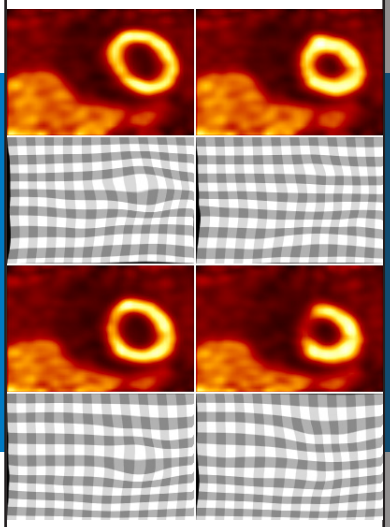
Computer Aided Medical Procedures
(CAMP)
Prof. Dr. Nassir Navab



Dissertation

Joint Image and Motion Reconstruction for Positron Emission Tomography

Moritz Blume

Fakultät für Informatik
Technische Universität München

TECHNISCHE UNIVERSITÄT MÜNCHEN
Computer-Aided Medical Procedures & Augmented Reality

Joint Image and Motion Reconstruction for Positron Emission Tomography

Moritz Blume

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr. Klaus A. Kuhn
Prüfer der Dissertation:	1. Univ.-Prof. Dr. Nassir Navab
	2. Prof. Dr. Magdalena Rafecas, Universitat de València, Spanien

Die Dissertation wurde am 6. April 2011 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 23. August 2011 angenommen.

This work has been partly supported by the CAMP TUM Scholarship, the Deutscher Akademischer Austausch Dienst (DAAD), the Spanish Ministry of Education and Science (TEC2007-61047) and the Spanish Ministry of Science and Innovation (FPA2008-02419-E/FPA).

Abstract

Patient motion is one of the main contributing factors to image degradation in PET imaging. If ignored in the reconstruction algorithm, motion can lead to blurring in the reconstructed image and thus be a severe limiting factor for correct diagnosis or quantification.

Motion compensation methods currently used for cardiac PET imaging involve gating, followed by static image reconstruction of individual gated data. The main disadvantage of such techniques is that they trade motion blurring for image noise: the less motion blurring present within a gate, the more gates are needed, reducing the data in each gate. As the data present in each gate is reduced, a corresponding increase in noise in the reconstructed image is observed.

As the major contribution of this thesis, a novel motion compensation method which jointly reconstructs both image and motion based on gated PET data is proposed. The reconstructed image can be transformed to each gate by a simultaneously reconstructed motion field. Since the event counts of all gates are used, in the reconstructed image there is no noise increase for an increased number of gates. The proposed method makes use of non-rigid motion models in order to address complex deformations which can be present in PET (e. g. cardiac imaging).

The method is compared to several other motion compensation techniques in an analytical simulation study in which additive noise is also considered. Cardiac PET simulations are used since heart motion is especially challenging: the combination of respiratory and cardiac motion leads to particularly complicated deformations. Both quantitative and visual inspection provide favorable results for the method developed in this thesis, especially in a low count rate scenario. In addition to simulated data, true clinical data was used in reconstruction. While initial results of clinical data are promising, a quantitative analysis is subject to future research.

The thesis is concluded with a discussion on future fields of research for the joint motion compensation method, namely attenuation correction, quantitative extensions and the role of motion compensation in PET-MRI.

Keywords
PET, Joint Reconstruction, Motion Compensation, Cardiac Image Reconstruction, Gating

Zusammenfassung

Die Patientenbewegung stellt einen wesentlichen Degradierungsfaktor für die Bildqualität in der klinischen PET Bildgebung dar. Wird die Bewegung ignoriert, kann sie im rekonstruierten Bild zu Unschärfe führen und erschwert somit eine korrekte Diagnose oder Quantifizierung.

Bewegungskompensationsalgorithmen, die derzeit für die Herz-PET Bildgebung verwendet werden, benutzen Gating. Die einzelnen Gates werden dann von gewöhnlichen statischen Bildrekonstruktionsalgorithmen rekonstruiert. Der größte Nachteil dieser Methoden ist, dass Bewegungsunschärfe auf Kosten von Bildrauschen gesenkt wird: je weniger Bewegungsunschärfe in einem rekonstruierten Gate vorhanden sein soll, desto mehr Gates werden benötigt. Je mehr Gates definiert sind, desto weniger Counts entfallen auf ein Gate, und desto größer ist das Rauschen im rekonstruierten Bild.

Der wesentliche wissenschaftliche Beitrag dieser Arbeit ist ein neuer Bewegungskompensationsalgorithmus, der gemeinsam sowohl Bild als auch Bewegung auf Basis der gegateten Daten rekonstruiert. Das rekonstruierte Bild kann zu jedem Gate durch das gleichzeitig rekonstruierte Bewegungsfeld transformiert werden. Die Anzahl der Gates wirkt sich bei dem hier vorgestellten Algorithmus nicht auf das Bildrauschen aus, da die Events der gesamten Aufnahme verwendet werden. Der Algorithmus benutzt deformierbare Bewegungsmodelle, um auch mit komplexen Deformationen (wie sie z. B. in der Herz-PET vorkommen) umgehen zu können.

Die vorgestellte Methode wird zu anderen Bewegungskompensationstechniken in einer analytischen Simulationsumgebung, die additives Rauschen berücksichtigt, verglichen. Es werden Simulationen von Herz-PET verwendet, da die Kombination von Atem- und Herzbewegung zu komplizierten Verformungen führt und somit eine besondere Herausforderung darstellt. Sowohl die quantitative als auch die visuelle Analyse ergibt positive Ergebnisse für das in dieser Dissertation vorgestellte Verfahren. Das gilt vor allem dann, wenn nur eine geringe Anzahl von Events vorhanden ist. Neben simulierten Daten wurden klinische Patientendaten rekonstruiert. Auch wenn erste Ergebnisse der klinischen Daten vielversprechend sind, ist eine umfassende quantitative Analyse notwendig und Gegenstand zukünftiger Forschung.

Die Arbeit endet mit einer Diskussion über mögliche künftige Forschungsschwerpunkte für die gemeinsame Rekonstruktion von Bild und Bewegung. Konkret werden Themen wie Schwächungskorrektur, quantitative Erweiterungen und die Rolle der Bewegungskompensation in PET-MRT diskutiert.

Stichwörter
PET, Bewegungskompensation, Bildrekonstruktion, Gating

Acknowledgements

No thesis is ever the product of one person's efforts, and certainly this one was no different. It would never have become reality without the help and suggestions of many supportive friends and colleagues.

First of all I would like to thank my supervisor Nassir Navab for his confidence in me and my work. When I wanted to go abroad to Spain, he arranged the contact to Magdalena Rafecas and even organized a two year scholarship paid from his own research funds - I am very grateful for that. Thanks to Magdalena Rafecas for being my advisor in Spain and also for her unlimited confidence in what I wanted to do. I had her full support over the whole time and was able to visit lots of scientific conferences to present my work.

Another big thank you goes to John Gillam, who always chose the right prescription from his PhD first aid kit whenever I felt I was in trouble. It included encouraging words which most often helped to view my work from a more favorable perspective, unforgettable squash matches, as well as indispensable reading or video recommendations such as geekologie, failblog, star wars kid, keyboard cat, and much more. Thanks to Josep Oliver for always finding some mistakes in formulas on my slides. I also owe a great deal to the rest of the GFIM/IRIS group, namely Paola Solevi, Irene Torres-Espallardo, Karol Brzezinski, John Barrio, Gabriela Llosá and Jorge Cabello, for providing a very enjoyable work environment.

Many thanks to Sibylle Ziegler and Axel Martinez-Möller from the nuclear medicine department of the *Klinikum Rechts der Isar* in Munich for providing clinical PET data. Axel also was an experienced co-author of many publications and always made encouraging and helpful comments.

At the CAMP group in Munich, many thanks to Andreas Keil for extensive discussions on variational calculus and fast B-spline implementations, Darko Zikic for raising my interest in variational methods as my advisor of the *Diplomarbeit*, which established the mathematical basis for this thesis, Ben Glocker for providing his great image registration toolkit *drop* for public and helping me out finding the right parameters, Martin Groher for being the *funniest* student project advisor I ever had and Wolfgang Wein for being my *first* student project advisor at CAMP and co-advisor of my Diplomarbeit. Many thanks to all other people from CAMP, namely Stefan Hinterstoißer, Christian Wachinger, Marco Feuerstein, Hauke Heibel, Tobias Blum, Nicolas Padoy, Ahmad Ahmadi, Maximilian Baust, Selen Atasoy, Stefanie Demirci, Jose Gardiazabal, Tobias Lasser, Anabel Martin-Gonzalez, Olivier Pauly, Oliver Kutter, Jörg Traub and Jakob Vogel, who always provided a very enjoyable work environment when I was in Munich!

Thanks to Roman Worg, my high school physics teacher, who made physics the most enjoyable school subject I had. Thanks to my parents Brigitte and Nikolaus, who never questioned my professional choices and always supported me both morally and financially. I am particularly indebted to my wife Nuria, who accompanied me during all this time and always stood by my side to share ups and downs.

Contents

Thesis Outline

The thesis is subdivided in two parts: part one gives a general introduction which gently guides the reader to the nucleus of the thesis, explains physical basics of PET and basic image reconstruction techniques and finally reviews the state of art in motion compensation. Part two describes and evaluates the proposed joint reconstruction algorithm which is the major contribution of the thesis.

Part 1: Fundamentals of PET and Motion Compensation

Chapter 1: Introduction The need of motion compensation for PET is motivated. A rudimentary first overview on how the motion problem is tackled in PET is given followed by a brief introduction to our joint reconstruction approach. Finally, the contributions of this thesis to the scientific community are highlighted.

Chapter 2: Basics of Positron-Emission-Tomography Physical basics of PET are explained. Then, the linear imaging model including degradation effects is introduced. The need of data compression of the system matrix is motivated and discussed. Finally, an introduction to image reconstruction is given.

Chapter 3: Overview of Motion Compensation The state of the art of current motion compensation algorithms is discussed.

Part 2: A Novel Method for Joint Reconstruction

Chapter 4: Joint Reconstruction Method The joint reconstruction method, which is the main contribution of this thesis, is explained in detail. A discretization is given such that it can be implemented on a computer.

Chapter 5: Evaluation The evaluation methodology together with comparison methods is explained in this chapter.

Chapter 6: Results and Discussion Comparison results of the evaluation are presented and discussed.

Chapter 7: Future Work and Conclusion Initial ideas for future work are discussed, followed by a conclusion.

Appendix

Mathematical Notations An overview of the most important mathematical notations for this thesis.

Symbols and Abbreviations Important symbols and abbreviations used in this thesis.

List of Publications A list of all publications which were published within the scope of this thesis.

Part I.

Fundamentals of PET and Motion Compensation

1. Introduction

1.1. Motivation and Problem Statement

Research in image reconstruction for positron emission tomography (PET) has been dedicated to improve image quality since its beginnings. Maybe the most worth mentioning work in this regard is the seminal paper of Shepp and Vardi [187], who presented the maximum likelihood expectation maximization (ML-EM) algorithm for PET, and thereby paved the way towards statistical reconstruction algorithms which are much superior to analytical reconstruction algorithms with respect to image quality. It took some time until the ML-EM algorithm found its way to clinical routine usage, mainly due to its computational complexity compared to analytical methods. Both faster computers and the ordered subset expectation maximization (OS-EM) [91] alleviated this disadvantage and finally yielded the breakthrough.

With statistical reconstruction algorithms in their hands, researchers have included many physical aspects of PET into their models and thus were able to improve resolution, signal-to-noise ratio (SNR), lesion detectability, etc. Nowadays models include attenuation, scatter, randoms, detector efficiency, detector blur, depth of interaction, photon non-collinearity, positron range and motion.

Each of these degradation phenomena has been and still is subject of research in the PET community. This thesis is dedicated to the study of motion artifacts. Motion artifacts and any method to eliminate them depend much on the targeted scenario. E. g. it makes a difference whether one deals with animal or human motion, whether or not the subject is anesthetized, whether or not it is cooperative or which part of the subject (brain, heart, lung, whole body, etc.) is to be imaged.

Depending on the scenario, motion degradation can be more or less severe. In a recent simulation study [168], heart motion turned out to be the most severe type of degradation in cardiac PET. The study compared the degradation of the resolution for different types of degradation phenomena (motion, positron range, photon non-collinearity, intercrystal scattering and crystal penetration). When all degradation types were active, the resolution was degraded by 48.3%. When all degradation types *except for motion* were active, the resolution was degraded by only 31.6%. When *motion* was *the only* degradation type, the resolution was degraded by 41.4%. In this case, if one had the choice between (a) correcting only for motion and (b) all other effects but motion, he could potentially improve the resolution by 16.7 percentage points if he chose (a), compared to 6.9 if he chose (b).

These findings are not surprising considering that within the cardiac cycle, the maximum displacements of the coronary arteries have been reported to yield up to 41.78 mm [197] for the right coronary artery in the anteroposterior direction. In that study, angiographic breath hold images of several patients have been analyzed. Different features

Table 1.1.: Cardiac Motion: maximal displacements (MD) and mean maximal displacements (MMD) of selected features (in millimeters) for different views (FP = frontal projections, LP = lateral projections) in different directions (LR = left-right, IS = inferior-superior, PA = posterior-anterior). Values are taken from Wang *et al.* [197].

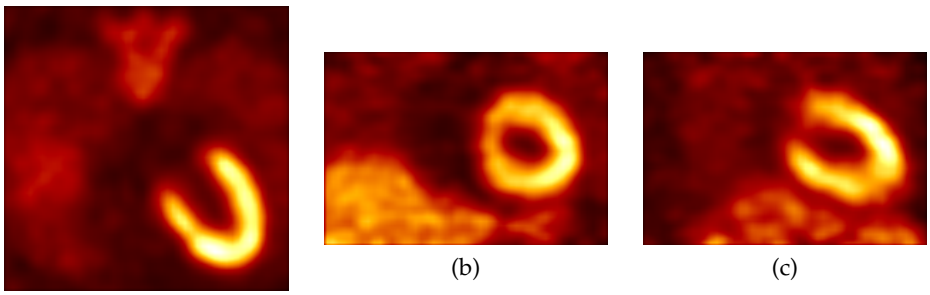
		Left Coronary Artery	Right Coronary Artery
FP	MD LR	3.14 - 20	8.75 - 30.0
	MMD LR	7.57	18.43
	MD IS	4.5 - 16.0	2.75 - 15.25
	MMD IS	8.92	8.89
LP	MD PA	4.25 - 16.57	6.0 - 41.78
	MMD PA	7.72	23.72
	MD IS	2.57 - 13.66	3 - 22
	MMD IS	7.96	9.86

of the left coronary arteries and right coronary arteries have been selected and their displacements for two different views (frontal and lateral projections) has been calculated. Their values are summarized in Table 1.1.

Maximal respiratory motion for the heart has been reported to be 8.1 mm (mean maximum 4.9 ± 1.9 mm) in the inferior-superior direction, -5.1 mm (mean maximum -1.3 ± 1.8 mm) in the anterior-posterior direction and 4.7 mm (mean maximum 0.4 ± 2.0 mm) in left-right direction [186]. In another study, translational motion was reported to be 11.9 mm at maximum (mean maximum 6.7 ± 3.0 mm) for the heart apex, 17.1 mm (mean 12.0 ± 3.7 mm) for the left kidney and 18.8 (mean maximum 12.0 ± 3.7 mm) for the right kidney [21]. Yet another study reports 13 ± 5 mm mean maximal displacement for the diaphragm and 9 ± 3 mm mean maximal displacement for the left main coronary artery [48].

Besides cardiac motion, head motion is another source of degradation in brain studies. Green *et al.* showed that for certain markers close to the head, the movement was up to five millimeters [78]. At the beginning of the study the movement was relatively low (not more than two millimeters), but then it increased approximately linearly to said maximal value after 40 minutes.

As all these studies show, both cardiac and respiratory motion displacements are much higher than today's PET physical resolution, which is approaching the one millimeter mark. If motion is simply ignored, this leads to a blurred image, as shown in Figure 1.1. The effect is comparable to taking a picture with a camera at night: since the camera's exposure time at night has to be much higher than at day, even small movements of the camera result in an image which consists of several overlaying perspectives - the image is blurred. Motion induces a huge gap between the effective resolution achieved by image reconstruction and the physical resolution. The objective of motion compensation is to close this gap.



(a) (b) (c)

Figure 1.1.: Images reconstructed with a standard static reconstruction method, ignoring motion. (a): transverse slice; (b): coronal slice; (c): sagittal slice. The data is simulated (refer to section 5.1 for details).

1.2. Dealing With Motion in PET

1.2.1. Motion Avoidance

One way to tackle motion blur is to avoid motion. Motion could be avoided by anesthesia, which is actually done for small animal studies where one can not hope on cooperative behavior by the animal. Depending on the type of anesthesia and radioactive tracer used, the influence of anesthesia on the tracer take-up can be big, and thus care has to be taken [120, 60]. In human studies anesthesia are usually not used since patients are cooperative and try to avoid motion as good as possible for the sake of a successful diagnosis. Accordingly, whole body movements are usually not the case. Furthermore, cardiac or respiratory motion can hardly be stopped by anesthesia, and thus motion artifacts related to such motion can not be avoided. In human brain studies, anesthesia is also not commonly used as it influences significantly the tracer take-up [1, 3, 2, 82]. It seems that in general, the advantage of avoiding motion by anesthesia is outweighed by the disadvantage of altering the tracer take-up (there might be specific application where this general statement is not true).

Another way to avoid motion in brain imaging is to fix the patient's head with a head restraint. Cruel ways to do this involve the placement of screws in the calvarium [10], whereas more moderate methods use molds [10] or a thermoplastic head immobilization system [157]. Nevertheless, motion can only be alleviated but not avoided completely by such methods. Green *et al.* found that a head restraint reduces the average motion, depending on the location of the markers, by factors of about one to five [78].

1.2.2. Motion Compensation

Instead of avoiding motion, motion compensation algorithms work with the motion contaminated data and seek for images as they would have been acquired had the subject not moved.

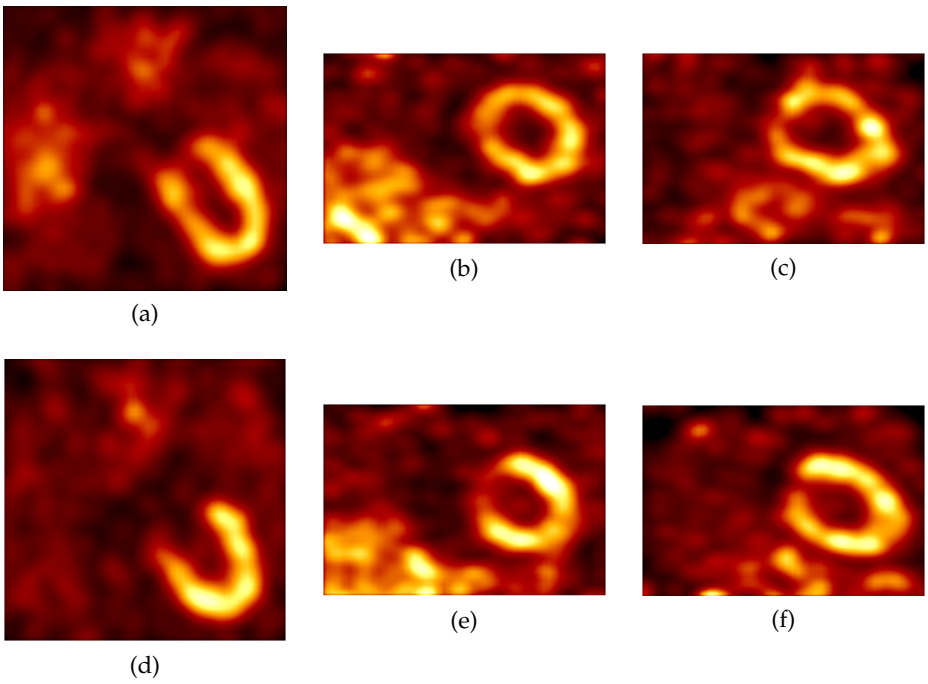


Figure 1.2.: Two selected gates, reconstructed individually. (a) - (c): inspiration; (d) - (f): expiration.

1.2.2.1. Gating and Framing

A very popular method for dealing with motion is gating or framing. The general idea is to partition the recorded data into gates (for cardiac and/or respiratory motion) or frames (brain studies). The partitioning is made in such manner that each gate/frame only contains data which approximately correspond to the same anatomical motion state. In cardiac applications this is achieved with an electrocardiogram (ECG) signal or a respiratory belt. In brain studies where no periodic motion is present, the motion is monitored and then according to its most proximate position the data is attributed to a certain frame. Once the gates/frames are defined, they are individually reconstructed by any standard reconstruction method. In terms of the camera comparison that we made before, framing would correspond to taking several pictures with a shorter shutter time. Each of these pictures then is less affected by motion.

Figure 1.2 shows some reconstructed cardiac gates. A major problem of gating is that the statistical data for each gate is reduced. For N gates, only approximately $1/N$-th of the total counts is available in each gate. The number of gates has to be chosen carefully. On the one hand, the more gates/frames are defined, the less inter frame motion is present. On the other hand, the more gates/frames are defined, the less statistical data is available for each gate/frame and accordingly the more noisy are the reconstructed images.

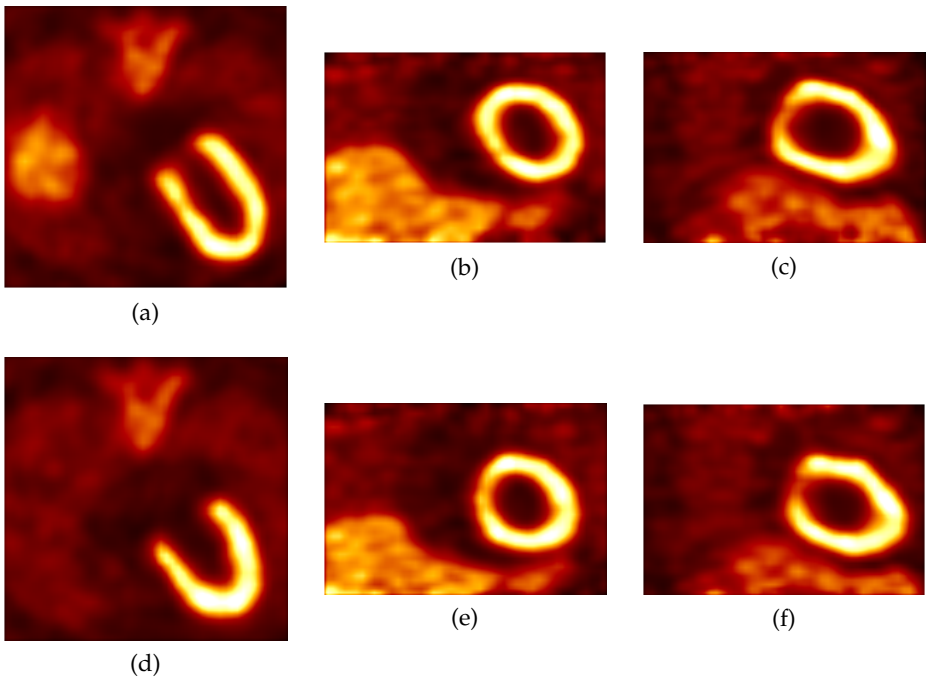


Figure 1.3.: Two selected gates (the same as in the previously shown figures), reconstructed with our joint reconstruction method.

1.2.2.2. Our Approach: Joint Reconstruction

In this work, we present a motion compensation method that works for gated data. Instead of reconstructing each gate individually, it uses all available statistical data and thus is able to reconstruct gates with much less noise. This is achieved by a complete mathematical model containing the measured data for each gate, the image subject to reconstruction and the motion subject to reconstruction. Both image and motion are parameters which are sought in the reconstruction algorithm (not only the image as this is usually the case). We present an algorithm in order to find a pair of image and motion which fits well to this model. Figure 1.3 shows the resulting gates, reconstructed with our joint reconstruction method.

In this thesis, we choose human heart motion as a general motion scenario. The main reason for this choice is that human heart motion is extremely complex since it is constituted of both respiratory motion and cardiac motion. Our main motivation for the choice of such a complex type of motion was that an algorithm which works for a complex the scenario should be more likely to also work in a simple scenario than vice versa. Apart from that, practical issues (such as the data we had access to) played a role in the decision.

1.3. Contributions

We claim to make the following major contributions to the scientific community within the scope of this thesis:

- We present a novel joint reconstruction method. Our method reconstructs both image and motion in a joint fashion by optimizing an objective function. Motion is parametrized either by displacement fields or by 4D B-splines. Two different regularization approaches are evaluated.
- We make a visual and quantitative comparison of our presented method with the most common methods used in literature.
- We develop a thorough and consistent review of current motion compensation techniques. Each of the compared methods is derived from the same dynamic model. This results in a mathematical derivation of the rebinning method, which is - to the best of our knowledge - up to now only presented in an intuitive manner in the literature. Also, our derivation allows the modeling of the motion in the system matrix to be viewed from both an LOR perspective and an image space perspective.

2. Basics of Positron Emission Tomography

In the following, we will give a brief introduction to the physics of positron emission tomography (PET) as well as image reconstruction techniques. Our discussion is mainly based on an excellent review article by Leahy and Qi [119], the classical book on emission tomography by Wernick and Aarsvold [200] and another classical PET book by Bailey *et al.* [8].

2.1. PET Physics

PET is a non-invasive imaging technology used in nuclear medicine and pre-clinical studies using animals. A main application for PET in clinical routine is oncology, where it can be used for diagnosis, staging and monitoring of cancers. PET is a functional imaging modality since it is targeted at physiological processes (like metabolism or regional chemical concentrations). Structural information, if at all, is only indirectly measured. The concrete functional information that is sought in a PET scan depends on the injected tracer. In this sense, PET is an extremely flexible imaging modality whose applications are limited only by the research advances in the area of radioactive tracers.

Before a PET acquisition, a positron-emitting radiotracer is injected in the patient's body. There exists a large number of radiotracers, depending on the type of study which is going to be performed. The most used radiotracer is ^{18}FDG (Fluorodeoxyglucose), a glucose analogue, which accumulates in regions of high glucose uptake. ^{18}FDG is typically used in cancer detection and staging since cancer cells have a high glucose uptake. During a waiting period the active molecule becomes concentrated in tissues of interest. For ^{18}FDG the waiting period is typically an hour. After accomplishing the waiting period, the subject is placed in the PET scanner.

An emitted positron interacts with surrounding electrons in the body and looses kinetic energy with each interaction. When its energy is below a certain threshold, it annihilates with an electron and produces two back-to-back gamma quanta of 511 keV energy. The angle of the two photons is approximately 180 ± 1 degrees. We call the straight line on which the two gammas are traveling the line-of-response (LOR).

Note that other definitions of an LOR are found in the literature, e. g. it is common to define an LOR as the line connecting the center of the detector surfaces where the two back-to-back quanta have been detected. This may be a totally different line than the initial traveling line, since one or both of the quanta may be scattered. For a discussion on motion compensation however, we found it more convenient to use our proposed definition.

As illustrated in Figure 2.1, a PET scanner usually consists of one or more detector rings which then detect the emitted photons. For image reconstruction, it is important

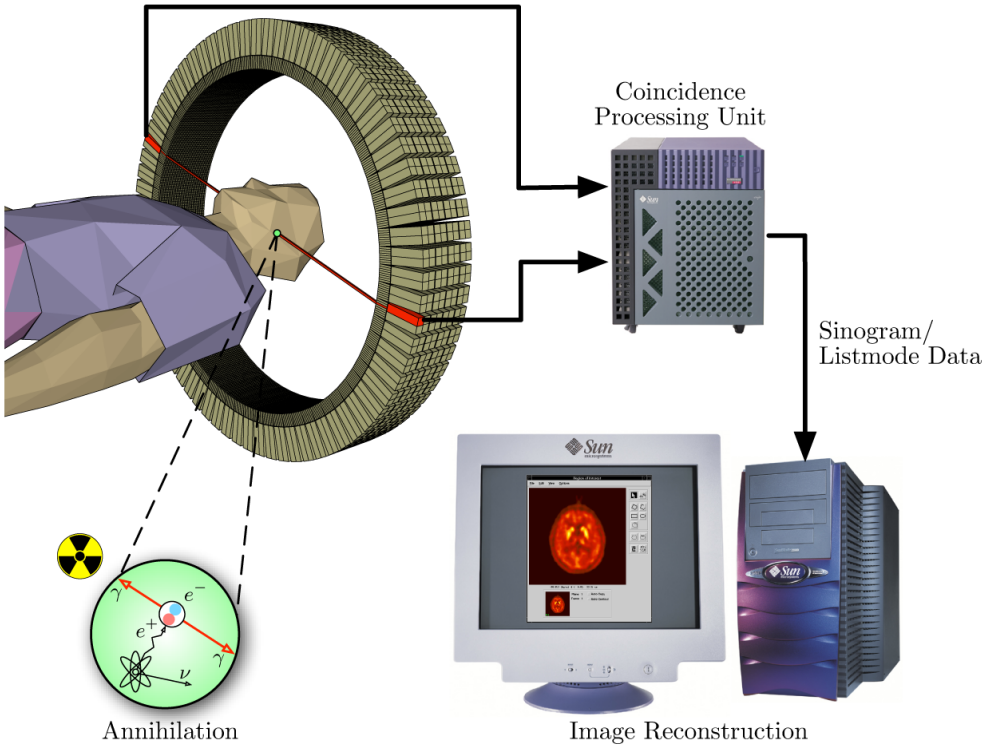


Figure 2.1.: Illustration of the principals of a PET acquisition: two back-to-back gammas are emitted and registered by two detectors in the detector ring. A coincidence sorting unit decides whether the photons belong to the same annihilation event. The image is then reconstructed on the base of such coincidence events. The image was taken from Jens Langner [117] (it is released to public domain).

to know which photons originate from the same annihilation process. This information is extracted by a coincidence sorting unit. The most common current approach is to use a time window: the first event "opens" the time window, and if a second event is measured within the time window, they are interpreted as a coincidence event. Time windows are also used for scatter and randoms estimation (see section 2.2.3). The time window approach is often implemented directly in hardware and the individual single events are not stored. This is an unfortunate situation from a researcher's perspective, since valuable information is lost and research on novel coincidence sorting algorithms is rendered more difficult.

A coincidence event is identified by an unique identification number and a timestamp. The identification number is related to the detector pair in which the back-to-back photons were detected. This type of data is referred to as list-mode data, since the data structure used is a list of coincidence events.

In contrast to list-mode, the measured data can also be stored in histogram mode. In histogram mode, all detected events which share the same identification number are summed together. The result is a vector which contains for each identification number

Table 2.1.: Some properties of isotopes used in PET

Isotope	Half-life (min)	Maximum positron energy (MeV)	Positron range in water (FWHM in mm)
^{11}C	20.3	0.96	1.1
^{13}N	9.97	1.19	1.4
^{15}O	2.03	1.70	1.5
^{18}F	109.8	0.64	1.0
^{68}Ga	67.8	1.89	1.7
^{82}Rb	1.26	3.15	1.7

the number of detected coincidence events. A typical format to store this kind of data is the sinogram (see section 2.3.1). Note that in this format the time information, which is important for motion compensation, is lost.

2.1.1. Positron Range

The positron range is the distance the positron travels before annihilation [149]. It depends on the kinetic energy with which the positron is emitted by the radiotracer. Table 2.1 shows positron ranges for frequently used isotopes [170, 7]. Since the positron range is still rather small for many tracers compared to the intrinsic resolution of the PET scanner it is often ignored. However, with increasing scanner resolution, the tendency is to incorporate the positron range into the imaging model.

2.1.2. Attenuation

It is well known that photons interact with matter. The kind and probability of interaction depends on the photon energy. In PET, the two main types of interaction of photons with matter are Compton scattering and the photoelectric effect.

Compton scattering refers to the inelastic scattering of photons with electrons. Part of the photon's energy is transferred to the electron. The photon changes its direction (the degree of change of direction is given by the scattering angle). The higher the scattering angle, the higher the energy loss. Arthur Holly Compton observed this effect in 1923 and derived a formula describing the wavelength shift (energy loss) based on the scattering angle. He received the Nobel Prize four years later. The probability of the Compton effect to take place at all is given by the Klein-Nishina formula.

The photoelectric effect describes the phenomenon that a photon is absorbed by an electron in matter and the electron is emitted. The emitted electron is called a photoelectron. The photoelectric effect was first observed by Alexandre Edmond Becquerel. After systematic experiments of Heinrich Hertz and Wilhelm Hallwachs, Lenard showed in 1900 that the charged elements were electrons and that the maximal kinetic energy depended only on the wavelength of the light, not its intensity. The total current produced depended on the light intensity. Below a certain frequency, no current was measured at

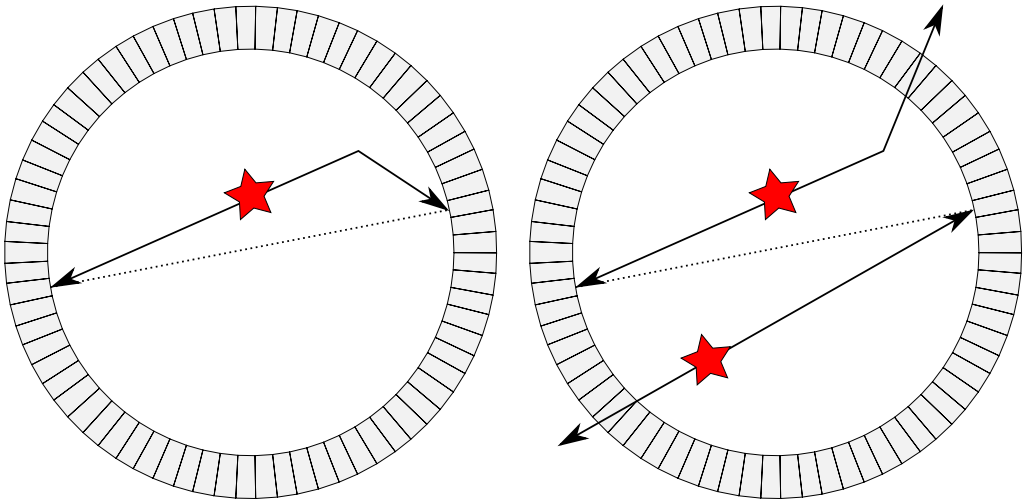


Figure 2.2.: (a) A scattered coincidence. (b) A random coincidence.

all. This was explained by Albert Einstein in 1905 by his theory of quantization of light, for which he received the Nobel Prize in physics.

For PET, the predominant photon-matter interaction effect is Compton scattering. Since the photoelectric is dominant in human tissue for energies below 100 keV, its importance in PET (where annihilation photons have an energy of 511 keV) is negligible. Compton scattering dominates in human tissue for energies between 100 keV and 2 MeV [8].

Both Compton scattering and the photoelectric effect lead to attenuation, since they cause photons not to be "seen" by a certain detector pair. The attenuation factor is the probability of a photon to *not* interact with matter while traveling through an attenuating medium (attenuation coefficient μ) for a distance d. For a well collimated source, the attenuation factor is given by

$$e^{-\mu d} \quad . \tag{2.1}$$

Since in PET two photons have to be counted by two detectors (let the detector surfaces be separated by a distance D), the probability that both photons arrive at the detector surface is $e^{-\mu(D-d)} \cdot e^{-\mu d} = e^{-\mu D}$. Therefore, the attenuation factor does not depend on the spatial position of the annihilation (this makes attenuation correction especially simple compared to e. g. SPECT)!

2.1.3. Scattered Coincidences

It often happens that even though one or both of the photons are scattered (thus changed their direction), they are still detected in coincidence (Figure 2.2a). This type of coincidence events is called scattered coincidence. If not treated by the reconstruction algorithm, scattered coincidences degrade the image quality since the reconstruction algo-

rithm assumes that the event happened on a different LOR than it actually did. As a simple pre-processing step, a significant amount of such scattered events can be filtered out by considering the energy of the detected photon (the photons loose energy in a Compton interaction).

2.1.4. Randoms

If only one photon is detected (since the other escaped the FOV due to scattering or simply because its initial direction was not covered by any detector), this is called a *single* (Figure 2.2b). If two uncorrelated singles happen within the same coincidence window, this is called a *random* or *accidental coincidence*.

2.1.5. Detector Efficiency

The detector efficiency is the fraction of photons of a certain energy stopped by the crystal. It depends on several geometric and non-geometric factors [146, 147].

Geometric factors are classified into

- Radial distance of the LOR: obliquity and solid angle subtended by the detectors
- Position of crystal in a block: energy resolution, probability of scattering out of the block
- Angle of incidence of photon: probability of interaction in the block, e. g. very low angle photons are more likely to travel within a gap of the detector and therefore not to interact at all.

Non-geometric factors are the fluctuations in readout system gains and variations of crystal characteristics.

Several models and estimation methods which deal with detector efficiency have been proposed [34, 52, 144]. When scattered events are included, these considerations are not valid anymore, due to the change of the direction (which affects the geometric factors) and the lower energy (affects the non-geometric factors) [146, 147].

2.1.6. Detector Dead Time

The detector dead time is the amount of time after an event in which the detector cannot detect new events (it is "dead") [87]. Detector dead time is the limiting factor for the injected dose. Before detector dead time plays a role, the number of true coincidences depends linearly on the dose, and the number of randoms depends quadratically on the trues [149]. At a certain dose level both the trues and the randoms peak and then decline due to detector saturation.

2.2. The Linear Imaging Model

In the following, we develop the commonly used PET linear imaging model step-by-step. The model encapsulates most of the previously described physical phenomena mathematically, and is of vital importance since it is the base for any reconstruction method. If a reconstruction method uses a model that ignores one of the effects, the result is a degraded image. For example, ignoring the positron range results in a blurred reconstructed image. Note that we will postpone anything related to motion to the next chapter.

We start with the system response function

$$H(a, \boldsymbol{x}) \ , \tag{2.2}$$

which models the probability of detecting a true coincidence of a positron emission from position $\boldsymbol{x}$ at a detector pair identified by an index a. The number a can be chosen arbitrarily, as long as it is possible to geometrically identify the detector pair with it.

The system response function is of fundamental importance since it relates the unknown tracer distribution to the number of counts which we expect to measure in a detector pair. We denote the tracer distribution as a function $f : \mathbb{R}^3 \mapsto \mathbb{R}$, where $f(\boldsymbol{x})$ stands for the density of the expected number of annihilation events to happen during the whole scan. This implies that the integral

$$\int_{\omega} f(\boldsymbol{x}) \, d\boldsymbol{x} \tag{2.3}$$

is the expected number of counts to be emitted during the scan for a sub-volume ω. The integral

$$\int_{\omega} H(a, \boldsymbol{x}) f(\boldsymbol{x}) \, d\boldsymbol{x} \tag{2.4}$$

is then the expected number of annihilation events to be emitted from region ω and detected at detector pair a. If we integrate over the whole image support Ω, we get

$$\hat{g}(a) = \int_{\Omega} H(a, \boldsymbol{x}) f(\boldsymbol{x}) \, d\boldsymbol{x} \ , \tag{2.5}$$

which is the expected number of counts measured in detector pair a during the scan. The system response function H thus defines a linear relationship between the tracer density and the expected number of counts to be measured in a certain detector pair.

Up to now, only true coincidences have been considered. Scattered and random coincidences are usually included as additive terms

$$\hat{g}(a) = \int_{\Omega} H(a, \boldsymbol{x}) f(\boldsymbol{x}) \, d\boldsymbol{x} + s(a) + r(a) \ , \tag{2.6}$$

where $s(a)$ is expected number of scattered events and $r(a)$ the expected number of randoms to be detected in detector pair a. While scatter could be also directly modeled in the system matrix [171], this is not possible for the randoms.

2.2.1. Discretization

Although it would be possible to stay within the continuous domain, it is convenient to discretize the model, since the system matrix can then be factored to a matrix-matrix product of several matrices (see next section). Each of these matrix factors then represents one of the previously described physical effects. The system model is discretized by

$$\hat{\boldsymbol{g}} = \boldsymbol{H}\boldsymbol{f} + \boldsymbol{s} + \boldsymbol{r} \ , \tag{2.7}$$

where $\boldsymbol{H}$ is the discretized system response function and usually called the system matrix, $\boldsymbol{f}$ is the discretized image vector, $\boldsymbol{s}$ the discretized scatter vector (defined by $[\boldsymbol{s}]_a := s(a)$) and $\boldsymbol{r}$ the discretized randoms vector (defined by $[\boldsymbol{r}]_a := r(a)$). For H and f, discretization is done by defining the continuous functions as a sum of basis functions:

$$f(\boldsymbol{x}) = \sum_k^{N^{\boldsymbol{f}}} [\boldsymbol{f}]_k [\boldsymbol{v}]_k(\boldsymbol{x}) \ , \tag{2.8}$$

$$H(a, \boldsymbol{x}) = \sum_k^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{ak} [\boldsymbol{v}]_k(\boldsymbol{x}) \ . \tag{2.9}$$

$[\boldsymbol{v}]_k$ is the k-th image space basis function. Usually voxel basis functions are used for image reconstruction, but also other types of basis functions like blobs have been investigated [124, 136, 4, 83, 29]. The voxels are usually numbered consecutively in lexicographical order (that is, first in x-direction, then y and then z). There are $N^{\boldsymbol{f}}$ total voxels ($N^{\boldsymbol{f}} = N_x^{\boldsymbol{f}} N_y^{\boldsymbol{f}} N_z^{\boldsymbol{f}}$).

2.2.2. The System Matrix

The system matrix ideally should model everything apart from accidental coincidences. In this discussion we assume that scatter is modeled as an additive vector outside the system matrix as explained before, simply because most researchers handle it that way. It is convenient for reasons of memory and computational efficient to factor the system matrix into

$$\boldsymbol{H} = \boldsymbol{H}_{\mathrm{e}} \boldsymbol{H}_{\mathrm{b}} \boldsymbol{H}_{\mathrm{a}} \boldsymbol{H}_{\mathrm{g}} \boldsymbol{H}_{\mathrm{p}} \ , \tag{2.10}$$

where $\boldsymbol{H}_{\mathrm{p}}$ is a $N^{\boldsymbol{f}} \times N^{\boldsymbol{f}}$ square matrix that models the positron range, $\boldsymbol{H}_{\mathrm{g}}$ is a $N^{\boldsymbol{g}} \times N^{\boldsymbol{f}}$ matrix that models the system geometry, $\boldsymbol{H}_{\mathrm{a}}$ is a $N^{\boldsymbol{g}} \times N^{\boldsymbol{g}}$ square diagonal matrix that models the photon attenuation, $\boldsymbol{H}_{\mathrm{b}}$ is a $N^{\boldsymbol{g}} \times N^{\boldsymbol{g}}$ square matrix that models the detector blur and $\boldsymbol{H}_{\mathrm{e}}$ is a $N^{\boldsymbol{g}} \times N^{\boldsymbol{g}}$ square diagonal matrix that models the detector efficiency [160, 159, 161, 119]. In the following sections, we will describe in more detail each of these factors.

It is also possible to calculate the system matrix by means of Monte Carlo methods [164, 206]. The main advantage of Monte Carlo methods for system matrix calculation is that they are usually more exact and can consider all physical phenomena. A disadvantage is the rather long calculation time and storage requirements.

Any matrix-vector product with the system matrix is called a forward projection, since it projects an image vector to measurement space. The opposite of a forward projection is a back projection, which is performed by multiplying a measurement vector with the transposed system matrix.

2.2.2.1. Positron Range Matrix

The effect of positron range is basically a blurring of the initial tracer distribution. It takes account for the fact that we are not interested in position of the positron-electron annihilation (from which the back-to-back gammas are emitted), but rather in the position of the positron emission since this is where the tracer molecule is located. Positron range can be therefore modeled by a square matrix $\boldsymbol{H}_\mathrm{p}$ which has a multi bend structure. Since it is the rightmost matrix, the image is first multiplied by it and $\boldsymbol{H}_\mathrm{p}\boldsymbol{f}$ results in a blurred image (mathematically the matrix-vector product is a discrete convolution). Most authors are ignoring the positron range since its effect is usually below the resolution of the scanner. However, for some studies especially with high energy positron emitters like ^{82}Rb it has been shown to be worth taking it into account [57, 169]. Both analytical models and Monte Carlo simulations have been used for modeling the positron range [169].

2.2.2.2. Geometric Matrix

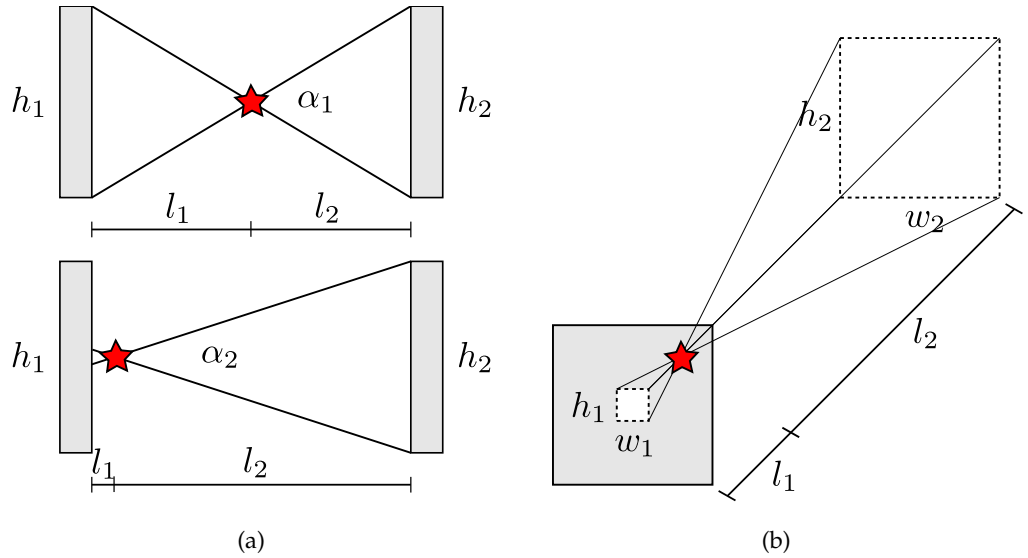
The geometric matrix ideally models the geometric probability of an emission in voxel $[\boldsymbol{v}]_k$ being detected at the *surfaces* of detector pair a. It assumes an ideal detector which detects every photon that touches its surface. Thus, the detection probability only depends on the angular coverage of the detector pair with the detector surfaces. The geometric matrix is by far the most important factor - without it, image reconstruction is not possible.

There are several methods for calculating the geometric matrix of the system matrix. In the literature, a classification into pixel-driven and ray-driven methods is made [132]. The classification is made from an algorithmic point-of-view: pixel-driven methods get a pixel index as input and calculate the weight of that pixel with all detector pairs, whereas ray-driven methods get a detector pair index as input and calculate the weight of that detector pair with all pixels.

There should not be a difference between the weight of a pixel with a detector pair given by a pixel-driven or ray-driven method. However, since in both types of methods approximations which are convenient for either case are made, in practice there is a difference in the resulting weights.

One should keep in mind that most of the literature dealing the geometric matrix comes from the CT community, and sometimes it is worth questioning its applicability to PET. We will discuss the literature from a PET perspective, so e. g., instead of talking about bins we will talk about detector pairs. Also, we will call the pixel-driven methods voxel-driven methods since this makes more sense for 3D PET.

2.2.2.2.1. Voxel-Driven Methods Voxel-driven methods operate from the point-of-view of a single voxel. They calculate the contribution of that voxel to different detector pairs.



(a) (b)

Figure 2.3.: Considerations of the geometric probability of detecting an event at a detector pair. (a) In 2D, the probability is proportional to the angle comprised by lines connecting the detector surfaces with the annihilation position. (b) In 3D, there is not a simple extension to the 2D concept. Instead, the surface of a rectangle projected to the detector surfaces can be used (see text).

From an algorithmic point-of-view, the input is a voxel index and the output is a list of weights for the different detector pairs. Voxel-driven methods naturally perform a back projection in which an algorithm iterates over the pixels in an outer loop and then seeks the contribution of each detector pair in an inner loop. Nevertheless, they can also be used for forward-projection by exchanging indices.

The simplest voxel-driven method checks whether or not there exists a straight line which connects the two detector surfaces (it does not matter where exactly) and the voxel center. If this is the case, the system matrix element corresponding to the detector pairs and the respective voxel is given a weight of one, otherwise zero. This simple method can be improved by dividing the voxel into subvoxels [154, 207, 83]. Then, the weight of a voxel with a detector pair is given by the number of centers of subvoxels for which a straight line between two detector surfaces exists, divided by the total number of subvoxels.

A more exact approach is the angle-of-view approach [36], depicted in Figure 2.3. It is based on the solid angle of all LORs that connect the emission position with both detector surfaces (from now on, we will colloquially say the detector pair "sees" the LOR). All LORs which could be possibly seen by the two detector surfaces define two (generally differently sized) rectangles on the detector surfaces (one rectangle for each detector surface). The solid angle of either of these rectangles with the emission position $\boldsymbol{x}$ (both solid angles are equal) is then a measure for the angular coverage of this detector pair. Note that the solid angle of an arbitrary surface subtended by a point is given by the area

of the projection of the surface to the unit sphere around the point.

In [36], a simple approximation

$$H(a, \boldsymbol{x}) = \frac{w \cdot h}{l} \tag{2.11}$$

of the solid angle is used (where w is the width of the rectangle, h its height and l the distance from the annihilation position $\boldsymbol{x}$ to the surface).

For a discrete system matrix, we need the probability that an annihilation in a voxel $[\boldsymbol{v}]_k$ is detected in detector pair a. This probability is given by the volume integral

$$[\boldsymbol{H}]_{ak} = \int_{\omega_k} H(a, \boldsymbol{x})\, d\boldsymbol{x} \ , \tag{2.12}$$

where ω_k is the volume covered by the voxel k. This integral can be approximated by a sum.

Since even with this approximation the solid angle is extremely computationally intensive, it mostly attracted attention in the small animal PET literature [190, 159], where a highly sophisticated geometric model is necessary.

2.2.2.2.2. Ray-Driven Methods As mentioned, the input of ray-driven methods is a detector pair and the output a list of voxels and corresponding weights. Ray-driven methods can be thought of an intent to approximate the estimated number of counts measured in a detector pair a

$$\hat{g}(a) = \int_{\Omega} H(a, \boldsymbol{x}) f(\boldsymbol{x})\, d\boldsymbol{x} \tag{2.13}$$

by a line integral

$$\hat{g}(a) \approx \int_{L} f\, ds \ , \tag{2.14}$$

where L is a straight line combining the centers of the detector surfaces. The line integral can be written as

$$\hat{g}(a) = \int_0^1 f(\boldsymbol{r}(s))\, |\boldsymbol{r}'(s)|\, ds \ , \tag{2.15}$$

where $\boldsymbol{r}$ is a parametrization of the line. Since the derivative of $\boldsymbol{r}$ with respect to s is constant, we can write

$$\hat{g}(a) = c \cdot \int_0^1 f(\boldsymbol{r}(s))\, ds \ , \tag{2.16}$$

Now, the discretization of f can be substituted

$$\hat{g}(a) = c \cdot \int_0^1 \left(\sum_k^{N^{\boldsymbol{f}}} [\boldsymbol{f}]_k [\boldsymbol{v}]_k (\boldsymbol{r}(s)) \right) ds \ , \tag{2.17}$$

which is equal to

$$\hat{g}(a) = c \cdot \sum_{k}^{N^{\boldsymbol{f}}} [\boldsymbol{f}]_k \int_0^1 [\boldsymbol{v}]_k(\boldsymbol{r}(s))\, ds \quad . \tag{2.18}$$

$\int_0^1 [\boldsymbol{v}]_k(\boldsymbol{r}(s))\, ds$ is the intersection length of a straight line (defined by $\boldsymbol{r}$) with voxel k.

This approximation is accurate when the detector surfaces are relatively small compared to a voxel. In this case, the only contributions from a voxel to a detector pair can lie on a straight line combining the detector surfaces (ideally two points) and thus the probability of an emission of a voxel with a detector pair is proportional to the length of the line segment. Since usually a voxel's edge length is smaller than the detector surface's side length, it is recommendable to approximate $\hat{g}(a)$ by several line integrals which ideally uniformly cover the detector surfaces (note that these lines are in general not parallel one to another).

Since the seminal paper of Siddon [188], in which a relatively simple and fast method for the calculation of the intersection length of a straight line with a voxel was presented, such ray-driven methods have received great attraction in the PET community. Further modifications and comparison studies have been performed in [207, 95, 205, 83].

Other attempts include interpolation of two or more voxels [98, 201, 207]. From a purist's point-of-view, the interpolation is in fact nothing else than using a different basis function than voxels, which rises the question how in general should be dealt with basis functions other than voxels (the same question would be applicable to voxel-driven methods, however, then they would not be called voxel-driven anymore - so we discuss it here). The question is answered by a quick look at equation (2.18), which suggests that the integral $\int_0^1 [\boldsymbol{v}]_k(\boldsymbol{r}(s))\, ds$ is not the intersection length of a ray with a voxel anymore but rather has to be approximated adequately regarding the new basis function.

Another method which belongs to the class of ray-driven methods (although the term tube-driven would be more appropriate) is proposed by Scheins *et al.* [178]. The general idea of [178] is to approximate the expected number of counts $\hat{g}$ by a volumetric integral

$$\hat{g}(a) = \int_T f\, dV \tag{2.19}$$

of the image f over a tube T which is a polyhedron defined by the two detector surfaces. Note that this method is not exact, even in a geometric sense, which can be explained by a simple argument: we suppose that a voxel is completely inside the polyhedron defined by two detector surfaces. Then, Schein's method returns the same contribution of that voxel to the detector pair regardless of the position of this voxel, as long as it stays within the polyhedron. However, in reality, a voxel which is close to one of the detectors has a lower contribution than a voxel that is centered in the middle of the two detectors.

