Dissertation

submitted to the

Combined Faculty for the Natural Sciences and Mathematics

of

Heidelberg University, Germany

for the degree of

Doctor of Natural Sciences.

Put forward by

Diplom-Inf. Jens Muders

born in

Boppard, Germany

Oral examination: _____

Title

Geometrical Calibration and Filter Optimization for Cone-Beam Computed Tomography

Advisors:
Prof. Dr. Jürgen Hesser
Prof. Dr. Bernd Jähne

Geometrical Calibration and Filter Optimization for Cone-Beam Computed Tomography

Abstract

This thesis will discuss the requirements of a software library for tomography and will derive a framework which can be used to realize various applications in cone-beam computed tomography (CBCT). The presented framework is self-contained and is realized using the MATLAB environment in combination with native low-level technologies (C/C++ and CUDA) to improve its computational performance, while providing accessibility and extendability through to use of a scripting language environment. On top of this framework, the realization of Katsevich's algorithm on multi-core hardware will be explained and the resulting implementation will be compared to the Feldkamp, Davis and Kress (FDK) algorithm. It will also be shown that this helical reconstruction method has the potential to reduce the measurement uncertainty. However, misalignment artifacts appear more severe in the helical reconstructions from real data than in the circular ones. Especially for helical CBCT (H-CBCT), this fact suggests that a precise calibration of the computed tomography (CT) system is inevitable. As a consequence, a self-calibration method will be designed that is able to estimate the misalignment parameters from the cone-beam projection data without the need of any additional measurements. The presented method employs a multi-resolution 2D-3D registration technique and a novel volume update scheme in combination with a stochastic reprojection strategy to achieve a reasonable runtime performance. The presented results will show that this method reaches sub-voxel accuracy and can compete with current state-of-the-art online- and offline-calibration approaches. Additionally, for the construction of filters in the area of limited-angle tomography a general scheme which uses the Approximate Inverse (AI) to compute an optimized set of 2D angle-dependent projection filters will be derived. Optimal sets of filters are then precomputed for two angular range setups and will be reused to perform various evaluations on multiple datasets with a filtered backprojection (FBP)-type method. This approach will be compared to the standard FDK algorithm and to the simultaneous iterative reconstruction technique (SIRT). The results of the study show that the introduced filter optimization produces results comparable to those of SIRT with respect to the reduction of reconstruction artifacts, whereby its runtime is comparable to that of the FDK algorithm.

Geometrical Calibration and Filter Optimization for Cone-Beam Computed Tomography

Zusammenfassung

Diese Arbeit diskutiert die Anforderungen an eine Software-Bibliothek für die Computertomographie (CT) und leitet ein Framework her, welches zur Umsetzung von verschiedenen Anwendungen im Bereich der digitalen Volumentomographie verwendet werden kann. Das vorgestellte Framework wird mittels MATLAB und nativer Technologien (C/C++ und CUDA) realisiert, um eine hohe Berechnungsgeschwindigkeit sowie eine einfache Bedienbarkeit und Erweiterbarkeit zu gewährleisten. Auf der Basis dieses Frameworks wird das Verfahren von Katsevich für die Helixtomographie auf Mehrkern-Hardware umgesetzt und mit der FDK-Methode verglichen. Die Ergebnisse dieses Vergleichs zeigen, dass die Helixtomographie die Möglichkeit bietet die Messungenauigkeit zu reduzieren. Jedoch zeigen sich bei der Helixbahn im Vergleich zur FDK-Methode im Bezug auf Echtdaten deutlich mehr Artefakte, welche auf Fehlstellungen des CT-Scanners zurückzuführen sind. Diese Tatsache legt nahe, dass eine präzise Kalibrierung, insbesondere bei Helixscans, unumgänglich ist. Aus diesem Grund wird ein Selbst-Kalibrierungs-Verfahren entworfen, welches die unbekannten Geometrie-Parameter aus den Projektionsdaten schätzen bzw. berechnen kann, ohne dass dafür zusätzliche Messungen notwendig sind. Die vorgestellte Methode verwendet ein 2D-3D Registrierungsverfahren und ein neuartiges Schema zur Neuberechnung der Volumina in Kombination mit einem stochastischen Projektionsoperator. Dadurch kann der Algorithmus akzeptable Laufzeiten und eine Präzision im Subvoxelbereich erreichen, so dass er mit anderen aktuellen Kalibriertechniken vergleichbar ist. Zusätzlich wird in dieser Arbeit ein neues generelles Verfahren für die Optimierung von Filtern für die Tomographie unter begrenztem Aufnahmewinkel diskutiert. Die präsentierte Methode kann mit Hilfe der Approximativen Inverse (AI) zweidimensionale winkelabhängige Filter vorberechnen, und diese Filter in einer gefilterten Rückprojektion für verschiedene Datensätze wiederverwenden. Ein Vergleich mit dem FDK- und dem SIRT-Verfahren zeigen, dass die optimierten Filter eine Bildqualität ähnlich zu der von iterativen Rekonstruktionsverfahren haben, wobei die Laufzeit der optimieren gefilterten Rückprojektion vergleichbar zu der der FDK-Methode ist.

