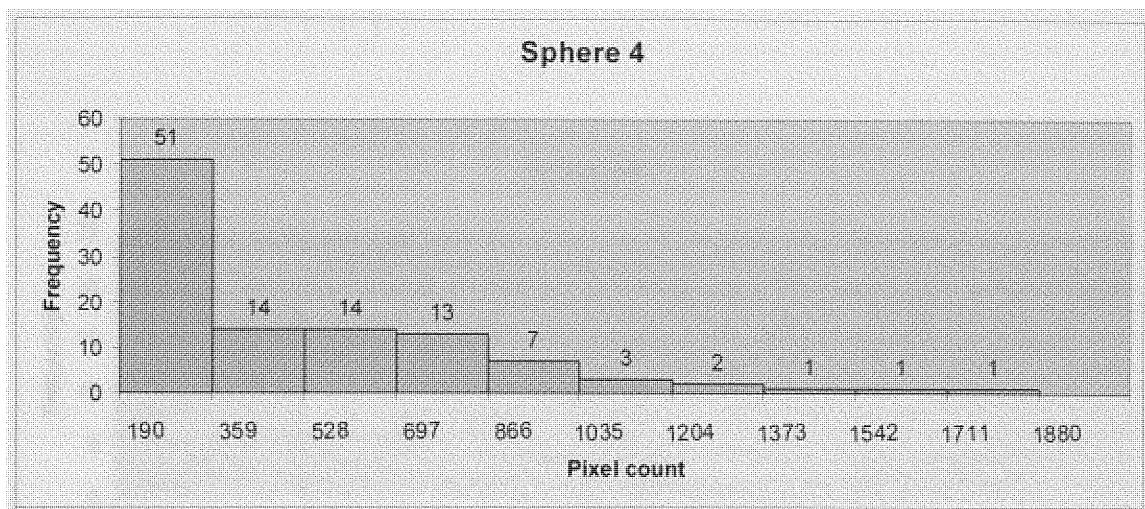
Figure 26. Histogram of pixel counts vs frequency of sphere 4
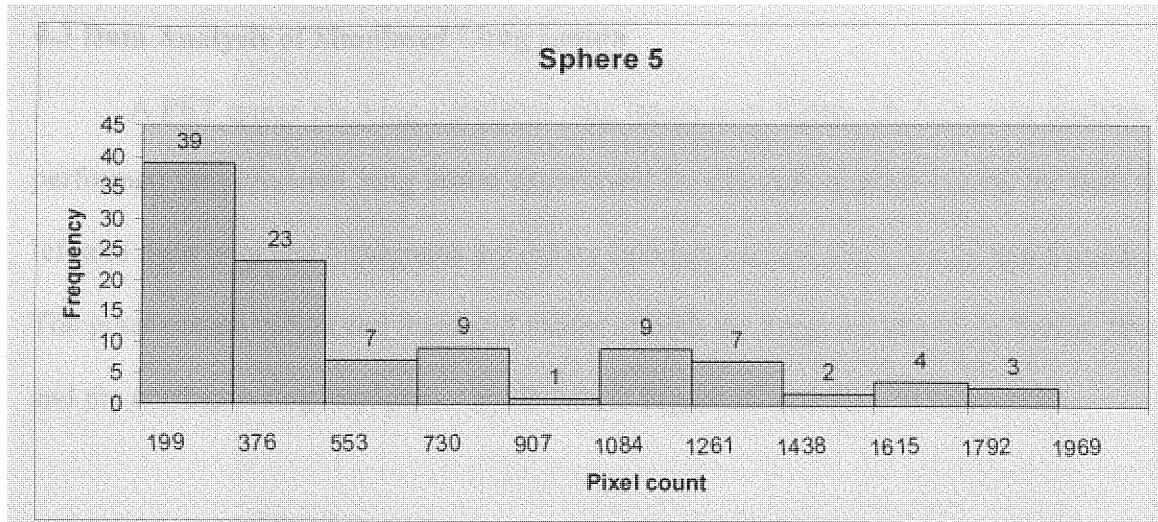
Figure 27. Histogram of pixel counts vs frequency of sphere 5
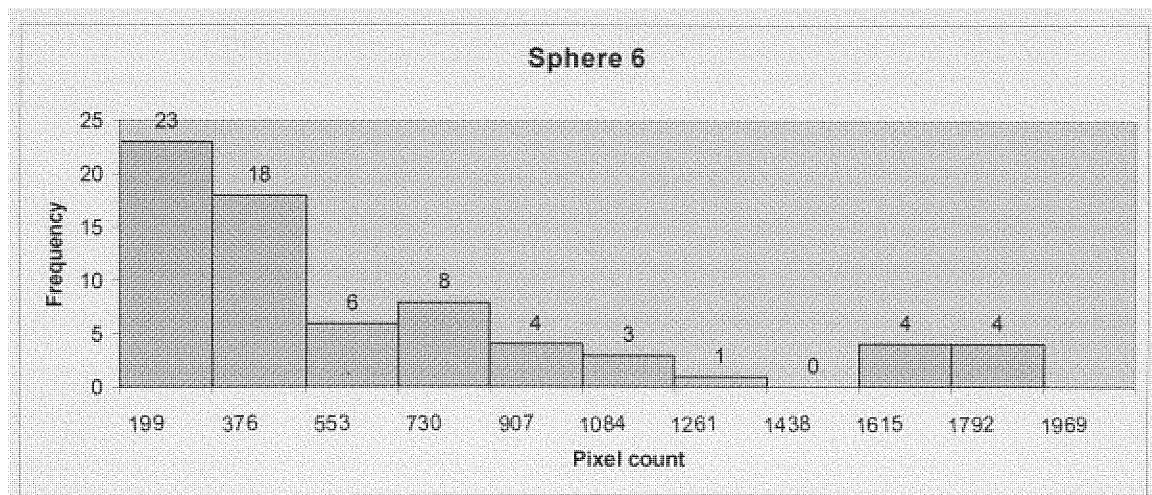
Figure 28. Histogram of pixel counts vs frequency of sphere 6

4.3 Data Analysis of Simulated Lung Lesion

A PET image showing simulated lung lesion is in Figure 29. Now data analysis is performed for simulated lung lesion experiment by calculating pixel counts for both the left and right lung and histograms are drawn between various pixel counts and frequency. The pixel counts values and threshold values of both the lungs are shown in Tables 39-43 and the corresponding histograms are shown in Figure 30 and 31.
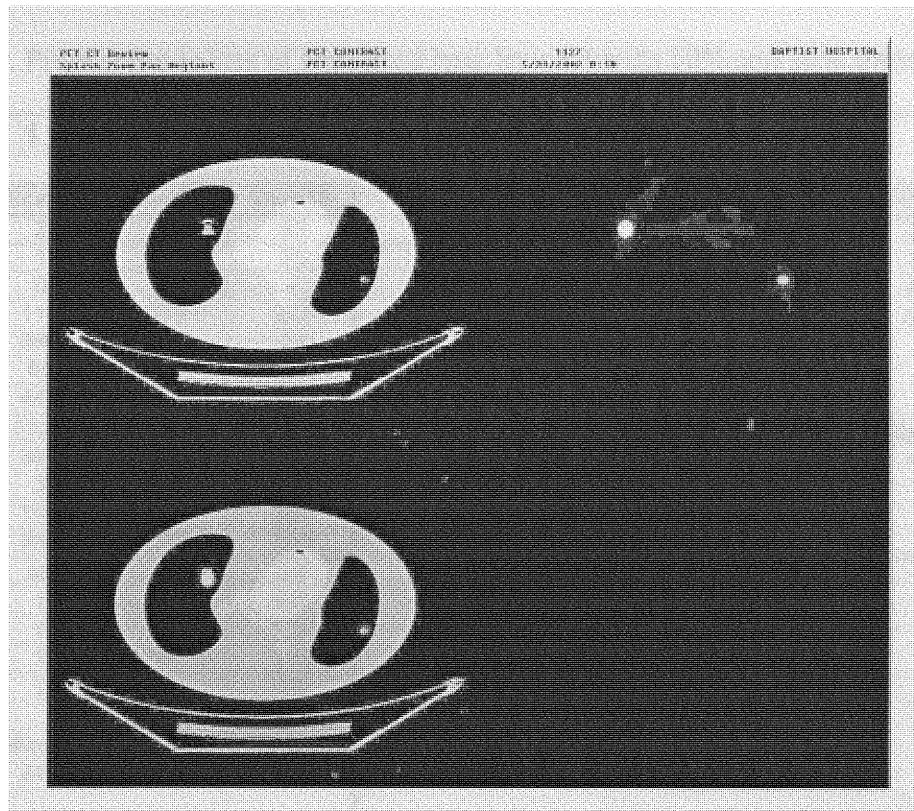
Figure 29. Right-TOP: CT image of simulated lung lesions in which we see lung lesion 1 on Right lung and lung lesion 2 on left lung; Left-TOP: PET image of lung lesions 1 and 2; Left Bottom: Combined PET-CT image of lung lesions 1 and 2

Table 39. Pixel count values of simulated lung lesion 1 with 10% and 20% threshold

LUNG 1 TABLE:

Slice 35:	Slice 36:	Slice37:	Slice38:
131	2	63	93
325	1	16	25
76	0	2	4
623	116	379	335
1638	398	447	72
382	95	95	6
475	449	693	429
1544	1638	1638	127
367	392	382	19
4	3	31	534
85	108	164	388
297	394	399	269
71	95	95	58
1	1	5	148
1	1	5	164
1	1	4	119
1	1	3	29
Mean:	**Mean:**	**Mean:**	**Mean:**
354	246	260	166
Max:	**Max:**	**Max:**	**Max:**
1638	1638	1638	534
Sum ALL:			
16953			

20% Threshold:

Slice 35:	Slice 36:	Slice37:	Slice38:
367	392	379	335
382	394	382	388
475	398	399	429
623	449	447	534
1544	1638	693	
1638		1638	
Mean:	**Mean:**	**Mean:**	**Mean:**
838	654	657	421
Sum ALL:			
13924			

10 % Threshold:

Slice 35:	Slice 36:	Slice37:	Slice38:
297	392	164	164
325	394	379	269
367	398	382	335
382	449	399	388
475	1638	447	429
623		693	534
1544		1638	
1638			
Mean:	**Mean:**	**Mean:**	**Mean:**
706	654	586	353
Sum All:			
15143			

Table 40. Pixel count values of simulated lung lesion 2.