In general, ray-driven methods are excellent for the forward projection, but can lead to artifacts in the back projection according to [51]. However, further investigation may be needed in order to determine whether these artifacts really occur for a good approximation.

2.2.2.2.3. Distance-Driven Method A method which is primarily aimed at CT and has not yet been used for PET (to the best of our knowledge) was presented by De Man and Basu [51, 131, 132]. De Man and Basu claim that both voxel-driven and ray-driven methods lead to artifacts due to their nature. Their proposed method can be considered as a mixture between voxel-driven and ray-driven methods. It allows for a fast implementation.

2.2.2.3. Attenuation Matrix

Recall from section 2.1.2 that the attenuation factor is the probability of photons *not* interacting with matter as they travel. For simplicity, we described the attenuation factor in terms of a constant attenuation coefficient. Since in PET one usually deals with inhomogeneous matter (the human body!), the attenuation coefficient is a function of space (not constant). We therefore adopt the attenuation factor to

$$e^{-\int_{L(a)} \mu \, ds} \quad , \tag{2.20}$$

where $\int_{L(a)} \mu \, ds$ is the line integral over the attenuation coefficient $\mu : \mathbb{R}^3 \mapsto \mathbb{R}$. The line $L(a)$ is the line connecting the two detector pairs identified by index a. Note that $L(a)$ is an approximation, since the true attenuation depends on the exact detection position of the photons in the detector pair. If a corresponds to a bin, that is, several mashed up detector pairs, then this approximation becomes more inaccurate.

The attenuation matrix contains the attenuation factors of all detector pairs and is written as

$$[\boldsymbol{H}]_{aa} = e^{-\int_{L(a)} \mu \, ds} \quad . \tag{2.21}$$

It is a diagonal matrix.

2.2.2.4. Detector Blur Matrix

In order to further refine the system matrix, the detector blur matrix includes effects like non-collinearity of the emitted photons, inter-crystal scatter and false attribution due to penetration depth (a photon may pass undetected through a detector and then be detected by a neighboring detector). These effects are difficult to model mathematically and thus may be included by terms of a Monte Carlo simulation [119]. Note that the detector blur matrix should still maintain the total number of counts detected in an detector pair, that is, $\sum_i [\boldsymbol{H}_\mathrm{b}]_{ai}$ should be equal to one.

2.2.2.5. Detector Efficiency matrix

Even if both photons reach the detector surfaces, they may not be detected due to physical properties of the detector or the readout electronics. This photon loss within the detector is accounted for in the detector efficiency matrix. It is a simple square diagonal matrix which defines for each detector pair the probability of being actually detected, once reaching the detector.

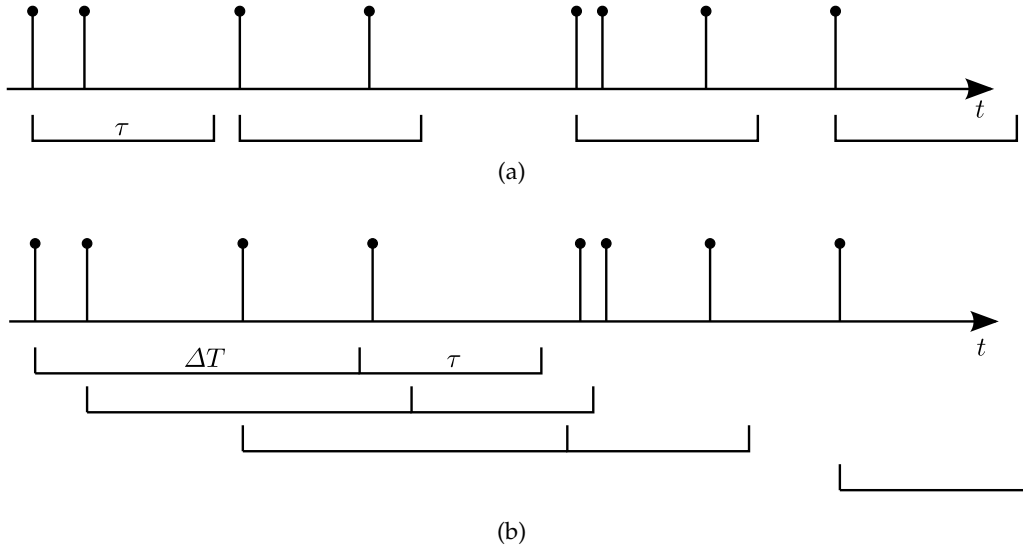


Figure 2.4.: Single events are drawn on a time bar. A normal coincidence window (a) measures true, scattered and random coincidences. The delayed coincidence window (b) just measures the random coincidences. $\Delta T \gg \tau$.

2.2.3. Scatter and Randoms Estimation

With the appearance of fully 3D PET scanners (without septa), scatter and randoms have become an important source of degradation of image quality which has to be corrected for. The main effect that scatter and randoms impose on the reconstructed image is noise.

For some commercial scanners, the number of scattered coincidences is more than 30 percent of the total number of coincidences [9]. Model-based approaches such as [150, 198, 148, 143] have established themselves among the most accurate methods to estimate scatter [9]. An estimate of the emission data and the transmission data is needed beforehand. An often made assumption for model based methods is the single-scatter assumption, which means that one of the annihilation photons is scattered exactly once, the other one hitting the detector unscattered. Many scatter positions are evaluated for a specific LOR (the LOR is defined before scatter occurred). The number of photons traveling towards the scatter position can be calculated using the estimated emission image. Then, the total number of photons being actually scattered at the assumed scatter position can be calculated from the transmission images. Finally, the fraction of photons which travel towards other detectors where they are ultimately detected is calculated using the Klein-Nishina formula for Compton scatter. The whole process is repeated for many LORs and scatter positions.

The random coincidences $\boldsymbol{r}$ are often estimated from the data itself by a delayed coincidence window approach [88] (Figure 2.4): within the normal coincidence window, true, scattered and random coincidences are measured. The first event "opens" a coincidence window and if within this window of length τ there is another event it is considered as a true coincidence. Now, if that window is delayed by a certain amount of time $\Delta T \gg \tau$, then any other event falling within that window can not belong to a true or scattered

coincidence. The number of events which fall into the delayed coincidence window over the whole acquisition must be approximately the same as the number of random events in the normal coincidence window. This gives a rough estimate of the number of random coincidences which can be expected. Other approaches for estimating the randoms include using a smoothed delayed window or singles-rate based (which is of equal importance as the delayed window approach) estimation [22].

Knowing $\boldsymbol{r}$ and $\boldsymbol{s}$ approximately, it is tempting to just subtract it from the measured data and thus pre-correct it. For $\boldsymbol{r}$ this is usually done on-the-fly during data acquisition, and many scanners even do not explicitly record the randoms rate and just save a pre-corrected measurement vector $\boldsymbol{g}$. Unfortunately, such pre-corrected data does not obey a Poisson distribution anymore, which is a common assumption of many reconstruction algorithms, and thus must be treated with care. Since the ideal distribution of randoms subtracted data is numerically unfeasible, approximations like the shifted Poisson model have been proposed [202, 203].

2.3. Data Storage

The linear imaging model, as discussed previously, represents an enormous amount of data. Both the measured data and the system matrix have to be kept in memory in order to allow for a fast processing. The measured data is usually stored and compressed in sinograms. Special attention needs to be drawn to the compression of the system matrix, which otherwise is unhandable due to its huge size.

2.3.1. Sinograms

In histogram mode, PET data is usually stored in form of sinograms. The sinogram storage comes originally from the CT community and orders the data according to the imaging angle. Even if list-mode is used, the coincidence events are often identified by their sinogram bin. We will discuss the concepts at the example of the Siemens Biograph 16 PET-CT scanner, since this model was used in our thesis for reconstructing patient data.

Figure 2.5 shows the concept of sinograms. A sinogram is a diagram in which each point represents a detector pair. A detector pair is represented by a line connecting the centers of the detector surfaces. In a 2D sinogram, the detector pair line is parameterized by its angle and distance to the center of the detector ring. In order to reduce the amount of memory, several detector pairs can be combined to a so-called bin (see next paragraph). For the Biograph 16, there are 192×192 bins in each sinogram. A row in the sinogram consists of detector pairs whose representing lines have the same angle, and thus can be interpreted as a parallel projection of the imaging subject. Often 2D sinograms are used even for a fully 3D PET scanner. A sinogram then represents the LORs between two specific detector rings. Also 4D sinograms are employed, which then include the axial distance from the center and the tilt angle [8]. In fact, if several 2D sinograms are used, the information about axial distance and tilt angle are incorporated in the sinogram number, and thus several 2D sinograms form a 4D sinogram.

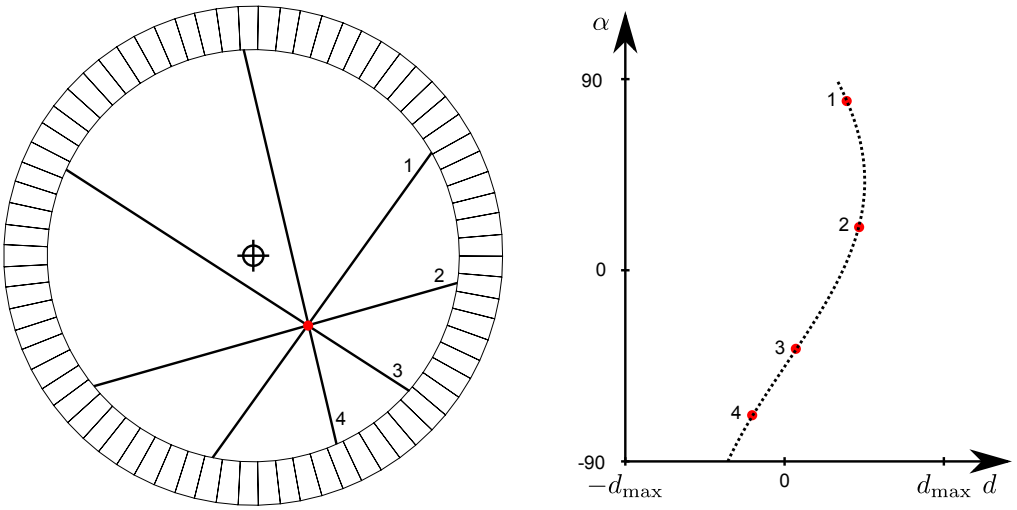


Figure 2.5.: Possible LORs of a point source (without scatter here), and the corresponding sinogram. An LOR is represented by a point in the sinogram. The dotted sine curve illustrates all possible LORs corresponding to the point source.

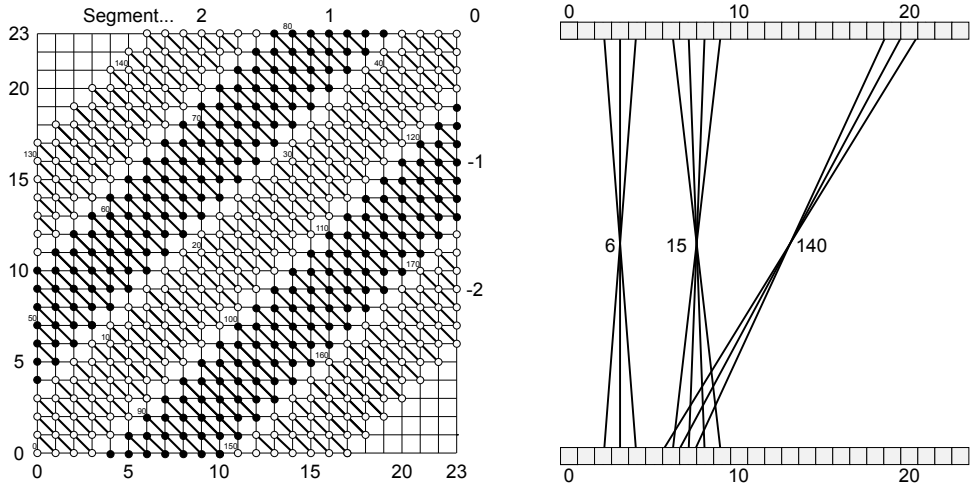
In order to reduce the amount of storage without decreasing too much the axial sampling, sinograms from different rings are combined to one sinogram [54]. The ring pairs which exactly belong to one sinogram are defined by the Michelogram. Figure 2.6a shows a possible Michelogram for the Biograph 16. Each point in the Michelogram represents a sinogram by defining the involved ring pairs. One axis stands for the first ring, the other one for the second ring. One of the detectors lies on the first ring, and the other one on the second ring. Note that, since a sinogram only comprehends an angle range of 180 degrees, only half of all possible lines between a ring pair can be covered by one sinogram (accordingly, two sinograms would be needed in order to cover all possible combinations). It does not matter which of the rings is defined as the first one, as long as a consistent definition is pursuited.

A line between different individual sinograms sums them to one combined sinogram (mashing). The number of unified sinograms alternates between an even and an odd number. The sum of both is called the *span*. In our example, there is a span of seven. A Michelogram is usually sufficiently described by the span and the maximal ring difference (which is 17 in our example).

2.3.2. System Matrix Compression

For any statistical reconstruction algorithm (as explained in the next section), the multiplication with the system matrix and its transpose is a key element which is repeated in each iteration. There are two strategies to implement this system matrix multiplication:

- The system matrix is calculated on-the-fly during the multiplication with the image/measurement vector.



(a) (b)

Figure 2.6.: (a) Michelogram of the Siemens Biograph 16 PET-CT scanner. (b) Illustration of selected sinograms.

- The system matrix is pre-calculated and stored on the disk/in the memory.

The on-the-fly technique has the big advantage that it does not need much storage, and it is also relatively simple to implement. Unfortunately, even if an extremely simple system model is used, it is quite slow. Efforts have been made towards a fast on-the-fly implementation on a graphics processing unit (GPU) [58].

Storing the system matrix in memory is feasible but some tricks have to be used since otherwise it would be too large. It has been reported that by using symmetries and efficient storage techniques, the system matrix of the microPET small animal scanner was reduced to 18 Mb for a $128 \times 128 \times 24$ number of voxels volume [159], while the system matrix of the Siemens ECAT EXACT HR+ has been reported to occupy 57 Mb for a $128 \times 128 \times 63$ volume [161]. Roughly 99% of the storage is occupied by the geometrical system matrix.

In the following, we will show how we compressed the system matrix of the Siemens Biograph 16 PET-CT scanner, since this is the scanner we work with in this thesis. We start with an estimate of how much storage would be needed if no compression at all would be applied. The Siemens Biograph 16 has 175 sinograms, each consisting of $192 \cdot 192$ bins (192 for distance sampling and 192 elements for angular sampling). Thus, in total there are $175 \cdot 192 \cdot 192 = 6,451,200$ bins. This is the number of rows of the system matrix. If the reconstruction volume is $128 \times 128 \times 48$ voxels large, this leads to $786,432$ columns. That is in total $5,073,430,118,400 \approx 5.1 \cdot 10^{12}$ elements. If each element occupies four bytes we arrive at $18,900$ Gb. It is clear that there is not feasible to store such a big amount of data, not even on a hard disk.

We make the following observation for the geometric system matrix: since a row represents the contribution of voxels with a certain detector pair (or bin), most entries of a

row are zero. The exact number depends on the length of the line intersecting the volume and on the geometrical model used. For example, the Siddon model will occupy less voxels per row than Scheins' algorithm. For this estimation, we roughly estimate that on average, about $128 \cdot 4$ voxels are distinct from zero for each row. Accordingly, we arrive at $6,451,200 \cdot 128 \cdot 4 = 3,303,014,400$ non-zero elements. We use the compressed sparse row format (CSR) in order to store sparse matrices either on the disk or in memory [177]. The CSR format is a compromise between storage and computational efficiency. The data of a matrix is stored in three vectors: $\boldsymbol{v}_{\text{val}}$ stores the values of the non-zero elements, $\boldsymbol{v}_{\text{col}}$ stores the column indices of the non-zero elements and $\boldsymbol{v}_{\text{row}}$ stores the index in $\boldsymbol{v}_{\text{val}}$ when a new row starts. For example, consider a matrix

$$A = \begin{pmatrix} 10 & 0 & 0 & 0 & -2 & 0 \\ 3 & 9 & 0 & 0 & 0 & 3 \\ 0 & 7 & 8 & 7 & 0 & 0 \\ 3 & 0 & 8 & 7 & 5 & 0 \\ 0 & 8 & 0 & 9 & 9 & 13 \\ 0 & 4 & 0 & 0 & 2 & -1 \end{pmatrix} . \tag{2.22}$$

The values are then stored in

$$\boldsymbol{v}_{\text{val}} = (10, -2, 3, 9, 3, 7, 8, 7, 3, 8, 7, 5, 8, 9, 9, 13, 4, 2, -1)^{\mathsf{T}} , \tag{2.23}$$

the column indices in

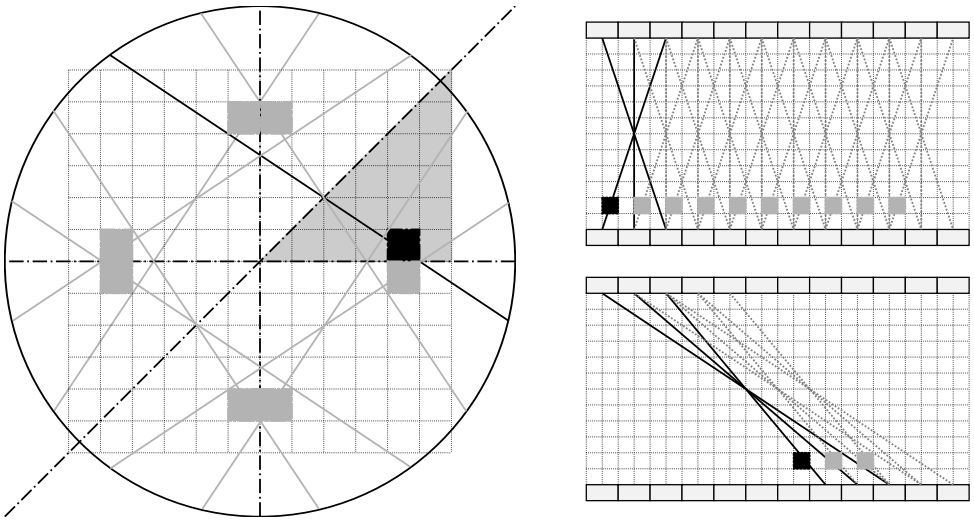
$$\boldsymbol{v}_{\text{col}} = (1, 5, 1, 2, 6, 2, 3, 4, 1, 3, 4, 5, 2, 4, 5, 6, 2, 5, 6)^{\mathsf{T}} \tag{2.24}$$

and the values indices where a new row starts is stored in

$$\boldsymbol{v}_{\text{row}} = (1, 3, 6, 9, 13, 17, 20)^{\mathsf{T}} . \tag{2.25}$$

The CSR storage format reduces the number of elements to be stored to less than three times the number of non-zero elements. The vector $\boldsymbol{v}_{\text{val}}$ occupies $4\cdot$ number of non-zero (NNZ) bytes (4 if a 32 bit floating point type is used), $\boldsymbol{v}_{\text{col}}$ also occupies $4\cdot$ NNZ bytes (a 32 bit integer type is necessary in order to store values greater than 65536, which we need since we have 786432 voxels), and $\boldsymbol{v}_{\text{row}}$ occupies less than $4\cdot$ NNZ bytes (also here a 32 bit integer type is necessary for storage and furthermore the size of $\boldsymbol{v}_{\text{row}}$ is at most the NNZ). Accordingly, an upper bound for the storage needed is $3 \cdot 4\cdot$ NNZ bytes. This means, that for $3,303,014,400$ NNZ elements which we estimated for the system matrix, we would still need roughly 37 Gb of storage.

In order to further reduce the amount of storage needed, symmetries can be exploited. The number of symmetries depends on the scanner (it would make sense to design the scanners with the symmetries in mind). In total, eight lateral symmetries can be defined: four rotational symmetries for rotation angles of $0°, 90°, 180°$ and $270°$ and one reflection symmetry at $45°$ (Figure 2.7a. The total reduction factor resulting from these symmetries is eight. Furthermore, in the axial direction, sinograms can be shift-wise repeated (Figure 2.7b shows some examples). The shift repetition depends on the sinogram mashing and the segment to which the sinogram belongs. In the Siemens Biograph 16, instead of in



(a) (b)

Figure 2.7.: (a) The probability of measuring an emission in the black voxel at the black detector pair (symbolized by the black straight line connecting the center of two detector surfaces) is geometrically the same as the probability of the respective gray voxels and lines. It is sufficient to calculate the system matrix for only one eigth of the voxel space (the gray triangle). Accordingly, the storage size reduces by factor eight. (b) Axial shift similarities, depending on the sinogram mashing and segment. Here, only two different types out of 32 types in total are illustrated.

total 175 sinograms, we arrive at only 32 different types of sinograms due to axial shift symmetries. Accordingly, we get a further factor of approximately five (the amount of saving here depends on the number of different sinogram types and therefore ultimately on the Michelogram). This should lead to a storage size of about one gigabyte, which is a pretty good approximation to the 1.1 Gb we are actually measuring from our calculated system matrix.

Technically, these symmetries can be implemented by factoring the geometric matrix as follows:

$$\boldsymbol{H}_{\mathrm{g}} = \begin{pmatrix} \boldsymbol{H_0} \\ \boldsymbol{H_1} \\ \boldsymbol{H_2} \\ \boldsymbol{H_3} \\ \boldsymbol{H_2} \\ \boldsymbol{H_3} \\ \boldsymbol{H_2} \\ \vdots \\ \boldsymbol{H_{25}} \\ \boldsymbol{H_{26}} \\ \boldsymbol{H_{25}} \\ \boldsymbol{H_{26}} \\ \boldsymbol{H_{27}} \\ \boldsymbol{H_{28}} \\ \boldsymbol{H_{29}} \\ \boldsymbol{H_{30}} \\ \boldsymbol{H_{31}} \end{pmatrix} + \boldsymbol{P}_{\boldsymbol{1}}^{\boldsymbol{g}\mathsf{T}} \begin{pmatrix} \boldsymbol{H_0} \\ \boldsymbol{H_1} \\ \boldsymbol{H_2} \\ \boldsymbol{H_3} \\ \boldsymbol{H_2} \\ \boldsymbol{H_3} \\ \boldsymbol{H_2} \\ \vdots \\ \boldsymbol{H_{25}} \\ \boldsymbol{H_{26}} \\ \boldsymbol{H_{25}} \\ \boldsymbol{H_{26}} \\ \boldsymbol{H_{27}} \\ \boldsymbol{H_{28}} \\ \boldsymbol{H_{29}} \\ \boldsymbol{H_{30}} \\ \boldsymbol{H_{31}} \end{pmatrix} \boldsymbol{P}_{\boldsymbol{1}}^{\boldsymbol{f}} + \boldsymbol{P}_{\boldsymbol{2}}^{\boldsymbol{g}\mathsf{T}} \begin{pmatrix} \boldsymbol{H_0} \\ \boldsymbol{H_1} \\ \boldsymbol{H_2} \\ \boldsymbol{H_3} \\ \boldsymbol{H_2} \\ \boldsymbol{H_3} \\ \boldsymbol{H_2} \\ \vdots \\ \boldsymbol{H_{25}} \\ \boldsymbol{H_{26}} \\ \boldsymbol{H_{25}} \\ \boldsymbol{H_{26}} \\ \boldsymbol{H_{27}} \\ \boldsymbol{H_{28}} \\ \boldsymbol{H_{29}} \\ \boldsymbol{H_{30}} \\ \boldsymbol{H_{31}} \end{pmatrix} \boldsymbol{P}_{\boldsymbol{2}}^{\boldsymbol{f}} + \ldots + \boldsymbol{P}_{\boldsymbol{7}}^{\boldsymbol{g}\mathsf{T}} \begin{pmatrix} \boldsymbol{H_0} \\ \boldsymbol{H_1} \\ \boldsymbol{H_2} \\ \boldsymbol{H_3} \\ \boldsymbol{H_2} \\ \boldsymbol{H_3} \\ \boldsymbol{H_2} \\ \vdots \\ \boldsymbol{H_{25}} \\ \boldsymbol{H_{26}} \\ \boldsymbol{H_{25}} \\ \boldsymbol{H_{26}} \\ \boldsymbol{H_{27}} \\ \boldsymbol{H_{28}} \\ \boldsymbol{H_{29}} \\ \boldsymbol{H_{30}} \\ \boldsymbol{H_{31}} \end{pmatrix} \boldsymbol{P}_{\boldsymbol{7}}^{\boldsymbol{f}} \quad . \tag{2.26}$$

$\boldsymbol{P}_{\boldsymbol{i}}^{\boldsymbol{f}}$ are $N^{\boldsymbol{f}} \times N^{\boldsymbol{f}}$ square permutation matrices that rotate or reflect the image. The first summand is without such a permutation since it represents the unrotated/reflected image. $\boldsymbol{P}_{\boldsymbol{i}}^{\boldsymbol{g}}$ are $N^{\boldsymbol{g}} \times N^{\boldsymbol{g}}$ square permutation matrices which perform the corresponding operation in sinogram space. Accordingly, $\boldsymbol{P}_{\boldsymbol{i}}^{\boldsymbol{g}\mathsf{T}}$ performs a back-rotation/back-reflection in order to correct for the previous rotation/reflection. $\boldsymbol{H_0} \ldots \boldsymbol{H_{31}}$ are the submatrices (for each sinogram type one). The exact ordering depends on the Michelogram. Note that each of these matrices is only defined for one eigth of the image voxel grid, since the rest is then deduced by the lateral rotations/reflections.

In conclusion, we have reduced the system matrix by a factor of about $18,900$. Further reduction by factor of four would be possible by using an eight bit storage for each system matrix element and by using an even tighter encoding instead of the CSR format [159, 161]. For our purposes, a one gigabyte matrix fitted easily in our computers memory and thus we did not make any further compression efforts.

Another interesting and both memory and computationally efficient storage scheme are polar voxels [142, 5, 30, 179]. Polar voxels fit perfectly in the circular structure of the PET ring and thus give rise to much more symmetries than Cartesian voxels do.

2.4. Image reconstruction

The goal of image reconstruction is to estimate the image vector $\boldsymbol{f}$ given the measured number of counts $\boldsymbol{g}$ for the detector pairs. Image reconstruction is an inverse problem, and image reconstruction algorithms attempt to invert the given forward model in order to estimate the image.

A major classification is made between analytical and statistical image reconstruction methods. The former have their origin in CT reconstruction and are still used in PET.

However, due to the statistical nature of PET, the latter gain over their analytical counterparts with respect to the reconstructed image quality. Many physical degradation phenomena which can not be modeled in an analytical method are considered in statistical methods. On the downside, statistical methods are much more computationally intense. Nevertheless, due to the advance in computation speed and also thanks to the ordered subset approach [91] this drawback is alleviated and it seems that the statistical methods have gained over analytical methods even in clinical practice. We will limit our discussion here to statistical reconstruction methods.

There are two components of a statistical reconstruction algorithm which can be discussed separately [200]: the first component is a criterion for an estimate of the image. The second component is the algorithm with which a solution to the specified criterion is found. In simple terms, the criterion asks the question *what* goal should be aimed at and the algorithm is the answer *how* this goal should be achieved.

2.4.1. Statistical Models

We briefly discuss the most common statistical models used for the probability of detecting a number of annihilation events in a certain detector pair.

2.4.1.1. Poisson Model

Both the number of annihilation events and the number of detected photon pairs obey a Poisson distribution. The probability of detecting a certain number of counts $[\boldsymbol{g}]_a$ at a detector pair a is therefore given by

$$P([\boldsymbol{g}]_a|\boldsymbol{f}) = \frac{e^{-[\hat{\boldsymbol{g}}]_a}[\hat{\boldsymbol{g}}]_a^{[\boldsymbol{g}]_a}}{[\boldsymbol{g}]_a!} \quad . \tag{2.27}$$

This number of detected counts is usually considered as statistically independent for all detector pairs. Apart from the detector dead-time, the Poisson model is a reasonable approximation to reality and thus is widely accepted and used. As mentioned earlier, the Poisson assumption is only valid if the data are not pre-corrected by scatter and/or randoms subtraction. If this is the case, then different models like the shifted-Poisson model [202, 203] should be used.

2.4.1.2. Gaussian Model

If the number of counts $[\boldsymbol{g}]_a$ is higher than 20, it has been shown that a Gaussian distribution

$$P([\boldsymbol{g}]_a|\boldsymbol{f}) = ke^{-\frac{1}{2}\frac{([\boldsymbol{g}]_a-[\hat{\boldsymbol{g}}]_a)^2}{[\hat{\boldsymbol{g}}]_a}} \tag{2.28}$$

is a sufficient approximation to the Poisson model [23]. In the Gaussian model there exists a probability for a negative number of events, which has to be accounted for in a reconstruction method using this model.

2.4.2. Reconstruction Criteria

As mentioned, the reconstruction criterion defines the goal to be aimed at by the reconstruction algorithm. We will examine three typical reconstruction criteria.

2.4.2.1. Maximum A Posteriori Estimation

The maximum a posteriori estimate

$$\underset{\boldsymbol{f}}{\operatorname{argmax}}\, P(\boldsymbol{f}|\boldsymbol{g}) \ , \tag{2.29}$$

is the most probable image given the measured data. The probability function $P(\boldsymbol{f}|\boldsymbol{g})$ can be expanded to

$$P(\boldsymbol{f}|\boldsymbol{g}) = \frac{P(\boldsymbol{g}|\boldsymbol{f})P(\boldsymbol{f})}{P(\boldsymbol{g})} \tag{2.30}$$

according to Bayes' theorem. $P(\boldsymbol{g}|\boldsymbol{f})$ is usually assumed to be Poisson distributed (as described in section 2.4.1.1). Since $P(\boldsymbol{g})$ is independent of $\boldsymbol{f}$ it does not play a role in the optimization and can be discarded. $P(\boldsymbol{f})$ is the prior probability of the image $\boldsymbol{f}$. There exists a vast amount of research covering different choices for $P(\boldsymbol{f})$. They can be broadly classified into two categories: anatomical priors and smoothness priors.

Anatomical priors define the prior probability based on specific knowledge of what is expected to be measured [70, 118, 39, 151]. This knowledge may be gathered e. g. from an emission atlas or anatomical imaging modalities such as MRI or CT. A major problem with an emission atlas is that patient specific features may be suppressed. This is especially dangerous since these features may be the most important elements for diagnosis. For example, suppose we want to detect perfusion defects in cardiac imaging, and the atlas used in order to generate an image prior does not contain a similar image to the one we expect to measure. Then, the perfusion defect in the image estimate might be less pronounced or even not visible at all. Priors based on MRI or CT do not have this problem since they can be obtained for each patient individually. However, since they are no emission tomography (ET) images, they do not provide the same features as a such, and a prior probability defined on MRI or CT therefore is often misleading.

For such reasons, less specific priors which e. g. encourage smooth images are preferred. Such priors are generic and thus the same prior can be used for different anatomies like brain or whole body PET. Gibbs priors [65]

$$P(\boldsymbol{f}) = \frac{1}{Z} e^{-\beta U(\boldsymbol{f})} \tag{2.31}$$

are a popular choice for such a less specific prior. Z is a normalization constant, β a parameter which defines the sharpness of the peak around the distribution's maximum, U is called an energy function. The energy function is a weighted sum of potential functions, which are functions defined on so-called *cliques*. A clique is a set of voxels which usually are neighbors of a certain central voxel. The outer sum of the potential function goes over all voxels which are considered as central voxels in the inner sum. In the inner

sum is a sum over the potential functions which works on a clique with respect to the current central voxel. Many sorts of potential functions and cliques have been investigated [123, 125, 109, 110, 111, 121, 122]. The potential functions are usually bowl shaped. They are zero for cliques which contain equal intensity voxels, and increase with increasing intensity difference between the voxels. Noisy images, which obey a high variation of voxel intensity within a local neighborhood, are therefore given a high potential which results in a low probability. The kind and degree of smoothness is influenced by the type of Gibbs prior employed. Since the parameter β has a major influence on the degree of smoothness and therefore the resulting image, it is of big importance to choose it wisely. The difficulty in choosing β is a considerable disadvantage of Gibbs priors.

2.4.2.2. Maximum Likelihood Estimation

If a uniform prior distribution on $\boldsymbol{f}$ is assumed, then we arrive at

$$\underset{f}{\operatorname{argmax}}\, P(\boldsymbol{f}|\boldsymbol{g}) = \underset{f}{\operatorname{argmax}}\, P(\boldsymbol{g}|\boldsymbol{f})P(\boldsymbol{f}) = \underset{f}{\operatorname{argmax}}\, P(\boldsymbol{g}|\boldsymbol{f}) \tag{2.32}$$

which is the maximum likelihood estimate (MLE). $P(\boldsymbol{g}|\boldsymbol{f})$ can be interpreted as the probability of measuring the data given a certain image estimate. The ML estimate is then the image estimate which most likely lead to the given measurement.

A nice property of an ML estimate is that it is unbiased. Furthermore, no asymptotically unbiased estimator has lower asymptotic mean squared error than the MLE. Unfortunately this comes at a price: its variance is relatively high, which means that the reconstructed images are rather noisy. The noise in the measured data is directly reflected in the reconstructed image. In order to deal with this, regularization of the image is employed (see section 2.4.3.4).

2.4.2.3. Least-Squares and Weighted Least-Squares

The least-squares criterion is defined as

$$\underset{f}{\operatorname{argmin}} \|\hat{\boldsymbol{g}} - \boldsymbol{g}\|^2 \ , \tag{2.33}$$

and thus seeks the image $\boldsymbol{f}$ which leads to an estimation vector $\hat{\boldsymbol{g}} = \boldsymbol{H}\boldsymbol{f} + \boldsymbol{r} + \boldsymbol{s}$ which is closest to the measured data $\boldsymbol{g}$ in a least-squares sense. In theory there is an analytical solution to the least-squares criterion using the pseudo-inverse of the system matrix $\boldsymbol{H}$. In practice, the pseudo-inverse of $\boldsymbol{H}$ would not be a sparse matrix anymore and thus is intractably huge.

Usually a weighted least-squares criterion is used instead of the simple least-squares criterion:

$$\underset{f}{\operatorname{argmin}} \|(\hat{\boldsymbol{g}} - \boldsymbol{g})^{\mathsf{T}}\boldsymbol{W}\hat{\boldsymbol{g}} - \boldsymbol{g}\|^2 \ , \tag{2.34}$$

where $\boldsymbol{W}$ is a diagonal weighting matrix. Typically the weight is the reciprocal of the variance of the respective detector pair (the variance is the same as the mean value for Poisson distributions and can therefore be approximated by $\hat{\boldsymbol{g}}$).

It is well known that weighted least-squares estimation is equivalent to maximum likelihood estimation assuming a Gaussian distribution. Accordingly, a least-squares approach should only be employed in a not too low statistics scenario. Weighted least-squares methods were among the first algorithms to be applied to emission tomography [92].

2.4.3. Reconstruction Algorithms

A huge number of reconstruction methods has been presented for PET. The most successful one is probably the maximum likelihood expectation-maximization method and its variations. Since also the motion compensation method which we present in this thesis is based on a maximum likelihood estimate, we will limit our discussion to the ML-EM algorithm. The ML-EM algorithm as it is applied to PET has been known in astronomy as the Richardson-Lucy algorithm [173, 128]. Shepp and Vardi [187] and Lange and Carson [116] recognized it as a specialization of the general expectation maximization algorithm by Dempster *et al.* [53] and applied it to PET. The knowledge that the Richardson-Lucy algorithm is in fact an instance of the EM algorithm opened the door to a variety of proofs which were already known from the general EM algorithm.

2.4.3.1. ML-EM

In the following, we show a derivation of the ML-EM algorithm which is different to the classical derivation of Shepp and Vardi. It leads to the very same iterative formula but does not provide the proof that this algorithm is actually an instance of the EM algorithm. Our goal is to present the tools which are later needed when a motion incorporated maximum likelihood method is derived. For the detailed proof, the reader is referred to [200] where an excellent explanation is given.

2.4.3.1.1. The Likelihood Function As discussed earlier, the number of measured events in a detector pair follows a Poisson distribution [187, 195]

$$P([\boldsymbol{g}]_a|\boldsymbol{f}) = \frac{e^{-[\hat{\boldsymbol{g}}]_a}[\hat{\boldsymbol{g}}]_a^{[\boldsymbol{g}]_a}}{[\boldsymbol{g}]_a!} \quad , \tag{2.35}$$

where $[\hat{\boldsymbol{g}}]_a$ is the expected number of counts in LOR a. The expected number of counts is estimated by projecting the image f to the measurement space using the system matrix:

$$[\hat{\boldsymbol{g}}]_a = \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{ai}[\boldsymbol{f}]_i \quad . \tag{2.36}$$

The probability of obtaining a complete measurement vector $\boldsymbol{g}$ which comprises all possible detector pairs is given by

$$P(\boldsymbol{g}|\boldsymbol{f}) = \prod_{a=1}^{N^{\boldsymbol{g}}} P([\boldsymbol{g}]_a|\boldsymbol{f}) \quad . \tag{2.37}$$

We define a likelihood function as $L(\boldsymbol{f}) := P(\boldsymbol{g}|\boldsymbol{f})$, and the corresponding log-likelihood (which is used since it is easier to handle mathematically) is

$$l(\boldsymbol{f}) := \log(L(\boldsymbol{f})) = \sum_{a=1}^{N^{\boldsymbol{g}}} -[\hat{\boldsymbol{g}}]_a + [\boldsymbol{g}]_a \log([\hat{\boldsymbol{g}}]_a) - \log[\boldsymbol{g}]_a! \ . \tag{2.38}$$

Expanding this equation using (2.36) leads to

$$l(\boldsymbol{f}) = \sum_{a=1}^{N^{\boldsymbol{g}}} - \left(\sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{ai}[\boldsymbol{f}]_i \right) + [\boldsymbol{g}]_a \log \left\{ \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{ai}[\boldsymbol{f}]_i \right\} - \log[\boldsymbol{g}]_a! \ . \tag{2.39}$$

2.4.3.1.2. Maximization of the Likelihood Function We seek the tracer density distribution $\boldsymbol{f}^\star$ with $\boldsymbol{f}^\star >= \boldsymbol{0}$ for which $l(\boldsymbol{f}^\star)$ is maximal. If $\boldsymbol{f}^\star >= \boldsymbol{0}$ and $l(\boldsymbol{f}^\star)$ is maximal, then there exist constants $\boldsymbol{\mu}$ (the so-called KKT multipliers, here written as a vector) such that the Karush-Kuhn-Tucker (KKT) conditions are fulfilled:

$$\nabla_{\boldsymbol{f}} l(\boldsymbol{f}^\star) + \sum_i [\boldsymbol{\mu}]_i \nabla_{\boldsymbol{f}} [\boldsymbol{c}]_i(\boldsymbol{f}^\star) = \boldsymbol{0} \tag{2.40}$$

$$[\boldsymbol{\mu}]_i \geq \boldsymbol{0} \ \forall i \tag{2.41}$$

$$[\boldsymbol{\mu}]_i [\boldsymbol{c}]_i(\boldsymbol{f}^\star) = \boldsymbol{0} \ \forall i \ . \tag{2.42}$$

The positivity constraints are modeled by a function $\boldsymbol{c}$ with $[\boldsymbol{c}]_i(\boldsymbol{f}) := [\boldsymbol{f}]_i$. The term $\sum_i [\boldsymbol{\mu}]_i \nabla_{\boldsymbol{f}} [\boldsymbol{c}]_i(\boldsymbol{f})$ can be simplified:

$$\sum_i [\boldsymbol{\mu}]_i \nabla_{\boldsymbol{f}} [\boldsymbol{c}]_i(\boldsymbol{f}^\star) = \sum_i [\boldsymbol{\mu}]_i \nabla_{\boldsymbol{f}} [\boldsymbol{f}]_i^\star = \boldsymbol{\mu} \ . \tag{2.43}$$

Then, equation (2.40) rewrites to

$$\nabla_{\boldsymbol{f}} l(\boldsymbol{f}^\star) = -\boldsymbol{\mu} \ . \tag{2.44}$$

With (2.44), the KKT conditions can be simplified to the following two equations:

$$-\nabla_{\boldsymbol{f}} l(\boldsymbol{f}^\star) \geq \boldsymbol{0} \tag{2.45}$$

$$-\nabla_{\boldsymbol{f}} l(\boldsymbol{f}^\star) \boldsymbol{f} = \boldsymbol{0} \ . \tag{2.46}$$

In the following, we will use the KKT condition from equation (2.46) in order to define the image update equation. Note that equation (2.46) can be written as

$$\frac{\partial}{\partial [\boldsymbol{f}]_j} l(\boldsymbol{f}^\star)[\boldsymbol{f}]_j = 0 \qquad \forall j \ . \tag{2.47}$$

By calculating the derivative, we arrive at

$$[\boldsymbol{f}]_j \sum_{a=1}^{N^{\boldsymbol{g}}} -[\boldsymbol{H}]_{aj} + [\boldsymbol{g}]_a \frac{[\boldsymbol{H}]_{aj}}{\sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{ai}[\boldsymbol{f}]_i} = 0 \ . \tag{2.48}$$

Breaking the sum and shifting one term to the other side leads to

$$[\boldsymbol{f}]_j \sum_{a=1}^{N^{\boldsymbol{g}}} [\boldsymbol{g}]_a \frac{[\boldsymbol{H}]_{aj}}{\sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{ai}[\boldsymbol{f}]_i} = [\boldsymbol{f}]_j \sum_{a=1}^{N^{\boldsymbol{g}}} [\boldsymbol{H}]_{aj} \ . \tag{2.49}$$

Dividing the whole equation by the sensitivity term $\sum_{a=1}^{N^{\boldsymbol{g}}} [\boldsymbol{H}]_{aj}$ leads to the fixed point equation

$$[\boldsymbol{f}]_j = [\boldsymbol{f}]_j \frac{\sum_{a=1}^{N^{\boldsymbol{g}}} [\boldsymbol{H}]_{aj} \frac{[\boldsymbol{g}]_a}{\sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{ai}[\boldsymbol{f}]_i}}{\sum_{a=1}^{N^{\boldsymbol{g}}} [\boldsymbol{H}]_{aj}} \ . \tag{2.50}$$

Now, we arrive at the the ML-EM algorithm by interpreting the left $\boldsymbol{f}$ as the updated image $\boldsymbol{f}^{\text{new}}$, and the other $\boldsymbol{f}$'s as the current image estimate:

$$[\boldsymbol{f}]_j^{\text{new}} = [\boldsymbol{f}]_j \frac{\sum_{a=1}^{N^{\boldsymbol{g}}} [\boldsymbol{H}]_{aj} \frac{[\boldsymbol{g}]_a}{\sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{ai}[\boldsymbol{f}]_i}}{\sum_{a=1}^{N^{\boldsymbol{g}}} [\boldsymbol{H}]_{aj}} \ . \tag{2.51}$$

The algorithm converges to the maximum likelihood estimate. This was proved by Shepp and Vardi [187] by deriving the algorithm as an instance of the expectation-maximization algorithm [53].

2.4.3.2. List-mode ML-EM

Considering that histogram mode acquisitions do not contain any time information (that is, *when* a certain event has happened), and considering further that motion is a time dependent phenomena, it seems clear that histogram mode acquisitions are not suitable for motion compensation algorithms. Since motion compensation algorithms have been proposed based on the list-mode ML-EM algorithm, we discuss it here in a static context. The extensions necessary for motion compensation are reviewed in the next chapter.

2.4.3.2.1. Classical Derivation In the following, we will derive the list-mode maximum likelihood expectation-maximization (list-mode ML-EM) algorithm. Our model is similar to the one used by Parra and Barrett [153]. In the derivation of the algorithm however, we will use the KKT conditions instead of the EM algorithm (which is used by Parra and Barrett). Our motivation for using the KKT conditions is firstly that they facilitate the derivation of the list-mode ML-EM algorithm and secondly to show an alternative derivation. Note that the KKT conditions do not include a proof of convergence as the EM algorithm does implicitly (since any instance of the EM algorithm converges).

The probability that a detected event originates in voxel i is

$$P(i|\boldsymbol{f}) = \frac{[\boldsymbol{f}]_i [\boldsymbol{s}]_i}{\sum_{j=1}^{N^{\boldsymbol{f}}} [\boldsymbol{f}]_j [\boldsymbol{s}]_i} \ , \tag{2.52}$$

with $[\boldsymbol{s}]_i$ being the sensitivity of voxel i, that is, the probability that an emission at voxel i is detected at all. The probability that a detected event (originating from any voxel) is detected in LOR a is given by

$$P(a|\boldsymbol{f}) = \sum_{i=1}^{N^{\boldsymbol{f}}} P(a|i)P(i|\boldsymbol{f}) \ , \tag{2.53}$$

where $P(a|i)$ is the probability that a *detected* emission in voxel i is detected in LOR a. The probability of measuring a series of events is given by the product of the probability of measuring individual events:

$$P(a_1 \dots a_N|\boldsymbol{f}) = \prod_{k=1}^{N} P(a_k|\boldsymbol{f}) \ . \tag{2.54}$$

In fact, the number N of measured events is also a random variable and depends on the acquisition time T. Parra and Barrett refer to the case where T is fixed the preset-time case:

$$P(a_1 \dots a_N, N|\boldsymbol{f}, T) = P(a_1 \dots a_N|\boldsymbol{f})P(N|T, \boldsymbol{f})) \ . \tag{2.55}$$

The number of measured counts n is Poisson distributed

$$P(N|T, \boldsymbol{f}) = \frac{e^{-\lambda}\lambda^N}{N!} \ , \tag{2.56}$$

with the expected value being

$$\lambda = \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{s}]_i[\boldsymbol{f}]_i \ . \tag{2.57}$$

We will seek the image $\boldsymbol{f}$ which maximizes $P(a_1 \dots a_N, N|\boldsymbol{f}, T)$. To this end we start with the log-likelihood (which is easier to handle mathematically):

$$l(\boldsymbol{f}) := \log \{P(a_1 \dots a_N, N|\boldsymbol{f}, T)\} \tag{2.58}$$

The logarithm of products converts to sums of logarithms:

$$l(\boldsymbol{f}) = \sum_{k=1}^{N} \log P(a_k|\boldsymbol{f}) + \log P(N|T, \boldsymbol{f}) \tag{2.59}$$

Further application of this rule leads to

$$l(\boldsymbol{f}) = \sum_{k=1}^{N} \log \left\{ \sum_{i=1}^{N^{\boldsymbol{f}}} P(a_k|i)P(i|\boldsymbol{f}) \right\} - \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{s}]_i[\boldsymbol{f}]_i + N \log \left\{ \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{s}]_i[\boldsymbol{f}]_i \right\} - \log N! \ . \tag{2.60}$$

Replacing $P(i|\boldsymbol{f})$ leads to

$$l(\boldsymbol{f}) = \sum_{k=1}^{N} \left\{ \log \left\{ \frac{1}{\sum_{j=1}^{N^{\boldsymbol{f}}} [\boldsymbol{f}]_j [\boldsymbol{s}]_j} \sum_{i=1}^{N^{\boldsymbol{f}}} P(a_k|i)[\boldsymbol{f}]_i [\boldsymbol{s}]_i \right\} \right\} \tag{2.61}$$

$$- \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{s}]_i [\boldsymbol{f}]_i + N \log \left\{ \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{s}]_i [\boldsymbol{f}]_i \right\} - \log N! \ . \tag{2.62}$$

The fraction $\frac{1}{\sum_{j=1}^{N^{\boldsymbol{f}}} [\boldsymbol{f}]_j [\boldsymbol{s}]_j}$ can be extracted from the logarithm and we arrive at

$$l(\boldsymbol{f}) = \sum_{k=1}^{N} \left\{ \log \left\{ \sum_{i=1}^{N^{\boldsymbol{f}}} P(a_k|i)[\boldsymbol{f}]_i [\boldsymbol{s}]_i \right\} - \log \left\{ \sum_{j=1}^{N^{\boldsymbol{f}}} [\boldsymbol{f}]_j [\boldsymbol{s}]_j \right\} \right\}$$

$$- \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{s}]_i [\boldsymbol{f}]_i + N \log \left\{ \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{s}]_i [\boldsymbol{f}]_i \right\} - \log N! \ . \tag{2.63}$$

The logarithm $-\log \left\{ \sum_{j=1}^{N^{\boldsymbol{f}}} [\boldsymbol{f}]_j [\boldsymbol{s}]_j \right\}$ can be extracted from the sum and then cancels out:

$$l(\boldsymbol{f}) = \sum_{k=1}^{N} \left\{ \log \sum_{i=1}^{N^{\boldsymbol{f}}} P(a_k|i)[\boldsymbol{f}]_i [\boldsymbol{s}]_i \right\} - \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{s}]_i [\boldsymbol{f}]_i - \log N! \tag{2.64}$$

At this point, we will reuse equation (2.46) which was derived above from the KKT conditions in order to derive the list-mode ML-EM algorithm. The idea is exactly the same, we will just use the list-mode log-likelihood function instead of the non-list-mode one from above: calculating the derivative equation (2.46) results in

$$[\boldsymbol{f}]_j \sum_{k=1}^{N} \left\{ \frac{P(a_k|j)[\boldsymbol{s}]_j}{\sum_{i=1}^{N^{\boldsymbol{f}}} P(a_k|i)[\boldsymbol{f}]_i [\boldsymbol{s}]_i} \right\} - [\boldsymbol{f}]_j [\boldsymbol{s}]_j = 0 \ . \tag{2.65}$$

This equation gives rise to the ML-EM algorithm for list mode data:

$$[\boldsymbol{f}]_j^{\text{new}} = [\boldsymbol{f}]_j \sum_{k=1}^{N} \frac{P(a_k|j)}{\sum_{i=1}^{N^{\boldsymbol{f}}} P(a_k|i)[\boldsymbol{f}]_i [\boldsymbol{s}]_i} \ . \tag{2.66}$$

In order to stay within our familiar notation, instead of working with the probability of detecting a detected coincidence event at a certain LOR we will work with the probability of detecting a coincidence event (be it detected finally or not) at a certain LOR:

$$[\boldsymbol{H}]_{a_k i} = P(a_k|i)[\boldsymbol{s}]_i \ . \tag{2.67}$$

The list-mode ML-EM algorithm then rewrites to

$$[\boldsymbol{f}]_j^{\text{new}} = [\boldsymbol{f}]_j \frac{\sum_{k=1}^{N} \frac{[\boldsymbol{H}]_{a_k i}}{\sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{a_k i} [\boldsymbol{f}]_i}}{[\boldsymbol{s}]_i} \ , \tag{2.68}$$

with $[\boldsymbol{s}]_i = \sum_a [\boldsymbol{H}]_{ai}$.

2.4.3.2.2. Alternative Derivation We start from the non-list-mode log-likelihood function from equation (2.38)

$$l(\boldsymbol{f}) = \sum_{a=1}^{N^{\boldsymbol{g}}} -[\hat{\boldsymbol{g}}]_a + [\boldsymbol{g}]_a \log([\hat{\boldsymbol{g}}]_a) - \log[\boldsymbol{g}]_a! \ .$$

The sum can be split which leads to

$$l(\boldsymbol{f}) = \sum_{a=1}^{N^{\boldsymbol{g}}} [\boldsymbol{g}]_a \log([\hat{\boldsymbol{g}}]_a) - \sum_{a=1}^{N^{\boldsymbol{g}}} [\hat{\boldsymbol{g}}]_a - \sum_{a=1}^{N^{\boldsymbol{g}}} \log[\boldsymbol{g}]_a! \ . \tag{2.69}$$

Now, Huesman *et al.* propose a simple list-mode conversion of this likelihood function by just rewriting the first sum as a sum over all individual events [93]:

$$l(\boldsymbol{f}) = \sum_{k=1}^{N} \log([\hat{\boldsymbol{g}}]_{a_k}) - \sum_{a=1}^{N^{\boldsymbol{g}}} [\hat{\boldsymbol{g}}]_a - \sum_{a=1}^{N^{\boldsymbol{g}}} \log[\boldsymbol{g}]_a! \ , \tag{2.70}$$

where a_k is the LOR corresponding to event k. Note that most of the LORs are repeatedly visited in the sum.

Furthermore, the second sum can be rewritten: since $\sum_{a=1}^{N^{\boldsymbol{g}}} [\hat{\boldsymbol{g}}]_a = \sum_{a=1}^{N^{\boldsymbol{g}}} \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{ai} [\boldsymbol{f}]_i$, it is more efficient to calculate $\sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{f}]_i \sum_{a=1}^{N^{\boldsymbol{g}}} [\boldsymbol{H}]_{ai}$, and we arrive at

$$l(\boldsymbol{f}) = \sum_{k=1}^{K} \log([\hat{\boldsymbol{g}}]_{a_k}) - \sum_{j=1}^{N^{\boldsymbol{f}}} [\boldsymbol{s}]_j [\boldsymbol{f}]_j - \sum_{a=1}^{N^{\boldsymbol{g}}} \log[\boldsymbol{g}]_a! \ , \tag{2.71}$$

with $[\boldsymbol{s}]_j = \sum_{a=1}^{A} [\boldsymbol{H}]_{aj}$ being the sensitivity of voxel j, that is, the probability that an emission from voxel j will be detected at all (by any detector).