Publications

During the work on his doctoral degree the author has contributed to the research fields discussed within this thesis with the following publications:

- J. Muders, J. Hesser, A. Lachner, and C. Reinhart, "Accuracy Evaluation and Exploration of Measurement Uncertainty for Exact Helical Cone Beam Reconstruction Using Katsevich Filtered Backprojection in Comparison to Circular Feldkamp Reconstruction with Respect to Industrial CT Metrology," in *Proc. International Symposium on Digital Industrial Radiology and Computed Tomography*, Jun. 2011, pp. 1–8
- J. Muders and J. Hesser, "Stable and Robust Geometric Self-Calibration for Cone-Beam CT Using Mutual Information," *IEEE Trans. Nucl. Sci.*, vol. 61, no. 1, pp. 202–217, Feb. 2014
- ——, "Projection-wise filter optimization for limited-angle cone-beam CT using the Approximate Inverse," 2014, submitted to *IEEE Trans. Nucl. Sci.*, accepted and in print

As a consequence, a part of the presented material and results within the following chapters of this thesis are based upon the aforementioned publications.

Contents

For Senta.

♡

Acknowledgments

I would like to thank my advisor, Prof. Dr. Jürgen Hesser, Heidelberg University, for his patient guidance, encouragement and understanding during the past five years. Jürgen has been extremely supportive and has also provided me with insightful discussions about the research. I would also like to thank all members of the Experimental Radiation Oncology group at the University Medical Center Mannheim of Heidelberg University. In particular, I want to thank Dzmitry Stsepankou for the long discussions over the research projects and the many hints and comments on my work. Moreover, I also express my gratitude to the Faculty of Mathematics and Computer Science for providing the foundation for my research making this thesis possible.

I must express my deepest gratitude to Senta, my wife, for her continued support and encouragement. You made me believe that I can put together some meaningless mathematics and computer science to finally build this thesis on top. You helped me through all the ups and downs of my research and gave me the ability to see clearer and to keep focused when I needed it. Without you this work would not have been possible and this thesis would not exist. I love you beyond ∞. Additionally, I thank my parents, my brother and my whole family for their patience and their support. I love you all so much, and I would not have made it this far without you.

I would also like to thank the whole team of Volume Graphics GmbH. Especially, Andreas Kutscheid and Christof Rheinhardt for their support and for providing me always with a current development version of VGStudio MAX, additional helical and circular tomographic datasets and their knowledge in the area of computed tomography. I would also like to express my thanks to Tobias Schön at the Fraunhofer Development Center for X-ray Technology EZRT for providing me the Pen and the TP09 datasets. Finally, I would like to thank Teresa Fiebig and Stefanie Kirschner from the Department of Neuroradiology at the Medical Faculty Mannheim of the University of Heidelberg for helping me with the μCT scans.

Anybody who has been seriously engaged in scientific work of any kind realizes that over the entrance to the gates of the temple of science are written the words: 'Ye must have faith.'

Max Planck

1 Introduction

Cone-beam computed tomography is an imaging technique to acquire the internal structures of an object using X-ray projections from several views around the scanned item. The X-rays which are cast from their source to a detector form a cone-shaped bundle, such that a 2D projection of the object of interest is obtained at each source position. Over recent years, this imaging process has had a tremendous impact on applications in the medical and in the industrial sector. In medicine the high quality 3D images produced by cone-beam computed tomography (CBCT) allow more precise diagnostics than conventional fan-beam tomography and have become increasingly important in treatment planning for the patient positioning and verification in image-guided radiotherapy. In the areas of industrial measurement and non-destructive testing (NDT) tomography has facilitated the inspection and quality assurance process by providing high resolution images up to the nanometer scale [4], which can be used to detect and analyze manufacturing inaccuracies (e.g. casting or soldering defects).

