NTNU – Trondheim
Norwegian University of
Science and Technology

Attenuation correction in PET/MR

A performance assessment of UTE-based and Dixon-based methods for PET attenuation correction

Lars Birger Aasheim

Master of Science in Physics and Mathematics
Submission date: June 2014
Supervisor: Pål Erik Goa, IFY
Co-supervisor: Live Eikenes, DMF

Norwegian University of Science and Technology
Department of Physics

Abstract

Hybrid positron emission tomography/magnetic resonance imaging (PET/MR) scanners are one of the latest tools available in the field of medical imaging, and are expected to outperform the well-established PET/X-ray computed tomography (CT) scanners in a large range of fields. The perhaps largest challenge that has to be overcome before this can be achieved, is that of attenuation correction (AC) of the acquired PET images, as there is no direct relation between the MR image intensity of a tissue and its attenuating properties, as is the case in CT.

This study investigated the performance of two PET AC methods provided with the biograph mMR PET/MR scanner installed at St. Olavs Hospital (Trondheim, Norway); one for head imaging based on an ultra-short echo-time (UTE) sequence, and one for whole-body imaging based on a Dixon sequence. These AC methods were compared to the ‘gold standard’ of CT-based AC, based on activity concentrations in PET images from mMR examinations of lymphoma and lung cancer patients, corrected with the different AC methods (UTE, Dixon and CT).

The results of the study show that the UTE-based AC method leads to an underestimation of PET activity in the brain of up to 9 % in the investigated regions of interest. This is caused by underestimation of bone in the cranial region. The exclusion of bone in the Dixon-based AC method leads to underestimation of PET activity in the thorax/abdomen, indicated by an underestimation of 4 % in the liver. The two MR-based AC methods are thus not sufficiently accurate to be utilised for quantification in PET imaging.

Sammendrag

Positronemisjonstomografi/magnetisk resonanstomografiskannere (PET/MR) er et av de siste tilgjengelige verktøyene innen medisinsk avbildning, og er forventet å prestere bedre enn de veletablerte PET/computertomografi-skannerene (CT) innen en rekke områder. Den kanskje største hindringen i veien mot å oppnå dette er problemet med attenuasjonskorreksjon (AC) av de innsamlede PET-bildene, ettersom det ikke eksisterer noen direkte sammenheng mellom et vevs signalintensitet i MR-bilder og dets attenuerende egenskaper, som er tilfelle i CT.

Denne studien undersøkte ytelsen til to AC-metoder som fulgte med biograph mMR PET/MR-skanneren installert ved St. Olavs Hospital (Trondheim); én for avbildning av hode basert på en ultra-short echo-time (UTE) sekvens, og én for fullkroppsavbildning basert på en Dixon-sekvens. Disse AC-metodene ble sammenlignet med 'gullstandarden' CT-basert AC, basert på aktivitetskonsentrasjoner i PET-bilder fra mMR undersøkelser av lymfom- og lungekreft-pasienter, korrigert med de forskjellige AC-metodene (UTE, Dixon og CT).

Studiens resultater viser at den UTE-baserte AC-metoden fører til en underestimering av PET-aktivitet i de undersøkte delene av hjernen på opptil 9 %. Dette skyldes en underestimering av bein i kranie-regionen. Ekskluderingen av bein i den Dixon-baserte AC-metoden fører til en underestimering av PET-aktivitet i toraks/abdomen, indikert av en underestimering på 4 % i leveren. De to MR-baserte AC-metodene er dermed ikke nøyaktige nok til å bli benyttet til kvantifisering i PET-avbildning.

Acknowledgements

First and foremost I would like to express my utmost gratitudes for the invaluable, continuous support I have received from my main supervisor, Associate Professor Live Eikenes (Norwegian University of Science and Technology (NTNU), Trondheim, Norway), during the work with this master thesis. She has by far surpassed any of my most optimistic assumptions regarding the level of support to expect from a supervisor. Through this, she has introduced me to a level of academic enjoyment I previously did not think was achievable.

I also want to thank my secondary supervisor, Associate Professor Pål Erik Goa (NTNU), for his valuable support during the work with this master thesis, and for offering new input whenever I found myself at a standstill.

In addition, a lot of other people deserve gratitude, including, but not necessarily limited to, the following: Medical Physicist Anna Karlberg (St. Olavs Hospital, Trondheim, Norway) for happily answering any questions I had regarding the technical aspects of the mCT and mMR. The radiographers and other clinical personnel at the St. Olavs Hospital PET centre, and then especially Per Arvid Steen and Kristian Wibe Eidissen, for putting up with all the disturbances caused by intrusive master students hogging the scanners. PhD student Ilja Bezrukov (Max Planck Institute for Intelligent Systems, Tübingen, Germany), for providing insight into the obscure workflow of Siemens' scanner software and ideas on script design. Dr. Afshin Akbarzadeh (Tehran University of Medical Sciences, Iran) for providing details from his work with the `elastix` software. Thomas Keil (St. Olavs Hospital) for providing input regarding the clinical part of nuclear medicine.

Lars Birger Aasheim
June 2014

Contents

I Introduction 1

Acronyms

1D	one-dimensional	15
2D	two-dimensional	9
3D	three-dimensional	10
AC	attenuation correction	1
ACF	attenuation correction factor	25
ALARA	as low as reasonably achievable	17
APD	avalanche photodiode	5
BP	breakpoint	26
CT	X-ray computed tomography	1
DOI	depth of interaction	7
FBP	filtered backprojection	11
FDG	fluorodeoxyglucose	3
FOV	field of view	5
FWHM	full width at half maximum	35
HU	Hounsfield units	16
LAC	linear attenuation coefficient	13
LOR	line of response	6
MI	mutual information	30
MRAC	MR-based AC	27

PET/MR positron emission tomography / magnetic resonance imaging 1

PMT photomultiplier tube 5

PSF point-spread-function 2

RCS reference coordinate system 9

REK regional ethics committee 33

RF radiofrequency 18

ROI region of interest 24

SiPM silicon photomultiplier 5

SPECT single-photon emission computed tomography 5

SSD sum of squared differences 30

SUV standardised uptake value 11

UTE ultra-short echo-time 2

VOI voxel of interest 16

Part I
Introduction

1 Motivation and objective

Hybrid positron emission tomography / magnetic resonance imaging (PET/MR) scanners are one of the latest tools available in the field of medical imaging. The Siemens biograph mMR was the world's first commercially available, fully integrated PET/MR scanner, and St. Olavs Hospital (Trondheim, Norway) was fortunate enough to receive one as a gift, which was installed autumn 2013.

Since its introduction in the late 1990's, the PET/X-ray computed tomography (CT) scanner has been the multi-modality imaging system of choice, fusing excellent skeletal contrast with the molecular imaging possibilities of a PET system, complete with hardware-based image registration. Hybrid PET/MR scanners are still in an initiation stage, but are believed to outperform the PET/CT in several fields, including, but not limited to, the following. The excellent soft-tissue contrast provided by the MR imaging (MRI) modality is expected to drastically increase the possibilities in neuroimaging especially, as well as other fields. The lower dose delivered in PET/MR examinations compared to PET/CT is of great importance in therapeutic progress monitoring, where multiple examinations are necessary, especially in paediatric oncology. [12] [35] [36]

Due to its recent invention, there are still many unresolved challenges connected with the new combination of image modalities in hybrid PET/MR scanners. The perhaps largest challenge yet to be resolved is that of attenuation correction (AC) of the acquired PET images [21][36]. In PET imaging a tracer molecule labelled with a positron-emitter is injected into the patient, and the resulting radioactivity distribution is imaged by detectors surrounding the patient. The emitted radiation is attenuated by different tissues on its way from inside the body towards the detectors, and to acquire accurate, quantifiable images, it is crucial to correct for this attenuation.

The attenuation of photons in the PET energy range (around 511 keV) depends on the electron density of the tissue. In PET/MR scanners there is no direct way of measuring the electron density, as is possible with PET/CT scanners. The issue of PET AC in PET/MR must thus be solved in a more indirect manner, as will be described in the theory section.

The mMR installed at St. Olavs Hospital is provided with two different MR sequences specialised for AC; a Dixon sequence for use in whole-body imaging, and an ultra-short echo-time (UTE) sequence for use in head imaging. The UTE sequence is a new version which has not yet been clinically validated. The main objective of this study was to validate the performance of these two sequences, compared to the 'gold standard' of CT based AC. Additionally, in the beginning of this year (2014) the scanner received an upgrade in the form of a new PET image reconstruction method; Siemens' new point-spread-function (PSF) algorithm, and the assessment of this algorithm's performance is the second objective of this study.

Part I will provide the reader with the theoretical basis needed to understand and appreciate the practical issues related to the objective of this study. The subsequent parts will present the methods used, the results, the discussion of these results, and the conclusions made from the interpreted results. The appendices provide the reader with important details supporting the understanding of the practical issues related to the performed tasks.

2 Theory

This section will first present the minimum needed basic theory on PET, photon attenuation, CT and MRI, before more specifically discussing AC in PET. Finally a short overview of image registration will be given.

2.1 PET

PET is a functional imaging modality utilising biologically active molecules labelled with positron emitters (tracers) to image and quantify physiologic processes. The objective in PET imaging is to image the distribution of radioactivity from the injected tracer with an accuracy high enough that the detected activity can be accurately quantified. [42]

2.1.1 Positron emitters and tracers

A positron emitter is a radioactive isotope with an excess number of protons compared to the number of neutrons in the nucleus. Through transmutation one proton of this neutron-deficient isotope is converted to a neutron, releasing an electron neutrino (ν_e) and a positron (β^+) in the process (known as β^+-emission). The former is uncharged and has virtually no mass, and thus it does not interact with any other particles. The latter is the antiparticle associated with the electron and is thus also known as an antielectron. A positron created through β^+-emission will have an initial kinetic energy, the amount of which depends on the positron-emitting isotope. While travelling through the surrounding material the positron will lose this kinetic energy through Coulomb interactions with the electrons of nearby atoms. Once it has lost (nearly) all of its kinetic energy, the positron will annihilate with a nearby electron in what is known as an event. The annihilation creates two gamma photons, travelling in (nearly) opposite directions, each with an energy of 511 keV, which is the resting energy ($E = mc^2$) of electrons/positrons. Figure 1 shows a schematic overview of the described process. [5] [33] [42]

In PET imaging the positron emitting isotope is usually bound to a biologic molecule. This molecule is typically a substrate of a metabolic process one desires to image, such as the consumption of glucose at the cellular level. In the case of glucose consumption the radiopharmaceutical (tracer) fluorodeoxyglucose (FDG) is especially useful. FDG is a glucose-analogue in which one of the hydroxyl groups of regular glucose has been replaced with the positron-emitting fluorine isotope ^{18}F, as shown in Figure 1. After injection into the patient, FDG is transported into any glucose-consuming cell, more so in cells with a higher relative glucose consumption such as brain cells and malignant tumour cells. Normal glucose is degraded in the cell through a ten-step process known as glycolysis, to free its stored energy [24]. Due to its chemical properties, degradation of FDG will halt after the first step, trapping the tracer inside the cell, and degradation will not continue until ^{18}F has undergone radioactive decay. One important consequence of this is that the detected photons from ^{18}F-FDG become an excellent indicator of the location of glucose consumption in the body. This is highly relevant in malignancy detection, due to the aforementioned high relative glucose consumption in malignant cells. [22]

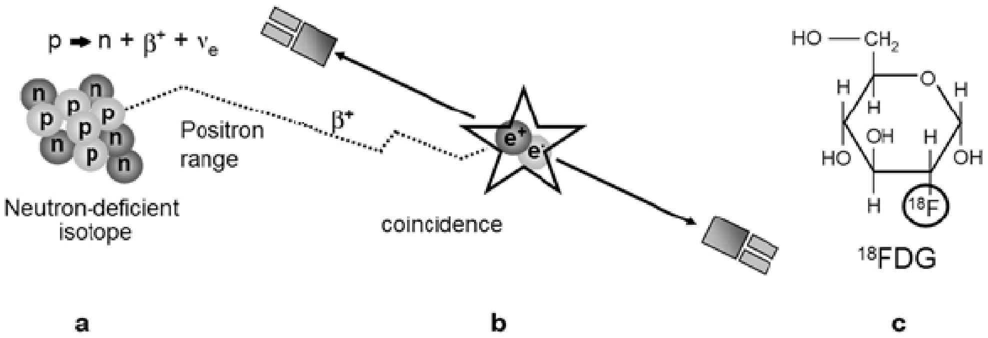


Figure 1: Overview of the basic principle of PET physics, and the molecular structure of FDG. A proton of a neutron-deficient isotope (a) is converted to a neutron, emitting a positron and a neutrino. The positron travels a certain distance (the positron range) before annihilating with an electron, emitting two photons in opposite directions (b). If transmitted in the right direction these photons can be detected by a pair of opposing detectors. c shows the molecular structure of ^{18}*F-labelled FDG, where one of the hydroxyl groups fo glucose has been replaced with the radioactive fluorine isotope* ^{18}*F. [42]*

Part of what makes PET imaging a flexible and valuable functional imaging modality is the large amount of tracers that can be produced. ^{11}C, ^{13}N, ^{15}O, and ^{18}F are all positron-emitting isotopes of chemical elements which are extremely abundant in organic and biologic compounds, thus enabling the conversion of a wide range of biologic molecules into tracers. Depending on the nature of the tracer, it is inhaled or injected either during, or a certain time before, the examination. A limiting factor is the cost of tracer production. Since many of the useful positron-emitting isotopes have a short half-life they have to be produced on-site, typically in a cyclotron. FDG is an exception, being relatively transportable due to ^{18}F's half-life of 109.8 minutes [5], contributing to FDG's position as the by far most utilised tracer in PET imaging. [33]

2.1.2 PET detectors

To detect the gamma photons emitted from the electron/positron annihilation events inside the patient, a suitable detector is needed. A typical PET detector consists of two main components; a scintillator and a photodetector. The scintillator is usually an inorganic crystal in which the gamma spectrum photons are converted to visible light (scintillation photons) through photoelectric absorption and Compton effects. The scintillation crystals are coupled to photodetectors, from which the output is a measurable signal proportional to the energy deposited in the scintillation crystal. The typical PET detector block consists of an array of crystals coupled to an array of position-sensitive photodetectors, as seen in Figure 2. [18] [33]

Scintillation crystals can be made from a range of different materials, including sodium iodide (NaI), bismuth germanium oxide (BGO), and lutetium oxyorthosilicate (LSO). Important crystal properties include stopping power (ability to attenuate incident gamma photons), conversion efficiency (relative amount of gamma energy converted to scintillation photons) and linearity. Likewise there are several photodetectors to choose from, including the photomultiplier tube (PMT), the avalanche photodiode (APD), and the silicon photomultiplier (SiPM). The PMT has been the workhorse in PET and single-photon emission computed tomography (SPECT) for several decades, but due to its size and its sensitivity even to weak magnetic fields, in PET/MR scanners it has been replaced by alternatives insensitive even to strong magnetic fields, such as the APD. [18] [33]

In a standard whole-body PET scanner the patient is positioned on a moveable patient bed. Mounted coaxially around the bed is the detector gantry. In the gantry the detector blocks are arranged in rings as seen in Figure 4, with the scintillator crystals oriented towards the center of the gantry. The axial field of view (FOV) of regular PET scanners is limited, often between 20 and 30 cm. Several acquisitions, at different bed positions, are therefore necessary to cover the whole body. [18] [42]

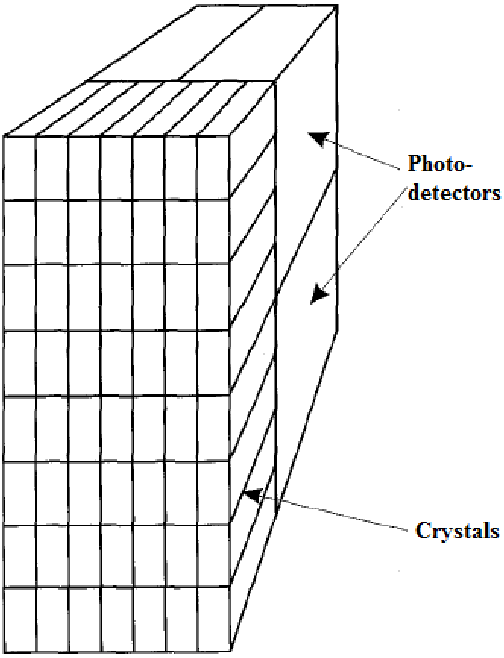


Figure 2: Example of a PET block detector, consisting of a grid of rectangular crystals coupled to a grid of photo-detectors. The photodetectors in this example image are PMTs. [33]

2.1.3 Detected events

The crucial principle behind PET imaging is the emission of *two* gamma photons from every annihilation event, travelling in *opposite* directions. Thus, when the two photons from an event are detected by two (opposing) detectors, one can assume that the event took place somewhere along a straight line connecting the two detectors. This line is known as the line of response (LOR). To ensure that two detected photons originate from the same event, a coincidence time window is set based on the known speed of gamma photons (30 cm/ns) and the time resolution of the detectors. To be accepted as originating from the same event, as a coincidence, the two photons must be detected within this time window ($\sim$ 5 - 10 ns). [5] [42]

From annihilation in patient tissue to detection in the gantry the two gamma photons can follow several paths. One is both photons passing out of the body without interacting significantly with other particles, known as a true count. Another possibility is for one or both photons to undergo Compton scattering and change direction prior to detection, known as a scattered count. Yet another possibility is for two photons originating from separate events to be detected within the coincidence time window, known as a random count. As seen in Figure 3 both scattered and random counts give rise to false LORs. Since photons lose energy if they undergo a Compton interaction, heavily scattered counts can be filtered out based on energy discrimination of detected photons. This is done by analysing the signal pulse from detected gamma photons, which is the task of the pulse-height analyser. The energy resolution of the detector determines the range of accepted energies. Random counts can be filtered out by more advanced techniques based on estimation from the singles rates of the detectors or by manipulating the coincidence time windows. [5] [27]

When the depth of interaction (DOI) in the crystal detecting a gamma photon is unknown, which is generally the case, the LOR is assumed to stretch between the center of the edge of the two detecting crystals, as indicated by the dashed line in Figure 4. Scintillator crystals with smaller cross-sections thus give better spatial resolution. However, when the crystal cross-section becomes small compared to the attenuation length of the detected photons, not knowing the DOI in the crystal becomes an issue. As shown in Figure 4, events originating in the center of the transaxial FOV will result in true LORs. Photons from events in the edges of the transaxial FOV may however traverse the incident crystal and interact in its neighbour. The detected event will then be attributed to another LOR than the true one, as indicated by the dashed line in the figure. Depending on scanner hardware and software, this can be corrected for, either by measuring the DOIs, or by mapping the PSF of the system. [5] [18]

PSF mapping The PSF of an image system is a function describing how an imaged mathematical point will appear in the resulting image. The PET PSF is spatially variant, and is influenced by a range of scanner properties, including, but not limited to, the positron range, non-collinearity of the anihiliation photons, distance between detectors, detector stopping power, and DOI. A way of improving the image quality in PET imaging is to map the PSF for every single position in the FOV. This can be done by simulation, analytical derivations, or through experimental measurements. Measuring can be done by imaging a point source in every position (or at least in a required minimum for the system model) inside the FOV of the scanner. The PSF map is incorporated in iterative reconstructions of the PET images. PSF algorithms improve resolution and scatter, and largely for the DOI problems described above. [4] [34] [40]

A well-known issue with PSF reconstruction algorithms is that of edge artefacts, or ringing artefacts, manifested as overshoots along activity edges in the PET images. The proposed explanations for this artefact include mismatches between assumed and actual system models, and insufficient support for high spatial frequencies, but will not be described here. [2] [41]

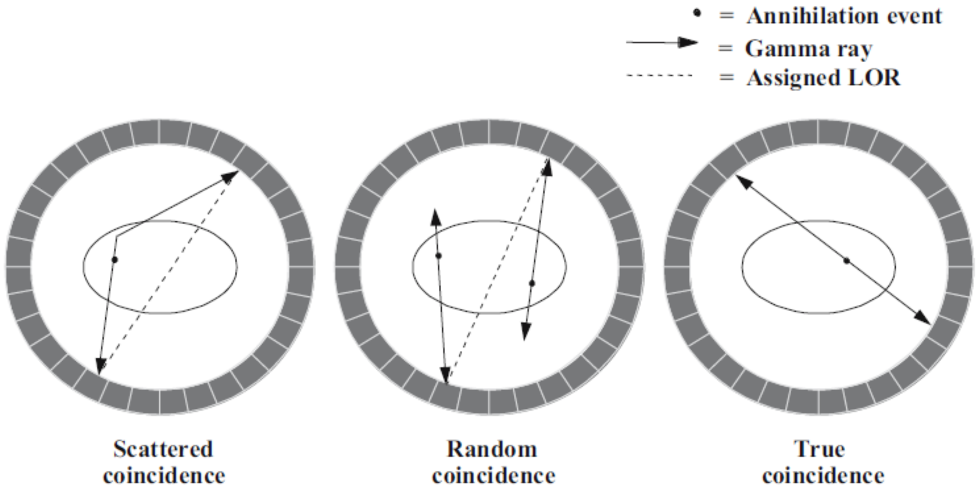


Figure 3: Schematic illustrations of different possible coincidence events recorded in PET scanners. [27]

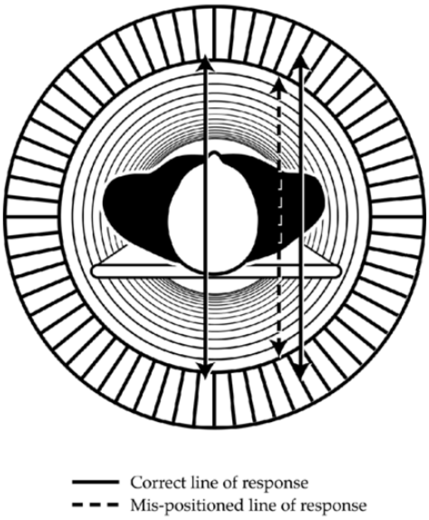


Figure 4: Schematic example of how a true event gives rise to a false LOR due to the depth of interaction-problem. [18]

2.1.4 Image reconstruction

Before continuing, the patient reference coordinate system (RCS) have to be defined. The RCS is an intuitive Cartesian coordinate system utilised when referring to the patient's body. The positive x-direction is defined to be from the right side of the patient to the left (say, from right to left shoulder). The positive y-direction is defined to be from anterior to posterior (chest to back). The positive z-direction is defined to be from bottom to top (feet to head). The x- and y-direction define the transversal plane, while z is the axial direction.

Figure 5 illustrates how two-dimensional (2D) PET measurements are saved in raw image files known as *sinograms*. The detected activity in each LOR is saved as a pixel in the sinogram, with coordinates as illustrated in the figure. Each row in the sinogram corresponds to a projection along a certain angle ϕ, while each column corresponds to a certain distance s along the axis perpendicular to the projection angle. Each detector can form LORs with a certain sector of (at least partly) opposing detectors. The angle θ defines the angle between the projection and the transversal plane. For 2D acquisitions, which are restricted to $\theta = 0$, the number of sinograms (for each bed-position) equals the number of detector rings. For three-dimensional (3D) acquisitions, which are not restricted to $\theta = 0$, the data acquisition is more complicated. The raw data can also be saved in a list file, which contains information on every single detected event. [4]

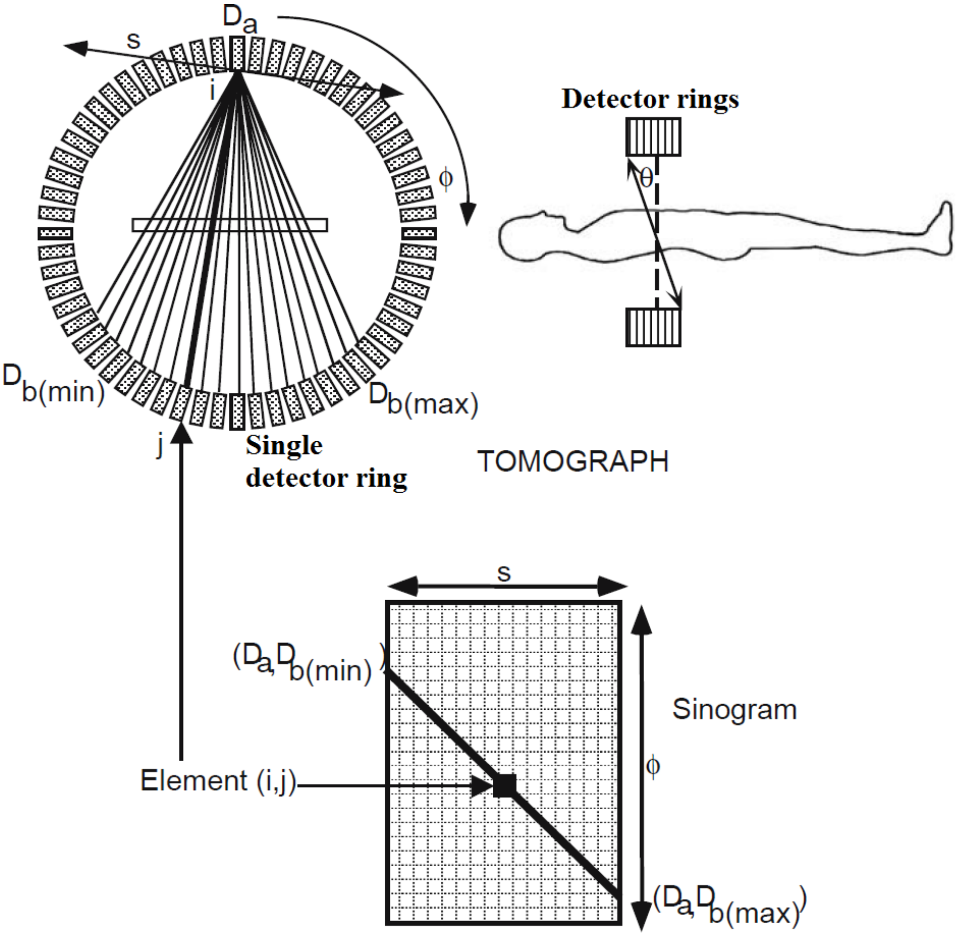


Figure 5: Illustration of how 2D PET data is saved in sinogram images. s gives the distance from the specific LOR to the center of the projection in the direction given by ϕ. The detector D_a can form LORs with all detectors between and including $D_{b(min)}$ and $D_{b(max)}$. Each row in the sinogram corresponds to a projection of the imaged object for a certain angle ϕ. The top right image illustrates how the polar angle θ is defined. [4]

Before image reconstruction the raw data is corrected for random counts, attenuation, scattered counts, etcetera. The corrected data is then used to generate an image, using one of several methods. In filtered backprojection (FBP) the data is simply backprojected into image space (as opposed to the forward projection at acquisition), after first being filtered to account for the oversampling of the center of the FOV. Iterative reconstruction methods are based on a starting presumption of the image distribution, followed by a calculation of the projections this distribution would give. These projections are compared with the projections actually measured during acquisition, and further iterations are performed to reduce the difference. [10]

2.1.5 SUV

Standardised uptake value (SUV) is a widely used quantification measure of localised activity in PET images. SUV is a calculation of the activity concentration in PET images, normalised for patient size and injected activity. The most frequently used calculation in clinical settings, SUV_{BW} (‘SUV bodyweight’), is defined as[1]

$$SUV_{\mathrm{BW}} = \frac{A}{D_{\mathrm{I}}} \cdot m_{\mathrm{patient}}, \tag{1}$$

where A is the measured absolute activity (corrected for attenuation, scattering, etcetera) in [Bq/ml], D_{i} is the injected dose in [Bq], and m_{patient} is the patient’s mass in [g]. 1 g is assumed to be equivalent to 1 ml, so the quantity is unitless. Another frequent SUV measure is SUV_{LBM} (‘SUV lean body mass’), defined as[1]

$$SUV_{\mathrm{LBM}} = \frac{A}{D_{\mathrm{I}}} \cdot LBM[\mathrm{g}], \tag{2}$$

where LBM is given as[1]

$$LBM_{\mathrm{M}}[\mathrm{kg}] = 1.10 \cdot m_{\mathrm{patient}} - 128 \left[\frac{m_{\mathrm{patient}}}{l_{\mathrm{patient}}}\right]^2,$$

for males, and as[1]

$$LBM_{\mathrm{F}}[\mathrm{kg}] = 1.07 \cdot m_{\mathrm{patient}} - 148 \left[\frac{m_{\mathrm{patient}}}{l_{\mathrm{patient}}}\right]^2,$$

---

[1] This definition is gathered from the Siemens’ syngo software directly.

for females. where m_{patient} is the patient weight given in [kg], and l_{patient} is the length of the patient given in [cm].

2.2 Photon attenuation

Attenuation of electromagnetic radiation is the reduction in intensity, caused by the scattering and absorption of photons, of a ray traversing a material with which it can interact. The attenuation depends on the energy of the photons and the properties of the traversed material. [39]

2.2.1 Attenuating interactions

X-ray and gamma radiation share much of the same energy spectrum and are thus both subject to the same attenuating interactions (see Figure 6). The interactions relevant for electromagnetic radiation of diagnostic energy include photoelectric absorption, Compton scattering, Rayleigh scattering, pair-production, triplet-production, and nuclear photoabsorption. For the photon energies observed in PET and X-ray imaging the two former are the most prominent attenuating interactions. [5] [39]

In photoelectric absorption a photon interacts with an inner shell electron of the absorbing atom. All of the photon energy is transferred to the electron, which is subsequently ejected from its shell, leaving a vacancy. The vacant shell is filled by an electron from an outer shell, emitting X-ray radiation of energy equal to the difference in binding energy between the two shells in the process. The energy of the interacting photon must be equal to or greater than the binding energy of the electron. The probability of photoelectric absorption decreases with increasing energy once this lower threshold is exceeded. Photoelectric absorption is thus more prominent for lower (X-ray) photon energies ($<$ 80-140 keV) than for electron/positron-annihilation photon energies (511 keV). [5] [39]

In Compton scattering, which as already mentioned is important in PET imaging, a photon interacts with an outer-shell electron with a low binding-energy relative to the energy of the incoming photon. Some of the photon's energy is transferred to the electron, which is subsequently ejected from the atom. The scattered photon will have a different direction and a lower energy than the incident photon. [5] [39]

The probability for all the possible interactions constitute the linear attenuation coefficient (LAC) μ. μ-values are generally given in units [cm^{-1}], and for X-ray and PET energies they are dependent on the electron density (Z-number) of the traversed material, and the incident photon energy. For a narrow beam (pencil beam) of incident intensity I_0 traversing a material of thickness x with uniform attenuation coefficient μ for the given energy, the transmitted intensity is given by

$$I_x = I_0 \mathrm{e}^{-\mu x}. \tag{3}$$

To normalise for the density of the material, the mass attenuation coefficient μ/ρ is often used. The importance of attenuation in PET (and CT) imaging will be discussed below. [5] [39]

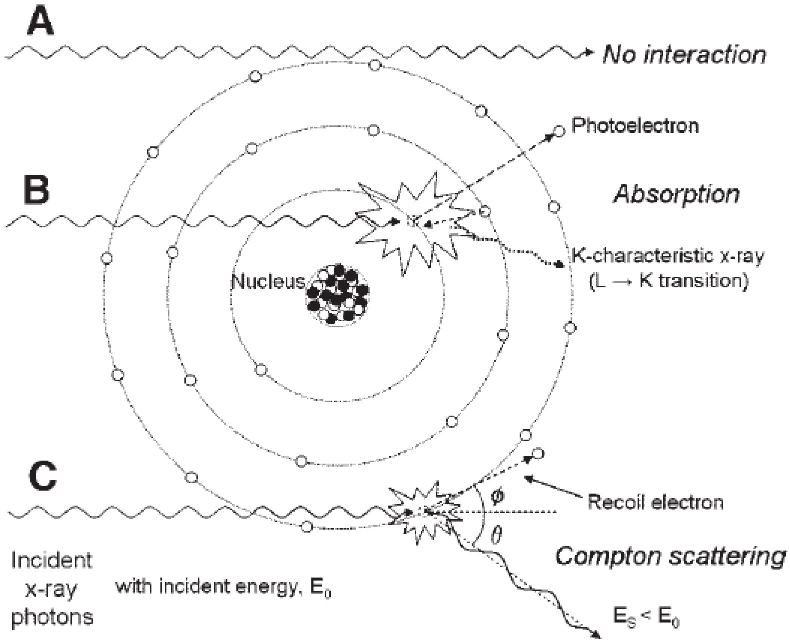


Figure 6: Schematic overview of possible photon/electron interactions relevant to attenuation of photons in PET and CT energy spectra. A: No interactions. B: Photoelectric absorption, photon fully absorbed. C: Compton scattering, the scattered photon has lower energy. [39]

2.3 CT

CT is a structural imaging modality utilising X-ray radiation to create images with excellent skeletal contrast and acceptable soft-tissue contrast. [16]

In medical imaging it is common to divide the imaged volume into voxels, the 3D equivalent to pixels in a regular 2D image. The smaller the voxels the better the resolution in the image. When an X-ray beam traverses the body it will pass through voxels of different tissue types, and thus with different attenuating properties (attenuation coefficients). The transmitted intensity I_t of a beam traversing n voxels will depend on the sum of the attenuation coefficients of all the traversed voxels according to

$$I_\mathrm{t} = I_0 \mathrm{e}^{-\sum\limits_{i=1}^{n} \mu_i x_i}, \tag{4}$$

where μ_i and x_i are the attenuation coefficient and the traversed thickness, respectively, of voxel i. This can be reformulated

$$-\ln(I_\mathrm{t}/I_0) = \sum_{i=1}^{n} \mu_i x_i, \tag{5}$$

which shows that when the incident and transmitted intensities of the beam is known, along with the direction of the beam through the voxels (determining the x_i's), one can calculate the sum of attenuation coefficients along the specific beam path. [16]

A typical CT scanner consists of a patient bed surrounded by a detector gantry. The radiation source, an X-ray tube, rotates about the patient bed, emitting a spectrum of energy determined by its peak voltage. With a peak voltage of for instance 100 kV, the emitted photons will have a maximum energy of 100 keV, but the mean energy of the emitted spectrum will be much lower, as shown in Figure 7. The radiation detectors are either rotating with the X-ray tube, or mounted in stationary rings in the gantry. [16]

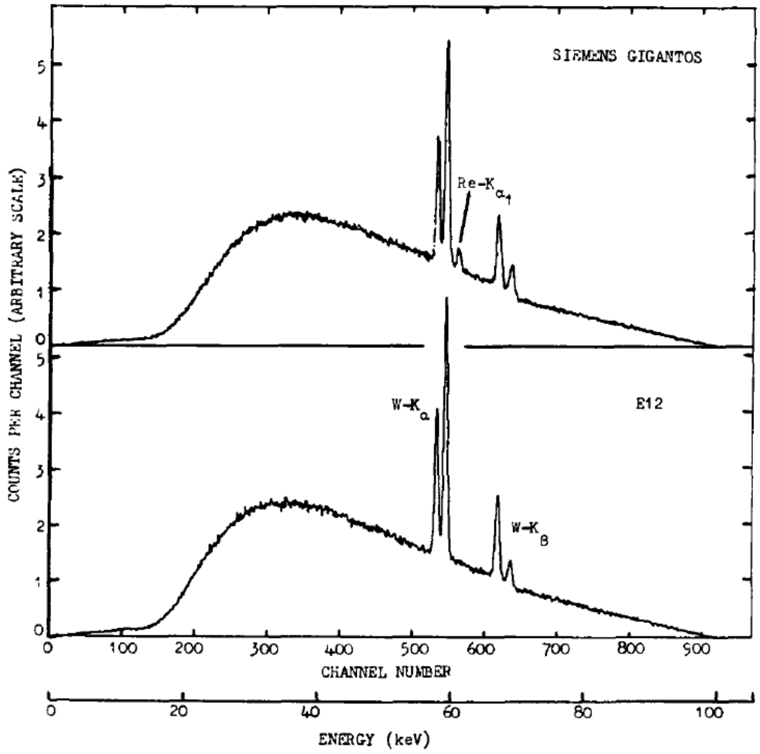


Figure 7: Illustrative examples of X-ray energy spectra, from two different tungsten targets bombarded with electrons accelerated with a peak voltage of 100 kV, 2 mm aluminium filter. The peaks in the spectra are the characteristic K-peaks of the tungsten targets, while the main body of the spectra originate from bremsstrahlung. [26]

A regular X-ray image is a 2D projection of the attenuation coefficients in the imaged 3D volume. In CT imaging several one-dimensional (1D) projections are acquired from transverse slices through the body. The acquisition is usually done performing a helical scan along the length of the patient, with a fan-shaped or cone-shaped X-ray beam, ensuring that enough information is acquired to calculate the attenuation coefficient in every individual voxel. The contrast in the resulting images is caused by the differences in attenuation coefficients between different tissues. Cortical bone is generally the most attenuating tissue, while lung tissue is the least attenuating. [16]

The calculated attenuation coefficients are commonly rescaled to the integer Hounsfield units (HU), according to the formula

$$HU_{\text{voxel}} = 1000 \cdot \frac{\mu_{\text{voxel}} - \mu_{\text{water}}}{\mu_{\text{water}}}, \tag{6}$$

where μ_{water} and μ_{voxel} are the attenuation coefficients for water and the voxel of interest (VOI), respectively. An example of a standard (low-dose) CT image given in HU can be seen in Figure 8. [16]

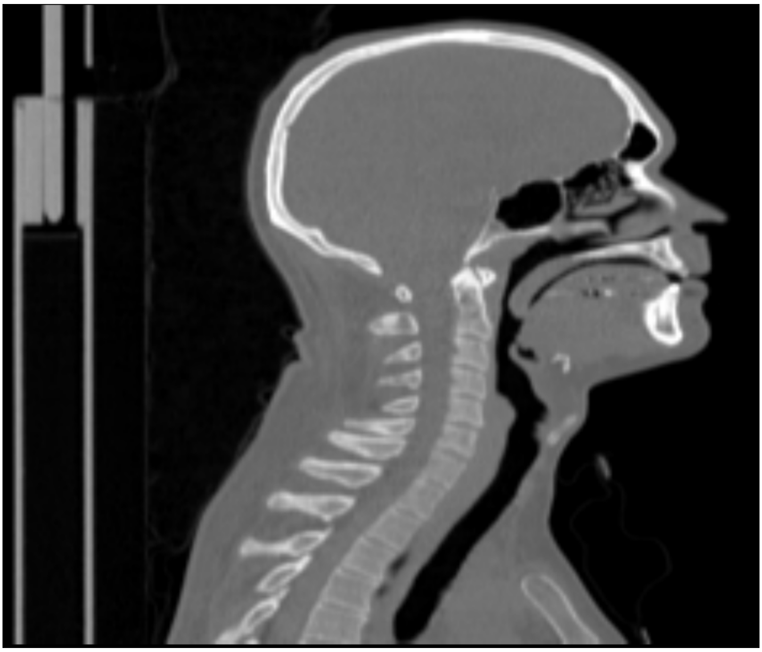


Figure 8: Example of a low-dose (as opposed to diagnostic) CT image slice. The image slice is a sagittal slice through the head and neck of a male patient. The patient bed is visible to the left. The brightest pixels in the tissue have the highest attenuating values, and are usually bone or metal. The darkest pixels are air, and the intermediate ones are soft-tissues. In this image the skull, jaws, and vertebrae are clearly visible, along with the trachea and sinuses. If the image seems stretched/compressed in a certain direction, this is a flawed property of the utilised image viewer. Patient ID: PETMR023.