Equation (2.71) is almost exactly equal to (2.64). The only difference is the constant factorial term. For the rest of the derivation however, this does not make a difference since this constant term falls away when calculating the derivative of the likelihood function with respect to $[\boldsymbol{f}]_j$. Note that the system matrix $\boldsymbol{H}$ includes the sensitivity: $[\boldsymbol{H}]_{ai} = P(a|j)[\boldsymbol{s}]_j$. Following the same steps as above in order to maximize this log-likelihood function, we arrive at the list-mode ML-EM algorithm from equation (2.68).

2.4.3.3. OS-EM

The ordered subsets expectation maximization (OS-EM) algorithm was presented by Hudson and Larkin in 1994 [91]. It is probably the most important enhancement of the ML-EM algorithm since it allows for a speed up of approximately factor ten or more, depending on the number of subsets. This speed up alleviates the most severe drawbacks the ML-EM algorithm had to this moment and accelerated its way to hospitals.

In OS-EM, the measured data is partitioned into a specific number of subsets. The number of subsets N is called the OS-EM *level*. One OS-EM iteration consists of N subiterations. Each subiteration is an ML-EM iteration on one of the subsets. Accordingly, after an OS-EM iteration, the complete data is traversed once. Hudson and Larkin made

experiments for up to 32 subsets, and even for the highest OS-EM level the reconstructed image is very similar to a 32 iteration ML-EM reconstruction. This results in an approximate speed up of factor N. OS-EM does not converge to the maximum likelihood estimate, and it is not yet clear in which circumstances OS-EM converges at all. These theoretical limitations have not prevented the enormous practical impact of OS-EM.

2.4.3.4. Regularization

Regularization is the process of encouraging physically meaningful solutions and can be seen as a measure to incorporate knowledge about the solution (e. g. smoothness of medical images). Ideally, this is done in the reconstruction criterion itself, as discussed for the MAP criterion. For the ML-EM algorithm (as said in the beginning of section 2.4.3, we limit our discussion of reconstruction algorithms to the ML-EM algorithm), this is not possible. There are three relatively simple remedies for this problem which are employed by many researchers:

1. Stopping the ML-EM algorithm prematurely before convergence.
2. Post-smoothing of the reconstructed image after running ML-EM to convergence.
3. A hybrid approach between the first two.

With respect to the premature stopping of the ML-EM: the earlier the ML-EM algorithm is stopped, the smoother is the reconstructed image. The key question is: when exactly should the ML-EM algorithm be stopped? In the literature, this question is discussed as the stopping criterion or stopping rule. It basically boils down to the definition of a certain figure of merit (FOM), which is a measure for the quality of a reconstructed image. If ground truth data is available (which is the case in a simulation environment), it is relatively easy to define a meaningful FOM: one could use e. g. the correlation coefficient of a reconstructed image with the ground truth images. Having defined a FOM, the ML-EM algorithm could be stopped when the FOM is optimal. A problem with this idea is that it is not easily transferable to the OS-EM algorithm, since one should not stop the OS-EM reconstruction within an iteration, that is, without completing a subset. Also, not all regions may converge at the same number of iterations.

For the post-smoothing approach, the degree of smoothness is defined by the size of the smoothing kernel (usually given by the FWHM of a Gaussian kernel). The optimal FWHM can be also defined by the FOM. This approach works well if the reconstructed image is not extremely noisy. In our experience, when employed on ML image estimates of very low statistics gates, it does not yield good results.

For these reasons, in this thesis, whenever we deal with OS-EM reconstructions in a simulation environment, we use a hybrid approach: we calculate the FOM after each OS-EM iteration. The FOM will increase until a certain amount k of iterations and then decrease for $k+1, k+2, \ldots$ iterations. We save the result after $k+1$ iterations. Then, when it comes to the evaluation, we perform Gaussian post-smoothing of this result in order to find the optimal image.

3. Overview of Motion Compensation

Since motion is a time dependent phenomenon, the linear model of data acquisition has to be adjusted for a dynamic setting. The expected number of counts $\hat{g}(a)$ for an LOR during the acquisition time is now calculated as an integral over the count rate

$$\hat{g}(a) = \int_0^T \hat{g}(a,t)\, dt \ . \tag{3.1}$$

Accordingly, due to the time dependence, we do not deal with the expected number of counts but with the expected number of counts *per time*, which is then called the expected count rate

$$\hat{g}(a,t) = \frac{1}{T} \int_\Omega H(a,\boldsymbol{x},t) f(\boldsymbol{x},t)\, d\boldsymbol{x} \ . \tag{3.2}$$

$H(a,\boldsymbol{x},t)$ is the probability of measuring an emission from voxel $\boldsymbol{x}$ at LOR a at time t. Note that $\boldsymbol{x}$ is the position where the emission actually took place at time t. Also the tracer density f now depends on both space and time.
Figure 3.1 illustrates this model.

In dynamic PET, we will use the following system matrix factorization:

$$H(a,\boldsymbol{x},t) = H_\mathrm{d}(a) H_\mathrm{a}(a,t) H_\mathrm{g}(a,\boldsymbol{x}) \ . \tag{3.3}$$

$H_\mathrm{d}(a)$ corrects for the detector efficiency, $H_\mathrm{a}(a,t)$ is the attenuation factor at time t and $H_\mathrm{g}(a,\boldsymbol{x})$ is the geometric probability of detecting an emission from position $\boldsymbol{x}$ at LOR a. Since working with a factored system model including motion is still a subject of active research, we will limit the discussion to these three factors. Note that $H_\mathrm{g}(a,\boldsymbol{x})$ does not depend on the time in this model. It is simply the probability of detecting an emission from position $\boldsymbol{x}$ at LOR a, at whatever time it took place. Also note that the detector efficiency factor H_d is not time dependent. Since attenuation depends on the moving anatomy, it is clear that when motion is present it is subject to change. For moving anatomy, the static attenuation factor from equation (**??**) has to be modified to

$$H_\mathrm{a}(a,t) = e^{-\int_{L(a)} \mu(\boldsymbol{x},t)\, ds} \ , \tag{3.4}$$

where the only difference is that μ is now a function of time. In case of rigid motion, the attenuation factor does not need to be recalculated every time. Pre-calculated integrals can be used by choosing the attenuation factor from an adequate LOR. We will come back to this in the next section on rigid motion.

The count rate then rewrites as

$$\hat{g}(a,t) = \frac{1}{T} H_\mathrm{d}(a) H_\mathrm{a}(a,t) \int_\Omega H_\mathrm{g}(a,\boldsymbol{x}) f(\boldsymbol{x},t)\, d\boldsymbol{x} \ . \tag{3.5}$$

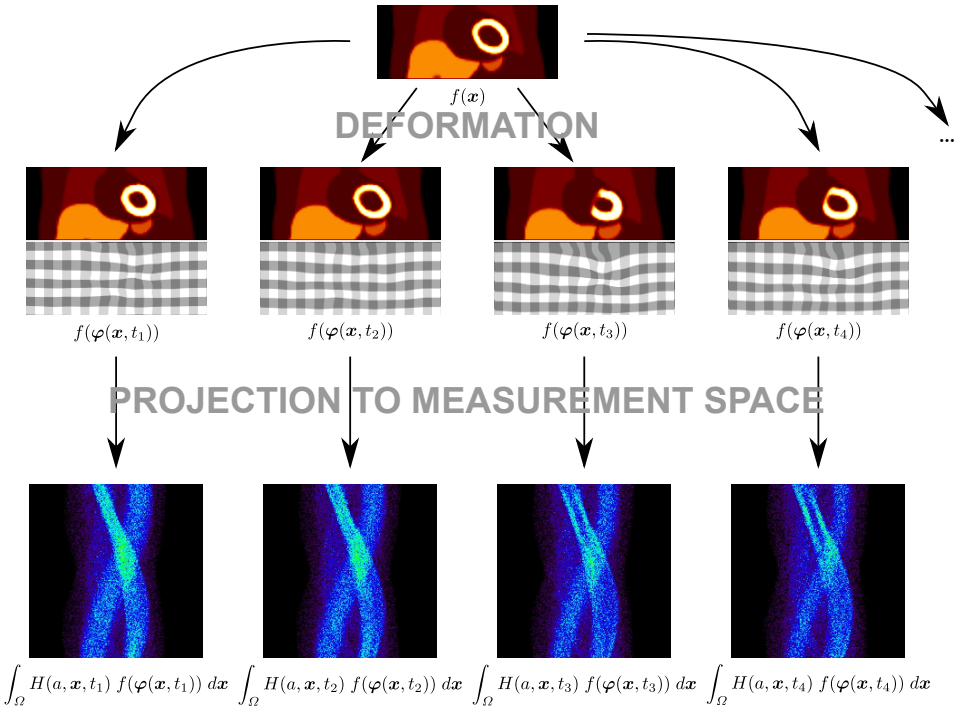


Figure 3.1.: Visualization of the dynamic expected count rate: an image f is transformed to its state at time t by a motion function $\boldsymbol{\varphi}$. The motion function is visualized by means of a checkerboard like grid. The transformed image is then projected to the measurement space according to the system model.

We arrive at the following dynamic acquisition model:

$$\hat{g}(a) = \frac{1}{T} H_\mathrm{d}(a) \int_0^T H_\mathrm{a}(a,t) \int_\Omega H_\mathrm{g}(a,\boldsymbol{x}) f(\boldsymbol{x},t)\, d\boldsymbol{x}\, dt \quad . \tag{3.6}$$

As it stands, the model is extremely general, since $f(\boldsymbol{x}, t)$ could represent anatomical motion (movements of organs or parts of the body) as well as changes of tracer distribution (kinetic tracer modeling). It is beyond the scope of this thesis to discuss the broad field of kinetic tracer modeling in PET, and we will restrict our discussion to anatomical motion. In the model, this restriction can be achieved by replacing $f(\boldsymbol{x}, t)$ with

$$f(\boldsymbol{\varphi}(\boldsymbol{x}, t)) \quad . \tag{3.7}$$

The tracer distribution f now depends on time only indirectly via a motion function $\boldsymbol{\varphi} : \mathbb{R}^3 \times \mathbb{R} \mapsto \mathbb{R}^3$. The motion function maps a position $\boldsymbol{x}$ to a new position $\boldsymbol{\varphi}(\boldsymbol{x}, t)$. Note that this restriction does not allow to model the tracer decay anymore. This is usually not a drawback when gating is employed, since each gate contains data from the whole

acquisition uniformly distributed over time. We arrive at the following model to which we will refer for the rest of this chapter.

$$\hat{g}(a) = \frac{1}{T} H_{\mathrm{d}}(a) \int_0^T H_{\mathrm{a}}(a,t) \int_\Omega H_{\mathrm{g}}(a,\boldsymbol{x}) f(\boldsymbol{\varphi}(\boldsymbol{x},t))\, d\boldsymbol{x}\, dt \ . \tag{3.8}$$

The rest of this chapter is organized as follows: we start by giving an overview of transformations of the previously shown dynamic PET data acquisition model. These transformations are developed in later sections, but for both a reference and an overview it is good to see them beforehand already. Then, we review existing motion compensation methods based on a classification into methods that separate motion from image estimation and methods that jointly estimate image and motion. The former start with estimating the motion either by measuring it with adequate devices or by algorithms on the acquired data. Then, based on this motion estimate, they calculate the motion compensated image. The latter do both image and motion estimation in a joint manner, that is, it is not possible to make a separation between the two steps. Mathematically they are based on a combined objective function that is subject to optimization.

3.1. Model Transformations

Here, we will briefly summarize the results of model transformations that we will develop step by step in the next sections. The equations are derived and explained in detail later, so the only purpose of showing them at the beginning is to show how the motion function $\boldsymbol{\varphi}$ first appears in the image, then in the system matrix and finally in the measured data. Since all of these equations are equivalent, the presented model is the basis for all those motion compensation methods which correspond to one of the transformations. As we will show in our literature review, this applies to many of the motion compensation methods we evaluated.

We start with the dynamic count rate from equation (3.5), now adapted to the anatomical motion model (with the motion function within the image):

$$\hat{g}(a,t) = \frac{1}{T} H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) \int_\Omega H_{\mathrm{g}}(a,\boldsymbol{x}) f(\boldsymbol{\varphi}(\boldsymbol{x},t))\, d\boldsymbol{x} \ . \tag{3.9}$$

In section 3.2.2.2.3 on page 56 we will show that by performing an integral substitution, we yield equation (3.23), where the (inverse) motion function $\boldsymbol{\varphi}$ appears in the system matrix and not in the image anymore:

$$\hat{g}(a,t) = \frac{1}{T} H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) \int_{\boldsymbol{\varphi}^{-1}(\Omega)} |\det(\nabla_{\boldsymbol{x}} \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)|\ H_{\mathrm{g}}(a, \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t))\ f(\boldsymbol{x})\, d\boldsymbol{x} \ .$$

$|\det(\nabla_{\boldsymbol{x}} \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)|$ is the absolute value of the determinant of the Jacobian matrix of the inverse motion function, a factor which comes from the integral substitution. The inverse of the motion function $\boldsymbol{\varphi}$ is meant with respect to $\boldsymbol{x}$, at gate t (that is, the time variable is considered to be constant when calculating the inverse).

We will further show in section 3.2.2.2.3 that the motion information can be theoretically even employed in LOR space (equation (3.24) on page 57):

$$\hat{g}(a,t) = \frac{1}{T} H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) \int_{\boldsymbol{\varphi}^{-1}(\Omega)} |\det(\nabla \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)|\ H_{\mathrm{g}}(\mathcal{L}(a,\boldsymbol{x},t),\boldsymbol{x})\ f(\boldsymbol{x})\ d\boldsymbol{x}$$

Simply speaking, instead of transforming image space, the detectors are moved accordingly. This works - in theory - for nonrigid motion. The detector movement is contained in function $\mathcal{L}$, which defines how an LOR a has to be transformed for an annihilation at a certain position $\boldsymbol{x}$ at time t.

In section 3.2.1.2.2, this principle can be taken even further. By asuming rigid motion, the motion function can be incorporated in the measured data (equation (3.12)):

$$\hat{g}(\mathcal{L}^{-1}(a,t),t) = \frac{1}{T} H_{\mathrm{d}}(\mathcal{L}^{-1}(a,t)) H_{\mathrm{a}}(a) \int_{\Omega} H_{\mathrm{g}}(a,\boldsymbol{x})\ f(\boldsymbol{x})\ d\boldsymbol{x}\ .$$

Equation 3.12 is the base for a mathematical derivation of the rebinning approach. Note that the LOR space operator $\mathcal{L}$ does not depend on the annihilation position for rigid motion.

3.2. Separate Motion Estimation and Image Estimation

Motion compensation methods that separate motion estimation from image estimation have been the first ones to be proposed (starting with the paper of Daube-Witherspoon in 1990), and are still the ones which are most widely employed (maybe due to their simplicity). In any case, from a software engineering point-of-view, their modularity is appealing: one could easily substitute the motion estimation part while maintaining the image estimation part, and vice versa.

Both motion estimation as well as image estimation can be fundamentally different for rigid and nonrigid motion. We therefore subdivide this section accordingly.

3.2.1. Rigid Motion

The only part of the body where the rigid motion assumption is justified is the brain. Some authors also apply it to the heart's respiratory motion, but actually, even if the heart moved completely rigidly due to respiratory motion, this would only be a local rigid movement. In this section, we consider global rigid motion which applies to the whole field-of-view, and as such the brain is the only organ to which the following considerations apply.

3.2.1.1. Motion Estimation

Almost all publications use motion tracking devices in order to measure the motion. There is only one exception (to the best of our knowledge), which will be discussed in the section "Image Registration".

3.2.1.1.1. Tracking Either self-made or commercially available tracking devices are used for motion measurement. Optical motion tracking systems consist of a tracking target and two cameras. The purpose is to obtain the 3D position/orientation of the tracking target relative to a world's coordinate origin. Also magnetical tracking systems are used rarely.

Self-Made Devices The self-made devices are all optical tracking systems which consist of a tracking target and two (usually monochrome) cameras. After extracting the 2D position of the tracking target elements in the image of the cameras, the position and orientation (six degrees of freedom) of the tracking target is calculated by triangulation.

Picard and Thompson use three LEDs as tracking target [155, 156]. The LEDs are fixed on the subject's head such that they can be seen by both cameras: one on the tip of the nose, one between the two eyes and one between an eye and an ear. Goldstein *et al.* use a wire-frame with three miniature lamps [73]. The lamp wire-frame is connected to the top of a tubular structure fitted into the ears and attached by adhesive pads to the forehead. Hu *et al.* place three round dots on the patient's forehead [90]. The tracking system was only tested for a model head, potential movements of the skin are not taken into account.

Picard and Thompson mount the cameras orthogonally such that both face and profile views of the patient's head can be displayed side by side on a monitor. Hu *et al.* mount the cameras at the back-end of the PET tunnel in two different directions. The angle between the principle axis of the cameras is less than 45 degrees.

Commercial Devices Daube-Witherspoon *et al.* use a commercially available magnetic tracking device [49]. The transducer is fixed directly on the object (a cross-hair of F-18 in a 1 mm inner diameter tubing, aligned within a single plane of the scanner). The system was not tested for patient data in this study. Green *et al.* use a commercially available magnetic tracking system [1] [78]. The transducer is attached to the subject's forehead. Also the Polaris optical tracking system is used frequently (see e. g. [64, 63, 11, 193, 62, 84, 167], just to name a few). Again, the tracking target is mounted on the patient's forehead.

3.2.1.1.2. Multiple Acquisition Frames The multiple acquisition frames (MAF) approach was initially proposed by Picard and Thompson in 1997 [156]. The general idea behind the multiple acquisition frames approach is to subdivide the data acquisition into several frames which suffer only from low intra-frame motion. The MAF approach is a motion estimation procedure in the sense that the motion information is only measured for different frames, not continuously for the whole scan.

The assumption that frames with low intra-frame motion can be obtained in practice is underpinned by Green *et al.* [78]. There, a transducer being part of a magnetic tracking system is attached to subject's forehead. Nine virtual point-sources are imagined on a transverse plane through the transducer. The movement of the virtual point-sources is calculated according to the orientation/position measurements from the tracking system. The positions of the point-sources over time can be interpreted as a point cloud

[1] Polhemus Navigation Sciences 3Space Tracker (Model 002)

whose extension is characterized roughly by calculating the full width at half maximum (FWHM) in all three dimensions. Two types of FWHM are calculated: for the first type, the FWHM is calculated for segments of increasing cumulative length: 0 to 5 minutes, 0 to 10 minutes, 0 to 15 minutes etc.. For the second type, the FWHM is calculated for consecutive constant 5 minute segments: 0 to 5 minutes, 5 to 10 minutes, 10 to 15 minutes etc.. As expected, the FWHM for the cumulative FWHM was increasing during the study (from roughly 1 mm to roughly 3 mm, depending on the considered virtual point source). Interestingly, for the consecutive constant segments, the FWHM did not increase with increasing acquisition time. This is an important result which motivates the multiple acquisition frames type methods. Note that these measurements were performed with and without head restraint. The head restraint alleviated the motion, but apart from that the general trend was the same.

Picard and Thompson subdivide the data into frames of low intra-frame motion by controlling the position and orientation with an optical tracking system (as described above). Once a certain similarity criterion between the current position and the initial position of the frame is violated, the current frame is abandoned and either a new frame is created or another similar (in the sense of this similarity criterion) frame is continued.

For completeness, we shortly mention the image estimation procedure (even though we should talk about motion estimation exclusively in this section) that Picard and Thompson apply in their original paper: after data acquisition, each frame is reconstructed individually and corrected for decay. The reconstructed frames are then rotated and translated to the first frame according to its associated initial position and added together (see section 3.2.1.2.1).

Picard and Thompson evaluated their method for a line source and a Hoffman brain phantom [86]. In both cases, discrete motion and continuous motion was simulated. For the discrete motion, the phantoms were put into different positions (three for the line source and seven for the brain phantom), each position corresponding to its own frame. Accordingly, no similarity criterion was needed. In case of continuous motion, a simple one degree of freedom tilting motion was simulated. Only one LED was used for position tracking, and the similarity criterion was based on a distance measure between the current position and the initial position of the active frame.

Fulton *et al.* use a MAF approach with fixed frame intervals [64]. They use both the direct framing capabilities of the scanner (frame duration is two minutes) and a sorting algorithm dividing list-mode data into several frames (frame duration is 30 seconds) after data acquisition (offline). The attenuation map was acquired at the initial head position, and the reconstruction programs were supplied with the aligned attenuation map. Thus, for both attenuation and scatter correction (which depends on the attenuation map) the motion information was taken into account.

3.2.1.1.3. Image Registration The basic idea behind this approach is to register reconstructed frames of data which contain low intra-frame motion [47]. Since [47] is based on a simulation study, no details on how the frames could be generated in practice are known - in fact it seems not trivial to determine the frames without using a MAF approach. If no external tracking is employed, then short time frames could be recon-

structed (they must be really short in order to make sure that there is low intra-frame motion) and co-registered. However, this procedure complicates the registration procedure since it would be based on extremely noisy images. If external motion tracking is employed, then the frames could be defined precisely, but then image registration is not necessary since the motion information is directly available by the tracking system. Maybe these considerations are the cause of lacking interest in such methods in the scientific community.

3.2.1.1.4. PET-MR PET-MR is an exciting and new combination of two known imaging modalities: PET and MRI. Just recently, the first prototype of a simultaneous PET-MR scanner has been shipped to a clinic [126]. At this point we will just shortly review some works in which PET and MRI have been used for motion compensation. A further discussion on PET-MR is done in the future work section 7.5. Catana *et al.* use the MR modality in order to compensate for motion in neurological studies [35]. Catana *et al.* work with the BrainPET prototype, which is a dedicated brain scanner inlay. It works inside the bore of the Siemens MAGNETOM 3-T MRI scanner. Motion is estimated by prospective acquisition correction [191], which consists of echo planar images combined with fast low angle shot magnetic resonance imaging (FLASH). By using rigid image registration techniques, the motion information is extracted from these low quality MRI images. Another motion estimation method implemented by Catana *et al.* is the use of embedded cloverleaf navigators [194]. There, a cloverleaf navigator is inserted after the readout of a 3D FLASH sequence every repetition time (TR). This requires no additional radio frequency pulses and has a minimal impact on the scan duration. A rigid body motion estimate is made for every TR.

3.2.1.2. Image Estimation

3.2.1.2.1. Sum of Reconstructed Frames The simplest way of estimating the motion free image, if frames are available, is to transform them to a reference frame and then add them. It is usually employed in conjunction with the MAF motion estimation method.

3.2.1.2.2. Rebinning The rebinning approach is the first known attempt to tackle motion compensation. It was introduced by Daube-Witherspoon in 1990 [49].

Intuitive Motivation An important assumption of the rebinning method is illustrated in Figure 3.2: for each event, after deformation, the orientation of the LOR corresponding to that event has the same orientation with respect to the anatomy as in case the object had not moved. This is true for both rigid and nonrigid motions (in the illustration, a nonrigid deformation is shown). In the nonrigid case, the orientation is to be interpreted tangentially to the deformation grid at the emission position.

Based on this observation, the basic idea behind the rebinning method is illustrated in Figure 3.3. A coincidence event which is detected by a detector pair D1/D2 is known to lie on a straight line connecting these two detectors. However, it is not known *where* on this line the event originates. In the rigid case we observe the following: if the object had

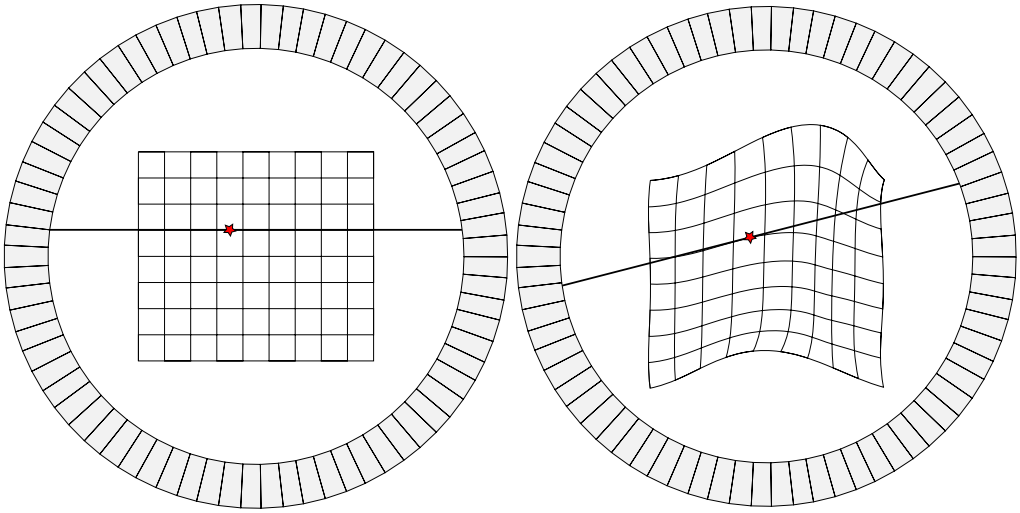


Figure 3.2.: (a) An emission of two back-to-back gammas at a certain position (indicated by the red star) along an LOR at time t_1. (b) The same emission at time t_2: due to the nonrigid deformation, the orientation and position of the LOR has changed. Note that it is still tangential to the deformation grid at the emission position.

not moved, then all the potential emission positions would lie on a straight line again and share the same LOR. Note that this observation is not true for nonrigid motion (refer Figure 3.3b)!

We conclude that, in the rigid case, instead of detecting the emission in the detector pair D1/D2, we would have detected the emission in a detector pair D3/D4 had the object not moved. D3/D4 are found by applying the inverse transformation (which is assumed to be known, usually it is measured by a tracking system) to the measured LOR. In the rebinning method then, for each list-mode event which is detected in detector pair D1/D2, we find the detector pair D3/D4 where the event would have been detected had the object not moved. We then count the event for detector pair D3/D4 instead of D1/D2.

Note that in the nonrigid case, had the object not moved, we would not know where the emission would have been detected since the direction of the back-to-back gammas depends on the actual emission position (which is unknown). Therefore, the rebinning procedure cannot be applied for nonrigid motions.

Mathematical Derivation The rebinning procedure can be motivated mathematically. For that, we have to use some of the results from one of the next sections on nonrigid motion compensation.

We start by showing how to get from equation (3.24), which incorporates motion in the system matrix LOR space, to equation (3.12), which incorporates motion in the measurement space. Recall equation (3.24):

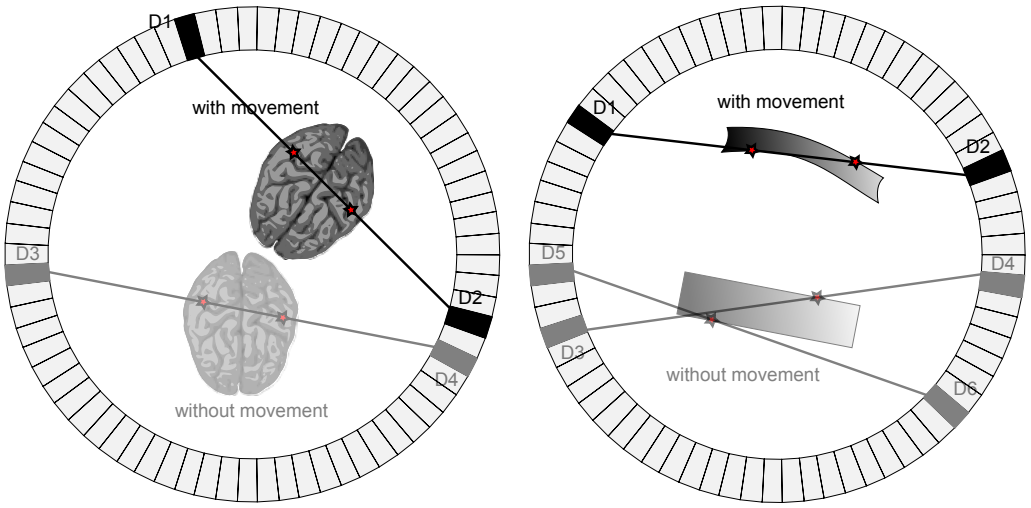


Figure 3.3.: (a) A detection at detector pair D1/D2 could originate from any position on the line connecting the two detectors. Two potential emission positions are shown by a red star. For rigid motion, had the object not moved: no matter which of the two potential emission positions is the correct one, it would be detected on a common LOR (at detector pair D3/D4). (b) The same scenario for nonrigid motion. Again, a detection at detector pair D1/D2 could originate from any position on the line connecting the two detectors (e. g. at the indicated stars). However, different from the rigid case, the emission from the two potential positions would not lie on a common LOR, had the object not moved. Therefore, the rebinning approach does not work for nonrigid motions.

$$\hat{g}(a,t) = \frac{1}{T} H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) \int_{\boldsymbol{\varphi}^{-1}(\Omega)} |\det(\nabla \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)|\; H_{\mathrm{g}}(\mathcal{L}(a,\boldsymbol{x},t),\boldsymbol{x})\; f(\boldsymbol{x})\; d\boldsymbol{x}$$

Now, for rigid motion, $\mathcal{L}$ is actually *independent* of the spatial position. Furthermore, the determinant of the Jacobian of a rigid transformation (and its inverse, which is also a rigid transformation) is equal to one. Also, we can write $\int_{\Omega}$ instead of $\int_{\boldsymbol{\varphi}^{-1}(\Omega)}$ since the image function is zero outside its support anyway. We arrive at

$$\hat{g}(a,t) = \frac{1}{T} H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) \int_{\Omega} H_{\mathrm{g}}(\mathcal{L}(a,t),\boldsymbol{x})\; f(\boldsymbol{x})\; d\boldsymbol{x} \quad . \tag{3.10}$$

In a rigid setting, the dynamic attenuation factor $H_{\mathrm{a}}(a,t)$ is equal to the static attenuation factor $H_{\mathrm{a}}(\mathcal{L}(\mathrm{a},t))$, which leads to

$$\hat{g}(a,t) = \frac{1}{T} H_{\mathrm{d}}(a) H_{\mathrm{a}}(\mathcal{L}(\mathrm{a},t)) \int_{\Omega} H_{\mathrm{g}}(\mathcal{L}(a,t),\boldsymbol{x})\; f(\boldsymbol{x})\; d\boldsymbol{x} \quad . \tag{3.11}$$

We can substitute a by $\mathcal{L}^{-1}(a,t)$ for the whole equation and then get

$$\hat{g}(\mathcal{L}^{-1}(a,t),t) = \frac{1}{T} H_{\mathrm{d}}(\mathcal{L}^{-1}(a,t)) H_{\mathrm{a}}(a) \int_{\Omega} H_{\mathrm{g}}(a,\boldsymbol{x})\ f(\boldsymbol{x})\ d\boldsymbol{x}\ . \tag{3.12}$$

Now, the idea of the rebinning procedure is to change the measured data such that it fits to the static model. Once this is achieved, then a static reconstruction algorithm can be used! To this end, we first compare (3.12) with the following equation from motion-free PET acquisition (expressed as a rate function here in order to make it comparable):

$$\frac{1}{T}\hat{g}(a) = \frac{1}{T} H_{\mathrm{d}}(a) H_{\mathrm{a}}(a) \int_{\Omega} H_{\mathrm{g}}(a,\boldsymbol{x})\ f(\boldsymbol{x})\ d\boldsymbol{x}\ . \tag{3.13}$$

We express $\frac{1}{T}\hat{g}(a)$ in terms of $\hat{g}(\mathcal{L}^{-1}(a,t),t)$ by replacing $\int_{\Omega} H_{\mathrm{g}}(a,\boldsymbol{x})\ f(\boldsymbol{x})\ d\boldsymbol{x}$ from equation (3.13) with the same term extracted from equation (3.12):

$$\frac{1}{T}\hat{g}(a) = \hat{g}(\mathcal{L}^{-1}(a,t),t) \frac{H_{\mathrm{d}}(a)}{H_{\mathrm{d}}(\mathcal{L}^{-1}(a,t))}\ . \tag{3.14}$$

Next, we can replace a by $\mathcal{L}(a,t)$:

$$\frac{1}{T}\hat{g}(\mathcal{L}(a,t)) = \hat{g}(a,t) \frac{H_{\mathrm{d}}(\mathcal{L}(a,t))}{H_{\mathrm{d}}(a)}\ . \tag{3.15}$$

This result has to be interpreted as follows: for each list-mode event measured in a dynamic setting, we can convert it to an event which would have been detected in absence of motion in a static setting. This is done as follows:

- the event has to be re-weighted according to the factor $\frac{H_{\mathrm{d}}(\mathcal{L}(a,t))}{H_{\mathrm{d}}(a)}$.
- the event has to be attributed to LOR $\mathcal{L}(a,t)$ in the static measurement vector.

These two steps are exactly the rebinning procedure which we derived intuitively beforehand!

Out-Of-Field-Of-View LORs Two types of LORs need special attention [138]:

- Type-I: LORs that are measured, but, when transformed to the reference frame do not correspond to a valid LOR anymore (e. g. when they fall out of the field-of-view). In other words: LORs that are measured but would not have been measured had the object not moved. Mathematically, this is the case for an LOR a_{motion} measured at time t when $a_{\text{no motion}} = \mathcal{L}(a_{\text{motion}},t) \notin \Lambda$.
- Type-II: LORs that are not measured but would have been measured had the object not moved. This is the case for an LOR $a_{\text{no motion}}$ which would have been measured at time t had the object not moved for which $a_{\text{motion}} = \mathcal{L}^{-1}(a_{\text{no motion}},t) \notin \Lambda$.

Some authors simply ignore both types of LORs [138, 11].

Fulton *et al.* use a sinogram mode filtered backprojection (FBP) reconstruction in which they ignore Type-I LORs, but they also use a list mode ML-EM algorithm for reconstruction in which these LORs were considered [61]. Bloomfield *et al.* suggest to include these LORs by means of an extended sinogram, but do not provide an implementation or proof-of-concept of this idea [11]. Thielemans *et al.* [193] ignore Type-I LORs. Bühler *et al.* state that Type-I LORs are not a source of artifacts but rather decrease the statistics and thus can be safely ignored, however, if considered, they would potentially increase the count rate and thus improve the image quality [28].

Type-II LORs however do cause artifacts since they lead to count losses. In order to correct for this, Bühler *et al.* calculate for each bin for which fraction of the whole acquisition time it was detectable in the FOV. Then, after the data has been sorted to the sinogram bins, the number of counts for each bin is increased by factor $\frac{\text{complete acquisition time}}{\text{detectable time}}$. Thus, if the bin is detectable for the whole acquisition time the increase factor is equal to one. Thielemans *et al.* [193] reconstruct in 2D mode and leave out sinograms which are too much affected by LORs which fell out of the field-of-view due to motion.

Weighting Correction All necessary aspects in order to understand weighting correction are included in equation (3.15). The essence of this equation is as follows: a measurement in detector pair a has to be attributed to detector pair $\mathcal{L}(a,t)$, where it would have been measured had the object not moved. The detector efficiency that applies to the measurement corresponds to detector pair a. However, had the object not moved, it would have suffered from detection efficiency of detector pair $\mathcal{L}(a,t)$. This is why a rebinned measurement has to be re-weighted by factor $\frac{H_\mathrm{d}(\mathcal{L}(a,t))}{H_\mathrm{d}(a)}$.

Historically, this problem is first identified (but not treated) in [138] and treated in [11, 193, 61, 28].

Discretization Correction Another challenge related to the rebinning approach is referenced as "LOR Discretization Correction" in [28] (note that the term LOR is often used synonymously to a detector pair in the literature). When a coincidence event is rebinned, it is first represented by two points in space which are transformed according to the current motion state to the reference frame. Then, in general, the transformed point pair will not correspond exactly to a point pair of an existing detector pair. The measurement can now be simply attributed to the "closest" detector pair (nearest neighbor) [61, 193], or, some kind of interpolation can be applied which redistributes a measurement to several detector pairs according to some weighting scheme. Bühler *et al.* state that a detector pair should be defined as the volume confined by the planes connecting the two detector surfaces. Then, the transformation which transforms the detector pair would act on the volume instead of two points which are often used for representation. Again, in general the transformed volume does not fit exactly a detector pair volume in the reference frame. The weight would then be defined by the intersection volume of the transformed detector pair with the intersecting detector pair. Bühler *et al.* consider this idea to be too computationally expensive and thus implement an approximation using 2D projections of the detector pairs in the axial planes.

Comparison to Other Approaches In a comparison study, the rebinning approach is considered superior to the MAF approach [62]. The MAF approach has been used in several human brain studies [63, 189, 85, 84, 47] and in small animal imaging [108].

3.2.1.2.3. Beyond the Rebinning Approach: System Matrix Modeling The basic idea of system matrix modeling is explained for the general case of nonrigid motion in section 3.2.2.2.3, where we derive equation (3.30) for a motion compensated system matrix

$$\overline{H}(a,\boldsymbol{x}) = \frac{1}{T}\int_0^T |\det(\nabla\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)| H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) H_{\mathrm{g}}(\mathcal{L}(a,\boldsymbol{x},t),\boldsymbol{x})\; dt \quad . \tag{3.16}$$

In case of rigid motion, the determinant is equal to one and also the LOR transformation function $\mathcal{L}$ does not depend on the annihilation position anymore, thus we arrive at

$$\overline{H}(a,\boldsymbol{x}) = \frac{1}{T}\int_0^T H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) H_{\mathrm{g}}(\mathcal{L}(a,t),\boldsymbol{x})\; dt \quad . \tag{3.17}$$

Rahmim *et al.* use the system matrix modeling approach for both list-mode and histogram data [165]. Further work includes scattered and random events and also discusses briefly an image space point of view (within the scope of an algorithmic acceleration of the calculation of the sensitivity factors) [167, 166].

3.2.1.2.4. Deconvolution An interesting though not popular approach on finding the image estimate based on deconvolution is investigated by Menke *et al.* [138]. Even though this method is not restricted to rigid motion, we discuss it here since the only reference we found applied it to rigid motion. Reordering of the integrals of the dynamic model which was introduced in equation (3.8) on page 41 leads to

$$\hat{g}(a) = \frac{1}{T} H_{\mathrm{d}}(a) \int_\Omega H_{\mathrm{g}}(a,\boldsymbol{x}) \int_0^T H_{\mathrm{a}}(a,t) f(\boldsymbol{\varphi}(\boldsymbol{x},t))\; dt\; d\boldsymbol{x} \quad . \tag{3.18}$$

Now, $\int_0^T H_{\mathrm{a}}(a,t) f(\boldsymbol{\varphi}(\boldsymbol{x},t))\; dt$ can be interpreted as a blurred image. Blur is usually modeled mathematically by convolution, and thus one could write

$$\int_0^T H_{\mathrm{a}}(a,t) f(\boldsymbol{\varphi}(\boldsymbol{x},t))\; dt = k_{\boldsymbol{\varphi}} * f \quad , \tag{3.19}$$

where $k_{\boldsymbol{\varphi}}$ is a shift-variant convolution kernel that includes the effects of motion and attenuation.

Then, the dynamic model rewrites to

$$\hat{g}(a) = \frac{1}{T} H_{\mathrm{d}}(a) \int_\Omega H_{\mathrm{g}}(a,\boldsymbol{x}) k_{\boldsymbol{\varphi}} * f\; d\boldsymbol{x} \quad . \tag{3.20}$$

Now we see from equation (3.20) that if a static image reconstruction is performed, we actually obtain $k_{\boldsymbol{\varphi}} * f$ as a result. The deconvolution motion-compensation method acts on this result and tries to find the image f by deconvolving $k_{\boldsymbol{\varphi}} * f$. Note that this is a shift-variant deconvolution with a known convolution kernel $k_{\boldsymbol{\varphi}}$ ($k_{\boldsymbol{\varphi}}$ can be pre-calculated from the presumably known motion and attenuation map).

3.2.2. Nonrigid Motion

Nonrigid motion is harder to deal with than rigid motion. Firstly, it is much harder to estimate the motion, since motion tracking techniques can only acquire the position and orientation of a tracking target. Nonrigid motion could be approximated by using several tracking targets, but these would be restricted to the surface. The most challenging and interesting motion takes place in the inner body, usually due to heart beat or respiration. For nonrigid motion, it is much more common to estimate the motion by registration of individually reconstructed frames.

Also the image estimation part requires some rethinking. For example, the event driven rebinning approach for image estimation is not applicable anymore: nonrigid motion converts straight lines into curves, which thus do not have a representation in the sinogram anymore. Image estimation by summation of reconstructed and registered frames is very similar to the MAF approach for rigid motion. Also deconvolution would be feasible but has not been tried, probably due to the little success the approach had for nonrigid motion.

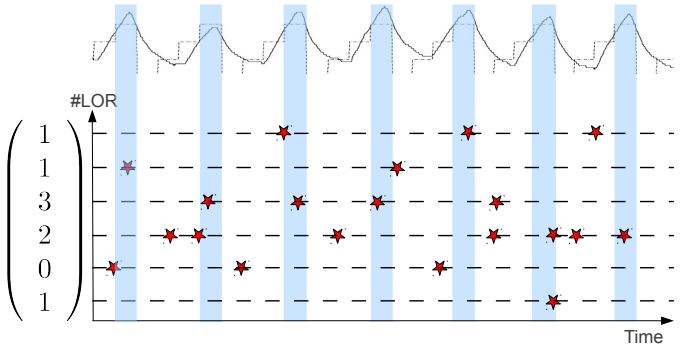
3.2.2.1. Motion Estimation

3.2.2.1.1. Tracking In general, tracking approaches are much less effective for nonrigid motion than they are for rigid motion. At its best, one could estimate the thorax's surface, but this is not a direct estimate for the organs. Therefore, only few researchers have used tracking alone as a motion estimate. Usually, tracking by means of optical motion tracking systems or time-of-flight cameras (a camera system that creates distance data) is combined with a previously established motion model (see section 3.2.2.1.5). Koshino *et al.* use optical tracking together with a rigid motion assumption for respiratory motion [107].

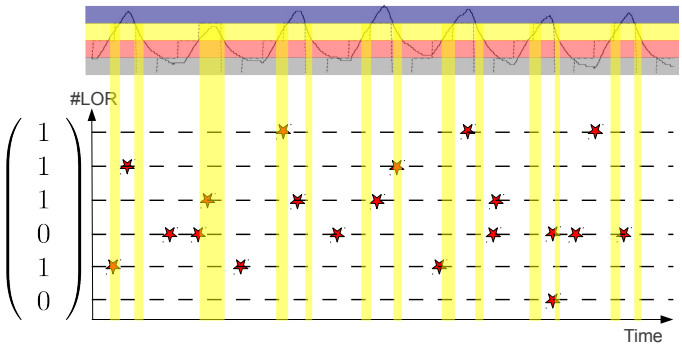
3.2.2.1.2. Gating Gating is not a motion estimation method on its own, but rather a preprocessing step for other motion estimation tasks. The idea of gating is very similar to the MAF approach discussed in section 3.2.1.1.2: the acquired data is partitioned into sets of data in such manner that each set corresponds to only one anatomical motion state. A motion free image results from reconstructing one of the gates. However, this motion free image is noisier than a complete reconstruction of the whole data, since it arises from only one gate which has a reduced number of counts compared to the whole acquisition.

For respiratory gating, traditional methods gather the respiratory signal by using a respiratory belt, a spirometer, a thermopile or an infrared stereo-vision system tracking the motion of thoracic markers [133]. Also methods that find the respiratory signal from the measured list-mode data itself have been presented [24, 26], even for cardiac gating [25].

Based on this respiratory signal, either amplitude based gating or phase based gating can be applied (Figure 3.4). While phase based gating is more robust against shifts of the respiratory signal's baseline (as they usually occur with a respiratory belt), amplitude based gating seems to make more sense since it measures more directly the actual extension.



(a)



(b)

Figure 3.4.: (a) Phase-based gating: the respiratory signal is divided into four gates (in this case) according to the phase of the signal. (b) Amplitude-based gating: the respiratory signal is divided into four gates based on predefined amplitude bands.
To the left, the data vector of one respiratory gate is shown. As can be seen, the two gating methods lead to different gates.

In dual gating approaches both respiratory signal and cardiac signal are used as a base for gating [134].

3.2.2.1.3. Image Registration The image registration motion estimation method consists of two steps, based on gated data:

1. Reconstruct each gate individually by any static reconstruction method.
2. Choose a reference image and estimate the motion for the remaining images with respect to the reference image by means of image registration methods.

Since medical image registration is an active field of research for several decades already, it is beyond the scope of this thesis to give a detailed overview (the interested reader is referred to [80, 139, 75]). Here, we will focus on image registration applied to

reconstructed gated PET images in order to estimate the motion field. A building block of most of these methods is an objective function which measures the squared L^2 norm of the difference between the reference image f_R and the deformed template image f_T:

$$\mathcal{D}(\boldsymbol{\varphi}) = \int_{\Omega} (f_R(\boldsymbol{x}) - f_T(\boldsymbol{\varphi}(\boldsymbol{x}))^2 \, d\boldsymbol{x} \ , \tag{3.21}$$

where $\boldsymbol{\varphi} : \mathbb{R}^3 \mapsto \mathbb{R}^3$ is a deformation function that maps a spatial position to a new position. The interim goal of image registration is to find a deformation function $\boldsymbol{\varphi}$ which minimizes equation (3.21). We explicitly state that this is an interim goal since the minimization problem is ill-posed up to now. In general, there is not a unique deformation which will minimize the given functional. Furthermore, we usually even do not want to minimize $\mathcal{D}$ the way it is stated at the moment.

In order to understand this, let us look at a simple virtual example: we want to register an image of an apple with another similar apple. The apples are both red and the images are taken with a blue sky as background. Further on, the template apple has a little sticker on it which provides information about its brand. Let us assume that the sticker has the same color as the blue sky in the reference image. Now the problem with our interim goal becomes obvious: in order to maximize the similarity, the displacement function will map the sticker into the blue sky. What we really want is a deformation function that minimizes the similarity *while* respecting physical constraints. In the case of an apple we do not want to change the relative position of the sticker at all. In the case of human tissue we accept strains, but the bigger they are the more they have to be restricted. Accordingly, depending on which kind of tissue we are registering, the motion function should respect the laws of physics and not change the relative position of points too much. We do not want a an ear to end up in the brain just because the images become more similar then (in a least-squares sense).

Mathematically, the constraint is often included by a regularization term which is simply added to the similarity term:

$$\mathcal{J}(\boldsymbol{\varphi}) := \mathcal{D}(\boldsymbol{\varphi}) + \alpha \, \mathcal{R}(\boldsymbol{\varphi}) \ . \tag{3.22}$$

The regularization term $\mathcal{R}$ includes physically motivated conditions on the deformation function, e. g. many authors use the elastic potential energy. $\mathcal{R}$ together with α change the minimum of $\mathcal{J}$, and thus a more physically meaningful $\boldsymbol{\varphi}$ is the minimizer. Note that the regularization parameter α has a major impact on the result, and thus has to be chosen carefully in advance. Often this is a trial and error manual procedure, and one of the open problems in image registration is to find a way to adjust α automatically. We will come back to this discussion when presenting our joint estimation method which also has such a regularization parameter.

First attempts in cardiac PET have been made by Klein *et al.* [103, 102, 100]. They minimize the squared L^2 norm between the reference and a template image. Additionally, a smoothness constraint based on the strain energy and non-compressibility constraint is used in order to regularize the motion field.

In a further work, Klein *et al.* extend their approach to 4D motion applied to dual gating [104, 101]. Dual gating is more challenging since with an increasing number of

gates each gate has less counts. Accordingly, the reconstructed gates are extremely noisy which complicates the image registration. Klein *et al.* tackle this problem by introducing a temporal prior: the gates are sorted in such manner that adjacent gates correspond to a similar motion state. This temporal prior is included in the objective function as a regularization term. Additionally, all reconstructed gates are included in the objective function (in contrast to an objective function which is defined on two images only and then applied pairwise).

Qiao *et al.* use a registration method from the Insight Toolkit [204, 94] applied on gated CT images [163]. The motion function is parametrized by B-splines. Dawood *et al.* use a pairwise registration based on the Lucas-Kanade optical flow algorithm [127, 50]. Gilland *et al.* present a modification of the Horn and Schunck optical flow method [89, 68, 69]. Parker *et al.* use the Horn and Schunck optical flow method and compare it to a joint estimation approach which is discussed in section 3.3.1 [152, 129]. In SPECT, Gravier *et al.* use the Horn and Schunck method [77, 76]. Lamare *et al.* use the ideal images of a simulation study for affine registration [115]. In a subsequent study, these ideal images are used for minimization of an L^2 norm (as described above) and the motion function is discretized as an interpolation of B-splines [112]. Also Chen and Tsui use a B-spline motion model for registration [38].

3.2.2.1.4. PET-MR Although, to the best of our knowledge, no papers have been presented with respect to the prospective acquisition correction for nonrigid motion (refer section 3.2.1.1.4), this technique principally is extendible to general, deformable motions. It could be the case however that the image quality is not good enough in order to reliably estimate the extended parameter set needed for nonrigid motions. The cloverleaf navigator is not extendible to nonrigid motions.

Another MR technique for motion estimation is tagged MRI. Tagged MRI produces noninvasive markers using spin tagging concepts. The markers persist for at least some tenth of seconds in the tissue. Technically, the longitudinal magnetization of the tissue is modulated shortly before sequence of images is acquired. The most common technique used to produce tag patterns is spatial modulation of magnetization (SPAMM). Tagged MRI has been used for over a decade in order to assess myocardial deformation [137].

Chun *et al.* use gated PET and gated tagged MR images [44]. The MR gates are then registered to each other using a B-Spline motion model in order to find the motion between the gates. Two figures of merit (FOM) for registration were compared: Mutual information (MI) and sum of squared differences (SSD). The algorithm was applied to a physical phantom that mimics respiratory motion. Lesion detectability (measured by improvement of the SNR) was improved by more than 50% by the MR gating motion compensation procedure compared to pure PET gating.

Though in the abstract it is not discussed why tagged MR images are used, presumably the registration procedure is improved/simplified by using tags. It seems that untagged MR is not yet fast enough (for the desired image quality) in order to acquire real-time 4D video sequences, and thus gating must be used in conjunction.

3.2.2.1.5. Model Based Approaches Model based approaches have been presented for respiratory motion. The general idea is to use the respiratory signal and deduce from it the deformation field based on a previous model generated from CT or MR images. The model can be patient specific or general.

Reyes *et al.* take two MR scans, one for maximal inspiration and one for maximal expiration [172]. One is used as a template and registered to the other. Then, an mean of the two images is co-registered to the PET image in order to calibrate the model to the PET coordinate system. Based on the current motion signal, the motion field is now deduced from a scaling of the acquired motion field.

Fayad *et al.* build a patient specific model from 4D CT scans [56]. The CT scans are subdivided into 10 respiratory frames and co-registered in order to obtain the motion field. Then, during the PET acquisition, the CT scans are adopted to the current motion phase by means of a time-of-flight camera, which results in the sought deformation field during the PET acquisition. In a follow up work, Fayad *et al.* build a generic model from four 4D CT scans of different patients [55]. The obtained motion fields are adapted to the specific patient by means of two static CT scans. During the PET acquisition, again a time-of-flight camera provides the necessary information in order to obtain the sought motion field at any desired time.

3.2.2.2. Image Estimation

In the following, we review methods of estimating the image based on a known motion function.

3.2.2.2.1. Reconstruction of Individual Gates The most simple way to obtain motion-free images is to reconstruct each gate individually by any static PET reconstruction algorithm. As usual for static image reconstruction algorithms, spatial priors (like e. g. Gibbs priors) can be included in order to encourage smoother images. Gravier *et al.* define a likelihood function which includes all frames in a likelihood function [77, 76]. Additionally they add a temporal prior which encourages a smooth transition between the reconstructed gates.

3.2.2.2.2. Summation of Reconstructed Gates The summation of reconstructed frames approach is based on reconstructed gates. The motion field is used in order to transform each gate to a reference gate. Then, the gates are added and the final result is an image with a higher signal-to-noise ratio. Many of the papers we reviewed in section 3.2.2.1.3 use this approach, e. g. [103, 102, 100, 50] just to name a few.

3.2.2.2.3. System Matrix Modeling It is possible to model the motion directly into the system matrix. We start with equation (3.8) which we motivated at the beginning of this chapter:

$$\hat{g}(a,t) = \frac{1}{T} H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) \int_{\Omega} H_{\mathrm{g}}(a,\boldsymbol{x}) f(\boldsymbol{\varphi}(\boldsymbol{x},t))\, d\boldsymbol{x} \quad .$$

In the following we will show how to model the object motion into the geometric system matrix in two different manners: one is what we call the image space perspective, and the other the LOR space perspective.