LUNG 2 TABLE:					
Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:	Slice 35:
293	448	735	746	907	376
967	1051	1202	1183	1712	720
721	760	553	506	722	411
1935	2296	2907	2852	3094	1468
4949	4890	4873	4402	5266	2304
1263	1327	1334	1218	1536	682
4499	5001	5107	4571	4583	2593
7156	7156	7156	7156	7156	3508
1719	1795	1945	2037	1884	871
2405	3991	4392	3683	4339	2809
2246	3727	5466	5464	5616	3075
1025	1260	1438	1489	1318	659
266	321	191	151	270	232
363	943	1170	943	1238	909
201	1100	1887	1602	1347	812
204	391	512	425	305	163
48	45	15	12	52	38
5	69	101	71	130	103
5	153	241	157	88	62
9	47	69	41	20	12
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
1514	1839	2065	1935	2079	1090
Max:	Max:	Max:	Max:	Max:	Max:
7156	7156	7156	7156	7156	3508
Sum ALL:					
210441					

Table 41. 10% Threshold pixel count values of simulated lung lesion 2.

10 % Threshold:					
Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:	Slice 35:
967	1051	735	746	907	720
721	760	1202	1183	1712	1468
1935	2296	2907	2852	722	2304
4949	4890	4873	4402	3094	2593
1263	1327	1334	1218	5266	3508
4499	5001	5107	4571	1536	871
7156	7156	7156	7156	4583	2809
1719	1795	1945	2037	7156	3075
2405	3991	4392	3683	1884	909
2246	3727	5466	5464	4339	812
1025	1260	1438	1489	5616	
	943	1170	943	1318	
	1100	1887	1602	1238	
				1347	
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
2626	2715	3047	2873	2908	1907
SUMALL:					
200927					

Table 42. 20% and 30% Threshold pixel count values of simulated lung lesion 2.

20 % Threshold:					
Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:	Slice 35:
1935	2296	2907	2852	1712	1468
4949	4890	4873	4402	3094	2304
4499	5001	5107	4571	5266	2593
7156	7156	7156	7156	1536	3508
2405	1795	1945	2037	4583	2809
2246	3991	4392	3683	7156	3075
	3727	5466	5464	1884	
		1438	1489	4339	
		1887	1602	5616	
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
3865	4122	3908	3695	3909	2626
Sum all:					
171418					
30 % Threshold:					
Slice 30:	Slice 31:	Slice 32:	Slice 33:	Slice 34:	Slice 35:
4949	2296	2907	2852	3094	2304
4499	4890	4873	4402	5266	2593
7156	5001	5107	4571	4583	3508
2405	7156	7156	7156	7156	2809
2246	3991	4392	2037	4339	3075
	3727	5466	3683	5616	
		1887	5464		
Mean:	Mean:	Mean:	Mean:	Mean:	Mean:
4251	4510	4541	4309	5009	2858
Sum All:					
154615					

Table 43. 40% and 50% Threshold pixel count values of simulated lung lesion 2.

40 % Threshold:					
Slice 30:	**Slice 31:**	**Slice 32:**	**Slice 33:**	**Slice 34:**	**Slice 35:**
4949	4890	2907	4402	3094	3508
4499	5001	4873	4571	5266	3075
7156	7156	5107	7156	4583	
	3991	7156	3683	7156	
	3727	4392	5464	4339	
		5466		5616	
Mean:	**Mean:**	**Mean:**	**Mean:**	**Mean:**	**Mean:**
5535	4953	4984	5055	5009	3292
Sum all:					
133185					

50 % Threshold:					
Slice 30:	**Slice 31:**	**Slice 32:**	**Slice 33:**	**Slice 34:**	
4949	4890	4873	4402	5266	
4499	5001	5107	4571	4583	
7156	7156	7156	7156	7156	
	3991	4392	3683	4339	
	3727	5466	5464	5616	
Mean:	**Mean:**	**Mean:**	**Mean:**	**Mean:**	
5535	4953	5399	5055	5392	
Sum all:					
120601					
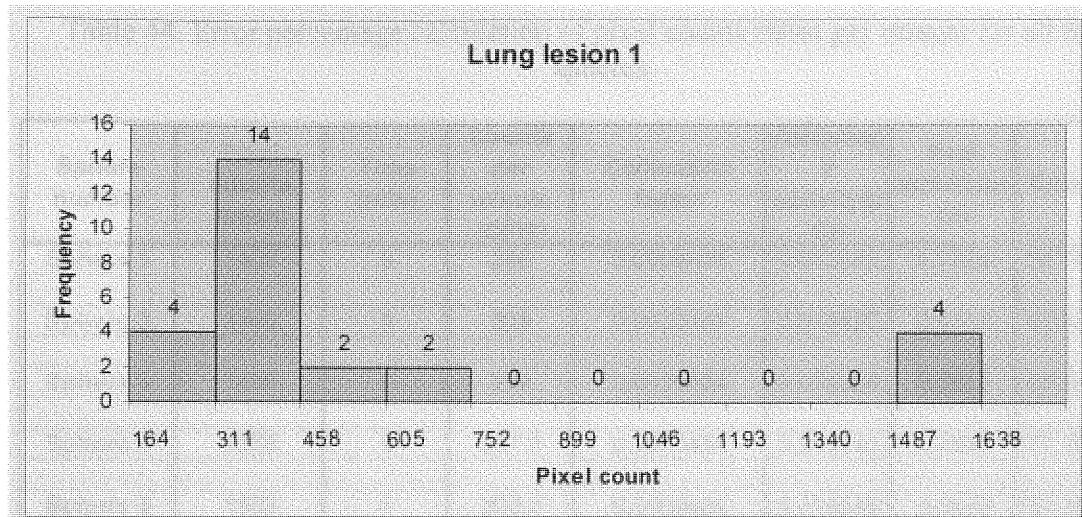
Figure 30. Histogram of pixel counts vs frequency of Lung 1
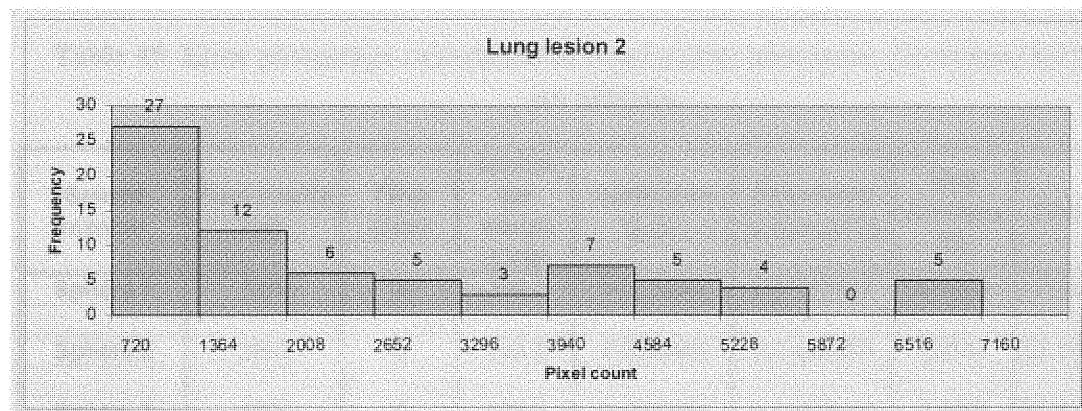
Figure 31. Histogram of pixel counts vs frequency of Lung 2

Table 44. Error percentage calculation of pixel counts mean per minute for all the six spheres.

Sphere number	Pixel counts mean:	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(μCi)	Real activity(μCi)	Error(%)
Sphere 1	1430	5	286	0.0000484	0.013842	0.7228	98.08489
Sphere 2	1490	5	298	0.0000484	0.014423	0.6747	97.86228
Sphere 3	1480	5	296	0.0000484	0.014326	0.5957	97.59503
Sphere 4	1389	5	277.8	0.0000484	0.013446	0.4644	97.10475
Sphere 5	1503	5	300.6	0.0000484	0.014549	0.6578	97.78823
Sphere 6	1570	5	314	0.0000484	0.015198	0.5181	97.06667

Table 45. Error percentage calculation of pixel counts maximum per minute for all the six spheres.

Sphere number	Pixel counts max:	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(μCi)	Real activity(μCi)	Error(%)
Sphere 1	1966	5	393.2	0.0000484	0.01903088	0.7228	97.367061
Sphere 2	1966	5	393.2	0.0000484	0.01903088	0.6747	97.179357
Sphere 3	1966	5	393.2	0.0000484	0.01903088	0.5957	96.805291
Sphere 4	1880	5	376	0.0000484	0.0181984	0.4644	96.081309
Sphere 5	1966	5	393.2	0.0000484	0.01903088	0.6578	97.10689
Sphere 6	1966	5	393.2	0.0000484	0.01903088	0.5181	96.326794

Table 46. Error percentage calculation of pixel counts sum per minute for all the six spheres.

Sphere number	Pixel counts sum:	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(µCi)	Real activity(µCi)	Error(%)
Sphere 1	141831	5	28366.2	0.0000484	1.37292408	0.7228	-89.9452241
Sphere 2	133580	5	26716	0.0000484	1.2930544	0.6747	-91.6487921
Sphere 3	103914	5	20782.8	0.0000484	1.00588752	0.5957	-68.8580695
Sphere 4	72850	5	14570	0.0000484	0.705188	0.4644	-51.8492679
Sphere 5	81119	5	16223.8	0.0000484	0.78523192	0.6578	-19.3724415
Sphere 6	64955	5	12991	0.0000484	0.6287644	0.5181	-21.3596603

Table 47. Error percentage calculation of 10% threshold pixel counts sum per minute for all the six spheres.

10 % Threshold:							
Sphere number	Pixel counts sum:	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(µCi)	Real activity(µCi)	Error(%)
Sphere 1	135116	5	27023.2	0.0000484	1.30792288	0.7228	-80.9522524
Sphere 2	119111	5	23822.2	0.0000484	1.15299448	0.6747	-70.8899481
Sphere 3	83391	5	16678.2	0.0000484	0.80722488	0.5957	-35.5086251
Sphere 4	55414	5	11082.8	0.0000484	0.53640752	0.4644	-15.5054953
Sphere 5	69291	5	13858.2	0.0000484	0.67073688	0.6578	-1.96668896
Sphere 6	47348	5	9469.6	0.0000484	0.45832864	0.5181	11.5366454

Table 48. Error percentage calculation of 20% threshold pixel counts sum per minute for all the six spheres.