The huge success of 3D CBCT is based on the theoretical work of Cormack and on the first computed tomography (CT) scanner built and patented by Hounsfield and his colleagues in 1972. For this invention and their work in the area of X-ray technology the two researches were awarded with the Nobel Prize in 1979. Since then, new developments and advances in the field of X-ray engineering together with the increasing availability and computational performance of modern computer technology have led to faster scanning and reconstruction times, more efficient dose usage, improvements of image quality and in total to wider spread of applications using CT imaging systems. Along with these proceedings many novel and more efficient image reconstruction algorithms have been derived. A first mathematical solution to the fundamental problem of calculating an image from its

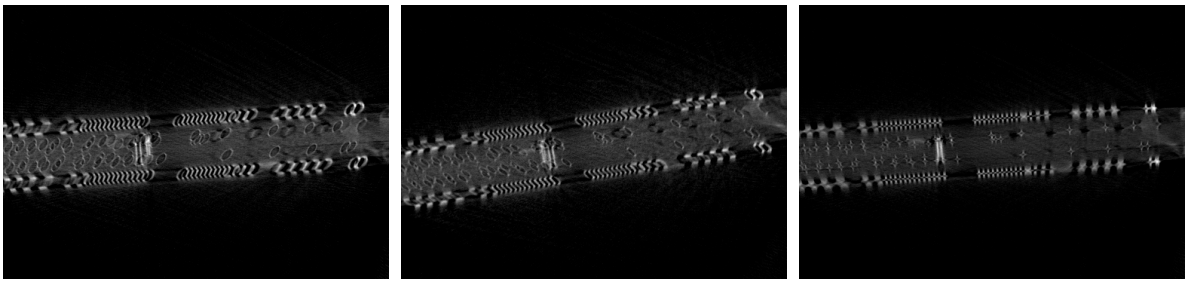
Figure 1.1: Various X-ray CT systems[1]: the YXLON Y.CT Precision (left) is used for precise industrial computed tomography applications of large objects in the fields of inspection and metrology, the microfocus YXLON Y.Fox (center) is suited for the inspection of smaller parts ($17'' \times 21''$) and the Elekta Synergy system for combined kV imaging and MV radiation therapy is used in the medical area.

line integrals (i.e. projections) has been solved by Johann Radon in 1917. These finding have been transfered to the field of tomography and are used to date to introduce and to constantly evolve novel reconstruction algorithms for various CT systems.

In general, there exist many types of CT scanners depending on the area of their application. They can differ in their general geometric and mechanical setup, in their source and detector properties, in the size of the scanner itself and of the scanable objects, in their ease of use, as well as in the possible source trajectories that the machines are able to run. For CBCT the imaging systems in Figure 1.1 depicts three different machines. The left image shows a large-scale CT scanner which has to be installed in radiation-shielded bunker and that designed for precise inspection and metrology processes in industrial CT by delivering highly accurate 3D data, long-term stability through its heavy mechanical construction and rapid results. Also from the field of industrial tomography stems the μCT scanner depicted in the center of Figure 1.1. However, this versatile solution has found a way to other 2D and 3D applications (e.g. high-resolution in vivo scans of mice cerebral vessels [5]) because of its rapid and precise inspection results, its easy operation and its low space requirements. The third machine displayed in Figure 1.1 is an Elekta Synergy system, which combines large field-of-view kilovoltage (kV) and megavoltage (MV) imaging for the utilization in image-guided radiation therapy (IGRT) and intensity-modulated radiation therapy (IMRT). It is equipped with imaging tools to visualize tumor targets and normal tissue during the treatment to optimize the therapy outcome. For this system, the combined use of kV and MV imaging has been suggested by Wertz et al. [6] to provide fast image guidance with low dose and sufficient image quality for accurate patient positioning of lung cancer patients under breathhold.

Although the three aforementioned CT systems differ in their application area and in their mechanical setup, they have several properties in common: (1) they all employ a single X-ray source to acquire projection images, (2) the projections are captured using a flat-panel detector and (3) they all produce cone-beam images of the object of interest. These properties also fit to all CBCT systems

[1]The left and the central images can be found online under www.yxlon.com, whereas the right photo has been taken at the Department of Radiation Oncology of the University Medical Center Mannheim (see www.umm.uni-heidelberg.de/inst/radonk).



(a) (b) (c)

Figure 1.2: Reconstructed images of a memory module. In (a) the reconstruction has been computed from an non-calibrated system by introducing a random misalignment, in (b) the horizontal detector displacement has been estimated manually by fitting lines to features of the projection sinogram and in (c) the true measured geometric parameters have been used.

used for the studies carried out in this thesis and they will be used as preconditions for the developed algorithms in the later chapters. In addition to this, two types of source trajectories will be discussed in the following work: (1) circular and (2) helical scanning geometries. A circular acquisition trajectory can be realized by either placing the object of interest at the rotational center of the imaging system while rotating the source and the detector around the object or by a rotation of the object itself while keeping source and detector fixed. The helical scanning trajectory requires and additional shift of the object along the rotational axis, so that the projected image of the object is moving vertically on the detector. For both of these setups several reconstruction algorithms have been developed and well documented in the books of Natterer and Wübbeling [7], Faridani [8], Buzug [9], Kalender [10] and Shaw [11]. In practice, methods which use a filtered backprojection (FBP) approach are often favored over other techniques, because of their straightforward implementation and their computational performance [9]. For circular CBCT (C-CBCT) an approximate solution of the reconstruction problem has been presented by Feldkamp, Davis and Kress (FDK) [12] in 1984, whereas as for the helical trajectory Katsevich [13] was the first to derive a theoretically exact FBP-type algorithm in 2002. Both of these methods will be used for the developments presented in this thesis.