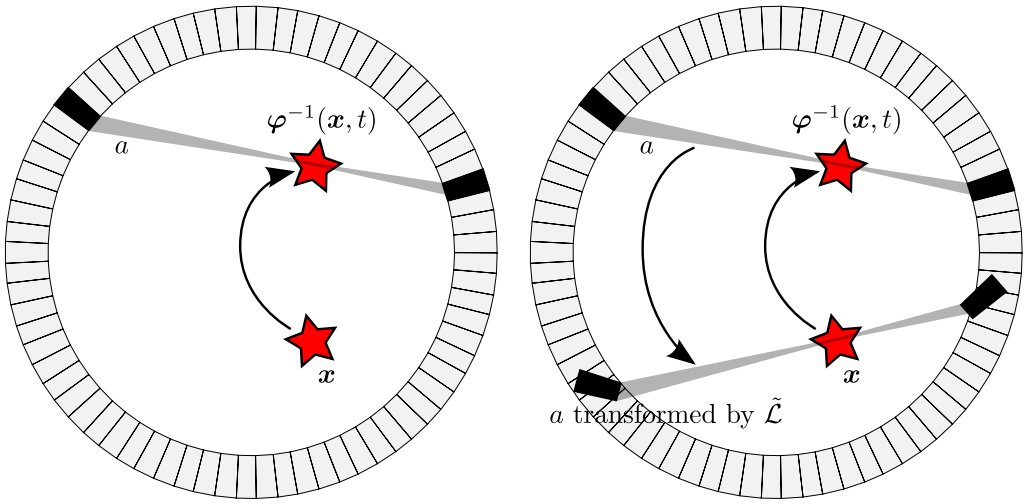
Dynamic Geometric Factor From an Image Space Perspective By performing an integral substitution (see e. g. [175]), we can equivalently write

$$\hat{g}(a,t) = \frac{1}{T} H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) \int_{\boldsymbol{\varphi}^{-1}(\Omega)} |\det(\nabla \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)|\; H_{\mathrm{g}}(a, \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t))\; f(\boldsymbol{x})\; d\boldsymbol{x}\;. \tag{3.23}$$

If $\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)$ is the position where the photon pair is actually emitted at time t, then $\boldsymbol{x}$ is the position where the emission would have taken place had the object not moved. Note that the inversion of $\boldsymbol{\varphi}$ is meant to be made only with respect to parameter $\boldsymbol{x}$, not with respect to time.
From an implementation point-of-view we distinguish between on-the-fly calculations and pre-calculated system matrices. We assume that in the static case the geometric probability is approximated by calculating the average solid angle of several uniformly distributed positions within a certain voxel. In presence of motion, if the system matrix is calculated on-the-fly, one could move these positions according to the present motion and then average the solid angle for these new positions. For a discrete pre-calculated system matrix (which means that the probability for an emission from a voxel to be detected by a certain detector pair is pre-calculated), one could move the voxel center and then interpolate the detection probability from its neighboring voxels (since usually the voxel center will not be mapped exactly to a new voxel center).
From now on, for the sake of a simpler notation, we will replace $\int_{\boldsymbol{\varphi}^{-1}(\Omega)}$ by $\int_{\Omega}$ since we assume that any function (f, H, ...) is zero outside its support.

Dynamic Geometric Factor From an LOR Space Perspective Now, in theory, instead of moving the emission position with respect to the detector pair, one could also move the detector pair with respect to the emission position (see 3.5b).
How exactly should the detectors be transformed? If we are faced with a rigid transformation $\boldsymbol{\varphi}^{-1}$, then this would be clearly the inverse rigid transformation $\boldsymbol{\varphi}$. In case of a nonrigid transformation, the detectors still have to be moved by a rigid transformation which we will define in the following by two point correspondences. Consider an event emitted at $\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)$ and another, very close point $\boldsymbol{\varphi}^{-1}(\boldsymbol{x}+d\boldsymbol{x},t)$. Then, $\boldsymbol{x}$ is the position where the event would have been emitted and $\boldsymbol{x}+d\boldsymbol{x}$ is the corresponding point which is still very close. Now, the rigid transformation which transforms $\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)$ and $\boldsymbol{\varphi}^{-1}(\boldsymbol{x}+d\boldsymbol{x},t)$ to $\boldsymbol{x}$ and $\boldsymbol{x}+d\boldsymbol{x}$ is the rigid transformation which is needed for the detectors. We will denote this rigid transformation as $\tilde{\mathcal{L}} : \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R} \mapsto \mathbb{R}^3$. The rigid transformation of any position $\boldsymbol{y}$ is then given by $\tilde{\mathcal{L}}(\boldsymbol{y},\boldsymbol{x},t)$, where $\boldsymbol{x}$ is the annihilation position had the object not moved and t is the respective time.

In practice, detector pairs or sinogram bins are identified by an index, and so we define a corresponding operator $\mathcal{L} : \mathbb{N} \times \mathbb{R}^3 \times \mathbb{R} \mapsto \mathbb{N}$ which maps a detector pair a to a new (existing) detector pair $\mathcal{L}(a,\boldsymbol{x},t)$ which is closest to the transformed detector pair a



(a) (b)

Figure 3.5.: (a) An event which would have been emitted at time t at position $\boldsymbol{x}$ had the object not moved, at time t is emitted at position $\boldsymbol{\varphi}(\boldsymbol{x},t)$ due to motion. (b) Had the object not moved, the event would have been detected with the same probability by a detector pair a being rigidly transformed by the operator $\tilde{\mathcal{L}}$ (see text for details).

transformed by the rigid operator $\tilde{\mathcal{L}}$. Again, $\boldsymbol{x}$ is the annihilation position had the object not moved. Note that in an implementation one would usually use interpolation between several neighboring detector pairs, however, for the sake of a simpler notation we assume a nearest neighbor interpolation (the closest detector pair is chosen).

We then equivalently write

$$\hat{g}(a,t) = \frac{1}{T} H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) \int_{\boldsymbol{\varphi}^{-1}(\Omega)} |\det(\nabla \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)|\; H_{\mathrm{g}}(\mathcal{L}(a,\boldsymbol{x},t),\boldsymbol{x})\; f(\boldsymbol{x})\; d\boldsymbol{x} \tag{3.24}$$

It seems that at least for nonrigid motion, it is easier to model the motion in image space than in LOR space. This may be due to the fact that motion *is* an image space phenomenon.

List-mode First attempts to model the motion into the reconstruction algorithm have been made by Qi and Huesman for list-mode data [158]. Remember the static list-mode log-likelihood function which was derived in equation (2.64)

$$L(\boldsymbol{f}) = \sum_{k=1}^{N} \left\{ \log \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{a_k i} [\boldsymbol{f}]_i \right\} - \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{s}]_i [\boldsymbol{f}]_i - \log N! \tag{3.25}$$

(we replaced $P(a_k|i)[\boldsymbol{s}]_i$ by $[\boldsymbol{H}]_{a_k i}$). Now, Qi and Huesman include motion directly into

the system model, which is then time dependent (as explained above). Accordingly, the likelihood function rewrites to

$$L(\boldsymbol{f}) = \sum_{k=1}^{N} \left\{ \log \sum_{i=1}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{a_k i}^{t(a_k)} [\boldsymbol{f}]_i \right\} - \sum_{i=1}^{N^{\boldsymbol{f}}} [\tilde{\boldsymbol{s}}]_i [\boldsymbol{f}]_i - \log N! \ , \tag{3.26}$$

where $[\boldsymbol{H}]_{a_k i}^{t(a_k)}$ is a discrete version of the time dependent system response function $H(a_k, \boldsymbol{x}, t(a_k))$ (instead of $\boldsymbol{x}$ we have a certain voxel i), where $t(a_k)$ is the time at which event a_k is measured. Some ideas on how to factor $[\boldsymbol{H}]_{ai}^{t}$ and calculate the different parts were given above.

The time dependence of the system model implies that the sensitivity has to be adapted to the motion (it is denoted as $\tilde{\boldsymbol{s}}$ in the motion case):

$$[\tilde{\boldsymbol{s}}]_i = \sum_a \int [\boldsymbol{H}]_{ai}^{t} \, dt \ . \tag{3.27}$$

A more elaborate and general derivation based on the list-mode derivation of Parra and Barrett [153] is given by Qiao *et al.* [162, 163]. Lamare *et al.* arrive at the same algorithm with similar considerations and give some nice intuitive explanations [113, 114].

Histogram Mode Once the motion is modeled inside the system matrix we can calculate a motion compensated system matrix. This is very attractive since then any existing static reconstruction method can be employed. We start with equation (3.1) which states the expected number of counts measured for each LOR within the whole scan given the dynamic count rate:

$$\hat{g}(a) = \int_0^T \hat{g}(a,t) \, dt \ .$$

Inserting the image perspective equation (3.23) of the dynamic count rate, we arrive at (after rearranging the integrals)

$$\hat{g}(a) = \int_\Omega \frac{1}{T} \int_0^T |\det(\nabla \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)| H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) H_{\mathrm{g}}(a, \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)) \ dt \ f(\boldsymbol{x}) \, d\boldsymbol{x} \ . \tag{3.28}$$

The motion compensated system response function $\overline{H}$ is defined as

$$\overline{H}(a,\boldsymbol{x}) = \frac{1}{T} \int_0^T |\det(\nabla \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)| H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) H_{\mathrm{g}}(a, \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)) \ dt \ . \tag{3.29}$$

Exactly the same procedure can be employed in the LOR perspective based on equation (3.24):

$$\overline{H}(a,\boldsymbol{x}) = \frac{1}{T} \int_0^T |\det(\nabla \boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)| H_{\mathrm{d}}(a) H_{\mathrm{a}}(a,t) H_{\mathrm{g}}(\mathcal{L}(a,\boldsymbol{x},t),\boldsymbol{x}) \ dt \ . \tag{3.30}$$

The motion compensated system matrix $\overline{H}$ can be discretized and plugged into an ML-EM algorithm or any other static reconstruction method.

3.3. Joint Estimation of Image and Motion

All the previously discussed methods separate motion estimation from image estimation. In contrast, joint estimation approaches are characterized by estimating both image and motion in a combined manner. Note that “in a combined manner" is not a very solid definition, since at least when it comes to the concrete optimization procedure even joint estimation methods alternate between image and motion updates. Accordingly, to be more precise, when we talk about joint estimation, we refer to methods which use an objective function which contains both image and motion and is subject to optimization. Joint estimation approaches differ from each other by the objective function that is being optimized and the optimization procedure (that is, how the objective function is optimized).

3.3.1. Several Image Estimates

The first published work (to the best of our knowledge) of defining an objective function which includes both image reconstruction and motion estimation terms is known from Cao *et al.* [130, 67, 33, 32, 31, 129]. Cao *et al.* use an objective function

$$E(f_1, f_2, \boldsymbol{m}) = \gamma\, E_I(f_1, f_2, \boldsymbol{m}) + \alpha\, E_S(\boldsymbol{m}) + L(f_1, f_2) \ , \tag{3.31}$$

which depends on two images f_1 and f_2 and a motion displacement field $\boldsymbol{m}$. The goal is to find parameters (images and motion) which minimize E. E is a weighted sum of several sub-terms, namely E_I, E_S and L.

$$E_I(f_1, f_2, \boldsymbol{m}) = \int_\Omega \left(f_1(\boldsymbol{x}) - f_2(\boldsymbol{x} + \boldsymbol{m}(\boldsymbol{x}))\right)^2 \, d\boldsymbol{x} \tag{3.32}$$

is a dissimilarity term between the first image and the deformed second image. If $\boldsymbol{m}$ is the correct displacement between f_1 and f_2, then the squared difference should be relatively low compared to a bad matching. Note that due to noise E_I should not be excessively minimized, since otherwise high peaks of f_1 would be matched to high peaks of f_2. A way to avoid such an over-optimization is to encourage smooth motion fields, which is achieved by the regularization term E_S, defined as the strain energy induced by motion. Finally, there is a negative log-likelihood term which arises from the probability of detecting the measured data given the current image estimates f_1 and f_2

$$L(f_1, f_2) = \sum_{t=1}^{2} \sum_{a=1}^{p} \hat{g}(a,t) - g(a,t) \log \hat{g}(a,t) \ , \tag{3.33}$$

where the expected measured data is given by the projection of the respective image

$$\hat{g}(a,t) = \int_\Omega H(a, \boldsymbol{x}) f_t(\boldsymbol{x}) \, d\boldsymbol{x} \ . \tag{3.34}$$

3.3.2. One Image Estimate

Joint estimation approaches that use one image estimate are based on an objective function of type

$$\mathcal{J}(f, \boldsymbol{\varphi}) = \mathcal{D}(f, \boldsymbol{\varphi}) + \alpha\, \mathcal{R}_1(f) + \beta\, \mathcal{R}_2(\boldsymbol{\varphi}) \quad . \tag{3.35}$$

The difference term $\mathcal{D}$ measures the similarity of the dynamic estimated data to the measured data. The estimated data arises from the image estimate f which is transformed by motion function $\boldsymbol{\varphi}$ and then projected to sinogram space.

Such an approach for ET was at first presented by Jacobson and Fessler [97]. They use the negative log-likelihood function for the measured data given an image and motion estimate

$$\mathcal{D}(f, \boldsymbol{\varphi}) := \sum_{a,t} \hat{g}(a,t) - g(a,t) \log\left(\hat{g}(a,t)\right) \quad , \tag{3.36}$$

where

$$\hat{g}(a,t) = \int_{\Omega} H(a, \boldsymbol{x}) f(\varphi(\boldsymbol{x}, t))\, d\boldsymbol{x} \quad . \tag{3.37}$$

Regularization terms on image and data are mentioned but not further specified in [97].

In the experimental part of the paper, Jacobson and Fessler work with a 64×64 torso phantom which contains a circular hot lesion of 2 cm diameter. They generate two time frames, and the second time frame is stretched by 10% in one dimension. In their experiments, all cost function minimizations are unregularized. The solution space for the motion phantom is limited to one dimension and is discretized to 22 values, ranging from 5% stretching to 15% stretching. Optimization with respect to the image is accomplished by the ML-EM algorithm (for a fixed motion function), while the optimization of motion parameters is done by an exhaustive search.

At this point, both the method and its evaluation has to be considered premature for several reasons: first, only a very restricted parameter space for motion was investigated. Second, the regularization terms for image and motion were proposed but not used. Third, optimization was done by an exhaustive parameter search. Even though Jacobson and Fessler derive an optimization scheme based on optimization transfer with surrogate functions, it seems that this optimization scheme is not applied in [97].

In a subsequent work, a more realistic motion model based on 3D B-splines is proposed in [96]. Five synthetic 3D PET data frames are derived from a 4D CT scan. No motion regularization is applied, but an "activity roughness penalty function" (no more details are given in the paper) is used in order to regularize the image. Usually these penalty functions are based on penalizing high gradients and thus lead to a smoother image. Since it is not desired to have an extremely smooth image in the tumor region (this would lead to a similar effect as the blurring due to motion), Jacobson and Fessler define a region-of-interest (ROI) within which no regularization is applied. The CT scan is not only used for PET simulation, but also in order to mark this ROI around the lung tumor.

Chun and Fessler combine a novel motion regularization term [41] with joint estimation in [42]. They use a weighted least squares term

$$\mathcal{D}(f, \boldsymbol{\varphi}) := \sum_{t,a} \frac{1}{\hat{g}(a,t)} \left(\hat{g}(a,t) - g(a,t)\right)^2 \quad . \tag{3.38}$$

Furthermore, they suppose that the motion function is an identity transformation for one of the gates/frames. All experimental results are calculated in 2D, no details on the optimization procedure are given.

As will be seen later, our proposed joint estimation approach works with a very similar objective function as the one given in (3.35). We also use an optimization scheme which alternates between image and motion updates. Our image update is similarly calculated as in Jacobson's and Fessler's work. However, instead of using optimization transfer by surrogate functions (also known as majorize-minorize optimization), we use a gradient based optimization scheme for the motion update. A major problem with the gradients involved in the motion update is that they require projections to measurement space (sinogram space). These projections are needed several times for each motion update (compare the gradient equations derived in our proposed joint estimation method). Thus, this implies a significant computational cost and limits practical applications at the moment. In [59], Fessler proposes an optimization transfer approach using a surrogate function which operates completely in image space, and thus does not suffer from the time consuming projections to measurement space. Fessler exemplarily derives his approach for penalized weighted least squares (PWLS) and claims it is easily extendible to other similarity terms like e. g. likelihood terms. Since no evaluation nor experimental results are given in [59], further investigation is needed in order to make a proof-of-concept.

In SPECT, Schumacher *et al.* use a least squares similarity measure [182, 183]

$$\mathcal{D}(f, \boldsymbol{\varphi}) := \sum_{t,a} \left(\hat{g}(a,t) - g(a,t)\right)^2 \quad . \tag{3.39}$$

$\mathcal{J}$ is minimized subject to a positive image f (Schumacher *et al.* replace f by e^z in order to convert the constrained optimization problem into an unconstrained one). The motion is parametrized as a rigid six degrees-of-freedom transformation. Regularization of both image and motion is employed. $\mathcal{R}_1(f)$ is a smoothing term which encourages smooth image estimates. $\mathcal{R}_2(\boldsymbol{\varphi})$ is a regularization term which encourages small motions, in order to avoid ambiguities (otherwise the image would be only estimated up to translation).

3.4. Conclusion

In many of the reviewed papers, the authors do not state the underlying acquisition model on which they base their algorithms. For example, the rigid motion rebinning methods rely on the intuitive idea of moving the detector pair to another detector pair where the measurement would have measured had the object not moved. It is undoutable that the initial idea is convincing, but, by not having a mathematical model at hand, they run into several issues later which are less intuitive to explain and could be

more easily read off from an equation than by verbal arguments. Equation 3.15 on page 48 is a good example of how it comes for free to find the correct weighting parameter when using mathematical derivations. Other examples are the methods which first estimate nonrigid motion by registering individually reconstructed gates and then reconstruct the image based on this motion. Intuitively it is clear that this must somehow lead to an enhanced image.

We do not claim that it is generally a bad idea to solve a problem without a clear mathematical model behind. In fact, it seems natural that researchers come up first with the most simple and intuitive solutions. However, if we do not know which model is behind an algorithm, (i) we cannot judge to which degree the model is realistic and (ii) to which degree the algorithm deviates from the model. In this sense, we believe that the joint estimation approaches start from the correct basis, which is: a mathematical acquisition model containing the system response function, the image function, the motion function and the measured data.

Apart from these considerations, it is difficult to compare the discussed methods in terms of image quality, and has not been done so far in a comprehensive manner. An enormous amount of work would have to be invested to implement all these methods in a comparable framework. It seems that the only way to get this done one day is by an open comparison platform, which contains simulated PET data for an open PET system specification. All methods to be compared would have to use the same geometrical system model. In chapter 5 and 6 we will make the effort to at least compare our proposed method to a few other methods, although this list is by far not complete.

In terms of speed, joint reconstruction methods have a clear disadvantage compared to e. g. the rebinning approach for rigid motion. Rebinning can be even done online while the data is recorded, and therefore does not imply any additional waiting time compared to static reconstruction. Also the image registration followed by reconstruction approaches are considerably faster than the joint reconstruction methods.

With respect to the joint reconstruction methods, it seems not reasonable to have several image estimates, since in the end there can be only one image which is the true tracer distribution.

Part II.

A Novel Method for Joint Reconstruction

4. Joint Reconstruction Method

When we started our work in 2007, the only joint reconstruction method we found in an initial literature review was [182]. After our first results on rigid motion were presented at the IEEE Medical Imaging Conference 2007 in Dresden, M. Jacobson directed us to his work from 2003 [97]. His general tenor was that joint reconstruction, even for rigid motion, does not really work well, and it would probably not work for non-rigid motion. By the time we found some more publications (all referenced in the previous chapter). Nevertheless, the amount of papers dealing with joint reconstruction is still infinitesimal compared to other motion compensation methods. We believe that this situation is unfair, since for the reasons explained in the last chapter's conclusion, joint image and motion reconstruction is the physically best motivated way to tackle the problem. Admittedly, it is not the most simple approach - but this should not be a reason to not explore it deeper.

In the following, we design a cost functional which depends on both an image and a motion function and is minimal for a good fit of image and motion to the data. As a measure of fit, we use an adapted motion-aware likelihood function. Then, we derive a general framework for minimizing this cost functional. The most essential part of the minimization framework are two update equations (one for the image and one for the motion function).

4.1. Cost Functional

An explanation on the notations we use in this thesis are found in appendix C. The number of counts $g(a)$ that is measured for a line-of-response (LOR) a is Poisson distributed:

$$P(g(a)|f) = e^{-\hat{g}(a)} \cdot \frac{\hat{g}(a)^{g(a)}}{g(a)!} \quad . \tag{4.1}$$

$\hat{g}(a) = \frac{1}{T} \int H(a, \boldsymbol{x}) f(\boldsymbol{x})\, d\boldsymbol{x}$ is the expected number of counts given the tracer distribution f (which is subject to reconstruction) and the system response function H.

As introduced already in the previous chapter, in case of a moving anatomy, f can be modeled as $f(\boldsymbol{\varphi}(\boldsymbol{x}, t))$, where $\boldsymbol{\varphi}$ is a deformation function $\boldsymbol{\varphi} : \mathbb{R}^4 \mapsto \mathbb{R}^3$ which maps a position $\boldsymbol{x}$ in space at time t to a new position $\boldsymbol{\varphi}(\boldsymbol{x}, t)$ in the reference frame. Accordingly, f on its own (without the motion function) corresponds to a virtual reference/reconstruction frame and not to any of the input gates/frames. The probability distribution has to be modified to

$$P(g(a,t)|f, \boldsymbol{\varphi}) = e^{-\hat{g}(a,t)} \cdot \frac{\hat{g}(a,t)^{g(a,t)}}{g(a,t)!} \quad , \tag{4.2}$$

where

$$\hat{g}(a,t) = \frac{1}{T} \int H(a,\boldsymbol{x}) f(\boldsymbol{\varphi}(\boldsymbol{x},t))\, d\boldsymbol{x} \tag{4.3}$$

is the expected count rate in LOR a at time t given an image f and transformation $\boldsymbol{\varphi}$.

Note that t does not necessarily refer to a specific point of time. In our case, we define $t \in \mathbb{N}$ as a gate index (gating was introduced in the previous chapter) in which no movement takes place. Then, $\hat{g}(a,t)$ is interpreted as the expected number of counts for gate t.

The likelihood function for all measured events is

$$\mathcal{L}(f,\boldsymbol{\varphi}) = \prod_{a,t} P(g(a,t)|f,\boldsymbol{\varphi}) \ . \tag{4.4}$$

We seek to find a pair of image f and motion $\boldsymbol{\varphi}$ that maximizes the likelihood function. This is equivalent to minimizing the negative log-likelihood function

$$-\log(\mathcal{L}(f,\boldsymbol{\varphi})) = \sum_{a,t} \hat{g}(a,t) - g(a,t)\log\left(\hat{g}(a,t)\right) + \log(g(a,t)!) \ . \tag{4.5}$$

Here, $\sum_{t,a} \log(g(a,t)!)$ can be omitted since it does not affect the minimum. So, finally we arrive at

$$\mathcal{D}(f,\boldsymbol{\varphi}) := \sum_{a,t} \hat{g}(a,t) - g(a,t)\log\left(\hat{g}(a,t)\right) \tag{4.6}$$

which is subject to minimization. The next sections explain how this objective functional is optimized and how this is implemented on a computer.

4.2. General Optimization Scheme

Figure 4.1 describes the general optimization framework. The algorithm takes as an input an image f and a motion function $\boldsymbol{\varphi}$. As initial estimate, we use $f(\boldsymbol{x},t) = 1$ (for all gates t) and $\boldsymbol{\varphi}(\boldsymbol{x}) = \boldsymbol{x}$. The optimization algorithm we employ alternates between image updates and motion updates.

The image update consists of a fixed-point iteration which arises from the Karush-Kuhn-Tucker (KKT) conditions. It will be discussed in detail in section 4.5.1. After the image update, regularization (in this case Gaussian smoothing) is applied to the image. We choose Gaussian post-smoothing since it is easy to adjust: the standard deviation is meaningful enough in order to quickly find a reasonable value. Alternatively, a regularization term could be mathematically added to the objective functional, with the drawback of adding a regularization parameter which may be difficult to adjust.

The motion update consists of adding the negative gradient of the objective function weighted by a step-size factor (for details see section 4.5.2). If the motion update was

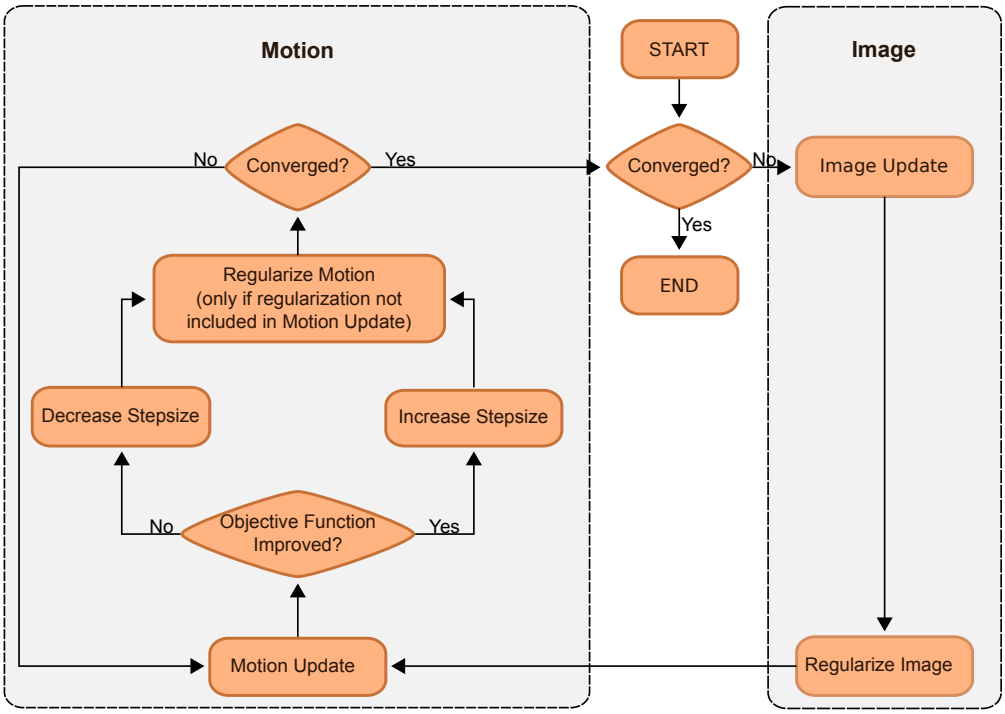


Figure 4.1.: General framework of our joint reconstruction algorithm.

successful, that is, the new motion parameter leads to a reduced objective function, the step-size will be increased by a positive factor. We usually use a moderate factor of 1.2. If the motion update was not successful, the step-size is multiplied by 0.5. Regularization can be applied in two manners: After the motion update, similar as for the image update, or, alternatively, the objective function can be modified and so the regularization takes place already in the summation with the gradient. In this thesis we explore both ways of regularization of the motion. Section 4.6 discusses regularization.

4.3. Gâteaux Derivatives of $\mathcal{D}$

For both the image and the motion update, derivatives of the cost functionals are needed. Since the cost functional is a function of functions (image and motion), we have to make use of Gâteaux derivatives. The Gâteaux derivative is introduced in appendix B as a generalization of the directional derivative. In the following, the Gâteaux derivative of $\mathcal{D}$ will be calculated with respect to image and motion separately.

4.3.1. Gâteaux Derivative of $\mathcal{D}$ With Respect to f

In order to calculate the Gâteaux derivative of our objective functional with respect to the image, we start with equation B.13:

$$
\begin{aligned}
& d_f \mathcal{D}(f, \boldsymbol{\varphi}; \eta) && (4.7)\\
&= \sum_{t,a} \left[d_f \hat{g}(a,t,f,\boldsymbol{\varphi};\eta) - g(a,t) d_f \log\left(\hat{g}(a,t,f,\boldsymbol{\varphi});\eta\right) \right] && (4.8)\\
&= \sum_{t,a} \left[d_f \hat{g}(a,t,f,\boldsymbol{\varphi}) - g(a,t) \frac{d_f \hat{g}(a,t,f,\boldsymbol{\varphi};\eta)}{\hat{g}(a,t,f,\boldsymbol{\varphi})} \right] && (4.9)\\
&= \sum_{t,a} d_f \hat{g}(a,t,f,\boldsymbol{\varphi}) \left(1 - \frac{g(a,t,f,\boldsymbol{\varphi})}{\hat{g}(a,t,f,\boldsymbol{\varphi})} \right) \ . && (4.10)
\end{aligned}
$$

We now need to know how $d_f \hat{g}(a,t,f,\boldsymbol{\varphi})$ looks like:

$$
\begin{aligned}
& d_f \hat{g}(a,t,f,\boldsymbol{\varphi}) && (4.11)\\
&= \frac{\partial}{\partial \varepsilon} \hat{g}(a,t,f+\varepsilon\eta) \Big|_{\varepsilon=0} && (4.12)\\
&= \frac{1}{T} \int H(a,\boldsymbol{x}) \ \frac{\partial}{\partial \varepsilon} [f+\varepsilon\eta](\boldsymbol{\varphi}(\boldsymbol{x},t)) \Big|_{\varepsilon=0} \, d\boldsymbol{x} && (4.13)\\
&= \frac{1}{T} \int H(a,\boldsymbol{x}) \ \eta(\boldsymbol{\varphi}(\boldsymbol{x},t)) \, d\boldsymbol{x} && (4.14)
\end{aligned}
$$

For reasons that become clear later, it is necessary to isolate $\eta(\boldsymbol{x})$. By substituting the motion function by its inverse (applying the rule for integration by substitution for multiple variables), equation (4.14) can be rewritten as

$$
d_f \hat{g}(a,t,f,\boldsymbol{\varphi}) = \frac{1}{T} \int H(a, \varphi^{-1}(\boldsymbol{x},t)) \, |\det(\nabla \varphi^{-1})(\boldsymbol{x},t)| \, \eta(\boldsymbol{x}) \, d\boldsymbol{x} \ . \tag{4.15}
$$

Together with (4.10) we arrive at

$$
d_f \mathcal{D}(f, \boldsymbol{\varphi}; \eta) = \frac{1}{T} \int \sum_{t,a} H(a, \varphi^{-1}(\boldsymbol{x},t)) \, |\det(\nabla \varphi^{-1})(\boldsymbol{x},t)| \left(1 - \frac{g(a,t,f,\boldsymbol{\varphi})}{\hat{g}(a,t,f,\boldsymbol{\varphi})} \right) \ \eta(\boldsymbol{x}) \, d\boldsymbol{x} \ . \tag{4.16}
$$

4.3.2. Gâteaux Derivative of $\mathcal{D}$ With Respect to φ

The first steps towards the Gâteaux derivative of $\mathcal{D}$ are very similar to (4.7) - (4.10), and we arrive at

$$
d_{\boldsymbol{\varphi}} \mathcal{D}(f, \boldsymbol{\varphi}; \boldsymbol{\eta}) = \sum_{t,a} d_{\boldsymbol{\varphi}} \hat{g}(a,t,f,\boldsymbol{\varphi};\boldsymbol{\eta}) \left(1 - \frac{g(a,t)}{\hat{g}(a,t,f,\boldsymbol{\varphi})} \right) \ . \tag{4.17}
$$

The Gâteaux derivative is written as

$$d_{\boldsymbol{\varphi}}\hat{g}(a,t,f,\boldsymbol{\varphi};\boldsymbol{\eta}) = \frac{\partial}{\partial\varepsilon}\hat{g}(a,t,f,\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta})\bigg|_{\varepsilon=0} \ . \tag{4.18}$$

Expansion leads to

$$d_{\boldsymbol{\varphi}}\hat{g}(a,t,f,\boldsymbol{\varphi};\boldsymbol{\eta}) = \frac{1}{T}\int H(a,\boldsymbol{x}) \ \frac{\partial}{\partial\varepsilon} f([\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta}](\boldsymbol{x},t))\bigg|_{\varepsilon=0} d\boldsymbol{x} \ . \tag{4.19}$$

The chain rule yields

$$d_{\boldsymbol{\varphi}}\hat{g}(a,t,f,\boldsymbol{\varphi};\boldsymbol{\eta}) = \frac{1}{T}\int H(a,\boldsymbol{x}) \, \langle \nabla f(\boldsymbol{\varphi}(\boldsymbol{x},t)), \boldsymbol{\eta}(\boldsymbol{x},t)\rangle \, d\boldsymbol{x} \ . \tag{4.20}$$

Putting (4.20) into (4.17) leads to

$$d_{\boldsymbol{\varphi}}\mathcal{D} = \frac{1}{T}\int \sum_{t,a} H(a,\boldsymbol{x}) \ \left(1 - \frac{g(a,t,f,\boldsymbol{\varphi})}{\hat{g}(a,t,f,\boldsymbol{\varphi})}\right) \langle \nabla f(\boldsymbol{\varphi}(\boldsymbol{x},t)), \boldsymbol{\eta}(\boldsymbol{x},t)\rangle \, d\boldsymbol{x} \ . \tag{4.21}$$

Later we will only need the Gâteaux derivative of $\mathcal{D}$ with respect to one component of $\boldsymbol{\varphi}$, that is $d_{[\boldsymbol{\varphi}]_i}\mathcal{D}(f,\boldsymbol{\varphi};\eta)$. This is simply given by

$$d_{[\boldsymbol{\varphi}]_i}\mathcal{D}(f,\boldsymbol{\varphi};[\boldsymbol{\eta}]_i) = d_{\boldsymbol{\varphi}}\mathcal{D}(f,\boldsymbol{\varphi};\boldsymbol{\eta_i}) \ , \tag{4.22}$$

where $\boldsymbol{\eta_i}$ is defined as a vector of size three for which $[\boldsymbol{\eta_i}]_i = [\boldsymbol{\eta}]_i$ and $[\boldsymbol{\eta_i}]_{j\neq i} = 0$. E. g., $\boldsymbol{\eta_0} = ([\boldsymbol{\eta}]_0, 0, 0)^\mathsf{T}$. Accordingly, equation (4.21) can be alternatively written as

$$d_{[\boldsymbol{\varphi}]_i}\mathcal{D}(f,\boldsymbol{\varphi};\eta) = \frac{1}{T}\int \sum_{t,a} H(a,\boldsymbol{x}) \ \left(1 - \frac{g(a,t,f,\boldsymbol{\varphi})}{\hat{g}(a,t,f,\boldsymbol{\varphi})}\right) \frac{\partial}{\partial_i} f(\boldsymbol{\varphi}(\boldsymbol{x},t)) \, \eta(\boldsymbol{x},t) \, d\boldsymbol{x} \ . \tag{4.23}$$

4.4. Discretization

Since our joint reconstruction procedure will finally be executed on a digital computer, we need to approximate the continuous functions by a formulation which is based on discrete parameters. In the following, we will introduce a discrete representation for the image, the motion function and the system response function. Then, based on these discrete representations and the previously derived continuous Gâteaux derivatives, the discrete gradients are calculated. These discrete gradients will finally be used in order to define the concrete update equations for both image and motion.

4.4.1. Image

We will model the image as a weighted sum of voxel basis functions $\boldsymbol{v}$ (from now on simply called voxels):

$$f(\boldsymbol{x}) \approx \sum_{k}^{N^{\boldsymbol{f}}} [\boldsymbol{f}]_k [\boldsymbol{v}]_k(\boldsymbol{x}) \ . \tag{4.24}$$

$[\boldsymbol{f}]_k$ is the value of voxel k. Voxels are cuboids of the same size and are aligned in a non-overlapping manner on a regular grid. The voxels are numbered consecutively in lexicographical order (that is, first in x-direction, then y and then z). There are $N^{\boldsymbol{f}}$ total voxels ($N^{\boldsymbol{f}} = N_x^{\boldsymbol{f}} N_y^{\boldsymbol{f}} N_z^{\boldsymbol{f}}$).

4.4.2. Motion

We model the motion function as a weighted sum of basis functions $\boldsymbol{b}$, governed by an $N^{\boldsymbol{\Phi}} \times 3$ matrix of control points $\boldsymbol{\Phi}$:

$$\boldsymbol{\varphi}(\boldsymbol{x}, t) = \boldsymbol{x} + \sum_{i=1}^{N^{\boldsymbol{\Phi}}} \begin{pmatrix} [\boldsymbol{\Phi}]_{i1} \\ [\boldsymbol{\Phi}]_{i2} \\ [\boldsymbol{\Phi}]_{i3} \end{pmatrix} [\boldsymbol{b}]_i(\boldsymbol{x}, t) \ . \tag{4.25}$$

The grid of control points is of dimensions $N_x^{\boldsymbol{\Phi}} \times N_y^{\boldsymbol{\Phi}} \times N_z^{\boldsymbol{\Phi}} \times N_t^{\boldsymbol{\Phi}}$. An entry $[\boldsymbol{\Phi}]_{ij}$ of matrix $\boldsymbol{\Phi}$ corresponds to basis function $[\boldsymbol{b}]_i$ and governs its influence in the j-th dimension. We further define a basis function $[\boldsymbol{b}]_i$ to be a product of four B-splines:

$$[\boldsymbol{b}]_i(\boldsymbol{x}, t) = b_{i1}([\boldsymbol{x}]_1) b_{i2}([\boldsymbol{x}]_2) b_{i3}([\boldsymbol{x}]_3) b_{i4}(t) \ . \tag{4.26}$$

For the forthcoming discretization of the derivatives, the degree of the B-spline basis function does not matter. We will later try different variations, including also different degrees in different dimensions (the time domain will differ from the spatial domain). For more information about B-splines and its application in image deformations, the reader is referred to [176].

Another motion model we explore in this thesis are displacement fields: in a displacement field, each voxel has a displacement vector which defines its position in the new frame. We will later compare a displacement field motion model to a B-spline based motion model. Note that the B-splines model is general enough to capture displacement fields by using B-splines of degree zero together with an appropriate number of control points in each dimension. Accordingly, we can make all derivations based on the B-splines model and later choose the degree and number of control points in order to effectively have a displacement field.

4.4.3. System Response Function

Similar to the image, the system response function $H(a, \boldsymbol{x})$ will be discretized as a weighted sum of voxels:

$$H(a, \boldsymbol{x}) \approx \sum_{k}^{N^{\boldsymbol{f}}} [\boldsymbol{H}]_{ak} [\boldsymbol{v}]_k(\boldsymbol{x}) \ . \tag{4.27}$$

4.4.4. Discrete Image Derivative

Since the image and motion functions are now discretized, the functional $\mathcal{D}$ does not depend on the continuous image and motion functions anymore but rather depends on

the discrete parameters $\boldsymbol{f}$ and $\boldsymbol{\Phi}$. We define

$$D(\boldsymbol{f}, \boldsymbol{\Phi}) := \mathcal{D}(f_{\boldsymbol{f}}, \boldsymbol{\varphi}_{\boldsymbol{\Phi}}) \tag{4.28}$$

in order to make clear that D depends on the discrete image and motion parameters.

In order to calculate the discrete derivatives, fortunately we can reuse the previously derived Gâteaux derivatives. We first note that any scalar derivative can be written as a Gâteaux derivative, simply since any scalar variable can be considered as a constant function. Accordingly, the scalar derivative of D with respect to $[\boldsymbol{f}]_i$ can be written as

$$\frac{\partial}{\partial [\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}) = d_{[\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}; \eta) \ . \tag{4.29}$$

Note that the direction η in this case is obsolete since scalars do not have directions.

Now we introduce the chain rule for Gâteaux derivatives:

$$d_b \mathcal{A}(\mathcal{B}(b); \eta) = d_a \mathcal{A}(\mathcal{B}(b); d_b \mathcal{B}(b; \eta)) \ . \tag{4.30}$$

Transferred to our case this means that

$$\begin{aligned}
&\frac{\partial}{\partial [\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}) && (4.31)\\
&\quad = d_{[\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}; \eta) && (4.32)\\
&\quad = d_{[\boldsymbol{f}]_i} \mathcal{D}(f_{\boldsymbol{f}}, \boldsymbol{\varphi}_{\boldsymbol{\Phi}}; \eta) && (4.33)\\
&\quad = d_f \mathcal{D}(f_{\boldsymbol{f}}, \boldsymbol{\varphi}_{\boldsymbol{\Phi}}; d_{[\boldsymbol{f}]_i} f(\boldsymbol{x}, \boldsymbol{f}; \eta)) && (4.34)\\
&\quad = d_f \mathcal{D}\left(f_{\boldsymbol{f}}, \boldsymbol{\varphi}_{\boldsymbol{\Phi}}; \frac{\partial f}{\partial [\boldsymbol{f}]_i}\right) \ . && (4.35)
\end{aligned}$$

Since $\frac{\partial f}{\partial [\boldsymbol{f}]_i} = [\boldsymbol{v}]_i$, we arrive at

$$\frac{\partial}{\partial [\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}) = \frac{1}{T} \int \sum_{t,a} H(a, \varphi^{-1}(\boldsymbol{x}, t)) \, |\det(\nabla \varphi^{-1})(\boldsymbol{x}, t)| \left(1 - \frac{g(a,t)}{\hat{g}(a,t)}\right) [\boldsymbol{v}]_i(\boldsymbol{x}) \, d\boldsymbol{x} \ . \tag{4.36}$$

We replace the continuous system model by its weighted sum of basis functions:

$$\begin{aligned}
\frac{\partial}{\partial [\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx & \frac{1}{T} \int \sum_{t,a} \left(\sum_j [\boldsymbol{H}]_{aj} [\boldsymbol{v}]_j (\varphi^{-1}(\boldsymbol{x}, t)) \right) \\
& |\det(\nabla \varphi^{-1})(\boldsymbol{x}, t)| \left(1 - \frac{g(a,t)}{\hat{g}(a,t)}\right) [\boldsymbol{v}]_i(\boldsymbol{x}) \, d\boldsymbol{x} \ .
\end{aligned} \tag{4.37}$$

Reordering of the sums leads to

$$\frac{\partial}{\partial[\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{1}{T} \int [\boldsymbol{v}]_i(\boldsymbol{x}) \sum_t |\det(\nabla \varphi^{-1})(\boldsymbol{x}, t)| \sum_j [\boldsymbol{v}]_j(\varphi^{-1}(\boldsymbol{x}, t)) \sum_a [\boldsymbol{H}]_{aj} \left(1 - \frac{g(a,t)}{\hat{g}(a,t)}\right) d\boldsymbol{x} \ . \tag{4.38}$$

The rightmost sum can be expressed in a more compact manner using matrix-vector notation

$$\sum_a [\boldsymbol{H}]_{aj} \left(1 - \frac{g(a,t)}{\hat{g}(a,t)}\right) = \left[\boldsymbol{H}^\mathsf{T} \left(\mathbf{1} - \frac{\boldsymbol{g_t}}{\hat{\boldsymbol{g}}_t}\right)\right]_j \ , \tag{4.39}$$

where $[\boldsymbol{g_t}]_a := g(a,t)$ and $[\hat{\boldsymbol{g}}_t]_a := \hat{g}(a,t)$. By inserting (4.39) in (4.38), we get

$$\frac{\partial}{\partial[\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{1}{T} \int [\boldsymbol{v}]_i(\boldsymbol{x}) \sum_t |\det(\nabla \varphi^{-1})(\boldsymbol{x}, t)| \sum_j \left[\boldsymbol{H}^\mathsf{T} \left(\mathbf{1} - \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t}\right)\right]_j [\boldsymbol{v}]_j(\boldsymbol{\varphi}^{-1}(\boldsymbol{x}, t))\, d\boldsymbol{x} \ . \tag{4.40}$$

In the next step, we will express the absolute value of the Jacobian in terms of the voxel basis $\boldsymbol{v}$ with coefficients $\boldsymbol{j}$. Note that this is for the moment a purely notational operation, which in practice is accomplished by a numerical approximation.

$$\frac{\partial}{\partial[\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{1}{T} \int [\boldsymbol{v}]_i(\boldsymbol{x}) \sum_t \left(\sum_j [\boldsymbol{j}]_j [\boldsymbol{v}]_j(\boldsymbol{x})\right) \sum_j \left[\boldsymbol{H}^\mathsf{T} \left(\mathbf{1} - \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t}\right)\right]_j [\boldsymbol{v}]_j(\boldsymbol{\varphi}^{-1}(\boldsymbol{x}, t))\, d\boldsymbol{x} \ . \tag{4.41}$$

At this point, the term $[\boldsymbol{v}]_j(\varphi^{-1}(\boldsymbol{x}, t))$ is problematic, since the motion function is within the basis function and we cannot further simplify. In order to resolve this issue, we perform a basis transformation:

$$\sum_j \left[\boldsymbol{H}^\mathsf{T} \left(\mathbf{1} - \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t}\right)\right]_j [\boldsymbol{v}]_j(\boldsymbol{\varphi}^{-1}(\boldsymbol{x}, t)) \approx \sum_j \left[\boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \left(\mathbf{1} - \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t}\right)\right]_j [\boldsymbol{v}]_j(\boldsymbol{x}) \ , \tag{4.42}$$

where $\boldsymbol{T}_t^{-1}$ is a $N^{\boldsymbol{f}} \times N^{\boldsymbol{f}}$ matrix which encapsulates the transformation $\boldsymbol{\varphi}^{-1}(\boldsymbol{x}, t)$. Mathematically speaking, $\boldsymbol{T}_t^{-1}$ is a linear operator that transforms the coefficients such that they fit approximately to the new basis. Intuitively, one has to see that $\boldsymbol{H}^\mathsf{T}(\mathbf{1} - \boldsymbol{g}_t/\hat{\boldsymbol{g}}_t)$ is a quantity in image space, and instead of transforming the voxels their values can be modified accordingly. So, from an implementation point of view, the multiplication with

$\boldsymbol{T}_t^{-1}$ is a deformation of an image (which is usually not implemented as a matrix vector multiplication). Plugging (4.42) into (4.41), we arrive at

$$\frac{\partial}{\partial [\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{1}{T} \int [\boldsymbol{v}]_i(\boldsymbol{x}) \sum_t \left(\sum_j [\boldsymbol{j}]_j [\boldsymbol{v}]_j(\boldsymbol{x}) \right) \sum_j \left[\boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \left(\mathbf{1} - \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t} \right) \right]_j [\boldsymbol{v}]_j(\boldsymbol{x}) \, d\boldsymbol{x} \ . \tag{4.43}$$

Now, note that $[\boldsymbol{v}]_i(\boldsymbol{x})$ is only one within voxel i and zero outside. The sums based on the basis $[\boldsymbol{v}]_j$ are thus zero except when $j = i$. Accordingly, we get

$$\frac{\partial}{\partial [\boldsymbol{f}]_i} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{1}{T} \sum_t [\boldsymbol{j}]_i \left[\boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \left(\mathbf{1} - \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t} \right) \right]_i \int [\boldsymbol{v}]_i(\boldsymbol{x}) \, d\boldsymbol{x} \ . \tag{4.44}$$

All voxels have the same volume

$$V := \int [\boldsymbol{v}]_i(\boldsymbol{x}) \, d\boldsymbol{x} \qquad \forall i \ . \tag{4.45}$$

Finally, the image gradient, which is a vector of all partial derivatives with respect to the image parameters $\boldsymbol{f}$, can be compactly written as

$$\nabla_{\boldsymbol{f}} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{V}{T} \sum_t \boldsymbol{j_t} \boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \left(\mathbf{1} - \frac{\boldsymbol{g_t}}{\hat{\boldsymbol{g}}_{\boldsymbol{t}}} \right) \ . \tag{4.46}$$

The estimated measurement is given by

$$[\hat{\boldsymbol{g}}_{\boldsymbol{t}}]_a = \hat{g}(a, t) = \frac{1}{T} \int H(a, \boldsymbol{x}) f(\boldsymbol{\varphi}(\boldsymbol{x}, t)) \, d\boldsymbol{x} \ . \tag{4.47}$$

Again, replacing the continuous functions by their respective weighted sums of basis functions leads to the approximation

$$[\hat{\boldsymbol{g}}_{\boldsymbol{t}}]_a \approx \frac{1}{T} \int \left(\sum_j [\boldsymbol{H}]_{aj} [\boldsymbol{v}]_j(\boldsymbol{x}) \right) \left(\sum_j [\boldsymbol{f}]_j [\boldsymbol{v}]_j(\boldsymbol{\varphi}(\boldsymbol{x}, t)) \right) \, d\boldsymbol{x} \ . \tag{4.48}$$

As before, we do a basis transformation such that we get the motion function out of the voxel (another approximation):

$$[\hat{\boldsymbol{g}}_{\boldsymbol{t}}]_a \approx \frac{1}{T} \int \left(\sum_j [\boldsymbol{H}]_{aj} [\boldsymbol{v}]_j(\boldsymbol{x}) \right) \left(\sum_j \left[\boldsymbol{T}_t \boldsymbol{f} \right]_j [\boldsymbol{v}]_j(\boldsymbol{x}) \right) \, d\boldsymbol{x} \ . \tag{4.49}$$

Due to distributivity we get:

$$[\hat{\boldsymbol{g}}_{\boldsymbol{t}}]_a \approx \frac{1}{T} \int \sum_j \sum_k [\boldsymbol{H}]_{aj} [\boldsymbol{v}]_j(\boldsymbol{x}) [\boldsymbol{T}_t \boldsymbol{f}]_k [\boldsymbol{v}]_k(\boldsymbol{x}) \, d\boldsymbol{x} \tag{4.50}$$

Again, the product of $[\boldsymbol{v}]_j$ and $[\boldsymbol{v}]_k$ can only be non-zero if $k = j$ since voxels are not overlapping. Accordingly, we get

$$[\hat{\boldsymbol{g}}_t]_a \approx \frac{1}{T} \int \sum_j [\boldsymbol{H}]_{aj} [\boldsymbol{T}_t \boldsymbol{f}]_j [\boldsymbol{v}]_j(\boldsymbol{x}) \, d\boldsymbol{x} \ , \tag{4.51}$$

which is equal to

$$\hat{\boldsymbol{g}}_t \approx \frac{V}{T} \boldsymbol{H} \boldsymbol{T}_t \boldsymbol{f} \tag{4.52}$$

written in matrix-vector form.

4.4.5. Discrete Motion Derivative

Similar to the relation between the discrete derivative with respect to an image parameter and the Gâteaux derivative of the image, in case of the motion we have

$$\frac{\partial}{\partial [\boldsymbol{\Phi}]_{ij}} D(\boldsymbol{f}, \boldsymbol{\Phi}) = d_{\boldsymbol{\varphi}} \mathcal{D} \left(f, \boldsymbol{\varphi}; \frac{\partial \boldsymbol{\varphi}}{\partial [\boldsymbol{\Phi}]_{ij}} \right) \ . \tag{4.53}$$

From the definition of the discretized motion function in (4.25), the derivative with respect to the motion parameters is

$$\begin{aligned} &\frac{\partial}{\partial [\boldsymbol{\Phi}]_{ij}} [\boldsymbol{\varphi}]_j(\boldsymbol{x}, t) \\ &\qquad = \frac{\partial}{\partial [\boldsymbol{\Phi}]_{ij}} \sum_{k=1}^{N^{\boldsymbol{\Phi}}} [\boldsymbol{\Phi}]_{kj} [\boldsymbol{b}]_k(\boldsymbol{x}, t) &(4.54) \\ &\qquad = [\boldsymbol{b}]_i(\boldsymbol{x}, t) \ . &(4.55) \end{aligned}$$

$[\boldsymbol{b}]_i(\boldsymbol{x}, t)$ is inserted in the Gâteaux derivative from (4.23):

$$\frac{\partial}{\partial [\boldsymbol{\Phi}]_{ij}} D(\boldsymbol{f}, \boldsymbol{\Phi}) = \frac{1}{T} \int \sum_{t,a} H(a, \boldsymbol{x}) \, \frac{\partial}{\partial_j} f(\boldsymbol{\varphi}(\boldsymbol{x}, t)) \left(1 - \frac{g(a,t)}{\hat{g}(a,t)} \right) [\boldsymbol{b}]_i(\boldsymbol{x}, t) \, d\boldsymbol{x} \ . \tag{4.56}$$

Substituting the discrete approximation of H and f:

$$\begin{aligned} \frac{\partial}{\partial [\boldsymbol{\Phi}]_{ij}} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx & \frac{1}{T} \int \sum_{t,a} \left(\sum_k [\boldsymbol{H}]_{ak} [\boldsymbol{v}]_k(\boldsymbol{x}) \right) \left(\sum_k [\boldsymbol{f}]_k \frac{\partial}{\partial_j} [\boldsymbol{v}]_k(\boldsymbol{\varphi}(\boldsymbol{x}, t)) \right) \\ & \left(1 - \frac{g(a,t)}{\hat{g}(a,t)} \right) [\boldsymbol{b}]_i(\boldsymbol{x}, t) \, d\boldsymbol{x} \ . \end{aligned} \tag{4.57}$$

At this point, as usual, we have to express all factors in terms of the same basis in order to make further simplifications. This time, instead of the basis function, its derivative is

deformed by a transformation. Note that the derivative of a voxel does not even exists at its borders.

We solve this problem by applying two transformations. The first one transforms the coefficients $\boldsymbol{f}$ such that the derivative with respect to index j is discretely approximated. It will be denoted by the matrix $\boldsymbol{G_j}$ ("G" for gradient). The second one is a basis transformation (as seen before in the image gradient discretization) which encapsulates the transformation. This matrix is denoted by $\boldsymbol{T_t}$ ("T" for transformation).