Sphere number	Pixel counts sum:	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(µCi)	Real activity(µCi)	Error(%)
20 % Threshold:							
Sphere 1	114764	5	22952.8	0.0000484	1.11091552	0.7228	-53.6961151
Sphere 2	95516	5	19103.2	0.0000484	0.92459488	0.6747	-37.037925
Sphere 3	64009	5	12801.8	0.0000484	0.61960712	0.5957	-4.01328185
Sphere 4	41248	5	8249.6	0.0000484	0.39928064	0.4644	14.0222567
Sphere 5	58591	5	11718.2	0.0000484	0.56716088	0.6578	13.7791304
Sphere 6	40599	5	8119.8	0.0000484	0.39299832	0.5181	24.146242

Table 49. Error percentage calculation of 30% threshold pixel counts sum per minute for all the six spheres.

Sphere number	Pixel counts sum:	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(µCi)	Real activity(µCi)	Error(%)
30 % Threshold:							
Sphere 1	93665	5	18733	0.0000484	0.9066772	0.7228	-25.4395683
Sphere 2	73228	5	14645.6	0.0000484	0.70884704	0.6747	-5.06107011
Sphere 3	43828	5	8765.6	0.0000484	0.42425504	0.5957	28.7804197
Sphere 4	33763	5	6752.6	0.0000484	0.32682584	0.4644	29.6240655
Sphere 5	47534	5	9506.8	0.0000484	0.46012912	0.6578	30.050301
Sphere 6	32865	5	6573	0.0000484	0.3181332	0.5181	38.5961783

Table 50. Error percentage calculation of 40% threshold pixel counts sum per minute for all the six spheres.

40 % Threshold:							
Sphere number	Pixel counts sum:	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(µCi)	Real activity(µCi)	Error(%)
Sphere 1	67959	5	13591.8	0.0000484	0.65784312	0.7228	8.98684007
Sphere 2	62614	5	12522.8	0.0000484	0.60610352	0.6747	10.1669601
Sphere 3	32870	5	6574	0.0000484	0.3181816	0.5957	46.5869397
Sphere 4	20949	5	4189.8	0.0000484	0.20278632	0.4644	56.3336951
Sphere 5	42000	5	8400	0.0000484	0.40656	0.6578	38.1939799
Sphere 6	28177	5	5635.4	0.0000484	0.27275336	0.5181	47.3550743

Table 51. Error percentage calculation of 50% threshold pixel counts sum per minute for all the six spheres.

50 % Threshold:							
Sphere number	Pixel counts sum:	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(µCi)	Real activity(µCi)	Error(%)
Sphere 1	45815	5	9163	0.0000484	0.4434892	0.7228	38.6428888
Sphere 2	48723	5	9744.6	0.0000484	0.47163864	0.6747	30.0965407
Sphere 3	24987	5	4997.4	0.0000484	0.24187416	0.5957	59.3966493
Sphere 4	13859	5	2771.8	0.0000484	0.13415512	0.4644	71.1121619
Sphere 5	36926	5	7385.2	0.0000484	0.35744368	0.6578	45.6607358
Sphere 6	23189	5	4637.8	0.0000484	0.22446952	0.5181	56.6744798

Table 52. Error percentage calculation of pixel counts mean per minute for simulated lung lesions 1 and 2.

Lung	Pixel counts mean	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(μCi)	Real activity(μCi)	Error(%)
Lung 1	838	7	119.7143	0.0000484	0.005794171	0.09801	94.08818
Lung 2	5535	7	790.7143	0.0000484	0.038270571	0.8478	95.4859

Table 53. Error percentage calculation of pixel counts maximum per minute for simulated lung lesions 1 and 2.

Lung	Pixel counts max	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(μCi)	Real activity(μCi)	Error(%)
Lung 1	1638	7	234	0.0000484	0.0113256	0.09801	88.44444
Lung 2	7156	7	1022.286	0.0000484	0.049478629	0.8478	94.16388

Table 54. Error percentage calculation of pixel counts sum per minute and threshold pixel counts sum per minute for simulated lung lesions 1 and 2.

Lung	Pixel counts sum	Time (min)	Counts per minute (CPM)	Calibration factor	Calculated activity(μCi)	Real activity(μCi)	Error(%)
Lung 1	16949	7	2421.286	0.0000484	0.117190229	0.09801	-19.5697
10 % Threshold Lung 1	15143	7	2163.286	0.0000484	0.104703029	0.09801	-6.82892
20 % Threshold Lung 1	13924	7	1989.143	0.0000484	0.096274514	0.09801	1.770723
Lung 2	210441	7	30063	0.0000484	1.4550492	0.8478	-71.6265
10 % Threshold Lung 2	200927	7	28703.86	0.0000484	1.389266686	0.8478	-63.8673
20 % Threshold Lung 2	171418	7	24488.29	0.0000484	1.185233029	0.8478	-39.801
30 % Threshold Lung 2	154615	7	22087.86	0.0000484	1.069052286	0.8478	-26.0972
40 % Threshold Lung 2	133185	7	19026.43	0.0000484	0.920879143	0.8478	-8.61986
50 % Threshold Lung 2	120601	7	17228.71	0.0000484	0.833869771	0.8478	1.643103

Table 55. t-test analysis of linear regression graphs

Linear regression analysis	Regression equation	Regression coefficient, r2	Significance
Sphere activity (vs) Mean pixel count	y = 20.551x + 1464.6	0.0011	NS
Sphere activity (vs) Maximum pixel count	y = 248.16x + 1801.4	0.4889	NS
Sphere activity (vs) Sum of pixel count	y = 267717x - 62417	0.6726	P<0.025
10% Sphere activity (vs) Sum of pixel count	y = 304507x - 99459	0.7297	P<0.02
20% Sphere activity (vs) Sum of pixel count	y = 267858x - 93089	0.7744	P<0.02
30% Sphere activity (vs) Sum of pixel count	y = 244599x - 97036	0.7921	P<0.02
40% Sphere activity (vs) Sum of pixel count	y = 178616x - 65739	0.8464	P<0.01
50% Sphere activity (vs) Sum of pixel count	y = 131496x - 47382	0.8836	P<0.01

Significance was determined for p<0.05.

Table 56. Histogram analysis of simulated tumors and simulated lung lesion.

Histogram Analysis:	Mean (μCi)	Kurtosis	Median (μCi)	Skewness
Sphere 1	0.01	1.87	0.01	0.68
Sphere 2	0.01	1.79	0.01	0.66
Sphere 3	0.01	2.6	0.01	0.96
Sphere 4	0.01	1.3	0.01	4.28
Sphere 5	0.01	0.81	0.01	0.36
Sphere 6	0.01	1.06	0.01	0.48
Lung 1	0.01	1	0.01	0.53
Lung2	0.03	0.85	0.01	0.45

5. DISCUSSION

Quantitation of 18-FDG uptake is essential in oncologic PET studies in assessing tumor response to radiotherapy or chemotherapy. The method we propose for the accuracy of quantitation of 18-FDG uptake with simulated tumors correlates pixel count values with the activity uptake by the tumor. We did three experiments, namely PET calibration, experiment with simulated tumors and experiment with simulated lung lesions. In the PET calibration method we found out how many counts per pixel per minute are equal to one micro-curie of 18-FDG. In the experiment with simulated tumors and simulated lung lesion we have measured the activity for all the spheres in all the slices for images with attenuation correction.

Our method of choosing a ROI and calculating pixel counts differs from the traditional quantitative uptake method in which the activity of only one 2D transaxial slice is used to calculate either the mean or maximum pixel count to quantify the activity uptake in the tumor. In this study, from a stack of transaxial slices we calculate all the pixel counts that are distributed in a chosen ROI over a simulated lesion. The ROI chosen for one transaxial slice will be same for all the transaxial slices. The mean, maximum of one single transaxial slice and sum of all the pixel counts in all the transaxial slices were also calculated. We applied threshold values like 10-50% of maximum pixel counts to eliminate those pixel counts that really don't correspond to the activity and hence to increase the accuracy of activity quantitation.

From all the correlation coefficients of sphere activity vs pixel counts we can see that the sum of all pixel counts has the highest significance when compared to that of

mean pixel count and maximum pixel count. Thus we can say that there is a linear relation between the sum of all pixel counts and the sphere activity in simulated tumors. One can calculate the activity of the simulated tumor if we know the sum of all pixel counts. It is also proved that the sum of all the pixel count is a better index to determine the tumor's activity rather than mean pixel count or maximum pixel count. Thus sum of all pixel counts is needed to get good accuracy in PET quantitation.

We are able to find the activity of a tumor if we know the sum of the entire pixel counts per minute of all the transaxial slices and multiply it by the calibration factor. The activity is calculated for both simulated tumors and simulated lung lesions for the sum of pixel count values, threshold pixel count sum and also for the mean and maximum pixel counts of one single transaxial slice and compared with real activity. The sum activity is closer to that of the real activity whereas the activity calculated from the mean and maximum pixel counts is lower compared to that of the real activity.

The error in calculation of activity is shown in terms of percentage. The error percentage is high for the mean and maximum pixel count, greater than 96% in all the six spheres. Also, the error value for simulated lung lesions for the mean and maximum pixel count is greater than 88%.

From the error percentage values it is understood that the better accuracy for activity for spheres of volume 15 ml and 8 ml occurs using 30% to 40% threshold pixel counts sum. For sphere of volume 4 ml better accuracy occur using 20% to 30% threshold pixel counts sum. For sphere of volume 2 ml and 1 ml better accuracy occur using 10% to 20% threshold pixel counts sum. For sphere of volume 0.5 ml better accuracy occurred for 10% threshold pixel counts sum. Thus it is understood that

applying threshold values are important in activity quantitation for better accuracy of uptake.

In the error percentage calculation for the sphere images with attenuation correction, the percentage of error at 30-40% threshold for spheres of smaller volume are higher compared to that of spheres of larger size. This may be due to the partial volume effect in which the activity of the adjacent voxel may influence the activity of voxels of interest in smaller volume spheres.