2.3.1 Beam hardening

In CT imaging there is a desire to keep the radiation dose delivered to the patient as low as possible, while still acquiring images of sufficient quality, according to the principle of keeping the dose as low as reasonably achievable (ALARA). As mentioned, as an X-ray beam traverses the body it is attenuated, and the attenuating interactions have different probabilities for different photon energies. Specifically, after traversing tissue of some thickness, an X-ray beam will have a different spectrum of photon energies than the incident beam; there will be relatively fewer low-energy photons. This is known as beam hardening, and is an issue when the travel length of the beam becomes too large, since high-energy photons are less attenuated, and thus contribute less to image contrast. To compensate for beam hardening one can either increase the input energy, and thus the dose; or one can decrease the travel length of the X-ray beam through the body. One way of decreasing the travel length of the beam is to position the patient in a way optimal for imaging of the specific body part, for instance by positioning the arms above the head in an abdominal examination. [16]

2.4 MRI

MRI is an imaging modality utilising the nuclear resonance of hydrogen nuclei (protons) placed in an external magnetic field to generate structural and functional images[19]. This section will provide the minimum of information needed to understand the MRI-related parts touched upon in this report. First a short overview of the basic principles of MRI is given, before two types of sequences especially important in connection to PET AC are presented.

2.4.1 Basic principles

Protons (as all elementary particles) possess an intrinsic angular momentum known as *spin*, which, since the proton also has charge, gives rise to a magnetic dipole moment $\vec{\mu}$. When a proton is placed in a magnetic field $\vec{B}_0$, $\vec{\mu}$ will interact with the field and start to precess around the field direction, at a frequency known as the *Larmor frequency*. The Larmor frequency ω_0 is linearly dependent on the field strength $B_0 = |\vec{B}_0|$:

$$\omega_0 = \gamma B_0, \tag{7}$$

where γ is the gyromagnetic ratio. Water-bound hydrogen nuclei have a gyromagnetic ratio of $\bar{\gamma} \equiv \gamma/(2\pi) = 42.6$ MHz/T. Thus, hydrogen nuclei experiencing a magnetic field of 3 Tesla (T) will have a Larmor frequency of about 128 MHz, which is within the radiofrequency (RF) spectrum. [15] [19]

In a tissue voxel experiencing an external, homogeneous magnetic field, the individual magnetic moments will sum up to a net magnetisation $\vec{M}$ in the direction of the field. This net magnetisation vector can be manipulated by exciting the tissue with (Larmor frequency) RF pulses, which flip the vector a certain angle (flip angle) away from the direction of $\vec{B}_0$, depending on the strength and direction of the RF pulse. The transversal component of $\vec{M}$, $M_\perp$, will precess about the field direction (at the Larmor frequency) and can be detected by strategically placed coils in which it will induce time-varying currents. After excitation the signal will decay as $\vec{M}$ returns to equilibrium; $M_\perp$ decays and $M_\parallel$ (longitudinal component of $\vec{M}$) regrows. [19] [30]

The regrowth of $M_\parallel$ follows the equation

$$M_\parallel(t) = M_0 \left[1 - \mathrm{e}^{-t/T_1}\right] + K_1 \mathrm{e}^{-t/T_1}, \tag{8}$$

where K_1 is a constant depending on the initial conditions, $M_0 = |\vec{M}|$, and T_1 is the longitudinal relaxation time constant, which is tissue-dependent. The decay of $M_\perp$ follows the equation

$$M_\perp(t) = K_2 \mathrm{e}^{-t/T_2}, \tag{9}$$

where K_2 is a constant depending on initial conditions, and T_2 is the transversal relaxation time constant, which is tissue-dependent. $M_\perp$ is the only detectable part of $\vec{M}$ in regular MRI, and T_2 thus governs how fast the MRI signal irreversibly decays. Magnetic field inhomogeneities induce additional decay of the signal (dephasing of individual $\vec{\mu}$'s), determined by T_2', but this dephasing is largely constant over time (static) and thus reversible. [19] [30]

According to Equation (7) the Larmor frequency, and thus the frequency of the detected signal, depend on the magnetic field strength. By introducing linear magnetic field gradients in specific directions at specific times during excitation and detection, a spatial dependence can be imposed on the Larmor frequency. [19] [30]

The detected MRI signal consists of a weighted sum of all the Larmor frequencies of the different tissue types in the different voxels. The frequency content of the signal is acquired and saved in k-space (see Figure 9). In MR sequences a range of exciting RF pulses and space-encoding gradients are employed in order to fill k-space with sufficient amount of values (k-space mapping). When sufficiently mapped, the inverse FT of k-space will yield the image of the object. The lower frequencies determine the image intensity, while the higher frequencies determine the resolution. [14] [19] [30]

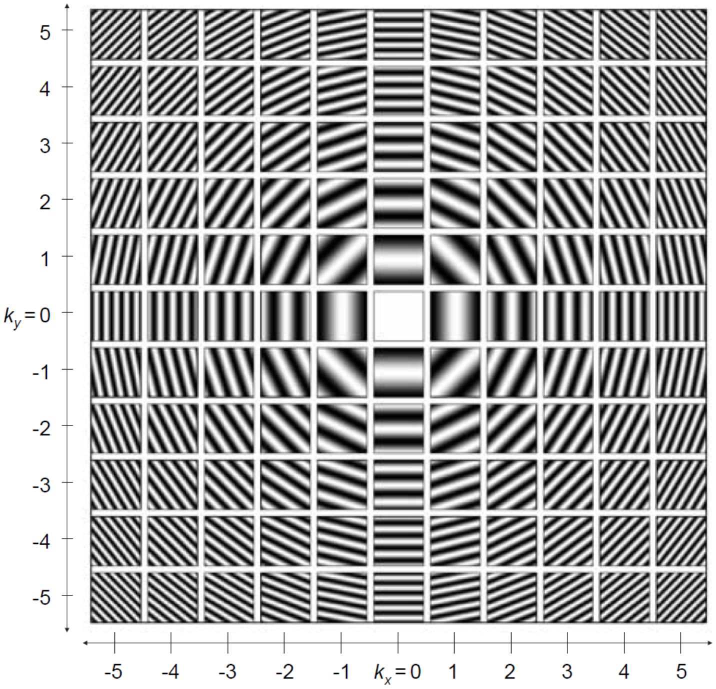
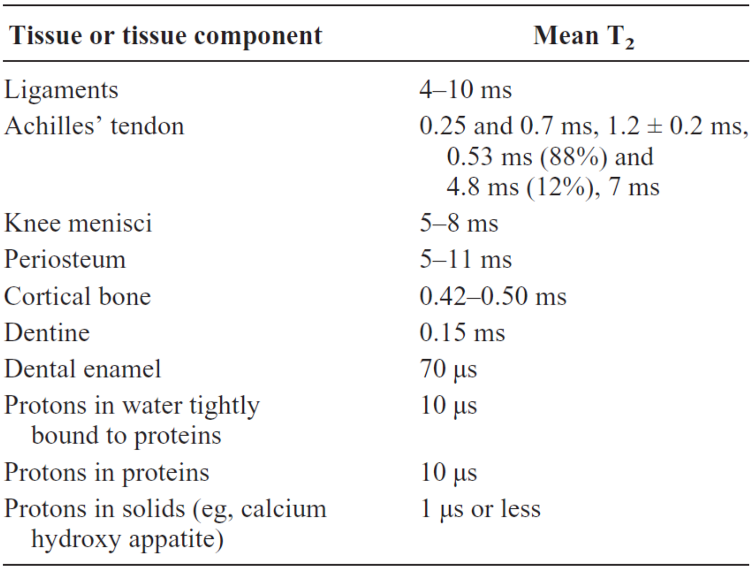


Figure 9: Illustration of 2D k-space and its corresponding spatial oscillations. Each coordinate in the illustration contains a square displaying the spatial oscillation corresponding to that specific point in k-space. The lowest spatial frequencies are found in the center, increasing with radial distance. The lower frequencies decide the image intensity, while the higher frequencies decide the resolution. [14]

2.4.2 UTE sequences

As mentioned above, the MRI signal decays exponentially as a function of the T_2 time constant. T_2 varies depending on the molecular environment of the proton, from a few hundred milliseconds to less than a microsecond, for different tissues and tissue components present in the body. An overview of tissues with short mean T_2 values can be found in Table 1. Cortical bone, with a mean T_2 of about 0.5 ms, is a large tissue class in the body that does not yield any significant signal in regular MR sequences, since its signal have long since decayed at the time of signal acquisition. [1] [37]

Table 1: Approximate mean T_2s of some short-T_2 tissues and tissue components, from adult clinical results and tissue sample results estimated for 1.5 T field strength. [37]



Tissue or tissue component	Mean T_2
Ligaments	4–10 ms
Achilles' tendon	0.25 and 0.7 ms, 1.2 ± 0.2 ms, 0.53 ms (88%) and 4.8 ms (12%), 7 ms
Knee menisci	5–8 ms
Periosteum	5–11 ms
Cortical bone	0.42–0.50 ms
Dentine	0.15 ms
Dental enamel	70 µs
Protons in water tightly bound to proteins	10 µs
Protons in proteins	10 µs
Protons in solids (eg, calcium hydroxy appatite)	1 µs or less

UTE sequences have been specifically designed to acquire signal from tissues with short T_2, in some cases specifically from cortical bone. The echo-time (T_E) of a sequence is the time between tissue excitation and signal acquisition. UTE sequences acquire signal immediately after RF excitation, before the signal from bone and other short-T_2 tissues have had time to decay beyond detection, often with $T_\mathrm{E} \sim 50$ µs. This acquired signal will consist of signals from both tissues with short and long T_2s. To separate the two signals, a second acquisition is performed when most of the signal from the short-T_2 tissues have fully decayed, when most of the remaining signal originate from long-T_2 tissues, as illustrated in Figure 10. Two images are reconstructed from the signals from the two different T_Es, and the latter is subtracted from the former to suppress the long-T_2 signal. [1] [37]

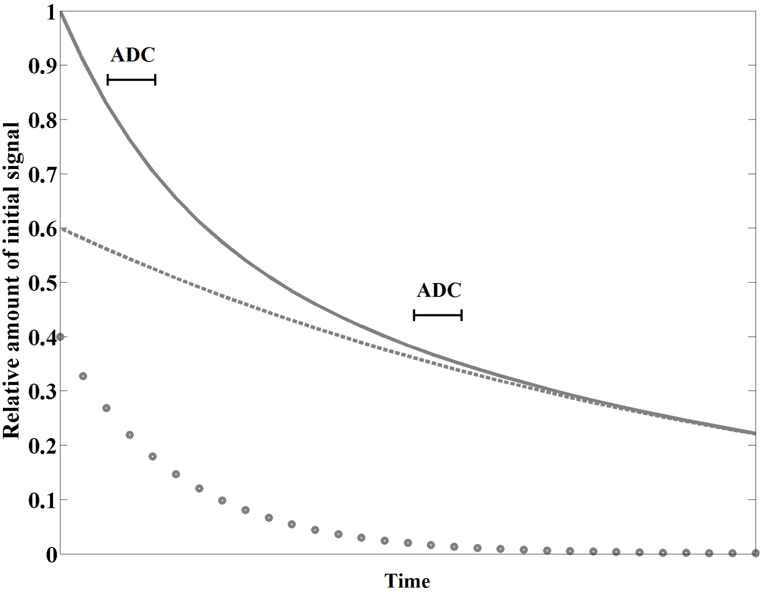


Figure 10: Illustration of the decay of a signal consisting of signals with different T_2s. The dotted line is the signal from tissue with a short T_2, and the dashed line is the signal from tissue with a long T_2, while the continuous line is the total signal. The ADC (analogue-to-digital converter) sections indicate the acquisition times of the two different images in a UTE sequence.

For the short-T_2 acquisition radial k-space mapping is performed, from the center and outwards, to maximise the amplitude of the intensity-determining lower frequencies. The higher frequencies will however inevitably be sampled when the signal amplitude is very low, leading to inherent poor resolution. [1] [37]

Since every aspect of a UTE sequence occur on such a small timescale compared to regular MR sequences, there are several challenges related to UTE imaging in addition to those normally encountered. The short-T_2 signal decays so fast that the signal acquisition must be performed immediately after excitation, during ramp-up of the magnetic gradients, which themselves must have a high slew rate and maximum amplitude. The change in the magnetic field resulting from the ramp-up of the gradients induces eddy currents in the gradient coils and other conductors in the vicinity. These eddy currents produce a magnetic field that opposes the original change in the field, leading to a deviation between the desired and actual k-space trajectories. This, along with shifts in k-space trajectories due to system latencies, degrade the reconstructed image. The k-space trajectories can be corrected and shifted back into the correct position, either by acquiring the actual k-space trajectories through gradient monitoring, or by simpler calculations based on the expected symmetry of radially sampled signal, as well as other methods not described here. [1] [7] [37]

2.4.3 Dixon sequences

The bulk signal in MRI originates from protons bound in water molecules, but there is also a substantial contribution from lipid-bound (fat-bound) protons. Water-bound and lipid-bound protons have slightly different chemical environments, resulting in slightly different experienced magnetic fields, and thus in slightly different precession frequencies. As mentioned above, the spatial information in the MR signal depends on induced spatial dependencies in the precession frequency. This means that if the fat-signal is not separated from the water-signal, the fat-signal may be assigned a shifted position in the image relative to the actual spatial origin of the signal. [9] [25]

The (complex) MRI signal and its contributions from the fat and water can be modelled as illustrated in Figure 11; as a sum of the signal from fat (F) and the signal from water (W), with a phase difference (α) between them. In addition there are error phases caused by magnetic field inhomogeneities (ϕ) and system imperfections (ϕ_0):

$$S(x,y,z) = [W(x,y,z) + F(x,y,z)\mathrm{e}^{\mathrm{i}\alpha}]\mathrm{e}^{\mathrm{i}[\phi(x,y,z)+\phi_0(x,y,z)]}. \tag{10}$$

α arises from the difference in Larmor frequency, and can be manipulated through choice of T_E and other timing properties. The error phase due to field inhomogeneities can be corrected for in various ways not described here (phase unwrapping, mapping of shift smoothness). [9] [25]

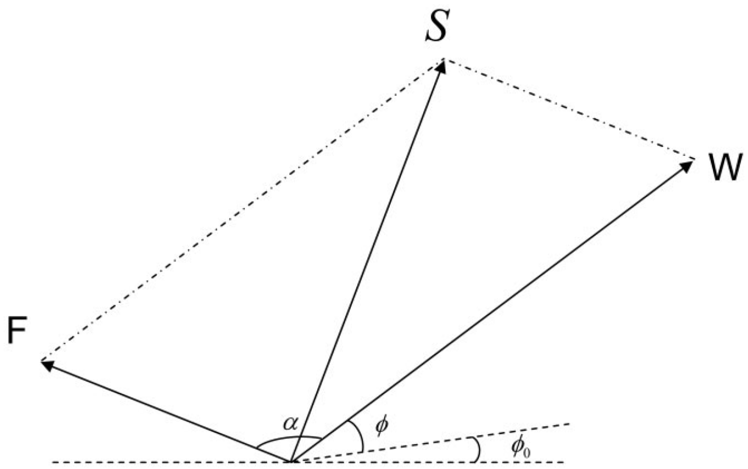


Figure 11: Illustration of the complex MRI signal (S) and its contributions from fat (F) and water (W). α is the phase difference between the fat and water signals, ϕ is an error phase caused by magnetic field inhomogeneities, while ϕ_0 is an error phase due to system imperfections. [25]

By sampling signals at two different α's, at 0 and π radians, one gets the following signal equations, respectively:

$$S_0 = (W + F)\mathrm{e}^{\mathrm{i}\phi_0}, \tag{11}$$

and

$$S_\pi = (W - F)\mathrm{e}^{\mathrm{i}[\phi+\phi_0]}, \tag{12}$$

where the spatial dependency (x, y, z) is omitted for simplicity. When $\phi = 0$ the water and fat signals can be calculated directly:

$$2W = |S_0 + S_\pi|, \tag{13}$$

$$2F = |S_0 - S_\pi|. \tag{14}$$

Usually $\phi \neq 0$, but as mentioned above this can be corrected for. A sequence acquiring two images to separate fat and water such as the one described here is known as a two-point Dixon sequence, named after the man first proposing the technique, W. T. Dixon. Dixon sequences are able to separate fat and water signals wuite successfully, creating water-only images and fat-only images, in addition to the acquired in-phase and out-of-phase images. [9] [25]

2.5 PET AC

As described above, the radiation from the PET tracer inside the body is attenuated to different degrees by different tissues; different tissues have different μ-values. To accurately quantify the amount of tracer in a specific region of interest (ROI), for instance to assess the uptake in a tumour and thus assess its properties, the PET raw data must be corrected for this attenuation. Without AC the activity values in the resulting image will be underestimated, as seen in Figure 12. This is especially evident for LORs passing through a lot of or/and highly attenuating material, such as in the central parts of the imaged object, or close to bone structures. [27]

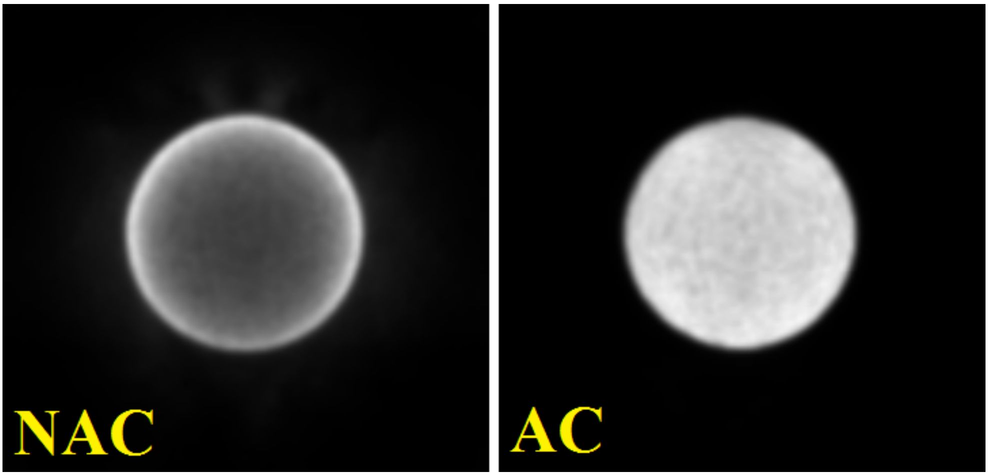


Figure 12: A comparison of two PET images of a uniform-activity phantom, both reconstructed from the same PET raw data, one without AC (NAC) and one with AC (AC). In the NAC image the activity seems to be concentrated in the edge of the phantom, since most of the detected photons originate from this area. The AC image displays the activity distribution more correctly.

The detected PET signal can be modelled in the following way. Let

f be the activity distribution in a cross-section with the attenuation distribution μ, while $L(\phi, s)$ is a straight line $x \cos \phi + y \sin \phi = s$ in an xy-plane. The detected PET signal can then be modelled as

$$g(\phi, s) = \mathrm{e}^{-\int_{L(\phi,s)} \mu(x)\mathrm{d}x} \cdot \int_{L(\phi,s)} f(x)\mathrm{d}x, \tag{15}$$

which in its discrete and noisy form is what is stored in each pixel in the sinogram files described above. When μ is known, the detected activity can be corrected for attenuation:

$$\int_{L(\phi,s)} f(x)\mathrm{d}x = \mathrm{e}^{\int_{L(\phi,s)} \mu(x)\mathrm{d}x} \cdot g(\phi, s), \tag{16}$$

where

$$\mathrm{e}^{\int_{L(\phi,s)} \mu(x)\mathrm{d}x}$$

is the attenuation correction factor (ACF) of the specific LOR along $L(\phi, s)$. The attenuation distribution can be imaged by a μ-map, and when it is known the ACF of every LOR can be calculated and saved in an AC sinogram through the same kind of forward projection used when creating the raw data sinogram. As shown by Equation (16), attenuation correction can then be performed by simply multiplying the raw data sinogram with the AC sinogram. [27] [31]

Accurate μ-maps are crucial for the quality of the PET images. There are several ways of acquiring the needed μ-maps, including transmission scans, CT scans, and a range of MR-based methods. [27]

2.5.1 Transmission scans

In traditional stand-alone PET scanners, μ-maps were acquired by performing a transmission scan of the patient either before, after, or during the PET scan; and comparing it with a blank scan (a scan performed without any patient present in the scanner). In a typical transmission scan one or several rod sources (usually ^{68}Ge, decays to ^{68}Ga, a β^+-emitter) are rotated about the patient along the edge of the transaxial PET FOV. The detected activity in detectors collinear with the rod sources is used as the basis for calculating the μ-map. Transmission scans can be quite lengthy, and also produce noisy μ-maps. [27]

2.5.2 CT-based AC

With the successful combination of PET and CT into fully integrated PET/CT scanners, the possibility for efficiently using CT data for creating PET μ-maps became a reality. As described above, CT images are essentially μ-maps, but for a different energy range than that of PET annihilation photons. In general, a simple energy scaling is not sufficient to translate $\mu(E_{\text{X-ray}})$ to $\mu(E_\gamma)$ based on a measurement of $\mu(E_{\text{X-ray}})$ only. However, when limiting the attenuating material to biological tissue, it has been shown, amongst others by Carney et al.[8], that a bilinear scaling is sufficiently accurate (see Figure 15). The basic assumption is that the CT image can be segmented into two different tissue mixtures based on HU-values. Voxels of HU-values below a certain breakpoint (BP) is considered a mixture of air and water, while voxels above the BP is considered a mixture of bone and water. Two different linear scaling functions are applied, one for each side of the breakpoint, translating the HU-values to 511 keV μ-values. An example of such a scaling function is plotted in Figure 15 in Part II. [27] [43]

Using CT data to generate μ-maps have several advantages over transmission scans. The CT data is virtually noiseless compared to the transmission scan, and provides a much better resolution. Also, since CT scans are routinely acquired as a part of any PET/CT examination, there is almost no time added due to the additional AC scans. There is however an issue with movement. CT scans are acquired so fast that patient movement such as breathing usually is not a problem. PET scans however last for several minutes for each bed position, and are thus sensitive to breathing and other patient movements. This leads to a mismatch between the μ-map and the dynamic shape of the tissue from which the PET data is collected. A way of correcting for this is to gate the PET signal based on monitoring of for instance patient respiration (respiratory gating). If the PET signal is to be gated, a longer scan is necessary to compensate for the discarded data. [27]

2.5.3 MR-based AC

As mentioned in the beginning of this part, scanners combining PET with MRI in a fully integrated PET/MR system recently became commercially available. How to acquire μ-maps is one of the great challenges in PET/MR, and has been subject to intensive research since the idea of combining PET and MR in a single scanner first was introduced. Especially difficult is the imaging of bone in MR, which is a tissue notorious for not contributing any signal in ordinary MR sequences. [6] [17] [21]

The signal in MRI depends on proton densities, and there is no direct correlation between proton (^{1}H-nuclei) density and electron density, which is the crucial property in PET attenuation. There is thus no way of directly scaling MR image intensities to acquire an attenuation map, as can be done with CT images. The acquisition of traditional transmission or CT scans is largely out of the question in PET/MR, due to spatial constraints and the need for components unaffected by strong magnetic fields. In addition the radiation dose should be kept ALARA. [6] [17] [21]

Keereman et al.[21] suggested the following requirements for an ideal MR-based AC (MRAC) method.

- The acquired/derived attenuation map should be accurate and detailed enough to distinguish between air, lung, soft-tissue, spongious bone and cortical bone, as the improper discrimination of these has been shown to result in large quantification errors, especially the mismapping of lung tissue.

- Due to variations between patients in attenuating properties of lung tissue in particular, patient-specific lung tissue attenuation coefficients should be derived.

- The limited axial FOV of MR systems often leads to the exclusion of the arms from the image, and this issue must be solved.

- Patient bed and MR coils should be included in the attenuation maps, which may be a challenge regarding flexible coils.

- The acquisition of the attenuation map must not be time-consuming, as time is the largest constraint in MRI; and PET and MR image quality must be unaffected.

- If any additional system components are to be installed post system installation, they must be small enough to fit inside the bore of the scanner, or be placed outside the bore, as well as not being unreasonably impractical to implement.

Although there are several different strategies for performing MRAC, no current method fulfils all these criteria, including template, or machine-learning-based methods; transmission-based methods; and segmentation-based methods.

Template/atlas methods are based on the acquisition of a conventional MR image set, to which a template, based on general patient images, is non-linearly registered and used as an attenuation map. The template may be based on PET transmission scans or CT images. In some proposed methods the template is a single image set averaged from several individuals. In other cases the template consists of several image sets which are each registered to the patient MR image set, and the subsequent attenuation map is derived from a weighted sum of these. Machine-learning-based methods may be based on template/atlas approaches, additionally employing pattern recognition algorithms to construct pseudo-CTs which are subsequently scaled to construct attenuation maps. Other methods are based on MR images only. [6] [21]

Mollet et al.[28] proposed a method utilising a form of transmission scan to acquire attenuation maps, only suitable in TOF PET systems (at present not available in the mMR). By inserting a static annular-shaped transmission source in the FOV of the PET system, transmission scans can be acquired during regular emission scans. The coincident photons originating from the transmission source are distinguished from the emission coincidences based on the TOF capability of the system. The reconstructed attenuation map was segmented to successfully eliminate noise. The main disadvantage of this type of MRAC is that the increased activity in the PET FOV degrades the count rate performance of the system.

2.5.4 Segmentation-based MRAC

Segmentation-based methods are based on deriving an attenuation map largely from MR sequences alone. One of the great challenges when attempting this is that in conventional MR images, both bone and lung tissues yield very little signal, due to their short (T_2) decay times. This leads to very little contrast between bone, lung tissue and air, materials with widely differing attenuating properties, with bone being the most attenuating tissue in the human body. In brain imaging in particular, where the amount of bone is relatively high, and where there are complex mixtures of soft-tissue, air and bone in the post-nasal cavities, distinguishing bone from other tissues is of vital importance. Another example of tissues differing in attenuating properties but yielding similar signal intensities in conventional MR images is fat and soft-tissue. [6] [21]

The earliest segmentation-based methods proposed were based on segmenting conventional MR images into different tissue-classes and assigning the segmented areas a predefined attenuation coefficient. Additional anatomical information or use of pattern recognition methods improved the attenuation maps, but the general results were poor compared to CT data[6] [21]. By introducing additional MR sequences capable of acquiring signal from bone and lung tissue (UTE sequences), or capable of distinguishing fat and soft-tissue from each other (Dixon sequences) , improved attenuation maps can be acquired. [6] [21]

Dixon-based MRAC methods are able to separate fat from water, and can thus segment tissues with differing contents of these, such as adipose tissue and soft-tissue. Dixon sequences are generally very fast, and thus clinically applicable in whole-body imaging. They are however not capable of generating signal from bone, and are thus not suitable for creating accurate head μ-maps (despite this they are still utilised in brain imaging in lack of better alternatives). [6] [9] [21]

UTE-based MRAC methods are able to discriminate between bone and other tissues, but do not work perfectly, due to the technical difficulties and increased sensitivity to inhomogeneities and frequency off-sets when operating on UTE-relevant time-scales. The post-nasal cavities present a particular difficult area, as they are riddled with bone/air interfaces. UTE-sequences are generally too time consuming to be employed in whole-body imaging, but are routinely acquired in PET/MR brain imaging. [6] [9] [21]

2.6 Image registration

A substantial task of this study involved image registration between images acquired not only from different scanners and modalities, but also imaged with the patient in different positions. To increase the appreciation of the complexity of this task, a short introduction to image (co-)registration is given below.

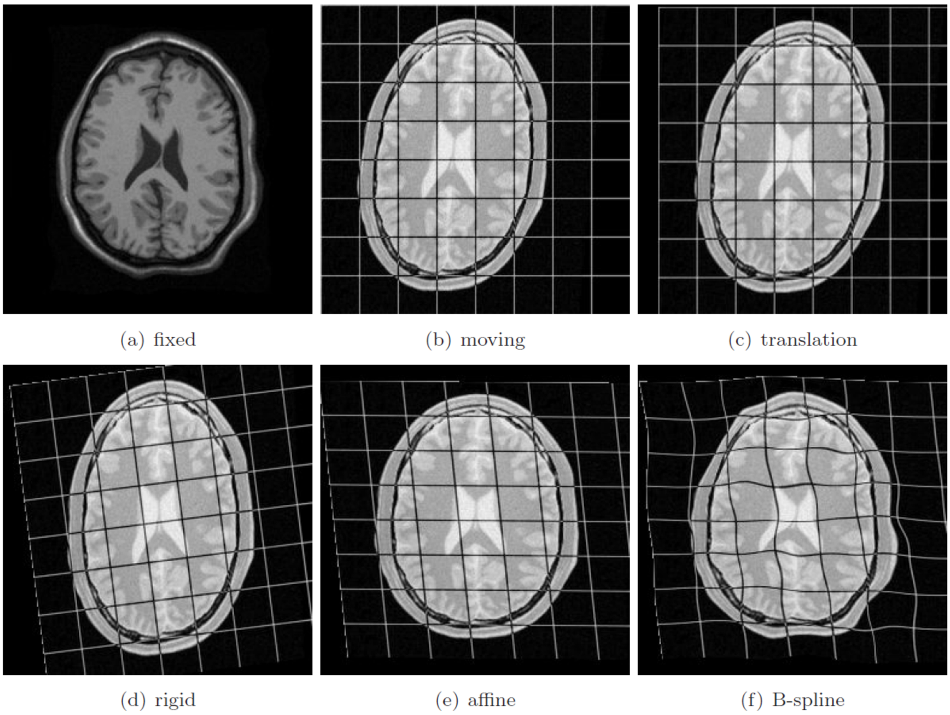
Medical images are created on a range of different scanners, in a range of different modalities such as CT, MRI, PET, SPECT, ultrasound, etcetera. The images are stored along with information on how the image coordinate system (the image domain) relate to some physical coordinate system defined on each different scanner. Images from different modalities acquired on the same (hybrid) scanner are automatically co-registered through hardware techniques. However, when images from different scanners, regardless of modality, are to be co-registered, software methods have to be employed. [23]

The objective in image registration is to find a one-to-one voxel mapping between the image domains of two different images (of equal number of dimensions). In other words, the objective is to find a transformation that deforms one image, the *moving image*, so that it aligns with another image, the *fixed image*, such that there exists a defined relation between the image domain of the moving image and the physical domain of the fixed image. This transformation is defined as a mapping from the fixed image to the moving image. [23]

The alignment's quality is defined by a similarity measure, such as the sum of squared differences (SSD) or mutual information (MI). The latter is very useful when co-registering images of different modalities, such as CT and MRI. The optimum image transformation is acquired by maximising the similarity measure, while minimising a penalty term which constrains the transformation regarding compression and deformation. [23]

There are several kinds of transformations that can be performed, with different numbers of degrees of freedom. The simplest transform is a translation transform, where the moving image can be translated in all image directions. In rigid transformations the moving image can in addition be rotated about a center of rotation. In similarity transforms the moving image can also be scaled isotropically. By adding shear possibilities for shearing the image the transformation becomes affine. [23]

The transformation methods mentioned until now are all rigid transformations with a restricted amount of degrees of freedom. Non-rigid transformations such as B-spline and thin-plate spline transformations may have tens of thousands of degrees of freedom, depending on number of image dimensions, and image size and resolution, since they allow deformation, compression, bending, etcetera, of the moving image. Examples of an image transformed with different methods can be seen in Figure 13. [23]


(a) fixed (b) moving (c) translation

(d) rigid (e) affine (f) B-spline

Figure 13: Examples of different image transformations, all performed with the same moving and fixed images. [23]

32

Part II

Materials and methods

1 Patients

The patients included in the study suffered from lymphoma or lung cancer. Their diagnostic routine included a PET/CT examination, which involved an injection of the PET pharmaceutical ^{18}F-marked Fluorodeoxyglucose (FDG) 60 minutes prior to the examination. Before the FDG injection, the patients were informed about the PET/MR research study and the extra PET/MR examination, and volunteers signed a written consent. The PET/MR examination did not necessitate any additional injection of FDG. The research project was approved by the regional ethics committee (REK).