1.1 Motivation

The overall quality of the reconstructed image depends on many factors including effects that are caused by parameters which are inherent in the CT system setup and parameters manipulating the output of the used algorithms. The here presented work will by mainly concentrated on two of these effects.

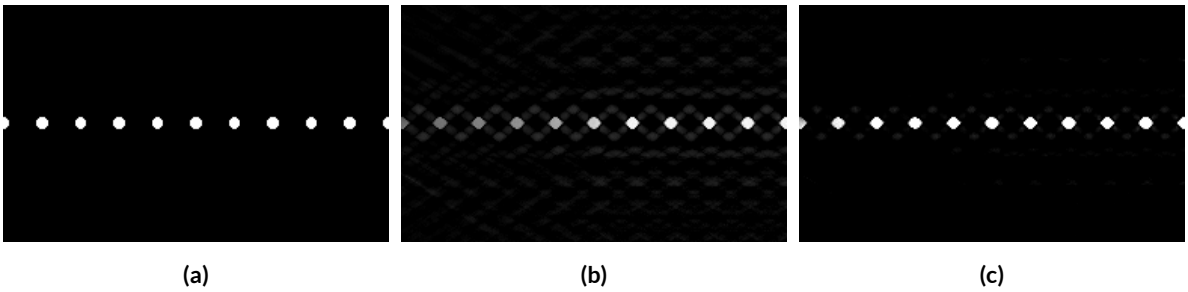


Figure 1.3: Comparison of reference and reconstructed images. The top view (a) and the side view (d) show a BGA, which has been used to generate simulated projections. The images in (b) and (e) show the reconstruction with the standard FBP method. The views in (c) and (f) visualize the reconstruction of the BGA using the iterative SIRT algorithm.

1.1.1 Scanner Calibration

One of the factors that will be studied in this thesis concerns the alignment of a CT system. For computing an accurate solution to the inverse tomographic problem a precise measurement of the geometric system parameters are need in addition to the acquired projection data. If these measurements do not correspond with the true system parameters the reconstruction algorithm will use a false configuration to compute the final image and will introduce artifacts due to the geometric misalignment. Figure 1.2 demonstrates on a scanned memory chip the effects which a misaligned scanning system has on the reconstructed image. The slice image in Figure 1.2a shows how severe these artifacts are when the wrong set of geometric parameters is used to reconstruct the object. Ringing and streak artifacts are introduced, such that the true shape of the scanned item is hardly recognizable. For the reconstruction in the center of Figure 1.2 the value for the horizontal detector offset has been estimated manually. Therefore, the projections have been displayed in sinogram space and pairs of lines have been fitted by hand to corresponding features in the sinogram similar to the method described by Mao et al. [14]. By this straightforward approach the artifacts in the final image have been reduced significantly. However, in comparison to the reference image in Figure 1.2c which has been computed using the precisely measured geometric setup, it can be seen that the hand-tuned reconstruction is still worse than the calibrated one. This example clearly shows that an exact alignment of the scanning geometry is inevitable to achieve accurate results from the chosen reconstruction algorithm. Since a precise and a priori measurement of the geometric parameters which have been used to acquire the projection data might not always be available, methods are needed that can compute the calibration parameters from the dataset itself.

1.1.2 Filter Design

The second aspect which can become a main influence for the image quality of a FBP-type reconstruction algorithm is the design of the used filter. Especially when dealing with undersampled data where the projection count is limited by the angular range or by the number of acquired views, the resulting

image can suffer from artifacts due to missing frequency information. In this case iterative reconstruction techniques can provide a better image quality than FBP methods. This fact is shown in Figure 1.3 which compares a reference image with the reconstructions from limited-angle projection data of a simulated BGA using standard FBP and an iterative simultaneous iterative reconstruction technique (SIRT) algorithm. The SIRT reconstruction in Figure 1.3c shows significantly fewer artifacts than the reconstruction of the FBP in Figure 1.3b. However, the runtime of an iterative algorithm is commonly longer than a single execution of a FBP algorithm. This is why iterative methods are not used as widely in practice as such that employ simple filtration and backprojection operations. As the resolution of projection data increases the computational complexity of iterative algorithms becomes more and more severe, such that methods are needed that achieve the image quality of iterative methods, but have a runtime in the magnitude similar to those of FBP-type algorithms.