In the sum of pixel error percentage calculations the better accuracy in terms of error percentage of attenuation corrected spheres1-6 are 9% at 40 % threshold, -5% at 30% threshold, -4% at 20% threshold, 14% at 20%threshold, -2% at 10% threshold and 11% at 10% threshold respectively.

Better accuracy of activity for simulated lung lesions 1 and 2 were obtained at 10% and 50% thresholds respectively. The error percentage was of 1.7% and 1.6% respectively. Hence applying threshold values are important for calculating the uptake of lung lesions.

Histograms were drawn between pixel count and frequency for images from simulated tumor experiment and also for the lung lesion experiment. Histogram is drawn for each sphere by taking 10 intervals of pixel count and finding the frequency or number of pixel count in each interval. This is done for all the six spheres, with and without attenuation correction and also for two simulated lung lesion. The mean, median, skewness and kurtosis were found for all the 14 histograms.

Skewness is a measure of symmetry, or more precisely, the lack of symmetry. A distribution, or data set, is symmetric if it looks the same to the left and right of the center

point. A normal distribution is symmetric about its mean, which indicates that there is an equal probability of an outcome above or below the mean, or average. The skewness for a Gaussian normal distribution is zero, and any symmetric data should have skewness near zero. The skewness is greater than 0 in all the histograms thus showing positive skewness and the distribution lean toward left and the right end of the distribution is trailing off. Thus we can say the distribution does not follow Gaussian normal distribution and the data here are skewed to the right. Thus the distribution of pixel count value does not depend on the mean pixel count.

Kurtosis is a measure of whether the data are peaked or flat relative to a normal distribution. That is, data sets with high kurtosis tend to have a distinct peak near the mean, decline rather rapidly, and have heavy tails. Data sets with low kurtosis tend to have a flat top near the mean rather than a sharp peak. From table 44 we have seen that we have positive kurtosis values for all the spheres and have a flat distribution. The kurtosis for a Gaussian normal distribution is 0 and we have values between 0.81-2.6 which are not close to 0. Thus we can predict from the kurtosis data that distribution of activity does not follow Gaussian normal distribution. The distribution of pixel count value does not depend on the mean pixel count and hence considering the mean pixel counts for quantitation of uptake may lead to false results and hence the distribution of entire pixel counts is necessary for better understanding of the uptake.

The errors percentages obtained in this experiment are unacceptable for clinical studies. This demonstrates that the performance of instrument is not proper for quantitative studies. The software initially used in this research has been changed by the manufacturer with the new improved version in which the bugs have been eliminated.

The new experiment following the same methodology used in this study will be undertaken in order to validate the new software. In other words the results of this research are not satisfactory for clinical studies but the methodology are same for further studies.

6. CONCLUSIONS

We have used a new procedure to calculate all the pixel counts in the chosen ROI in all the 3D transaxial slices and their corresponding sum of all transaxial slices, mean and maximum of each transaxial slices. We also applied threshold values and obtain their corresponding sum pixel counts of all transaxial slices, mean and maximum pixel counts of individual transaxial slice.

The results demonstrated that we can calculate the activity of the tumor if we know the sum of all pixel counts with applied threshold and multiplying them by the calibration factor. The importance of applying threshold is that we can eliminate pixel counts that really don't contribute to the tumor activity and thus increase the accuracy of activity quantitation of the tumor.

The linear regression analysis and error percentage values show that the sum of the pixel counts of all the transaxial slices (3D) with applied threshold is better than mean and maximum pixel counts of one single transaxial slice (2D) to calculate the activity of a tumor. Better accuracy of error percentage of attenuation corrected spheres1-6 are 9% at 40 % threshold sum, -5% at 30% threshold sum, -4% at 20% threshold sum, 14% at 20% threshold sum, -2% at 10% threshold sum and 11% at 10% threshold sum respectively. The error percentage of mean and maximum of one single transaxial slice of all six spheres are greater than 95%.

From the correlation coefficient values of the sum of pixel counts, mean pixel counts and maximum pixel counts we have shown that the sum of the pixel counts varies linearly with the activity of tumor compared to mean pixel count and maximum pixel count values. The t-test analysis proved that the linear relation between sum of all the pixel counts and activity is significant.

Comparing the calculated activity with the known real activity proves that a threshold value around 20%-40% will yield better results for calculating activities in bigger tumors of volume 15 ml, 8 ml, and 4 ml.

Similarly, this method proves effective to calculate the activity of a lung lesion if we know the sum of the entire pixel counts and multiply by the calibration factor. A 40%-50% threshold value would be appropriate for the bigger lung lesion of volume 0.34 ml.

Skewness and kurtosis of histograms of pixel counts showed that the 3D distribution of pixel count does not follow Gaussian normal distribution. This indicates that the mean value is not a reliable parameter to correlate with the activity of tumor and the knowledge of entire pixel counts is essential to characterize the uptake.

The calculated activity of smaller spheres at 20%-40% threshold is not close to the real activity as compared to that of bigger spheres. This may be due to the partial volume

effect in which the activity of adjacent voxel may influence the activity of the voxels of interest.

Finally, experimental errors in quantitation of the activity of the sources and their dependence with threshold levels are unacceptable for clinical studies. This is probably due to bugs and limitations of the first software version that was installed in a very new instrument. A new software version has been installed and it will be validated by the same methodology.

REFERENCES

1. Anne M J Paans, Aren van Waarde, Philip H. Elsinga, , Antoon T. M. Willemsen and Willem Vaalburg 2002. Positron Emission Tomography: the conceptual idea using multidisciplinary approach. Methods Volume 27, Issue 3 , July 2002, Pages 195-207. PET Center, Groningen University Hospital, P.O. Box 30.001, 9700 RB, Groningen, The Netherlands.

2. Annemeike C. kole, Omgo E. Nieweg, Jan Pruim, Anne M.J. Paans, John Th. M Plukker, Harald J. Hoekstra, Heimen Schraffordt Koops and Willem Vaalburg. " Standardized Uptake Value and Quantification of Metabolism for Breast Cancer Imaging with FDG and L-[1-^{11}C] Tyrosine ". PET.J.Nucl Med 1997; 38:692-696.

3. C. Labbé, K. Thielemans, H. Zaidi C. Morell. An object-oriented library incorporating efficient projection/backprojection operators for volume reconstruction in 3D PET 1997. Division of Nuclear Medicine, Geneva University Hospital, CH-1211 Geneva 4, Switzerland.

4. C.S. Patlak, R.G. Blasberg and J.D. Fenstermacher, Graphical evaluation of blood-to-brain transfer constants from multiple-time uptake data. J. Cereb. Blood Flow Metab. 3 (1983), pp. 1–7.

5. Dahlbom M, Hoffman EJ, Hoh CK et al. Whole- body positron emission tomography: Part 1: Methods and performance characterstics. J.Nucl Med 1992; 33:1191-1199.

6. Degrado TR, Turkington TG, Williams JJ et al. Performance characterstics of whole-body PET scanner. J.Nucl Med 1994;35 1398-1406.

7. Digital imaging and communications in medicine (Dicom), 2000. National Electrical Manufacturers Association 1300 N. 17th Street, Rosslyn, Virginia 22209 USA.

8. Dominique Delbeke, D.Michael Rose, William Chapman, C.Wright Pinson. Optimal Interpretation of FDG PET in the Diagnosis, Staging and Management of Pancreatic Carcinoma. J Nucl Med 1999; 40: 1794-1791.

9. G. Friedlander, J.W. Kennedy, E.S. Macias and J.M. Miller , Nuclear Radiochemistry. (third ed.),, Wiley, New York (1981).

10. G. Stocklin and V.W. Pike, Editors, Radiopharmaceuticals for Positron Emisssion Tomography, Kluwer, Dordrecht/Norwell, MA (1993) ISBN 0-7923-2340-8 .

11. Graham M.M, Peterson L.M, Hayward R.M. Comparison of simplified quantitative analyses of FDG uptake. Journal of Nuclear Medicine and Biology 2000.

12. Hamberg LM, Hunter GJ, Alpert NM, Choi NC, Babich JW, Fischman AJ. The dose uptake ratio as an index of glucose metabolism: useful parameter or oversimplicification? J Nucl Med 1994; 35; 1308-1312.

13. Hani A. Nabi, MD, Phd and Jose M. Zubeldia, MD. Clinical Applications of 18-FDG in Oncology. J. Nucl Med 2002.

14. HE Johns and JR Cunningham 1983. The Physics of Radiology (fourth edition). Charles C Thomas, III. USA, ISBN 0-398-04669-7.

15. (Henry) Sung-Cheng Huang 2000. Anatomy of SUV. Division of Biophysics and Nuclear Medicine, Department of Molecular and Medical Pharmacology, UCLA School of Medicine, Los Angeles, California, USA. Nuclear Medicine and Biology Volume 27, Issue 7 , October 2000, Pages 643-646

16. J. Helus, Editor, Radionuclides Production vols. I & II, CRC Press, Boca Raton, FL (1983) ISBN 0-8493-6003-X & 0-8493-6004-8.

17. Jacobus A.K.Blokland Petar Trindev, Marcel P. M. Stokkel and Ernest K. J. Pauwels 2001. Positron emission tomography: a technical introduction for clinicans. Department of Radiology, Division of Nuclear Medicine, C4Q Leiden University Medical Centre, Albinusdreef 2, 2333 ZA, Leiden, The Netherlands Medical Faculty, Centre of Nuclear Medicine, Sofia, Bulgaria.

18. Joseph A. Thie, KarlnF.bHubner, Gary T. Smith. " The Diagnostic Utility of the Lognormal Behavior of PET Standardized Uptake Values in Tumors". J Nucl Med 2000; 41:1664-1672.

19. Keyes JW JR. SUV: standard uptake or silly useless value? J Nucl Med 1995; 36; 1836-1839

20. M.M. Ter Pogossian, Positron-emission tomography. Sci. Am. 243 (1980), pp. 171– 181.

21. Maguire, GQ., et al 1991: Graphics applied to medical image registration. IEEE Computer graphics and applications, (3): p 20-28.