2 Image acquisition

2.1 Scanners

Two multi-modality scanners were used to acquire data in this study: The biograph mCT (serial number 11094) PET/CT scanner running software version syngo MI.PET/CT 2012A was used to acquire the necessary CT images; while the biograph mMR (serial number 51033) PET/MR scanner running software version syngo MR VB20P was used for both MR and PET image acquisition. Both scanners were delivered by Siemens Healthcare (Erlangen, Germany), and installed in autumn 2013.

2.2 Examination protocols

The total examination flow is illustrated in Figure 14. The patient was injected with FDG 60 minutes prior to the PET/CT examination, and thus about 120 min ± 15 min prior to the PET/MR examination, depending on the length of the PET/CT examination. The injected dose was 4 MBq/kg, and ranged from a minimum total injected dose of 150 MBq to a maximum total injected dose of 530 MBq, depending on patient size. Directly after the PET/CT examination the patient was moved to the mMR and the PET/MR examination was performed.

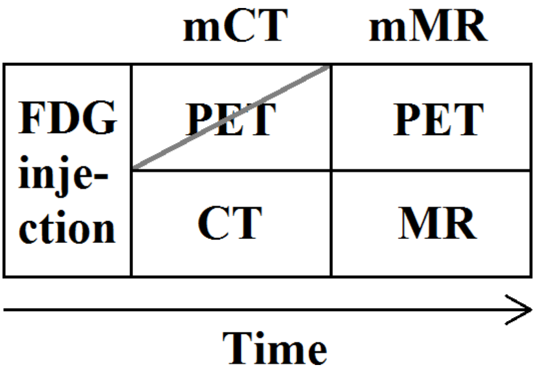


Figure 14: Overview of the examination flow from FDG injection to PET/MR examination. FDG was injected 60 minutes prior to the PET/CT examination. The PET data from the mCT was not utilised in this study.

2.2.1 PET/CT

The PET/CT examination consisted of a single, low-dose CT-scan; a multi-bed-position PET scan; and in some cases also a diagnostic CT scan of parts of the body. All scans were performed with the patient lying on the back with the arms raised above the head to avoid beam hardening artefacts in the thorax/abdomen. Only the low-dose CT scan was relevant in this study(scan direction: caudocranial, tube current-exposure time product: 40 mAs, peak tube voltage: 120 kV, slice (width(coll): 3.0 mm / 64x0.6, rotation time: 0.5 s, pitch: 0.95). The low-dose scan was performed for the purpose of acquiring an AC-CT image utilised for the AC of PET images. The PET data/images from the PET/CT examination was *not* used in this study.

2.2.2 PET/MR

The PET/MR examination consisted of a multi-bed-position (one to five bed positions) PET scan; in addition to a range of MR sequences for each bed-position. The choice of MR sequences varied depending on the bed position, patient, and disease. The examination was performed with the patient lying on the back with the arms positioned alongside the side of the body. The MR sequences relevant to this study was a Dixon sequence (T_{R}: 3.60 ms, T_{E}: 1.23 ms (2.46 ms), FOV: 328x500 mm^2, T_{A}: 17.95 s, flip angle: 10 degrees, slice thickness: 3.12 mm), and a UTE sequence (T_{R}: 11.94 ms, T_{E}: 0.07 ms (2.46 ms), FOV: 300x300 mm^2, T_{A}: 99 s, flip angle: 10 degrees, slice thickness: 1.56 mm). The patients wore hearing protection covering their ears to protect from operation noise.

The Dixon scans resulted in four different images (in-phase, opposing-phase, fat-only, water-only), from which a whole-body μ-map was derived; segmenting air, lung tissue, fat, and soft-tissue, with predefined LAC values of 0 cm^{-1}, 0.0224 cm^{-1}, 0.0854 cm^{-1}, and 0.1000 cm^{-1}, respectively. The UTE scans resulted in two different images, one from each echo time, from which a head μ-map was derived; segmenting air, soft-tissue, and bone, with predefined LAC values of 0 cm^{-1}, 0.1000 cm^{-1}, and 0.1510 cm^{-1}, respectively. Both μ-maps, the in-phase Dixon image, and the second-echo UTE image were utilised in this study.

The previous version of the UTE-based AC method (VB18P) was based solely on the two images acquired from the two different T_{E}s. The new version (VB20P) utilised in this study additionally employs streaking artefact reduction, gradient delay (eddy current) correction and template-based segmentation.

2.3 PET image reconstruction

The PET reconstruction software `RetroRecon` on the mMR was used for the necessary image reconstructions with different μ-maps and reconstruction methods. All image reconstructions were performed with the following `RetroRecon` settings: 3 iterations, 344x344 output resolution, and Gaussian filter of 4.0 mm full width at half maximum (FWHM). All images were reconstructed twice for each different μ-map: once using regular 3D-iterative reconstruction, and once using the recently added option of PSF reconstruction.

3 Image processing

3.1 Hardware and software

The image handling and modification was performed on a laptop computer with a 64-bit quad-core Intel Core i7-4750HQ CPU, clock speed 2.0 GHz, 8 Gb DDR3 RAM memory, running 64-bit Windows 8.1 pro. The software used for image processing was `MATLAB R2014a` (8.3.0.532) 64-bit, `MRIcron`[1] (6.6.2013), `dcm2nii`[2] (6.6.2013), `elastix` v4.7 for windows 64-bit, and its sister-program `transformix` v4.7 for windows 64-bit.

3.1.1 `MRIcron` and `dcm2nii`

`MRIcron` is a simple image viewing software developed for medical images, and was used for viewing overlaid images and for manual editing of image masks. `dcm2nii` is an image converter specialised at converting DICOM (Digital Imaging and Communications in Medicine) images to NIFTI image format (Neuroimaging Informatics Technology Initiative), and was used to convert the output images from the scanner (DICOM format) to the simpler, and in this study's case more flexible 3D NIFTI format (.nii). The NIFTI images were handled using the `MATLAB` functions `load_nii.m` and `save_nii.m`, both written and published on MATLAB Central File Exchange by Jimmy Shen[3].

3.1.2 Image formats - DICOM vs NIFTI

When discussing medical images, an ‘image’ refers to a 3D image volume saved along with a header containing important and informative metadata relevant to the image. When referring to 2D medical images, the term ‘image slice’, or just ‘slice’, is used.

The output images from the biograph scanners, from both MR, CT and PET scans, are saved in DICOM format (.ima). Images saved in DICOM format are saved in directories, with one file for each image slice. Each DICOM file contains the raw image data of a slice, along

---

[1] http://www.mccauslandcenter.sc.edu/mricro/mricron/

[2] http://www.mccauslandcenter.sc.edu/mricro/mricron/dcm2nii.html

[3] Rotman Research Institute, Canada

with a large amount of informative and vital, series-specific and slice-specific metadata. This metadata is saved as tags, and the DICOM standard defines a large number of public tags which are well described, and thus decoded by most software able to read DICOM images. Each tag contains a tag name, a group number, and an element number, along with information on how the actual data in the tag is saved and thus how this can be properly decoded.

Saved along with the public tags, there is a large amount of manufacturer-specific private tags. These private tags are not necessarily as well-described as the public tags, as they are defined in manufacturer DICOM conformance statements, which do not contain very much information. The private tags are of great importance for enabling manufacturer-specific software, such as that on the scanners, to properly handle the images.

Along with patient information, the DICOM tags contain information on a range of different image properties, such as the physical size of the voxels, the physical position of the image, the spacing between slices, etcetera. The preservation of the metadata is vital to maintain the integrity of the DICOM file, and crucial when exporting an image from the scanner, editing it, and then importing it back into the scanner. The most important tag in this study was the ‘Pixel Data’-tag, which contains the raw image of the DICOM file as a binary stream.

DICOM directories are more cumbersome to handle than a 3D image format containing the entire image in a single file. The NIFTI image format is well-suited for saving entire 3D image volumes in single files. Just like in the DICOM format, the NIFTI files contain important metadata, but less detailed. A conversion from NIFTI to DICOM is thus not as straightforward as a conversion from DICOM to NIFTI. NIFTI is still a versatile and popular image format, used as input to a range of different image software, including the image registration program utilised in this study; `elastix`.

3.1.3 `elastix`

`elastix` is a command-line interface, image registration program taking a minimum of four inputs: a fixed image, a moving image, an output directory, and a registration parameter file. The registration parameter file is a plain-text file (.txt) containing parameter-value pairs defining all necessary registration parameters, such as the transform method, the metric, which resolutions to use, how many samples to draw, etcetera. A few examples of registration parameter files used in this study can be found in Appendix B. `elastix` supports multiple image file formats, but not DICOM directories, so all registrations were performed on images in 3D NIFTI format.

The output from `elastix` consists of detailed log files, a transform parameter file, and (optionally) the transformed image. The transform parameter file is a plain-text file which can be used as input, along with an image and an output directory, to elastix' sister-program `transformix`, which transforms the image according to the input transformation, returning the transformed image as well as a log file.

Optional inputs to `elastix` include: initial transform parameter files, additional registration parameter files, a fixed-image mask, and a moving-image mask. A mask is a binary image used to limit the area in the image from which the samples can be drawn. The mask has pixel value one in areas where a drawn sample can be accepted, and zero where it is to be discarded. The use of registration masks and their creation have been of great importance in this study, both for image registration and image fusion when creating modified μ-maps. How this is the case will be explained in the results.

3.2 Conversion of HU to PET LAC

As described in the theory section, CT images provide excellent maps of the electron density distribution and thus also of linear attenuation coefficients, but they have to be converted to fit the PET photon energy range. The conversion method developed by Carney et al.[8] is visualised in Figure 15. As shown, the conversion function is bilinear, with different slopes on each side of the breakpoint. The coefficients defining the slopes are shown in Table 2. The peak tube voltage relevant in this study was 120 keV.

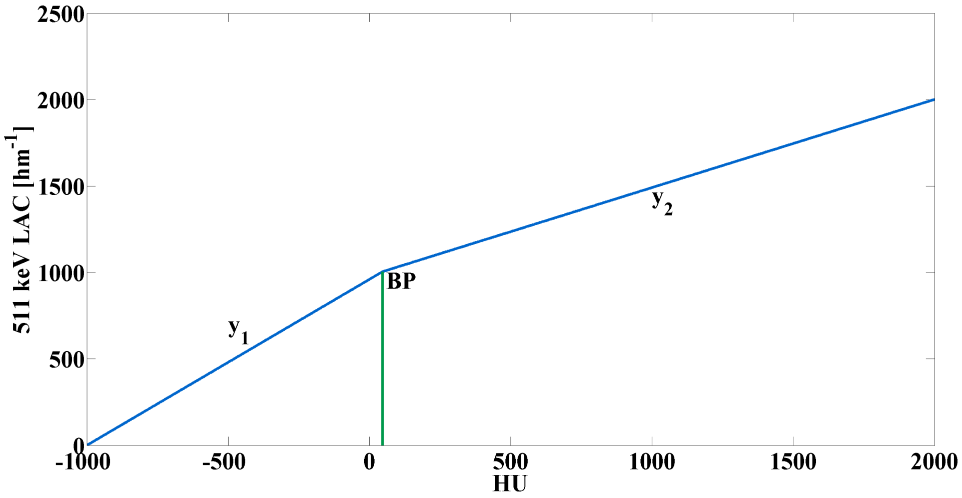


Figure 15: Example of the bilinear conversion function between HU and 511 kev LACs described by Carney et al.[8]. The peak voltage for which the plotted function applies is 120 kV, with a breakpoint at 47 HU. $y_1 = 9.6e\text{-}05(HU + 1000)$ *cm*$^{-1}$, $y_2 = a(HU + 1000) + b$. *The values for a and b are given in Table 2.*

Table 2: Overview of the slope-defining coefficients a and b in the HU to PET LAC conversion method described by Carney et al.[8]. BP is the break point, peak voltage is the peak voltage of the X-ray tube in the CT scanner. The slopes are shown in Figure 15.

Peak voltage [kV]	a [10^{-5} cm^{-1}]	b [10^{-2} cm^{-1}]	BP [HU]
80	3.64	6.26	50
100	4.43	5.44	52
110	4.92	4.88	43
120	5.10	4.71	47
130	5.51	4.24	37
140	5.65	4.08	30

4 µ-map creation

The main objective of this study was to assess the performance of the MR-based AC methods compared with the 'gold standard' of CT-based AC. One of the prerequisites for this to be achieved, was to develop methods for creating mMR-compatible, CT-based µ-maps, both for head and for whole-body. Developing these methods was by far the most substantial task of this study. These methods will thus be described in Part III, and only a minimal description of the resulting µ-maps will be given here, since this is necessary to enable the reader to make sense of the remaining sections of this part.

A total of three different types of µ-maps were created: one for whole-body, two for head. The whole-body µ-map and one of the head µ-maps were created from both CT and MR data (CTMR µ-map), while the other head µ-map was created using CT data only (CT µ-map). The whole-body µ-map had to be create from both CT and MR data due to the differing positions of the arms between the two examinations. The neck posture also differed between the two scanners, and thus two different µ-maps were created for the head, to assess the impact inaccurate µ-map data in the neck region might have on the brain region.

As was mentioned in Section 2.3, the PET data was reconstructed twice for each µ-map; once for each reconstruction method. For the head PET scans this resulted in a total of six reconstructions; twice for the original MR µ-map, twice for the CT µ-map, and twice for the CTMR µ-map. For the whole-body PET scans the result was a total of four reconstructions; twice for the original MR µ-map, and twice for the CTMR µ-map.

5 Head µ-map bone volume

The head µ-maps (MR, CT, CTMR) were compared regarding the amount and localisation of bone. The bone voxels in the MR µ-maps all have a value of 0.1510 cm^{-1}, and thus the bone volume in the MR µ-maps was easily calculated by extracting every voxel assigned this value. The threshold for bone in the CT µ-maps was set to 300 HU = 0.1146 cm^{-1}, as this was determined to be an optimal value by Delso et al.[11]. According to the work of de Oliveira et al. [32] HU $>$ 300 correspond to dense trabecular bone and cortical bone, while HU $<$ 300 correspond to low-density trabecular bone.

The amount of bone was calculated for the entire head μ-map, and for the cranial region of the μ-map. The cranial region was defined as all the transaxial image slices containing cranial bone, and thus also included parts of the lower jaw as well as the cranium. The border between the cranial region and the neck was set manually by identifying the lowest transaxial image slice containing cranial bone. The relative difference in bone volume between the different μ-maps was calculated and tested for significance, as described in Section 7.

In addition, the overlap between cranial bone in the MR μ-maps and the CT(MR) μ-maps was calculated. This was done by creating a binary mask for each different μ-map, covering all bone voxels with value one, zero elsewhere. The bone mask of the MR μ-map was subtracted from the bone mask of the CT μ-map, and the resulting images would contain values of -1, 0, and 1. -1 corresponded to bone voxels unique to the MR μ-map, while 1 corresponded to bone voxels unique to the CT μ-map.

6 ROIs and SUV measurements

To quantitatively compare intra-patient PET images, several ROIs were placed, in the brain, liver and aorta in each patient. For each ROI, SUV_{BW} and SUV_{LBM} was calculated for every voxel, according to Equation (1) and (2), respectively. For each ROI the minimum, maximum, and mean values was acquired from the different SUV calculations. The minimum and maximum values are very sensitive to noise, and thus the less noise-sensitive mean value was chosen for further analysis.

All ROIs set in the PET images were approximately spherical in shape, and were set in the exact same position in every intra-patient PET image, based on coordinates in the corresponding MR images. The MR image in which the coordinates were set was first registered to the PET images (translational registration) such as to acquire the same resolution and FOV, and thus the same image coordinate system, as the PET images.

Two ROIs were placed in the whole-body images (Figure 16), in concordance with the guidelines from a nuclear physician at St. Olavs Hospital. One was placed in the aorta, covering 123 voxels (1.1 ml), in such a way as to ensure that the ROI did not include parts of the vessel wall. The other ROI was placed in the liver, covering 1419 voxels (12.5 ml), in an as homogeneous area of the liver as possible.

Twelve ROIs were placed in the brain (Figure 17), six in each hemisphere. They were placed in the following regions: frontal, temporal, parietal, and occipital lobes; cerebellum; and thalamus. All ROIs covered 515 voxels (4.6 ml), except those of the thalamus, which only covered 123 voxels due to size constraints.

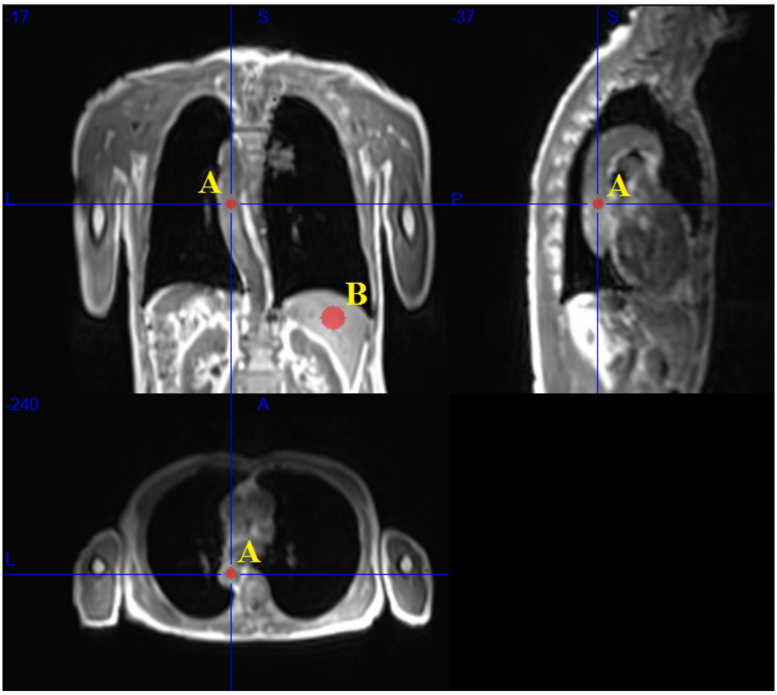


Figure 16: Example of typical ROI placements in the aorta(A) and liver(B), displayed as overlays on an in-phase Dixon image. Both ROIs have a spherical shape. The aorta ROI is placed so that it does not include any part of the vessel walls. The liver ROI is placed in an as homogenous area of the liver as possible. The sample sizes/volumes of the aorta and liver ROIs are 123 voxels/1.1 ml and 1419 voxels/12.5 ml, respectively. Patient ID: PETMR021.

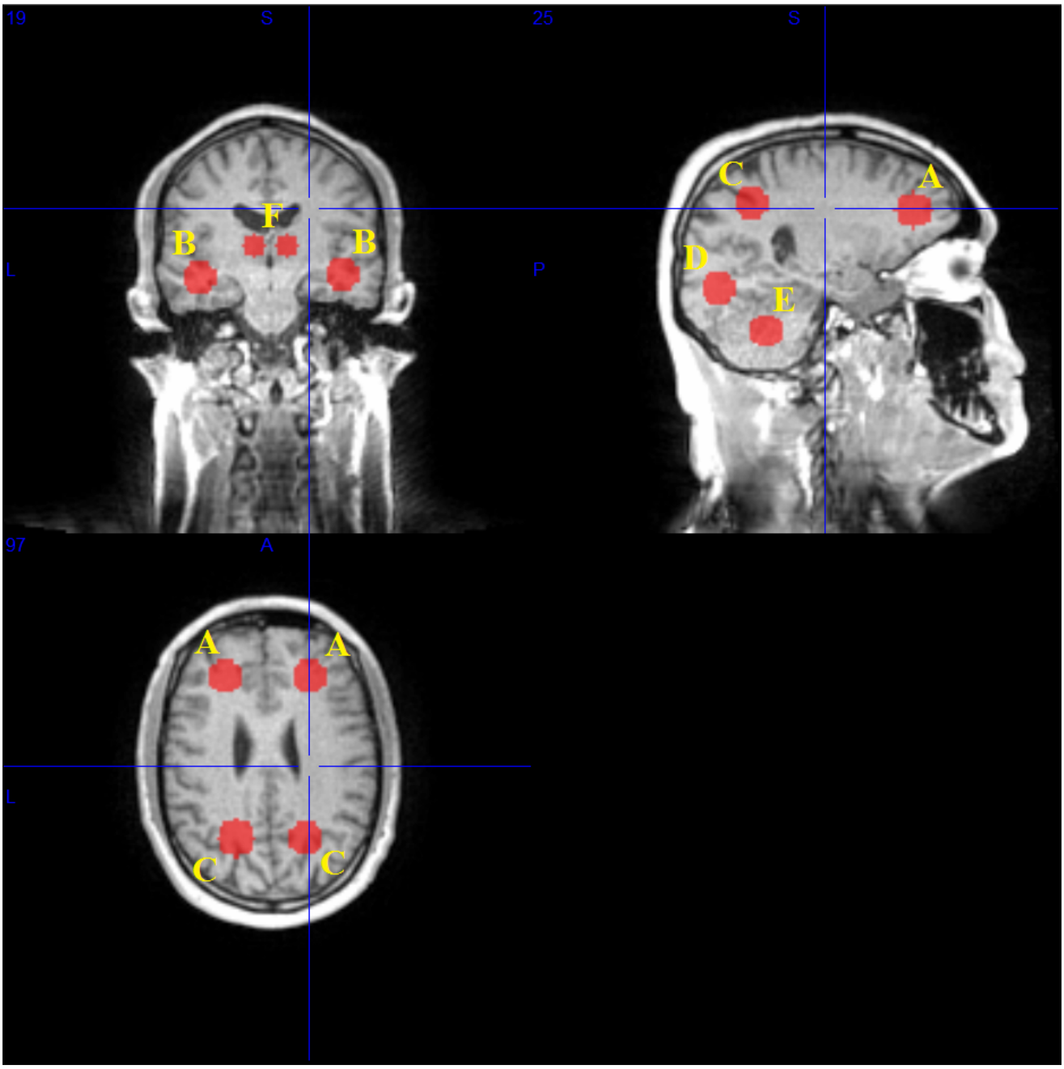


Figure 17: Example of typical placements of all twelve head ROIs, displayed as overlays on a second-echo UTE image. All ROIs have a spherical shape, and are placed in both left and right hemispheres in the following brain regions: Frontal(A), temporal(B), parietal(C), and occipital(D) lobes; cerebellum(E); and thalamus(F). All ROIs have a sample size of 515 voxels and a voloume of 4.5 ml, except those of the thalamus which have a sample size of 123 voxels and a volume of 1.1 ml. Patient ID: PETMR023.

7 Statistics

The measure of relative difference between the quantity x and the reference quantity x_{ref} is in this report defined as

$$RD = \frac{x - x_{\text{ref}}}{x_{\text{ref}}}. \tag{17}$$

All results in the form of relative differences were tested for significance using a t-test on the null hypothesis that the dataset comes from a normal distribution and have a mean equal to zero, using the built-in `MATLAB` function `ttest.m`. The similarity of the left and right hemisphere ROIs were tested using a two-sample t-test on the null hypothesis that the data in the left and right hemisphere ROIs comes from independent random samples from normal distributions with equal means and equal but unknown variances, using the built-in `MATLAB` function `ttest2.m`.

7.1 SUV

To compare the intra-patient PET images, the following calculations were made for each ROI:

- The relative difference in mean SUV values between reconstructions using the original MR µ-map for AC, and reconstructions using the CTMR µ-map for AC.
- Head only: The relative difference in mean SUV values between reconstructions using the original MR µ-map for AC, and reconstructions using the CT µ-map for AC.
- Head only: The relative difference in mean SUV values between reconstructions using the CT µ-map for AC, and reconstructions using the CTMR µ-map for AC.
- The relative difference in mean SUV values between reconstructions performed using the regular 3D-iterative reconstruction method, and reconstructions using the PSF reconstruction method.

As a response to the SUV results extracted from the ROIs, for a few selected patients the relative difference in activity concentration was calculated for the entire PET head images, comparing the same reconstructions as for the ROIs above. The resulting images, referred to as difference images, indicated in which areas the compared image had lower and higher activity concentrations than the reference image, with negative and positive relative values, respectively.

Part III

Results

1 Introduction

This part will first present the developed methods for creating CT-based μ-maps compatible with the mMR, then the results from the assessment of volume and localisation of bone in the head μ-maps. Finally the results from the PET reconstructions utilising the different μ-maps and reconstruction methods will be presented, comparing the two reconstruction methods and different μ-maps based on SUV values and voxel-by-voxel comparison of PET image activity concentrations (difference images as described in the previous part).

All figures in this part displaying examples from patient images, only display selected slices from the images they are depicting. To simplify the writing, these figures are still referred to as if they were the images from which they were collected. The patient images are naturally 3D, and thus difficult to display fully in a 2D medium, while still conserving space. The crosshairs visible in the patient images indicate the relationship between the image planes (sagittal, coronal, transaxial).

The patient images in the figures may appear disproportionate, i.e. as if they have been stretched or compressed in a certain direction. This is just a flawed property of the utilised image viewer, and not an indication of inaccuracies in the images themselves. Some of the figures display multiple images for the sake of visual comparison between them. Even if images are placed side-by-side, it can be hard to identify subtle differences between them. A better way to visually compare images is to view them as overlays in an image viewer program where the images can be manipulated. This was done for all compared images, but in (2D) paper format overlaid images were found to convey less information than the chosen side-by-side comparisons.

Unless otherwise stated, such as by referring to it as built-in, or by naming its creator, *all* `MATLAB` functions and scripts described and referred to in this text have been written single-handedly by the author of this report.

For all box plots in this part (and Appendix C) the following applies: The center line in each box indicates the median, the lower and higher edges are the 25 % and 75 % markers, and the whiskers extend to the most extreme data points not considered outliers, which are marked by red dots. An outlier is defined as a data point outside the interval $\pm 2.7\sigma$, where σ is the standard deviation. The number N given in the figure legends is the number of patients.

When referring to '3D reconstruction', this is a simplified way of referring to the regular 3D PET reconstruction method. The PSF method is naturally also a 3D reconstruction method, but is referred to simply as 'PSF', since this notation simplifies the writing.

2 µ-map creation

This section will present the methods developed for creating head and whole-body µ-maps based on CT-data (CT images). It is written partly in the form of a discussion, justifying the method choices continuously.

2.1 Template

The CT-based µ-maps had to be compatible with the mMR scanner, meaning that it had to be accepted for use in the RetroRecon software. The least cumbersome way of ensuring this was to use the original MR µ-map image, exported as DICOM format from the scanner, as a template for the CT-based µ-map.

`MATLAB` R2014a contains built-in functions for reading and writing DICOM files (`dicomread.m` and `dicomwrite.m`), but these do not successfully decode all private tags when reading, or preserve them when writing, and thus could not be used. Instead, the function `ReadDicomElementList.m` was used to read DICOM files, and the function `WriteDicomElementList.m` was used to write DICOM files, while preserving all tags, both public and private. These functions were written and published on MATLAB Central File Exchange[1] by

---

[1] http://www.mathworks.com/matlabcentral/fileexchange/

Dirk-Jan Kroon[2].

The function `createmodmumap.m` (Appendix A.3) was written to take care of the editing of the template μ-map, replacing the raw image only in each DICOM file. The function takes an image volume saved as a 3D array as input, along with a directory containing the template μ-map, and an output directory. The image array has to be identical in size, orientation, and units as the template image. Each DICOM file from the template directory is read into a MATLAB structure array, the raw image data (the pixel data tag) is replaced with the corresponding slice of the image array, and the modified file is written to the output directory. Regarding the metadata, the resulting modified DICOM image is identical to the template in every single detail, and is to the scanner software indistinguishable from the original MR μ-map.

2.2 Method overview

This section provides an overview of the developed methods for creating head and whole-body μ-maps. The overview is given here to motivate for the subsequent sections which explain two techniques especially important to the developed methods (registration mask creation and conversion from HU to 511 keV LACs), before the methods finally are described in detail.

2.2.1 Head μ-map creation

Three images are utilised in the creation of head μ-maps; a whole-body AC-CT image, a second-echo UTE head image, and a UTE-based head μ-map. The main steps in the developed method are as follows:

1. The AC-CT image is cropped such as to discard the image below the shoulders, result: ‘CThead1’.

2. ‘CThead1’ is registered to the MR UTE image; rigid registration, result: ‘CThead2’.

---

[2] Senior Vision Engineer at Focal Machine Vision en Optical Systems, Netherlands

3. ‘CThead2’ is registered to the MR UTE image; rigid registration, using a fixed-image mask covering the cranial region of the image, result: ‘CThead3’.
4. ‘CThead3’ is converted from HU to PET LACs, result: ‘preCTμ-map’.
5. Two μ-maps are created: ‘CT μ-map’ - only containing ‘preCTμ-map’ image data, excluding the arms of the patient; and ‘CTMR μ-map’ - created by fusing ‘preCTμ-map’ with the MR UTE μ-map, such as to only keep the spatially accurate data from ‘preCTμ-map’ (skull and parts of the neck).

The registrations are controlled visually for each patient. The focus area of the head μ-map is the cranial region, to ensure PET image accuracy in the brain.

2.2.2 Whole-body μ-map creation

Three images are utilised in the creation of whole-body μ-maps; a whole-body AC-CT image, an in-phase Dixon whole-body image, and a Dixon-based whole-body μ-map. The main steps in the developed method are as follows:

1. The AC-CT image is registered to the MR Dixon image, rigid registration, result: ‘CT1’.
2. ‘CT1’ is registered to the MR image, B-spline registration, fixed-image mask excluding the arms, shoulders and chest muscles; moving-image mask excluding the patient bed; result: ‘CT2’.
3. ‘CT2’ is converted from HU to PET LACs, result: ‘preCTμ-map’.
4. A single μ-map is created, ‘CTMR μ-map’, by fusing the MR Dixon μ-map with ’preCTμ-map’, such as to only keep the spatially accurate data from ‘preCTμ-map’ (legs, abdomen, ribcage, neck).

The registrations are controlled visually for each patient. The focus area of the whole-body μ-map is the thorax, as this was the disease area in most of the patients included in the study. The thorax is also an area containing significant amounts of bone, which are left unsegmented in the standard Dixon-based μ-maps.

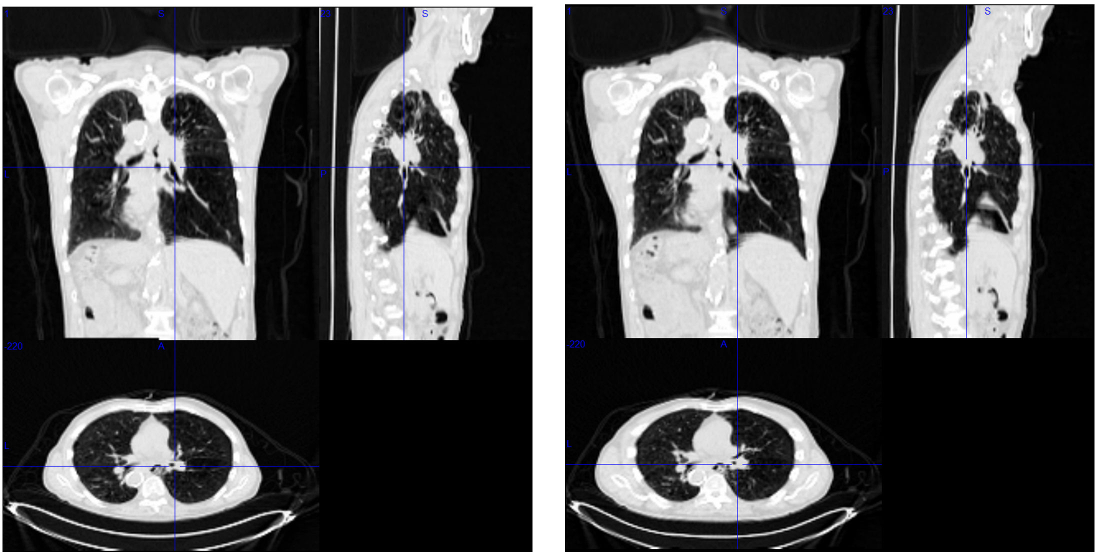
2.3 Image registration masks

As mentioned in Part II, a registration mask is an image restricting the area from which the registration image sampler can draw samples. In other words; the mask focuses the registration on certain areas of its corresponding image, and excludes other areas.

As the reader may recall, the patient position differed between the mCT and the mMR; arms up in the mCT, arms down in the mMR. In addition, the different patient beds in the two scanners resulted in different angles of bending and twisting of the neck between examinations. This led to differences in patient position between the images resulting from the two examinations, of such a degree that a whole-body image from the mCT could not be accurately registered to a corresponding image from the mMR for all parts of the body. The parts that could not be accurately transformed include the arms, the shoulders, the chest muscles, a range of back muscles connected to the shoulders, and the neck and head.

When registering an image from the mCT to a corresponding image from the mMR, without using any registration mask, the result was negatively influenced by the differences in patient position in the two images. Rigid registrations failed to correctly transform vital regions, and non-rigid registrations resulted in unrealistic deformations of the image when trying to transform non-corresponding bodyparts to eachother. A visual comparison between two registrations of a CT image to an MR image, both with the same registration parameters, but one registered with mask and one without, can be found in Figure 18. The image registered without the mask has obvious, unrealistic deformations of the sides of the thorax, as a result of an attempt to fit the thorax to cover the arms in the MR image.

To improve the registrations, masks were used, masking out the parts of the images that could not possibly be transformed to the correct position, focusing the registration on 'the least common denominator'. The manual creation of each single mask needed was a time-consuming and inaccurate process, and an effort was made to automate it. The subsequent section describes the `MATLAB` functions written to aid in the creation of these masks, but not necessarily the complete method used to create the masks, this difference should be kept in mind.



18.1: With mask.

18.2: Without mask.

Figure 18: Visual comparison of two registered CT images, both with the same fixed (MR image, arms down) and moving image (CT image, arms up), and both using the same registration settings (B-spline), but only one of them registered with a fixed-image mask excluding the arms, shoulders, and chest muscles. The crosshairs are placed in the same image coordinate in both images. The most obvious difference is the deformation of the thorax in 2, as a result of trying to fit the thorax to the arms of the fixed image. Patient ID: PETMR021.

2.3.1 Segmentation functions

Two different functions were written to take care of mask creation; one creating masks segmenting the entire imaged body based on simple thresholding, applicable on both head and whole-body images; and one creating masks from whole-body images, segmenting the entire body, excluding the arms, shoulders, and chest muscles. The former was used both when creating head masks and when creating whole-body masks.

The first function, `createmask.m` (Appendix A.2), uses thresholding of image voxel values to discriminate between voxels inside and outside the body. All mask voxels coinciding with the body are set to one, before they are dilated (spherical dilation) to ensure the inclusion of weak-signal voxels at the edges of the body. The final step fills the holes in the mask caused by weak-signal tissue-types inside the body (lung and bone in MR) or by image artefacts (susceptibility artefacts in MR).