1.2 Objectives and Contributions

On top of the motivations given above, the main aim of this thesis is the derivation and analysis of novel algorithms that support the improvement of the image quality in the area of CBCT reconstruction. To achieve this objective the following contributions are presented in this work:

- A detailed description of the implementation of Katsevich's theoretically exact FBP method for helical CBCT (H-CBCT). It will point out major aspects that have to be kept in mind when realizing the algorithm efficiently on the central processing unit (CPU) and graphics processing unit (GPU). Additionally, a study on the accuracy and the measurement uncertainty of Katsevich's H-CBCT reconstruction in comparison to the circular FDK method will show the advantages and shortcomings of both approaches.

- A derivation and analysis of a novel calibration technique for 3D helical and circular CBCT. The method will be able to estimate a set of a priori unknown geometric system parameters from the provided projection data, such that no additional calibration measurements are needed. The proposed algorithm is robust against projection noise, has a reasonable runtime between 5 and 20 minutes depending on the size of the dataset, achieves an alignment of the system such that sub-voxel accuracy is reached and therefore reduces the misalignment artifacts to a minimum.

- A presentation of a general scheme for the construction and optimization of an angle-dependent set of 2D filters for limited-angle CBCT which can be integrated into a FBP-type algorithm. This algorithm is called angle-optimized FBP (AO-FBP). The resulting filters can be precomputed and reused for a specific geometric setup and achieve an image quality that is better than that of the FDK algorithm with respect to image contrast and artifact reduction and is comparable to current state-of-the-art iterative reconstruction techniques. The runtime performance of the algorithm is similar to that of the FDK algorithm.

In addition to each of the contributions above, this thesis contains substantial parts that provide a collection and a discussion of algorithms published by other research groups world-wide. These sections hopefully contribute as an introduction to circular and helical reconstruction techniques, geometric calibration and filter optimization in the area of CBCT. They also can be used as a joint source of references which provides further details on the cited methods. The main contributions above also provide the general outline of the following work.

1.3 Outline

This thesis is structured as follows. For the derivation of new tomographic algorithms a better understanding of the essential prerequisites and needs to process CT datasets is inevitable. Therefore, Chapter 2 will analyze and describe the requirements of a software framework which can be used to develop and evaluate novel algorithms in the area of 3D CBCT. The discussed design will then be used throughout the thesis as a basis for the presented algorithms.

Chapter 3 will give an introduction to FBP methods for circular and helical CBCT. It will present details on the implementation of Katsevich's algorithm in context of the aforementioned software framework. Additionally, this chapter will perform a full evaluation and comparison of the FDK algorithm and the exact method developed by Katsevich. The results will show possible reconstruction artifacts and will underline how important a precise knowledge of geometric proprieties of the tomographic imaging system is.

Based on the previous discussion and on the review of different calibration method, Chapter 4 will derive a novel strategy for the calibration of CBCT scanners and point out how such a technique can be efficiently implemented for helical and circular trajectories. Moreover, a complete evaluation of the novel technique will be conducted to determine its noise and convergence properties.

In Chapter 5 a review of filter design and optimizations methods will provide an overview on current techniques used in the area of tomosynthesis and computed laminography. Then, a new approach for the calculation of optimized filters that can be used in a FBP scheme for limited-angle CBCT will be described. Studies on simulated and real projection data will be performed to demonstrate the image quality of the algorithm.

Finally, the conclusion in Chapter 6 will summarize the presented work of this thesis. Additionally, it will give an outlook on future directions of research with respect to the here derived and discussed algorithms.

I learned very early the difference between knowing the name of something and knowing something.

Richard P. Feynman

2 Foundations

The previous chapter has provided an introduction to the application areas of computed tomography and has given a motivation for the study of novel techniques in the fields of cone-beam computed tomography (CBCT) calibration and filter optimization. This chapter will describe the software tools and the framework which has been specifically developed for computed tomography (CT) applications. The here described foundations will be used throughout this thesis to implement the algorithms which will be derived in following chapters.

In the first part of this chapter the software requirements encompassing the needs and conditions of a framework for computed tomography applications will be analyzed. The second part of this chapter presents in Section 2.2 the design of the software framework with all of its modules and demonstrate how it can be used easily to build CBCT applications. Additionally, this part will then discuss the realization and implementation of the three layers of the derived software framework using MATLAB, C/C++ and Compute Unified Device Architecture (CUDA).