22. Michael Schulte, Doris Brecht-Krauss, Berno Heymer, Albrecht Guhlmann, Erich Hartwig. "Grading of Tumors and Tumor like Lesions of Bone: Evaluation by FDG PET". J Nucl Med 2000; 41:1695-1701.
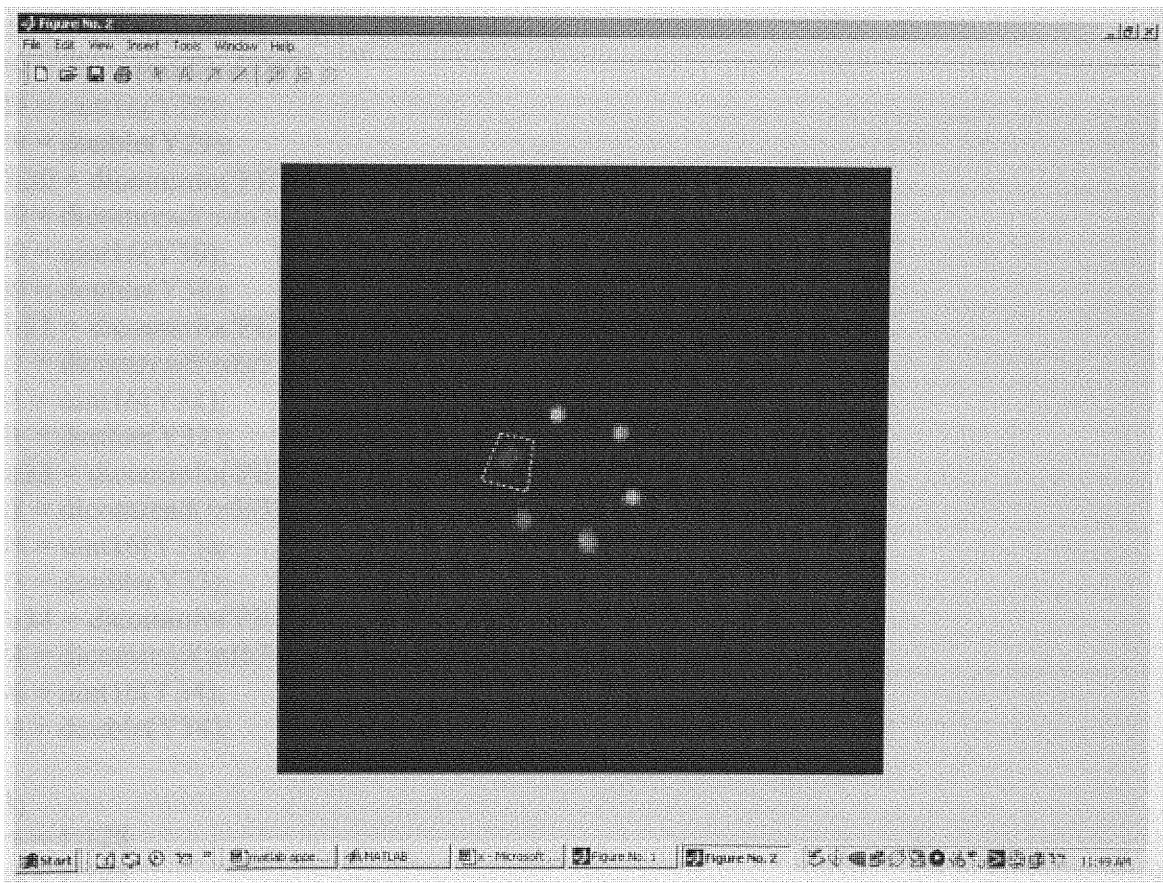
23. Phelps, M.E., Positron Emission Tomography. In: Mazziotta, J. and Gilman, S., Eds., Clinical Brain Imaging: Principles and Applications, 1992, F.A. Davis Company, pp71-107.

24. S. Aronow , Positron scanning. In: G.J. Hine, Editor, Instrumentation in nuclear medicine, Academic Press, New York (1967), pp. 461–483.

25. Tai, Y.C., et al. 1997: Utilization of 3-D elastic transformation in the registration of chest X-ray CT and Whole Body PET. IEEE Transactions on Nuclear Science., 44(4): p. 1606-1612.

26. Timothy G. Turkington, Ph.d. 2001. Introduction to PET Instrumentation. Journal of nuclear medicine technology, Mar, 29(1): 4-11.

27. U.Bepler, A.Dimitrakopoulou-strauss, G.Kontaxakis, L.G.Strauss. "Functional studies of ^{18} F-Deoxyglucose (FDG) in small cell lung tumors using positron emission tomography (PET)". Greek J.Nucl Med, Volume1 No 4, October-december 1998,1(4):231-239.

28. V.R. McCready. Milestones in nuclear medicine. Eur. J. Nucl. Med. 27 (2000), pp. S49–S79.

29. http://www.gemedicalsystems.com.

30. http://www.biodex.com.

31. http://www.eee.ntu.ac.uk

Appendix

The matlab program for drawing ROI for the sphere of interest and calculating the pixel counts of mean, maximum of each transaxial slice and sum of all the transaxial is shown here. The sequence of steps of the algorithm of the program is shown below.

1. The transaxial slice number 32 'PET_CT_AC2032.dcm' is read by dicomread command and assigned to variable Y32. imshow command displays the image.

2. ROIPOLY Select polygonal region of interest. BW(x,y) = roipoly(Y1) displays the image Y1 on the screen and allow us to draw a region of interest. BW is a binary image the same size as Y1 with 0's outside the region of interest and 1's inside.

3. The DICOM image of transaxial slices 28-34 are read and assigned to variables Y28-Y34.

4. Now we can find the pixel counts of the region of interest by a conditional statement which states that whenever the i,j position corresponds to ROI of image BW then it will calculate the pixel counts of the same i,j position for images Y28-Y34 and store in variables counts28 to counts 34.

5. The calculated counts are multiplied with a slope factor a constant which corresponds to the PET-CT scanner.

6. Now the original pixel counts are stored in variables pixelcounts28-pixelcounts34.

7. The zeros are eliminated by the command 'sparse' and new pixel counts are stored in variables Spixelcounts28-Spixelcounts34.

8. The SUM and MAXIMUM pixel counts of each slice are calculated by built-in functions sum and max.

9. The number of non-zero array elements in individual variables is found by using the command 'nnz'. The mean of the pixel counts is found by dividing the sum of the pixel counts by the total number of elements i.e., is the number pixels in individualvariables.

10. Also the overall sum and maximum pixel counts are calculated and stored in variables TOTALSUMPIXELCOUNTS and Resultpixelcounts.

11. The maximum pixel counts found in variable Resultpixelcounts is used to find threshold values and pixel counts which are less than this threshold values are eliminated.

12. 10% Threshold value is assigned in variables tenthresholdpixelcounts28-tenthresholdpixelcounts34 for corresponding slices.

13. Values in variables Spixelcounts28-Spixelcounts34 which less than these threshold are eliminated and remaining pixel counts are stored in corresponding variables

14. Again sparse command is applied to eliminate the zeros and 10%thresholdpixel counts of slices 28-34 are stored in variables tenthresholdpixelcountsS28-tenthresholdpixelcountsS34