The second function, `segmentRibcage.m` (Appendix A.11), also uses voxel-value thresholding, and is based on the assumption that the distance from the outside of the ribcage to the start of the air-filled lung-tissue closest to it, i.e. the thickness of the ribcage wall (thoracic wall), is approximately spatially constant. The function takes a (T_1-weighted) whole-body MR image as input, and progresses through the following steps:

1. Identify which parts of the image that are part of the patient body using `createmask.m`.
2. Segment all gas-filled (weak-signal) volumes inside the body.
3. Detect which of these volumes that contain the lungs; the lungs will be contained within either the largest or the two largest volumes, depending on the tracheal connection (the air in the trachea connecting the lungs) being detectable or not. All other volumes are removed from the mask. Ask for manual confirmation that the lung segmentation was successful.
4. Ask for manual input of top and bottom z-coordinate of lungs, remove parts of volume exceeding these values (typically the trachea, and gas-filled cavities in the abdomen appearing to be connected to the lungs).
5. Dilate the lung volume in negative z-direction, towards the bottom of the lungs.
6. Dilate the resulting volume in both positive and negative x-direction, towards the center of gravity of the lungs ($x = 0$). ($x > 0 \Rightarrow$ dilation in negative x-direction, and vice versa).

7. Dilate the resulting volume an equal distance in all directions (inflate the volume). The resulting volume should include the ribcage and the organs within, excluding everything else.

8. Segment the rest of the body below the lungs using `createmask.m`, ensuring that the arms are not included in the mask.

9. Segment the head and neck based on manual input of the z-coordinate of the top of the shoulders.

`segmentRibcage.m` successfully segmented the least common denominator between the mCT and the mMR in all patient whole-body images, creating fixed-image registration masks that improved the necessary non-rigid registrations performed during the creation of whole-body CT-based μ-maps massively. An example of a registration mask can be seen in Figure 19.

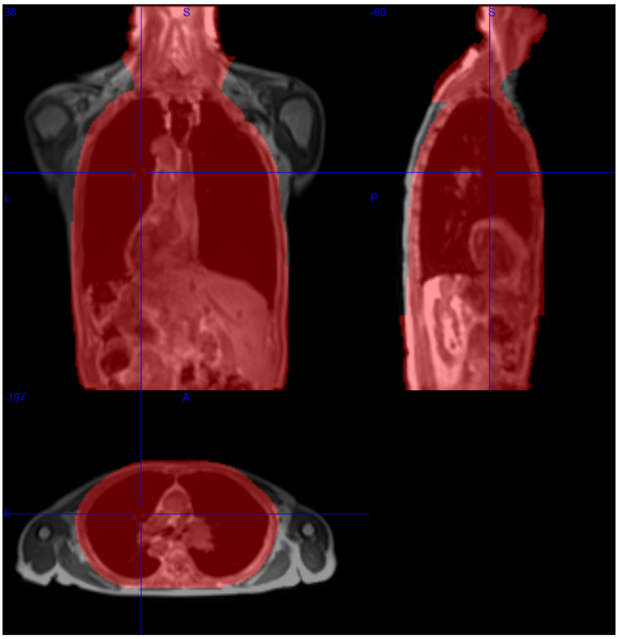
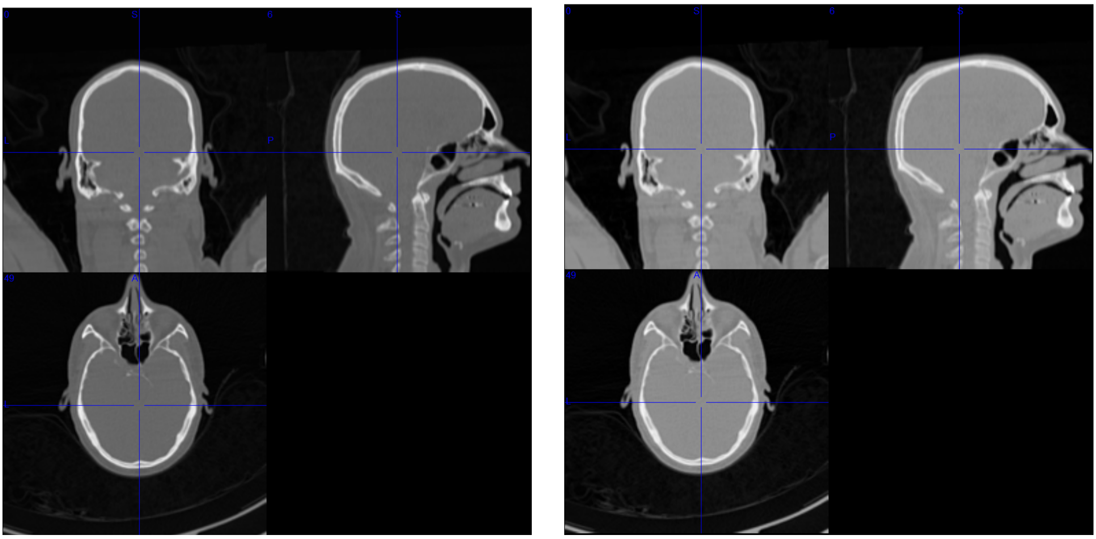


Figure 19: Example of a fixed-image mask utilised during B-spline registration of a whole-body image. The mask excludes the arms, sholders and chest muscles. The mask is layered on the in-phase Dixon image it was created from. Patient ID: PETMR021.

2.4 HU to PET LAC conversion

The function `HU2PETLAC.m` converts CT image volumes with voxel values in HU to 511 keV LAC μ-maps, and can be found in Appendix A.9. The conversion is performed as described in Part II Section 3.2, and the peak tube voltage can be chosen among the values given in Table 2. The output image is scaled to units [hm^{-1}] to match that of the original MR μ-maps.

Figure 20 shows an example of a CT image with voxel values in Hounsfield units and the same image converted to 511 keV PET LAC values using `HU2PETLAC.m`. The visual consequence of the conversion is slightly brighter soft-tissue voxels, due to the smaller relative range from minimum to maximum value in the converted image. The voxel values were evaluated for a few random voxels in bone and in soft-tissue, and the values corresponded well with known values for these tissue types.



20.1 *20.2*

Figure 20: A CT image with voxel values in HU (1) and in LAC (2). The LAC image has a smaller relative range from minimum to maximum value, which is manifested visually as slightly brighter soft-tissue compared to the HU image. Patient ID: PETMR023.

2.5 Head µ-map creation

The script `CT2MRmumap_head.m` was written to take care of the different steps constituting the creation of head µ-maps, and can be found in Appendix A.4. This particular version of the script creates CTMR µ-maps, but is nearly identical to the one creating CT µ-maps.

The first step of creating a µ-map is to convert the three images from DICOM format to 3D NIFTI format. This is mainly because of the use of `elastix`, but also introduces the advantage of only having to handle a single image file at a time until the end of the process. The resulting images from the conversions of the second-echo MR UTE image, the AC-CT image, and the MR UTE-based µ-map are referred to simply as 'MR', 'CT', and 'MR µ-map', respectively.

In the second step a fixed-image registration mask is created from the MR image, the CT image is cropped, and two rigid registrations are performed, both with the MR image as the fixed image. The mask is created using the function `createmask.m`, and is cropped to cover the cranial region only, i.e. all transaxial image slices containing cranial bone. To ease the first registration, the CT image is cropped to better fit the FOV of the MR image. This is done by simply excluding all transaxial slices below a manually chosen z-value, typically just below the shoulders, and the result is referred to as 'CT head'.

Figure 21 displays examples of a CT head image and an MR image. Notable differences between the two images include different FOV, different resolution, arms and patient bed visible in CT image, and different neck postures.

The first registration is performed with the MR image as the fixed image, the CT head image as the moving image, and with the registration parameter file 'rigidHeadCT.txt' found in Appendix B.2. The purpose of this registration is to create a good starting point for the second registration. The second registration uses the MR image as the fixed image, and the transformed image from the first registration as the moving image, along with the registration mask described above. The registration parameters are defined in 'rigidHeadCT2.txt' found in Appendix B.3. The resulting image is referred to as 'transformed CT image'.

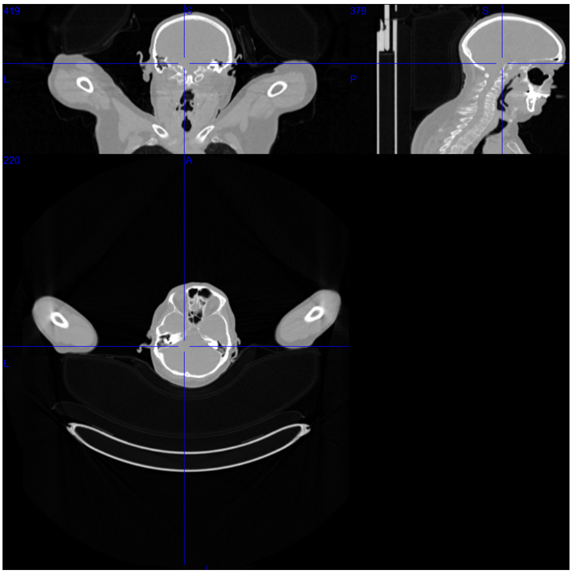
Examples of images resulting from the two registrations are displayed in Figure 22. The most obvious visual difference between the two images is the difference in skull position. This can be observed in the shifted nose in the transaxial slice, in the right ear channel in the coronal slice, and the angle of the vertebrae in the sagittal slice. Opening them as overlays in an image viewer shows that the result of the second registration is an excellent alignment of the cranium between the registered CT image and the MR image.

In the third step a fusion mask is created, either based on the MR image or the CT image, depending on the similarity between the initial images regarding neck posture, and thus the accuracy of the transformation of the CT image. The fusion mask is manually edited to only include the parts of the transformed image from the second registration that is sufficiently spatially accurate. This is often limited to the skull (cranium and lower jaw), and part of the neck.

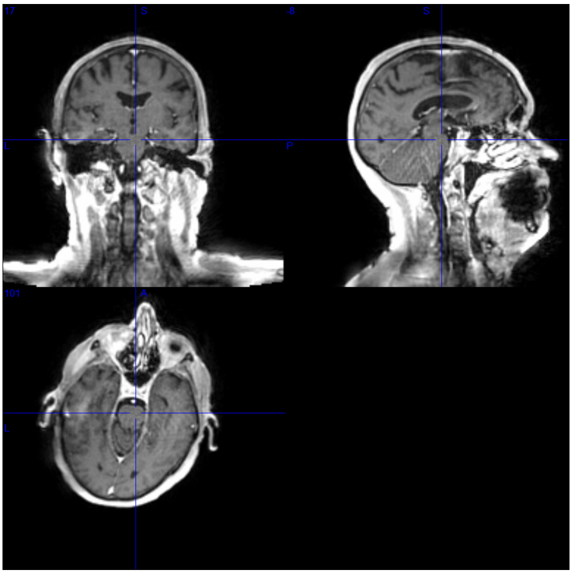
The final step includes conversion of the transformed CT image using the function `HU2PETLAC.m`, resulting in a CT μ-map. This image is fused with the MR μ-map, using the fusion mask as a boundary, with CT μ-map inside the mask and MR μ-map outside it. This CTMR μ-map is then saved as a DICOM directory, using `createmodmumap.m` with the original MR μ-map as the template.

Figure 23 displays examples of a CT μ-map and a CTMR μ-map. As shown, the CT μ-map consists only of the converted CT image, with most of the arms cut away. Because of time constraints, and the assumption that they would not have any effect on the brain region, the arm stumps were not completely removed. The CTMR μ-map consists of the skull and part of the neck from the converted CT image, while the rest of the neck and the shoulders are kept from the MR μ-map. There are also some parts from the MR μ-map around the outer ear of the CTMR μ-map, which is caused by the hearing protection wore by the patient in the mMR.

The difference between the creation of CTMR μ-maps and CT μ-maps, is that in the creation of the latter there is no fusion of the two images. The CT μ-map is edited so that the arms (which are raised above the head) are (almost completely) removed. Other than this the procedure is identical to that described above.

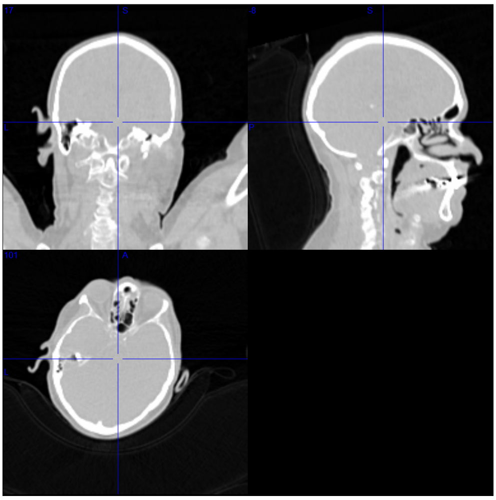


21.1: CT.

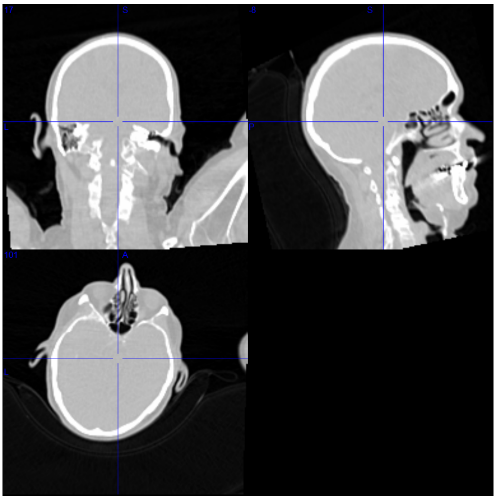


21.2: MR.

Figure 21: Examples of a CT head image and a second-echo UTE MR image. In the CT image the patient bed and the raised arms are visible. The two images also differ in patient neck posture. Patient ID: PETMR036.

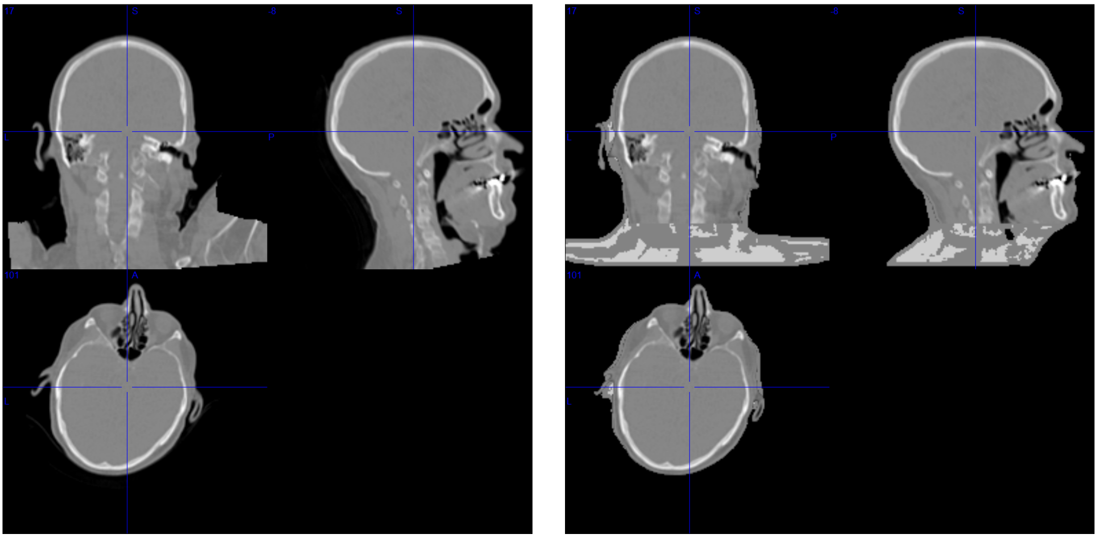


22.1



22.2

Figure 22: Examples of CT images resulting from the registrations performed when creating head μ-maps. 1 is a CT image (Figure 21.1) registered to an MR head image (Figure 21.2). 2 is 1 registered to the same MR image again, with a fixed-image mask covering the cranial region. The crosshairs are positioned in the exact same image coordinate as in Figure 21.2. The most obvious visual difference is the rotation of the image, caused by focusing the second registration on the cranium. Patient ID: PETMR036.



23.1: CT μ-map.

23.2: CTMR μ-map.

Figure 23: Examples of CT and CTMR head μ-maps. The crosshairs are positioned in the exact same image coordinate in these two images as in Figure 21.2. In 1 the remaining arm stumps are visible, while in 2 most of the neck and shoulders are taken from the MR μ-map. Patient ID: PETMR036.

2.6 Whole-body μ-map creation

The script `CT2MRmumap_wb.m` was written to take care of the different steps constituting the creation of whole-body μ-maps, and can be found in Appendix A.5. As with `CT2MRmumap_head.m`, the first step of creating a μ-map is to convert the three input images from DICOM to NIFTI format. The resulting images from the conversions of the AC-CT image, the in-phase whole-body Dixon image, and the Dixon-based whole-body μ-map are referred to as ‘CT’, ‘MR’, and ‘MR μ-map’, respectively.

Figure 24 displays examples of an AC-CT image and a whole-body in-phase Dixon image. Notable differences between the two images include different FOV, different resolution, different arm positions, a visible patient bed in the CT image, and different neck postures.

In the second step the whole-body fixed-image registration mask is created by feeding the MR image to `segmentRibcage.m`. The function is run several times until a suitable combination of input parameters has been achieved.

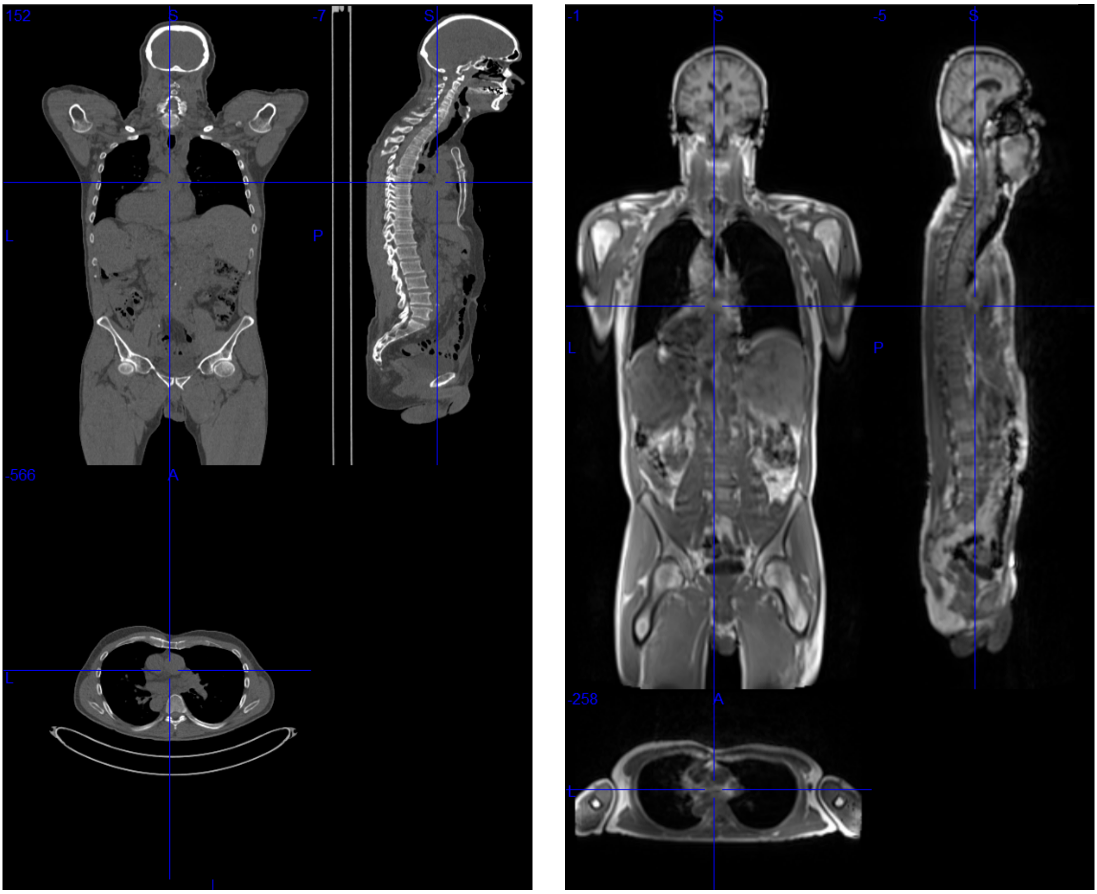
The third step performs the rigid registration, with the MR image as the fixed image, the CT image as the moving image, and the registration parameter file ‘rigidWBCT.txt’ found in Appendix B.4. For the resulting transformed CT image a mask is created such as to include the entire patient, excluding the patient bed.

The non-rigid registration is performed in the fourth step, again using the MR image as the fixed image, the transformed image from the previous registration as the moving image, and with the fixed-image and moving-image masks created in the previous steps. The registration is performed with the settings defined in ‘BsplineWBCT.txt’ found in Appendix B.1.

Figure 25 displays examples of two resulting images from the two described registrations. The most obvious visual difference between the two images can be seen in the top of the lungs and in the pelvic area in the coronal slices, in the sinuses in the sagittal slices, and in the right lung in the transaxial slice. Further visual analysis reveals that the B-spline-registered CT image provides a much better fit to the MR image than the rigidly registered one.

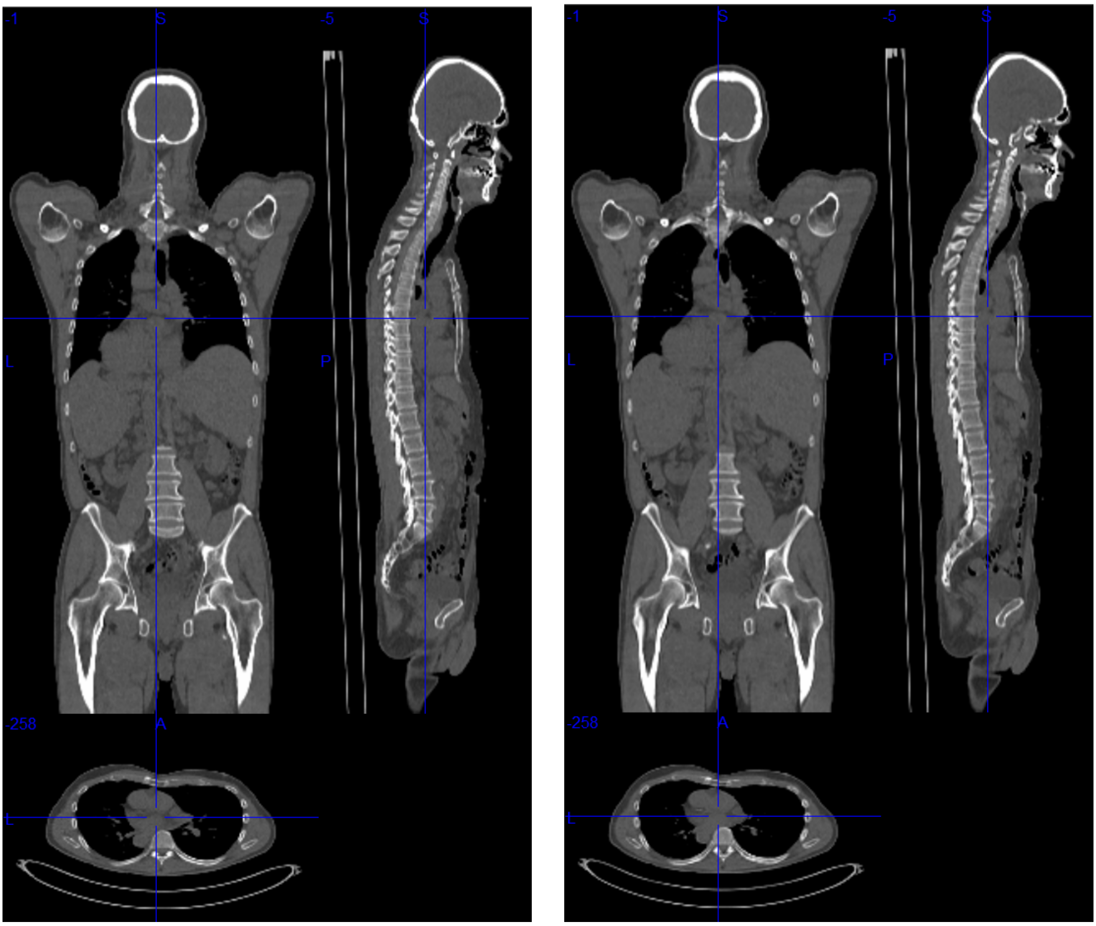
The final step converts the transformed CT image using `HU2PETLAC.m`, and fuses the resulting CT μ-map with the MR μ-map, using the fixed-image mask from the non-rigid registration as a boundary in the same fashion as in the creation of fused head μ-maps. If the final registration fails to successfully transform the head, the mask is cropped to exclude this bodypart. The resulting CTMR μ-map is saved as DICOM using `createmodmumap.m` with the MR μ-map as the template.

An example of an original whole-body MR μ-map and a CTMR μ-map can be seen in Figure 26. The most notable difference between the two images is the visible bone in the CT parts of the CTMR μ-map. Other than that the CTMR μ-map is more detailed in the abdomen and the lung, and offers continuous μ-values in the CT-parts of the image.



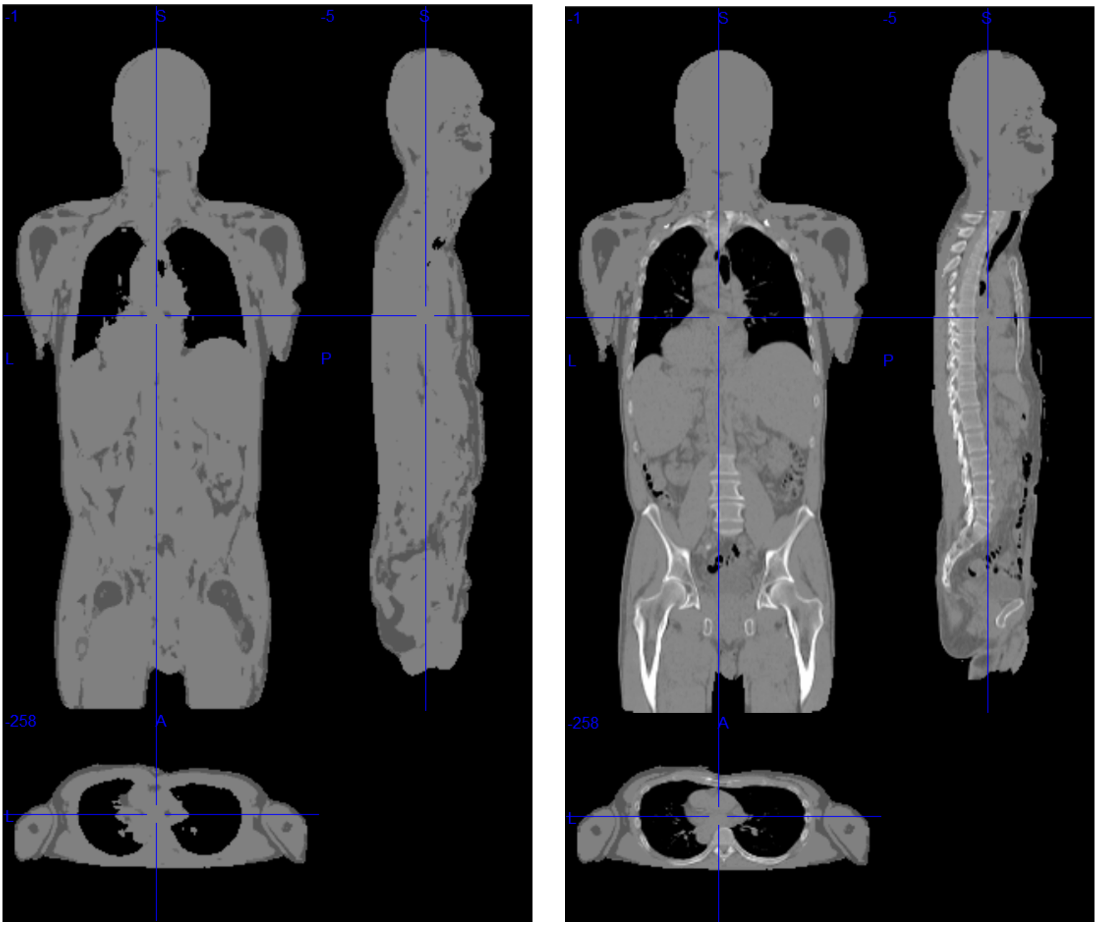
24.1: CT. *24.2: MR.*

Figure 24: Examples of an AC-CT image (1) and a whole-body in-phase Dixon MR image (2). In the CT image the patient bed is visible. The positions of the arms and neck posture differ between the two images. Patient ID: PETMR024.



25.1 25.2

Figure 25: The result of two registrations performed when creating whole-body μ-maps. 1 is a CT image (Figure 24.1) registered to an MR image (Figure 24.2), rigid registration. 2 is 1 registered to the same MR image again, non-rigid registration with a fixed-image mask covering the least common denominator between the fixed and moving image. The crosshairs in these two images are placed in the exact same position as in the image in Figure 24.2. Differences between the two registered images can be seen in the top of the lungs and in the pelvic area in the coronal slices, in the sinuses in the sagittal slices, and in the right lung in the transaxial slice. Patient ID: PETMR024.



26.1 *26.2*

Figure 26: Examples of whole-body MR (1) and CTMR (2) μ-maps. The crosshairs in these two images are placed in the exact same position as in the image in Figure 24.2. The most obvious differences include the visible bone and finer detail in 2 compared to 1. Patient ID: PETMR024.

3 Head bone segmentation

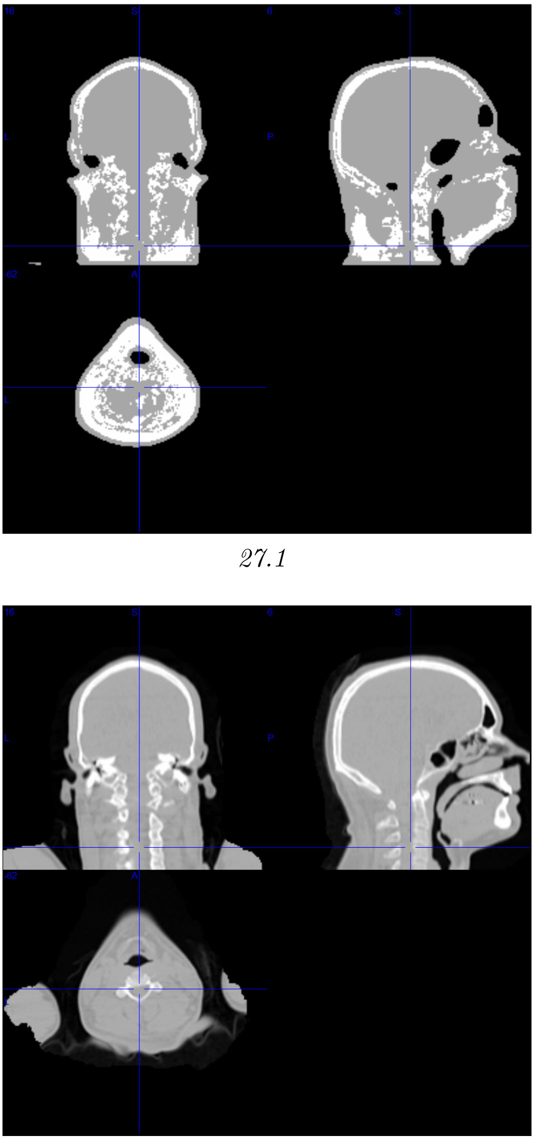
When examining a typical MR UTE μ-map of the head and neck, like the one seen in Figure 27.1, it is clear that the MR μ-maps overestimate the amount of bone in the neck region of the patient. When visually comparing the MR μ-map with a CT μ-map, such as the one seen in Figure 27.2, assuming that the CT μ-map is relatively close to render the real amount of bone, the gross overestimation becomes even more obvious. In addition, it seems as if the MR μ-map is misclassifying some of the real bone as soft-tissue, as seems to be the case in the forehead of the sagittal slice. There is also obvious misclassification of tissue around the ears.

3.1 Volume

The results of the bone volume measurements in the head μ-maps demonstrate that the MR μ-map has significantly more bone than the CT μ-map (not CTMR μ-map) when assessing the μ-map as a whole. When assessing the cranial region only, the MR μ-map has significantly less bone than both the CT-based μ-maps.

Figure 28 displays a box plot of the absolute volume of bone in the different μ-maps. Figure 29 displays a box plot of the relative amount of bone in the CT-based μ-maps compared with the MR μ-map. A one-sample t-test performed on the datasets ‘Full mumap - CTMR’, ‘Cranium - CT’ and ‘Cranium - CTMR’ in Figure 29 returned p-values of 0.0741, 0.0142 and 0.0135, respectively, for the null hypothesis that the dataset comes from a normal distribution and have a mean equal to zero; indicating the statistical significance of the differences.

When considering the full μ-map, the MR and CTMR μ-maps contain a lot more bone than the CT μ-map (with a mean of about 30 % more), while the CTMR μ-map does not contain significantly more bone than the MR μ-map. For the cranial region, both the CT and the CTMR μ-map have significantly more bone than the MR μ-map, both with a mean of 8.3 % more.



27.1

27.2

Figure 27: Visual comparison of the amount of bone in the head and neck area between a CT μ-map (2) and an MR UTE μ-map (1). The crosshairs are placed in the same image coordinate in both images, but the registration of the CT image is slightly inaccurate in the neck area. Patient ID: PETMR023.

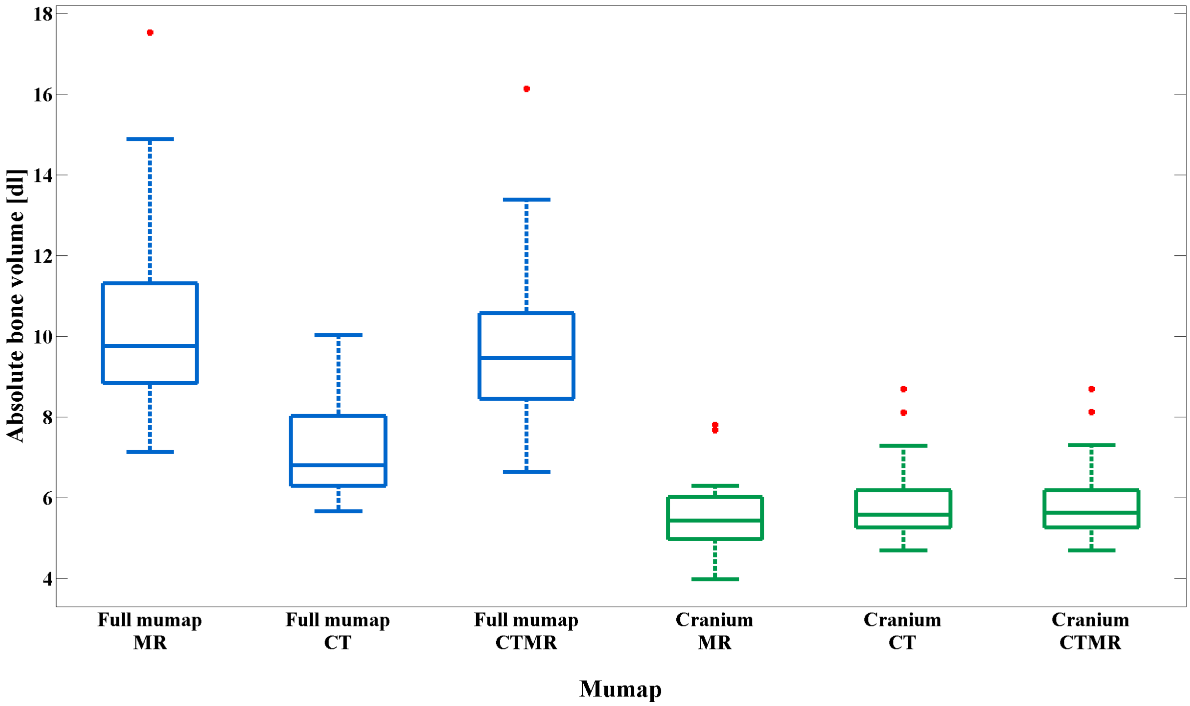


Figure 28: Box plot of the absolute bone volume in the different head μ-maps, both for the entire μ-map (blue) and for just the cranial region (green). $N = 19$.