2.1 Requirements

Before starting to design the actual CBCT software framework, the software requirements for applications in tomography will be analyzed. The general design issues presented below will then form the basis for the design phase of the framework in Section 2.2. The following list of specifications will point out important goals that have to be fulfilled by the final set of tools.

Full-Stack: The developed framework has to be complete in the sense that no additional installations of software development tools are needed to build tomography applications. This means

that the framework provides a full set of functionality that suffices to develop algorithms and perform evaluations on tomographic datasets. Therefore the library has to contain methods for data input and output, data visualization and evaluation, as well as methods for simulation of and reconstruction from tomographic projection data.

Compatibility: Multiple operating systems will be supported by the framework. It will run on Windows, Linux and Mac OS X. In addition to this, it has to be compatible with graphics processing units (GPUs) using a current NVIDIA chipset supporting a compute capability higher than 2.0.

Languages: The framework will have to be written in the MATLAB scripting language supported by the use of C/C++ and CUDA to implement performance optimizations at several locations inside the code. The build process of the native source code will have to be organized using CMake by Kitware, Inc. [15].

In-/Output: For data import and export the framework will have to support different file types. These types include: raw binary files with an optional header having a given size, single TIFF images and TIFF image stacks to facilitate simple data exchange between different applications and HIS projection files from an Elekta Synergy® System. Additionally, dBase and CSV files will be needed for the import of projection metadata in to MATLAB. Since, projection data can be stored with different numerical precisions into raw binary files the framework will also have to provide a possibility to load data from raw files with the following common precision types: unsigned integer and signed integer numbers, as well as single and double precision floating point numbers.

Algorithms: To solve various problems in the area of tomography some basic algorithmic tools need to be contained in the framework. As a consequence, the framework needs to provide methods for the generation and the filtering of projection data and for the backprojection of volumes from the computed projections. In addition to this, the offered implementations of several non-iterative and iterative reconstruction methods support the derivation of new algorithms, as they can be used as templates during the development process.

Modeling: Since CT scanners can have or can be configured to use different system geometries, the framework needs to provide methods to model, store and load these different system setups. For the integration with the simulation and reconstruction techniques, the library has to contain functions to compute projections matrices from the given system configurations.

Speed: Applications in CBCT usually have to process large amounts of projection and volume data. Especially, the simulation of projection data and the reconstruction of volumes can be very intensive to compute, in particular for larger datasets. The framework needs to be able to provide means of optimization for such computationally expensive tasks.

Multi-Resolution: During the development of new algorithms for CT, a large amount of time can be spent for the processing of large datasets. As a consequence the framework needs to provide methods to run the same algorithm at different resolution scales. By this the derivation of new approaches can be tested quickly on coarser resolution scales with respect to volume and projection data, whereas the final algorithm can be run exploiting the full amount of available data.

Visualization: To facilitate the development of CBCT algorithms with the novel framework, a set of methods will be provided to visualize the given geometric system setup of a CBCT scanner. The framework will be able to animate a given configuration of a CT system interactively in 3D with the X-ray source moving along the defined trajectory. In addition to this, methods are needed to quickly show, inspect and compare the results delivered by different algorithms.

Evaluation: In addition to the visualization part, the framework needs methods to compute key parameters from the algorithmic results to quantify their image quality and to support the comparison of the implemented methods.

Modularity: To support the integration into a final product (e.g. VGStudio MAX) the framework needs to be modular, so that single modules can be easily extracted from the framework and integrated into another software development environment. Moreover, modularity helps in the optimization and performance improvement of algorithms since single modules which support a specific implementation can be optimized independently from the rest of the procedure.

Extensible: Since the main purpose of the framework is to develop new algorithms for CBCT, it needs to be extensible in the sense to existing and novel methods can be integrated into the framework with a minimal effort.

Simplicity: A requirement that can be seen more or less as a consequence of aforementioned needs is that the framework keeps things simple and straightforward. This means that although the framework provides a full basis for tomographic applications, it is clearly structured, free from complex class hierarchies, easy to learn, not cluttered with optional functionality and thus straightforward to uses and extend.