15. The mean and maximum of individual slices and the overall sum are calculated.

16. The program is again repeated for 20%, 30%, 40% and 50% thresholds for achieving better accuracy in calculation.

17. A typical ROI drawn is shown below.

114

Matlab Program:

```
x=[1:128];
y=[1:128];
a=[1:16384];
info = dicominfo('PET_CT_AC2032.dcm');
Y = dicomread(info);
imshow(Y,[]);
Y1 = uint16(Y);
%figure; imshow(Y1,[]);
Y32 = double(Y1);
figure; imshow(Y32,[]);
BW(x,y) = roipoly(Y1);
figure; imshow(BW(x,y));
info28 = dicominfo('PET_CT_AC2028.dcm');
Y28 = dicomread(info28);
Y28 = uint16(Y28);
Y28 = double(Y28);
info29 = dicominfo('PET_CT_AC2029.dcm');
Y29 = dicomread(info29);
Y29 = uint16(Y29);
Y29 = double(Y29);
info30 = dicominfo('PET_CT_AC2030.dcm');
Y30 = dicomread(info30);
Y30 = uint16(Y30);
Y30 = double(Y30);
info31 = dicominfo('PET_CT_AC2031.dcm');
Y31 = dicomread(info31);
Y31 = uint16(Y31);
Y31 = double(Y31);
info33 = dicominfo('PET_CT_AC2033.dcm');
Y33 = dicomread(info33);
Y33 = uint16(Y33);
Y33 = double(Y33);
info34 = dicominfo('PET_CT_AC2034.dcm');
Y34 = dicomread(info34);
Y34 = uint16(Y34);
Y34 = double(Y34);

a=1;
for i = 1:128
for j = 1:128

if BW(i,j) == 1
```

```
 v32(a) = Y32(i,j);
 counts32(a) = v32(a);
 pixelcounts32(a) = counts32(a).*slope;
 v28(a) = Y28(i,j);
 counts28(a) = v28(a);
 pixelcounts28(a) = counts28(a).*slope;
 v29(a) = Y29(i,j);
 counts29(a) = v29(a);
 pixelcounts29(a) = counts29(a).* slope;
 v30(a) = Y30(i,j);
 counts30(a) = v30(a);
 pixelcounts30(a) = counts30(a).*slope;
 v31(a) = Y31(i,j);
 counts31(a) = v31(a);
 pixelcounts31(a) = counts31(a).*slope;
 v33(a) = Y33(i,j);
 counts33(a) = v33(a);
 pixelcounts33(a) = counts33(a).*slope;
 v34(a) = Y34(i,j);
 counts34(a) = v34(a);
 pixelcounts34(a) = counts34(a).*slope;
 a=a+1;
 end
 j=j+1;
 end
 i=i+1;
 end


 Spixelcounts28 = sparse(pixelcounts28);
 Spixelcounts29 = sparse(pixelcounts29);
 Spixelcounts30 = sparse(pixelcounts30);
 Spixelcounts31 = sparse(pixelcounts31);
 Spixelcounts32 = sparse(pixelcounts32);
 Spixelcounts33 = sparse(pixelcounts33);
 Spixelcounts34 = sparse(pixelcounts34);

 SUMpixelcounts(28) = sum(Spixelcounts28)
 nnz(Spixelcounts28);
 MEANpixelcounts28 = SUMpixelcounts(28)/nnz(Spixelcounts28)
 MAXpixelcounts(28) = max(Spixelcounts28)
 SUMpixelcounts(29) = sum(Spixelcounts29)
 nnz(Spixelcounts29);
 MEANpixelcounts29 = SUMpixelcounts(29)/nnz(Spixelcounts29)
 MAXpixelcounts(29) = max(Spixelcounts29)
```

```
SUMpixelcounts(30) = sum(Spixelcounts30)
nnz(Spixelcounts30);
MEANpixelcounts30 = SUMpixelcounts(30)/nnz(Spixelcounts30)
MAXpixelcounts(30) = max(Spixelcounts30)
SUMpixelcounts(31) = sum(Spixelcounts31)
nnz(Spixelcounts31);
MEANpixelcounts31 = SUMpixelcounts(31)/nnz(Spixelcounts31)
MAXpixelcounts(31) = max(Spixelcounts31)
SUMpixelcounts(32) = sum(Spixelcounts32)
nnz(Spixelcounts32);
MEANpixelcounts32 = SUMpixelcounts(32)/nnz(Spixelcounts32)
MAXpixelcounts(32) = max(Spixelcounts32)
SUMpixelcounts(33) = sum(Spixelcounts33)
nnz(Spixelcounts33);
MEANpixelcounts33 = SUMpixelcounts(33)/nnz(Spixelcounts33)
MAXpixelcounts(33) = max(Spixelcounts33)
SUMpixelcounts(34) = sum(Spixelcounts34)
nnz(Spixelcounts34);
MEANpixelcounts34 = SUMpixelcounts(34)/nnz(Spixelcounts34)
MAXpixelcounts(34) = max(Spixelcounts34)

for s = 1:7
    SUMALLpixelcounts(s) = SUMpixelcounts(s+27);
    s=s+1;
end

TOTALSUMPIXELCOUNTS = sum(SUMALLpixelcounts)

for m = 1:7
    MAXALLpixelcounts(m) = MAXpixelcounts(m+27);
    m=m+1;
end

Resultpixelcounts = max(MAXALLpixelcounts)


% 10%CALCULATIONS:

tenthresholdpixelcounts32 = Resultpixelcounts.*0.1;
tenthresholdpixelcounts32;
tenthresholdpixelcounts28 = Resultpixelcounts.*0.1;
tenthresholdpixelcounts28;
tenthresholdpixelcounts29 = Resultpixelcounts.*0.1;
tenthresholdpixelcounts29;
tenthresholdpixelcounts30 = Resultpixelcounts.*0.1;
```

```
tenthresholdpixelcounts30;
tenthresholdpixelcounts31 = Resultpixelcounts.*0.1;
tenthresholdpixelcounts31;
tenthresholdpixelcounts33 = Resultpixelcounts.*0.1;
tenthresholdpixelcounts33
tenthresholdpixelcounts34 = Resultpixelcounts.*0.1;
tenthresholdpixelcounts34;

n32 = numel(Spixelcounts32)
for  a = 1:n32

if  Spixelcounts32(a) > tenthresholdpixelcounts32
    tenanswerpixelcounts32(a) = Spixelcounts32(a);
end
if  Spixelcounts28(a) > tenthresholdpixelcounts28
    tenanswerpixelcounts28(a) = Spixelcounts28(a);
end
if  Spixelcounts29(a) > tenthresholdpixelcounts29
    tenanswerpixelcounts29(a) = Spixelcounts29(a);
end
if  Spixelcounts30(a) > tenthresholdpixelcounts30
    tenanswerpixelcounts30(a) = Spixelcounts30(a);
end
if  Spixelcounts31(a) > tenthresholdpixelcounts31
    tenanswerpixelcounts31(a) = Spixelcounts31(a);
end
if  Spixelcounts33(a) > tenthresholdpixelcounts33
    tenanswerpixelcounts33(a) = Spixelcounts33(a);
end
if  Spixelcounts34(a) > tenthresholdpixelcounts34
    tenanswerpixelcounts34(a) = Spixelcounts34(a);
end

a=a+1;
end
tenanswerpixelcounts32(a);

%warning off MATLAB:m_warning_end_without_block

tenthresholdpixelcountsS32 = sparse(tenanswerpixelcounts32);
SUMtenthresholdpixelcounts(32) = sum(tenthresholdpixelcountsS32)
MEANtenthresholdpixelcounts(32) = mean(tenthresholdpixelcountsS32)
MAXtenthresholdpixelcounts(32) = max(tenthresholdpixelcountsS32)
tenthresholdpixelcountsS28 = sparse(tenanswerpixelcounts28);
SUMtenthresholdpixelcounts(28) = sum(tenthresholdpixelcountsS28)
```

nnz(tenthresholdpixelcountsS28);
MEANtenthresholdpixelcounts28 = SUMtenthresholdpixelcounts(28)/
nnz(tenthresholdpixelcountsS28)
MAXtenthresholdpixelcounts(28) = max(tenthresholdpixelcountsS28)
tenthresholdpixelcountsS29 = sparse(tenanswerpixelcounts29);
SUMtenthresholdpixelcounts(29) = sum(tenthresholdpixelcountsS29)
nnz(tenthresholdpixelcountsS29);
MEANtenthresholdpixelcounts29 = SUMtenthresholdpixelcounts(29)/
nnz(tenthresholdpixelcountsS29)
MAXtenthresholdpixelcounts(29) = max(tenthresholdpixelcountsS29)
tenthresholdpixelcountsS30 = sparse(tenanswerpixelcounts30);
SUMtenthresholdpixelcounts(30) = sum(tenthresholdpixelcountsS30)
nnz(tenthresholdpixelcountsS30);
MEANtenthresholdpixelcounts30 = SUMtenthresholdpixelcounts(30)/
nnz(tenthresholdpixelcountsS30)
MAXtenthresholdpixelcounts(30) = max(tenthresholdpixelcountsS30)
tenthresholdpixelcountsS31 = sparse(tenanswerpixelcounts31);
SUMtenthresholdpixelcounts(31) = sum(tenthresholdpixelcountsS31)
nnz(tenthresholdpixelcountsS31);
MEANtenthresholdpixelcounts31 = SUMtenthresholdpixelcounts(31)/
nnz(tenthresholdpixelcountsS31)
MAXtenthresholdpixelcounts(31) = max(tenthresholdpixelcountsS31)
tenthresholdpixelcountsS33 = sparse(tenanswerpixelcounts33);
SUMtenthresholdpixelcounts(33) = sum(tenthresholdpixelcountsS33)
nnz(tenthresholdpixelcountsS33);
MEANtenthresholdpixelcounts33 = SUMtenthresholdpixelcounts(33)/
nnz(tenthresholdpixelcountsS33)
MAXtenthresholdpixelcounts(33) = max(tenthresholdpixelcountsS33)
tenthresholdpixelcountsS34 = sparse(tenanswerpixelcounts34);
SUMtenthresholdpixelcounts(34) = sum(tenthresholdpixelcountsS34)
nnz(tenthresholdpixelcountsS34);
MEANtenthresholdpixelcounts34 = SUMtenthresholdpixelcounts(34)/
nnz(tenthresholdpixelcountsS34)
MAXtenthresholdpixelcounts(34) = max(tenthresholdpixelcountsS34)