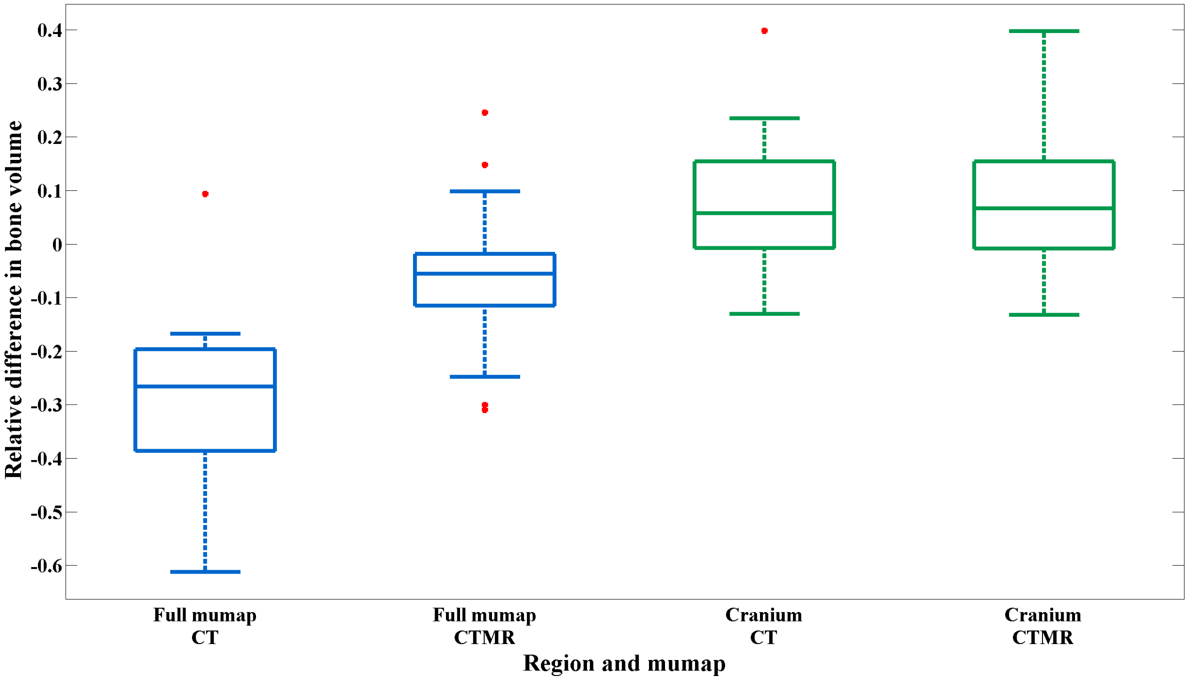


Figure 29: Box plot of the relative difference in bone volume in the head/neck between the original MR UTE μ*-map and the CT-based* μ*-maps, both for the entire* μ*-map and for just the cranial region. The MR* μ*-map is used as the reference point. P-values from t-test for the null hypothesis that the dataset comes from a normal distribution and have a mean equal to zero: 1.518e-07, 0.074, 0.014, and 0.014, from left to right, respectively.* $N = 19$.

3.2 Localisation

Figure 30 displays a box plot of the relative amount of unique bone in the cranial region of each of the MR and the CT μ-maps, i.e. the relative amount of cranial bone voxels (voxels classified as bone) unique to the specific μ-map compared with the other μ-map. About 35 % (mean difference) of the cranial bone voxels in the MR μ-map are unique to the MR μ-map, while about 40 % (mean difference) of the cranial bone voxels in the CT μ-map are unique to that μ-map.

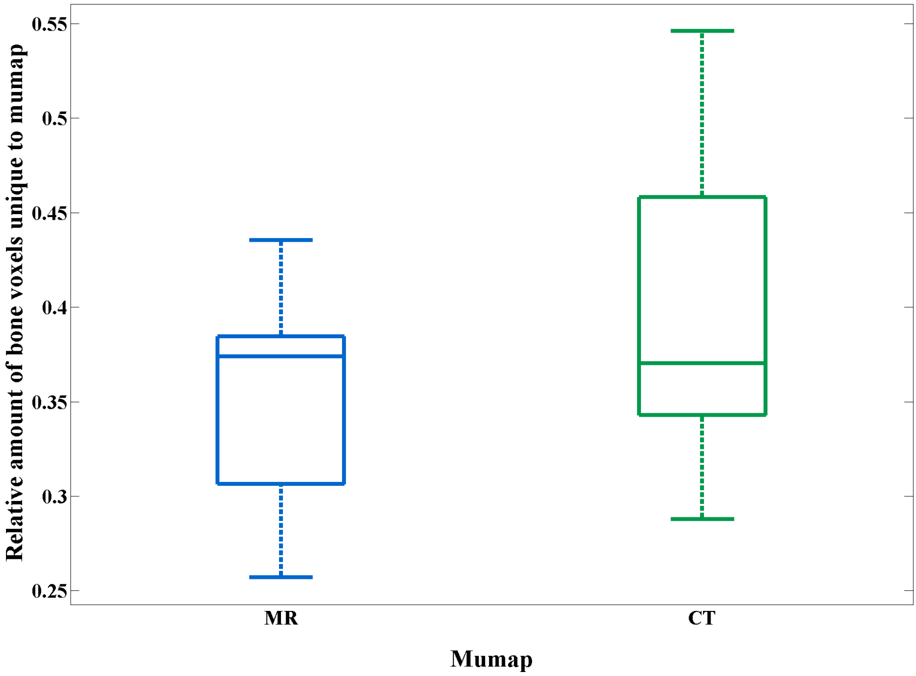


Figure 30: Box plot of the relative amount of bone voxels in the cranial region of the head μ-map which are not corresponding to bone voxels in the other μ-map. $N = 19$.

4 PET image analysis

This section will present the results of the analysis of the different PET image reconstructions performed with the different μ-maps and reconstruction methods, as described in Part II Section 7.1. First the comparisons of the MR μ-map and the CT-based μ-maps are presented, and then the comparison of the regular 3D and the PSF reconstruction methods.

4.1 Example images

4.1.1 Head

Figure 31 displays an example of a head PET image overlaid on a head MR image. The highest FDG concentrations are found in the brain, especially in the cerebral cortex. This is in concordance with the known high relative glucose consumption in the brain. There are no visible malignancies in any of the patients.

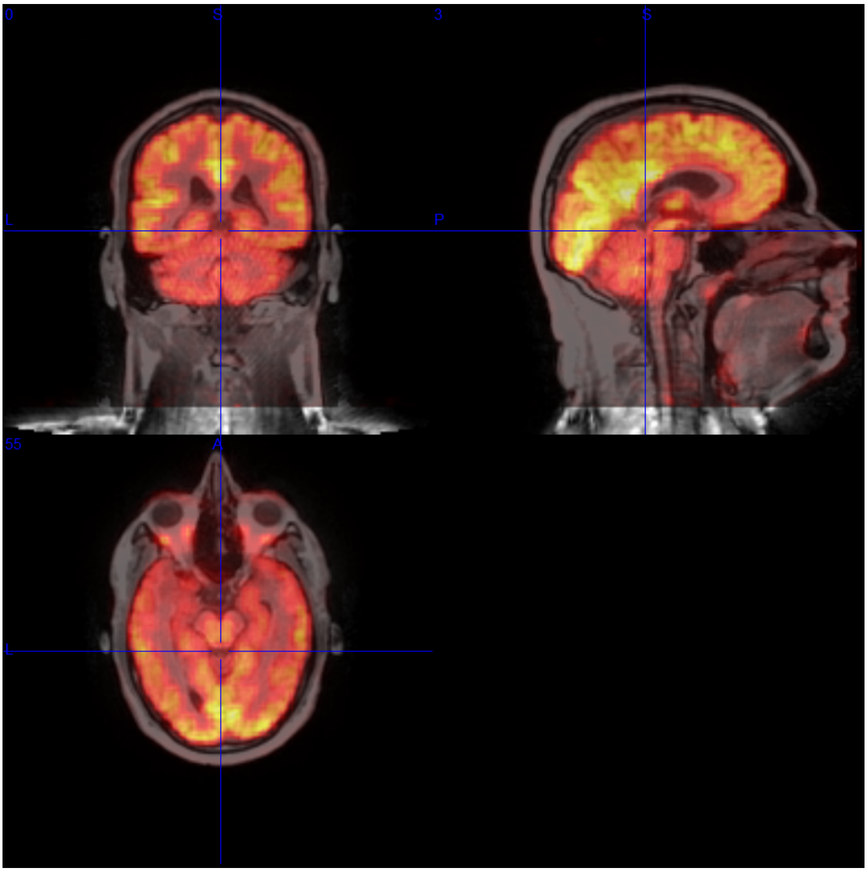


Figure 31: Example of a head FDG-PET image overlaid on a second-echo UTE MR image. The MR image is the greyscale background, while the PET image is displayed in a hot colour scale. The scale's low and high ends are dark red and bright yellow, respectively. Higher activity indicates higher glucose consumption. Patient ID: PETMR023.

4.1.2 Whole-body

Figure 32 displays an example of a whole-body PET image overlaid on a whole-body MR image. The high activity seen in the brain, the bladder, the stomach, and in the liver are all normal. The smaller bright spots between the lungs and the shoulders in this lymphoma patient indicate lymph nodes with metastases.

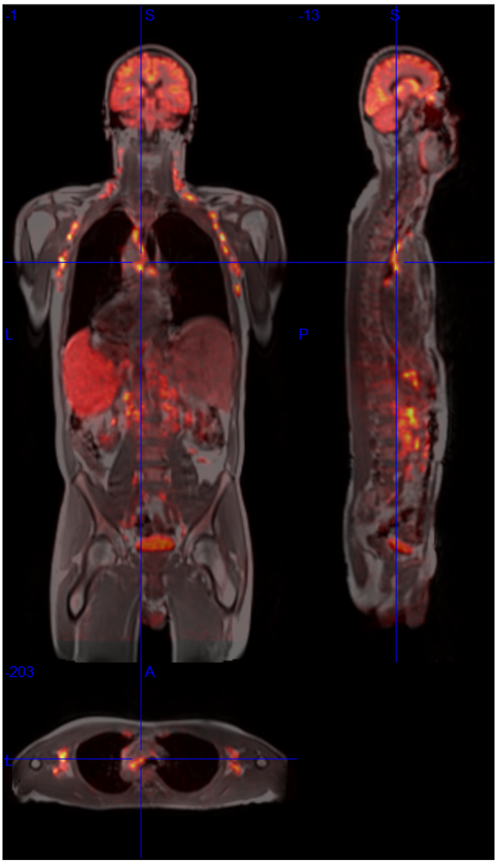


Figure 32: Example of a whole-body FDG-PET overlaid on a whole-body in-phase Dixon MR image. The MR image is the greyscale background, while the PET image is displayed in a hot colour scale. The scale's low and high ends are dark red and bright yellow, respectively. Higher activity indicates higher glucose consumption. This particular patient suffers from lymphoma, and the high glucose consumption in the lymph nodes is clearly visible. Patient ID: PETMR024.

4.2 MR - CTMR

The results presented below show that the reconstructions with the CTMR μ-map result in significantly higher mean SUV values relative to the reconstructions with the MR μ-map in all ROIs, except for the aorta in which the mean SUV values are not significantly different from zero.

4.2.1 Head

Figure 33 displays a box plot of the relative differences in mean SUV values for the head ROIs between PET image reconstructions performed using the original MR UTE-based μ-map, and the CTMR μ-map for AC. All reconstructions were performed using the PSF reconstruction method, and the MR μ-map reconstructions have been used as the reference point, according to Equation (17). The corresponding results for comparisons of MR & CTMR 3D reconstructions, and MR & CT 3D and MR & CT PSF reconstructions are very similar, and can be found in Appendix C.

All ROIs display a significant increase in mean SUV values (maximum p-value < 0.004) in the CTMR reconstructions compared with the MR reconstructions. The similarity of the left and right hemisphere ROIs were tested using a two-sample t-test on the null hypothesis that the data in the left and right ROIs comes from independent random samples from normal distributions with equal means and equal but unknown variances; yielding a minimum p-value of 0.50, meaning that the left and right hemispheres are not significantly different. The cerebellum ROIs display a larger variance than the others. The mean increase in mean SUV values, for both right and left hemispheres, for the frontal, temporal, parietal and occipital lobes; cerebellum; and thalamus ROIs are 8.4 %, 2.8 %, 5.8 %, 9.4 %, 9.1 %, and 3.2 %, respectively.

Figure 34 shows a difference image comparing images reconstructed with the CTMR μ-map and the MR μ-map for AC, where the negative values (blue) indicate voxels in which the CTMR reconstruction result in lower values than the MR reconstruction, while positive values (red) indicate voxels in which CTMR result in higher values than MR. The values in the difference image are in the order of - 50 % to + 10000 % in tissue in general, while in the brain they range from about +2 % in the center to about +20 % closer to the bone for most areas. The highest positive values and lowest negative values of the difference image seem to correspond with the areas where the CTMR and MR µ-maps differ obviously in (bone) structure. The resulting difference image from the 3D reconstructions did not display any obvious differences from the PSF reconstructions, and is thus included.

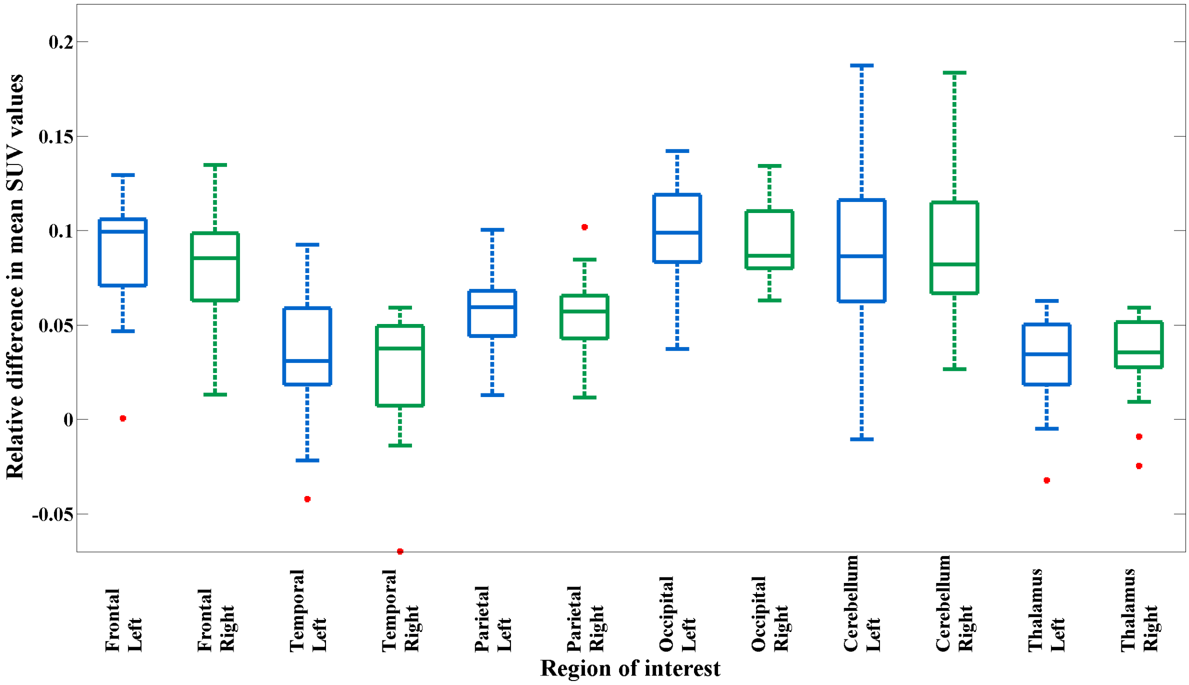
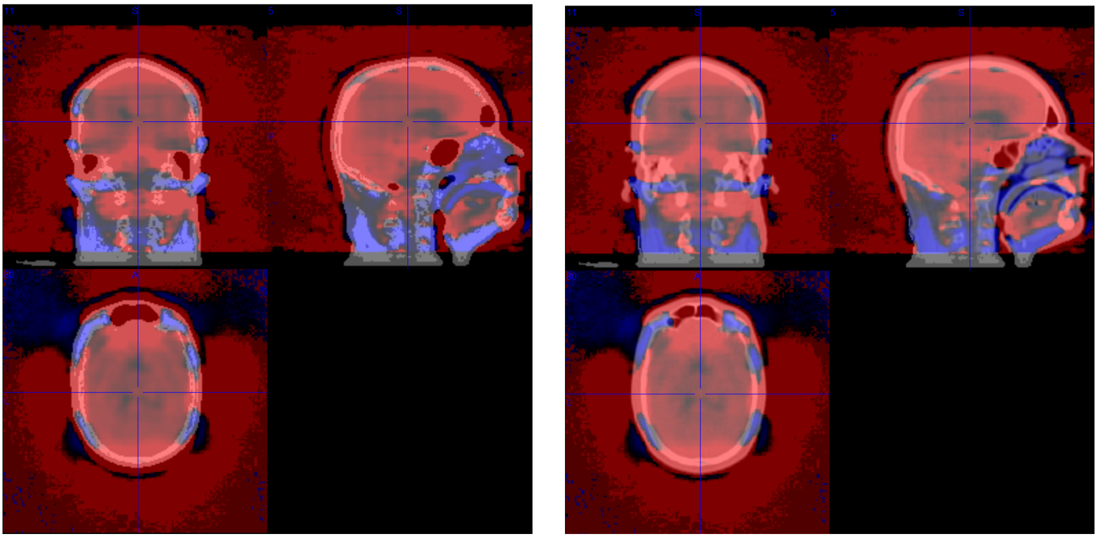


Figure 33: Box plot of the relative difference in mean SUV values between PET head images reconstructed using the original MR UTE μ-map, and using the CTMR μ-map. The reconstruction method was PSF, and the original MR UTE μ-map reconstruction results were used as the reference point. $N = 19$.



34.1: MR μ-map.

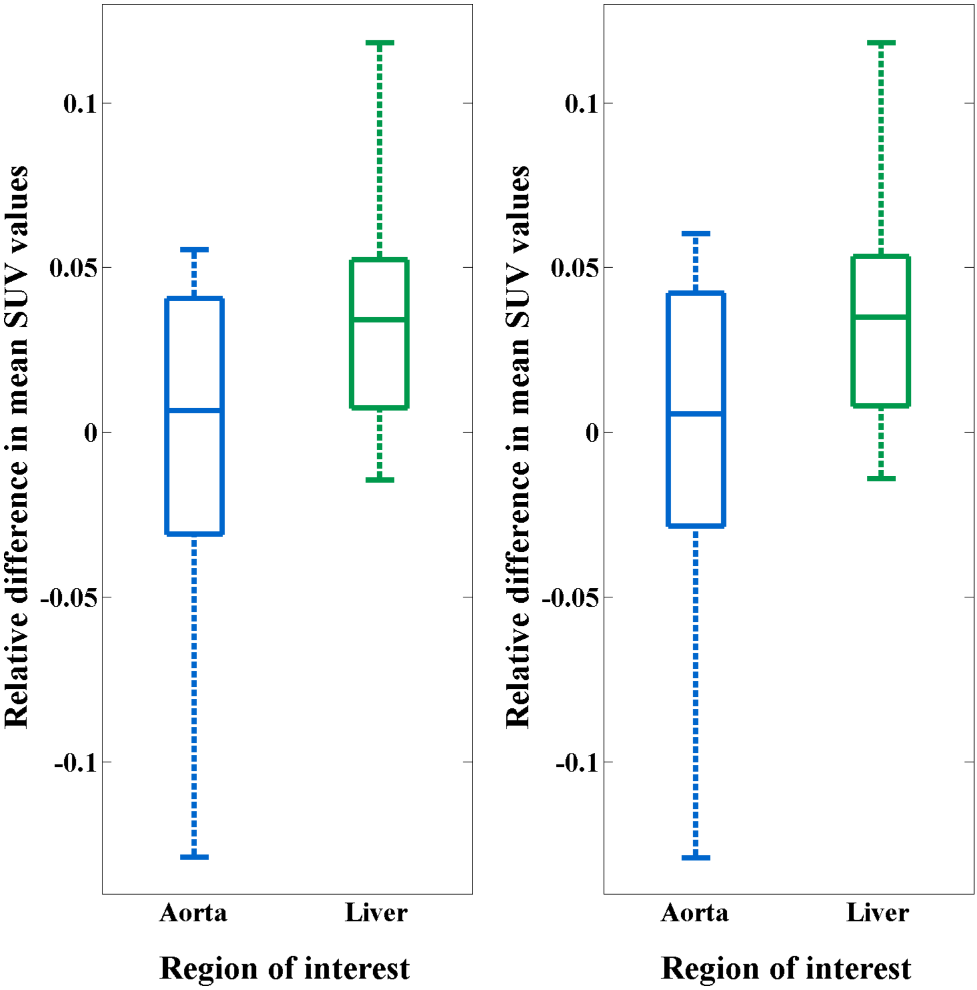
34.2: CTMR μ-map.

Figure 34: A difference image comparing images reconstructed using the two different μ-maps MR μ-map and CTMR μ-map for AC, overlaid on the MR μ-map and on the CTMR μ-map. The μ-map images are shown as greyscale backgrounds, while the negative and positive values of the difference image are shown in blue and red, respectively. Both PET images were reconstructed using the PSF reconstruction method, and the MR μ-map reconstruction was used as reference. Blue indicates voxels in which the CTMR reconstruction result in lower values than the MR reconstruction, while red indicates voxels in which CTMR result in higher values than MR. Patient ID: PETMR023.

4.2.2 Whole-body

Figure 35 displays plots for both 3D reconstruction and PSF reconstruction, of the relative difference in mean SUV values in the aorta and liver ROIs between PET image reconstructions using the original MR Dixon μ-map for AC, and reconstructions using the CTMR μ-map for AC. The MR reconstructions have been used as the reference point according to Equation (17).

For the 3D reconstructions the mean SUV values are significantly (p-value of 6.255e-04) higher for the liver ROI (mean of +3.78 %), while the mean SUV values in the aorta are not significantly different from zero (p-value of 0.579). For the PSF reconstructions the mean SUV values are significantly (p-value of 5.7539e-04) higher for the liver ROI (mean of +3.82 %), while the mean SUV values in the aorta are not significantly different from zero (p-value of 0.510).



35.1

35.2

Figure 35: Box plot of the relative difference in mean SUV values between PET whole-body images reconstructed using the original MR UTE μ-map, and using the CTMR μ-map. The MR UTE μ-map results have been used as the reference point. Both the results using the 3D (1) and the PSF reconstruction (2) methods are shown. $N = 18$.

4.3 CT - CTMR (head only)

The results below show that there is a significant difference in resulting mean SUV values from images reconstructed with the two different CT-based head μ-maps (CTMR and CT); the CTMR μ-map yields significantly lower mean SUV values.

Figure 36 displays a box plot of the relative difference in mean SUV values in all head ROIs between PET image reconstructions using the CT μ-map for AC, and using the CTMR μ-map for AC. The reconstructions were performed using the PSF reconstruction method, and the CT reconstructions were used as the reference point according to Equation (17). The corresponding results for 3D reconstructions are very similar and can be found in Appendix C.

All ROIs display significant decreases in mean SUV values in the CTMR reconstruction compared to the CT reconstruction. The mean decrease in mean SUV values, for both right and left hemispheres, for the frontal, temporal, parietal and occipital lobes; cerebellum; and thalamus ROIs are 1.1 %, 1.5 %, 1.9 %, 2.2 %, 1.7 %, and 1.4 %, respectively.

Figure 37 shows a difference image comparing the images reconstructed with the CTMR μ-map and the CT μ-map for AC, where the negative values (blue) indicate voxels in which the CTMR reconstruction result in lower values than the CT reconstruction, while positive values (red) indicate voxels in which CTMR result in higher values than CT. The values in the difference image range from -50 % to +150 % in tissue. The hot spots seen in the tissue seems to correlate with where the two μ-maps differ, i.e. around the outer ears and the cheeks. Closer examination of the image reveals that the difference image contains small negative values in the entire brain, ranging in the order of -0.1 % to -3 %.

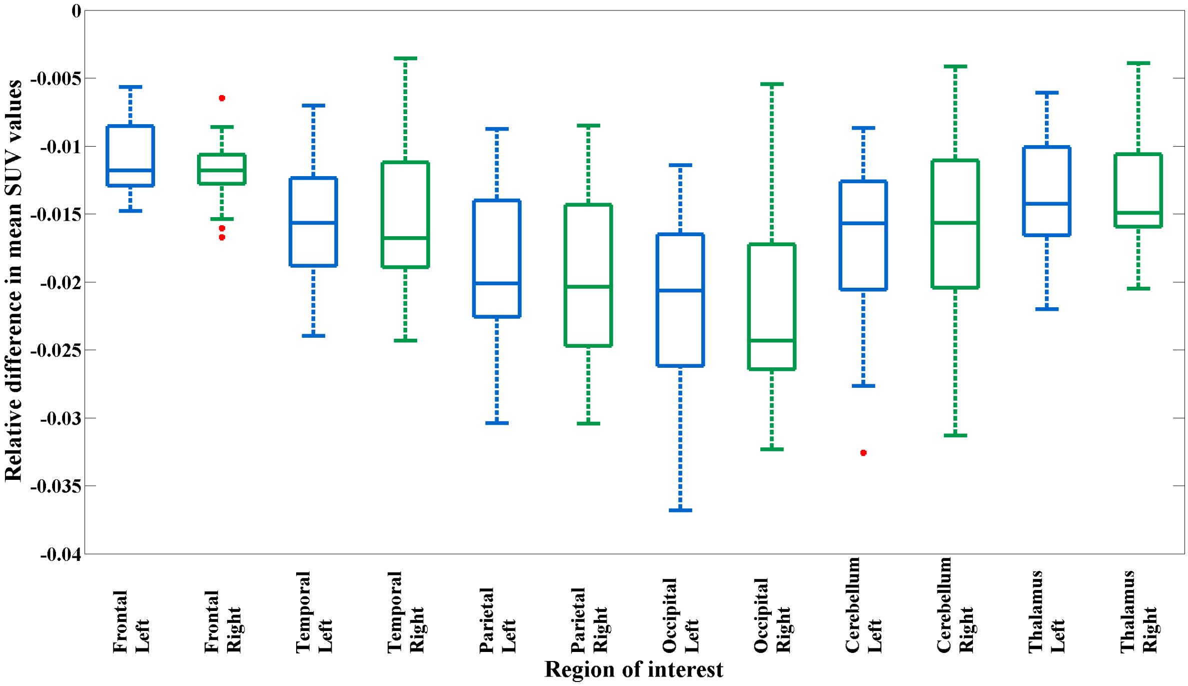
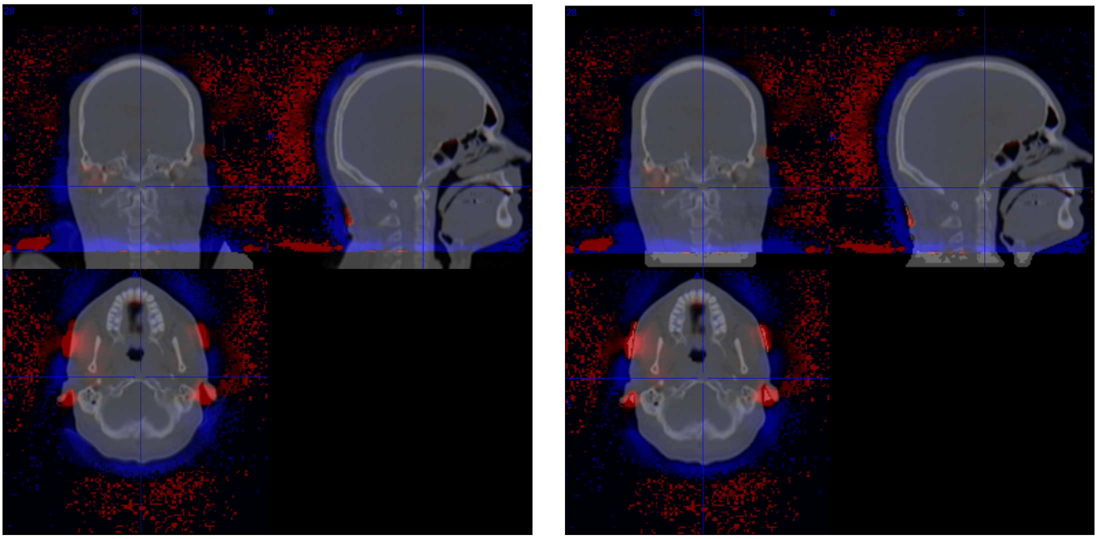


Figure 36: Box plot of the relative difference in mean SUV values between PET head images reconstructed using the CT μ-map, and using the CTMR μ-map. The reconstrucion method was PSF, and the CT reconstruction results were used as the reference point. $N = 19$.



37.1: *CT μ-map.*

37.2: *CTMR μ-map.*

Figure 37: A difference image comparing images reconstructed using the two different μ-maps CT μ-map and CTMR μ-map for AC, overlaid on the CT μ-map and on the CTMR μ-map. The μ-map images are shown as greyscale backgrounds, while the negative and positive values of the difference image are shown in blue and red, respectively. Both PET images were reconstructed using the PSF reconstruction method, and the CT μ-map reconstruction was used as reference. Blue indicates voxels in which the CTMR reconstruction result in lower values than the CT reconstruction, while red indicates voxels in which CTMR result in higher values than CT. Patient ID: PETMR023.

4.4 3D - PSF

The results below show that in the brain the PSF reconstruction results in both higher and lower mean SUV values relative to the regular 3D reconstruction method, depending on where in the brain the SUV is measured. The mean differences are small, but significant. For the liver ROI the mean SUV values are significantly lower in the PSF reconstructions, while the data from the aorta ROI is less conclusive.

4.4.1 Head

Figure 38 displays a box plot of the relative difference in mean SUV values for each head ROI between PET image reconstructions performed using the regular 3D-iterative reconstruction method and the PSF reconstruction method. Both reconstructions were performed using the CTMR head μ-map for AC, and the 3D-iterative reconstruction results have been used as the reference point according to Equation (17). The comparison of 3D and PSF reconstructions for the other μ-maps (MR and CT μ-maps) show very similar results, and can be found in Appendix C.

All ROIs except the thalamus' show a small, but significant (maximum p-value < 0.0001) decrease in mean SUV values in the PSF reconstruction relative to the 3D reconstruction. The mean decrease, for both right and left hemispheres, in the frontal, temporal, parietal, and occipital lobes; and cerebellum is 1.5 %, 1.0 %, 0.7 %, 1.4 %, and 0.9 %, respectively. The thalamus ROIs show a significant *increase* in mean SUV values of 2.7 %. As seen in Figure 17, the thalamus is the measured brain region positioned closest to the center of the skull, while the other ROIs are placed in more peripheral regions.

The similarity of the left and right hemisphere ROIs were tested using a two-sample t-test on the null hypothesis that the data in the left and right ROIs comes from independent random samples from normal distributions with equal means and equal but unknown variances; yielding a minimum p-value of 0.35, meaning that the left and right hemispheres are not significantly different.

Figure 39 shows a difference image from a single patient comparing the two reconstruction methods, where the negative values (blue) indicate voxels in which the PSF reconstruction result in lower values than the 3D reconstruction, while positive values (red) indicate voxels in which PSF result in higher values than 3D. Based on visual comparison with other patients, this image is found to be representative for the observed general trends.

The values in the difference image are in the order of - 20 % to + 10 % in tissue. There are bands of positive values (+1 % to +10 %) along the cortex and skin, while there are bands of negative values (-1 % to -20 %) along the cranial bone and outside the skin. Other than that most of the image is dominated by noise, which in the transaxial image slices displays a radial dependency. By comparing this image with that of Figure 17 one can see that the peripheral brain ROIs mostly cover negative values, while the thalamus ROIs mostly cover positive values.

The results comparing 3D and PSF from the reconstructions with the MR μ-map and CT μ-map show no obvious differences from the CTMR results, and are therefore not included here. The corresponding box plots for MR μ-map and CT μ-map can be found in Appendix C.

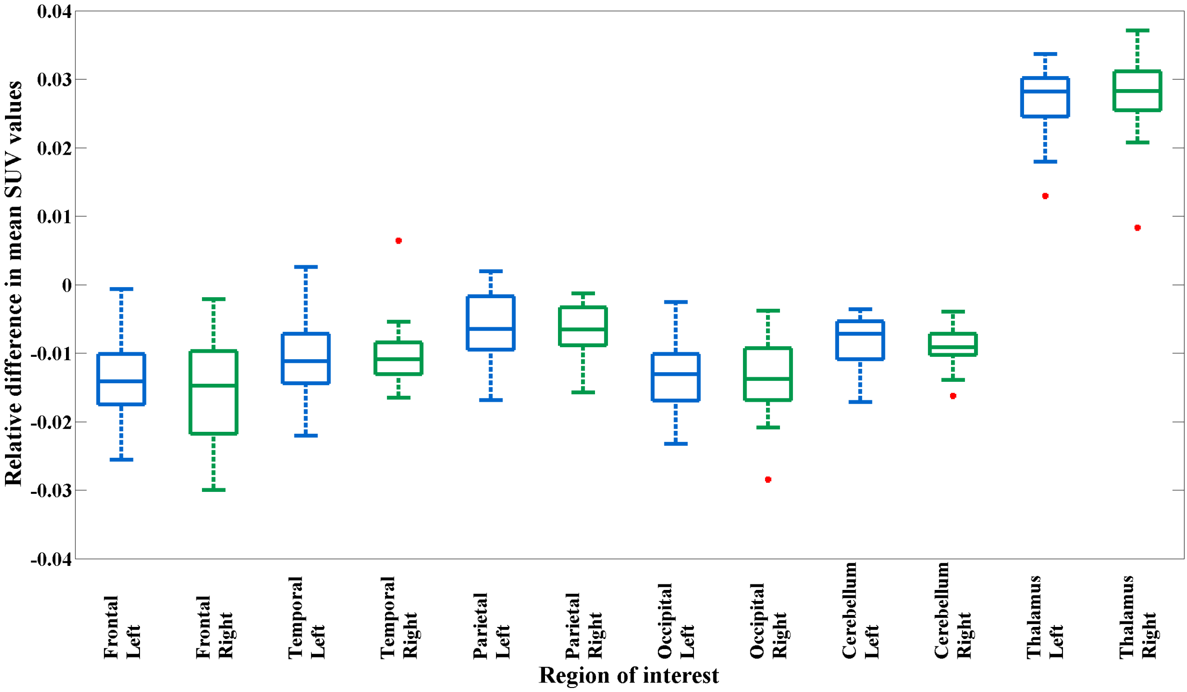


Figure 38: Box plot of the relative difference in mean SUV values between PET head images reconstructed using regular 3D-iterative reconstruction, and using the PSF reconstruction method. The CTMR μ-map was used for reconstruction, and the 3D-iterative reconstruction results have been used as the reference point. All ROIs show a significant decrease in mean SUV values, except the thalamus ROIs which show a significant increase in mean SUV. $N = 19$.