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ij}} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{1}{T} \int \sum_{t,a} \left(\sum_k [\boldsymbol{H}]_{ak} [\boldsymbol{v}]_k(\boldsymbol{x}) \right) \left(\sum_k \left[\boldsymbol{T}_t \boldsymbol{G}_j \boldsymbol{f} \right]_k [\boldsymbol{v}]_k(\boldsymbol{x}) \right) \left(1 - \frac{g(a,t)}{\hat{g}(a,t)} \right) [\boldsymbol{b}]_i(\boldsymbol{x}, t) \, d\boldsymbol{x} \ . \tag{4.58}$$

Now, the B-Spline basis will also be expressed in terms of a voxel basis:

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ij}} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{1}{T} \int \sum_{t,a} \left(\sum_k [\boldsymbol{H}]_{ak} [\boldsymbol{v}]_k(\boldsymbol{x}) \right) \left(\sum_k \left[\boldsymbol{T}_t \boldsymbol{G}_j \boldsymbol{f} \right]_k [\boldsymbol{v}]_k(\boldsymbol{x}) \right) \left(1 - \frac{g(a,t)}{\hat{g}(a,t)} \right) \left(\sum_k [\boldsymbol{s}_{it}]_k [\boldsymbol{v}]_k(\boldsymbol{x}) \right) \, d\boldsymbol{x} \ . \tag{4.59}$$

Rearranging the sums with the distributive law leads to

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ij}} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{1}{T} \int \sum_{t,a} \left(1 - \frac{g(a,t)}{\hat{g}(a,t)} \right) \sum_{k,l,m} [\boldsymbol{H}]_{ak} [\boldsymbol{v}]_k(\boldsymbol{x}) \ \left[\boldsymbol{T}_t \boldsymbol{G}_j \boldsymbol{f} \right]_l [\boldsymbol{v}]_l(\boldsymbol{x}) \ [\boldsymbol{s}_{it}]_m [\boldsymbol{v}]_m(\boldsymbol{x}) \, d\boldsymbol{x} \ . \tag{4.60}$$

Remember that voxel basis functions do not overlap and are equal to one within their support. If $[\boldsymbol{v}]_k(\boldsymbol{x}) = 0$, then the whole term is zero. If $[\boldsymbol{v}]_k(\boldsymbol{x}) = 1$, then the whole term can only be non-zero if $k = l = m$. Therefore, we get

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ij}} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{1}{T} \int \sum_{t,a} \left(1 - \frac{g(a,t)}{\hat{g}(a,t)} \right) \sum_k [\boldsymbol{H}]_{ak} \left[\boldsymbol{T}_t \boldsymbol{G}_j \boldsymbol{f} \right]_k [\boldsymbol{s}_{it}]_k [\boldsymbol{v}]_k(\boldsymbol{x}) \, d\boldsymbol{x} \ . \tag{4.61}$$

In matrix-vector notation we finally arrive at

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ij}} D(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{V}{T} \sum_t \left(\boldsymbol{H} \boldsymbol{T}_t \boldsymbol{G}_j \boldsymbol{f} \boldsymbol{s}_{it} \right)^\mathsf{T} \left(\boldsymbol{1} - \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t} \right) \ . \tag{4.62}$$

We use Chen's method [37] (with five iterations) in order to calculate the inverse transformation which is needed for the image update. Chen's method is extremely simple to

implement (and derive) displacement field inversion algorithm based on a fixed point iteration. Since it works only for displacement fields, the B-spline representation has to be transformed to a displacement field before inversion. Inversion may lead to unpredictable results if the motion function is not invertible. Even though Chen's method is stable against such issues, it is recommendable to ensure that the motion field is invertible at any time in the reconstruction algorithm. Appropriate regularization can take care of this (e. g. by choosing a high enough parameter α, see also section 5.1.3.4 and especially Figure 5.4).

4.5. The Update Equations

In this section, the two main update equations

$$\boldsymbol{f}^{\text{new}} = \tau_f(\boldsymbol{f}, \boldsymbol{\Phi}) \tag{4.63}$$
$$\boldsymbol{\Phi}^{\text{new}} = \tau_{\boldsymbol{\varphi}}(\boldsymbol{f}, \boldsymbol{\Phi}) \tag{4.64}$$

are derived. Equation (4.63) defines the image update and (4.64) defines the motion update.

Note that both update equations derived here will not include any regularization of the problem. The regularization issue will be addressed in section 4.6.

4.5.1. Image Update

We will develop the image update based on the previous discretization of the image and the gradient. The goal is to find a vector $\boldsymbol{f}$ that minimizes $D(\boldsymbol{f}, \boldsymbol{\Phi})$ subject to the positivity constraint $[\boldsymbol{c}]_i(\boldsymbol{f}) >= 0$, with

$$[\boldsymbol{c}]_i(\boldsymbol{f}) := [\boldsymbol{f}]_i \ . \tag{4.65}$$

4.5.1.1. KKT Conditions

If $\boldsymbol{f}$ is a (local) minimum, then there exist KKT multipliers $\boldsymbol{\mu}$ (encoded in a vector) such that the KKT conditions which are necessary conditions for a local minimum are fulfilled:

$$\nabla_{\boldsymbol{f}} D(\boldsymbol{f}, \boldsymbol{\Phi}) - \sum_i [\boldsymbol{\mu}]_i \nabla_{\boldsymbol{f}} [\boldsymbol{c}]_i(\boldsymbol{f}) = \boldsymbol{0} \tag{4.66}$$
$$[\boldsymbol{\mu}]_i \geq \boldsymbol{0} \ \forall i \tag{4.67}$$
$$[\boldsymbol{\mu}]_i [\boldsymbol{c}]_i(\boldsymbol{f}) = \boldsymbol{0} \ \forall i \ . \tag{4.68}$$

The term $\sum_i [\boldsymbol{\mu}]_i \nabla_{\boldsymbol{f}} [\boldsymbol{c}]_i(\boldsymbol{f})$ can be simplified according to

$$\sum_i [\boldsymbol{\mu}]_i \nabla_{\boldsymbol{f}} [\boldsymbol{c}]_i(\boldsymbol{f}) = \sum_i [\boldsymbol{\mu}]_i \nabla_{\boldsymbol{f}} [\boldsymbol{f}]_i = \boldsymbol{\mu} \ . \tag{4.69}$$

Then, equation (4.66) rewrites to

$$\nabla_{\boldsymbol{f}} D(\boldsymbol{f}, \boldsymbol{\Phi}) = \boldsymbol{\mu} \ . \tag{4.70}$$

With (4.70), the KKT conditions can be simplified to the following two equations:

$$\nabla_{\boldsymbol{f}} D(\boldsymbol{f}, \boldsymbol{\Phi}) \geq \mathbf{0} \tag{4.71}$$
$$\nabla_{\boldsymbol{f}} D(\boldsymbol{f}, \boldsymbol{\Phi}) \boldsymbol{f} = \mathbf{0} \ . \tag{4.72}$$

We will later use the second condition to define a fixed-point iterative scheme which serves as the image update sequence given a known motion.

4.5.1.2. The Update Equation

In the following, we will use the KKT condition from equation (4.72) in order to define the image update equation.

$$\nabla_{\boldsymbol{f}} D(\boldsymbol{f}, \boldsymbol{\Phi}) \boldsymbol{f} = \mathbf{0} \tag{4.73}$$
$$\Leftrightarrow \boldsymbol{f} \frac{V}{T} \sum_t \boldsymbol{j}_t \boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \left(\mathbf{1} - \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t} \right) = 0 \tag{4.74}$$
$$\Leftrightarrow \boldsymbol{f} \sum_t \boldsymbol{j}_t \boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \mathbf{1} = \boldsymbol{f} \sum_t \boldsymbol{j}_t \boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t} \tag{4.75}$$
$$\Leftrightarrow \boldsymbol{f} = \boldsymbol{f} \frac{\sum_t \boldsymbol{j}_t \boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t}}{\sum_t \boldsymbol{j}_t \boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \mathbf{1}} \tag{4.76}$$

The image update equation is then defined as

$$\boldsymbol{f}^{\text{new}}(\boldsymbol{x}) = \tau_f(\boldsymbol{f}, \boldsymbol{\Phi}) \tag{4.77}$$

with

$$\tau_f(\boldsymbol{f}, \boldsymbol{\Phi}) := \boldsymbol{f} \frac{\sum_t \boldsymbol{j}_t \boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t}}{\sum_t \boldsymbol{j}_t \boldsymbol{T}_t^{-1} \boldsymbol{H}^\mathsf{T} \mathbf{1}} \ . \tag{4.78}$$

4.5.2. Motion Update

The motion update equation

$$\boldsymbol{\Phi}^{\text{new}} = \tau_{\boldsymbol{\varphi}}(\boldsymbol{f}, \boldsymbol{\Phi}) \tag{4.79}$$

will be simply defined as a gradient descent scheme. For minimization, we have to add the negative gradient. Additionally, we will use a step-size δ in order to achieve faster convergence.

$$[\tau_{\boldsymbol{\varphi}}(\boldsymbol{f},\boldsymbol{\Phi})]_{ij} := [\boldsymbol{\Phi}]_{ij} - \delta \frac{\partial}{\partial [\boldsymbol{\Phi}]_{ij}} D(\boldsymbol{f},\boldsymbol{\Phi}) \tag{4.80}$$

$$= [\boldsymbol{\Phi}]_{ij} - \delta \frac{V}{T} \sum_t (\boldsymbol{H}\boldsymbol{T}_t\boldsymbol{G}_j\boldsymbol{f}\boldsymbol{s}_{it})^{\mathsf{T}} \left(\mathbf{1} - \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t}\right) \quad . \tag{4.81}$$

4.6. Regularization

Image reconstruction is an ill-posed inverse problem and as a such demands regularization. Principally, regularization is necessary for all parameters subject to estimation - in our case image and motion. According to our experience, the motion regularization is much more important than the image regularization, and we will therefore put a special emphasis on motion regularization.

A possible way to impose regularization is to modify the objective functional. For example, if we were looking for smooth motion, we could add a term to the objective function which is high for non-smooth motion and low for smooth motion. In this way, by optimizing the objective function, smooth motion would be encouraged. Another way to impose regularization, especially appealing in iterative algorithms, is to enforce or encourage a certain constraint on the parameters after each iteration.

In this thesis we have employed both alternatives. For the image, we apply regularization after each image update, and for the motion we explored both alternatives.

One should note that regularization can have many flavors. Already the choice of the model (e. g. the choice of a B-spline motion model together with the number of nodes) is a kind of regularization. Also the choice of the number of iterations for an iterative algorithm is regularization, e. g., for the ML-EM algorithm it is known that stopping before convergence leads to smoother images. For images e. g., blobs basis functions have been used successfully in order to limit noise in the literature [124, 135, 136, 4, 83, 29].

4.6.1. Image Regularization

Many image regularization measures have been proposed. In order to regularize the image, we perform a Gaussian smoothing after each image update. Only little smoothing is applied for each update since it is performed many consecutive times.

4.6.2. Motion Regularization

As mentioned, we have explored two ways of regularization: the regularization modeled within the objective function (discussed in section 4.6.2.1) and after each image update (refer to section 4.6.2.2).

4.6.2.1. Regularization in Objective Functional

We modify the objective functional by adding a term $\mathcal{S}$ which only depends on the motion function φ:

$$\mathcal{J}(f,\varphi) = (1-\alpha)\ \mathcal{D}(f,\varphi) + \alpha\ \mathcal{S}(\varphi)\ . \tag{4.82}$$

$\mathcal{S}$ is defined as follows:

$$\mathcal{S}(\boldsymbol{\varphi}) = \sum_t \sum_{i=1}^{3} \int \|\nabla_{\boldsymbol{x}} [\boldsymbol{\varphi}]_i(\boldsymbol{x},t)\|^2\ d\boldsymbol{x}\ . \tag{4.83}$$

The norm of each gradient of each $[\boldsymbol{\varphi}]_i$ is valued to the power of two. In this way, large gradients lead to a higher objective functional and therefore smaller gradients, that is, smoother motion, are preferred.

Note that by adding this smoothing term the image update is not affected at all, since it only needs gradients with respect to f for which the motion term simply falls away:

$$\begin{aligned} & d_f \mathcal{J}(f,\boldsymbol{\varphi}) && (4.84)\\ & \quad = d_f \mathcal{D}(f,\boldsymbol{\varphi}) + d_f \mathcal{S}(\boldsymbol{\varphi}) && (4.85)\\ & \quad = d_f \mathcal{D}(f,\boldsymbol{\varphi})\ . && (4.86) \end{aligned}$$

For the motion derivatives however we get

$$d_{\boldsymbol{\varphi}} \mathcal{J}(f,\boldsymbol{\varphi}) = d_{\boldsymbol{\varphi}} \mathcal{D}(f,\boldsymbol{\varphi}) + \alpha\ d_{\boldsymbol{\varphi}} \mathcal{S}(\boldsymbol{\varphi})\ , \tag{4.87}$$

and so we need to know the Gâteaux derivative of $\mathcal{S}$ for the motion update.

4.6.2.1.1. Gâteaux Derivative of $\mathcal{S}$ We start as usual:

$$\begin{aligned} & d_{[\boldsymbol{\varphi}]_i} \mathcal{S}(\boldsymbol{\varphi}) && (4.88)\\ & \quad = \frac{\partial}{\partial \varepsilon} \mathcal{S}(\ldots [\boldsymbol{\varphi}]_i + \varepsilon\eta \ldots)\Big|_{\varepsilon=0} && (4.89)\\ & \quad = \sum_{t,j} \int_\Omega \frac{\partial}{\partial \varepsilon} \left(\frac{\partial}{\partial x_j} ([\boldsymbol{\varphi}]_i + \varepsilon\eta)(\boldsymbol{x},t) \right)^2 \Bigg|_{\varepsilon=0} d\boldsymbol{x} && (4.90)\\ & \quad = \sum_{t,j} \int_\Omega \frac{\partial}{\partial x_j} [\boldsymbol{\varphi}]_i(\boldsymbol{x},t) \frac{\partial}{\partial x_j} \eta(\boldsymbol{x},t)\ d\boldsymbol{x}\ . && (4.91) \end{aligned}$$

With integration by parts we arrive at

$$d_{[\boldsymbol{\varphi}]_i} \mathcal{S}(\boldsymbol{\varphi}) = \sum_{t,j} \left[\frac{\partial}{\partial x_j} [\boldsymbol{\varphi}]_i(\boldsymbol{x},t) \eta(\boldsymbol{x},t) \right]_\Omega - \int_\Omega \frac{\partial^2}{\partial^2 x_j} [\boldsymbol{\varphi}]_i(\boldsymbol{x},t) \eta(\boldsymbol{x},t)\ d\boldsymbol{x}\ . \tag{4.92}$$

We require the variation η to be zero on the boundaries of Ω, and so we arrive at

$$d_{[\boldsymbol{\varphi}]_i} \mathcal{S}(\boldsymbol{\varphi}) = -\int_\Omega \sum_{t,j} \frac{\partial^2}{\partial^2 x_j} [\boldsymbol{\varphi}]_i(\boldsymbol{x},t)\ \eta(\boldsymbol{x},t)\ d\boldsymbol{x}\ . \tag{4.93}$$

4.6.2.1.2. Discretization

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ki}} S(\boldsymbol{\Phi}) = d_{[\boldsymbol{\varphi}]_i} \mathcal{S}\left(f, \boldsymbol{\varphi}; \frac{\partial[\boldsymbol{\varphi}]_i}{\partial[\boldsymbol{\Phi}]_{ki}}\right) \ . \tag{4.94}$$

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ki}} S(\boldsymbol{f}, \boldsymbol{\Phi}) = -\int_\Omega \sum_{t,j} \frac{\partial^2}{\partial^2 x_j}[\boldsymbol{\varphi}]_i(\boldsymbol{x}, t)\ \frac{\partial}{\partial[\boldsymbol{\Phi}]_{ki}}[\boldsymbol{\varphi}]_i(\boldsymbol{x}, t)\ d\boldsymbol{x} \ . \tag{4.95}$$

Together with equation (4.55) we get

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ki}} S(\boldsymbol{f}, \boldsymbol{\Phi}) = -\int_\Omega \sum_{t,j} \frac{\partial^2}{\partial^2 x_j}[\boldsymbol{\varphi}]_i(\boldsymbol{x}, t)\ [\boldsymbol{b}]_k(\boldsymbol{x}, t)\ d\boldsymbol{x} \ . \tag{4.96}$$

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ki}} S(\boldsymbol{f}, \boldsymbol{\Phi}) = -\int_\Omega \sum_{t,j} \frac{\partial^2}{\partial^2 x_j}\left(\sum_l [\boldsymbol{\Phi}]_{li}[\boldsymbol{b}]_l(\boldsymbol{x}, t)\right)\ [\boldsymbol{b}]_k(\boldsymbol{x}, t)\ d\boldsymbol{x} \ . \tag{4.97}$$

At this point, in order to facilitate the integration numerically, we will convert the B-splines to a voxel basis with new coefficients $\tilde{\boldsymbol{\Phi}}$. Note that usually the voxels will cover a much smaller volume than the B-splines, which makes them suitable for approximation. We write

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ki}} S(\boldsymbol{f}, \boldsymbol{\Phi}) \approx -\int_\Omega \sum_{t,j} \frac{\partial^2}{\partial^2 x_j}\left(\sum_l \left[\tilde{\boldsymbol{\Phi}}_{it}\right]_l [\boldsymbol{v}]_l(\boldsymbol{x})\right)\ \sum_m [\boldsymbol{s}_{kt}]_m\ [\boldsymbol{v}]_m(\boldsymbol{x})\ d\boldsymbol{x} \ . \tag{4.98}$$

The partial differential operator $\frac{\partial^2}{\partial^2 x_j}$ will be discretized by a matrix $\boldsymbol{G}_j^2$:

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ki}} S(\boldsymbol{f}, \boldsymbol{\Phi}) \approx -\int_\Omega \sum_{t,j,l,m} \left[\boldsymbol{G}_j^2 \boldsymbol{d}_{it}\right]_l [\boldsymbol{v}]_l(\boldsymbol{x})\ [\boldsymbol{s}_{kt}]_m\ [\boldsymbol{v}]_m(\boldsymbol{x})\ d\boldsymbol{x} \ . \tag{4.99}$$

Since $[\boldsymbol{v}]_l(\boldsymbol{x})[\boldsymbol{v}]_m(\boldsymbol{x})$ can only be non-zero for $m = l$ we get

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ki}} S(\boldsymbol{f}, \boldsymbol{\Phi}) \approx -\int_\Omega \sum_{t,j,l} \left[\boldsymbol{G}_j^2 \boldsymbol{d}_{it}\right]_l [\boldsymbol{s}_{kt}]_l\ [\boldsymbol{v}]_l(\boldsymbol{x})\ d\boldsymbol{x} \ . \tag{4.100}$$

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ki}} S(\boldsymbol{f}, \boldsymbol{\Phi}) \approx -V \sum_{t,j,l} \left[\boldsymbol{G}_j^2 \boldsymbol{d}_{it}\right]_l [\boldsymbol{s}_{kt}]_l \ . \tag{4.101}$$

In matrix-vector form this becomes

$$\frac{\partial}{\partial[\boldsymbol{\Phi}]_{ki}} S(\boldsymbol{f}, \boldsymbol{\Phi}) \approx -V \sum_{t,j} \mathbf{1}^\mathsf{T} \boldsymbol{G}_j^2 \boldsymbol{d}_{it} \boldsymbol{s}_{kt} \ . \tag{4.102}$$

4.6.2.1.3. Refined Motion Update The motion update equation from (4.80) then rewrites as

$$[\tau_{\boldsymbol{\varphi}}(\boldsymbol{f}, \boldsymbol{\Phi})]_{ij} := [\boldsymbol{\Phi}]_{ij} - \delta \, \frac{\partial}{\partial [\boldsymbol{\Phi}]_{ij}} J(\boldsymbol{f}, \boldsymbol{\Phi}) \ , \tag{4.103}$$

with

$$\frac{\partial}{\partial [\boldsymbol{\Phi}]_{ij}} J(\boldsymbol{f}, \boldsymbol{\Phi}) \approx \frac{V}{T} \sum_t \left((\boldsymbol{H}\boldsymbol{T}_t \boldsymbol{G}_j \boldsymbol{f} \boldsymbol{s}_{it})^\mathsf{T} \left(\mathbf{1} - \frac{\boldsymbol{g}_t}{\hat{\boldsymbol{g}}_t} \right) - \alpha \sum_k \mathbf{1}^\mathsf{T} \boldsymbol{G}_k^2 \boldsymbol{d}_{jt} \boldsymbol{s}_{it} \right) \ . \tag{4.104}$$

4.6.2.1.4. Terminology According to the terminology and classification of Weickert [199], the presented regularization leads to a linear, homogeneous and isotropic diffusion filtering: linear, since the filter does not depend on the motion function itself; homogeneous, since the filter does not change with the spatial position; isotropic, because the diffusion flux is parallel to the image gradients.

4.6.2.2. Regularization After Each Motion Update

The basic idea of this type of regularization is to define a constraint which the motion function should fulfill, and to enforce this constraint after each update by certain measures. Our main motivation behind such an approach is to avoid the use of the regularization parameter α.

An interesting candidate for such a constraint is the determinant of the Jacobian matrix (simply called Jacobian determinant) [6]. The Jacobian determinant is only defined for $\mathbb{R}^n \mapsto \mathbb{R}^n$ functions, since the determinant in general is only defined for quadratic matrices. For any $\mathbb{R}^n \mapsto \mathbb{R}^n$ function, if the Jacobian determinant is zero, then the function is not invertible (this is part of the Inverse function theorem). The absolute value of the Jacobian determinant defines the expansion factor of the volume near a certain position. For values greater than one, the volume is expanding. For values equal to one, the volume is preserved. For values smaller than one, the volume is shrinking. Furthermore, if the value of the Jacobian determinant is positive, the function preserves its orientation near a certain position. If the value is negative, the function reverses its orientation near a certain position.

4.6.2.2.1. The Jacobian determinant in medical imaging In the following, we will mention some publications which seem most relevant in our context. Christensen *et al.* [40] perform automatic re-gridding in a hierarchical non-rigid registration algorithm by propagating templates as the Jacobian determinant of the transformations evaluated on the finite spatial lattice fall below 0.5. Rohlfing *et al.* [174] use it as a volume preserving constraint by punishing local deviations from one and thus encouraging volume preserving deformations. Chun *et al.* [45] derive a sufficient condition for local invertibility of spatio-temporal 4D B-spline deformations. Such conditions can be used in an optimization framework in order to enforce invertible motion [41].

4.6.2.2.2. Our Approach We impose an invertibility and orientation preservation constraint by requiring that the Jacobian determinant is greater than zero at any spatial position. The constraint is enforced in the following manner: After each motion update, the discrete Jacobian determinant is calculated for each voxel position. If we find any voxel position for which the Jacobian determinant is below or equal to zero, we perform a Gaussian smoothing. The Gaussian smoothing is performed by using a small filter kernel covering $3 \times 3 \times 3$ voxels (the Gaussian is scaled such that the center of the border voxels correspond to 2.5σ). The above steps are repeated until there is no voxel position which violates the invertibility and orientation preservation constraint.

This approach assures that the motion function is always invertible, which is a necessary condition for the image update. Note that the approach is not very restrictive, as volume compression or extension are allowed to any degree. In our experiments, we found that the reconstructed motion are sufficiently smooth in order to represent realistic deformations. Otherwise, one could increase the threshold for the determinant to a value of e. g. 0.5.

Note that repeated application of Gaussian smoothing is equivalent to Gaussian smoothing with a larger standard deviation. Also, Gaussian smoothing is equivalent to linear, homogeneous and isotropic diffusion filtering [199]. Accordingly, what we are actually doing is to adjust the time of the linear diffusion filtering process as long as needed in order to obtain invertibility and orientation preservation. It is intuitively clear that for infinite time the determinant will result in one for all positions, and thus both constraints are fulfilled.

4.7. Historical Remarks

The joint reconstruction method as it is presented in this chapter is a result of a long process of mathematical modelling and experiments on simulated and clinical data. So far we have mainly described the current result, but not how we got there. In the following, we will shortly sketch this process.

We first started with a rigid motion model, using 2D simulations of brain PET [19]. Similar to our current description we used an objective functional for which a pair of image and motion was sought for which the objective functional was minimal. Different to our current approach, the difference term of the objective functional was a least-squares term (we now use a Poisson term). There was a total variation image regularization term (now we do not use an explicit image regularization in the objective function anymore). The motion regularization term was still the same. Not only the motion update, but also the image update was accomplished by a gradient descent algorithm. The problem with this approach was that the image update needed a lot of iterations and thus converged extremely slowly.

In the next step, the least-squares term was replaced by a Poisson term [13]. Then, an adopted ML-EM algorithm could be used for the image step which greatly improved the performance. Still, a rigid motion model was used. Two additional motion regularization terms were introduced, one encouraging smooth transitions from gate to gate and the other one encouraging small motions.

Next, we derived the joint reconstruction method for a non-rigid motion model [15, 16]. Displacement fields were used in order to model the motion of each image voxel for each gate. 3D simulations were made using the XCAT phantom by Paul Segars [184].

At this time we also started to use clinical data, as we had access to cardiac patient data thanks to a collaboration with the "Nuklearmedizinische Klinik im Klinikum rechts der Isar" in Munich. In order to use this clinical data, we had to implement a system model of the Siemens Biograph 16 PET-CT scanner. Since the system matrix is extremely huge (if uncompressed), we implemented a ROI approach in which the system matrix is defined on a fine grid for a certain ROI and a coarser grid outside of the ROI. A great deal of time had to be invested in understanding the encoding of list-mode data and geometry of the Biograph, mostly because it was difficult to get information about the scanner. The ROI system matrix was abandoned later in favor of a fine grid whole FOV system matrix. It is compressed by using eight geometrical symmetries of the scanner and thus fits in roughly 2Gb of memory [17].

Since a problem was (and still is) the computational burden of our method, we made some experiments with accelerations on a graphics processing unit (GPU) [58]. On-the-fly calculations were used, but they were not as fast as using the pre-calculated system model on the CPU, so we did not further continue in that direction.

We then derived the method for a B-spline motion model and compared it to the previous displacement field model [14]. Since the B-spline model uses much less free parameters subject to estimation, it turned out to be more robust and faster to estimate. One big practical issue was still the use of a regularization parameter which was difficult to adjust. We therefore experimented with incorporating the regularization into the algorithm itself rather than the objective function. We were able to define non-invertibility of the motion function as simple constraint which had to be fulfilled after each motion update, which eliminated the need for a regularization parameter. The reconstruction quality was still as good as before [18].

5. Evaluation

In this chapter, we will detail the evaluation methods (which includes their configuration and some hints about the concrete implementation) used in order to generate the results shown in the next chapter.

5.1. Simulation

This section explains how synthetic data is generated and describes in detail the algorithms that are compared and how they are compared.

5.1.1. Data generation

Synthetic data is generated using the XCAT phantom of Paul Segars [184]. The XCAT phantom is a highly configurable synthetic whole body phantom, based on segmentation of the Visible Human Project datasets from the U. S. National Library of Medicine as well as patient datasets. Sixteen frames using the XCAT phantom are generated (Figure 5.1). Both cardiac and respiratory motion were simulated. One complete respiratory cycle of a length of five seconds is simulated, together with five cardiac cycles of one second each. For the rest of the motion parameters, the standard parameters are used (e. g. the maximal diaphragm motion is 2 cm, the anteroposterior diameter of the ribcage, body, and lungs is 1.2 cm etc.). The activities of the organs are defined differently from the XCAT standard parameters: we use average ^{18}FDG standard uptake values (SUV) measured for healthy patients according to the SUV atlas of Wang *et al.* [196]. In that study, the SUV was calculated by

$$\text{SUV} = \frac{\text{tracer concentration (Activity per Volume)}}{\text{injected dose (Activity)/body surface area}} \quad . \tag{5.1}$$

The tracer concentration is decay corrected. For further details on the calculation of the SUV, refer to [196].

The expected number of counts for each LOR is calculated by a simple analytical simulation method[1] which includes additive noise. Each gate is projected to measurement space according to a discretized version of the dynamic count rate from chapter 3

$$\hat{g}(a,t) = \frac{1}{T} \int_{\Omega} H(a,\boldsymbol{x}) f(\boldsymbol{\varphi}(\boldsymbol{x},t))\, d\boldsymbol{x} \quad , \tag{5.2}$$

where $t \in \{0, 1, \ldots, 15\}$ is the number of the gate to be simulated. The measurements are finally generated from the expected number of counts by a Poisson random generator. In

[1] We use the term analytical simulation in the context of [46, 27].

this way we take into account the acquisition time and activity. Four levels of statistical noise are simulated: $2.25 \cdot 10^8$ counts, $4.5 \cdot 10^7$ counts, $9 \cdot 10^6$ counts and $1.8 \cdot 10^6$ counts. In order to isolate motion from other image degradation effects (like scatter, randoms, etc.), and in order to focus on motion compensation algorithm (making Monte Carlo simulations with a moving phantom would have requested a considerable effort), we do not make use of Monte Carlo simulation packages.[2]

5.1.2. System Model

For the simulation, we use a system model of the Siemens Biograph 16 PET-CT scanner. Since we have a clinical dataset acquired with this scanner, the same model was used for reconstruction of both simulated and clinical data.

The Siemens Biograph 16 PET-CT scanner consists of three rings, each of which consists of 48 detector blocks. Each detector block contains a 8×8 matrix of LSO crystals. Accordingly, the system has in total 9216 crystals. The ring diameter is 82.7 cm, the axial field of view (FOV) is 16.2 cm and the transverse FOV 58.5 cm. There are $6,451,200$ sinogram bins and usually the reconstruction volume is divided into $128 \times 128 \times 48$ voxels.

Our system model only includes geometrical considerations and does not comprise effects like attenuation, scatter, randoms etc. For the calculation of the geometrical probability of detecting an event in a certain bin, we use Scheins' algorithm [178] (details about geometric system models can be found in section 2.2.2.2 on page 16). For data compression, we take advantage of eight symmetries such as explained in section 2.3.2 on page 23.

5.1.3. Comparison

In the following we describe the algorithms which are subject to comparison and describe the exact comparison procedure.

5.1.3.1. Algorithms

We compare our joint reconstruction approaches (namely JR-DF-OF, JR-DF-MU, JR-BS-OF, JR-BS-MU, to be defined in detail later) to:

- an OS-EM [91] reconstruction of motion-contaminated data (MC),
- an OS-EM reconstruction of the individual gates (IG),
- a registration and fusion of reconstructed frames method with image summation (RFRF1),
- a registration and fusion of reconstructed frames method with image re-reconstruction (RFRF2),

[2] Note that even though the degradation effects could be ignored in Monte Carlo simulations, there would be still the mismatch between the simulated system model and the one used for reconstruction.

- a motion compensating reconstruction based on the ideal motion (IM, as described below) and
- the original images (OI).

5.1.3.1.1. MC MC is an OS-EM reconstruction (20 subsets)[3] which works on the non-gated, motion-contaminated data. We include MC in the comparison in order to show how the result would look like if motion was completely ignored. Image regularization of the OS-EM algorithm is accomplished by a hybrid approach of iteration numbers and post-smoothing, as explained in section 2.4.3.4. As a figure of merit (FOM), we use the correlation coefficient of the reconstructed image and the original image.

5.1.3.1.2. IG IG is an OS-EM reconstruction which works on the individual gates. We use the same regularization procedure as for MC. IG is included in the comparison since it is a commonly used motion compensation approach in clinical practice.

5.1.3.1.3. RFRF1 & RFRF2 RFRF is a class of methods that extract the motion from the gated data by image registration. Motion estimation and image estimation are completely separated. We implement two types of RFRF methods which are used in practice. They are based on the same motion estimation procedure and differ only in the image estimation part.

Motion Estimation Motion estimation consists of (a) reconstructing each gate individually, and (b) registration of each of the reconstructed gates to a reference gate (in our case the first gate).

The reconstruction of the individual gates is done as previously described by the IG method. The Gaussian post-smoothing is chosen in such manner that the reconstructed gate has the maximal correlation coefficient with the original gate. This ensures the robustness of the registration procedure - without post-smoothing the images would be extremely noisy which would lead to poor registration results.

Image registration has already been introduced in section 3.2.2.1.3. The goal of the image registration step is to obtain a motion function $\varphi_{\text{RFRF}}(\boldsymbol{x}, t)$ which defines the deformation of a reference frame (we use the first frame) to an arbitrary frame t. In order to estimate this motion function, we use *drop*, a software for deformable image registration using discrete optimization [71, 72, 106]. In the following, we will shortly describe the relevant parameters of *drop*. The exact parameters used for registration with *drop* are given in Table 5.1.

The most important parameter is the number of grid levels and the initial grid size. We obtained the best registration results by setting the the initial grid size to $3 \times 3 \times 2$ and the number of grid levels to five. The grid is refined in each level (actually the number of patches, which is one less than the grid size, is doubled), so in the fifth level we arrive at a grid size of $33 \times 33 \times 17$. In general it is advisable not to use more control points

[3] We choose a relatively high number of subsets (Hudson and Larkin choose a maximal of 32 subsets [91]) in order to get a high acceleration factor.

Table 5.1.: Registration parameters used by *drop* for registration and fusion of reconstructed frames for both synthetic and patient data.

Parameter	Value
Image Levels	5
Grid Levels	5
Starting Grid Size	$3 \times 3 \times 2$
Final Grid Size	$33 \times 33 \times 17$
Min. Dimension	0
Interpolation	Cubic B-Splines
Image Margin	0
Update Mode	Consecutive
Sampling	Sparse
Link to Maximum Grid	yes
Steps	5
Label Factor	0.67
Optimizer	FastPD
Data Cost	SAD
Distance	Tr. Quad. Diff.
Projection	Linear
Incremental Regularization	no
Iterations	10
Gamma	0
Lambda	0.01 (IM) / 10 (IG)
Truncation	0
Histogram Bins	32

than necessary in order to reproduce the motion, since then the registration may become unstable.

Figure 5.2 shows a justification that a grid of $33 \times 33 \times 17$ control points is enough in order to model the original motion. It shows the original frames aligned to the first frame (reference frame) by using the inverse ideal motion. For a good motion estimate, deformation of a frame using the inverse motion should lead to an image similar to the reference frame. This is the case for most of the frames. The only visible differences can be seen in gates five, six, seven, eight and nine, where the spine has artifacts. The reason for the artifacts is misregistration: since *drop* starts with a coarse grid level, only few motion parameters are used. The spine is then affected by the same motion vector as the lung and the heart. This is suboptimal since the spine should not move at all between the different gates. Later, on finer grid levels, this wrong initial motion is partly but not completely corrected. We made several trials with different numbers of grid levels, including only one grid level. The effect is that, on the one hand, the less grid levels are used the less artifacts show up, but on the other hand the registration of the heart and lungs performs increasingly worse with decreasing number of grid levels. Over all, the

five grid levels approach performed best, despite the artifact. In addition, the more grid levels are used, the more robust the registration is against image noise and wrong choices of the regularization parameter λ.

The number of iterations was set to 10 (per grid level). The objective function did not further improve after 10 iterations, which indicates that more iterations were not necessary.

There is a parameter λ to be set which defines the stiffness of the displacement field. We choose λ in the following manner: we start with $\lambda = 1$. If the result is a non-invertible displacement field, which can be visually deduced by identifying overlapping areas, we increase λ until no overlapping can be noticed anymore. If the displacement field is invertible, then we reduce λ until we get as close as possible to a non-invertible displacement field (in order to leave as many freedom to the deformation as possible). Note that λ depends on the image intensity ranges used in the source and target image and accordingly has to be adjusted for different count numbers. We found that for all experiments, $\lambda = 10$ was a good choice. For the registration of the original images we chose $\lambda = 0.1$ which is due to the different intensity range.

Image Estimation RFRF1 Based on the motion function $\boldsymbol{\varphi}$, RFRF1 computes the image as the summation of all deformed gates (similar to [103, 100, 104, 101, 50], compare also section 3.2.2.2.2):

$$f_{\text{RFRF}}(\boldsymbol{x}) = \sum_t f_t(\boldsymbol{\varphi}^{-1}_{\text{RFRF}}(\boldsymbol{x}, t)) \ . \tag{5.3}$$

Image Estimation RFRF2 Instead of summing the deformed reconstructed gates, RFRF2 re-reconstructs the image based on the measured gated data together with the motion information from the motion estimation step. The re-reconstruction is done by omitting the complete motion estimation part in our joint reconstruction algorithm. Instead, the deformation field that results from the previously explained registration is used in JR as a constant value for all iterations.

Alternatively, this step could be accomplished by a system matrix modeling approach (see section 3.2.2.2.3) or similar to [163, 172, 112]. Note that, in contrast to [163, 172, 112], we adjust the motion model from the individually reconstructed gates (as explained previously) and not from gated CT data. In this way, we achieve a fair comparison since all methods work exclusively with the PET data (except for IM, of course).

5.1.3.1.4. JR-DF-OF JR-DF-OF stands for the joint reconstruction method using displacement fields (DF) representing the motion. The motion regularization is modeled in the objective function (OF), as explained in section 4.6.2.1. Note that this kind of regularization involves a regularization parameter α. We start with a $32 \times 32 \times 24$ voxel image, which implies that the displacement field is of the same size (since each voxel has its own displacement vector). The grid is refined until a grid of $128 \times 128 \times 48$ is reached. For each grid level, a set of JR iterations is performed. A JR iteration consists of an image update, followed by a number of motion updates. The motion updates are performed

Table 5.2.: Reconstruction parameters for the JR-DF-OF and JR-DF-MU methods. The motion grid is the same as the image grid (since the displacement fields contain the displacement of each voxel). The iterations are specified by the number of JR iterations, image iterations and motion iterations. E. g. "20/1/max. 100" means that there are 20 JR iterations, each consisting of one image iteration followed by maximal 100 motion iterations. Image iterations are OS-EM subiterations, e. g. for 20 subsets and 20 image iterations the whole dataset is traversed once.

Image Grid	Motion Grid	Iterations (JR/Image/Motion)
$32 \times 32 \times 24$	$32 \times 32 \times 24$	10/1/0
$32 \times 32 \times 24$	$32 \times 32 \times 24$	20/1/max. 100
$64 \times 64 \times 24$	$64 \times 64 \times 24$	20/1/max. 100
$128 \times 128 \times 48$	$128 \times 128 \times 48$	10/1/max. 100
$128 \times 128 \times 48$	$128 \times 128 \times 48$	30/1/0

Table 5.3.: Reconstruction parameters for the JR-BS-OF and JR-BS-MU methods. The motion grid is defined by the uniformly distributed B-spline nodes. The explications from Table 5.2 apply.

Image Grid	Motion Grid	Iterations (JR/Image/Motion)
$32 \times 32 \times 24$	$8 \times 8 \times 4 \times 16$	10/1/0
$32 \times 32 \times 24$	$8 \times 8 \times 4 \times 16$	20/1/max. 100
$64 \times 64 \times 24$	$16 \times 16 \times 8 \times 16$	20/1/max. 100
$128 \times 128 \times 48$	$32 \times 32 \times 16 \times 16$	10/1/max. 100
$128 \times 128 \times 48$	$32 \times 32 \times 16 \times 16$	30/1/0

until the maximal motion vector of the gradient is below 0.5 mm. The maximum number of iterations is limited to 100 in order to avoid exaggerated fine tuning at a premature level. Image and motion are interpolated linearly when switching to a finer grid. Details are given in Table 5.2.

In order to determine the best value for the regularization parameter α, the whole reconstruction procedure was repeated several times for different values of α: $0.5, 0.6, 0.7, 0.8, 0.85, 0.9, 0.95,$ and 0.99. Not all parameters lead to a complete reconstruction, since if for a certain grid level no valid motion update is found the algorithm aborts reconstruction (see chapter 6).

5.1.3.1.5. JR-DF-MU JR-DF-MU stands for the joint reconstruction method using displacement fields (DF) representing the motion. Motion regularization is applied after the motion update (MU), as explained in section 4.6.2.2. The number of iterations and the grid levels are the same as for JR-DF-OF and shown in Table 5.2.

5.1.3.1.6. JR-BS-OF JR-BS-OF is the joint reconstruction method using B-spline (BS) interpolation of the motion. Motion regularization is modeled in the objective function (OF). The number of iterations and image grid levels is shown in Table 5.3. In the spatial domain we use quadratic B-splines, and in the time domain B-splines of degree zero. The same parameters α are used for the reconstruction.

5.1.3.1.7. JR-BS-MU JR-BS-MU is the joint reconstruction method using B-spline (BS) interpolation of the motion. Motion regularization is applied after the motion update (MU). The number of iterations and image grid levels is shown in Table 5.3. The degree of the B-splines is the same as for JR-BS-OF.

5.1.3.1.8. IM The IM method is basically the same as the RFRF2 method, with the only difference that the motion is not estimated from the reconstructed noisy gates but from the original gates that were used for the simulation. This is why we call it IM: the motion is supposed to be close to ideal (note the previous discussion on the artifacts). In a clinical setting, instead of the original gates, gated CT images could be used for motion estimation.

5.1.3.1.9. OI The original images which come directly from the XCAT phantom are denoted as OI. This is of course not an image reconstruction method.

5.1.3.2. Comparison of Reconstructed Images

We compare the resulting images from the respective reconstruction methods to the original images (OI) by calculating their correlation coefficient (CC). A border of 40 pixels at each side of the lateral direction is cut off, and 2 pixels in the axial direction. The CC between two images $\boldsymbol{x}$ and $\boldsymbol{y}$ (represented as vectors) is defined as

$$\mathcal{CC}(\boldsymbol{x}, \boldsymbol{y}) = \frac{\boldsymbol{x}^\mathsf{T}\boldsymbol{y}}{\|\boldsymbol{x}\|\|\boldsymbol{y}\|} \quad . \tag{5.4}$$

Both $\boldsymbol{x}$ and $\boldsymbol{y}$ are shifted such that their mean value is zero. The CC ranges between -1 for totally anti-correlated images and 1 for perfectly correlated images.

In the case of reconstruction methods which deliver different images for each gate (IG, RFRF1, RFRF2 and JR), the CC is calculated for each reconstructed gate with the respective original gate and then the average value is taken (for the RFRF methods and JR this involves a deformation of the reconstructed image to the respective motion gates):

$$\mathcal{CC}^{\text{IG}}_{\text{image}} = \frac{1}{T}\sum_{t=1}^{T} \mathcal{CC}(\boldsymbol{f}_t^{\text{IG}}, \boldsymbol{f}_t^{\text{OI}}) \ , \tag{5.5}$$

$$\mathcal{CC}^{\text{RFRF}}_{\text{image}} = \frac{1}{T}\sum_{t=1}^{T} \mathcal{CC}(\boldsymbol{T}_{\boldsymbol{\varphi}_t^{\text{RFRF}-1}} \boldsymbol{f}^{\text{RFRF}}, \boldsymbol{f}_t^{\text{OI}}) \ , \tag{5.6}$$

$$\mathcal{CC}^{\text{JR}}_{\text{image}} = \frac{1}{T}\sum_{t=1}^{T} \mathcal{CC}(\boldsymbol{T}_{\boldsymbol{\varphi}_t^{\text{JR}}} \boldsymbol{f}^{\text{JR}}, \boldsymbol{f}_t^{\text{OI}}) \ . \tag{5.7}$$

Since the CC depends heavily on the smoothness of the image, we have to make sure that we compare the images for the same level of background noise. We measure the noise by calculating the percent standard deviation (the standard deviation related to the mean value in percent) for a specific background ROI. The reconstructed images are post-smoothed with differently sized Gaussian kernels, and the CC is calculated for each instance.

Selected slices of the ROIs are shown in Figure 5.3. The ROIs are defined as follows: from the original XCAT images, the background values are segmented by a simple intensity based segmentation procedure: as a first approximation, all voxels whose values are below an SUV of $0.3\frac{\text{m}^2}{\text{ml}}$ are considered to be part of the background region. Then, the ROI is grown by ≈ 2.5 cm in each dimension. This step is necessary since otherwise foreground values would enter in the background region due to the post-smoothing which is applied to the images. Between roughly 8.000 and 9.000 voxels ($\approx 8\%$ to $\approx 10\%$ of all voxels) fall within the background ROI. The ROI for the MC method is calculated as the intersection set of all other ROIs, since the heart is in a different position in the different gates. With 5.491 voxels ($\approx 5.4\%$ of all voxels) it is the smallest ROI.

5.1.3.3. Comparison of Reconstructed Motion

We visually compare the reconstructed motion fields one to another for different views (transverse, coronal and sagittal) by extracting the two dimensional in-plane components of the respective motion field and applying them to a two dimensional checkerboard-like pattern. Note that for JR these motion fields are shown relative to a virtual reconstruction frame, and therefore they are not directly comparable to the RFRF motion fields.

5.1.3.4. Choice of Regularization Parameter α

Both JR-DF-OF and JR-BS-OF have a regularization parameter α which has to be adjusted to a value between zero and one. The correct choice of α is crucial for successful motion compensation. Figure 5.4 shows three JR reconstructions with different values for α. When α is too low, extreme deformations in the motion field and image artifacts are present. Also, a too small α may lead to non-invertible deformation fields (for a discussion on regularization specifically dedicated to preventing non-invertible deformation fields, see [43]). When α is too high, almost no deformations are present and the image looks like as if no motion compensation had taken place (it is blurred).

The correct choice of such regularization parameters is still an unsolved problem in general. For some specific problems, an analysis of the L-curve has been successful [81, 20]. Also, generalized cross-validation is a potential candidate for finding the correct regularization parameter [74].

We perform different reconstructions for JR-DF-OF and JR-BS-OF with the following values of α: $0.5, 0.6, 0.7, 0.8, 0.85, 0.9, 0.95, 0.975, 0.99$. For later comparisons, usually the α whose reconstruction leads to the highest CC with the original data is chosen.

5.2. Patient Data

In the following, we explain how clinical patient data was acquired and preprocessed and how comparison took place. We apply the same reconstruction algorithms as in the simulation study. In the evaluation, we limit the evaluation to a visual impression rather than quantitative values.

5.2.1. Data Acquisition and Pre-processing

The data from a patient referred for an ^{18}F-FDG PET-CT examination for assessment of the myocardial viability is used. The patient was injected 400 MBq of ^{18}F-FDG and, 60 minutes after injection, a ten minutes list-mode acquisition is performed using a Siemens Biograph Sensation 16 PET-CT scanner. For reconstruction, we only use two minutes of the list-mode data in order to show that motion compensation techniques can deal with such small amount of data. Note that since we use 24 gates, each gate corresponds to data which would be acquired in only five seconds (of course, due to gating, the data for each gate does not correspond to a continuous frame but is distributed over the whole scan)!

We propose a novel image based gating procedure in order to divide the data into $T = 24$ gates: one-second frames z_i of data are reconstructed at a low resolution ($32 \times 32 \times 48$ voxels) and with only five ML-EM iterations. A ROI surrounding the heart is cropped from these reconstructed frames. Then, a gating function $\xi : \mathbb{N} \mapsto \mathbb{N}$ which maps a frame b to a gate $t = \xi(b) \in \{0, 1, \ldots, T-1\}$ is sought. The gating function ξ should be of such manner that similar frames (frames with a high CC) are grouped into the same gate. This is achieved by maximizing the following objective function

$$\mathcal{G}(\xi) := \sum_{t=0}^{T-1} \sum_{\{i,j\,:\,\xi(i)=\xi(j)=t \,\wedge\, i \neq j\}} \mathcal{CC}(z_i, z_j) \ . \tag{5.8}$$

Here, $\{i, j \ : \ \xi(i) = \xi(j) = t \ \wedge \ i \neq j\}$ is simply the set of frames which belong to gate t. So, for each gate, the CC of all frames which belong to this gate is summed up.

The objective function is maximized by a brute force technique: we start with a randomly initialized gating function, each gate containing approximately the same number of frames. Then, the gates of each pair of distinct frames i and j are swapped. After each swap the objective function is evaluated. If the objective function did not increase by the swap they are swapped back to their initial state again. Otherwise we continue with the

next pair of frames. This procedure is repeated until no improvement of the objective function is found anymore.

For reconstruction of clinical data, the same system matrix as in the simulation study is used. Corrections for attenuation, scatter or randoms were not applied.

5.2.2. Comparison

In the following we describe the algorithms which are subject to comparison and describe the comparison procedure.

5.2.2.1. Algorithms

We compare the same reconstruction methods like in the simulation case described in section 5.1.3. We leave out the IM method since we do not have CT images for each respective gate (also, there does not seem to be a simple way to acquire CT images which correspond to the gates found with our proposed gating method - not talking about the dose exposition of the patient). Of course we also have to discard the OI method since the ideal tracer distribution is not known (we do not have a ground truth).

Since ground truth data is not available for our clinical data, also the number of iterations and Gaussian post-smoothing can not be based on the CC of the reconstructed image with the ground truth. We therefore reuse the FWHM values for post-smoothing which were found to be optimal in the simulation study (and there from the $1.8 \cdot 10^6$ count case since it seems to be most similar to the patient data).

5.2.2.2. Comparison of Reconstructed Images

Since we do not know the original tracer distribution as we do in the simulation study, we cannot perform the same quantitative evaluation in order to determine which method is performing best. We will rely on a visual comparison.

5.2.2.3. Comparison of Reconstructed Motion

Also for the motion fields we rely on a visual comparison.

5.3. Historical Remarks

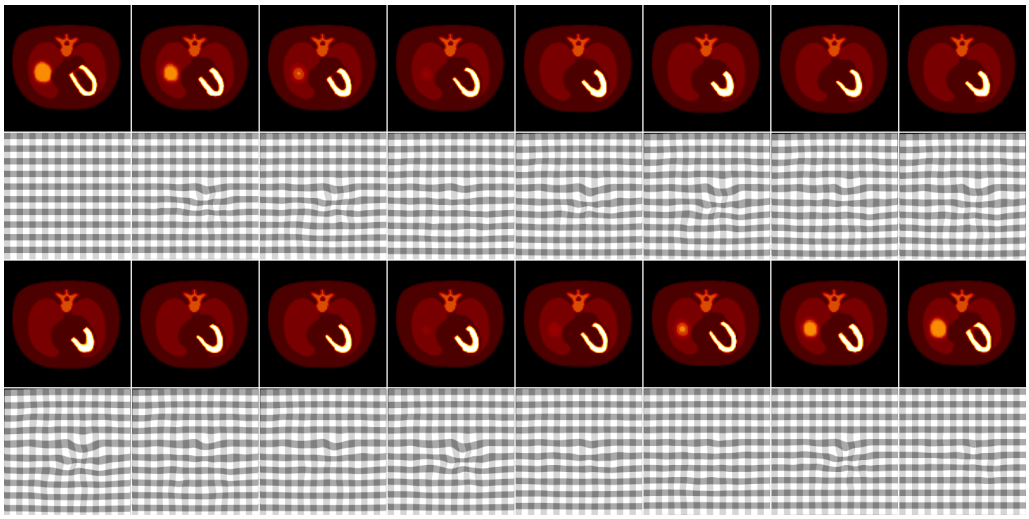
Our first publications used either purely visual comparisons [19] or rather simple quantitative comparisons based exclusively on the CC of the reconstructed images with the original images [13, 16, 15]. We learned from the reviewers of our TMI paper [17] that the correlation coefficient depends heavily on the image noise, and thus we started to plot the background noise of the reconstructed images versus its CC with the ideal image. The image noise is now adjusted by Gaussian post-smoothing of the reconstructed image.

We also compared the reconstructed motion quantitatively to the ideal motion. However, this approach was abandoned later since we consider it to not be meaningful: for homogeneous regions, none of the algorithms can find the true motion field. We decided

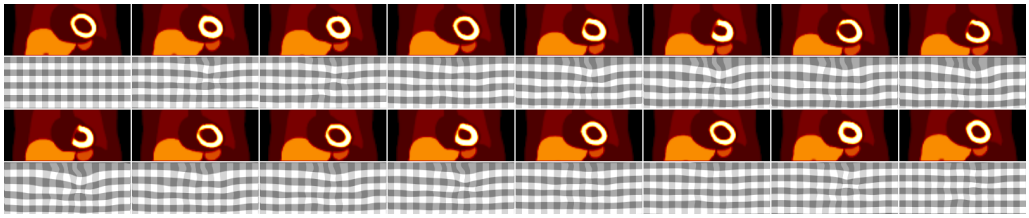
that not the motion field itself, but its effect on the image is important. This effect of the motion on the image is ultimately already included in the comparison of the CC of the reconstructed images with the ideal images.

For clinical data, we initially had included a quantitative measure which is based on cross-validation [105] in our TMI paper [17]. We partitioned the list-mode events S equally into five consecutive sets $s_1, \ldots, s_5$. As the total acquisition time of the scan is ten minutes, each subset s_i represents two minutes of list-mode data. Then, the different reconstruction methods were applied to the set $s_2 \cup s_2 \cup s_3 \cup s_4 \cup s_5$ (the training set). For the remaining set s_1 (the validation set), the likelihood (refer equation 4.4) of the resulting pairs of image and motion of the different reconstruction methods was calculated. The method which corresponds to the highest likelihood is the one that reconstructs the tracer distribution closest to the likelihood in the validation set. The same procedure was performed four more times for the remaining s_i as validation set and $S \setminus s_i$ as training set.

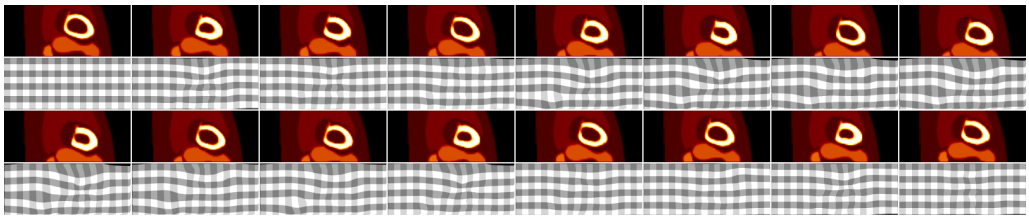
Table 5.4 shows the results of this comparison. The joint reconstruction method turned out to be best according to this measure for all validation sets. However, the reviewers' opinions were mixed about this approach. While all of them acknowledged that it could be a good way to compare methods when ground truth data is not available, the general tenor was that the resulting numbers were difficult to interpret. In fact, it turned out that our joint reconstruction method had a likelihood which was about 10^6 to 10^7 times bigger than the best competitive method. Since this huge number is not reflected in the images, it is admittedly hard to be interpreted and we thus removed this comparison from the TMI paper. We still believe that there is a huge potential in such a comparison, however, we focused on other investigations and did not come back to it up to now.



(a) Transverse slices



(b) Coronal slices



(c) Sagittal slices

Figure 5.1.: Generated original frames using the XCAT phantom. A complete respiratory cycle (two centimeter diaphragm movement) and five cardiac cycles are generated. These frames are used for simulating measurement data for different statistical scenarios. The motion fields are visualized by a deformed grid (the first frame is the reference frame). They were acquired by image registration using the registration toolkit *drop*.