SUMALLtenthresholdpixelcounts = sum(SUMtenthresholdpixelcounts(28)+
SUMtenthresholdpixelcounts(29)+ SUMtenthresholdpixelcounts(30)+
SUMtenthresholdpixelcounts(31)+ SUMtenthresholdpixelcounts(32)+
SUMtenthresholdpixelcounts(33)+ SUMtenthresholdpixelcounts(34))

```
% 20%CALCULATIONS:

twentythresholdpixelcounts32 = Resultpixelcounts.*0.2;
twentythresholdpixelcounts32;
twentythresholdpixelcounts28 = Resultpixelcounts.*0.2;
twentythresholdpixelcounts28;
twentythresholdpixelcounts29 = Resultpixelcounts.*0.2;
twentythresholdpixelcounts29;
twentythresholdpixelcounts30 = Resultpixelcounts.*0.2;
twentythresholdpixelcounts30;
twentythresholdpixelcounts31 = Resultpixelcounts.*0.2;
twentythresholdpixelcounts31;
twentythresholdpixelcounts33 = Resultpixelcounts.*0.2;
twentythresholdpixelcounts33
twentythresholdpixelcounts34 = Resultpixelcounts.*0.2;
twentythresholdpixelcounts34;

n32 = numel(Spixelcounts32)
 for  a = 1:n32


if  Spixelcounts32(a) > twentythresholdpixelcounts32
    twentyanswerpixelcounts32(a) = Spixelcounts32(a);
end
if  Spixelcounts28(a) > twentythresholdpixelcounts28
    twentyanswerpixelcounts28(a) = Spixelcounts28(a);
end
if  Spixelcounts29(a) > twentythresholdpixelcounts29
    twentyanswerpixelcounts29(a) = Spixelcounts29(a);
end
if  Spixelcounts30(a) > twentythresholdpixelcounts30
    twentyanswerpixelcounts30(a) = Spixelcounts30(a);
end
if  Spixelcounts31(a) > twentythresholdpixelcounts31
    twentyanswerpixelcounts31(a) = Spixelcounts31(a);
end
if  Spixelcounts33(a) > twentythresholdpixelcounts33
    twentyanswerpixelcounts33(a) = Spixelcounts33(a);
end
if  Spixelcounts34(a) > twentythresholdpixelcounts34
   twentyanswerpixelcounts34(a) = Spixelcounts34(a);
end

a=a+1;
end
twentyanswerpixelcounts32(a);
```

```
twentythresholdpixelcountsS32 = sparse(twentyanswerpixelcounts32);
SUMtwentythresholdpixelcounts(32) = sum(twentythresholdpixelcountsS32)
nnz(twentythresholdpixelcountsS32);
MEANtwentythresholdpixelcounts32 = SUMtwentythresholdpixelcounts(32)/
nnz(twentythresholdpixelcountsS32)
MAXtwentythresholdpixelcounts(32) = max(twentythresholdpixelcountsS32)
twentythresholdpixelcountsS28 = sparse(twentyanswerpixelcounts28);
SUMtwentythresholdpixelcounts(28) = sum(twentythresholdpixelcountsS28)
nnz(twentythresholdpixelcountsS28);
MEANtwentythresholdpixelcounts28 = SUMtwentythresholdpixelcounts(28)/
nnz(twentythresholdpixelcountsS28)
MAXtwentythresholdpixelcounts(28) = max(twentythresholdpixelcountsS28)
twentythresholdpixelcountsS29 = sparse(twentyanswerpixelcounts29);
SUMtwentythresholdpixelcounts(29) = sum(twentythresholdpixelcountsS29)
nnz(twentythresholdpixelcountsS29);
MEANtwentythresholdpixelcounts29 = SUMtwentythresholdpixelcounts(29)/
nnz(twentythresholdpixelcountsS29)
MAXtwentythresholdpixelcounts(29) = max(twentythresholdpixelcountsS29)
twentythresholdpixelcountsS30 = sparse(twentyanswerpixelcounts30);
SUMtwentythresholdpixelcounts(30) = sum(twentythresholdpixelcountsS30)
nnz(twentythresholdpixelcountsS30);
MEANtwentythresholdpixelcounts30 = SUMtwentythresholdpixelcounts(30)/
nnz(twentythresholdpixelcountsS30)
MAXtwentythresholdpixelcounts(30) = max(twentythresholdpixelcountsS30)
twentythresholdpixelcountsS31 = sparse(twentyanswerpixelcounts31);
SUMtwentythresholdpixelcounts(31) = sum(twentythresholdpixelcountsS31)
nnz(twentythresholdpixelcountsS31);
MEANtwentythresholdpixelcounts31 = SUMtwentythresholdpixelcounts(31)/
nnz(twentythresholdpixelcountsS31)
MAXtwentythresholdpixelcounts(31) = max(twentythresholdpixelcountsS31)
twentythresholdpixelcountsS33 = sparse(twentyanswerpixelcounts33);
SUMtwentythresholdpixelcounts(33) = sum(twentythresholdpixelcountsS33)
nnz(twentythresholdpixelcountsS33);
MEANtwentythresholdpixelcounts33 = SUMtwentythresholdpixelcounts(33)/
nnz(twentythresholdpixelcountsS33)
MAXtwentythresholdpixelcounts(33) = max(twentythresholdpixelcountsS33)
twentythresholdpixelcountsS34 = sparse(twentyanswerpixelcounts34);
SUMtwentythresholdpixelcounts(34) = sum(twentythresholdpixelcountsS34)
nnz(twentythresholdpixelcountsS34);
MEANtwentythresholdpixelcounts34 = SUMtwentythresholdpixelcounts(34)/
nnz(twentythresholdpixelcountsS34)
MAXtwentythresholdpixelcounts(34) = max(twentythresholdpixelcountsS34)
```

```
SUMALLtwentythresholdpixelcounts = sum(SUMtwentythresholdpixelcounts(28)+
SUMtwentythresholdpixelcounts(29)+ SUMtwentythresholdpixelcounts(30)+
SUMtwentythresholdpixelcounts(31)+ SUMtwentythresholdpixelcounts(32)+
SUMtwentythresholdpixelcounts(33)+ SUMtwentythresholdpixelcounts(34))


% 30%CALCULATIONS:

thirtythresholdpixelcounts32 = Resultpixelcounts.*0.3;
thirtythresholdpixelcounts32;
thirtythresholdpixelcounts28 = Resultpixelcounts.*0.3;
thirtythresholdpixelcounts28;
thirtythresholdpixelcounts29 = Resultpixelcounts.*0.3;
thirtythresholdpixelcounts29;
thirtythresholdpixelcounts30 = Resultpixelcounts.*0.3;
thirtythresholdpixelcounts30;
thirtythresholdpixelcounts31 = Resultpixelcounts.*0.3;
thirtythresholdpixelcounts31;
thirtythresholdpixelcounts33 = Resultpixelcounts.*0.3;
thirtythresholdpixelcounts33
thirtythresholdpixelcounts34 = Resultpixelcounts.*0.3;
thirtythresholdpixelcounts34;

n32 = numel(Spixelcounts32)
 for  a = 1:n32

if  Spixelcounts32(a) > thirtythresholdpixelcounts32
    thirtyanswerpixelcounts32(a) = Spixelcounts32(a);
end
if  Spixelcounts28(a) > thirtythresholdpixelcounts28
   thirtyanswerpixelcounts28(a) = Spixelcounts28(a);
end
if  Spixelcounts29(a) > thirtythresholdpixelcounts29
    thirtyanswerpixelcounts29(a) = Spixelcounts29(a);
end
if  Spixelcounts30(a) > thirtythresholdpixelcounts30
    thirtyanswerpixelcounts30(a) = Spixelcounts30(a);
end
if  Spixelcounts31(a) > thirtythresholdpixelcounts31
    thirtyanswerpixelcounts31(a) = Spixelcounts31(a);
end
if  Spixelcounts33(a) > thirtythresholdpixelcounts33
    thirtyanswerpixelcounts33(a) = Spixelcounts33(a);
```

```
  end
  if  Spixelcounts34(a) > thirtythresholdpixelcounts34
      thirtyanswerpixelcounts34(a) = Spixelcounts34(a);
  end


  a=a+1;
  end
  thirtyanswerpixelcounts32(a);

  % warning off MATLAB:m_warning_end_without_block

  thirtythresholdpixelcountsS32 = sparse(thirtyanswerpixelcounts32);
  SUMthirtythresholdpixelcounts(32) = sum(thirtythresholdpixelcountsS32)
  nnz(thirtythresholdpixelcountsS32);
  MEANthirtythresholdpixelcounts32 = SUMthirtythresholdpixelcounts(32)/
  nnz(thirtythresholdpixelcountsS32)
  MAXthirtythresholdpixelcounts(32) = max(thirtythresholdpixelcountsS32)
  thirtythresholdpixelcountsS28 = sparse(thirtyanswerpixelcounts28);
  SUMthirtythresholdpixelcounts(28) = sum(thirtythresholdpixelcountsS28)
  nnz(thirtythresholdpixelcountsS28);
  MEANthirtythresholdpixelcounts28 = SUMthirtythresholdpixelcounts(28)/
  nnz(thirtythresholdpixelcountsS28)
  MAXthirtythresholdpixelcounts(28) = max(thirtythresholdpixelcountsS28)
  thirtythresholdpixelcountsS29 = sparse(thirtyanswerpixelcounts29);
  SUMthirtythresholdpixelcounts(29) = sum(thirtythresholdpixelcountsS29)
  nnz(thirtythresholdpixelcountsS29);
  MEANthirtythresholdpixelcounts29 = SUMthirtythresholdpixelcounts(29)/
  nnz(thirtythresholdpixelcountsS29)
  MAXthirtythresholdpixelcounts(29) = max(thirtythresholdpixelcountsS29)
  thirtythresholdpixelcountsS30 = sparse(thirtyanswerpixelcounts30);
  SUMthirtythresholdpixelcounts(30) = sum(thirtythresholdpixelcountsS30)
  nnz(thirtythresholdpixelcountsS30);
  MEANthirtythresholdpixelcounts30 = SUMthirtythresholdpixelcounts(30)/
  nnz(thirtythresholdpixelcountsS30)
  MAXthirtythresholdpixelcounts(30) = max(thirtythresholdpixelcountsS30)
  thirtythresholdpixelcountsS31 = sparse(thirtyanswerpixelcounts31);
  SUMthirtythresholdpixelcounts(31) = sum(thirtythresholdpixelcountsS31)
  nnz(thirtythresholdpixelcountsS31);
  MEANthirtythresholdpixelcounts31 = SUMthirtythresholdpixelcounts(31)/
  nnz(twentythresholdpixelcountsS31)
  MAXthirtythresholdpixelcounts(31) = max(thirtythresholdpixelcountsS31)
  thirtythresholdpixelcountsS33 = sparse(thirtyanswerpixelcounts33);
  SUMthirtythresholdpixelcounts(33) = sum(thirtythresholdpixelcountsS33)
  nnz(thirtythresholdpixelcountsS33);
  MEANthirtythresholdpixelcounts33 = SUMthirtythresholdpixelcounts(33)/
```

nnz(twentythresholdpixelcountsS33)
MAXthirtythresholdpixelcounts(33) = max(thirtythresholdpixelcountsS33)
thirtythresholdpixelcountsS34 = sparse(thirtyanswerpixelcounts34);
SUMthirtythresholdpixelcounts(34) = sum(thirtythresholdpixelcountsS34)
nnz(thirtythresholdpixelcountsS34);
MEANthirtythresholdpixelcounts34 = SUMthirtythresholdpixelcounts(34)/
nnz(twentythresholdpixelcountsS34)
MAXthirtythresholdpixelcounts(34) = max(thirtythresholdpixelcountsS34)

SUMALLthirtythresholdpixelcounts = sum(SUMthirtythresholdpixelcounts(28)+
SUMthirtythresholdpixelcounts(29)+ SUMthirtythresholdpixelcounts(30)+
SUMthirtythresholdpixelcounts(31)+ SUMthirtythresholdpixelcounts(32)+
SUMthirtythresholdpixelcounts(33)+ SUMthirtythresholdpixelcounts(34))