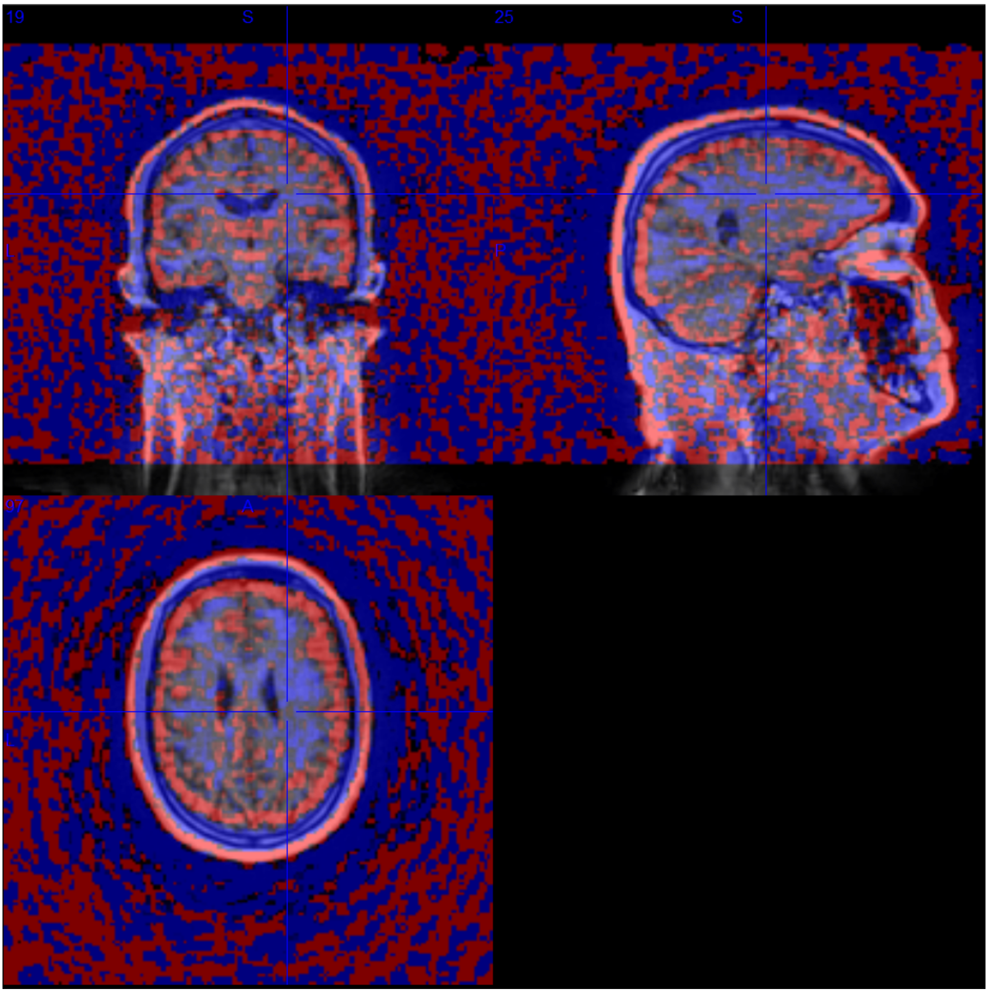
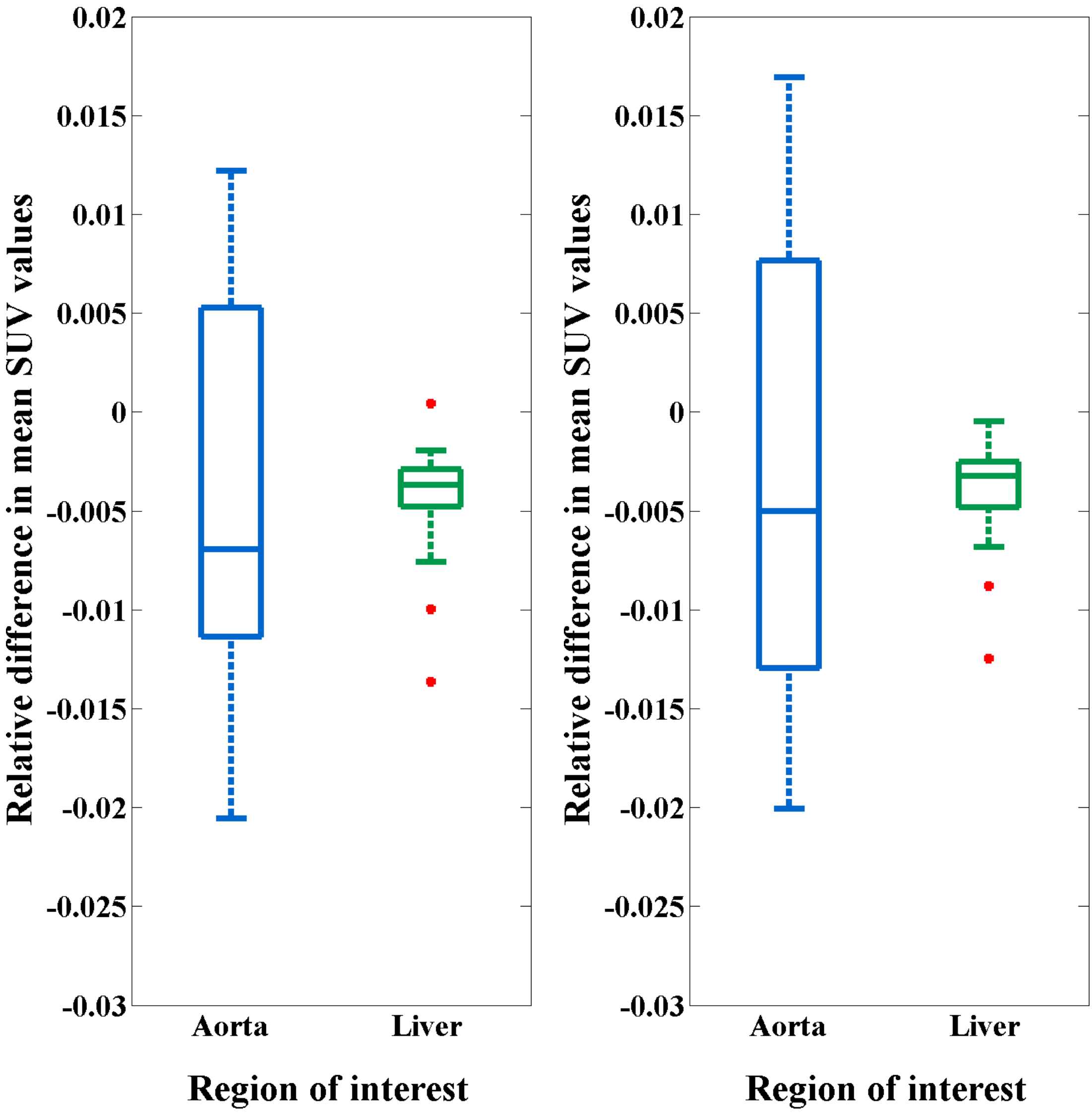


Figure 39: A difference image comparing images reconstructed using the two reconstruction methods regular 3D-iterative and PSF, overlaid on a second-echo UTE MR image. The MR image is shown as a greyscale background, while the negative and positive values of the difference image are shown in blue and red, respectively. Both images were reconstructed using the CTMR μ-map for AC, and the 3D reconstruction was used as the reference. The crosshairs are placed in the same image coordinate as in Figure 17. Blue indicates voxels in which the PSF reconstruction result in lower values than the 3D reconstruction, while red indicates voxels in which PSF result in higher values than 3D. Patient ID: PETMR023.

4.4.2 Whole-body

Figure 40 displays a box plot of the relative difference in mean SUV values for the aorta and liver ROIs between PET image reconstructions performed using the 3D-iterative reconstruction method, and the PSF reconstruction method. Both the results from the reconstructions using the MR μ-map for AC (Figure 40.1) and from using the CTMR μ-map for AC (Figure 40.2) are shown. The 3D reconstructions have been used as the reference according to Equation (17).

For the MR μ-map the mean SUV values are significantly (p-values of 0.040 for aorta and 1.566e-05 for liver) lower for both the aorta (mean of -0.52 %) and the liver ROI (mean of -0,45 %). For the CTMR μ-map the mean SUV values are significantly (p-value of 1.407e-05) lower for the liver ROI (mean of -0.41 %), while the mean SUV values in the aorta are not significantly different from zero (p-value of 0.178).



40.1 *40.2*

Figure 40: Box plot of the relative difference in mean SUV values between PET whole-body images reconstructed using regular 3D-iterative reconstruction, and using the PSF reconstruction method. Both the results from the reconstructions using the MR μ-map for AC (1) and from using the CTMR μ-map for AC (2) are shown. The 3D-iterative reconstruction results have been used as the reference point. $N = 18$.

Part IV
Discussion

In this part the results presented in Part III will be interpreted and discussed, starting with the developed method for creating μ-maps, and continuing with the rest of the results as they appear above. The results will be compared with relevant findings from other research groups, if available, and suggestions for improvement of the experiments and for further inquiries will be proposed.

1 μ-map creation

Many of the choices regarding the creation of CT-based μ-maps were justified when the methods were described in Part III. This section will mainly discuss the accuracy and robustness of the developed methods.

1.1 Image registration accuracy

As mentioned, the registrations were only evaluated through visual comparison. For rigid registrations this was considered sufficiently accurate, as there are only six degrees of freedom to consider. For non-rigid registrations (B-spline), the proper validation of registration results should include manual recognition of a large number of anatomical landmarks in both the fixed and moving image [38]. However, the recognition of these landmarks is a time-consuming process. The expected potential gain of a more accurate registration evaluation in this study was a slight increase in registration accuracy, resulting in a minor increase in PET AC accuracy. This increase in accuracy would likely be negligible in the PET image reconstruction compared to other sources of error, and thus the validation was found too time-consuming when considering the expected gain and, equally important, the time schedule of this study.

Murphy et al.[29] have proposed a semi-automatic method for using landmarks for registration evaluation, where most of the landmarks are automatically recognised after the initial manual recognition of a subset of them. To improve the non-rigid image registrations performed in this study, this method could be incorporated, either for use in the creation of each μ-map, or for validating the registration parameters on a sufficiently large patient population. The latter would demand a large initial effort, but would make the need for a validation of each subsequent registration redundant.

1.1.1 Whole-body masks

The whole-body masks were essential to achieve acceptable accuracy of the non-rigid whole-body image registrations. `segmentRibcage.m` is a versatile function, in that it has several modifiable parameters as part of its input. This facilitated tuning of the creation of masks for each patient, compensating for the inter-patient differences in MR image intensity distributions.

The function also facilitates manual editing of both the intermediate segmentations and the resulting masks, such as to compensate for significant anatomical abnormalities caused by large tumours or other malignancies. In the case of the segmentation of the lungs, two patients demanded manual editing of the segmented volume: One due to a large tumour connected to the edge of the lung, and one due to an abnormal thickness of the connective tissue between the lungs and the ribs in part of the ribcage. Several of the patients, especially those with a large abdominal circumference, demanded manual separation of the segmentations of arms and hips/abdomen, due to their proximity to each other.

The lung segmentation could be automated further by analysing the image intensity distributions of the input MR image, and setting the thresholds based on that. Automatic detection of where the shoulders narrow up to the neck could make redundant the need to manually detect a suitable neck slice, and the top and bottom of the lungs could probably also be automatically detected based on recognising z-direction-dependent changes in segmented lung volume. Separation of connected arms/hips could probably be done automatically based on shape recognition and tracing of separate areas in the stack of transaxial slices.

All in all, if the mask creation could be made more robust and fully automatic, a large number of whole-body masks and thus whole-body μ-maps could be created with no user input during the procedure.

As of now `segmentRibcage.m` is sufficiently robust to manage most patients, but is too slow to be used on a large scale, due to the need for some manual input.

1.1.2 Accuracy for obese patients

For patients with a high body fat percentage, and thus with a lot of subcutaneous fat, the skin of the patient often shifted position relative to the rest of the body between the examinations in the mCT and in the mMR. This was partly caused by the differing examination positions, but also by the coils utilised in the mMR examinations. The coils restrict the possible positions of the skin in the mMR relative to the mCT examination. For obese patients this caused the registration to perform less accurately in the abdominal area than for patients closer to normal weight. This was deemed acceptable, as the area of interest in this study was the thorax, and the observed inaccuracies in the abdomen were assumed to not have any significant affect on the PET image reconstruction of the thorax.

1.2 Head μ-maps

The head μ-maps are found to be accurate for at least the cranium, and all tissue parts in a fixed position relative to it, such as the brain, inner ears, nose, etcetera. In patients where the lower jaw is in the same position relative to the cranium in both examinations, which was observed to be the case for nearly all patients, the μ-maps are assumed to be accurate for the lower jaw as well.

All patients wore hearing protection in the mMR, but not in the mCT, leading to deformation of the ears in the mMR relative to the mCT. This led to slight inaccuracies in the CT-based μ-maps in the corresponding area, as seen in Figure 23 in the coronal and transaxial slices.

For every patient the posture of the neck changed from the mCT to the mMR. For the CT μ-maps especially this led to inaccuracies in the neck below the lower jaw, as can be seen in Figure 23 in the sagittal slices. The remaining parts of the arms also contributed to the inaccuracy of the CT μ-maps, in that they contribute falsely to the ACF of LORs in the neck region. In the CTMR μ-maps the replacement of the inaccurate parts of the CT µ-map with the corresponding parts of the MR µ-map increased their spatial accuracy.

1.3 Whole-body µ-maps

Based on visual assessment the whole-body CTMR µ-maps are found to be sufficiently spatially accurate for the entire body. The parts of the µ-map which contain CT data are more detailed than the parts from the original MR µ-map, and also provide a continuous μ-value distribution. The focus area of the visual evaluation of the registrations was the thorax, which is thus assumed to be the most accurate area of the µ-maps. Sources of error include the aforementioned differing positions of skin, along with non-stationary gas-filled cavities in the abdomen.

2 Head bone

The results show that the MR UTE µ-map grossly overestimates the amount of bone in the neck, while slightly underestimating the amount of bone in the cranium (Figure 28 and 29). The results from the overlap analysis show that even though the amount of cranial bone is similar in the MR and CT µ-maps, the bone voxels are far from completely overlapping, with more than 30 % of the bone voxels in each µ-map being unique to that µ-map (Figure 30).

This is in compliance with the work of Delso et al.[11]. They investigated the feasibility of using UTE sequences to create MR µ-maps, and observed decreased performance in the bone segmentation in the base of the skull and in the neck. They performed their investigations on a scanner from a different manufacturer (GE healthcare) than this study, and thus with a different UTE sequence, and their results are therefore not directly comparable with this study. Nonetheless, both studies indicate that bone segmentation based on UTE sequences perform worse in the neck than in the skull. Delso et al. suggest an explanation in short-T_2 structures associated with the neck muscles being classified as bone, and since this is inherent to all UTE sequences, it is also the most likely explanation for the results observed in this study.

Visual comparison of the CT-based and MR µ-maps reveal that the latter consistently misclassify tissue in the air/tissue interfaces as bone, such as those found in the ear and the paranasal sinuses. This was also observed in the study by Delso et al. who attributed them to MR artefacts known to be connected to air/tissue interfaces, such as signal ringing, partial-volume artefacts, and local magnetic field inhomogeneities[11]. These artefacts are most likely the source of the misclassification observed around air/tissue interfaces in this study as well, but they do not explain why so much (more than 30 %) of the bone in the MR μ-maps are attributed to the wrong position.

When assessing the UTE sequence for MR μ-map creation, Aitken et al.[1] observed similar misclassification of bone in the UTE-based μ-maps. They found that the misclassification was caused by eddy current artefacts due to the rapidly changing magnetic field gradients. They corrected for this by mapping the true k-space trajectories using a magnetic field camera, and demonstrated that the correction improved the bone segmentation significantly. The VB20P UTE sequence incorporates a different method for eddy current correction, described by Block et al.[7], but the results of this study indicate that the incorporated method does not perform satisfactory. Thus eddy current artefacts are a likely cause of the bulk of the spatial misclassification of bone in the MR μ-maps. A better method for correcting for eddy currents should be incorporated to improve the performance of the UTE sequence.

Another source of error in the bone segmentations, both in MR and CT-based μ-maps, is dental implants. Older patients (constituting the majority of the patients in this study) are likely to have dental implants made from metal. In MRI the magnetic susceptibility artefacts caused by (ferromagnetic) metal are manifested as signal voids in the image, and in CT metal implants lead to streaking artefacts caused by excessive beam hardening due to their general high LACs. Metal implants are thus likely to contribute to an underestimation of bone in MR μ-map, and an overestimation of bone in the CT μ-map. However, only four of the patients presented with large enough artefacts in a position where they had any effect on the measured cranial bone. Metal-related artefacts are thus not regarded as a significant source of error in this study, but major metal artefacts may have an impact in the clinic, and a method for correcting for them should be incorporated in the clinic.

When calculating the amount and localisation of bone, the bone threshold was set to >300 HU. According to the work of de Oliveira et al.[32] a threshold of 300 HU will lead to an inclusion of most of the dense trabecular bone and cortical bone. Most of the low-density trabecular bone ($\text{HU} < 200$) will not be included. There is as mentioned no direct correlation between T_2-relaxation times and CT HU units, and thus it is difficult to assess what types of bone that will be segmented by the UTE sequence. However, the 300 HU limit has been successfully employed in other studies [11], and is deemed a reasonable limit in this study as well.

3 PET image analysis

3.1 ROI placement

3.1.1 Head

The brain ROIs were placed based on manual recognition of the different brain regions, by an operator with no previous experience in brain segmentation (Figure 17). Some of the ROIs may therefore have been placed not entirely inside the target region, and thus also cover parts of neighbouring brain regions. This is however of little consequence for the presented results, as the ROIs were placed in the exact same position in each intra-patient PET image, and the same method for recognising brain regions was used on every patient. This ensures that the data is inter-patient comparable.

The size of the ROIs was chosen as large as possible, while still maintaining a size small enough to ensure some degree of localisation within the target region. Most of the ROIs were placed in peripheral brain regions, as these are the largest and easiest to recognise. The thalamus is the only central brain structure assessed with ROIs in this study.

A possible improvement on the placement of ROIs would be to (non-rigidly) register a suitable MR brain image from each patient to an atlas, such as the MNI152[1]. By placing the ROIs in the atlas, and

---

[1] Montreal Neurological Institute
http://www.bic.mni.mcgill.ca/ServicesAtlases/HomePage

performing the reverse transformation of the mentioned registrations on the ROI masks, the ROIs would be placed in the same location in each patient, given a high accuracy of the registration. This procedure was found too time consuming at the time of data analysis in this study, but will be incorporated in future work.

3.1.2 Whole-body

The activity in the aorta and liver was chosen to be measured as this is the standard routine for nuclear physicians when assessing the background activity in whole-body PET images. The liver ROIs' sample size of 1419 voxels provides data with little noise-sensitivity, while the smaller aorta ROIs (123 voxels) are more sensitive to noise. The type, amount, and most importantly the position of malignancies varied widely between patients, so measuring the activity in these would provide data of little comparability between patients. Several more ROIs could be placed throughout the thorax, in landmarks known to be located in approximately the same regions in all patients, to improve the mapping of the spatial dependency in the MR μ-maps' performance relative to the CT-based μ-maps.

3.2 MR - CTMR

As the CT-based μ-maps are used as the gold standard in this study, underestimation is defined as when the MRAC PET data returns lower activity values than the CT-based AC PET data, and vice versa for overestimation. This section only discusses the comparison of the MR and CTMR μ-maps, but for head the results were virtually equal for the comparison of MR and CT μ-maps, and thus the same discussion applies for those results.

3.2.1 Head

The results show that the UTE-based MRAC results in underestimation of mean SUV values in all brain ROIs (Figure 33). This underestimation can be explained by the amount of bone in the cranial region, which was found to be higher in the CT-based μ-maps than in the MR UTE μ-maps. When bone is segmented as soft-tissue, the ACFs of the ROI LORs passing through the relevant tissue are underestimated, resulting in underestimation of the activity in the ROI as well. The peripheral brain regions display larger underestimation than the central, which is most likely correlated with the proximity to bone.

Examination of the difference images shows a connection between the areas of the most extreme differences in PET activity (the highest and lowest values) and the areas where the MR UTE μ-map is assumed to be less accurate (Figure 34). These areas include tissues containing a lot of air/tissue interfaces such as the sinuses and the pharyngeal region; and the neck, where the MR μ-map is known to misclassify soft-tissue as bone, as discussed in Section 2.

Dickson et al.[13] performed similar investigations as this study, comparing the performance of UTE-based and CT-based AC in the head based on differences in PET image activity concentrations. They used FDG PET/CT data from GE Discovery ST and VCT PET/CT systems (GE Healthcare Systems, Waukesha, USA) as the gold standard, and assessed the performance of the previous version (VB18P) of the UTE-based MRAC method by comparing the PET/CT data to the PET/MR data acquired on a biograph mMR. They defined underestimation as where the (MRAC) PET/MR data returned lower activity values than the PET/CT data, and overestimation as the opposite.

They found significant underestimation in activity in all brain regions: 15.7 %, 15.2 %, 12.2 %, 15.2 %, 17.3 %, and 4.3 %, in the frontal, temporal, parietal, and occipital lobes, the cerebellum, and the thalamus, respectively. Qualitatively this is in concordance with the corresponding results from this study (8.4 %, 2.8 %, 5.8 %, 9.4 %, 9.1 %, and 3.2 %). By comparing different AC methods performed on the same PET data, this study utilises a more direct method of comparison, eliminating many of the variables influencing PET activity values when comparing PET data from different scanners.

Quantitatively, if accepting the results of Dickson et al. as comparable, the results of this study show that the new version of the UTE-based MRAC method (VB20P) performs better in the brain than the previous version (VB18P). This is most likely due to the additional correction methods (reduction of streak and eddy current artefacts) and the use of a template to segment bone utilised in the VB20P. The observed underestimations throughout the brain are however still of such a magnitude (up to 9 %) that the UTE-based MRAC method still is not sufficiently accurate to produce quantifiable PET images. The performance is worse closer to bone, which is as expected.

However, SUV is a measure that is known to be notoriously prone to differences in system (inter-scanner variability, image reconstruction parameters, detector sensitivity, scanner geometry, etcetera), injected dose, amount of time between injection and scanning (biologic effects which are difficult to predict), etcetera. This is the main reason for performing the reconstructions with the different µ-maps on the exact same PET data in this study. Even though Dickson et al. corrected for all the differences they were able to correct for (physical radioactive decay, normalisation, alternating examination order, etcetera), the observed differences in SUV values in their study may largely be caused by the different systems from which the PET data sets were collected, as well as the different times of biological decay (washout) of the FDG tracer. The results in this study are not prone to such systematic errors, and differences are solely caused by differences in the µ-maps.

Most studies that compare CT-based and MR-based AC methods, compare PET data from a PET/CT scanner corrected with CT-based AC, with PET data from a PET/MR scanner corrected with MR-based AC. It is therefore difficult to find results in the literature that are comparable to this study.

3.2.2 Whole-body

The observed underestimation in mean SUV values in the liver (3.8 %), when reconstructing with the MR µ-maps compared with the CT-based µ-maps, shows that the exclusion of bone in MRAC has a significant effect on PET activity quantification in the abdomen as well. The underestimation is most likely caused by decreased ACFs when excluding segmentation of bone in the MR µ-maps.

The results show no significant over- or underestimation in the aorta, but this is likely due to the low size of the ROI and thus high noise-sensitivity. To increase the possibility of observing any significant difference in the aorta the patient population must be increased, as the size of the ROI is restricted due to the size of the aorta.

As there is significant underestimation of PET activity in the liver ROI, which was placed relatively far from any large bone structures, additional mapping of the underestimation closer to bone structures should be performed. This due to, as observed in the head, a tendency for ROIs closer to bone to display more severe estimation.

As for the head there are few studies in the literature that are comparable to this study. Izquierdo-Garcia et al.[20] investigated the performance of whole-body MRAC versus CT-based AC on a Phillips PET/MR scanner. The MRAC method on that scanner is similar to the Dixon-based whole-body MRAC method on the mMR in that it attributes roughly the same LAC values to soft-tissue, air, and lung tissue, but it only segments those three tissue classes, and the utilised MR sequence (atMR) is not a Dixon sequence. Their results were still similar to this study, with less than 10 % underestimation in SUV values in the PET images reconstructed with the MRAC method compared with the CT-based AC method, for the entire thorax/abdomen, including the liver. The underestimation observed in the liver in this study indicates that there is underestimation in the the rest of the body as well.

3.3 CT - CTMR (head only)

The CT and CTMR head μ-maps were compared to assess the potential equality of the two CT-based head μ-maps in the brain region. The reason for making both CT and CTMR μ-maps was the differing neck postures between the mCT and mMR examinations and the obvious amounts of false bone in the MR μ-map: Either the μ-map would be anatomically spatially accurate, in that it would place the neck in the correct position (CT + MR); or it would place the neck in the wrong position, but give a correct amount of bone in the neck (CT only). The choice between these is a choice between two suboptimal alternatives.

The results show that the two μ-maps do *not* perform identically in the brain, but display small, but significant differences in mean SUV value for the different ROIs (Figure 36 and 37). The only differences between the μ-maps are the CT arm stubs, neck, and the minor differences in the outer ears and around the cheeks. These differences must therefore be the cause of the observed differences between the PET images. The conclusion is that the PET images might be significantly influenced by what initially is assumed to be minor inaccuracies in the μ-maps. This must be taken into account when designing CT-based μ-maps, both for phantoms and patients. In the setting of this study, alterations could have been made in the method for positioning the patients in the two scanners, such as to achieve identical neck postures, and thus improved the CT-only μ-maps. A more complete removal of the arms would also improve the CT μ-maps, and this will be performed in future studies.

3.4 3D - PSF

3.4.1 Head

The results from the relative differences in mean SUV values show that, compared with the regular 3D algorithm, the PSF reconstruction algorithm consistently results in lower SUV values in the peripheral brain regions, and in higher SUV values in the thalamus, a central brain region (Figure 38). This is in concordance with what is observed in the difference image and indicates that the difference image indeed is representative for the entire population (Figure 39). Comparing the image with the placement of the ROIs explains the results from the ROIs, in that all ROIs except for the thalamus were by chance placed in areas where the PSF yield lower values than the 3D algorithm (Figure 17).

Thus the results show that the PSF algorithm is prone to edge artefacts, manifested throughout the brain as alternating undershoots and overshoots in activity compared to the regular 3D algorithm. Bai & Esser[3] investigated the quantitative effects of edge artefacts on images reconstructed with PSF (phantoms scanned on a biograph mCT), and found that the PSF algorithm resulted in significant overshoots in activity compared to the regular 3D reconstruction method. This is in concordance with the results of this study.

The observed differences attributed to the PSF algorithm are large enough to have clinical impact. This is an important factor to consider when quantitatively assessing PET images in a clinical setting, and the performance of the PSF should be further examined to determine the consistency of the observed performance. This could be done by registering all patients' PET images to a common template, and create an average image of the resulting difference images. This was not done in this study due to time constraints. Any dependency on the number of reconstruction iterations, resolution, etcetera should also be investigated.

3.4.2 Whole-body

The results from the aorta ROIs are inconclusive regarding the statistical significance of the observed decrease in mean SUV values in reconstructions performed with the PSF compared to the regular 3D reconstruction algorithm. With only the results from the MR μ-map being found to be barely significantly different from zero, and the results from the CTMR μ-map being found to not be significantly different from zero, the conclusion is that the patient population is too small to observe any significant change in mean SUV in the aorta between the two reconstruction methods. An increase in ROI size would likely reduce the variance of the data, but a larger ROI would be more difficult to place inside the aorta without including any parts of the vessel walls.

The liver ROIs display a much lower variance than the aorta ROIs, most likely due to the much larger ROI sample size. The observed decrease in mean SUV in PSF compared to 3D is statistically significant for both μ-maps, but it is of a minute magnitude. Further assessment of the PSF in whole-body images should examine the potential significance of edge artefacts, as those observed in the brain.

Part V
Conclusion

Compared with CT-based PET AC methods, the VB20P version of the UTE-based head MRAC method provided with the biograph mMR grossly overestimates the amount of bone in the neck region, while it slightly underestimates the amount of bone in the cranial region. The method especially struggles to correctly classify tissue in air/tissue interfaces. The misclassification of tissue is most likely caused by a combination of artefacts related to eddy currents, signal ringing, partial-volume artefacts, and local magnetic field inhomogeneities. The misclassification of bone as soft-tissue leads to an underestimation of PET activity in the brain, and also in the liver, in compliance with what other studies have found. The observed underestimation in the liver using Dixon-based μ-maps indicates that there is underestimation in the rest of the body as well. The current MR-based AC methods of the biograph mMR are therefore not sufficiently accurate to be utilised for quantification in PET imaging.

Compared to the regular 3D reconstruction, the PSF reconstruction algorithm results in significant edge artefacts throughout the brain. These artefacts result in a spatially dependent change in activity, negative for some, positive for others. The effect of these edge artefacts must be taken into account in clinical settings. For the rest of the body there is a minute difference in the liver, but the edge artefacts should be mapped to assess the PSF's clinical performance in these regions as well.

The comparison of the two different CT-based head μ-maps demonstrates that inaccuracies assumed to be of minimal influence may still have significant effects, emphasising the need to minimise all possible inaccuracies when creating modified CT-based μ-maps.

The developed methods for creating CT-based μ-maps have potential for improvement, especially in the areas of (non-rigid) registration and registration validation, but also in their level of automation, which could be significantly increased.

Part VI

Appendices

The work with this master thesis resulted in more than 2500 lines of MATLAB code, excluding empty lines; and in tens of different `elastix` registration parameter files. A few of these MATLAB scripts/functions and registration parameter files can be found below.

A MATLAB functions & scripts

A.1 calculateSUV.m

Contents

- Gather necessary values from header data.
- Perform calculations.
- Structure returned data.

```
function data = calculateSUV(PETvolume,mask,meta,missingDose,...
    missingInjTime,missingSex,missingHeight,missingWeight)
%CALCULATESUV - Calculate SUV values in PET image.
%   DATA=CALCULATESUV(PETVOLUME,MASK,META,MISSINGDOSE,
%   MISSINGINJTIME,MISSINGSEX,MISSINGHEIGHT,MISSINGWEIGHT)
%   calculates the SUV values in PETVOLUME in the ROI defined by
%   MASK. META contains a struct of the form returned by
%   DICOMINFO.m from one of the DICOM files in the PET image
%   series. MISSING* are optional inputs containing data to
%   replace the corresponding values in META.

assert(length(PETvolume(:))==length(mask(:)))
```

Gather necessary values from header data.

```
% Optionally replace data with MISSING*.
seriesTime=strTime2numTime(meta.SeriesTime); % [s since midnight]
patientSex=meta.PatientSex; % string
if exist('missingSex','var') && ~isempty(missingSex)
    patientSex=missingSex;
```

```
end
if exist('missingHeight','var') && missingHeight>0
    patientHeight=missingHeight;
else
    patientHeight=meta.PatientSize; % [m]
end
if exist('missingWeight','var') && missingWeight>0
    patientWeight=missingWeight*1000;
else
    patientWeight=meta.PatientWeight*1000; % [g]
end
sliceThickness=meta.SliceThickness; % [mm]
pixelSpacing=meta.PixelSpacing; % [mm]
injectionTime=strTime2numTime(...
    meta.RadiopharmaceuticalInformationSequence...
    .Item_1.RadiopharmaceuticalStartTime); % [s since midnight]
if exist('missingInjTime','var') && missingInjTime>0
    injectionTime=missingInjTime;
end
injectedDose=meta.RadiopharmaceuticalInformationSequence...
    .Item_1.RadionuclideTotalDose; % [Bq]
if exist('missingDose','var') && missingDose>0;
    injectedDose=missingDose;
end
timeHalflife=meta.RadiopharmaceuticalInformationSequence...
    .Item_1.RadionuclideHalfLife; % [s]
```

Perform calculations.

```
% Radioactive decay correction factor.
decayTime=seriesTime-injectionTime; % [s]
decayCorrFactor=exp(log(2)*decayTime/timeHalflife);
% SUVbw correction factor.
bodyWeightCorrFactor=patientWeight/injectedDose;
% SUVlbm correction factor.
switch patientSex
    case 'M'
        LBMcorrFactor=1000*(1.10*patientWeight/1000-128*...
```

```
            (patientWeight/1000/patientHeight/100)^2)/...
            injectedDose;
    case ’F’
        LBMcorrFactor=1000*(1.07*patientWeight/1000-148*...
            (patientWeight/1000/patientHeight/100)^2)/...
            injectedDose;
    otherwise
        error(’Sex not recognised!’)
end

% Calculate SUV.
VOI=PETvolume(mask>0);
SUVbw=decayCorrFactor*bodyWeightCorrFactor*VOI;
SUVlbm=decayCorrFactor*LBMcorrFactor*VOI;
```

Structure returned data.

```
% Return VOI, VOI volume, voxel volume, and all calculated SUV
% values; along with the name of the image series.
data.SeriesName=strrep(strrep(meta.SeriesDescription,’*’,’’),...
    ’ ’,’_’);
data.VOI=VOI;
data.VOIvolume=nnz(mask)*pixelSpacing(1)*pixelSpacing(2)*...
    sliceThickness;
data.VoxelVolume=pixelSpacing(1)*pixelSpacing(2)*sliceThickness;

data.SUVbw=SUVbw;
data.SUVbwMin=min(SUVbw(:));
data.SUVbwMax=max(SUVbw(:));
data.SUVbwMean=mean(SUVbw(:));

data.SUVlbm=SUVlbm;
data.SUVlbmMin=min(SUVlbm(:));
data.SUVlbmMax=max(SUVlbm(:));
data.SUVlbmMean=mean(SUVlbm(:));
```

A.2 createmask.m

```
function mask = createmask(iva,thresholds,radius)
%CREATEMASK - Create a binary mask from a 3D image volume.
%   MASK = CREATEMASK(IVA,THRESHOLDS,RADIUS) returns a mask
%   created from the image volume array IVA, based on the
%   thresholds given as input. If RADIUS is given the mask will
%   be dilated using a sphere with the given radius. The function
%   firsts sets the lower and upper thresholds based on input,
%   and then creates an initial binary mask, which is
%   (optionally) dilated before all holes are filled using the
%   function fillMask.

% Initialize threshold variables.
if exist('thresholds','var') && ~isempty(thresholds)
    if length(thresholds)<2
        lower=thresholds;
        upper=max(iva(:));
    else
        lower=thresholds(1);
        upper=thresholds(2);
    end
else
    lower=40;
    upper=max(iva(:));
end

% Create mask
mask=iva;
mask(:)=0;
mask(iva>=lower&iva<=upper)=1;

% Optional: dilate mask.
if exist('radius','var') && radius
    [x,y,z]=ndgrid(-radius:radius);
    se=strel(sqrt(x.^2+y.^2+z.^2)<=radius);
    mask=imdilate(mask,se);
end
```

```
% Fill holes inside the mask.
mask=fillMask(mask);
```

A.3 createmodmumap.m

```
function createmodmumap(IVA,dirIN,dirOUT)
%CREATEMODMUMAP - Create a new DICOM image series with the 3D
%   image IVA. CREATMODMUMAP(IVA,DIRIN,DIROUT) creates a DICOM
%   series in DIROUT, identical to the one in DIRIN, with the
%   exception of PixelData, which is filled with corresponding
%   slices from the 3D image volume array IVA. IVA must be
%   oriented in the same direction as in the original DICOM
%   series, and will for all intents and purposes be a modifed
%   version of an image volume created from the DICOM series in
%   DIRIN.

frames=size(IVA,3);
createORempty(dirOUT) % Create or empty folder.
cd(dirIN)
directory=dir;
files=directory(3:end);

h=waitbar(0,'Writing images...');

% The code commented out below is left in place to keep the
% options for replacing more than just the Pixeldata, such as the
% SeriesNumber etc. The philosophy at present is to change as
% little as possible, and 'Don't change a winning team... So long
% as it's winning fast enough.'
% t=now;
% cl=datestr(t,13);
% seriesNumber=501;
% seriesInstanceUID=dicomuid;
for i=1:frames
    elements=ReadDicomElementList(files(i).name);
%     SOPInstUID=dicomuid;
%     for j=1:length(elements)
```

```
%         if strcmpi(strcat(elements(j).group,...
%                 elements(j).number),’00020003’)
%             elements(j).data=SOPInstUID;
%             elements(j).length=length(SOPInstUID);
%         elseif strcmpi(strcat(elements(j).group,...
%                 elements(j).number),’00080018’)
%             elements(j).data=SOPInstUID;
%             elements(j).length=length(SOPInstUID);
%         elseif strcmpi(strcat(elements(j).group,...
%                 elements(j).number),’00200011’)
%             elements(j).data=seriesNumber;
%             elements(j).length=length(seriesNumber);
%         elseif strcmpi(strcat(elements(j).group,...
%                 elements(j).number),’0020000E’)
%             elements(j).data=seriesInstanceUID;
%             elements(j).length=length(seriesInstanceUID);
%         end
%     end
    elements=imagearrayinsertion(IVA(:,:,i),elements);
    modfilename=files(i).name;
%     modfilename(31:34)=strcat(’0’,seriesNumber);
%     modfilename(41:59)=strcat(datestr(t,10),’.’,...
%         datestr(t,5),’.’,datestr(t,7),’.’,cl(1:2),’.’,...
%         cl(4:5),’.’,cl(7:8));
    h=waitbar(i/frames,h,sprintf(’Writing image %u/%u.’,i,...
        frames));
    cd(dirOUT)
    WriteDicomElementList(elements,modfilename);
    cd(dirIN)
end
close(h)
up; % Funtion that returns the current directory to the userpath.
```

A.4 CT2MRmumap_head.m

The script creating head μ-maps containing only CT data is very similar to this script, with the exception of the fusion of CT and MR μ-maps, which is replaced by a simple removal of most of the visible arms in the CT image.