Figure 5.2.: The original frames aligned to the first original frame using the inverse ideal motion. The ideal motion is obtained from registration of the ideal images with the registration toolkit *drop*. Five grid levels (starting with $3 \times 3 \times 2$ and ending with $33 \times 33 \times 17$) and ten iterations per level were used. For most frames there is hardly any visible difference between the aligned gates, which indicates that $33 \times 33 \times 17$ control points are sufficient in order to approximate the original motion.

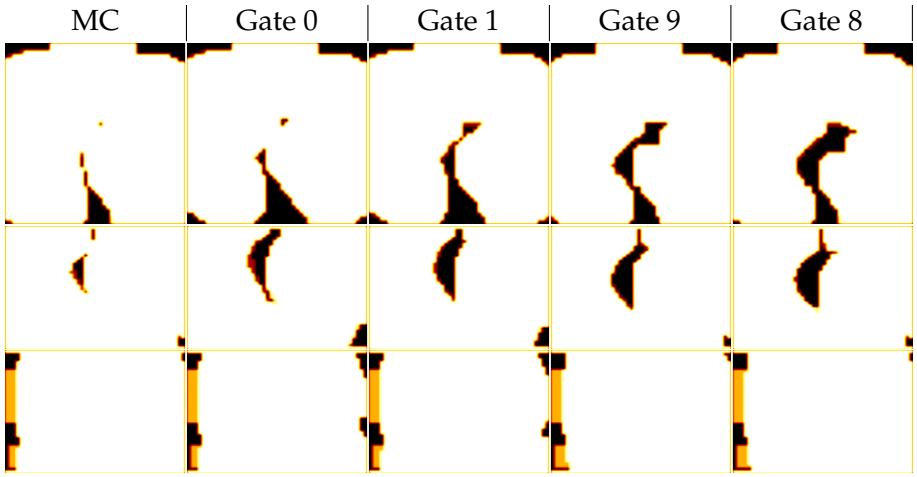
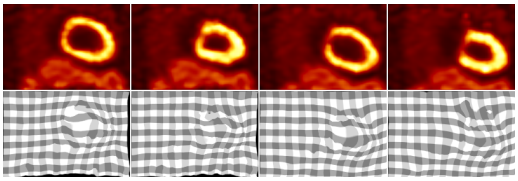
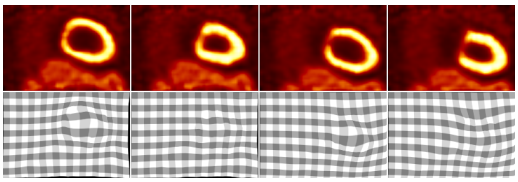


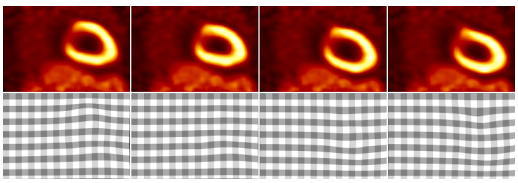
Figure 5.3.: Transverse (first row), coronal (second row) and sagittal (third row) slices for different background ROIs used for the calculation of the FOM. The dark voxels are part of the ROI. Note that the ROI for the MC method is smaller than all other ROIs, since it is the intersection set of all 24 gates (not just the four gates shown).



(a) $\alpha = 0.5$ (too low)



(b) $\alpha = 0.85$ (good choice)



(c) $\alpha = 0.99$ (too high)

Figure 5.4.: Comparison of our JR method for different values of α for the simulation study. For a too low α, the reconstructed motion field contains extreme deformations which result in artifacts in the reconstructed image. For a too high α, deformations are almost completely suppressed which results in a motion blurred image, similar to as if no motion compensation had taken place.

Table 5.4.: Evaluation of the reconstructed image for patient data by five-fold cross-validation. The likelihood of the image reconstructed from the training set is calculated for the validation set. This is done for five different pairs of training and validation set and for different reconstruction methods (MC, IG, RFRF 1, RFRF 2 and JR). JR provides the highest likelihood in all cases.

Validation Set	**MC**	**IG**	**RFRF 1**	**RFRF 2**	**JR**
s_1	$2.57 \cdot 10^{-4829925}$	$8.90 \cdot 10^{-4829925}$	$1.58 \cdot 10^{-5021813}$	$4.08 \cdot 10^{-4829924}$	$\mathbf{1.01 \cdot 10^{-4829918}}$
s_2	$1.78 \cdot 10^{-4881447}$	$3.38 \cdot 10^{-4881446}$	$4.89 \cdot 10^{-5112894}$	$4.01 \cdot 10^{-4881446}$	$\mathbf{6.80 \cdot 10^{-4881440}}$
s_3	$1.89 \cdot 10^{-4877181}$	$2.70 \cdot 10^{-4877178}$	$1.38 \cdot 10^{-5106066}$	$9.12 \cdot 10^{-4877180}$	$\mathbf{3.98 \cdot 10^{-4877173}}$
s_4	$3.27 \cdot 10^{-4866030}$	$6.52 \cdot 10^{-4866029}$	$2.87 \cdot 10^{-5086925}$	$5.59 \cdot 10^{-4866029}$	$\mathbf{2.02 \cdot 10^{-4866023}}$
s_5	$5.25 \cdot 10^{-4849996}$	$1.09 \cdot 10^{-4849992}$	$1.15 \cdot 10^{-5059501}$	$1.13 \cdot 10^{-4849994}$	$\mathbf{1.26 \cdot 10^{-4849988}}$

6. Results and Discussion

In this chapter we present reconstruction results for both simulated and patient data. The simulations were run for four different statistical noise levels, represented here by the number of detected counts. The chapter is divided into a simulation section and a patient data section.

6.1. Simulation

This section is organized according to the performed analysis.

6.1.1. Influence of Parameter α

Before a comparison involving JR-DF-OF and JR-BS-OF can be done, we have to know which parameter α leads to the best results. Figures 6.1, 6.2, 6.3 and 6.4 show the background noise plotted versus the CC of the reconstructed image for a specific parameter α. The Gaussian kernels start with a FWHM of 0 cm and then increase by 0.25 cm until a maximum value of 1.25 cm (for comparison, the voxel size of the highest resolution images is 0.46 mm). Higher FWHMs could not be chosen since then the background ROIs would not contain enough voxels in order to allow for a stable calculation of the noise. The rightmost point on the curves corresponds to FWHM 0. The leftmost point on the curves corresponds to FWHM 1.25 cm.

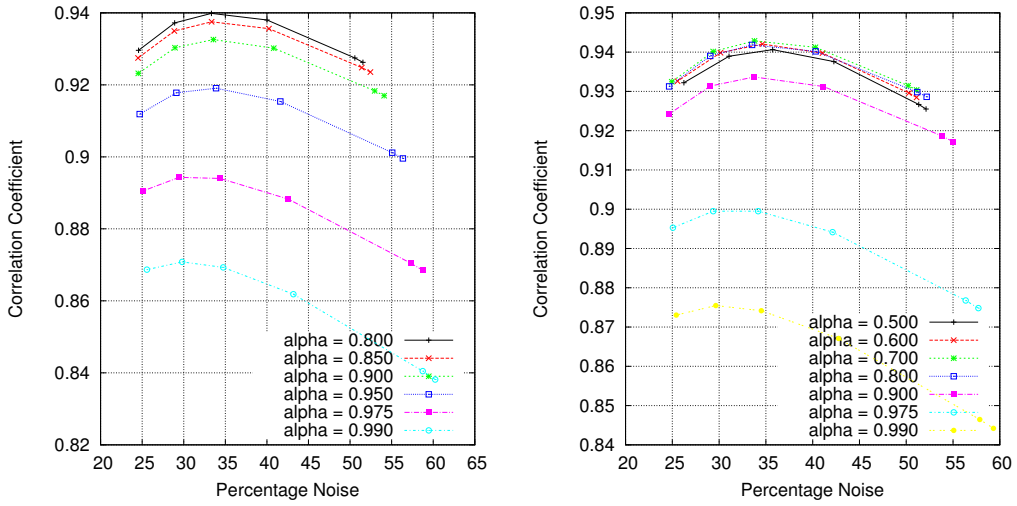
We usually choose the optimal value of α according to the maximal CC. However, if two or more α lead to extremely close maximal CCs, we choose the highest one since it leads to a more stable reconstruction. The chosen values are shown in Table 6.1. It can be seen that the optimal values are equal for JR-DF-OF and JR-BS-OF in almost all cases, except for the $2.25 \cdot 10^8$ counts case.

The influence of α on the CC is highest for high statistical noise (more than 0.06 difference) and lowest for low statistical noise (less than 0.005). This may explain the difference in the optimal choice of α for JR-DF-OF and JR-BS-OF in the $2.25 \cdot 10^8$ counts case.

Note that not all curves for all α are plotted, especially the curves corresponding to low α are sometimes missing. The reason is that in these cases the reconstruction became unstable and did not finish before reaching the highest image grid level, and thus could not be compared reasonably.

6.1.2. Quantitative Comparison of Reconstruction Methods

Figures 6.5, 6.6, 6.7 and 6.8 show a quantitative comparison of the different reconstruction methods. For the sake of a clearer presentation, the four JR methods (JR-DF-OF, JR-DF-



(a) JR-DF-OF (b) JR-BS-OF

Figure 6.1.: Simulation, $1.8 \cdot 10^6$ counts: influence of regularization parameter α on reconstruction quality for (a) displacement fields and (b) B-spline interpolation.

Table 6.1.: Chosen values for α

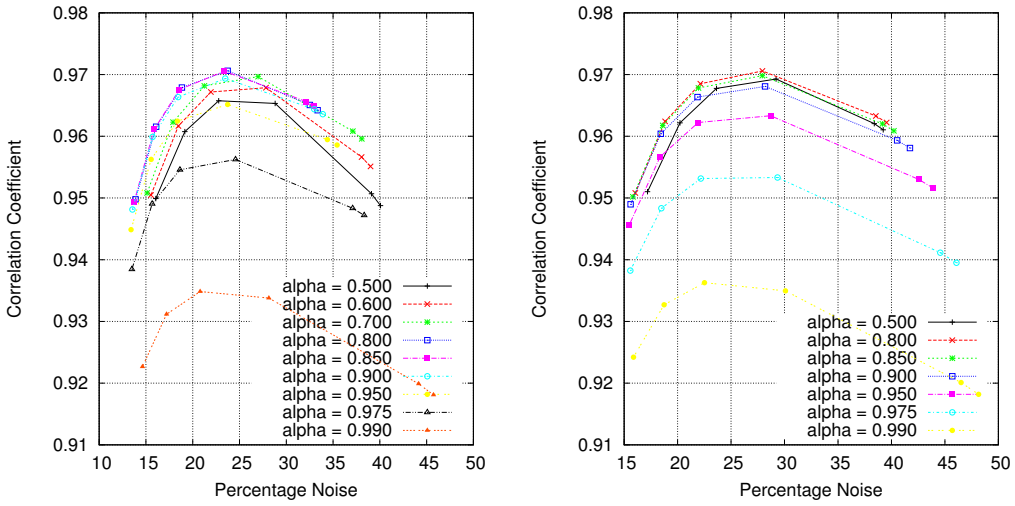
Number of Counts	Optimal α JR-DF-OF	Optimal α JR-BS-OF
$1.8 \cdot 10^6$	0.8	0.8
$9 \cdot 10^6$	0.9	0.9
$4.5 \cdot 10^7$	0.95	0.95
$2.25 \cdot 10^8$	0.8	0.95

MU, JR-BS-OF and JR-BS-MU) are compared separately. Then, as a representative of the JR methods, JR-BS-MU is shown in the complete comparison with the rest of the methods.

Comparing the JR methods, JR-BS-MU performs best (in terms of the maximal CC) for all scenarios. Differences become smaller with increasing number of counts.

In the global comparison, JR is always the best method (IM is outside the competition and considered to be the reference method, being based on the ideal motion information), followed by RFRF2, RFRF1, IG and finally MC. The term "best" is based on the maximal CC and, if the maximal CC of two methods is very close, the background noise characteristics. In the highest counts case ($2.25 \cdot 10^8$ counts) e. g. the IG method is slightly ahead of RFRF1 in terms of the maximal CC, however, its background noise level is much more elevated, which is why we consider it to be worse than RFRF1 in this case.

We further note that the differences between the compared methods become smaller with increasing number of counts. JR is always closer to IM than to RFRF2. MC is far behind all other methods for all count levels. With respect to noise characteristics, all methods are pretty similar except for IG which ranges on a much elevated background noise level.



(a) JR-DF-OF (b) JR-BS-OF

Figure 6.2.: Simulation, $9 \cdot 10^6$ counts: influence of regularization parameter α on reconstruction quality for (a) displacement fields and (b) B-spline interpolation.

6.1.3. Visual Comparison of Reconstructed Images

In order to see that seemingly small differences in the CC can have a visible effect, we now take a look at a visual comparison of the reconstructed methods. Figures 6.9, 6.10, 6.11 and 6.12 show transverse, coronal and sagittal slices for gates zero, one, nine and eight for the different reconstruction methods. A border of 40 pixels at each side of the lateral direction is cut off, and 2 pixels in the axial direction, which results in a zoom effect.

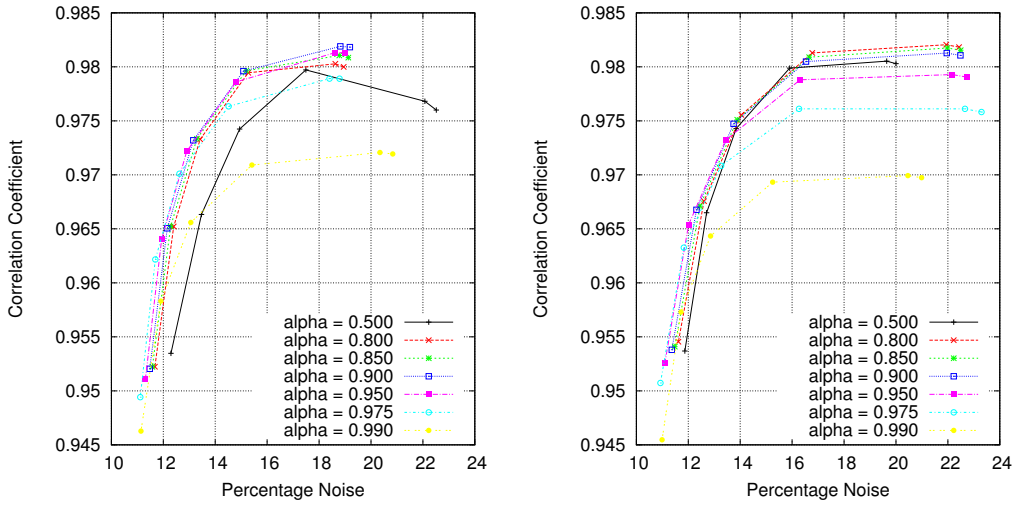
In accordance with the quantitative analysis, fewest differences can be seen in the $2.25 \cdot 10^8$ counts case, whereas the $1.8 \cdot 10^6$ counts case depicts the most notable differences between all methods. Usually the IM method can be regarded as the best achievable result for motion compensation, since the motion here comes from the original data frames which do not suffer any statistical degradation. JR comes very close to IM, even in the $1.8 \cdot 10^6$ counts case.

It is interesting to note that the registration error for the ideal motion which was observed at the spine in Figure 5.2 of the previous chapter is visible for the IM method at $4.5 \cdot 10^7$ counts (Figure 6.11) and $2.25 \cdot 10^8$ (Figure 6.12) counts. A similar registration artifact is visible for the RFRF methods at gates eight and nine for $2.25 \cdot 10^8$ counts. JR does not suffer from such artifacts.

6.1.4. Visual Comparison of Reconstructed Motion Fields

The reconstructed motion fields of the RFRF and JR methods are compared in Figures 6.13, 6.14, 6.15 and 6.16.

In general it can be noted that the motion fields of the RFRF methods are more turbu-



(a) JR-DF-OF (b) JR-BS-OF

Figure 6.3.: Simulation, $4.5 \cdot 10^7$ counts: influence of regularization parameter α on reconstruction quality for (a) displacement fields and (b) B-spline interpolation.

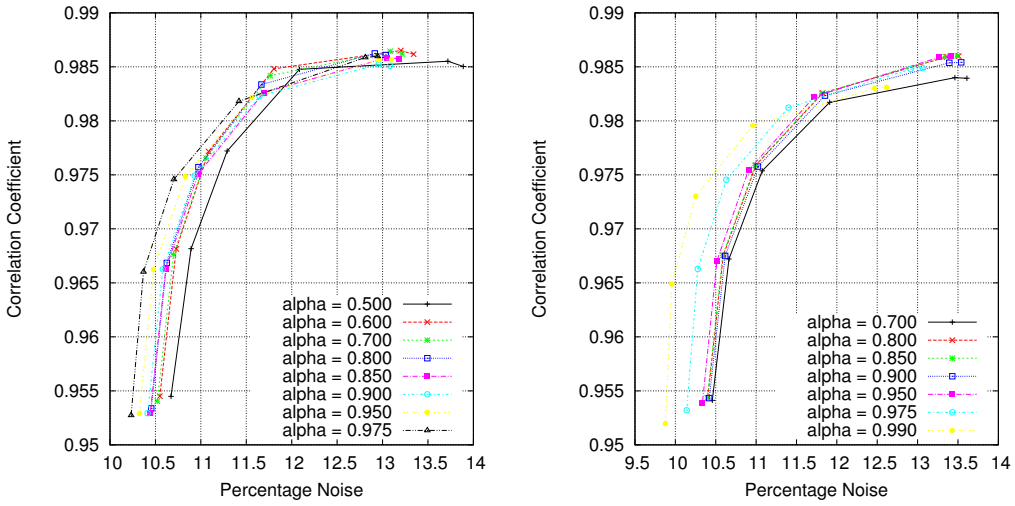
lent than the JR motion fields. This is for two reasons: first, the reference frame is a fixed gate, whereas the JR images are relative to a virtual reference frame. Virtual reference frame means that if the pure JR image was taken without any motion applied to it, it does not correspond to any of the motion states of the gates. For RFRF, gate zero is the reference gate which can be seen in the motion field: it is an non-deformed checkerboard image, which means that the corresponding deformation field is an identity transformation. This also implies that the registration result depends on the chosen reference gate. If a reference gate in between the inhalation and exhalation state is chosen, then less deformation is needed than for a reference gate which corresponds is in inhalation or exhalation state (same for the cardiac phases). The second reason for more turbulent RFRF motion fields is that the RFRF motion estimation works with the IG images which makes it more prone to noise.

Note that the black areas which are visible at the borders of the motion fields are due to the visualization: we first crop a 2D ROI of the 3D motion field and then apply it to a 2D checkerboard slice.

6.2. Patient Data

As already mentioned in the previous chapter, we only use two minutes acquisition time of the total ten minutes. This results in $\approx 4.5 \cdot 10^7$ detected counts. Note that the number of counts cannot be directly compared to a simulation case since it seems that the background region is more active in case of patient data compared to the simulated data.

We made a quantitative evaluation based on earlier data for a revision of our TMI paper [17]. Please refer to section 5.3 and Table 5.4 for details on this comparison.



(a) JR-DF-OF (b) JR-BS-OF

Figure 6.4.: Simulation, $2.25 \cdot 10^8$ counts: influence of regularization parameter α on reconstruction quality for (a) displacement fields and (b) B-spline interpolation.

6.2.1. Visual Comparison of Reconstructed Images

Like in the simulation case we visually compare the reconstructed images. Figure 6.17 shows the comparison. A border of 40 pixels at each side of the lateral direction is cut off, and 2 pixels in the axial direction, which results in a zoom effect. Since we lack the ground truth data IM and OI are not available for comparison. We chose two gates for comparison which seem to be most different, namely gates 11 and 12.

It can be noted that the quality of the reconstructed images for all methods is worse than in the simulation study. The main reason for this discrepancy lies in the system model used for reconstruction: we do not model attenuation, scatter nor randoms. Furthermore, the geometrical model probably has some differences to the ideal geometry of the Siemens Biograph 16, too. All these factors do not play a role in the simulation study where the same system model is used for both simulation and reconstruction.

6.2.2. Visual Comparison of Reconstructed Motion

In order to give the reader an impression about the movement of the complete data (especially since we cannot show the original gates for the patient data), we will show all 24 gates for IG (Figure 6.18), RFRF1 (Figure 6.19), RFRF2 (Figure 6.20) and JR (Figure 6.21). One notes again (and for the same reasons), like in the simulated cases, that the RFRF motion fields are much more turbulent than the JR motion fields.

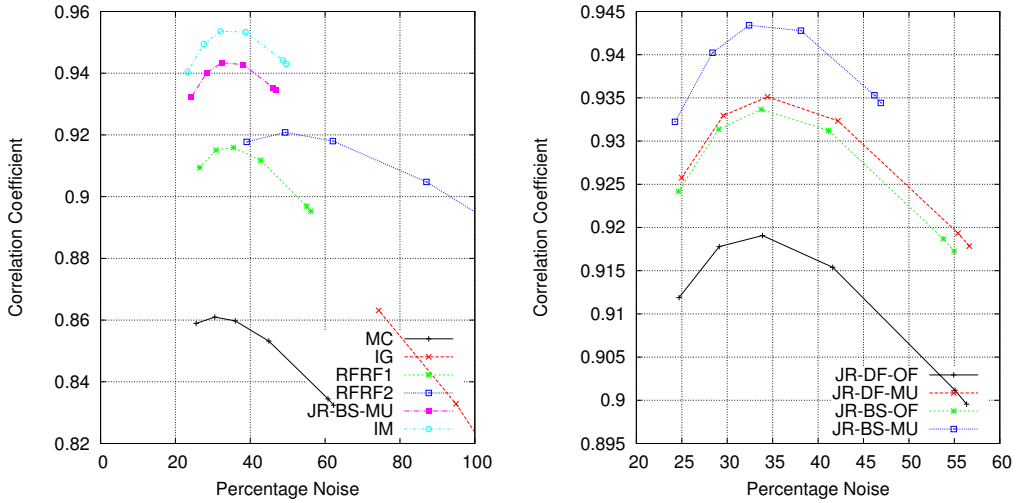


Figure 6.5.: Simulation, $1.8 \cdot 10^6$ counts: quantitative comparison of different reconstruction methods. (a) complete set of reconstruction methods, JR is represented by JR-BS-MU. (b) comparison of different JR methods.

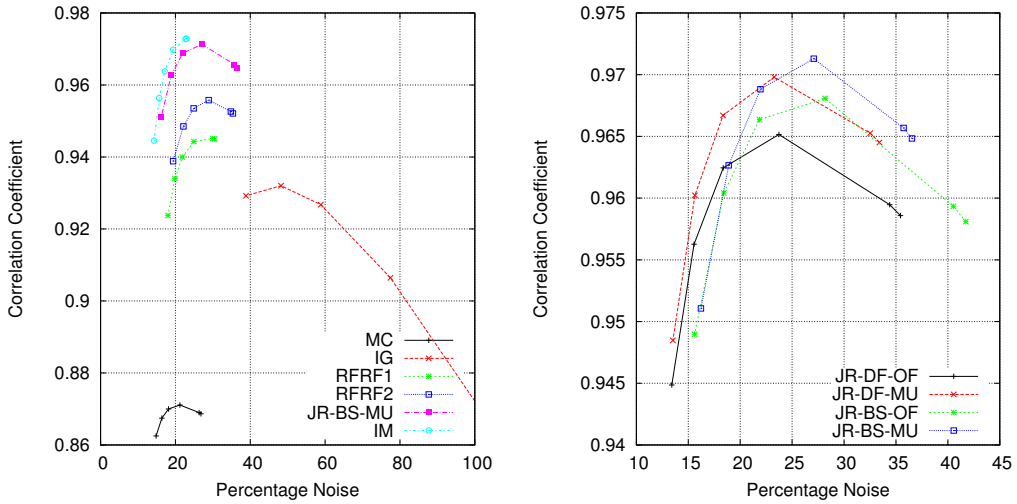
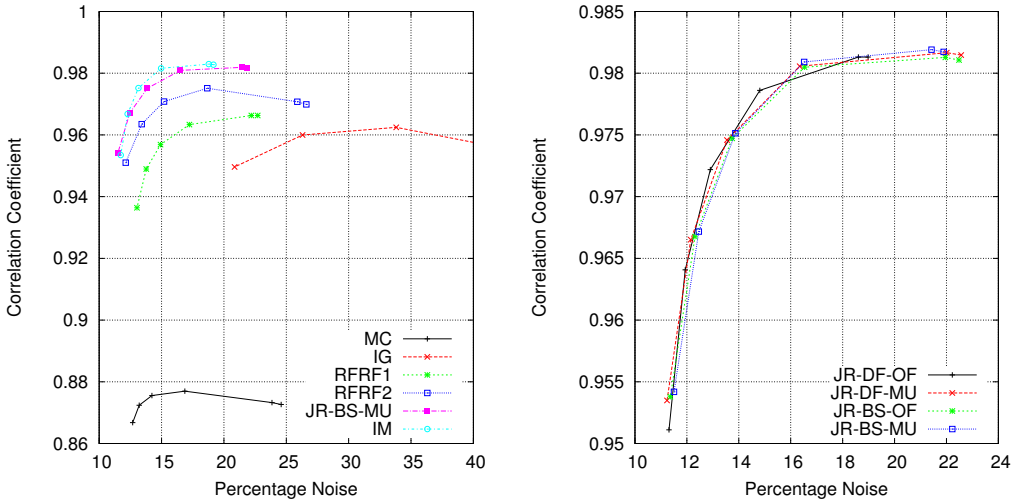
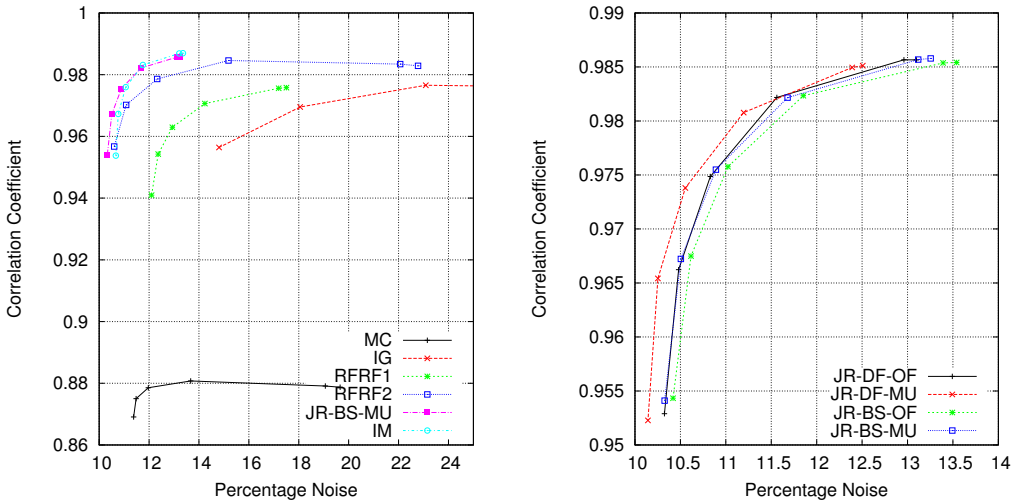


Figure 6.6.: Simulation, $9 \cdot 10^6$ counts: quantitative comparison of different reconstruction methods. (a) Complete set of reconstruction methods, JR is represented by JR-BS-MU. (b) Comparison of different JR methods.



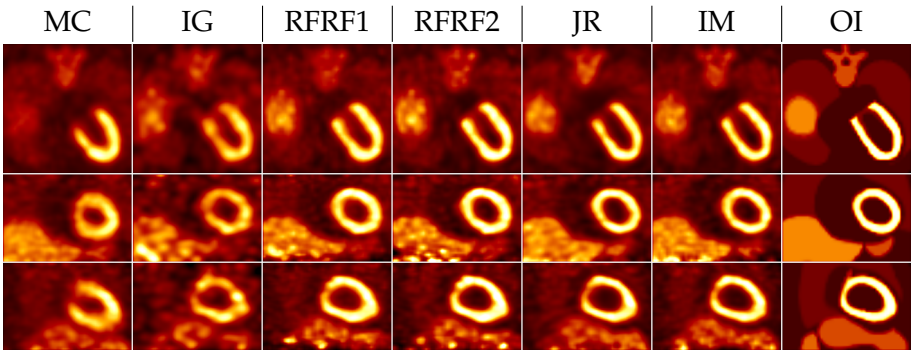
(a) (b)

Figure 6.7.: Simulation, $4.5 \cdot 10^7$ counts: quantitative comparison of different reconstruction methods. (a) complete set of reconstruction methods, JR is represented by JR-BS-MU. (b) comparison of different JR methods.

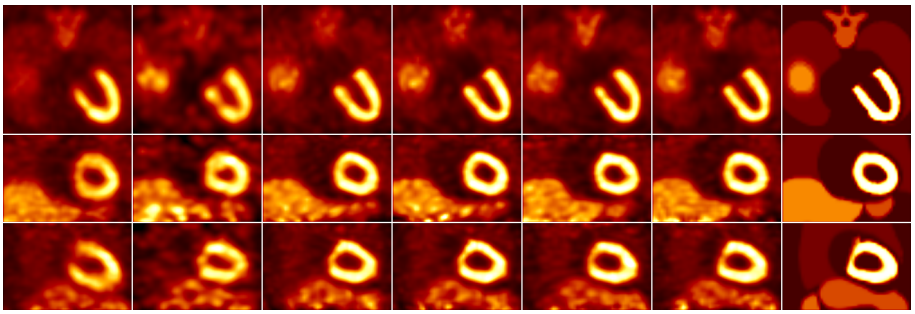


(a) (b)

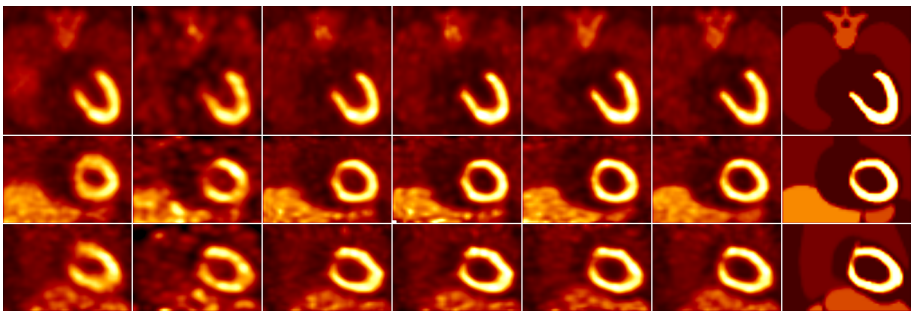
Figure 6.8.: Simulation, $2.25 \cdot 10^8$ counts: quantitative comparison of different reconstruction methods. (a) complete set of reconstruction methods, JR is represented by JR-BS-MU. (b) comparison of different JR methods.



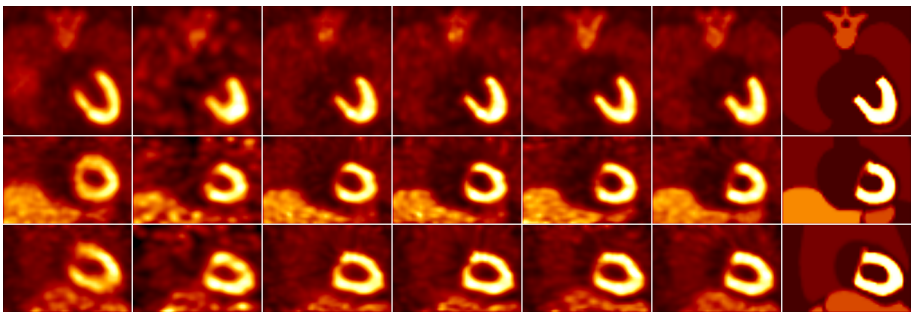
(a) Gate 0



(b) Gate 1

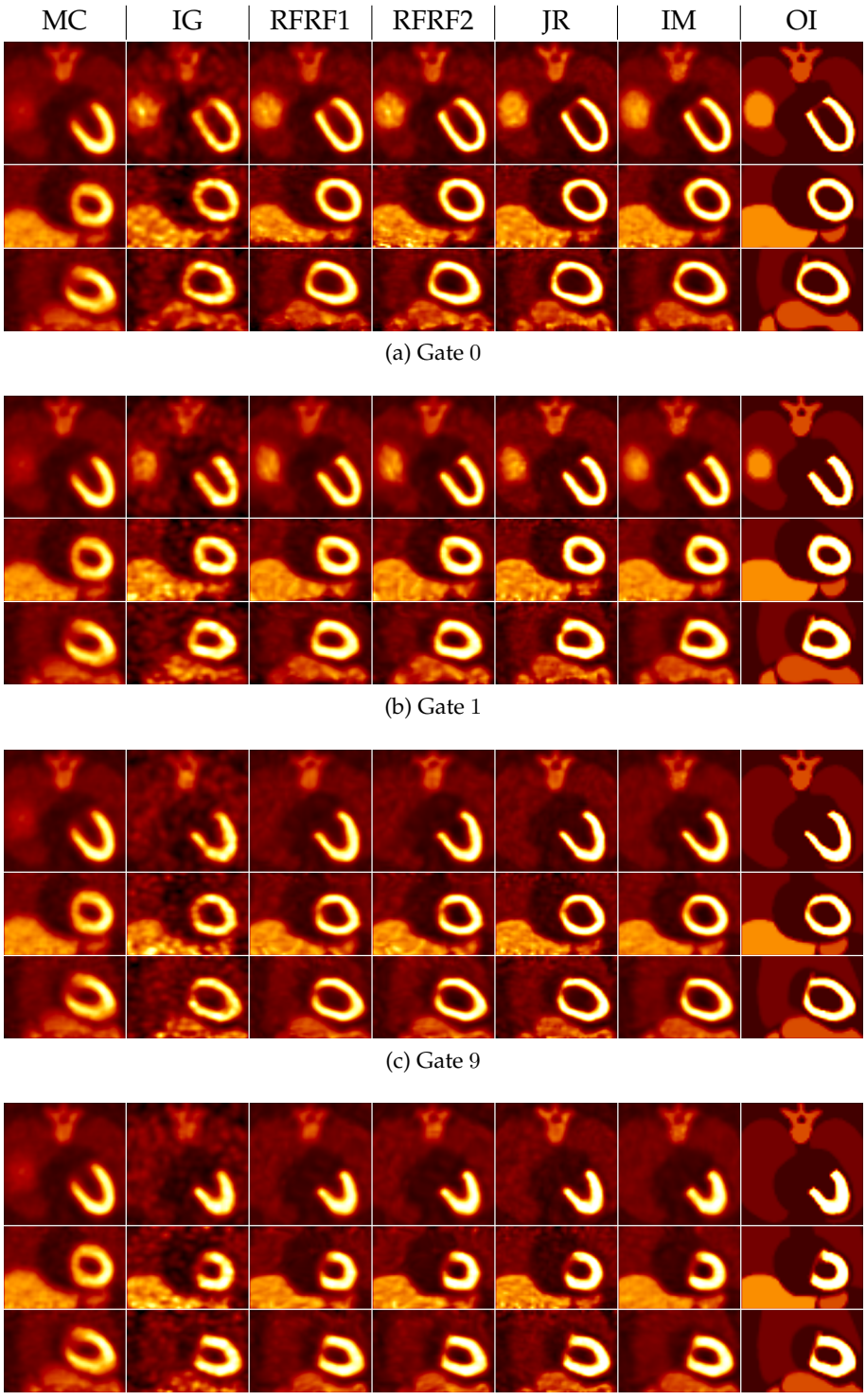


(c) Gate 9



(d) Gate 8

Figure 6.9.: Simulation, $1.8 \cdot 10^6$ counts: visual comparison of different reconstruction methods. JR-BS-MU is shown as an JR ambassador.



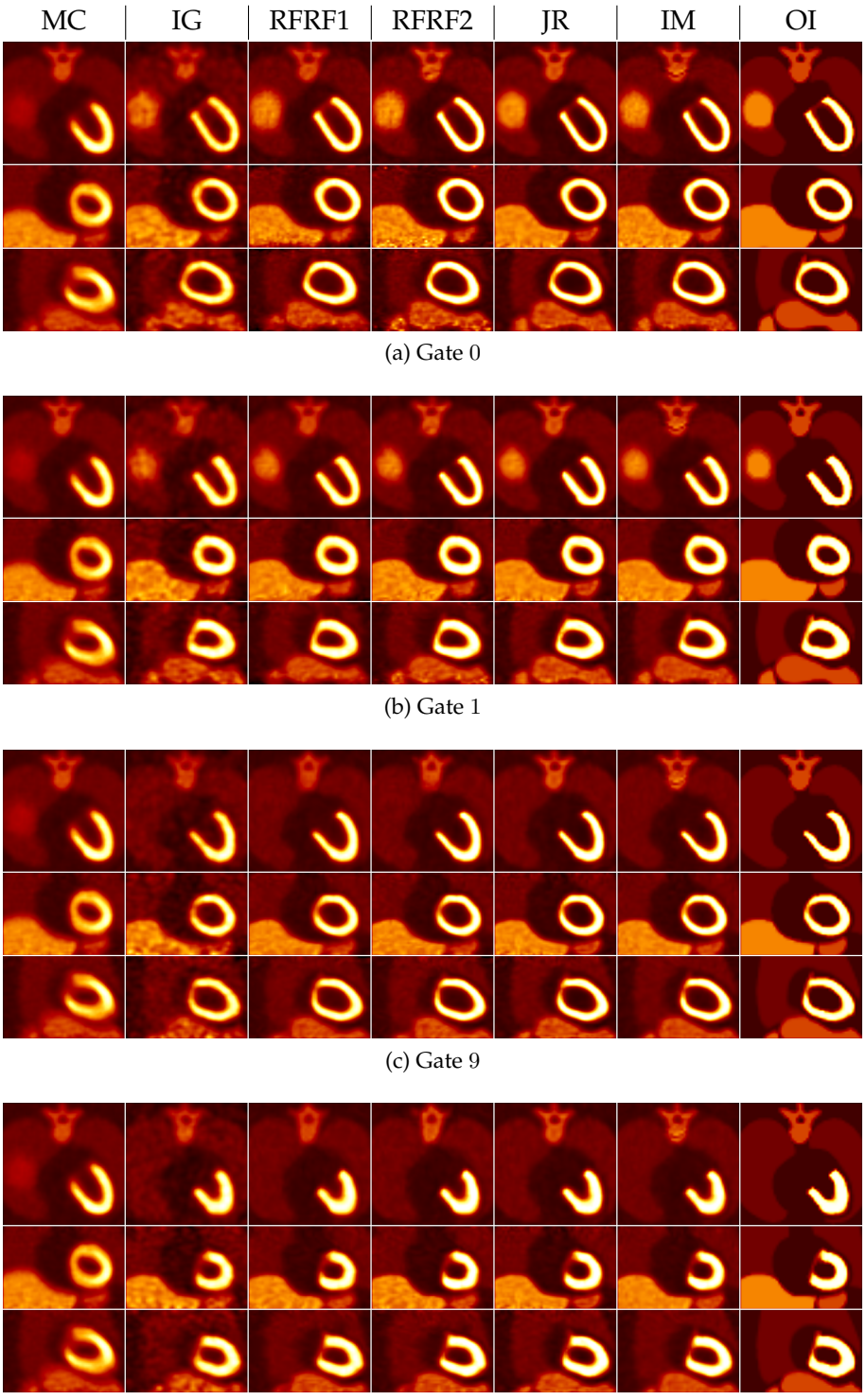
(a) Gate 0

(b) Gate 1

(c) Gate 9

(d) Gate 8

Figure 6.10.: Simulation, $9 \cdot 10^6$ counts: visual comparison of different reconstruction methods. JR-BS-MU is shown as an JR ambassador.



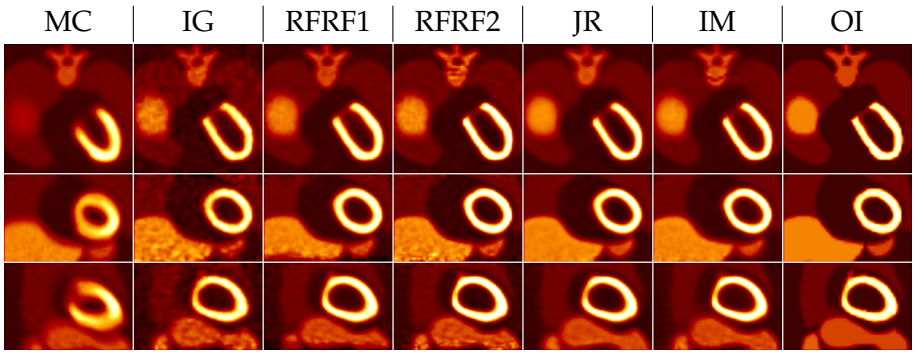
(a) Gate 0

(b) Gate 1

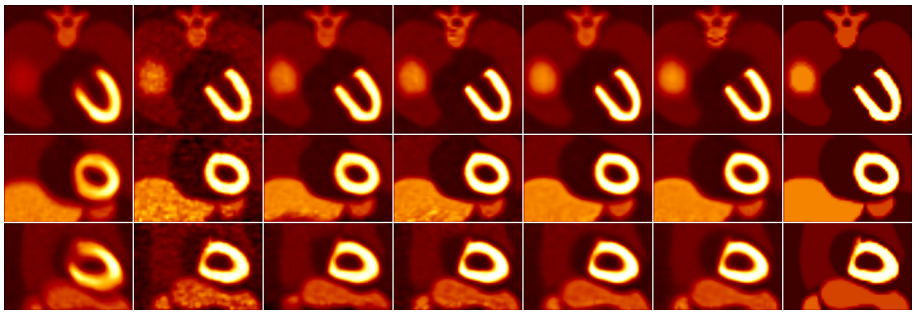
(c) Gate 9

(d) Gate 8

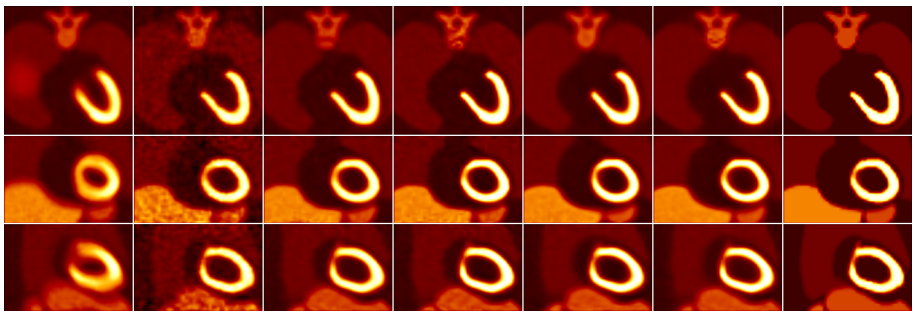
Figure 6.11.: Simulation, $4.5 \cdot 10^7$ counts: visual comparison of different reconstruction methods. JR-BS-MU is shown as an JR ambassador.



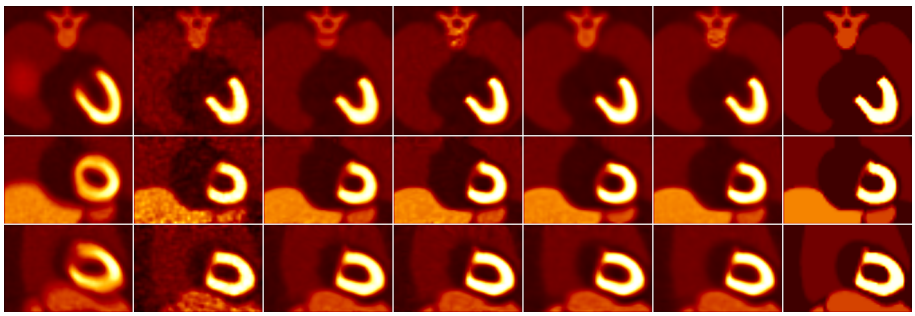
(a) Gate 0



(b) Gate 1

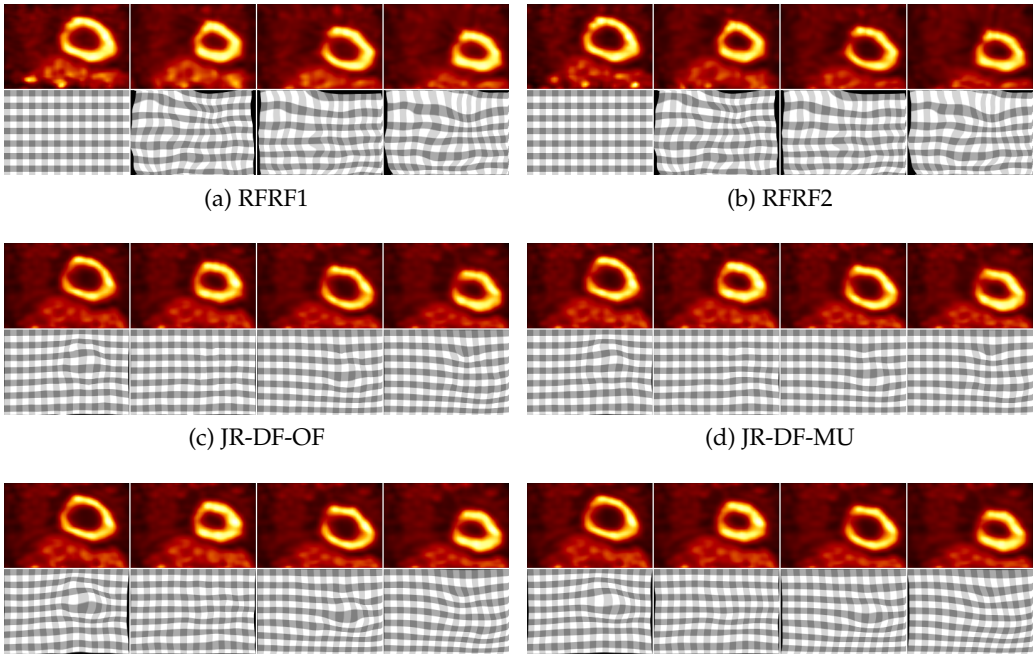


(c) Gate 9



(d) Gate 8

Figure 6.12.: Simulation, $2.25 \cdot 10^8$ counts: visual comparison of different reconstruction methods. JR-BS-MU is shown as a JR ambassador.

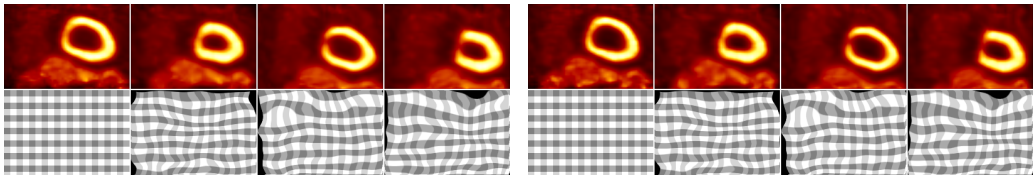


(a) RFRF1 (b) RFRF2

(c) JR-DF-OF (d) JR-DF-MU

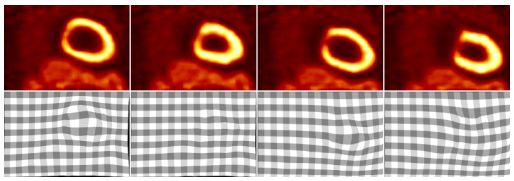
(e) JR-BS-OF (f) JR-BS-MU

Figure 6.13.: Simulation, $1.6 \cdot 10^6$ counts: visual comparison of the motion fields of gates 0, 1, 9 and 8 for different reconstruction methods. Note that the motion fields are not directly comparable one to another since they correspond to different reference gates.

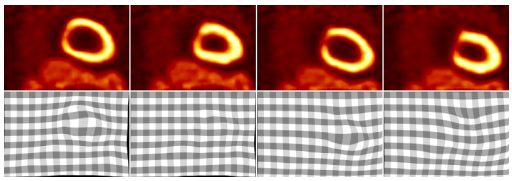


(a) RFRF1

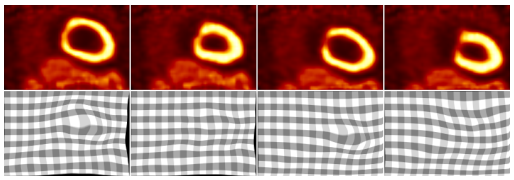
(b) RFRF2



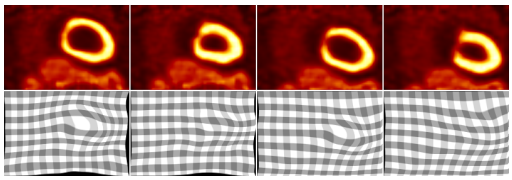
(c) JR-DF-OF



(d) JR-DF-MU

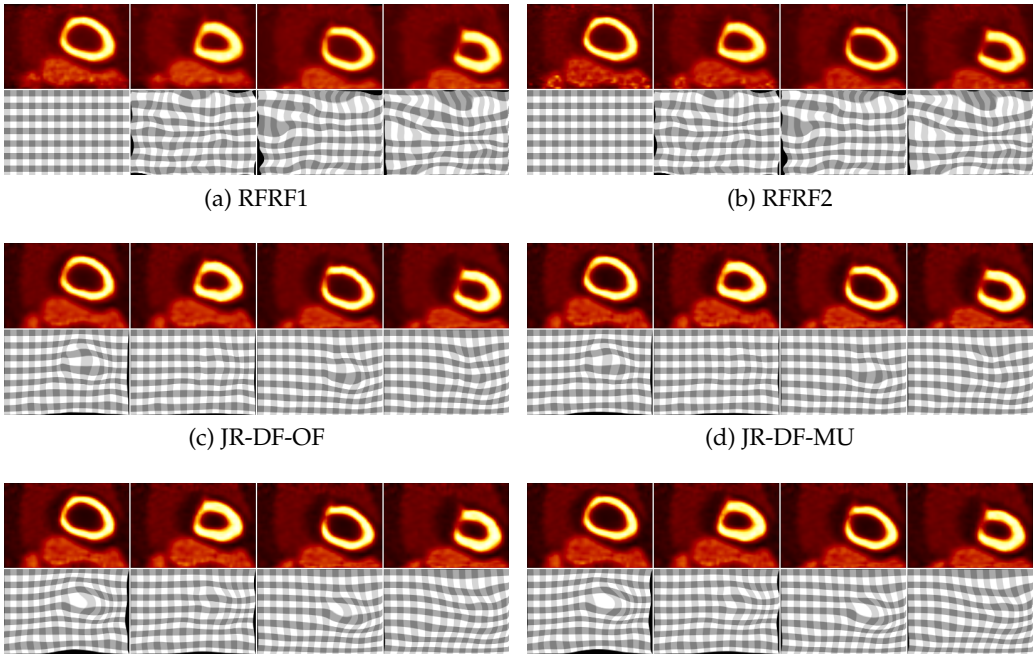


(e) JR-BS-OF



(f) JR-BS-MU

Figure 6.14.: Simulation, $9 \cdot 10^6$ counts: visual comparison of the motion fields of gates 0, 1, 9 and 8 for different reconstruction methods. Note that the motion fields are not directly comparable one to another since they correspond to different reference gates.

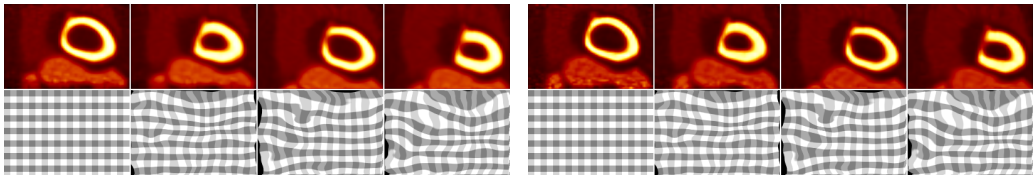


(a) RFRF1 (b) RFRF2

(c) JR-DF-OF (d) JR-DF-MU

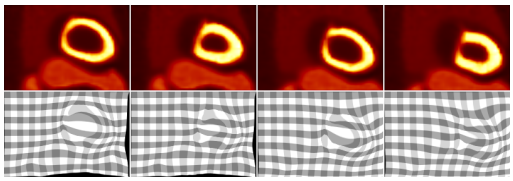
(e) JR-BS-OF (f) JR-BS-MU

Figure 6.15.: Simulation, $4.5 \cdot 10^7$ counts: visual comparison of the motion fields of gates 0, 1, 9 and 8 for different reconstruction methods. Note that the motion fields are not directly comparable one to another since they correspond to different reference gates.