2.2 Design

The following section will describe the structure of the framework that is used throughout this thesis for the presented tomography applications. During the work on this thesis the here described software library has been built from the ground up, so that the provided tools fit the needs of the applications that will be presented in the later chapters. This section will at first introduce the different layers of the framework to point out at which places of the library specific parts of the functionality are provided and how the source code is organized inside the framework. After this, each of the individual modules

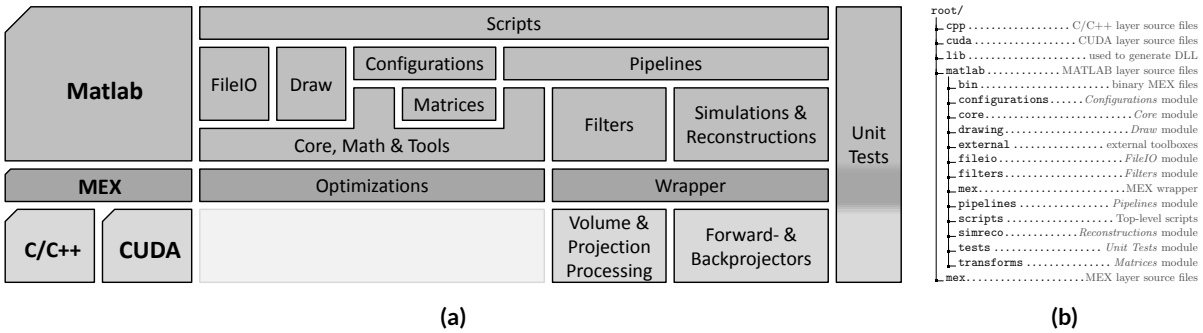


Figure 2.1: Schematic view of the tomography framework (a) combining the potentials of MATLAB and native code using CUDA & C/C++ which can be accessed through the provided MEX wrapper. The whole framework is split up into several modules each designed for specific tasks and covered by unit tests. The folder structure of the framework is shown in (b).

of the framework will be characterized and it will be outlined at which points these modules are open for extensions.

As can be seen in Figure 2.1 the framework is split up into three layers. The top layer is realized in the MATLAB, so that its build-in scripting language can be used to quickly and easily realize novel and modify existing logarithms. One drawback of the MATLAB environment with respect to the application in tomography can become its performance and its ability to optimize parts of the code for the processing of large datasets. This problem can become a bottleneck for CT applications because the capabilities of MATLAB to parallelize algorithms on multiple cores are quite limited [16]. If the amount of data and the complexity of the algorithms increase the overall processing can become slower in MATLAB than in a native implementation. The reason for this is that code written in the MATLAB scripting language is executed by the interpreter on a single processing thread preventing a parallel execution on different cores. As a consequence, the bottom layer of the framework realizes performance critical parts which are essential to CBCT techniques in C/C++ and CUDA on the central processing unit (CPU) and GPU, respectively, such that a parallel execution of specific code segments becomes possible. The two layers above are connected by a third layer which uses the MEX-functionality of MATLAB which allows the wrapping of C-functions and enables their direct usage from inside the MATLAB environment. In addition to this, the middle layer can be used to realize small-scale optimizations directly in a single MEX-file without the access to the functionality provided by the bottom layer. For example an in-place multiplication of large arrays containing complex numbers and the generation of permutations from a large array of numbers has been implemented directly on the MEX-layer. The folder structure of the framework is named in accordance to the three layers. Under the root directory the folders *matlab*, *mex*, *cpp* and *cuda* contain the corresponding parts of the source code. The complete structure of the modules and the layers is summarized schematically in Figure 2.1a and the folder organization is visualized as a commented directory tree in Figure 2.1b.

Altogether, the top, middle and bottom layers of the framework include several modules with each implementing a group of functions and techniques that can be used for various applications in CBCT

and which will be used as a basis for the implementation of the reconstruction techniques described in Chapter 3 and for the derivation of the novel algorithms in the Chapters 4 and 5. In the following the contents and purpose of each of these modules will be outlined and explained:

Core, Math & Tools: This module provides the basic tools used by many parts of the framework. All MATLAB functions belonging to these modules are organized in the *core* folder under the *matlab* part of the framework. The functionality in the *core* toolbox includes methods which allow type and unit conversion (e.g. *bool2string*, *n2s*, *sizeof*, *numdim*, *deg2rad*, *rad2deg*, ...), random number and noise generation (e.g. *shuffle*, *addnoise*, ...), basic image stack processing methods (e.g. *normalize_range*, *invert_image*, *flipud_slices*, ...), gradient computation methods, system utilities, statistical tools and other mathematical functions for evaluation purposes (i.e. *snr*, *cnr*, *mse*, *rmse*, *asf*, ...). All methods belonging to the *core* toolbox are available to all other parts of framework.