VI

```
%CT2MRMUMAP_HEAD is a script for creating CT-based head mumaps.
%   The script runs through five successive steps. Step 1
%   consists of manual selection of the needed starting images
%   (DICOM series), preparation of output folders, conversion of
%   DICOM series to 3D nifti, and initializing various variables
%   containing file names. Step 2 creates a fusion mask, with an
%   option for manual correction in case of image artefacts
%   caused by metallic dental implants. Step 3 asks for manual
%   graphical input on where to focus the registration and on how
%   to crop the CT image to ease the registration; before
%   performing two successive registrations, the latter with a
%   fixed-image mask. Step 4 creates a fusion mask, manually
%   edited to only contain accurate regions. Step 5 converts the
%   final CT image to LAC values and fuses the resulting CT mumap
%   with the MR mumap to create the CTMR mumap, which is
%   subsequently saved as a DICOM directory.

clc
% Variable initialization:
% Global parameters:
dirElxPar='C:\D\elx_par\CT2MRmumap\';
fileRegRigidCT=[dirElxPar 'rigidHeadCT.txt'];
fileRegRigidCT2=[dirElxPar 'rigidHeadCT2.txt'];

% Change/tune:
mainpath='G:\D\CTMRmumaps_intermediate\head\PETMR036\';
step=7;
valHUair=-1024;
dirData='G:\D\';

thresholdMR=40;
radiusMR=2;

thresholdCT=-400;
radiusCT=0;

% Parameters set based on those above:
dirNII=[mainpath 'nifti\'];
```

```
dirElxOut=[mainpath 'elxOutput\'];
dirElxOut_r2MRCT=[dirElxOut 'r2MRCT\'];
dirElxOut_r2MRr2MRwmCT=[dirElxOut 'r2MRr2MRwmCT\'];

switch step
    case 1
        dirCT=uigetdir(dirData,'Select CT scan, AC-CT.');
        dirMR=uigetdir(dirData,'Select MR UTE scan, second TE.');
        dirDataMR=fileparts(dirMR);
        dirMRmumap=uigetdir(dirDataMR,'Select MR UTE mumap.');
        dirCTMRmumap=fileparts(dirMRmumap);
        dirCTMRmumap=...
            [dirCTMRmumap filesep 'CTMRmumap_head' filesep];

        cd(dirCT)
        temp=dir;
        temp=dicominfo(temp(5).name);
        kVp=temp.KVP;
        up

        createORempty(dirNII)
        createORempty(dirElxOut_r2MRCT)
        createORempty(dirElxOut_r2MRr2MRwmCT)

        % Convert DICOM to NIFTI:
        MRniifiles=dcm2nii(dirMR,dirNII,'MR');
        CTniifiles=dcm2nii(dirCT,dirNII,'CT');
        MRmumapniifiles=dcm2nii(dirMRmumap,dirNII,'MRmumap');

        fileMR=[dirNII MRniifiles(end).name];
        fileCT=[dirNII CTniifiles(1).name];
        fileMRmumap=[dirNII MRmumapniifiles(1).name];
    case 2
        % Create fusion mask:
        MR=load_nii(fileMR);
        mask=createmask(MR.img,thresholdMR,radiusMR);
        MRmaskFuse=MR;
        MRmaskFuse.img=mask;
```

```
    fileMRmaskFuse=[dirNII 'maskFuse' MRniifiles(end).name];
    save_nii(MRmaskFuse,fileMRmaskFuse)
    viewWoverlay(fileMR,fileMRmaskFuse,true)

    % Ask if the mask was edited and saved using the MRIcron
    % option ‘create SPM5 masks' where the mask is saved as
    % ‘l*.nii'
    maskEdited=input(...
        'Did you edit and save the mask? (y/n), [n]: ','s');
    if strcmpi(maskEdited,'y')
        fileMRmaskFuse =[dirNII 'l' MRniifiles(end).name];
    end

case 3
    % Create registration mask from the fusion mask:
    MRmaskFuse=load_nii(fileMRmaskFuse);
    img=permute(MR.img,[3 2 1]);
    h=figure('Name',...
        'Choose where to cut the registration mask.');
    imshow(img(:,:,end/2),[])
    [~,slice]=ginput(1);
    close(h)
    slice=round(slice);
    MRmaskReg=MRmaskFuse;
    MRmaskReg.img(:,:,1:slice)=0;
    fileMRmaskReg=[dirNII 'maskReg' MRniifiles(end).name];
    save_nii(MRmaskReg,fileMRmaskReg)

    % Crop the CT image:
    CT=load_nii(fileCT);
    img=permute(CT.img,[3 2 1]);
    h=figure('Name',...
        'Choose where to cut, cut to bottom is kept.');
    imshow(img(:,:,end/2),[])
    [~,slice]=ginput(1);
    close(h)
    slice=round(slice);
    img=CT.img(:,:,slice:end);
```

```
        voxelsize=CT.hdr.dime.pixdim(2:4);
        nii=make_nii(img,voxelsize);
        fileCThead=[dirNII 'CThead.nii'];
        save_nii(nii,fileCThead)

        % Perform registrations:
        system(sprintf('elastix -f %s -m %s -out %s -p %s',...
            fileMR,fileCThead,dirElxOut_r2MRCT,fileRegRigidCT));
        filer2MRCT=[dirElxOut_r2MRCT 'result.0.nii'];
        system(sprintf(...
            'elastix -f %s -m %s -out %s -p %s -fMask %s',...
            fileMR,filer2MRCT,dirElxOut_r2MRr2MRwmCT,...
            fileRegRigidCT2,fileMRmaskReg));
        filer2MRr2MRwmCT=[dirElxOut_r2MRr2MRwmCT 'result.0.nii'];

        % Visually control registrations:
%         viewWoverlay(fileMR,fileMRmaskFuse)
        viewWoverlay(fileMR,filer2MRCT)
        viewWoverlay(fileMR,filer2MRr2MRwmCT)
        viewWoverlay(filer2MRr2MRwmCT,fileMRmaskFuse,true)

    case 4
        % If the MR fusion mask does not cover the CT image
        % sufficiently, create a fusion mask from the CT image
        % and edit it manually:
        useCTmask=false;
        maskEdited=input(...
            'Did you edit and save the mask? (y/n), [n]: ','s');
        if strcmpi(maskEdited,'y')
            fileMRmaskFuse =...
                [dirElxOut_r2MRr2MRwmCT 'lresult.0.nii'];
        else
            CT=load_nii(filer2MRr2MRwmCT);
            mask=createmask(CT.img,thresholdCT,radiusCT);
            CTmaskFuse=CT;
            CTmaskFuse.img=mask;
            fileCTmaskFuse=...
                [dirElxOut_r2MRr2MRwmCT 'maskFuseresult.0.nii'];
```

```
            save_nii(CTmaskFuse,fileCTmaskFuse)
            system(sprintf(...
                'mricron %s -o %s -c blue -v %s -t -1 -x',...
                filer2MRr2MRwmCT,fileMRmumap,fileCTmaskFuse));
            fileCTmaskFuse=...
                [dirElxOut_r2MRr2MRwmCT 'lresult.0.nii'];

            useCTmask=true;
        end

    case 5
        % Create and save CTMR head mumap in DICOM directory:
        CT=load_nii(filer2MRr2MRwmCT);
        MRmumap=load_nii(fileMRmumap);
        CTMRmumap=MRmumap;
        if useCTmask
            mask= load_nii(fileCTmaskFuse);
        else
            mask=load_nii(fileMRmaskFuse);
        end
        CTMRmumap.img=HU2PETLAC(CT.img,kVp,valHUair);
        CTMRmumap.img=fuseVolumes(CTMRmumap.img,MRmumap.img,...
            mask.img);
        fileCTMRmumap=[dirNII 'CTMRmumap.nii'];
        save_nii(CTMRmumap,fileCTMRmumap)
        viewWoverlay(fileCTMRmumap,fileCTmaskFuse)
        CTMRmumap=flip(flip(permute(CTMRmumap.img,[2 1 3]),1),2);
        createmodmumap(CTMRmumap,dirMRmumap,dirCTMRmumap)

    otherwise
        disp('Failsafe')
end
```

A.5 CT2MRmumap_wb.m

```
%CT2MRMUMAP_WB is a script for creating CT-based whole-body
%   mumaps. The script runs through five successive steps. Step 1
%   consists of manual selection of the needed starting images
```

```
%   (DICOM series), preparation of output folders, conversion of
%   DICOM series to 3D nifti, and initialising various variables
%   containing file names. Step 2 creates the registration mask
%   for the non-rigid registration. Step 3 performs the rigid
%   registration of the CT image to the MR image, and then
%   removes the patient bed from the resulting image. Step 4
%   performs the non-rigid registration. Step 5 converts the
%   resulting CT image to LAC values and fuses the CT mumap with
%   the MR mumap using the registration mask as the boundary,
%   before saving the mumap using the MR mumap DICOM series as a
%   template. The mask may be manually edited between the running
%   of step 4 and step 5, to exclude bodyparts not successfully
%   transformed to accurate positions (the head).

clc
% Variable initialization:
% Global:
dirElxPar='C:\D\elx_par\CT2MRmumap\';
fileRegRigidCT=[dirElxPar 'rigidWBCT.txt'];
fileRegBsplineCT=[dirElxPar 'BsplineWBCT.txt'];

% Change/tune:
dirRoot='G:\D\CTMRmumaps_intermediate\WB\PETMR013\';
step=7;
valHUair=-1024;
dirData='G:\D\';

thresholdsMR=[0 110 70 70 50 50];
radiusMR=6;
overlapMR=3;

thresholdsCT=-200;
radiusCT=1;

dirNII=[dirRoot 'nifti\'];
dirElxOut=[dirRoot 'elxOutput\'];
dirElxOut_r2MRCT=[dirElxOut 'r2MRCT\'];
dirElxOut_r2MRnr2MRCT=[dirElxOut 'r2MRnr2MRCT\'];
```

```
filer2MRCT=[dirElxOut_r2MRCT 'result.0.nii'];
filer2MRnr2MRCT=[dirElxOut_r2MRnr2MRCT 'result.0.nii'];
fileCTMRmumap=[dirNII 'CTMRmumap.nii'];
fileDataBackup=[dirNII 'dataBackup.mat'];
fileTemp=[dirNII 'temp.nii'];

switch step
    case 1
        if exist(fileDataBackup,'file')
            load(fileDataBackup);
        else
            dirCT=uigetdir(dirData,'Select CT scan, AC-CT.');
            dirMR=uigetdir(dirData,...
                'Select MR Dixon scan, composed.');
            dirDataMR=fileparts(dirMR);
            dirMRmumap=uigetdir(dirDataMR,...
                'Select MR Dixon mumap, composed.');
            dirCTMRmumap=fileparts(dirMRmumap);
            dirCTMRmumap=...
                [dirCTMRmumap filesep 'CTMRmumap_WB' filesep];

            cd(dirCT)
            temp=dir;
            temp=dicominfo(temp(5).name);
            kVp=temp.KVP;
            up

            createORempty(dirNII)
            createORempty(dirElxOut_r2MRCT)
            createORempty(dirElxOut_r2MRnr2MRCT)

            % Convert DICOM to NIFTI:
            MRniifiles=dcm2nii(dirMR,dirNII,'MR');
            CTniifiles=dcm2nii(dirCT,dirNII,'CT');
            MRmumapniifiles=dcm2nii(dirMRmumap,dirNII,'MRmumap');

            fileMR=[dirNII MRniifiles(end).name];
```

```
        fileCT=[dirNII CTniifiles(1).name];
        fileCTmask=[dirNII 'mask' CTniifiles(1).name];
        fileMRmumap=[dirNII MRmumapniifiles(1).name];

        save(fileDataBackup);
    end
case 2
    % Create registration mask:
    [~,fileMRmask]=segmentRibcage(fileMR,thresholdsMR,...
        radiusMR,overlapMR);
    viewWoverlay(fileMR,fileMRmask,true) % View in MRIcron.

case 3
    % Perform rigid registration:
    system(sprintf('elastix -f %s -m %s -out %s -p %s',...
        fileMR,fileCT,dirElxOut_r2MRCT,fileRegRigidCT));
    viewWoverlay(fileMR,filer2MRCT)

    % Mask out the patient bed from CT:
    CT=load_nii(filer2MRCT);
    CTmask=CT;
    CTmask.img=createmask(CT.img,thresholdsCT,radiusCT);
    CC=bwconncomp(CTmask.img,6);
    indexes=CC.PixelIdxList;
    lengths=cellfun('length',indexes);
    [~,indexLargest]=max(lengths);
    lengths(indexLargest)=0;
    [~,indexLargest(2)]=max(lengths);
    CTmask.img(:)=0;
    CTmask.img(indexes{indexLargest(1)})=1;
    save_nii(CTmask,fileCTmask)
    viewWoverlay(filer2MRCT,fileCTmask)

case 4
    % Peform non-rigid registration:
    system(sprintf(...
'elastix -f %s -m %s -out %s -p %s -fMask %s -mMask %s',...
        fileMR,filer2MRCT,dirElxOut_r2MRnr2MRCT,...
```

```
            fileRegBsplineCT,fileMRmask,fileCTmask));
        viewWoverlay(fileMR,filer2MRnr2MRCT)

    case 5
        % Convert CT image and fuse with MR mumap to create the
        % CTMR mumap.
        CT=load_nii(filer2MRnr2MRCT);
        MRmumap=load_nii(fileMRmumap);
        mask=load_nii(fileMRmask);
        mask=mask.img;
        CTMRmumap=MRmumap;
        CTMRmumap.img=HU2PETLAC(CT.img,kVp,valHUair);
        CTMRmumap.img=...
            fuseVolumes(CTMRmumap.img,MRmumap.img,mask);
        save_nii(CTMRmumap,fileCTMRmumap)
        viewWoverlay(fileCTMRmumap,fileMRmask)

        % Re-orient image to match DICOM image series and create
        % CTMR mumap
        % DICOM directory.
        CTMRmumap.img=flip(flip(flip(permute(CTMRmumap.img,...
            [3 1 2]),1),2),3);
        createmodmumap(CTMRmumap.img,dirMRmumap,dirCTMRmumap)

    otherwise
        disp('Failsafe')
end
```

A.6 extractimagearray.m

```
function image_array = extractimagearray(el)
%EXTRACTIMAGEARRAY - Create a 2D image from the PixelData in EL.
%   IMAGE_ARRAY = EXTRACTIMAGEARRAY(EL) returns a 2D image
%   created from the pixel data in EL. EL is a list of elements
%   of the form returned by READDICOMELEMENTLIST. IMAGE_ARRAY is
%   rescaled using corresponding image tags. This function is
%   taylored to extract mumap images from MR and CT mumaps, but
%   should also work well for other DICOM images.
```

```
% Extract the necessary DICOM tags from the header:
for i=1:length(el)
    if strcmpi(strcat(el(i).group,el(i).number),'00080008')
        imagetype=el(i).data;
    elseif strcmpi(strcat(el(i).group,el(i).number),'00280010')
        rows=el(i).data;
    elseif strcmpi(strcat(el(i).group,el(i).number),'00280011')
        columns=el(i).data;
    elseif strcmpi(strcat(el(i).group,el(i).number),'00281052')
        rescaleintercept=str2double(el(i).data);
    elseif strcmpi(strcat(el(i).group,el(i).number),'00281053')
        rescaleslope=str2double(el(i).data);
    elseif strcmpi(strcat(el(i).group,el(i).number),'7fe00010')
        pixeldata=el(i).data;
    end
end

if strcmpi(imagetype(end-5:end),'AC_MAP')
    CTmumap=true;
end

% Cast the pixel data to the correct type:
pixeldata=uint8(pixeldata);
image_vector=typecast(pixeldata,'int16');

% Rescale the pixel data:
rsi=0;
rss=1;
if exist('rescaleintercept','var')
    rsi=rescaleintercept;
end
if exist('rescaleslope','var')
    rss=rescaleslope;
end
if exist('CTmumap','var') && CTmumap
    rss=rss*10000;
end
```

```
image_vector=int16(double(image_vector)*rss+rsi);

% Reshape the image vector to a 2D image:
image_array=int16(zeros(rows,columns));
counter=0;
for i=1:rows
    for j=1:columns
        counter=counter+1;
        image_array(i,j)=image_vector(counter);
    end
end
```

A.7 fillmask.m

```
function filledMask = fillMask(mask)
%FILLMASK - Fill holes in the 3D binary image MASK.
%   FILLEDMASK = FILLMASK(MASK) fills the 3D binary image in all
%   three planes using imfill and returns it as FILLEDMASK.
filledMask=mask;
for i=1:size(filledMask,3)
    filledMask(:,:,i)=imfill(filledMask(:,:,i),'holes');
end
filledMask=permute(filledMask,[2 3 1]);
for i=1:size(filledMask,3)
    filledMask(:,:,i)=imfill(filledMask(:,:,i),'holes');
end
filledMask=permute(filledMask,[2 3 1]);
for i=1:size(filledMask,3)
    filledMask(:,:,i)=imfill(filledMask(:,:,i),'holes');
end
filledMask=permute(filledMask,[2 3 1]);
```

A.8 fuseVolumes.m

```
function fusedVolume = fuseVolumes(iva1,iva2,mask)
%FUSEVOLUMES - Fuse two image volumes using MASK as boundary.
%   FUSEDVOLUME = FUSEVOLUMES(IVA1,IVA2,MASK) fuses the two
%   equally sized arrays IVA1 and IVA2 using the binary image
```

```
%   MASK (equal size) as a boundary. Where MASK has value 1,
%   values from IVA1 is used, where MASK has value 0, values from
%   IVA2 is used.
imask=int16(~mask);
mask=int16(mask);
fusedVolume=iva1.*mask+iva2.*imask;
```

A.9 HU2PETLAC.m

```
function CTmumap = HU2PETLAC(CT,kVp,valHUair,scalingFactor)
%HU2PETLAC - Rescale CT HU image to 511 keV LAC.
%   CTMUMAP = HU2PETLAC(CT,KVP,VALHUAIR,CORRFACTOR) rescales the
%   CT image CT given in Hounsfield Units (HU), to a CT-based PET
%   mumap of 511 keV linear attenuation coefficients (LACs). KVP
%   is the peak voltage of the X-ray tube during the acquisition
%   given in kV. VALHUAIR is the HU value of air (either -1024 or
%   -1000), and SCALINGFACTOR is a scaling factor used to scale
%   the LAC values to the same order as those given in MR mumaps.

% This function takes a CT image volume in HU as minimum input,
% and converts it to an image volume with 511 keV LAC's scaled up
% with a factor of 10000, as is the case with MR mumaps.

% Default values:
if ~exist('kVp','var')
    kVp=120;
end
if ~exist('valHUair','var')
    valHUair=-1000;
end
if ~exist('corrFactor','var')
    scalingFactor=10^4;
end

% The function below loads the scaling factors based on the paper
% of Carney et al. 2006. The default (kVp=120 kV) is a=5.1e-5,
% b=4.71e-2, break point (BP) at 47 HU, and slope below BP (sbBP)
% equal to 9.6e-5.
```

```
[a,b,BP,sbBP] = getcoeffHU2PETLAC(kVp);

% During the registration some voxels acquire values below that
% of air.
CT(CT<valHUair)=valHUair;

CTmumap=CT;
CTmumap(CT<BP)=int16(scalingFactor*sbBP*(double(CT(CT<BP))+...
    abs(valHUair)));
CTmumap(CT>=BP)=int16(scalingFactor*(a*(double(CT(CT>=BP))+...
    abs(valHUair))+b));
```

A.10 imagearrayinsertion.m

```
function el = imagearrayinsertion(image_array,el)
%IMAGEARRAYINSERTION - Insert a 2D image into EL’s PixelData.
%   EL = IMAGEARRAYINSERTION(IMAGE_ARRAY,EL) simply reverses the
%   action of EXTRACTIMAGEARRAY, inserting the 2D image
%   IMAGE_ARRAY into the pixel data of EL.

% Extract the necessary DICOM tags from the header:
for i=1:length(el)
    if strcmpi(strcat(el(i).group,el(i).number),’00080008’)
        imagetype=el(i).data;
    elseif strcmpi(strcat(el(i).group,el(i).number),’00280010’)
        rows=el(i).data;
    elseif strcmpi(strcat(el(i).group,el(i).number),’00280011’)
        columns=el(i).data;
    elseif strcmpi(strcat(el(i).group,el(i).number),’00281052’)
        rescaleintercept=str2double(el(i).data);
    elseif strcmpi(strcat(el(i).group,el(i).number),’00281053’)
        rescaleslope=str2double(el(i).data);
    elseif strcmpi(strcat(el(i).group,el(i).number),’7fe00010’)
        pixeldata_elementposition=i;
    end
end

if strcmpi(imagetype(end-5:end),’AC_MAP’)
```

```
    CTmumap=true;
end

% Reshape 2D image to an image vector:
image_vector=int16(zeros(1,rows*columns));
counter=0;
for i=1:rows
    for j=1:columns
        counter=counter+1;
        image_vector(counter)=image_array(i,j);
    end
end

% Rescale the image vector values:
rsi=0;
rss=1;
if exist('rescaleintercept','var')
    rsi=rescaleintercept;
end
if exist('rescaleslope','var')
    rss=rescaleslope;
end
if exist('CTmumap','var')&&CTmumap
    rss=rss*10000;
end
image_vector=int16((double(image_vector)-rsi)/rss);

% Cast image vector to correct type and replace the PixelData tag
% in EL:
pixeldata=typecast(image_vector,'uint8');
el(pixeldata_elementposition).data=pixeldata;
el(pixeldata_elementposition).length=length(pixeldata);
```

A.11 segmentRibcage.m

Contents

- Parameter initialisation

- Initial segmentation
- Null non-lung volumes
- Dilate lung volumes to include ribcage
- Segment abdomen
- Segment neck and head
- Finallize mask and save it to file

```
function [mask,fileMRmask] = segmentRibcage(fileMR,thresholds,...
    radius,overlap)
```

```
%SEGMENTRIBCAGE segments the ribcage in an MR image.
%   SEGMENTRIBCAGE(FILEMR,THRESHOLDS,RADIUS,OVERLAP) takes the MR
%   NIFTI image FILEMR as input, along with THRESHOLDS: a vector
%   containing segmentation thresholds for the lungs, whole-body,
%   abdomen, neck, and head; RADIUS: the radius of the dilation
%   sphere; and OVERLAP: the number of slices of the top of the
%   lung that will have their mask extended by the neck mask
%   slices. The function makes a mask covering the whole body,
%   excluding the arms, shoulders, and chest muscles. The lungs
%   are identified as the largest volume (possibly the two
%   largest volumes) of air inside the body. The lung volume is
%   dilated to fill the entire inside of the ribcage, and the
%   resulting volume is dilated outwards to include the ribcage.
%   The thickness of the ribcage wall is assumed to be
%   approximately spatially constant. After segmenting the
%   ribcage the rest of the body below the ribs are segmented
%   based on simple thresholding, ensuring that the arms are not
%   in contact with the body below the ribs. The neck and head is
%   segmented based on simple thresholding. The mask is both
%   returned as an image volume array and saved to file.
```

Parameter initialisation

```
temp=[2 15 20 20 20 20];
if ~exist('thresholds','var')
    thresholds=temp;
elseif length(thresholds)<6
    thresholds=[thresholds temp(length(thresholds)+1:end)];
end
if ~exist('overlap','var')
```

```
    overlap=2;
end
[dirRoot,name,ext]=fileparts(fileMR);
dirRoot=[dirRoot filesep];
fileMRmask=[dirRoot 'ribMask' name ext];
fileTemp=[dirRoot 'temp.nii'];
fileDataFrames=[dirRoot 'dataFrames.mat'];
fileDataAbdomen=[dirRoot 'dataAbdomen.mat'];
fileDataLungthresholds=[dirRoot 'dataLungthresholds.mat'];

MR=load_nii(fileMR);
MRimg=MR.img;
MRmask=MR;
mask=MRimg;
lower=thresholds(1);
higher=thresholds(2);
bodyThreshold=thresholds(3);
abdomenThreshold=thresholds(4);
neckThreshold=thresholds(5);
headThreshold=thresholds(6);
```

Initial segmentation

```
mask(:)=0;
mask(MRimg>=lower&MRimg<=higher)=1;
tempmask=createmask(MRimg,bodyThreshold); % Bodymask
mask=mask.*tempmask; % Null voxels outside of body.

mask=fillMask(mask); % Fill holes in all three planes.

% volumeRelativeLungRight=0.55;
% volumeRelativeLungLeft=0.45;
```

Null non-lung volumes

Identify the two largest volumes:

```
CC=bwconncomp(mask,6);
```

```
indexes=CC.PixelIdxList;
lengths=cellfun('length',indexes);
[~,indexLargestVolume]=max(lengths);
lengths(indexLargestVolume)=0;
[~,indexSecondLargestVolume]=max(lengths);

% Detect if the two lung volumes are connected:
ratio=cellfun('length',indexes(indexLargestVolume))/cellfun(...
    'length',indexes(indexSecondLargestVolume));
if ratio>0.7/0.3
    lungsConnected=true;
else
    lungsConnected=false;
end

% Keep lung volume(s) only:
% If lung volumes are not connected, connect their centers of
% mass.
mask(:)=0;
if lungsConnected
    mask(indexes{indexLargestVolume})=1;
else
    mask(indexes{indexLargestVolume})=1;
    mask(indexes{indexSecondLargestVolume})=1;
    CC=bwconncomp(mask,6);
    STATS=regionprops(CC,'Centroid');
    centr1=STATS(1).Centroid;centr2=STATS(2).Centroid;
    x=round(centr1(1)); y=round(centr1(2)); z=round(centr1(3));
    x(2)=round(centr2(1)); y(2)=round(centr2(2)); z(2)=...
        round(centr2(3));
    mask(min(y):max(y),min(x):max(x),min(z):max(z))=1;
end

% Manually detect top and bottom of lungs:
% Also choose a neck slice for neck segmentation.
if exist(fileDataFrames,'file')
    load(fileDataFrames);
else
```

```
    system(sprintf('mricron %s -x',fileMR));
    neckFrame=input('Neck slice, z-value: ');
    topFrame=input('Top of lung, z-value: ');
    bottomFrame=input('Bottom of lung, z-value: ');
    if isempty(neckFrame); neckFrame=size(mask,3); end
    if isempty(topFrame); topFrame=size(mask,3); end
    if isempty(bottomFrame); bottomFrame=1; end
    save(fileDataFrames,'neckFrame','topFrame','bottomFrame');
end
mask(:,:,topFrame+1:end)=0;
mask(:,:,1:bottomFrame-1)=0;

% Ask for confirmation of successful lung segmentation:
if ~exist(fileDataLungthresholds,'file')
    temp=MRmask;
    temp.img=mask;
    save_nii(temp,fileTemp)
    viewWoverlay(fileMR,fileTemp,true)
    temp=load_nii(fileTemp);
    mask=temp.img;
    tuningNeeded=input(...
        'Tune lung segmentation thresholds? (y/n) [y]: ',...
        's');
    if isempty(tuningNeeded); tuningNeeded='y'; end
    if strcmpi(tuningNeeded,'y')
        error('Tuning needed!')
    else
        save(fileDataLungthresholds,'lower','higher',...
            'bodyThreshold');
    end
end
```

Dilate lung volumes to include ribcage

Define structuring elements:

Define structuring elements:

```
lengthSe=9;
middleSe=floor((lengthSe+1)/2);
seLeftward=zeros(lengthSe,1); seLeftward(1:middleSe)=1;
seLeftward=strel('arbitrary',seLeftward);
seRightward=zeros(lengthSe,1); seRightward(middleSe:end)=1;
seRightward=strel('arbitrary',seRightward);
seDownward=zeros(1,1,lengthSe); seDownward(1:middleSe)=1;
seDownward=strel('arbitrary',seDownward);

% Dilate towards the bottom of the lungs:
for i=1:30
    mask(:,:,bottomFrame:topFrame)=imdilate(mask(:,:,...
        bottomFrame:topFrame),seDownward);
end

% Find center of mass:
CC=bwconncomp(mask,6);
STATS=regionprops(CC,'Centroid','BoundingBox');
assert(length(STATS)==1,'Lungs not connected!')
centroid=round(STATS(1).Centroid);

% Dilate from left/right towards center of mass:
middleRow=centroid(2);
for i=1:10
    mask(1:middleRow,:,:)=imdilate(mask(1:middleRow,:,:),...
        seRightward);
    mask(middleRow:end,:,:)=imdilate(mask(middleRow:end,:,:),...
        seLeftward);
end

% Dilate volume outwards to include rib cage:
[x,y,z] = ndgrid(-radius:radius);
se = strel(sqrt(x.^2 + y.^2 + z.^2) <=radius);
mask(:,:,bottomFrame:end)=imdilate(mask(:,:,bottomFrame:end),se);
```

Segment abdomen

```
if exist(fileDataAbdomen,'file')
```

```
    load(fileDataAbdomen);
else
    tempmask=MRimg;
    tempmask(:)=0;
    tempmask(:,:,1:bottomFrame-1)=createmask(MRimg(:,:,...
        1:bottomFrame-1),...
        abdomenThreshold,1);

% Check if the arms are too close to body for automatic
% segmentation:
    armsincluded=true;
    CC=bwconncomp(tempmask);
    indexes=CC.PixelIdxList;
    lengths=cellfun('length',indexes);
    if length(lengths)>=3
        indexLargest=zeros(1,3);
        [~,indexLargest(1)]=max(lengths);
        lengths(indexLargest(1))=0;
        [~,indexLargest(2)]=max(lengths);
        lengths(indexLargest(2))=0;
        [~,indexLargest(3)]=max(lengths);
        lengths=cellfun('length',indexes);
        if lengths(indexLargest)>radius^4
            tempmask(:)=0;
            tempmask(indexes{indexLargest(1)})=1;
            armsincluded=false;
        end
    end

%     se=strel('disk',1);
%     tempmask=imdilate(tempmask,se);

% Manually null arms:
    while armsincluded
        temp=MRmask;
        temp.img=tempmask;
        save_nii(temp,fileTemp)
        viewWoverlay(fileMR,fileTemp,true)
```

```
        temp=load_nii(fileTemp);
        tempmask=temp.img;
        armsincluded=...
            input('Arms included in mask? (true/false): ');
    end

    save(fileDataAbdomen,'tempmask')
end

mask(:,:,1:bottomFrame-1)=tempmask(:,:,1:bottomFrame-1);
```

Segment neck and head

Neck: The neck is segmented by creating a 2D mask based on a selected trans-axial slice in the neck, and then placing this mask around the center of gravity of the slices between the selected slice and the top of the lungs plus the optional overlap.

```
MRneckslice=MRimg(:,:,neckFrame);
MRneckslicemask=createmask(MRneckslice,neckThreshold);
CC=bwconncomp(MRneckslicemask,4);
indexes=CC.PixelIdxList;
lengths=cellfun('length',indexes);
[~,indexLargest]=max(lengths);

STATS=regionprops(CC,'BoundingBox','ConvexImage');
bbox=round(STATS(indexLargest).BoundingBox);
MRneckslicemask(:)=0;
MRneckslicemask(bbox(2):bbox(2)+bbox(4)-1,bbox(1):bbox(1)+...
    bbox(3)-1)=STATS(indexLargest).ConvexImage;
CC=bwconncomp(MRneckslicemask,4);
STATS=regionprops(CC,'Centroid');
centroidneckslice=STATS.Centroid;

tempmask=mask;
tempmask(:)=0;
for i=topFrame+1-overlap:neckFrame-1
    tempmask(:,:,i)=createmask(MRimg(:,:,i),neckThreshold);
```

```
    CC=bwconncomp(tempmask(:,:,i),4);
    indexes=CC.PixelIdxList;
    lengths=cellfun('length',indexes);
    [~,indexLargest]=max(lengths);
    STATS=regionprops(CC,'Centroid');
    centroidtemp=STATS(indexLargest).Centroid;

    translationvector=round(centroidtemp-centroidneckslice);
    tempmask(:,:,i)=imtranslate(MRneckslicemask,...
        translationvector);
end

% Head:
tempmask(:,:,neckFrame:end)=createmask(MRimg(:,:,...
    neckFrame:end),headThreshold,1);
CC=bwconncomp(tempmask(:,:,neckFrame:end),6);
indexes=CC.PixelIdxList;
lengths=cellfun('length',indexes);
[~,indexLargest]=max(lengths);

tempmask(:,:,neckFrame:end)=0;
tempmask(indexes{indexLargest}+numel(tempmask(:,:,...
    1:neckFrame-1)))=1;
```

Finallize mask and save it to file

```
mask=mask+tempmask;
mask(mask>1)=1;

MRmask.img=mask;

save_nii(MRmask,fileMRmask)
```

B Registration parameter files

A couple of examples of the `elastix` registration parameter files used in the registrations performed during the work with this master, are shown below. The parameter files were developed based on existing parameter files from similar studies, information found in the `elastix` manual, and on trial-and-error when tweaking the parameters to improve the registrations.