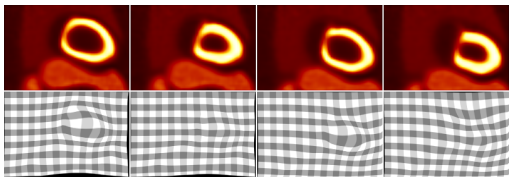


(a) RFRF1

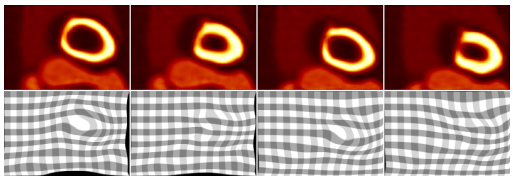
(b) RFRF2



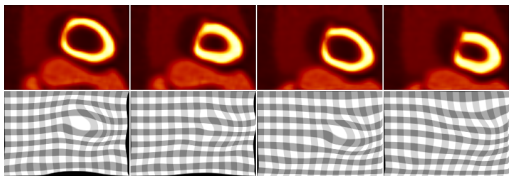
(c) JR-DF-OF



(d) JR-DF-MU

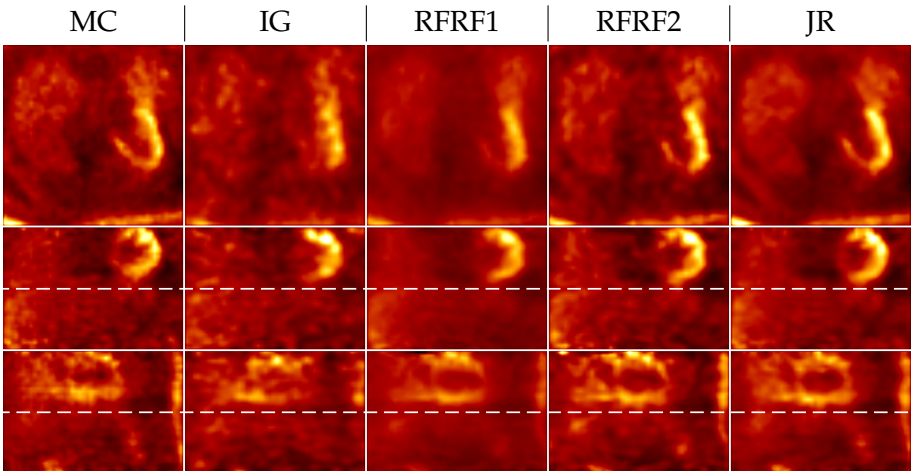


(e) JR-BS-OF

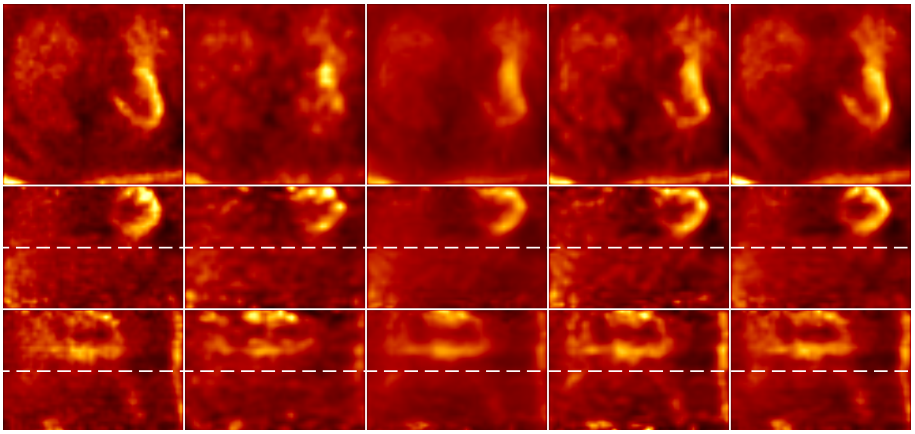


(f) JR-BS-MU

Figure 6.16.: Simulation, $2.25 \cdot 10^8$ counts: visual comparison of the motion fields of gates 0, 1, 9 and 8 for different reconstruction methods. Note that the motion fields are not directly comparable one to another since they correspond to different reference gates.

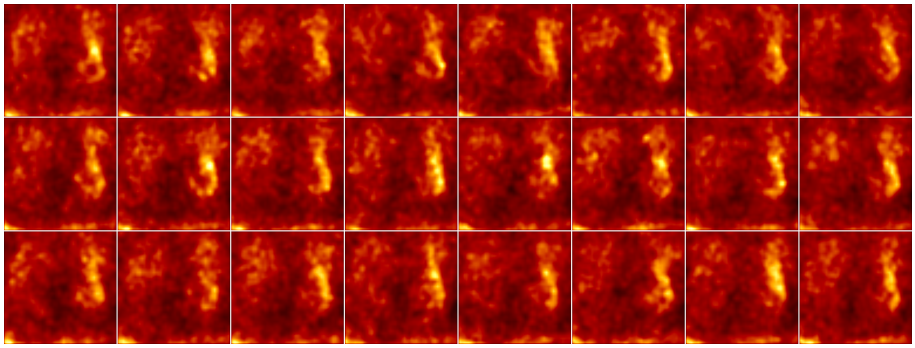


(a) Gate 11

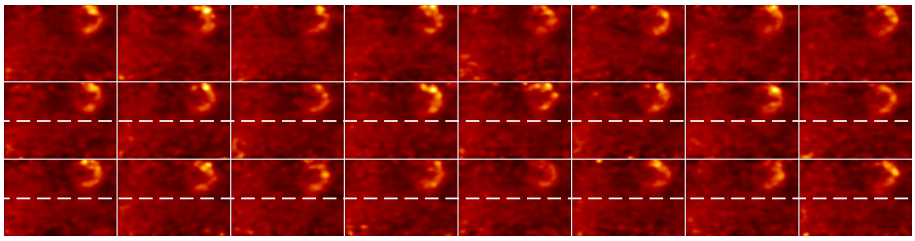


(b) Gate 12

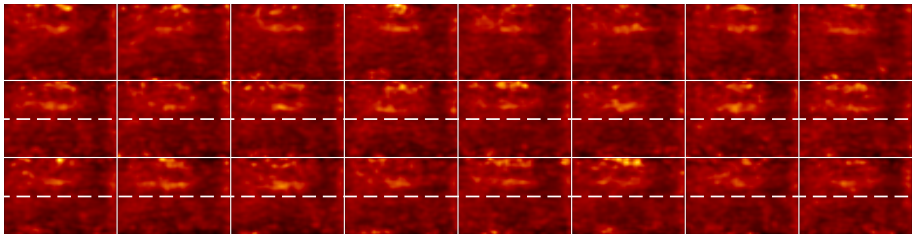
Figure 6.17.: Patient data: direct visual comparison of different reconstruction methods. JR-BS-MU is shown as an JR ambassador.



(a) Transverse View

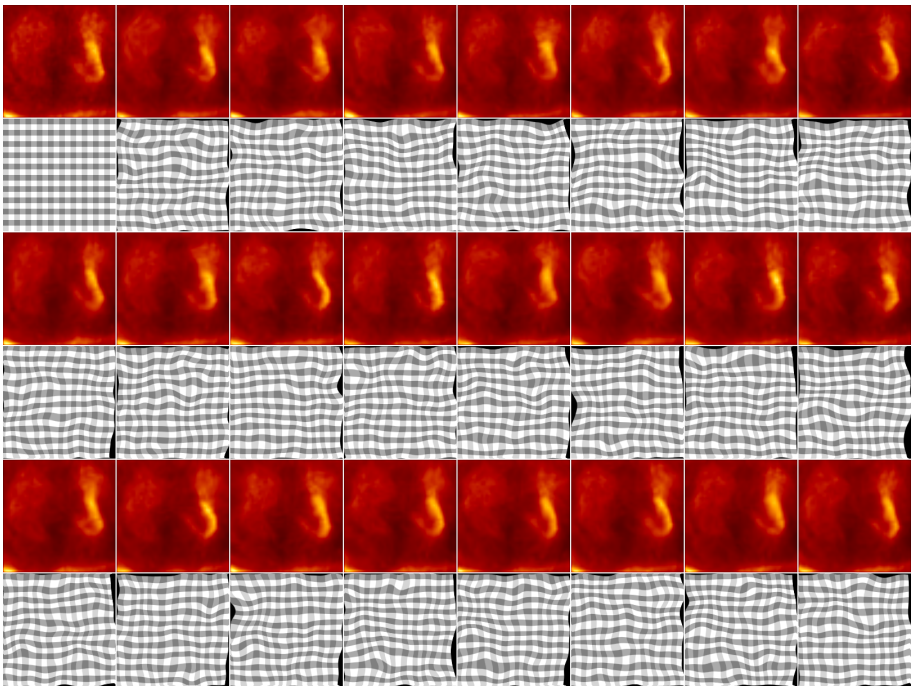


(b) Coronal View

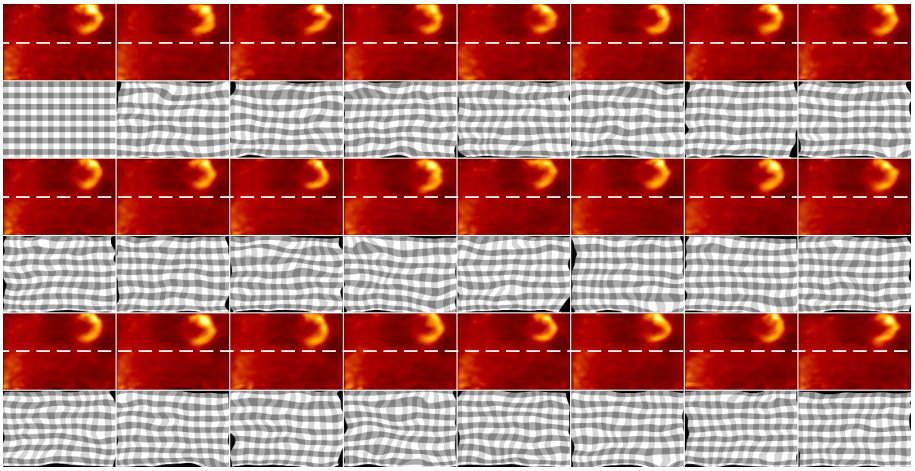


(c) Sagittal View

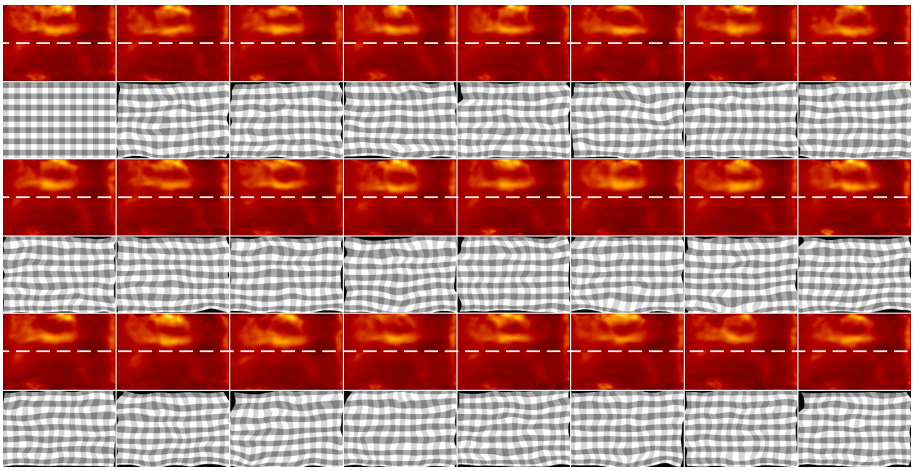
Figure 6.18.: Patient data: all 24 gates reconstructed by the IG method.



(a) Transverse View

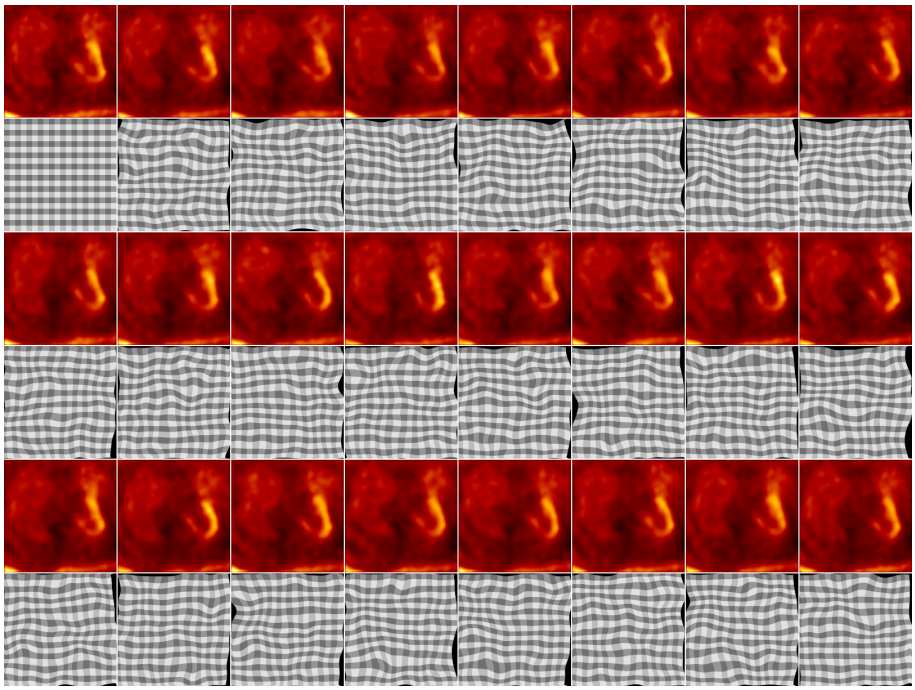


(b) Coronal View

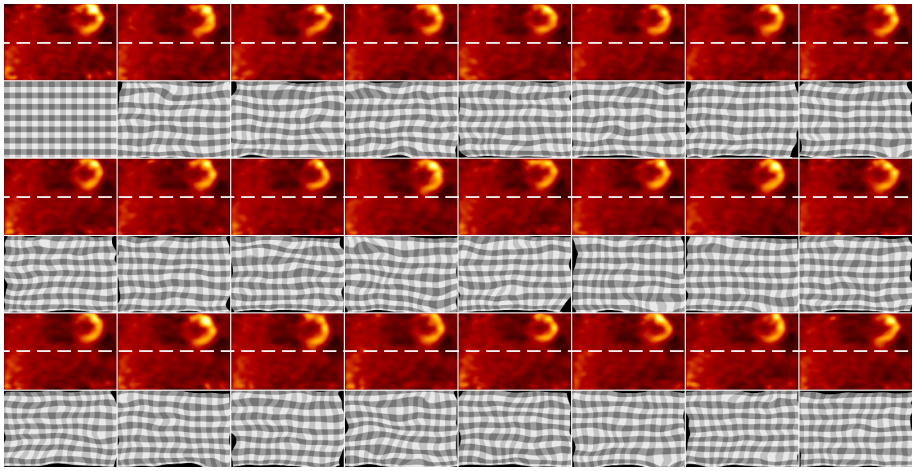


(c) Sagittal View

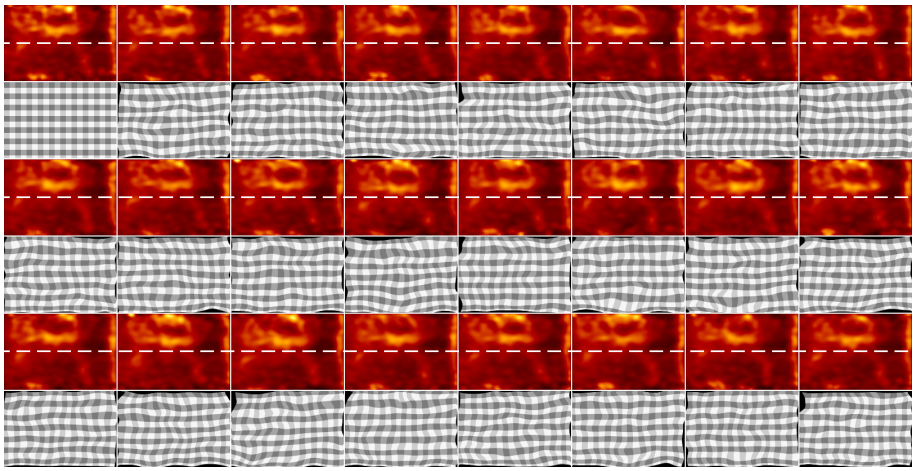
Figure 6.19.: Patient data: all 24 gates together with the respective motion field reconstructed by the RFRF1 method.



(a) Transverse View

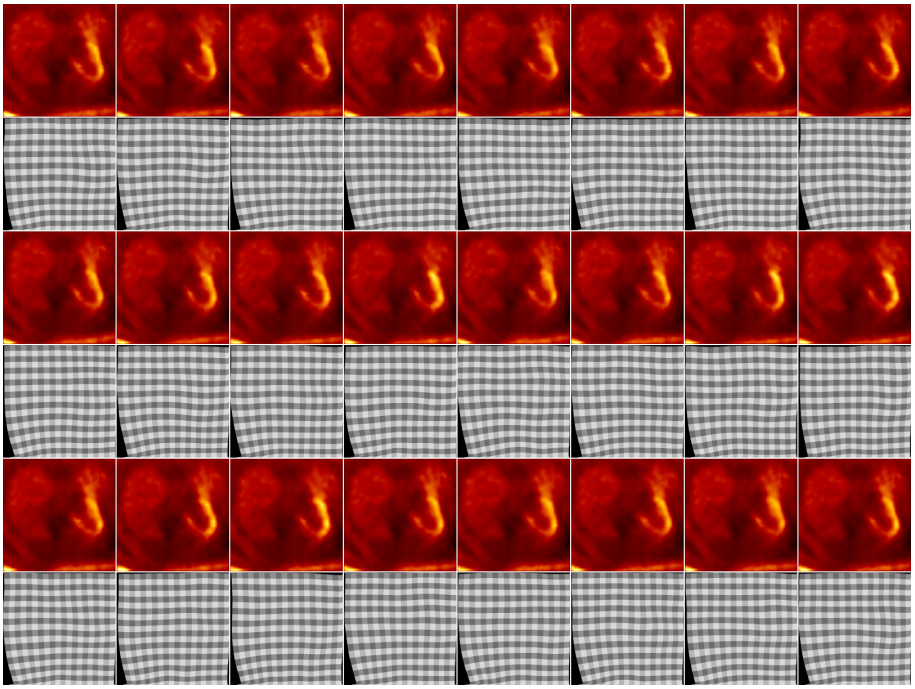


(b) Coronal View

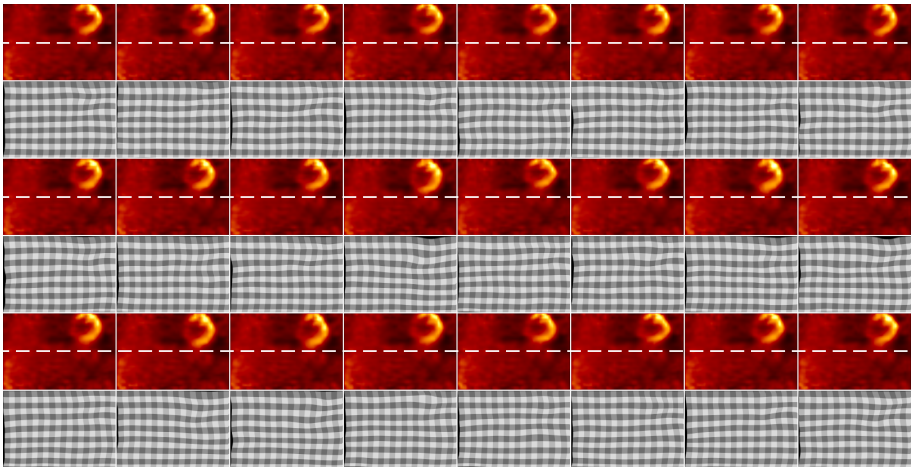


(c) Sagittal View

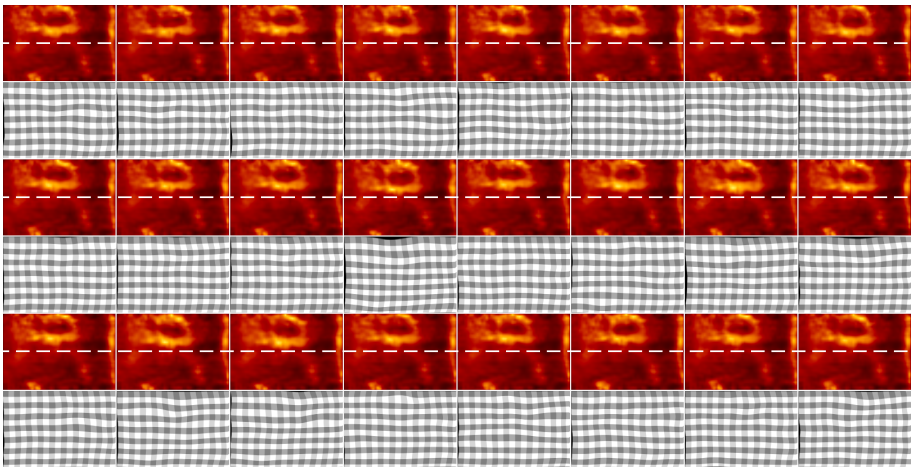
Figure 6.20.: Patient data: all 24 gates together with the respective motion field reconstructed by the RFRF2 method.



(a) Transverse View



(b) Coronal View



(c) Sagittal View

Figure 6.21.: Patient data: all 24 gates together with the respective motion field reconstructed by the JR-BS-MU method.

7. Conclusion and Future Work

In this chapter we will discuss some aspects related to our work which we consider important to investigate in future research. Some of these ideas were implemented in a premature manner already but could not be tested thoroughly.

7.1. Computational Burden

The joint reconstruction method as presented is highly computational intense, since in the motion update projections from image to measurement space and back-projections are needed for each inner iteration. On our system[1] the reconstruction for the JR-BS-MU method took 15:28 hours for the $1.8 \cdot 10^6$ counts case. We have made several attempts to further alleviate this computational burden (it was much higher initially), e. g. by using subset based motion updates (as similarly done in the image update). The preliminary results were not convincing enough in order to be included in this thesis.

J. Fessler has recognized this problem for joint reconstruction approaches and presents a solution based on optimization transfer techniques, using a penalized-likelihood objective function, in order to eliminate the computationally intense forward and back-projections with the system model from the motion updates [59].

7.2. Attenuation Correction

As stated in the description of our algorithm, we do not correct for attenuation. In the following we will discuss several ways to integrate attenuation correction in our joint reconstruction approach theoretically. In practice, all of these approaches need at least one attenuation map which corresponds to one of the gates. This is a major limitation since such attenuation maps are not easy to obtain. It depends on the chosen gating procedure whether it is feasible to find the corresponding attenuation map for a certain gate.

7.2.1. Integrating the Attenuation into the Model

From a mathematical point of view, the ideal solution consists of an integration of the attenuation map in the model. Of course, the attenuation map $\mu(\boldsymbol{x})$ then has to be transformed according to the respective gate, that is, we have some term $\mu(\boldsymbol{\varphi}(\boldsymbol{x}, t))$ in the objective function. This complicates the derivative of the objective function with respect to the motion function $\boldsymbol{\varphi}$, since not only the image f depends on $\boldsymbol{\varphi}$, but also the attenuation map μ.

[1] Intel Xeon E5430, 8 cores, 2.66GHz each, 16Gb RAM

In principle, these considerations should be manageable. A major problem however is to obtain an attenuation map which corresponds to the image frame. In the current implementation of JR this is impossible, since the image frame is a virtual frame: the image $f(\boldsymbol{x})$ never exists in reality (and thus neither does an attenuation map). Only the image $f(\boldsymbol{\varphi}(\boldsymbol{x}, t))$, which is the image transformed to a certain gate t, exists.

This problem could be overcome by choosing a reference gate, which is a relatively simple modification of our current JR algorithm (for the reference gate, one simply does not transform the image in the model). Note that by choosing a reference gate the performance of the JR algorithm might depend on the choice of the reference gate. A poorly chosen reference gate such as the maximal inhalation/exhalation state could lead to a decreased performance.

Then, the problem of obtaining an attenuation map which corresponds to the chosen reference gate remains. This is by no means a trivial task and depends on the gating procedure. If a relatively simple respiratory gating is done, the same respiratory gating can be done for the transmission scan and thus one obtains the attenuation map for a certain gate. For our more complex image based gating procedure, it is more complicated to find the gate to which an acquired attenuation map corresponds. One could think of image based methods which evaluate to which emission gate the transmission image is most similar to.

7.2.2. The Attenuation Map as a Constant

If one wants to avoid the hassle of recalculating the derivatives and testing the resulting code, a simpler solution might be to consider the attenuation map as a constant function which is given for all gates. The attenuation map then is still in the model, but it does not depend on the motion function $\boldsymbol{\varphi}$ and thus does not generate any problems in the derivation. The attenuation map is a constant when it is known for each gate in advance. We will discuss two ways in order to achieve this.

The first way may be feasible for relatively simple gating procedures like exclusively respiratory gating. Then, the same gating procedure can be applied to the CT projections, maybe acquired by a low dose CT scan in order not to harm the patient but still acquiring sufficient data for all gates. In the end, each set of CT projections is reconstructed individually and so gated attenuation maps can be generated from the reconstructed CT gates. For image based gating, one is confronted with the same problems as discussed previously.

The second way is to run a attenuation ignoring JR algorithm with the only aim to find the motion function. Knowing the motion, the attenuation map can be generated for all respective gates if a reference gate was chosen before and if an attenuation map for this reference gate is available. This second approach can be considered as a two-pass joint reconstruction algorithm: the first pass leads to the attenuation maps for all gates, and the second pass includes them as a constant function.

7.2.3. Ignoring Motion for Attenuation Correction

All of the previously discussed approaches have the same problem: they rely on an attenuation map which must perfectly fit to at least one of the gates. It is doubtable that this requirement can be achieved accurately. The following questions rise: can good results be obtained by simply assuming that the given attenuation map is correct for all gates, that is, ignoring the motion for attenuation correction? Is a mathematically correct treatment, based on a poorly chosen reference attenuation map, better than ignoring motion for attenuation?

7.3. True 4D Reconstruction

We earlier introduced gating as a necessary pre-processing step for all motion compensation techniques. Considering that joint reconstruction methods aim at a unified model which comprises all necessary aspects of motion compensated PET reconstruction, it seems that gating does not fit into this paradigm. Without gating, one has to work with non preprocessed data: the individual list-mode events. The result then would be a true 4D motion. With true 4D motion we refer to a motion function $\boldsymbol{\varphi}(\boldsymbol{x}, t)$, where t is not a natural number identifying a gate but rather a real number which corresponds to the time.

The joint reconstruction method as presented in this thesis is readily applicable to such a scenario. Instead of single events, we define short time frames (the shorter the closer we come to the assumption of using individual list-mode events). These frames are then treated as gates. In fact, they have the same key property as gates do: there is few motion within one frame (the shorter the frame is the better is this assumption fulfilled). As a first trial, we divided the data into one second time frames. Of course, if cardiac motion had to be resolved, shorter time frames would be necessary. Furthermore, we implemented a simple time scale smoothing in order to integrate information about the temporal smoothness of the motion into the reconstruction.

The reconstructed image and motion is viewable as a video [12]. It seems that the respiratory motion was correctly reconstructed, and as suspected, the cardiac motion is ignored since the time frames are too long.

7.4. Mass Preservation

Deforming an image can have undesirable consequences. In our formulation, the value of the integral over the deformed image

$$\int_{\Omega} f(\boldsymbol{\varphi}(\boldsymbol{x}, t))\, d\boldsymbol{x} \tag{7.1}$$

varies with the deformation: the motion function $\boldsymbol{\varphi}$ increases or decreases the total activity, unless it is a rigid motion. In PET however, the total activity should be constant since a pure compression or extension of the tissue does not change the overall amount

of dose. The increase or decrease of activity of course affects quantification of the reconstructed images. While this is of course not desirable, for small deformations as we are facing them in the real patient data images, the effect is negligible for diagnostics, in our opinion.

In the image registration community, the fact that the integral over an image should not change with the motion is commonly referred to as mass preservation [79, 192, 66]. A typical modification which leads to mass preservation is to include the absolute value of the determinant of the Jacobian of the motion field:

$$\int_{\boldsymbol{\varphi}^{-1}(\Omega)} f(\boldsymbol{\varphi}(\boldsymbol{x},t))\,|\det(\nabla\boldsymbol{\varphi})(\boldsymbol{x},t)|\,d\boldsymbol{x}\ . \tag{7.2}$$

Note that this formulation results directly from the rule of integration by substitution.

In the following, we will show how such a mass preservation approach can be included in our joint reconstruction approach. In our model, image and motion appear in the objective function $\mathcal{D}$ within the expected number of counts

$$\hat{g}(a,t,f,\boldsymbol{\varphi}) = \int H(a,\boldsymbol{x}) f(\boldsymbol{\varphi}(\boldsymbol{x},t)\,d\boldsymbol{x}\ . \tag{7.3}$$

We enforce mass preservation by

$$\hat{g}(a,t,f,\boldsymbol{\varphi}) = \int H(a,\boldsymbol{x}) f(\boldsymbol{\varphi}(\boldsymbol{x},t)\,|\det(\nabla\boldsymbol{\varphi})(\boldsymbol{x},t)|\,d\boldsymbol{x}\ . \tag{7.4}$$

Equation 7.4 can be rewritten as

$$\hat{g}(a,t,f,\boldsymbol{\varphi}) = \int H(a,\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)) f(\boldsymbol{x}) d\boldsymbol{x}\ . \tag{7.5}$$

It is easy to see that by replacing $\boldsymbol{x}$ by $\boldsymbol{\varphi}$ in equation (7.5), together with the rule for integration by substitution, one arrives at equation (7.4).

7.4.1. Gâteaux Derivative of $\mathcal{D}$ With Respect to f

In the following, we will recalculate the Gâteaux derivatives of $\mathcal{D}$ for the mass preserving formulation with respect to the image. From equation (4.10) we know that

$$d_f\mathcal{D}(f,\boldsymbol{\varphi};\eta) = \sum_{t,a} d_f\hat{g}(a,t,f,\boldsymbol{\varphi}) \left(1 - \frac{g(a,t,f,\boldsymbol{\varphi})}{\hat{g}(a,t,f,\boldsymbol{\varphi})}\right)\ .$$

We now need to calculate

$$d_f\hat{g}(a,t,f,\boldsymbol{\varphi}) = \frac{\partial}{\partial\varepsilon}\hat{g}(a,t,f+\varepsilon\eta)\Big|_{\varepsilon=0}\ , \tag{7.6}$$

which expands to either

$$d_f\hat{g}(a,t,f,\boldsymbol{\varphi}) = \frac{\partial}{\partial\varepsilon}\int H(a,\boldsymbol{x})[f+\varepsilon\eta](\boldsymbol{\varphi}(\boldsymbol{x},t))\,|\det(\nabla\boldsymbol{\varphi})(\boldsymbol{x},t)|\,d\boldsymbol{x}\Big|_{\varepsilon=0} \tag{7.7}$$

(by using equation 7.4), or

$$d_f\hat{g}(a,t,f,\boldsymbol{\varphi}) = \frac{\partial}{\partial\varepsilon}\int H(a,\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t))[f+\varepsilon\eta](\boldsymbol{x})d\boldsymbol{x}\Big|_{\varepsilon=0} \quad . \tag{7.8}$$

(by using equation 7.5). It seems much easier to use the second formulation, and we arrive at

$$d_f\hat{g}(a,t,f,\boldsymbol{\varphi}) = \frac{1}{T}\int H(a,\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t))\,\eta(\boldsymbol{x})\,d\boldsymbol{x} \quad . \tag{7.9}$$

Accordingly, we finally arrive at

$$d_f\mathcal{D}(f,\boldsymbol{\varphi};\eta) = \frac{1}{T}\int\sum_{t,a} H(a,\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t))\left(1-\frac{g(a,t,f,\boldsymbol{\varphi})}{\hat{g}(a,t,f,\boldsymbol{\varphi})}\right)\,\eta(\boldsymbol{x})\,d\boldsymbol{x} \quad . \tag{7.10}$$

7.4.2. Gâteaux Derivative of $\mathcal{D}$ With Respect to $\boldsymbol{\varphi}$

We start from equation (4.17):

$$d_{\boldsymbol{\varphi}}\mathcal{D}(f,\boldsymbol{\varphi};\boldsymbol{\eta}) = \sum_{t,a} d_{\boldsymbol{\varphi}}\hat{g}(a,t,f,\boldsymbol{\varphi};\boldsymbol{\eta})\left(1-\frac{g(a,t)}{\hat{g}(a,t,f,\boldsymbol{\varphi})}\right) \quad .$$

We need to expand

$$d_{\boldsymbol{\varphi}}\hat{g}(a,t,f,\boldsymbol{\varphi};\boldsymbol{\eta}) = \frac{\partial}{\partial\varepsilon}\hat{g}(a,t,f,\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta})\Big|_{\varepsilon=0} \quad , \tag{7.11}$$

which results in either

$$d_{\boldsymbol{\varphi}}\hat{g}(a,t,f,\boldsymbol{\varphi};\boldsymbol{\eta}) = \frac{1}{T}\int H(a,\boldsymbol{x})\,\frac{\partial}{\partial\varepsilon}f([\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta}](\boldsymbol{x},t))\,|\det(\nabla[\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta}])(\boldsymbol{x},t)|\,\Big|_{\varepsilon=0} d\boldsymbol{x} \tag{7.12}$$

(by using equation 7.4), or

$$d_{\boldsymbol{\varphi}}\hat{g}(a,t,f,\boldsymbol{\varphi};\boldsymbol{\eta}) = \frac{1}{T}\int\,\frac{\partial}{\partial\varepsilon}H(a,[\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta}]^{-1}(\boldsymbol{x},t))\,\Big|_{\varepsilon=0} f(\boldsymbol{x})\,d\boldsymbol{x} \quad . \tag{7.13}$$

(by using equation 7.5). Neither of the two formulations seems simple to handle. In equation (7.12), some way to deal with the absolute value has to be found. It could be either substituted by a similar but differentiable function, or simply omitted assuming that its value is greater than zero anyway (since the function has to be invertible this is a reasonable assumption). Then, the product rule yields

$$\begin{aligned}
&\frac{\partial}{\partial\varepsilon} f([\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta}](\boldsymbol{x},t))\,\det(\nabla[\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta}])(\boldsymbol{x},t)\Big|_{\varepsilon=0} \\
&\quad = \frac{\partial}{\partial\varepsilon} f([\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta}](\boldsymbol{x},t))\Big|_{\varepsilon=0} \det(\nabla\boldsymbol{\varphi})(\boldsymbol{x},t) \\
&\quad + f(\boldsymbol{\varphi}(\boldsymbol{x},t))\frac{\partial}{\partial\varepsilon}\det(\nabla[\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta}])(\boldsymbol{x},t)\Big|_{\varepsilon=0} \quad . \qquad (7.14)
\end{aligned}$$

Further derivation (see equation (4.20)) leads to

$$\begin{aligned}
&\frac{\partial}{\partial\varepsilon} f([\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta}](\boldsymbol{x},t))\,\det(\nabla\boldsymbol{\varphi}+\varepsilon\boldsymbol{\eta})(\boldsymbol{x},t)\Big|_{\varepsilon=0} \\
&\quad = \langle\nabla f(\boldsymbol{\varphi}(\boldsymbol{x},t)),\boldsymbol{\eta}(\boldsymbol{x},t)\rangle\,\det(\nabla\boldsymbol{\varphi})(\boldsymbol{x},t) \\
&\quad + f(\boldsymbol{\varphi}(\boldsymbol{x},t))\,[\det(\nabla\boldsymbol{\varphi})\,\mathrm{tr}((\nabla\boldsymbol{\varphi})^{-1}\nabla\boldsymbol{\eta})](\boldsymbol{x},t) \quad . \qquad (7.15)
\end{aligned}$$

At this point, one could start with the discretization by replacing $\boldsymbol{\eta}$ with the respective derivative of the discretized motion function (see section 4.4 for details). Considering that $\boldsymbol{\eta}$ appears two times, once with a differential operator, this task seems not trivial. We leave it open as a challenge for future PhD students.

Using the formulation from equation (7.13) seems complicated, too. Applying the chain rule leads to

$$d_{\boldsymbol{\varphi}}\hat{g}(a,t,f,\boldsymbol{\varphi};\boldsymbol{\eta}) = \frac{1}{T}\int\langle\nabla_{\boldsymbol{x}}H(a,\boldsymbol{\varphi}^{-1}(\boldsymbol{x},t)),\boldsymbol{\eta}^{-1}(\boldsymbol{x},t)\rangle\,f(\boldsymbol{x})\,d\boldsymbol{x} \quad . \qquad (7.16)$$

Also in this formulation $\boldsymbol{\eta}$ has to be replaced by the respective derivative and does not appears in form of its inverse. Furthermore, it would be necessary to calculate $\nabla_{\boldsymbol{x}}H$, which also seems complicated. At this point no conclusive indication can be given which of the two approaches is easier to follow.

7.5. PET-MR

PET-MR is still in research stage, and just recently the first simultaneous whole body PET-MR prototype, produced by Siemens, has been shipped to a hospital in Munich [126]. Other prototypes, often PET inlays which are placed within existing MRI scanners, have been presented [185, 180, 181, 99]. Such systems must not be confused with non-simultaneous hybrid PET-MR scanners [145]. Many of the advantages of PET-MR - including motion estimation - are only valid for simultaneous acquisitions.

The question is what future role can PET-MR play with respect to motion compensation. Very few publications dealing with this topic have been presented so far (see sections 3.2.1.1.4 and 3.2.2.1.4), and thus one can only speculate about the usefulness of

combined PET-MR machines with respect to motion compensation. In the reviewed articles, MR has been used exclusively as a motion estimation method. The acquired motion is then used in order to correct the PET data. One big disadvantage of such a usage is that special sequences which are adequate for motion estimation have to be used for MR, and thus, other studies which require other sequences can not be performed. To put it in a nutshell: motion information is traded for other useful information.

We will shortly discuss how PET-MR could be integrated in a joint reconstruction framework, maybe with the perspective of not requiring a specific imaging protocol. The idea is to use MR as an image based gating procedure. MR is not yet able to acquire high quality 3D+t image sequences, but dynamic 2D+t images can be acquired. For periodic motions, one can assume that if two 2D frames are identical, then also the rest of the volume is at least similar. With this assumption, image based gating can be achieved by partitioning 2D+t image sequences into sets of similar images, which define the gates. In fact, the same gating procedure as presented in section 5.2.1 could be used for that purpose. In this sense, the MR scanner would be used as a highly precise gating device. At the same time, the possibility to use MR for tasks apart from motion estimation is given. The gated data is then used as an input for the JR method, just as with any other gating method.

7.6. Conclusion

We present a method that jointly reconstructs both image and motion from gated PET data. The image can be transformed to the desired gate by the reconstructed motion field. Our method consists of optimizing an objective function which is derived from a linear dynamic imaging model (including motion). Since statistical data of the whole acquisition is used, the noise of the reconstructed image does not increase with the number of gates.

Regularization of the motion function plays an important role in any motion compensation algorithm. We compare two different regularization approaches: one which integrates the regularization in the objective function, and another one which is included within the optimization procedure itself. While the first introduces a regularization parameter which influences the result of the motion compensation process, the latter does not need such a regularization parameter and is therefore easier to handle in practice.

In an exhaustive simulation study, the presented joint reconstruction method is compared to several other motion compensation methods. According to the chosen figure of merit, it is superior to all compared methods for all count scenarios. The distance to other methods becomes bigger for low counts scenarios, which we attribute to the fact that our method is based on a complete and physically motivated imaging model.

Future fields of research have been identified, for some of them initial results, which did not make it into the core thesis since they are not yet mature enough, are presented. An important issue is the computational burden of our presented method. It seems that there may be solutions to this problem based on optimization transfer. Apart from that, since forward- and back-projection of the system model are parallelizable, one can always count on the exponential increase of computational power (compare Moore's law, which

predicts the duplication of computational power every one to two years [141, 140]). Attenuation has been identified as a problem which is difficult to handle for any motion compensation algorithm. The reason for this does not lie in the mathematical procedures themselves but rather in the fact that attenuation images need to be available at least for one of the gates, which seems difficult to achieve. Finally, a better clinical evaluation is needed. The clinical images shown in this thesis have to be understood as a proof-of-concept rather than a thorough analysis or comparison.

We hope that we could encourage at least some of the readers of this thesis to carry out further investigations on any of those open issues.

Part III.

Appendix

A. Mathematical Notations

A.1. Scalars and Vectors

Scalar quantities like constants

$$a \in \mathbb{R} \tag{A.1}$$

or functions

$$b : \mathbb{R}^M \mapsto \mathbb{R} \tag{A.2}$$

are printed non-bold. Vectorial quantities like constants

$$\boldsymbol{c} \in \mathbb{R}^N \tag{A.3}$$

or functions

$$\boldsymbol{d} : \mathbb{R}^M \mapsto \mathbb{R}^N \tag{A.4}$$

are printed in bold. The i-th element of $\boldsymbol{d}$,

$$[\boldsymbol{d}]_i \ , \tag{A.5}$$

is represented by square brackets in the subscript. Matrices like e. g.

$$\boldsymbol{E} \tag{A.6}$$

are written in bold capital letters. The element at row i and column j is accessed by

$$[\boldsymbol{E}]_{ij} \ . \tag{A.7}$$

The dot product of two vectors $\boldsymbol{c_1} \in \mathbb{R}^N$ and $\boldsymbol{c_2} \in \mathbb{R}^N$ is written as

$$\langle \boldsymbol{c_1}, \boldsymbol{c_2} \rangle = [\boldsymbol{c_1}]_0 [\boldsymbol{c_2}]_0 + \ldots + [\boldsymbol{c_1}]_{N-1} [\boldsymbol{c_2}]_{N-1} \ . \tag{A.8}$$

Note that the vector product is also often written as $\boldsymbol{c_1} \cdot \boldsymbol{c_2}$ in the literature.

A.2. Derivatives

The partial derivative with respect to the i-th argument of function b is denoted by

$$\frac{\partial}{\partial_i} b \ . \tag{A.9}$$

The Jacobi matrix for a vectorial function d (as defined above) is defined by

$$\nabla \boldsymbol{d} = \begin{pmatrix} \frac{\partial}{\partial_0}[\boldsymbol{d}]_0 & \cdots & \frac{\partial}{\partial_{M-1}}[\boldsymbol{d}]_0 \\ \vdots & \ddots & \vdots \\ \frac{\partial}{\partial_0}[\boldsymbol{d}]_{N-1} & \cdots & \frac{\partial}{\partial_{M-1}}[\boldsymbol{d}]_{N-1} \end{pmatrix} \ . \tag{A.10}$$

The operator ∇ applied to a scalar quantity like b then results in a column vector

$$\nabla b = \left(\frac{\partial}{\partial_0} b, \ldots, \frac{\partial}{\partial_{M-1}} b \right)^{\mathsf{T}} \ . \tag{A.11}$$

Sometimes a subscript for the operator ∇ will be used in order to define the parameters with respect to which the derivative should be taken. We will not give a strict formal definition for this convenience notation and therefore it should be possible to deduce it from the context. For example, $\nabla_{\boldsymbol{c}}$ means that the partial derivatives within the Jacobian should be with respect to the argument $\boldsymbol{c}$.

A.3. Calculus of Variations

The Gâteaux derivative of a functional $\mathcal{F} : (\mathbb{R}^M \mapsto \mathbb{R}^N) \mapsto \mathbb{R}$ in direction $\boldsymbol{\eta} : \mathbb{R}^M \mapsto \mathbb{R}^N$ at $\boldsymbol{d}$ is written as

$$d\mathcal{F}(\boldsymbol{d}; \boldsymbol{\eta}) \tag{A.12}$$

Similar to the Jacobian, if the functional depends on several functions, a subscript can denote the respective function.

B. Gâteaux Derivative

We will introduce the Gâteaux derivative as a generalization of the directional derivative. We show how the derivative concept of a scalar valued function can be extended to vector valued functions and then to functionals. In order to make it easy to see the analogy between the definitions, we will use the same symbol in each of the following three sections. The following three sections have to be read isolated from the rest of the thesis regarding the symbol choice.

B.1. Scalar Derivative

Let f be a function

$$f : \mathbb{R} \mapsto \mathbb{R} \tag{B.1}$$
$$x \mapsto f(x) \ . \tag{B.2}$$

that maps a real number x to a real number $f(x)$. The scalar derivative of f with respect to x is then defined as

$$\frac{d}{dx} f(x) := \lim_{\varepsilon \to 0} \frac{f(x+\varepsilon) - f(x)}{\varepsilon} \ . \tag{B.3}$$

B.2. Directional Derivative

Now, let f be a function

$$f : \mathbb{R}^d \mapsto \mathbb{R} \tag{B.4}$$
$$\boldsymbol{x} \mapsto f(\boldsymbol{x}) \ . \tag{B.5}$$

The function f maps a vector $\boldsymbol{x} \in \mathbb{R}^d$ to a scalar value $f(\boldsymbol{x})$. The directional derivative is defined analogously to the scalar derivative as

$$df(\boldsymbol{x}; \boldsymbol{\nu}) := \lim_{\varepsilon \to 0} \frac{f(\boldsymbol{x} + \varepsilon \boldsymbol{\nu}) - f(\boldsymbol{x})}{\varepsilon} \ . \tag{B.6}$$

Geometrically, the directional derivative is the slope of a function in a direction $\boldsymbol{\nu}$. The absolute value of $\boldsymbol{\nu}$ does not change the directional derivative.

It is important for later considerations that the directional derivative can be expressed in terms of a scalar derivative:

$$df(\boldsymbol{x};\boldsymbol{\nu}) = \frac{\partial}{\partial\varepsilon} f(\boldsymbol{x}+\varepsilon\boldsymbol{\nu})\bigg|_{\varepsilon=0} \quad . \tag{B.7}$$

This can be easily seen by making use of the definition of the scalar derivative from equation (B.3):

$$\frac{\partial}{\partial\varepsilon} f(\boldsymbol{x}+\varepsilon\boldsymbol{\nu})\bigg|_{\varepsilon=0} = \lim_{\delta\to 0} \frac{f(\boldsymbol{x}+(\varepsilon+\delta)\boldsymbol{\nu}) - f(\boldsymbol{x}+\varepsilon\boldsymbol{\nu})}{\delta}\bigg|_{\varepsilon=0} \quad . \tag{B.8}$$

By setting ε to zero and interchanging the role of ε and δ, one arrives at the definition of the directional derivative from equation (B.6).

This equality can be used in order to relate the directional derivative to the gradient. Together with the chain rule we get the following equality:

$$df(\boldsymbol{x};\boldsymbol{\nu}) = \langle \nabla f(\boldsymbol{x}), \boldsymbol{\nu}\rangle \quad . \tag{B.9}$$

B.3. Gâteaux Derivative

Now, let f be a functional

$$f : (\mathbb{R}^{d_1} \mapsto \mathbb{R}^{d_2}) \mapsto \mathbb{R} \tag{B.10}$$

$$\boldsymbol{x} \mapsto f(\boldsymbol{x}) \quad . \tag{B.11}$$

The functional f is a function that maps a function $\boldsymbol{x} : \mathbb{R}^{d_1} \mapsto \mathbb{R}^{d_2}$ to a scalar value $f(\boldsymbol{x})$. The Gâteaux derivative is in many terms very similar to the directional derivative. For a coarse understanding, one can simply replace the term "vector" with "function", and all previously made derivations arise analogously. The Gâteaux derivative is then defined as

$$df(\boldsymbol{x};\boldsymbol{\eta}) := \lim_{\varepsilon\to 0} \frac{f(\boldsymbol{x}+\varepsilon\boldsymbol{\eta}) - f(\boldsymbol{x})}{\varepsilon} \quad . \tag{B.12}$$

There is no direct geometrical interpretation as in the case of the directional derivative. Directions are now functions, and one wants to measure the change of the functional for a small variation $\boldsymbol{\eta}$ of function $\boldsymbol{x}$.

In order to make this definition manageable, we will express it in terms of a scalar derivative:

$$df(\boldsymbol{x};\boldsymbol{\eta}) = \frac{\partial}{\partial\varepsilon} f(\boldsymbol{x}+\varepsilon\boldsymbol{\eta})\bigg|_{\varepsilon=0} \quad . \tag{B.13}$$

This can be easily seen by making use of the definition of the scalar derivative from equation (B.3):

$$\frac{\partial}{\partial \varepsilon} f(\boldsymbol{x} + \varepsilon \boldsymbol{\eta}) \bigg|_{\varepsilon=0} = \lim_{\delta \to 0} \frac{f(\boldsymbol{x} + (\varepsilon + \delta)\boldsymbol{\eta}) - f(\boldsymbol{x} + \varepsilon \boldsymbol{\eta})}{\delta} \bigg|_{\varepsilon=0} . \tag{B.14}$$

By setting ε to zero and interchanging the role of ε and δ, one arrives at the definition of the Gâteaux derivative from equation (B.12).

C. Symbols and Abbreviations

Table C.1.: Most important symbols

Symbol	Explanation
$f, \boldsymbol{f}$	Image Subject to Reconstruction
$\varphi, \boldsymbol{\Phi}$	Motion Subject to Reconstruction
$g, \boldsymbol{g}$	Measured Data
$\hat{g}, \hat{\boldsymbol{g}}$	Expected Data
H	System Response Function
$\boldsymbol{H}$	System Matrix
$\boldsymbol{H_d}$	Detector Efficiency Factor
$\boldsymbol{H_a}$	Attenuation Factor
$\boldsymbol{H_g}$	Geometrical Factor
$\boldsymbol{H_p}$	Positron Range Factor

Table C.2.: Abbreviations

Abbreviation	Explanation
blob	Generalized Kaiser-Bessel Window Functions
CC	Correlation Coefficient
CSR	Compressed Sparse Row Format
CT	Computed Tomography
ECG	Electrocardiogram
EPI	Echo Planar Imaging (an MRI sequence)
ET	Emission Tomography
FBP	Filtered Back-Projection
FLASH	Fast Low Angle Shot (an MRI sequence)
FOM	Figure of Merit
FOV	Field of View
FWHM	Full Width at Half Maximum
GPU	Graphics Processing Unit
KKT	Karush-Kuhn-Tucker
LOR	Line of Response
LSO	Lutetium Oxyorthosilicate (a common PET scintillator used in the detectors)
MAF	Multiple Acquisition Frames
MAP	Maximum A Posteriori
MC	A motion compensation method that works on motion contaminated data (ignoring the motion information)
MI	Mutual Information
ML	Maximum Likelihood
ML-EM	Maximum Likelihood Expectation-Maximization
MR, MRI	Magnetic Resonance Imaging
NNZ	Number of Non Zero Elements of a Matrix
JR	Joint Reconstruction
JR-BS	Joint Reconstruction using a B-Spline Parametrization for the Motion Function
JR-DF	Joint Reconstruction using Displacement Fields
PET	Positron Emission Tomography (the imaging modality) / Positron Emission Tomograph (the scanner)
PMT	Photomultiplier Tube
RFRF	Registration and Fusion of Reconstructed Frames, a motion compensation method
ROI	Region of Interest
SNR	Signal to Noise Ratio
SPECT	Single Photon Emission Tomography
SSD	Sum of Squared Differences
SUV	Standard Uptake Value

Table C.3.: Terminology

Term	Explanation
sagittal	planes dividing the body into a left and a right part
coronal	planes dividing the body into a front and a back part
transverse/axial	planes dividing the body into an upper and a lower part
systole	contraction of the ventricular myocardium
diastole	relaxation of the ventricular myocardium

D. List of Publications

D.1. International Journals

Joint Reconstruction of Image and Motion in Gated Positron Emission Tomography
Moritz Blume, Axel Martinez-Möller, Andreas Keil, Nassir Navab, Magdalena Rafecas
IEEE Transactions on Medical Imaging
Volume 29, Issue 11, Pages 1892-1906, November 2010, DOI: 10.1109/TMI.2010.2053212

We present a novel intrinsic method for joint reconstruction of both image and motion in positron emission tomography (PET). Intrinsic motion compensation methods exclusively work on the measured data, without any external motion measurements. Most of these methods separate image from motion estimation: They use deformable image registration/optical flow techniques in order to estimate the motion from individually reconstructed gates.

Then, the image is estimated based on this motion information. With these methods, a main problem lies in the motion estimation step, which is based on the noisy gated frames. The more noise is present, the more inaccurate the image registration becomes. As we show both visually and quantitatively, joint reconstruction using a simple deformation field motion model can compete with state-of-the-art image registration methods which use robust multilevel B-spline motion models.

D.2. International Conferences

Joint Reconstruction of Image and Motion for PET: Displacement Fields Versus a B-Spline Motion Model
Moritz Blume, Andreas Keil, Nassir Navab, Magdalena Rafecas
Poster Presentation at IEEE Medical Imaging Conference in Knoxville, Tennessee, USA.
IEEE Nuclear Science Symposium Conference Record, October 2010

Joint reconstruction methods of image and motion for Emission Tomography have emerged recently. These methods usually consist in optimizing an objective function which measures the similarity of an estimated data vector to the measured one according to the given estimates for image and motion. Since image reconstruction in ET is dealing with highly noisy data, a robust motion model together with an effective regularization scheme is necessary. In this paper, we compare two joint reconstruction methods which differ in the used motion model: a displacement field model (JRDF) and a B-spline model (JRBS). In the quantitative analysis, JRBS provides a higher maximal correlation coefficient (MCC) than JRDF. Additionally, the MCC of JRBS is located at a lower noise level.

These quantitative results are confirmed by a visual comparison, in which JRBS provides straighter edges and a smoother displacement field. We conclude that the B-spline motion model is promising to provide better robustness against noise.

Simultaneous Reconstruction of Image and Motion in Gated Positron-Emission-Tomography
Moritz Blume, Andreas Keil, Magdalena Rafecas, Nassir Navab
Poster Presentation at IEEE Medical Imaging Conference in Orlando, Florida, USA.
IEEE Nuclear Science Symposium Conference Record, Pages 5485-5487, October 2009, DOI: 10.1109/NSSMIC.2009.5401958, ISBN: 978-1-4244-3962-1

We present a novel method for joint reconstruction of both image and motion in positron-emission-tomography (PET). Most of nowadays motion compensation methods consist of two completely separated steps: (i) motion estimation and (ii) image estimation. A major drawback of these methods lies in the motion estimation step, since it is completely based on the usually noisy individually reconstructed gates. As we show in a simulation study, a joint reconstruction approach alleviates this drawback and results in both visually and quantitatively better image quality. We attribute these results to the fact that for motion estimation always the currently best available image estimate is used and vice versa. Additionally, results for dual respiratory and cardiac gated patient data are presented.