% 40%CALCULATIONS:

fortythresholdpixelcounts32 = Resultpixelcounts.*0.4;
fortythresholdpixelcounts32;
fortythresholdpixelcounts28 = Resultpixelcounts.*0.4;
fortythresholdpixelcounts28;
fortythresholdpixelcounts29 = Resultpixelcounts.*0.4;
fortythresholdpixelcounts29;
fortythresholdpixelcounts30 = Resultpixelcounts.*0.4;
fortythresholdpixelcounts30;
fortythresholdpixelcounts31 = Resultpixelcounts.*0.4;
fortythresholdpixelcounts31;
fortythresholdpixelcounts33 = Resultpixelcounts.*0.4;
fortythresholdpixelcounts33
fortythresholdpixelcounts34 = Resultpixelcounts.*0.4;
fortythresholdpixelcounts34;

n32 = numel(Spixelcounts32)
 for a = 1:n32

if Spixelcounts32(a) > fortythresholdpixelcounts32
    fortyanswerpixelcounts32(a) = Spixelcounts32(a);
end
if Spixelcounts28(a) > fortythresholdpixelcounts28
   fortyanswerpixelcounts28(a) = Spixelcounts28(a);
end
if Spixelcounts29(a) > fortythresholdpixelcounts29
    fortyanswerpixelcounts29(a) = Spixelcounts29(a);
end

```matlab
if  Spixelcounts30(a) > fortythresholdpixelcounts30
    fortyanswerpixelcounts30(a) = Spixelcounts30(a);
end
if  Spixelcounts31(a) > fortythresholdpixelcounts31
    fortyanswerpixelcounts31(a) = Spixelcounts31(a);
end
if  Spixelcounts33(a) > fortythresholdpixelcounts33
    fortyanswerpixelcounts33(a) = Spixelcounts33(a);
end
if  Spixelcounts34(a) > fortythresholdpixelcounts34
    fortyanswerpixelcounts34(a) = Spixelcounts34(a);
end
a=a+1;
end
fortyanswerpixelcounts32(a);

% warning off MATLAB:m_warning_end_without_block

fortythresholdpixelcountsS32 = sparse(fortyanswerpixelcounts32);
SUMfortythresholdpixelcounts(32) = sum(fortythresholdpixelcountsS32)
nnz(fortythresholdpixelcountsS32);
MEANfortythresholdpixelcounts32 = SUMfortythresholdpixelcounts(32)/
nnz(fortythresholdpixelcountsS32)
MAXfortythresholdpixelcounts(32) = max(fortythresholdpixelcountsS32)
fortythresholdpixelcountsS28 = sparse(fortyanswerpixelcounts28);
SUMfortythresholdpixelcounts(28) = sum(fortythresholdpixelcountsS28)
nnz(fortythresholdpixelcountsS28);
MEANfortythresholdpixelcounts28 = SUMfortythresholdpixelcounts(28)/
nnz(fortythresholdpixelcountsS28)
MAXfortythresholdpixelcounts(28) = max(fortythresholdpixelcountsS28)
fortythresholdpixelcountsS29 = sparse(fortyanswerpixelcounts29);
SUMfortythresholdpixelcounts(29) = sum(fortythresholdpixelcountsS29)
nnz(fortythresholdpixelcountsS29);
MEANfortythresholdpixelcounts29 = SUMfortythresholdpixelcounts(29)/
nnz(fortythresholdpixelcountsS29)
MAXfortythresholdpixelcounts(29) = max(fortythresholdpixelcountsS29)
fortythresholdpixelcountsS30 = sparse(fortyanswerpixelcounts30);
SUMfortythresholdpixelcounts(30) = sum(fortythresholdpixelcountsS30)
nnz(fortythresholdpixelcountsS30);
MEANfortythresholdpixelcounts30 = SUMfortythresholdpixelcounts(30)/
nnz(fortythresholdpixelcountsS30)
MAXfortythresholdpixelcounts(30) = max(fortythresholdpixelcountsS30)
fortythresholdpixelcountsS31 = sparse(fortyanswerpixelcounts31);
SUMfortythresholdpixelcounts(31) = sum(fortythresholdpixelcountsS31)
nnz(fortythresholdpixelcountsS31);
```

MEANfortythresholdpixelcounts31 = SUMfortythresholdpixelcounts(31)/ nnz(fortythresholdpixelcountsS31)
MAXfortythresholdpixelcounts(31) = max(fortythresholdpixelcountsS31)
fortythresholdpixelcountsS33 = sparse(fortyanswerpixelcounts33);
SUMfortythresholdpixelcounts(33) = sum(fortythresholdpixelcountsS33) nnz(fortythresholdpixelcountsS33);
MEANfortythresholdpixelcounts33 = SUMfortythresholdpixelcounts(33)/ nnz(fortythresholdpixelcountsS33)
MAXfortythresholdpixelcounts(33) = max(fortythresholdpixelcountsS33)
fortythresholdpixelcountsS34 = sparse(fortyanswerpixelcounts34);
SUMfortythresholdpixelcounts(34) = sum(fortythresholdpixelcountsS34) nnz(fortythresholdpixelcountsS34);
MEANfortythresholdpixelcounts34 = SUMfortythresholdpixelcounts(34)/ nnz(fortythresholdpixelcountsS34)
MAXfortythresholdpixelcounts(34) = max(fortythresholdpixelcountsS34)

SUMALLfortythresholdpixelcounts = sum(SUMfortythresholdpixelcounts(28)+ SUMfortythresholdpixelcounts(29)+ SUMfortythresholdpixelcounts(30)+ SUMfortythresholdpixelcounts(31)+ SUMfortythresholdpixelcounts(32)+ SUMfortythresholdpixelcounts(33)+ SUMfortythresholdpixelcounts(34))

% 50%CALCULATIONS:

fiftythresholdpixelcounts32 = Resultpixelcounts.*0.5;
fiftythresholdpixelcounts32;
fiftythresholdpixelcounts28 = Resultpixelcounts.*0.5;
fiftythresholdpixelcounts28;
fiftythresholdpixelcounts29 = Resultpixelcounts.*0.5;
fiftythresholdpixelcounts29;
fiftythresholdpixelcounts30 = Resultpixelcounts.*0.5;
fiftythresholdpixelcounts30;
fiftythresholdpixelcounts31 = Resultpixelcounts.*0.5;
fiftythresholdpixelcounts31;
fiftythresholdpixelcounts33 = Resultpixelcounts.*0.5;
fiftythresholdpixelcounts33
fiftythresholdpixelcounts34 = Resultpixelcounts.*0.5;
fiftythresholdpixelcounts34;

n32 = numel(Spixelcounts32)
 for a = 1:n32

if Spixelcounts32(a) > fiftythresholdpixelcounts32
    fiftyanswerpixelcounts32(a) = Spixelcounts32(a);
end
if Spixelcounts28(a) > fiftythresholdpixelcounts28

```
      fiftyanswerpixelcounts28(a) = Spixelcounts28(a);
   end
 if  Spixelcounts29(a) > fiftythresholdpixelcounts29
      fiftyanswerpixelcounts29(a) = Spixelcounts29(a);
   end
 if  Spixelcounts30(a) > fiftythresholdpixelcounts30
      fiftyanswerpixelcounts30(a) = Spixelcounts30(a);
   end
 if  Spixelcounts31(a) > fiftythresholdpixelcounts31
      fiftyanswerpixelcounts31(a) = Spixelcounts31(a);
   end
 if  Spixelcounts33(a) > fiftythresholdpixelcounts33
      fiftyanswerpixelcounts33(a) = Spixelcounts33(a);
   end
 if  Spixelcounts34(a) > fiftythresholdpixelcounts34
      fiftyanswerpixelcounts34(a) = Spixelcounts34(a);
   end


 a=a+1;


 end
 fiftyanswerpixelcounts32(a);

 % warning off MATLAB:m_warning_end_without_block

 fiftythresholdpixelcountsS32 = sparse(fiftyanswerpixelcounts32);
 SUMfiftythresholdpixelcounts(32) = sum(fiftythresholdpixelcountsS32)
 nnz(fiftythresholdpixelcountsS32);
 MEANfiftythresholdpixelcounts32 = SUMfiftythresholdpixelcounts(32)/
 nnz(fiftythresholdpixelcountsS32)
 MAXfiftythresholdpixelcounts(32) = max(fiftythresholdpixelcountsS32)
 fiftythresholdpixelcountsS28 = sparse(fiftyanswerpixelcounts28);
 SUMfiftythresholdpixelcounts(28) = sum(fiftythresholdpixelcountsS28)
 nnz(fiftythresholdpixelcountsS28);
 MEANfiftythresholdpixelcounts28 = SUMfiftythresholdpixelcounts(28)/
 nnz(fiftythresholdpixelcountsS28)
 MAXfiftythresholdpixelcounts(28) = max(fiftythresholdpixelcountsS28)
 fiftythresholdpixelcountsS29 = sparse(fiftyanswerpixelcounts29);
 SUMfiftythresholdpixelcounts(29) = sum(fiftythresholdpixelcountsS29)
 nnz(fiftythresholdpixelcountsS29);
 MEANfiftythresholdpixelcounts29 = SUMfiftythresholdpixelcounts(29)/
 nnz(fiftythresholdpixelcountsS29)
 MAXfiftythresholdpixelcounts(29) = max(fiftythresholdpixelcountsS29)
 fiftythresholdpixelcountsS30 = sparse(fiftyanswerpixelcounts30);
 SUMfiftythresholdpixelcounts(30) = sum(fiftythresholdpixelcountsS30)
```

nnz(fiftythresholdpixelcountsS30);
MEANfiftythresholdpixelcounts30 = SUMfiftythresholdpixelcounts(30)/
nnz(fiftythresholdpixelcountsS30)
MAXfiftythresholdpixelcounts(30) = max(fiftythresholdpixelcountsS30)
fiftythresholdpixelcountsS31 = sparse(fiftyanswerpixelcounts31);
SUMfiftythresholdpixelcounts(31) = sum(fiftythresholdpixelcountsS31)
nnz(fiftythresholdpixelcountsS31);
MEANfiftythresholdpixelcounts31 = SUMfiftythresholdpixelcounts(31)/
nnz(fiftythresholdpixelcountsS31)
MAXfiftythresholdpixelcounts(31) = max(fiftythresholdpixelcountsS31)
fiftythresholdpixelcountsS33 = sparse(fiftyanswerpixelcounts33);
SUMfiftythresholdpixelcounts(33) = sum(fiftythresholdpixelcountsS33)
nnz(fiftythresholdpixelcountsS33);
MEANfiftythresholdpixelcounts33 = SUMfiftythresholdpixelcounts(33)/
nnz(fiftythresholdpixelcountsS33)
MAXfiftythresholdpixelcounts(33) = max(fiftythresholdpixelcountsS33)
fiftythresholdpixelcountsS34 = sparse(fiftyanswerpixelcounts34);
SUMfiftythresholdpixelcounts(34) = sum(fiftythresholdpixelcountsS34)
nnz(fiftythresholdpixelcountsS34);
MEANfiftythresholdpixelcounts34 = SUMfiftythresholdpixelcounts(34)/
nnz(fiftythresholdpixelcountsS34)
MAXfiftythresholdpixelcounts(34) = max(fiftythresholdpixelcountsS34)
SUMALLfiftythresholdpixelcounts = sum(SUMfiftythresholdpixelcounts(28)+
SUMfiftythresholdpixelcounts(29)+ SUMfiftythresholdpixelcounts(30)+
SUMfiftythresholdpixelcounts(31)+ SUMfiftythresholdpixelcounts(32)+
SUMfiftythresholdpixelcounts(33)+ SUMfiftythresholdpixelcounts(34))