B.1 BsplineWBCT.txt

This file set the parameters for the B-spline registration in the creation of whole-body CTMR mumaps.

```
(FixedInternalImagePixelType "float")
(MovingInternalImagePixelType "float")
(FixedImageDimension 3)
(MovingImageDimension 3)
(UseDirectionCosines "true")

// Main components
(Registration "MultiResolutionRegistration")
(Interpolator "LinearInterpolator")
(ResampleInterpolator "FinalBSplineInterpolator")
(Resampler "DefaultResampler")
(FixedImagePyramid "FixedSmoothingImagePyramid")
(MovingImagePyramid "MovingSmoothingImagePyramid")
(Optimizer "AdaptiveStochasticGradientDescent")
(Transform "BSplineTransform")
(Metric "NormalizedMutualInformation")

// Transformation
(FinalGridSpacingInPhysicalUnits 16)
(GridSpacingSchedule 4.0 4.0 4.0 2.0 2.0 2.0 1.0 1.0 1.0)
(HowToCombineTransforms "Compose")

// Similarity measure
(NumberOfHistogramBins 32)
(ErodeMask "true")
```

```
// Multiresolution
(NumberOfResolutions 3)
(ImagePyramidSchedule 1 1 1  1 1 1  1 1 1)

// Optimiser
(MaximumNumberOfIterations 4096 2048 1024)
(AutomaticParameterEstimation "false")
(SP_a 2000 7000 20000 20000 30000)
(SP_A 50)
(SP_alpha 0.602)
//(MaximumStepLength 1.0)
//(CheckNumberOfValidSamples "true")
//(RequiredRatioOfValidSamples 0.05)

// Image sampling
(NumberOfSpatialSamples 4096)
(NewSamplesEveryIteration "true")
(ImageSampler "RandomCoordinate")
//(UseRandomSampleRegion "true")
//(SampleRegionSize 50.0 50.0 50.0)
//(MaximumNumberOfSamplingAttempts 10)

// Interpolation and resampling
//(BSplineInterpolationOrder 1)
(FinalBSplineInterpolationOrder 3)
(DefaultPixelValue -1024)
(WriteResultImage "true")
(ResultImagePixelType "short")
(ResultImageFormat "nii")
```

B.2 rigidHeadCT.txt

This file set the parameters for the first rigid registration in the creation of head CT and CTMR μ-maps.

```
(FixedInternalImagePixelType "float")
(MovingInternalImagePixelType "float")
(FixedImageDimension 3)
(MovingImageDimension 3)
```

```
(UseDirectionCosines "true")

// Main components
(Registration "MultiResolutionRegistration")
(Interpolator "LinearInterpolator")
(ResampleInterpolator "FinalBSplineInterpolator")
(Resampler "DefaultResampler")

(FixedImagePyramid "FixedSmoothingImagePyramid")
(MovingImagePyramid "MovingSmoothingImagePyramid")

(Optimizer "AdaptiveStochasticGradientDescent")
(Transform "EulerTransform")
(Metric "NormalizedMutualInformation")

// Transformation
(AutomaticScalesEstimation "true")
(AutomaticTransformInitialization "true")
(HowToCombineTransforms "Compose")

// Similarity measure
(NumberOfHistogramBins 32)
//(ErodeMask "true")

// Multiresolution
(NumberOfResolutions 5)
(ImagePyramidSchedule 16 16 16  8 8 8  4 4 4  2 2 2  1 1 1)

// Optimiser
(MaximumNumberOfIterations 4096 2048 1024 512 256)
(AutomaticParameterEstimation "false")
(SP_a 4000)
(SP_A 50)
(SP_alpha 0.6)
//(MaximumStepLength 1.0)
//(CheckNumberOfValidSamples "true")
//(RequiredRatioOfValidSamples 0.05)
```

```
// Image sampling
(NumberOfSpatialSamples 5000)
(NewSamplesEveryIteration "true")
(ImageSampler "RandomCoordinate")
//(UseRandomSampleRegion "true")
//(SampleRegionSize 50.0 50.0 50.0)

// Interpolation and resampling
//(BSplineInterpolationOrder 1)
(FinalBSplineInterpolationOrder 3)
(DefaultPixelValue -1024)
(WriteResultImage "true")
(ResultImagePixelType "short")
(ResultImageFormat "nii")
```

B.3 rigidHeadCT2.txt

This file set the parameters for the second rigid registration in the creation of head CT and CTMR μ-maps.

```
(FixedInternalImagePixelType "float")
(MovingInternalImagePixelType "float")
(FixedImageDimension 3)
(MovingImageDimension 3)
(UseDirectionCosines "true")

// Main components
(Registration "MultiResolutionRegistration")
(Interpolator "LinearInterpolator")
(ResampleInterpolator "FinalBSplineInterpolator")
(Resampler "DefaultResampler")

(FixedImagePyramid "FixedSmoothingImagePyramid")
(MovingImagePyramid "MovingSmoothingImagePyramid")

(Optimizer "AdaptiveStochasticGradientDescent")
(Transform "EulerTransform")
(Metric "NormalizedMutualInformation")
```

```
// Transformation
(AutomaticScalesEstimation "true")
(AutomaticTransformInitialization "true")
(HowToCombineTransforms "Compose")

// Similarity measure
(NumberOfHistogramBins 32)
//(ErodeMask "true")

// Multiresolution
(NumberOfResolutions 3)
(ImagePyramidSchedule 1 1 1  1 1 1  1 1 1)

// Optimiser
(MaximumNumberOfIterations 4096 4096 4096)
(AutomaticParameterEstimation "false")
(SP_a 4000)
(SP_A 50)
(SP_alpha 0.6)
//(MaximumStepLength 1.0)
//(CheckNumberOfValidSamples "true")
//(RequiredRatioOfValidSamples 0.05)

// Image sampling
(NumberOfSpatialSamples 5000)
(NewSamplesEveryIteration "true")
(ImageSampler "RandomCoordinate")
//(UseRandomSampleRegion "true")
//(SampleRegionSize 50.0 50.0 50.0)

// Interpolation and resampling
//(BSplineInterpolationOrder 1)
(FinalBSplineInterpolationOrder 3)
(DefaultPixelValue -1024)
(WriteResultImage "true")
(ResultImagePixelType "short")
(ResultImageFormat "nii")
```

B.4 rigidWBCT.txt

This file set the parameters for the rigid registration in the creation of whole-body CTMR μ-maps.

```
(FixedInternalImagePixelType "float")
(MovingInternalImagePixelType "float")
(FixedImageDimension 3)
(MovingImageDimension 3)
(UseDirectionCosines "true")

// Main components
(Registration "MultiResolutionRegistration")
(Interpolator "LinearInterpolator")
(ResampleInterpolator "FinalBSplineInterpolator")
(Resampler "DefaultResampler")
(FixedImagePyramid "FixedSmoothingImagePyramid")
(MovingImagePyramid "MovingSmoothingImagePyramid")
(Optimizer "AdaptiveStochasticGradientDescent")
(Transform "EulerTransform")
(Metric "AdvancedMattesMutualInformation")

// Transformation
(AutomaticScalesEstimation "true")
(AutomaticTransformInitialization "true")
(HowToCombineTransforms "Compose")
(AutomaticTransformInitializationMethod "CenterOfGravity")

// Similarity measure
(NumberOfHistogramBins 32)
//(ErodeMask "true")

// Multiresolution
(NumberOfResolutions 5)
(ImagePyramidSchedule 16 16 16  8 8 8  4 4 4  2 2 2  1 1 1)

// Optimiser
(MaximumNumberOfIterations 4096 2048 1024 512 256)
(AutomaticParameterEstimation "false")
```

```
(SP_a 4000)
(SP_A 50)
(SP_alpha 0.6)
//(MaximumStepLength 1.0)
//(CheckNumberOfValidSamples "true")
//(RequiredRatioOfValidSamples 0.05)

// Image sampling
(NumberOfSpatialSamples 5000)
(NewSamplesEveryIteration "true")
(ImageSampler "RandomCoordinate")
//(UseRandomSampleRegion "true")
//(SampleRegionSize 50.0 50.0 50.0)

// Interpolation and resampling
//(BSplineInterpolationOrder 1)
(FinalBSplineInterpolationOrder 3)
(DefaultPixelValue -1024)
(WriteResultImage "true")
(ResultImagePixelType "short")
(ResultImageFormat "nii")
```

VI

C Additional results

This section presents additional results not presented in Part III.

C.1 PET - head

C.1.1 MR - CTMR

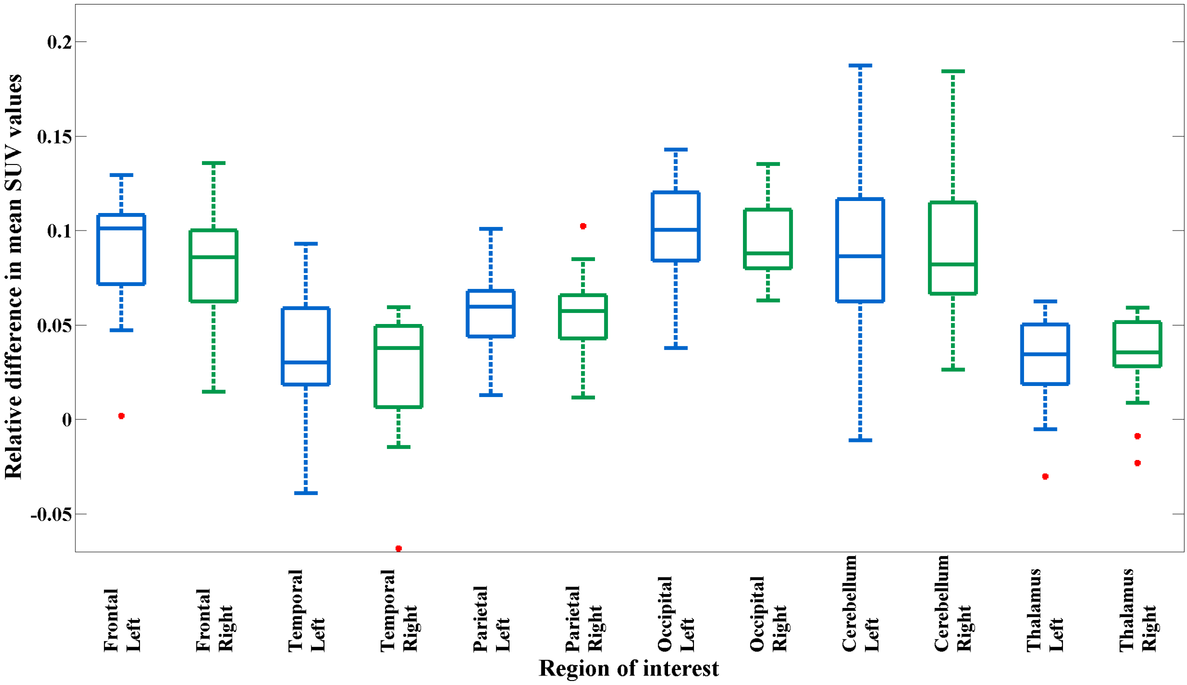


Figure 41: Box plot of the relative difference in mean SUV values between PET head images reconstructed using the original MR UTE μ*-map, and using the CTMR* μ*-map. The reconstruction method was 3D-iterative, and the original MR UTE* μ*-map reconstruction results were used as the reference point.* $N = 19$.

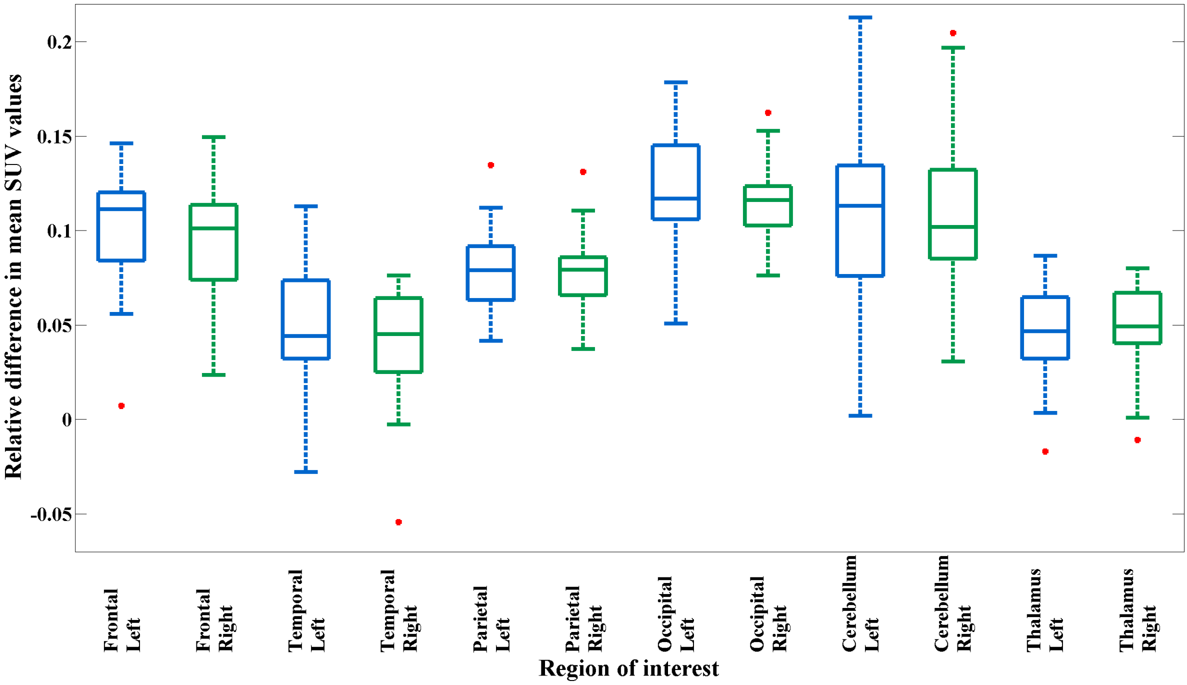


Figure 42: Box plot of the relative difference in mean SUV values between PET head images reconstructed using the original MR UTE μ*-map, and using the CTMR* μ*-map. The reconstruction method was PSF, and the original MR UTE* μ*-map reconstruction results were used as the reference point.* $N = 19$.

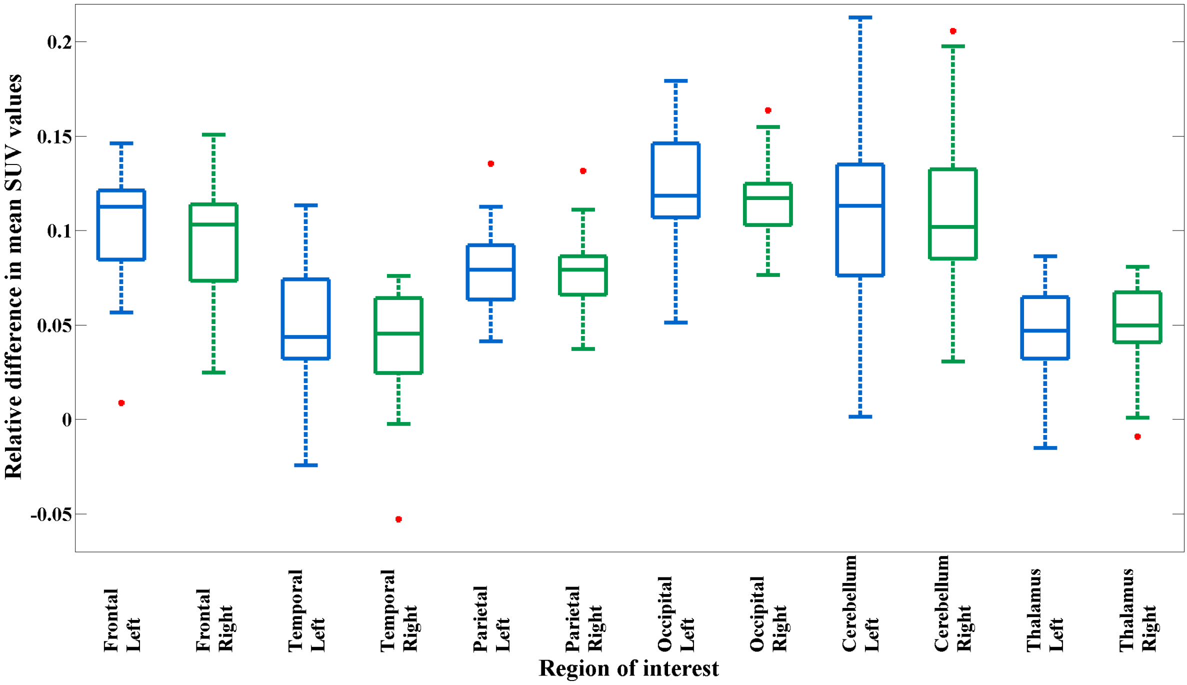


Figure 43: Box plot of the relative difference in mean SUV values between PET head images reconstructed using the original MR UTE μ-map, and using the CTMR μ-map. The reconstruction method was PSF, and the original MR UTE μ-map reconstruction results were used as the reference point. $N = 19$.

C.1.2 CT - CTMR

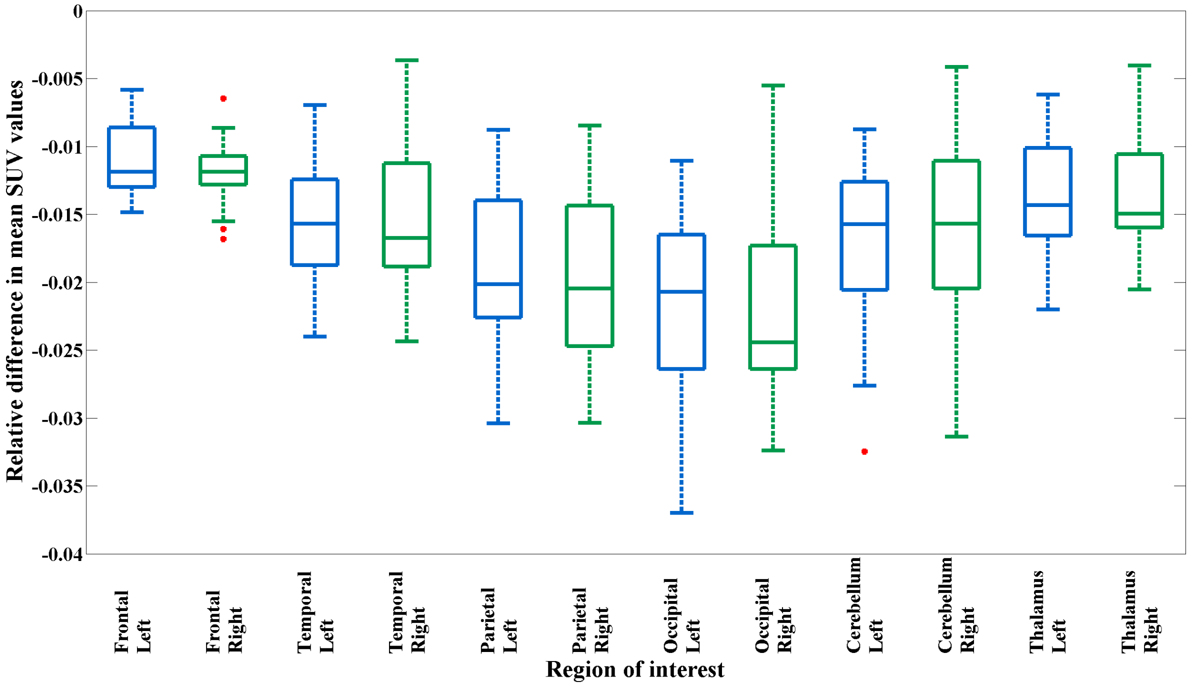


Figure 44: Box plot of the relative difference in mean SUV values between PET head images reconstructed using the CT μ*-map, and using the CTMR* μ*-map. The reconstrucion method was PSF, and the CT reconstruction results were used as the reference point.* $N = 19$*.*

C.1.3 3D - PSF

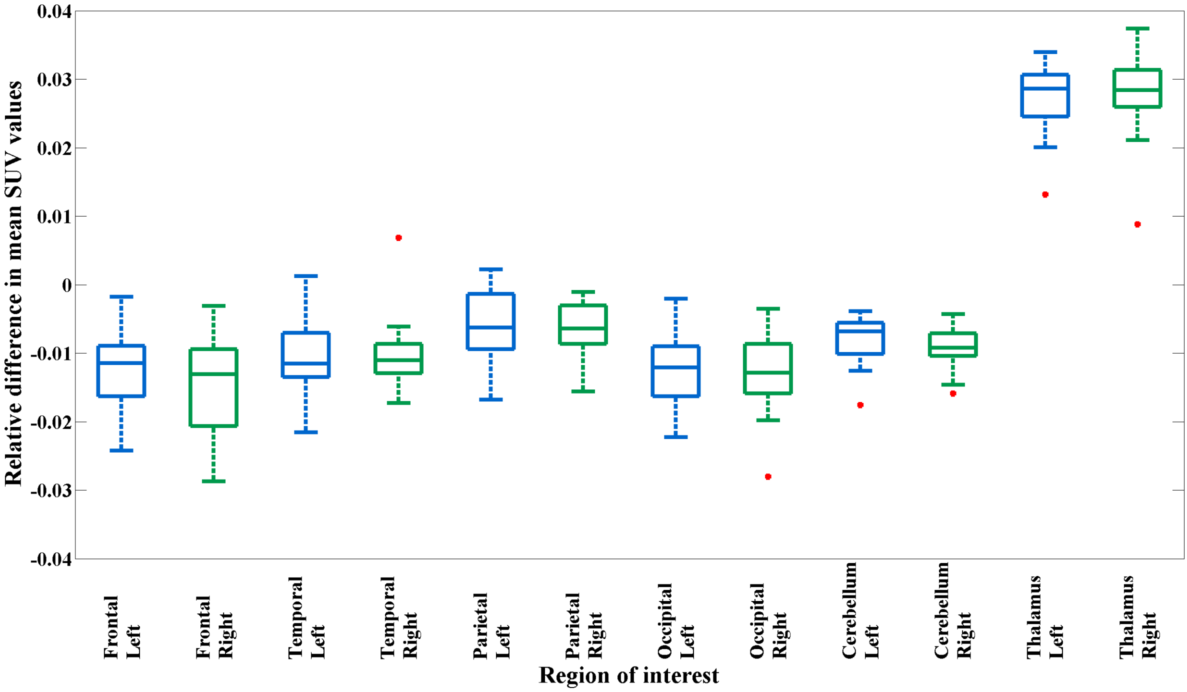


Figure 45: Box plot of the relative difference in mean SUV values between PET head images reconstructed using regular 3D-iterative reconstruction, and using the PSF reconstruction method. The MR μ-map was used for reconstruction, and the 3D-iterative reconstruction results have been used as the reference point. $N = 19$.

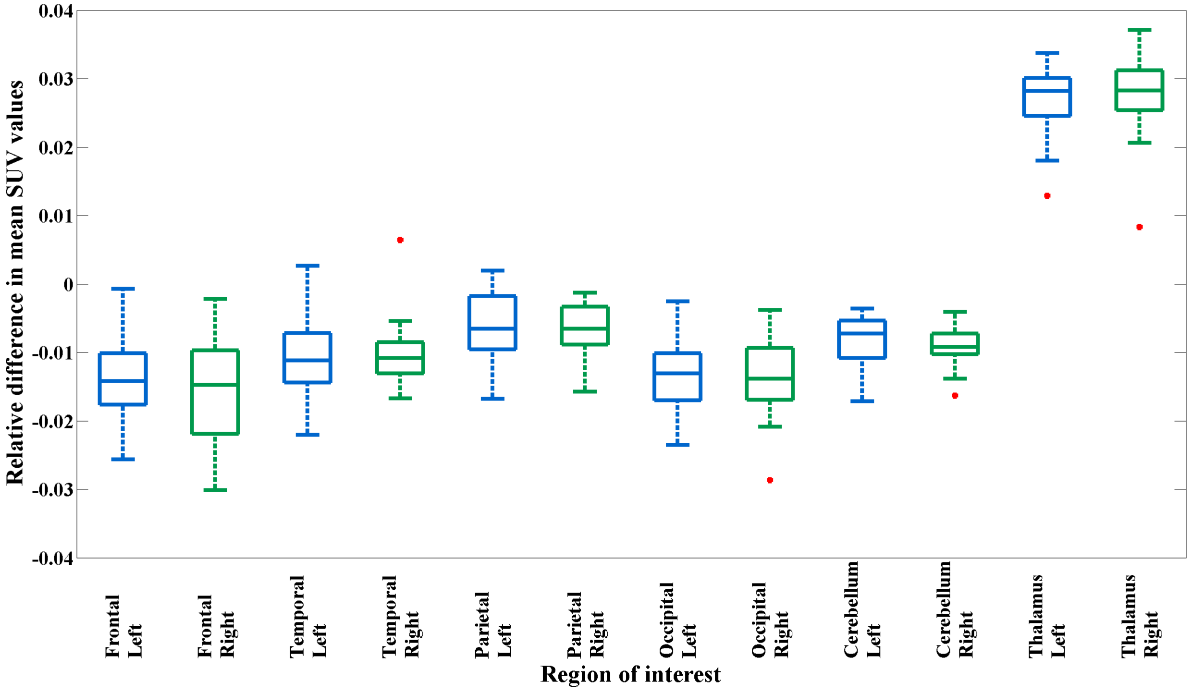


Figure 46: Box plot of the relative difference in mean SUV values between PET head images reconstructed using regular 3D-iterative reconstruction, and using the PSF reconstruction method. The CT μ*-map was used for reconstruction, and the 3D-iterative reconstruction results have been used as the reference point.* $N = 19$.

References

[1] A. P. Aitken et al. “Improved UTE-based attenuation correction for cranial PET-MR using dynamic magnetic field monitoring”. In: *Medical Physics* 41.1, 012302 (2014), pp. 1–13.

[2] A.M. Alessio et al. “Application and Evaluation of a Measured Spatially Variant System Model for PET Image Reconstruction”. In: *Medical Imaging, IEEE Transactions on* 29.3 (Mar. 2010), pp. 938–949.

[3] Bing Bai and P.D. Esser. “The effect of edge artifacts on quantification of Positron Emission Tomography”. In: *Nuclear Science Symposium Conference Record (NSS/MIC), 2010 IEEE.* Oct. 2010, pp. 2263–2266.

[4] Dale L. Bailey. “Data Acquisition and Performance Characterization in PET”. In: *Positron Emission Tomography.* Ed. by Dale L. Bailey et al. Springer London, 2005, pp. 41–62.

[5] Dale L. Bailey, Joel S. Karp, and Suleman Surti. “Physics and Instrumentation in PET”. In: *Positron Emission Tomography.* Ed. by Dale L. Bailey et al. Springer London, 2005, pp. 13–39.

[6] Ilja Bezrukov et al. “MR-Based PET Attenuation Correction for PET/MR Imaging”. In: *Seminars in Nuclear Medicine* 43.1 (2013), pp. 45–59.

[7] K. T. Block and M. Uecker. “Simple method for adaptive gradient-delay compensation in radial MRI”. In: *Proceedings of the International Society for Magnetic Resonance in Medicine.* 2011, p. 2816.

[8] Jonathan P. J. Carney et al. “Method for transforming CT images for attenuation correction in PET/CT imaging”. In: *Medical Physics* 33.4 (2006), pp. 976–983.

[9] Bernard D. Coombs, Jerzy Szumowski, and William Coshow. “Two-point Dixon technique for water-fat signal decomposition with B0 inhomogeneity correction”. In: *Magnetic Resonance in Medicine* 38.6 (1997), pp. 884–889.

[10] Michel Defrise, Paul E. Kinahan, and Christian J. Michel. "Image Reconstruction Algorithms in PET". In: *Positron Emission Tomography*. Ed. by Dale L. Bailey et al. Springer London, 2005, pp. 63–91.

[11] Gaspar Delso et al. "Anatomic Evaluation of 3-Dimensional Ultrashort-Echo-Time Bone Maps for PET/MR Attenuation Correction". In: *Journal of Nuclear Medicine* 55.5 (2014), pp. 780–785.

[12] Gaspar Delso et al. "Performance Measurements of the Siemens mMR Integrated Whole-Body PET/MR Scanner". In: *Journal of Nuclear Medicine* 52.12 (2011), pp. 1914–1922.

[13] John C. Dickson, Celia O'Meara, and Anna Barnes. "A comparison of CT- and MR-based attenuation correction in neurological PET". English. In: *European Journal of Nuclear Medicine and Molecular Imaging* 51 (2014), pp. 1–14.

[14] Olaf Dietrich. "MRI from k-Space to Parallel Imaging". In: *Parallel Imaging in Clinical MR Applications*. Ed. by Stefan O. Schoenberg, Olaf Dietrich, and Maximilian F. Reiser. Medical Radiology. Springer Berlin Heidelberg, 2007, pp. 3–17.

[15] Jürg Fröhlich. "Spin, or actually: Spin and Quantum Statistics". English. In: *The Spin*. Ed. by Bertrand Duplantier, Jean-Michel Raimond, and Vincent Rivasseau. Vol. 55. Progress in Mathematical Physics. Birkhäuser Basel, 2009, pp. 1–60.

[16] Lee W. Goldman. "Principles of CT and CT Technology". In: *Journal of Nuclear Medicine Technology* 35.3 (2007), pp. 115–128.

[17] Matthias Hofmann et al. "Towards quantitative PET/MRI: a review of MR-based attenuation correction techniques". English. In: *European Journal of Nuclear Medicine and Molecular Imaging* 36.1 (2009), pp. 93–104.

[18] John L. Humm, Anatoly Rosenfeld, and Alberto Guerra. "From PET detectors to PET scanners". English. In: *European Journal of Nuclear Medicine and Molecular Imaging* 30.11 (2003), pp. 1574–1597.

[19] E. Mark Haacke et al. *Magnetic resonance imaging*. Wiley-Liss New York: 1999.

[20] David Izquierdo-Garcia et al. “Comparison of MR-based attenuation correction and CT-based attenuation correction of whole-body PET/MR imaging”. English. In: *European Journal of Nuclear Medicine and Molecular Imaging* (2014), pp. 1–11.

[21] Vincent Keereman et al. “Challenges and current methods for attenuation correction in PET/MR”. English. In: *Magnetic Resonance Materials in Physics, Biology and Medicine* 26.1 (2013), pp. 81–98.

[22] Gary J. Kelloff et al. “Progress and Promise of FDG-PET Imaging for Cancer Patient Management and Oncologic Drug Development”. In: *Clinical Cancer Research* 11.8 (2005), pp. 2785–2808.

[23] Stefan Klein and Marius Staring. *elastix - The Manual.* Feb. 2014.

[24] Albert Lehninger, David L. Nelson, and Michael M. Cox. *Lehninger Principles of Biochemistry.* Fifth Edition. W. H. Freeman, June 2008.

[25] Jingfei Ma. “Dixon techniques for water and fat imaging”. In: *Journal of Magnetic Resonance Imaging* 28.3 (2008), pp. 543–558.

[26] M. Marshall et al. “A comparison of X-ray spectra and outputs from molybdenum and tungsten targets”. In: *The British journal of radiology* 48.565 (1975), pp. 31–39.

[27] Steven R. Meikle and Ramsey D. Badawi. “Quantitative Techniques in PET”. In: *Positron Emission Tomography.* Ed. by Dale L. Bailey et al. Springer London, 2005, pp. 93–126.

[28] P. Mollet et al. “Simultaneous MR-Compatible Emission and Transmission Imaging for PET Using Time-of-Flight Information”. In: *Medical Imaging, IEEE Transactions on* 31.9 (Sept. 2012), pp. 1734–1742.

[29] K. Murphy et al. “Semi-automatic construction of reference standards for evaluation of image registration”. In: *Medical Image Analysis* 15.1 (2011), pp. 71–84.

[30] Pierre-Jean Nacher. “Magnetic Resonance Imaging: From Spin Physics to Medical Diagnosis”. English. In: *The Spin.* Ed. by Bertrand Duplantier, Jean-Michel Raimond, and Vincent Rivasseau. Vol. 55. Progress in Mathematical Physics. Birkhäuser Basel, 2009, pp. 159–193.

[31] F. Natterer and H. Herzog. “Attenuation correction in positron emission tomography”. In: *Mathematical Methods in the Applied Sciences* 15.5 (1992), pp. 321–330.

[32] Rubelisa Cândido Gomes de Oliveira et al. “Assessments of trabecular bone density at implant sites on {CT} images”. In: *Oral Surgery, Oral Medicine, Oral Pathology, Oral Radiology, and Endodontology* 105.2 (2008), pp. 231–238.

[33] J. M. Ollinger and J. A. Fessler. “Positron-emission tomography”. In: *Signal Processing Magazine, IEEE* 14.1 (Jan. 1997), pp. 43–55.

[34] V. Y. Panin et al. “PET reconstruction with system matrix derived from point source measurements”. In: *Nuclear Science, IEEE Transactions on* 53.1 (Feb. 2006), pp. 152–159.

[35] Bernd J. Pichler et al. “PET/MRI: Paving the Way for the Next Generation of Clinical Multimodality Imaging Applications”. In: *Journal of Nuclear Medicine* 51.3 (2010), pp. 333–336.

[36] Harald H. Quick. “Integrated PET/MR”. In: *Journal of Magnetic Resonance Imaging* 39.2 (2014), pp. 243–258.

[37] Matthew D. Robson et al. “Magnetic resonance: an introduction to ultrashort TE (UTE) imaging”. In: *Journal of computer assisted tomography* 27.6 (2003), pp. 825–846.

[38] T. Rohlfing. “Image Similarity and Tissue Overlaps as Surrogates for Image Registration Accuracy: Widely Used but Unreliable”. In: *Medical Imaging, IEEE Transactions on* 31.2 (Feb. 2012), pp. 153–163.

[39] J. Anthony Seibert and John M. Boone. “X-Ray Imaging Physics for Nuclear Medicine Technologists. Part 2: X-Ray Interactions and Image Formation”. In: *Journal of Nuclear Medicine Technology* 33.1 (2005), pp. 3–18.

[40] S. Tong, A. M. Alessio, and P. E. Kinahan. “Noise and signal properties in PSF-based fully 3D PET image reconstruction: an experimental evaluation”. In: *Physics in Medicine and Biology* 55.5 (2010), p. 1453.

[41] Shan Tong et al. “Properties and Mitigation of Edge Artifacts in PSF-Based PET Reconstruction”. In: *Nuclear Science, IEEE Transactions on* 58.5 (Oct. 2011), pp. 2264–2275.

[42] David W. Townsend. “Basic Science of PET and PET/CT”. In: *Positron Emission Tomography*. Ed. by Peter E. Valk et al. Springer London, 2006, pp. 1–16.

[43] David W. Townsend. “Positron Emission Tomography/Computed Tomography”. In: *Seminars in Nuclear Medicine* 38.3 (2008). Developments in Instrumentation, pp. 152–166.