Joint Reconstruction of Image and Motion in Gated Positron-Emission-Tomography
Moritz Blume, Axel Martinez-Möller, Andreas Keil, Nassir Navab, Magdalena Rafecas
Oral Presentation at Fully 3D Conference in Beijing, China.
Proceedings of the 10th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine (Fully3D), Pages 118-121, September 2009, URL: `http://www.fully3d2009.org/download/proceedings_2009.pdf`

We present a novel method for joint reconstruction of both image and motion in positron-emission-tomography (PET). Most other methods separate image from motion estimation: They use deformable image registration/optical flow techniques in order to estimate the motion from individually reconstructed gates. Then, the image is estimated based on this motion information. With these methods, a main problem lies in the motion estimation step, which is based on the noisy gated frames. The more noise is present, the more inaccurate the image registration becomes. As we show in a simulation study, our joint reconstruction approach overcomes these drawbacks and results in both visually and quantitatively better image quality. We attribute these results to the fact that for motion estimation always the currently best available image estimate is used and vice versa. Additionally, results for real dual respiratory and cardiac gated patient data are presented.

ML-EM Implementation on a GPU: Avoiding Simultaneous Read-Modify-Write Processes
Thomas Felder, Moritz Blume, Josep Oliver, Magdalena Rafecas
Poster Presentation at HPIR Workshop in Beijing, China.

Proceedings of the 2nd Workshop on High Performance Image Reconstruction (HPIR), Pages 65-68, September 2009, URL: `http://www.fully3d2009.org/download/proceedings_2009.pdf`

Iterative image reconstruction algorithms for positron emission tomography (PET) became more and more common in the last decade. The reconstruction time of these computational intensive approaches can be reduced using graphics processing units (GPU). We implemented the ML-EM algorithm to reconstruct measurement data of a Biograph Sensation 16 PET/CT scanner (Siemens). To build the system matrix the Siddon's algorithm was used. The implementation of the back-projection operation on the GPU showed a possible data loss due to a simultaneous read-modify-write process of parallel threads. In this work we analyze the problem and show that this data loss can lead to worse image quality when the probability of the simultaneous memory access increases. We have developed several strategies on the GPU; a straight forward implementation, one that reduces the probability and one that avoids simultaneous access completely applying atomic functions. Our fastest GPU implementation is 33 faster than the CPU reconstruction.

Blind Motion Compensation for Positron-Emission-Tomography
Moritz Blume, Andreas Keil, Nassir Navab, Magdalena Rafecas
Oral Presentation at SPIE Conference in Orlando, Florida, USA.
Proceedings of the SPIE, Volume 7258, Pages 72580T-72580T-8, Feburary 2009

A major problem of high-resolution positron-emission-tomography (PET) are subject movements during acquisition. We propose a new motion compensation algorithm called "Blind Motion-Compensated Reconstruction" (BMCR) that is able to deal with frames of extremely low statistics in the case of smooth motion. Our algorithm reconstructs both image and rigid motion just from the recorded data and does not need any external motion tracking. This is achieved by combining image reconstruction and motion compensation into one mathematical framework which consists of a cost functional and an optimization method. The cost functional basically consists of a difference term which ensures consistency of the estimated parameters to the model and some regularization terms which render the problem mathematically well-posed. The optimization method aims at finding a pair of image and transformation/motion such that the cost functional is minimal. We believe that such a combined framework can overcome problems of existing algorithms which separate reconstruction and motion compensation. These algorithms usually try to get the motion information by registering reconstructed frames one to each other (in image space). Their main drawback is that the registration step is likely to be of low accuracy or even fail completely for low-statistics frames. We aim at filling this gap with our combined framework for registration and reconstruction. Initial results are promising and show that the quality of images reconstructed by BMCR for motion-contaminated data is significantly superior to a maximum-likelihood expectation-maximization (ML-EM) reconstruction for motion-contaminated data and even comparable to a ML-EM reconstruction for motion-free data.

Combined Motion Compensation and Reconstruction for PET
Moritz Blume, Magdalena Rafecas, Sibylle Ziegler, Nassir Navab
Oral Presentation at IEEE Medical Imaging Conference in Dresden, Germany.
IEEE Nuclear Science Symposium Conference Record, October 2008, DOI: 10.1109/NSSMIC.2008.4774494, ISBN: 978-1-4244-2715-4

We propose a new intrinsic motion-compensation algorithm for PET called "Blind Motion Compensated Reconstruction" (BMCR). BMCR is able to deal with frames of extremely low statistics in the case of smooth motion. This is achieved by combining image reconstruction and motion compensation into one mathematical framework which consists of a cost functional and an optimization method. The cost functional basically consists of a difference term which ensures consistency of the estimated parameters to the model and some regularization terms which render the problem mathematically well-posed. The optimization method aims at finding a pair of image and transformation/motion such that the cost functional is minimal. Up to now, for motion only translations are considered. Initial results are promising and show that the quality of images reconstructed by the BMCR algorithm for motion-contaminated data is (a) significantly superior to that of the Maximum-Likelihood Expectation-Maximization (ML-EM) algorithm for motion-contaminated data and (b) even comparable to an ML-EM reconstruction for motion-free data.

D.3. National Conferences

Joint Reconstruction of Image and Motion for PET Using Linear Diffusion Regularization
Moritz Blume, Magdalena Rafecas
Oral Presentation at CASEIB in Madrid, Spain.
Congreso Anual de la Sociedad Española de Ingeniería Biomédica, November 2010

We present a modification of our recently proposed joint reconstruction algorithm for PET. A problem related to the original presentation is the need of a regularization parameter which defines the trade-off between data term and regularization term. This regularization parameter is hard to find and usually there is no other way than trying different values and finally choosing the one which leads to the most satisfactory reconstruction. In our modified version, we eliminate the regularization parameter. Both quantitative and visual comparison show that the modified version of our joint reconstruction algorithm provides the same reconstruction quality and is therefore an acceptable substitute.

List of Figures

List of Tables

Bibliography

[1] M. Alkire, R. Haier, S. Barker, N. Shah, J. Wu, and Y. Kao. Cerebral metabolism during propofol anesthesia in humans studied with positron emission tomography. *Anesthesiology*, 82(2):393–403, Feburary 1995.

[2] M. T. Alkire, C. J. Pomfrett, R. J. Haier, M. V. Gianzero, C. M. Chan, B. P. Jacobsen, and J. H. Fallon. Functional brain imaging during anesthesia in humans: effects of halothane on global and regional cerebral glucose metabolism. *Anesthesiology*, 90(3):701–709, Mar 1999.

[3] M. T. M. Alkire, R. J. P. Haier, N. K. M. Shah, and C. T. M. Anderson. Positron emission tomography study of regional cerebral metabolism in humans during isoflurane anesthesia. *Anesthesiology*, 86(3):549–557, 1997.

[4] A. Andreyev, M. Defrise, and C. Vanhove. Pinhole SPECT reconstruction using blobs and resolution recovery. *IEEE Transactions on Nuclear Science*, 53(5):2719–2728, 2006.

[5] R. Ansorge. List mode 3D PET reconstruction using an exact system matrix and polar voxels. In *Nuclear Science Symposium Conference Record, 2007. NSS '07. IEEE*, volume 5, pages 3454 –3457, 262007-nov.3 2007.

[6] D. Arrowsmith and C. Place. *Dynamical Systems: Differential Equations, Maps, and Chaotic Behaviour (Chapman Hall/CRC Mathematics Series)*. Chapman and Hall/CRC, 1992.

[7] D. L. Bailey. *Quantification in 3D positron emission tomography. (1996)*. PhD thesis, University of Surrey, 1996.

[8] D. L. Bailey, D. W. Townsend, P. E. Valk, and M. N. Maisey. *Positron Emission Tomography: Basic Sciences*. Springer, 2005.

[9] H. H. Barrett and W. Swindell. *Radiological Imaging: The Theory of Image Formation, Detection, and Processing*. Academic Press, 1996.

[10] M. Bergström, J. Boëthius, L. Eriksson, T. Greitz, T. Ribbe, and L. Widén. Head fixation device for reproducible position alignment in transmission ct and positron emission tomography. *J Comput Assist Tomogr*, 5(1):136–141, Feb 1981.

[11] P. M. Bloomfield, T. J. Spinks, J. Reed, L. Schnorr, A. M. Westrip, L. Livieratos, R. Fulton, and T. Jones. The design and implementation of a motion correction scheme for neurological PET. *Physics in Medicine and Biology*, 48(8):959–978, 2003.

[12] M. Blume. 4D reconstruction of heart respiratory motion in PET. http://www.youtube.com/watch?v=OpTKALfjKx4.

[13] M. Blume, A. Keil, N. Navab, and M. Rafecas. Blind motion compensation for positron-emission-tomography. In *Proceedings of the SPIE*, volume 7258, pages 72580T–72580T–8, Feburary 2009.

[14] M. Blume, A. Keil, N. Navab, and M. Rafecas. Joint reconstruction of image and motion for PET: Displacement fields versus a b-spline motion model. In *IEEE Nuclear Science Symposium Conference Record*, 2010.

[15] M. Blume, A. Keil, M. Rafecas, and N. Navab. Simultaneous reconstruction of image and motion in gated positron-emission-tomography. In *IEEE Nuclear Science Symposium Conference Record*, pages 5485 – 5487, October 2009.

[16] M. Blume, A. Martinez-Möller, A. Keil, N. Navab, and M. Rafecas. Joint reconstruction of image and motion in gated positron-emission-tomography. In *Proceedings of the 10th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine (Fully3D)*, pages 118–121, September 2009.

[17] M. Blume, A. Martinez-Möller, A. Keil, N. Navab, and M. Rafecas. Joint reconstruction of image and motion in gated positron emission tomography. *IEEE Transactions on Medical Imaging*, 29(11):1892–1906, November 2010.

[18] M. Blume and M. Rafecas. Joint reconstruction of image and motion for PET using linear diffusion regularization. In *Congreso Anual de la Sociedad Española de Ingeniería Biomédica*, November 2010.

[19] M. Blume, M. Rafecas, S. Ziegler, and N. Navab. Combined motion compensation and reconstruction for PET. In *IEEE Nuclear Science Symposium Conference Record*, October 2008.

[20] N. Bose, S. Lertrattanapanich, and J. Koo. Advances in superresolution using L-curve. In *IEEE International Symposium on Circuits and Systems*, volume 2, pages 433–436, May 2001.

[21] L. Boucher, S. Rodrigue, R. Lecomte, and F. Bénard. Respiratory gating for 3-dimensional PET of the thorax: feasibility and initial results. *J Nucl Med*, 45(2):214–219, Feb 2004.

[22] D. Brasse, P. E. Kinahan, C. Lartizien, C. Comtat, M. Casey, and C. Michel. Fully 3D whole-body PET: Impact on data and image quality. *Journal of Nuclear Medicine*, 46:859 – 867, 2005.

[23] M. Bulmer. *Principles of Statistics*. Dover Publications, 1979.

[24] R. Bundschuh, A. Martínez-Möller, M. Essler, M.-J. Martínez, S. Nekolla, S. Ziegler, and M. Schwaiger. Post acquisition detection of tumor motion using list-mode data in positron emission tomography - a feasibility study. *Journal of Nuclear Medicine*, 48:758–763, 2007.

[25] F. Büther, M. Dawood, L. Stegger, F. Wubbeling, M. Schafers, O. Schober, and K. P. Schafers. List mode-driven cardiac and respiratory gating in PET. *J Nucl Med*, 50(5):674–681, 2009.

[26] F. Büther, I. Ernst, M. Dawood, P. Kraxner, M. Schäfers, O. Schober, and K. Schäfers. Detection of respiratory tumour motion using intrinsic list mode-driven gating in positron emission tomography. *European Journal of Nuclear Medicine and Molecular Imaging*, 37:2315–2327, 2010. 10.1007/s00259-010-1533-y.

[27] I. Buvat. A non-parametric bootstrap approach for analysing the statistical properties of SPECT and PET images. *Physics in Medicine and Biology*, 47(10):1761, 2002.

[28] P. Bühler, U. Just, E. Will, J. Kotzerke, and J. van den Hoff. An accurate method for correction of head movement in PET. *IEEE T. Med. Imaging.*, 23(9):1176–1185, Sept. 2004.

[29] J. Cabello, J. F. Oliver, and M. Rafecas. Using spherical basis functions on a polar grid for iterative image reconstruction in small animal PET. In *Proceedings of the SPIE*, volume 7961, page 796144, Feburary 2011.

[30] J. Cabello, J. F. Oliver, I. Torres-Espallardo, and M. Rafecas. Polar voxelization schemes combined with a monte-carlo based system matrix for image reconstruction in high resolution PET. In *IEEE Nuclear Science Symposium Conference Record*, 2010.

[31] Z. Cao. *Simultaneous Reconstruction and 3D Motion Estimation for Gated Myocardial Emission Tomography*. PhD thesis, University of Florida, 2003.

[32] Z. Cao, D. Gilland, B. Mair, and R. Jaszczak. Three-dimensional motion estimation with image reconstruction for gated cardiac ECT. *IEEE T. Nucl. Sci.*, 50(3):384–388, June 2003.

[33] Z. Cao, D. R. Gilland, B. A. Mair, and R. J. Jaszczak. 3d motion estimation with image reconstruction for gated cardiac ECT. In *Nuclear Science Symposium Conference Record, 2002 IEEE*, volume 3, pages 1869–1873, Nov. 2002.

[34] M. E. Casey and E. J. Hoffman. Quantitation in positron emission computed tomography: 7. a technique to reduce noise in accidental coincidence measurements and coincidence efficiency calibration. *Journal of Computer Assisted Tomography*, 10(5):845–850, 1986.

[35] C. Catana, T. Benner, A. van der Kouwe, L. Byars, M. Hamm, D. B. Chonde, C. J. Michel, G. El Fakhri, M. Schmand, and A. G. Sorensen. MRI-assisted PET motion correction for neurologic studies in an integrated MR-PET scanner. *J Nucl Med*, 52(1):154–161, 2011.

[36] C. M. Chen, S.-Y. Lee, and Z. H. Cho. Parallelization of the EM algorithm for 3-D PET image reconstruction. *IEEE Transactions on Medical Imaging*, 10(4):513–522, 1991.

[37] M. Chen, W. Lu, Q. Chen, K. J. Ruchala, and G. H. Olivera. A simple fixed-point approach to invert a deformation field. *Medical Physics*, 35(1):81–88, 2008.

[38] S. Chen and B. M. W. Tsui. Accuracy analysis of image-based respiratory motion estimation and compensation in respiratory-gated PET reconstruction. In *Proc. IEEE Nuclear Science Symp. Conf. Record NSS '08*, pages 4292–4295, 2008.

[39] P.-C. Chiao, W. L. Rogers, J. A. Fessler, N. H. Clinthorne, and A. O. Hero. Model-based estimation with boundary side information or boundary regularization [cardiac emission ct]. *IEEE Transactions on Medical Imaging*, 13(2):227–234, 1994.

[40] G. E. Christensen, S. C. Joshi, and M. I. Miller. Volumetric transformation of brain anatomy. *IEEE T Med Imaging*, 16(6):864–877, 1997.

[41] S. Y. Chun and J. A. Fessler. Regularized methods for topology-preserving smooth nonrigid image registration using b-spline basis. In *Proc. IEEE Intl. Symp. Biomed. Imag.*, pages 1099–1102, May 14–17, 2008.

[42] S. Y. Chun and J. A. Fessler. Joint image reconstruction and nonrigid motion estimation with a simple penalty that encourages local invertibility. In *Proceedings of the SPIE*, volume 7258, 2009.

[43] S. Y. Chun and J. A. Fessler. Spatial resolution and noise properties of regularized motion-compensated image reconstruction. In *Proc. IEEE Intl. Symp. Biomed. Imag.*, pages 863–866, Piscataway, NJ, USA, 2009. IEEE Press.

[44] S. Y. Chun, T. Reese, B. Guerin, X. Zhu, S. Cho, J. Ouyang, C. Catana, and G. El Fakhri. 4D tagged MR-based motion correction in simultaneous PET-MR. *J NUCL MED MEETING ABSTRACTS*, 51(2):80, 2010.

[45] S. Y. Chun, C. Schretter, and J. A. Fessler. Sufficient condition for local invertibility of spatio-temporal 4d b-spline deformations. In *Proc. IEEE Intl. Symp. Biomed. Imag.*, pages 1221–1224, 2010.

[46] C. Comtat, P. Kinahan, M. Defrise, C. Michel, C. Lartizien, and D. Townsend. Simulating whole-body PET scanning with rapid analytical methods. In *Nuclear Science Symposium, 1999. Conference Record. 1999 IEEE*, volume 3, pages 1260 –1264 vol.3, 1999.

[47] N. Costes, A. Dagher, K. Larcher, A. C. Evans, D. L. Collins, and A. Reilhac. Motion correction of multi-frame pet data in neuroreceptor mapping: Simulation based validation. *NeuroImage*, 47(4):1496 – 1505, 2009.

[48] P. Danias, M. Stuber, R. Botnar, K. Kissinger, R. Edelman, and W. Manning. Relationship between motion of coronary arteries and diaphragm during free breathing: lessons from real-time MR imaging. *Am. J. Roentgenol.*, 172(4):1061–1065, 1999.

[49] M. E. Daube-Witherspoon, Y. C. Yan, M. V. Green, and R. E. Carson. Correction for motion distortion in PET by dynamic monitoring of patient position. *The Journal of Nuclear Medicine*, 31:816, 1990.

[50] M. Dawood, N. Lang, X. Jiang, and K. P. Schäfers. Lung motion correction on respiratory gated 3-D PET/CT images. *IEEE T. Med. Imaging.*, 25(4):476–485, Apr. 2006.

[51] B. De Man and S. Basu. Distance-driven projection and backprojection. In *Proc. IEEE Nuclear Science Symp. Conf. Record*, volume 3, pages 1477–1480, 2002.

[52] M. Defrise, D. W. Townsend, D. Bailey, A. M. C. Geissbuhler, and T. Jones. A normalization technique for 3D PET data. *Physics in Medicine and Biology*, 36(7):939, 1991.

[53] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[54] F. H. Fahey. Data acquisition in PET imaging. *J Nucl Med Technol*, 30(2):39–49, 2002.

[55] H. Fayad, T. Pan, C. Roux, J. F., and D. Visvikis. A generic respiratory motion model for motion correction in PET/CT. In *Proc. IEEE Nuclear Science Symp. Conf. Record (NSS/MIC)*, 2010.

[56] H. Fayad, T. Pan, C. Roux, C. C. Le Rest, O. Pradier, J. F. Clement, and D. Visvikis. A patient specific respiratory model based on 4D CT data and a time of flight camera (TOF). In *Proc. IEEE Nuclear Science Symp. Conf. Record (NSS/MIC)*, pages 2594–2598, 2009.

[57] L. Fazendeiro, N. C. Ferreira, A. Blanco, and P. Fonte. EM reconstruction algorithm with resolution modeling applied to an RPC-PET prototype. In *Proc. IEEE Nuclear Science Symp. Conf. Record*, volume 4, pages 2511–2515, 2004.

[58] T. Felder, M. Blume, J. Oliver, and M. Rafecas. ML-EM implementation on a GPU: Avoiding simultaneous read-modify-write processes. In *Proceedings of the 2nd Workshop on High Performance Image Reconstruction (HPIR)*, pages 65–68, September 2009.

[59] J. Fessler. Optimization transfer approach to joint registration / reconstruction for motion-compensated image reconstruction. In *Biomedical Imaging: From Nano to Macro, 2010 IEEE International Symposium on*, pages 596 –599, Apr. 2010.

[60] B. J. Fueger, J. Czernin, I. Hildebrandt, C. Tran, B. S. Halpern, D. Stout, M. E. Phelps, and W. A. Weber. Impact of animal handling on the results of 18F-FDG PET studies in mice. *J Nucl Med*, 47(6):999–1006, 2006.

[61] R. Fulton, I. Nickel, L. Tellmann, S. Meikle, U. Pietrzyk, and H. Herzog. Event-by-event motion compensation in 3d PET. *2003 IEEE Nuclear Science Symposium Conference Record*, 5:3286–3289, Oct. 2003.

[62] R. Fulton, L. Tellmann, U. Pietrzyk, O. Winz, I. Stangier, I. Nickel, A. Schmid, S. Meikle, and H. Herzog. Accuracy of motion correction methods for PET brain imaging. In *Nuclear Science Symposium Conference Record, 2004 IEEE*, volume 7, pages 4226 –4230, 16-22 2004.

[63] R. R. Fulton, S. R. Meikle, S. Eberl, J. Pfeiffer, and C. J. Constable. Correction for head movements in positron emission tomography usingan optical motion-tracking system. *IEEE T. Nucl. Sci.*, 49:116–123, Feb. 2002.

[64] R. R. Fulton, L. Tellmann, U. Pietrzyk, and H. Herzog. Compensation for head movement in 3D PET. In *Nuclear Science Symposium Conference Record, 2001 IEEE*, volume 4, pages 2013–2017, San Diego, CA, USA, 2001.

[65] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, 1984.

[66] F. Gigengack, L. Ruthotto, M. Burger, C. H. Wolters, X. Jiang, and K. Schaefers. Motion correction of cardiac PET using mass-preserving registration. In *Proc. IEEE Nuclear Science Symp. Conf. Record (NSS/MIC)*, 2010.

[67] D. Gilland, B. Mair, J. Bowsher, and R. Jaszczak. Simultaneous reconstruction and motion estimation for gated cardiac ect. *IEEE T. Nucl. Sci.*, 49(5):2344–2349, Oct 2002.

[68] D. R. Gilland, B. A. Mair, and J. G. Parker. Improved 3d motion estimation for cardiac emission tomography. In *Proceedings of the 9th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine (Fully3D)*, pages 193–196, 2007.

[69] D. R. Gilland, B. A. Mair, and J. G. Parker. Motion estimation for cardiac emission tomography by optical flow methods. *Physics in Medicine and Biology*, 53(11):2991, 2008.

[70] G. Gindi, M. Lee, A. Rangarajan, and I. Zubal. Bayesian reconstruction of functional images using registered anatomical images as priors. In A. Colchester and D. Hawkes, editors, *Information Processing in Medical Imaging*, volume 511 of *Lecture Notes in Computer Science*, pages 121–131. Springer Berlin / Heidelberg, 1991. 10.1007/BFb0033747.

[71] B. Glocker and N. Komodakis. drop, October 2009. `http://www.mrf-registration.net/`.

[72] B. Glocker, N. Komodakisa, G. Tziritasc, N. Navab, and N. Paragios. Dense image registration through MRFs and efficient linear programming. *Medical Image Analysis*, 12:731–741, 2008.

[73] S. Goldstein, M. Daube-Witherspoon, and M. Green. A device for continuous measurement of head position during PET brain imaging. In *Journal of Nuclear Medicine*, volume 35, 1994.

[74] G. H. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[75] A. A. Goshtasby. *2-D and 3-D Image Registration: for Medical, Remote Sensing, and Industrial Applications*. Wiley-Interscience, 2005.

[76] E. Gravier, Y. Yang, M. A. King, and M. Jin. Fully 4d motion-compensated reconstruction of cardiac SPECT images. *Physics in Medicine and Biology*, 51(18):4603–4619, 2006.

[77] E. J. Gravier and Y. Yang. Motion-compensated reconstruction of tomographic image sequences. *IEEE T. Nucl. Sci.*, 52:51–56, Feb. 2005.

[78] M. V. Green, J. Seidel, S. D. Stein, T. E. Tedder, K. M. Kempner, C. Kertzman, and T. A. Zeffiro. Head movement in normal subjects during simulated PET brain imaging with and without head restraint. *J Nucl Med*, 35(9):1538–1546, 1994.

[79] E. Haber and J. Modersitzki. Numerical methods for volume preserving image registration. *Inverse Problems*, 20(5):1621, 2004.

[80] J. V. Hajnal, D. L. G. Hill, and D. J. Hawkes, editors. *Medical Image Registration (Biomedical Engineering)*. CRC Press, 2001.

[81] P. C. Hansen. Analysis of discrete ill-posed problems by means of the L-curve. *SIAM Review*, 34(4):561–580, 1992.

[82] W. Heinke and C. Schwarzbauer. In vivo imaging of anaesthetic action in humans: approaches with positron emission tomography (PET) and functional magnetic resonance imaging (fMRI). *British Journal of Anaesthesia*, 89(1):112–122, 2002.

[83] G. T. Herman. *Fundamentals of Computerized Tomography: Image Reconstruction from Projections (Advances in Pattern Recognition)*. Springer, 2009.

[84] H. Herzog, L. Tellman, R. Fulton, and U. Pietrzyk. Motion correction in PET brain studies. In *Multidimensional Systems, 2005. NDS 2005. The Fourth International Workshop on*, pages 178 – 181, 10-13 2005.

[85] H. Herzog, L. Tellmann, R. Fulton, I. Stangier, E. R. Kops, K. Bente, C. Boy, R. Hurlemann, and U. Pietrzyk. Motion artifact reduction on parametric PET images of neuroreceptor binding. *Journal of Nuclear Medicine*, 46(6):1059–1065, 2005.

[86] E. Hoffman, P. Cutler, W. Digby, and J. Mazziotta. 3-d phantom to simulate cerebral blood flow and metabolic images for pet. *IEEE Transactions on Nuclear Science*, 37(2):616 –620, Apr. 1990.

[87] E. J. Hoffman, T. M. Guerrero, G. Germano, W. M. Digby, and M. Dahlbom. PET system calibrations and corrections for quantitative and spatially accurate images. *IEEE Transactions on Nuclear Science*, 36(1):1108–1112, 1989.

[88] E. J. Hoffman, S.-C. Huang, M. E. Phelps, and D. E. Kuhl. Quantitation in positron emission computed tomography: 4. effect of accidental coincidences. *Journal of Computer Assisted Tomography*, 5(3):391–400, 1981.

[89] B. K. P. Horn and B. G. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[90] D. Hu, C. Hayden, M. Casey, and Z. Burbar. Stereo computer vision system for measuring movement of patient's head in PET scanning. In *Nuclear Science Symposium Conference Record, 2004 IEEE*, volume 5, pages 2864 –2867, 16-22 2004.

[91] H. M. Hudson and R. S. Larkin. Accelerated image reconstruction using ordered subsets of projection data. *IEEE T. Med. Imaging.*, 13(4):601–609, Dec. 1994.

[92] R. H. Huesman, G. T. Gullberg, W. L. Greenberg, and T. F. Budinger. Users manual: Donner algorithms for reconstruction tomography. Technical report, University of California, Lawrence Berkeley Laboratory, 1977.

[93] R. H. Huesman, G. J. Klein, W. W. Moses, J. Qi, B. W. Reutter, and P. R. G. Virador. List-mode maximum-likelihood reconstruction applied to positron emission mammography (PEM) with irregular sampling. *IEEE Transactions on Medical Imaging*, 19(5):532–537, 2000.

[94] L. Ibañez and W. Schröder. *The ITK Software Guide 2.4*. Kitware, Inc., 2005.

[95] F. Jacobs, E. Sundermann, B. D. Sutter, M. Christiaens, and I. Lemahieu. A fast algorithm to calculate the exact radiological path through a pixel or voxel space. *CIT. Journal of computing and information technology (CIT. J. comput. inf. technol.)*, 6(1):89–94, 1998.

[96] M. Jacobson and J. Fessier. Joint estimation of respiratory motion and activity in 4D PET using CT side information. In *Biomedical Imaging: Nano to Macro, 2006. 3rd IEEE International Symposium on*, pages 275 –278, Apr. 2006.

[97] M. Jacobson and J. Fessler. Joint estimation of image and deformation parameters in motion-corrected PET. In *Nuclear Science Symposium Conference Record, 2003 IEEE*, volume 5, pages 3290–3294, Oct. 2003.

[98] P. M. Joseph. An improved algorithm for reprojecting rays through pixel images. *IEEE T. Med. Imaging.*, 1(3):192–196, 1982.

[99] M. S. Judenhofer, H. F. Wehrl, D. F. Newport, C. Catana, S. B. Siegel, M. Becker, A. Thielscher, M. Kneilling, M. P. Lichy, M. Eichner, K. Klingel, G. Reischl, S. Widmaier, M. Rocken, R. E. Nutt, H.-J. Machulla, K. Uludag, S. R. Cherry, C. D. Claussen, and B. J. Pichler. Simultaneous pet-mri: a new approach for functional and morphological imaging. *Nat Med*, 14(4):459–465, Apr. 2008.

[100] G. J. Klein. Forward deformation of PET volumes using material constraints. In *Biomedical Image Analysis, 1998. Proceedings. Workshop on*, pages 64–71, Santa Barbara, CA, USA, June 1998.

[101] G. J. Klein. Four-dimensional processing of deformable cardiac PET data. In *Mathematical Methods in Biomedical Image Analysis, 2000. Proceedings. IEEE Workshop on*, pages 127–134, Hilton Head Island, SC, USA, 2000.

[102] G. J. Klein and R. H. Huesman. A 3D optical flow approach to addition of deformable PET volumes. In *Nonrigid and Articulated Motion Workshop, 1997. Proceedings., IEEE*, pages 136–143, San Juan, Puerto Rico, June 1997.

[103] G. J. Klein, B. W. Reutter, and R. H. Huesman. Non-rigid summing of gated PET via optical flow. In *Nuclear Science Symposium, 1996. Conference Record., 1996 IEEE*, volume 2, pages 1339–1342, Anaheim, CA, USA, Nov. 1996.

[104] G. J. Klein, B. W. Reutter, and R. H. Huesman. 4d affine registration models for respiratory-gated PET. In *Nuclear Science Symposium Conference Record, 2000 IEEE*, volume 2, pages 41–15, Lyon, France, 2000.

[105] R. Kohavi. A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 1137–1143. Morgan Kaufmann, 1995.

[106] N. Komodakis, G. Tziritas, and N. Paragios. Fast, approximately optimal solutions for single and dynamic MRFs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[107] K. Koshino, N. Kawachi, T. Hayashi, H. Watabe, S. Hasegawa, J. Hatazawa, and H. Iida. Effects of motion correction on quantification of myocardial blood flow with 15O-H2O PET. *International Congress Series*, 1265:106 – 110, 2004. Quantitation in Biomedical Imaging with PET and MRI. Proceedings of the International Workshop on Quantitation in Biomedical Imaging with PET and MRI.

[108] A. Z. Kyme, V. W. Zhou, S. R. Meikle, and R. R. Fulton. Real-time 3D motion tracking for small animal brain PET. *Physics in Medicine and Biology*, 53(10):2651, 2008.

[109] D. S. Lalush and B. M. W. Tsui. Simulation evaluation of gibbs prior distributions for use in maximum a posteriori spect reconstructions. In *Proc. IEEE Nuclear Science Symp. Conf. record : Including Sessions Nuclear Power Systems and Medical Imaging Conf*, pages 1600–1609, 1990.

[110] D. S. Lalush and B. M. W. Tsui. Simulation evaluation of gibbs prior distributions for use in maximum a posteriori spect reconstructions. *IEEE Transactions on Medical Imaging*, 11(2):267–275, 1992.

[111] D. S. Lalush and B. M. W. Tsui. A generalized gibbs prior for maximum a posteriori reconstruction in spect. *Physics in Medicine and Biology*, 38(6):729, 1993.

[112] F. Lamare, M. J. L. Carbayo, T. Cresson, G. Kontaxakis, A. Santos, C. Cheze, L. Rest, A. J. Reader, and D. Visvikis. List-mode-based reconstruction for respiratory motion correction in PET using non-rigid body transformations. *Physics in Medicine and Biology*, 52(17):5187–5204, 2007.

[113] F. Lamare, M. J. L. Carbayo, G. Kontaxakis, A. Santos, A. Turzo, Y. Bizais, C. Cheze Le Rest, and D. Visvikis. Incorporation of elastic transformations in list-mode based

reconstruction for respiratory motion correction in PET. *2005 IEEE Nuclear Science Symposium Conference Record*, 3, Oct. 2005.

[114] F. Lamare, M. J. L. Carbayo, A. J. Reader, O. R. Mawlawi, G. Kontaxakis, A. Santos, C. C.-L. Rest, and D. Visvikis. Respiratory motion correction in 4d PET/CT: Comparison of implementation methodologies for incorporation of elastic transformations in the reconstruction system matrix. *2006. IEEE Nuclear Science Symposium Conference Record*, 4:2365–2369, Oct. 2006.

[115] F. Lamare, T. Cresson, J. Savean, C. C. L. Rest, A. J. Reader, and D. Visvikis. Respiratory motion correction for PET oncology applications using affine transformation of list mode data. *Physics in Medicine and Biology*, 52(1):121–140, 2007.

[116] K. Lange and R. Carson. Em reconstruction algorithms for emission and transmission tomography. *Journal of Computer Assisted Tomography*, 8:306 – 316, 1984.

[117] J. Langner. *Event-Driven Motion Compensation in Positron Emission Tomography: Development of a Clinically Applicable Method*. PhD thesis, Technische Universität Dresden, 2008.

[118] R. Leahy and X. Yan. Incorporation of anatomical MR data for improved functional imaging with PET. In A. Colchester and D. Hawkes, editors, *Information Processing in Medical Imaging*, volume 511 of *Lecture Notes in Computer Science*, pages 105–120. Springer Berlin / Heidelberg, 1991. 10.1007/BFb0033746.

[119] R. M. Leahy and J. Qi. Statistical approaches in quantitative positron emission tomography. *Statistics and Computing*, 10:147–165, 2000. 10.1023/A:1008946426658.

[120] K.-H. Lee, B.-H. Ko, J.-Y. Paik, K.-H. Jung, Y. S. Choe, Y. Choi, and B.-T. Kim. Effects of anesthetic agents and fasting duration on 18F-FDG biodistribution and insulin levels in tumor-bearing mice. *J Nucl Med*, 46(9):1531–1536, 2005.

[121] S.-J. Lee, A. Rangarajan, and G. Gindi. A comparative study of the effects of using higher order mechanical priors in spect reconstruction. In *Proc. IEEE Conf Nuclear Science Symp. and Medical Imaging Conf. Record*, volume 4, pages 1696–1700, 1994.

[122] S.-J. Lee, A. Rangarajan, and G. Gindi. Bayesian image reconstruction in spect using higher order mechanical models as priors. *IEEE Transactions on Medical Imaging*, 14(4):669–680, 1995.

[123] E. Levitan and G. T. Herman. A maximum a posteriori probability expectation maximization algorithm for image reconstruction in emission tomography. *IEEE Transactions on Medical Imaging*, 6(3):185–192, 1987.

[124] R. M. Lewitt. Alternatives to voxels for image representation in iterative reconstruction algorithms. *Physics in Medicine and Biology*, 37(3):705, 1992.

[125] Z. Liang, R. Jaszczak, and K. Greer. On bayesian image reconstruction from projections: uniform and nonuniform a priori source information. *IEEE Transactions on Medical Imaging*, 8(3):227–235, 1989.

[126] N. Lossau. Fantastische Bilder aus dem Körperinnern. Newspaper "Die Welt", November 2010. http://www.welt.de/print/die_welt/wissen/article11071020/Fantastische-Bilder-aus-dem-Koerperinnern.html.

[127] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of Imaging Understanding Workshop*, pages 674–679, 1981.

[128] L. B. Lucy. An iterative technique for the rectification of observed distributions. *Astronomical Journal*, 79:745, 1974.

[129] B. Mair, D. Gilland, and J. Sun. Estimation of images and nonrigid deformations in gated emission ct. *IEEE T. Med. Imaging.*, 25(9):1130–1144, Sept. 2006.

[130] B. A. Mair, D. R. Gilland, and Z. Cao. Simultaneous motion estimation and image reconstruction from gated data. In *Biomedical Imaging, 2002. Proceedings. 2002 IEEE International Symposium on*, pages 661–664, 2002.

[131] B. D. Man. Method and apparatus for reduction of artifacts in computed tomography images. U. S. Patent No. 7023951, December 2003.

[132] B. D. Man and S. Basu. Distance-driven projection and backprojection in three dimensions. *Physics in Medicine and Biology*, 49(11):2463–2475, 2004.

[133] A. Martínez-Möller, R. Bundschuh, M. Riedel, N. Navab, S. Ziegler, M. Schwaiger, and S. Nekolla. Comparison of respiratory sensors and its compliance for respiratory gating in emission tomography. In *Journal of Nuclear Medicine*, volume 48, page 426, Wahington, USA, June 2007.

[134] A. Martínez-Möller, D. Zikic, R. Botnar, R. Bundschuh, W. Howe, S. Ziegler, N. Navab, M. Schwaiger, and S. Nekolla. Dual cardiac-respiratory gated PET: Implementation and results from a feasibility study. *European Journal of Nuclear Medicine and Molecular Imaging*, 2007.

[135] S. Matej and R. M. Lewitt. Efficient 3D grids for image reconstruction using spherically-symmetric volume elements. In *Proc. IEEE Conf Nuclear Science Symp. and Medical Imaging Conf. Record*, volume 3, pages 1177–1181, 1994.

[136] S. Matej and R. M. Lewitt. Efficient 3D grids for image reconstruction using spherically-symmetric volume elements. *IEEE Transactions on Nuclear Science*, 42(4):1361–1370, 1995.

[137] E. R. McVeigh and E. Atalar. Cardiac tagging with breath-hold cine mri. *Magn. Reson. Med.*, 28(2):318–327, 1992.

[138] M. Menke, M. S. Atkins, and K. R. Buckley. Compensation methods for head motion detected during PET imaging. *IEEE T. Nucl. Sci.*, 43:310–317, Feb. 1996.

[139] J. Modersitzki. *Numerical Methods for Image Registration (Numerical Mathematics and Scientific Computation)*. Oxford University Press, USA, 2004.

[140] G. E. Moore. Interview. Online, 2005.

[141] G. E. Moore. Cramming more components onto integrated circuits, reprinted from electronics, volume 38, number 8, april 19, 1965, pp.114 ff. *Solid-State Circuits Newsletter, IEEE*, 20(3):33 –35, Sept. 2006.

[142] C. Mora and M. Rafecas. Polar pixels for high resolution small animal PET. In *IEEE Nuclear Science Symposium Conference Record*, volume 5, pages 2812 –2817, 29 2006-nov. 1 2006.

[143] E. U. Mumcuoglu, R. M. Leahy, and S. R. Cherry. Bayesian reconstruction of PET images: methodology and performance analysis. *Physics in Medicine and Biology*, 41(9):1777, 1996.

[144] K. B. M.W. Stazyk, V. Sossi and T. Ruth. Normalization measurement in septa-less PET scanners. In *Journal of Nuclear Medicine*, volume 35, page 41, May 1994.

[145] N. Ojha, J. Griesmer, Z. Hu, L. Shao, D. Izquierdo, J. Machac, O. Ratib, H. Zaidi, V. Fuster, and Z. A. Fayad. PET performance of the GEMINI TF PET-MR: the world's first whole body PET-MRI scanner. In *IEEE Nuclear Science Symposium Conference Record*, pages 2013–2015, November 2010.

[146] J. M. Ollinger. Detector efficiency and Compton scatter in fully 3D PET. In *Proc. IEEE Conf Nuclear Science Symp. and Medical Imaging Conf. Record*, volume 3, pages 1280–1284, 1994.

[147] J. M. Ollinger. Detector efficiency and Compton scatter in fully 3D PET. *IEEE Transactions on Nuclear Science*, 42(4):1168–1173, 1995.

[148] J. M. Ollinger. Model-based scatter correction for fully 3D PET. *Physics in Medicine and Biology*, 41(1):153, 1996.

[149] J. M. Ollinger and J. A. Fessler. Positron-emission tomography. *IEEE Signal Processing Magazine*, 14(1):43–55, Jan. 1997.

[150] J. M. Ollinger, G. C. Johns, and M. T. Burney. Model based scatter correction in three dimensions [positron emission tomography]. In *Proc. Conf Nuclear Science Symp. and Medical Imaging Conf. Record of the 1992 IEEE*, pages 1249–1251, 1992.

[151] X. Ouyang, W. H. Wong, V. E. Johnson, X. Hu, and C.-T. Chen. Incorporation of correlated structural images in PET image reconstruction. *IEEE Transactions on Medical Imaging*, 13(4):627–640, 1994.

[152] J. Parker and D. Gilland. Wall motion estimation for gated cardiac emission tomography: Physical phantom evaluation. *IEEE T. Nucl. Sci.*, 55(1):531–536, Feb. 2008.

[153] L. Parra and H. H. Barrett. List-mode likelihood: EM algorithm and image quality estimation demonstrated on 2-D PET. *IEEE Transactions on Medical Imaging*, 17(2):228–235, 1998.

[154] T. M. Peters. Algorithms for fast back- and re-projection in computed tomography. *IEEE Transactions on Nuclear Science*, 28(4):3641–3647, 1981.

[155] Y. Picard and C. Thompson. Digitized video subject positioning and surveillance system for PET. In *Nuclear Science Symposium and Medical Imaging Conference, 1994., 1994 IEEE Conference Record*, volume 3, pages 1150 –1154 vol.3, 30 1994.

[156] Y. Picard and C. J. Thompson. Motion correction of PET images using multiple acquisition frames. *IEEE T. Med. Imaging.*, 16(2):137–144, Apr. 1997.

[157] M. N. Pilipuf, J. C. Goble, and N. F. Kassell. A noninvasive thermoplastic head immobilization system. technical note. *J Neurosurg*, 82(6):1082–1085, Jun 1995.

[158] J. Qi and R. H. Huesman. List mode reconstruction for PET with motion compensation: a simulation study. In *Biomedical Imaging, 2002. Proceedings. 2002 IEEE International Symposium on*, pages 413–416, 2002.

[159] J. Qi, R. M. Leahy, S. R. Cherry, A. Chatziioannou, and T. H. Farquhar. High-resolution 3D Bayesian image reconstruction using the microPET small-animal scanner. *Physics in Medicine and Biology*, 43(4):1001, 1998.

[160] J. Qi, R. M. Leahy, C. Hsu, T. H. Farquhar, and S. R. Cherry. Fully 3d bayesian image reconstruction for the ecat exact hr+. In *Proc. IEEE Nuclear Science Symp*, volume 2, pages 953–957, 1997.

[161] J. Qi, R. M. Leahy, C. Hsu, T. H. Farquhar, and S. R. Cherry. Fully 3d bayesian image reconstruction for the ecat exact hr+. *IEEE Transactions on Nuclear Science*, 45(3):1096–1103, 1998.

[162] F. Qiao, J. W. Clark, T. Pan, and O. Mawlawi. Expectation maximization reconstruction of PET image with non-rigid motion compensation. In *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*, pages 4453–4456, Sept. 2005.

[163] F. Qiao, T. Pan, J. W. C. Jr, and O. R. Mawlawi. A motion-incorporated reconstruction method for gated PET studies. *Physics in Medicine and Biology*, 51(15):3769–3783, 2006.

[164] M. Rafecas, B. Mosler, M. Dietz, M. Pogl, A. Stamatakis, D. McElroy, and S. Ziegler. Use of a Monte Carlo-based probability matrix for 3-D iterative reconstruction of MADPET-II data. *Nuclear Science, IEEE Transactions on*, 51(5):2597 – 2605, Oct. 2004.

[165] A. Rahmim, P. Bloomfield, S. Houle, M. Lenox, C. Michel, K. R. Buckley, T. J. Ruth, and V. Sossi. Motion compensation in histogram-mode and list-mode EM reconstructions: beyond the event-driven approach. *IEEE T. Nucl. Sci.*, 51:2588–2596, Oct. 2004.

[166] A. Rahmim, K. Dinelle, J.-C. Cheng, M. A. Shilov, W. P. Segars, S. C. Lidstone, S. Blinder, O. G. Rousset, H. Vajihollahi, B. Tsui, D. F. Wong, and V. Sossi. Accurate event-driven motion compensation in high-resolution PET incorporating scattered and random events. *IEEE T. Med. Imaging.*, 27(8):1018–1033, 2008.

[167] A. Rahmim, K. Dinelle, S. C. Lidstone, S. Blinder, J. C. Cheng, G. Topping, H. Vajihollahi, D. F. Wong, and V. Sossi. Impact of accurate motion-corrected statistical reconstruction on dynamic PET kinetic parameter estimation. In *Nuclear Science Symposium Conference Record, 2007. NSS '07. IEEE*, volume 4, pages 2697–2704, Oct./Nov. 2007.

[168] A. Rahmim, J. Tang, M. R. Ay, and F. M. Bengel. 4d respiratory motion-corrected rb-82 myocardial perfusion PET image reconstruction. In *IEEE Nuclear Science Symposium Conference Record*, 2010.

[169] A. Rahmim, J. Tang, M. A. Lodge, S. Lashkari, M. R. Ay, R. Lautamäki, B. M. W. Tsui, and F. M. Bengel. Analytic system matrix resolution modeling in PET: an application to rb-82 cardiac imaging. *Physics in Medicine and Biology*, 53(21):5947, 2008.

[170] R. R. Raylman, B. E. Hammer, and N. L. Christensen. Combined MRI-PET scanner: a Monte Carlo evaluation of the improvements in PET resolution due to the effects of a static homogeneous magnetic field. *IEEE Transactions on Nuclear Science*, 43(4):2406–2412, 1996.

[171] N. Rehfeld and M. Alber. A parallelizable compression scheme for Monte Carlo scatter system matrices in PET image reconstruction. *Physics in Medicine and Biology*, 52(12):3421, 2007.

[172] M. Reyes, G. Malandain, P. M. Koulibaly, M. A. González-Ballester, and J. Darcourt. Model-based respiratory motion compensation for emission tomography image reconstruction. *Physics in Medicine and Biology*, 52(12):3579–3600, 2007.

[173] W. H. Richardson. Bayesian-based iterative method of image restoration. *Journal of the Optical Society of America A*, 62:55 – 59, 1972.

[174] T. Rohlfing, J. Maurer, C. R., D. A. Bluemke, and M. A. Jacobs. Volume-preserving nonrigid registration of MR breast images using free-form deformation with an incompressibility constraint. *IEEE T Med Imaging*, 22(6):730–741, 2003.

[175] W. Rudin. *Real and Complex Analysis (International Series in Pure and Applied Mathematics)*. McGraw-Hill Science/Engineering/Math, 1986.

[176] D. Rueckert, L. I. Sonoda, C. Hayes, D. L. G. Hill, M. O. Leach, and D. J. Hawkes. Nonrigid registration using free-form deformations: Application to breast MR images. *IEEE T. Med. Imaging.*, 18(8):712–721, 1999.

[177] Y. Saad. *Iterative Methods for Sparse Linear Systems, Second Edition*. Society for Industrial and Applied Mathematics, 2003.

[178] J. J. Scheins, F. Boschen, and H. Herzog. Analytical calculation of volumes-of-intersection for iterative, fully 3-D PET reconstruction. *IEEE T. Med. Imaging.*, 25(10):1363–1369, Oct. 2006.

[179] J. J. Scheins, L. Tellmann, C. Weirich, E. R. Kops, and H. Herzog. Ultra fast 3-D PET image reconstruction using highly compressed, memory-resident system matrices with optimised SIMD access patterns. In *IEEE Nuclear Science Symposium Conference Record*, 2010.

[180] H.-P. Schlemmer, B. Pichler, K. Wienhard, M. Schmand, C. Nahmias, D. Townsend, W.-D. Heiss, and C. Claussen. Simultaneous MR/PET for brain imaging: First patient scans. *J NUCL MED MEETING ABSTRACTS*, 48(2):45P–a, 2007.

[181] H.-P. W. Schlemmer, B. J. Pichler, M. Schmand, Z. Burbar, C. Michel, R. Ladebeck, K. Jattke, D. Townsend, C. Nahmias, P. K. Jacob, W.-D. Heiss, and C. D. Claussen. Simultaneous MR/PET imaging of the human brain: Feasibility study. *Radiology*, 248(3):1028–1035, September 2008.

[182] H. Schumacher and B. Fischer. A new approach for motion correction in SPECT imaging. In *Bildverarbeitung für die Medizin*, Informatik aktuell. Springer Berlin Heidelberg, 2007.

[183] H. Schumacher, J. Modersitzki, and B. Fischer. Combined reconstruction and motion correction in SPECT imaging. *IEEE T. Nucl. Sci.*, 56:73–80, 2009.

[184] W. P. Segars, M. Mahesh, T. J. Beck, E. C. Frey, and B. M. W. Tsui. Realistic CT simulation using the 4D XCAT phantom. *Medical Physics*, 35(8):3800–3808, 2008.

[185] Y. Shao, S. R. Cherry, K. Farahani, K. Meadors, S. Siegel, R. W. Silverman, and P. K. Marsden. Simultaneous PET and MR imaging. *Physics in Medicine and Biology*, 42(10):1965, 1997.

[186] G. Shechter, C. Ozturk, J. Resar, and E. McVeigh. Respiratory motion of the heart from free breathing coronary angiograms. *IEEE T. Med. Imaging.*, 23(8):1046–1056, Aug. 2004.

[187] L. A. Shepp and Y. Vardi. Maximum likelihood reconstruction for emission tomography. *Medical Image Processing, IEEE Transactions on*, 1(2):113–122, 1982.

[188] R. L. Siddon. Fast calculation of the exact radiological path for a three-dimensional ct array. *Med Phys*, 12(2):252–255, 1985.

[189] L. Tellmann, R. R. Fulton, K. Bente, I. Stangier, O. Winz, U. Just, H. Herzog, and U. K. Pietrzyk. Motion correction of head movements in PET: realisation for routine usage. *2003 IEEE Nuclear Science Symposium Conference Record*, 5:3105–3107, Oct. 2003.

[190] A. Terstegge, S. Weber, H. Herzog, H. Muller-Gartner, and H. Halling. High resolution and better quantification by tube of response modelling in 3D PET reconstruction. In *Nuclear Science Symposium, 1996. Conference Record., 1996 IEEE*, volume 3, pages 1603 –1607 vol.3, Nov. 1996.

[191] S. Thesen, O. Heid, E. Mueller, and L. R. Schad. Prospective acquisition correction for head motion with image-based tracking for real-time fmri. *Magn Reson Med*, 44(3):457–465, Sep 2000.

[192] K. Thielemans, E. Asma, and R. M. Manjeshwar. Mass-preserving image registration using free-form deformation fields. In *Proc. IEEE Nuclear Science Symp. Conf. Record (NSS/MIC)*, pages 2490–2495, 2009.

[193] K. Thielemans, S. Mustafovic, and L. Schnorr. Image reconstruction of motion corrected sinograms. *2003 IEEE Nuclear Science Symposium Conference Record*, 4:2401–2406, Oct. 2003.

[194] A. J. W. van der Kouwe, T. Benner, and A. M. Dale. Real-time rigid body motion correction and shimming using cloverleaf navigators. *Magn Reson Med*, 56(5):1019–1032, Nov 2006.

[195] Vardi, Y., Shepp, L. A., and Kaufman, L. A statistical model for positron emission tomography. *J. Am. Stat. Assoc.*, 80(389):8–20, mar 1985.

[196] Y. Wang, E. Chiu, J. Rosenberg, and S. S. Gambhir. Standardized uptake value atlas: Characterization of physiological 2-deoxy-2-[.^{18}f]fluoro-d-glucose uptake in normal tissues. *Molecular Imaging and Biology*, 9(2):83–90, 3 2007.

[197] Y. Wang, E. Vidan, and G. W. Bergman. Cardiac Motion of Coronary Arteries: Variability in the Rest Period and Implications for Coronary MR Angiography. *Radiology*, 213(3):751–758, 1999.

[198] C. Watson, D. Newport, , and M. Casey. A single scatter simulation technique for scatter correction in 3D PET. In *International Meeting on Fully Three Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, pages 255–268, 1995.

[199] J. Weickert. *Anisotropic Diffusion in Image Processing*. B. G. Teubner Stuttgart, 1998.

[200] M. N. Wernick and J. N. Aarsvold. *Emission Tomography: The Fundamentals of PET and SPECT*. Academic Press, October 2004.

[201] X. Wu. An efficient antialiasing technique. In *SIGGRAPH '91: Proceedings of the 18th annual conference on Computer graphics and interactive techniques*, pages 143–152, New York, NY, USA, 1991. ACM.

[202] M. Yavuz and J. A. Fessler. Objective functions for tomographic reconstruction from randoms-precorrected PET scans. In *Proc. IEEE Nuclear Science Symp. Conf. Record*, volume 2, pages 1067–1071, 1996.

[203] M. Yavuz and J. A. Fessler. Statistical image reconstruction methods for randoms-precorrected PET scans. *Medical Image Analysis*, 2(4):369 – 378, 1998.

[204] T. S. Yoo, M. J. Ackerman, W. E. Lorensen, W. Schroeder, V. Chalana, S. Aylward, D. Metaxas, and R. Whitaker. Engineering and algorithm design for an image processing api: A technical report on itk - the insight toolkit, 2002.

[205] G. L. Zeng and G. T. Gullberg. Unmatched projector/backprojector pairs in an iterative reconstruction algorithm. *IEEE Transactions on Medical Imaging*, 19(5):548–555, 2000.

[206] L. Zhang, S. Staelens, R. Van Holen, J. De Beenhouwer, J. Verhaeghe, I. Kawrakow, and S. Vandenberghe. Fast and memory-efficient Monte Carlo-based image reconstruction for whole-body PET. *Medical Physics*, 37(7):3667–3676, 2010.

[207] W. Zhuang, S. S. Gopal, and T. J. Hebert. Numerical evaluation of methods for computing tomographic projections. *IEEE Transactions on Nuclear Science*, 41(4):1660–1665, 1994.