FileIO: The file handling module contains methods to import and export projection images and volume data from different file formats into MATLAB. The folder of this toolbox has been named accordingly to the module name (see Figure 2.1b). The supported file formats include: binary RAW files, TIFF image stacks and HIS projections files. For the reading and writing of binary data the corresponding methods provide parameters to save and load MATLAB array data with different numerical precisions (e.g. *int*, *single*, *double*, ...). Additionally, the *FileIO* module contains functions which read and write metadata for projections, volumes and system setups. The meta information will be imported into MATLAB structures and can be exported to different text file formats. By the use of the *FileIO* methods any metadata can be stored as and retrieved from JavaScript Object Notation (JSON), Extensible Markup Language (XML), or comma-separated values (CSV) files. Moreover, the geometric information from an Elekta Synergy can be loaded from dBase files. In addition to the import and export functionality, the *FileIO* module handles automatic ordering and resizing of projection images and volumes in accordance to the supplied geometric system information of the CT scanner.

Draw: The *Draw* module bundles an easy interface to access basic drawing routines which can be used to visualize a given CBCT system setup (see function *draw_geometricsetup*). This includes the visualization the X-ray source position and its trajectory, the location and size of the detector and the bounding box of the reconstructed volume of interest. Additionally, this module provides methods to quickly inspect all projections, filters and slices contained in a set of projection images, in a filter set and inside a given volume, respectively. All of these functions can be found under the *drawing* folder inside the *matlab* tree of the framework. The export of all the figures that can be drawn with these and with other MATLAB routines is also handled inside the *drawing* toolbox by the use of the *export_fig* function.

Matrices: In general, the modeling of the geometry of a CT scanner is addressed through the system matrix A, which for a given object of interest f computes the projection data g by the definition

Table 2.1: Naming conventions for the fields of the configuration structure

Parameter Description	Field Name [Unit]	Symbol
System Geometry		
Number of detector columns	*sizeU*	N_u
Number of detector rows	*sizeV*	N_v
Number of projections	*sizeV*	N_s
Detector column width	*deltaU* [mm]	Δ_u
Detector row height	*deltaV* [mm]	Δ_v
Distance source-to-projection	*Dsp* [mm]	D_{SP}
Distance source-to-ISO-center	*Dsi* [mm]	D_{SI}
Start angle	*startS* [°]	s_0
Angular increment	*deltaS* [°]	Δ_s
Start z-position	*startZ* [mm]	z_0
z-increment	*deltaZ* [mm]	Δ_z
Volume Geometry		
Number of voxels in x-direction	*sizeX*	N_x
Number of voxels in y-direction	*sizeY*	N_y
Number of voxels in z-direction	*sizeZ*	N_z
Voxel edge length in x-direction	*deltaX* [mm]	Δ_x
Voxel edge length in y-direction	*deltaY* [mm]	Δ_y
Voxel edge length in z-direction	*deltaZ* [mm]	Δ_z

Parameter Description	Field Name [Unit]
File Import	
Number of projection columns	*loadU*
Number of projection rows	*loadV*
Number of projections	*loadS*
File format (RAW, TIFF or HIS)	*FileFormat*
Size of the file header (to be skipped)	*HeaderSize* [byte]
Bit depth (*uint16*, *single*, *double*, ...)	*BitDepth*
Flip projections upside-down	*flip_ud*
Flip projections left-right	*flip_lr*
Flip projections first-last	*flip_fl*
Transformation Matrices	
Projection of voxel indices to pixel coordinates on the aligned detector	*Mva*
Projection of voxel indices to pixel coordinates on the misaligned detector	*Mv*
Projection of world space to pixel coordinates on the misaligned detector	*Mw*

of the connected inverse problem: $Af = g$. However, in the case of 3D CBCT the matrix A cannot be computed or stored directly because of its enormous size [17]. As a result, the modeling of the system is commonly represented by projection operators which are steered by projection matrices as described by Galigekere et al. [18]. For this reason, the framework comes with several routines that facilitate the generation of matrices in homogeneous coordinates. For the creation of the transformation matrices (translation, rotation and scale) as described in Appendix A, basic functions are provided by the framework. These methods are then used to compose more complex transformations between coordinate systems that can be defined for various system geometries (compare Section 4.3). In addition to this, the generation of a fundamental projection matrix with respect to a given coordinate system and X-ray source position is handled by the *matrix_projection* function as described in Section 4.3.1. All tools associated with the calculation of matrices are bundled under the *transforms* directory of the *matlab* tree.

Configurations: The *Configurations* module provides on top of the *Matrices* module the possibility to define the geometric parameters of a CT scanner. Inside the *configurations* folder each system setup can be declared in a separate MATLAB script file (m-file) which additionally defines a parameter-less function that returns a structure containing a field for each system parameter and fields with the associated transformation matrices. The field names for the geometric parameters inside the structure follow the naming conventions defined in Table 2.1. For helical CBCT (H-CBCT) the *Configurations* module provides the function *compute_maximal_pitch* which computes the helical pitch in such a way that the maximal possible area of the detector

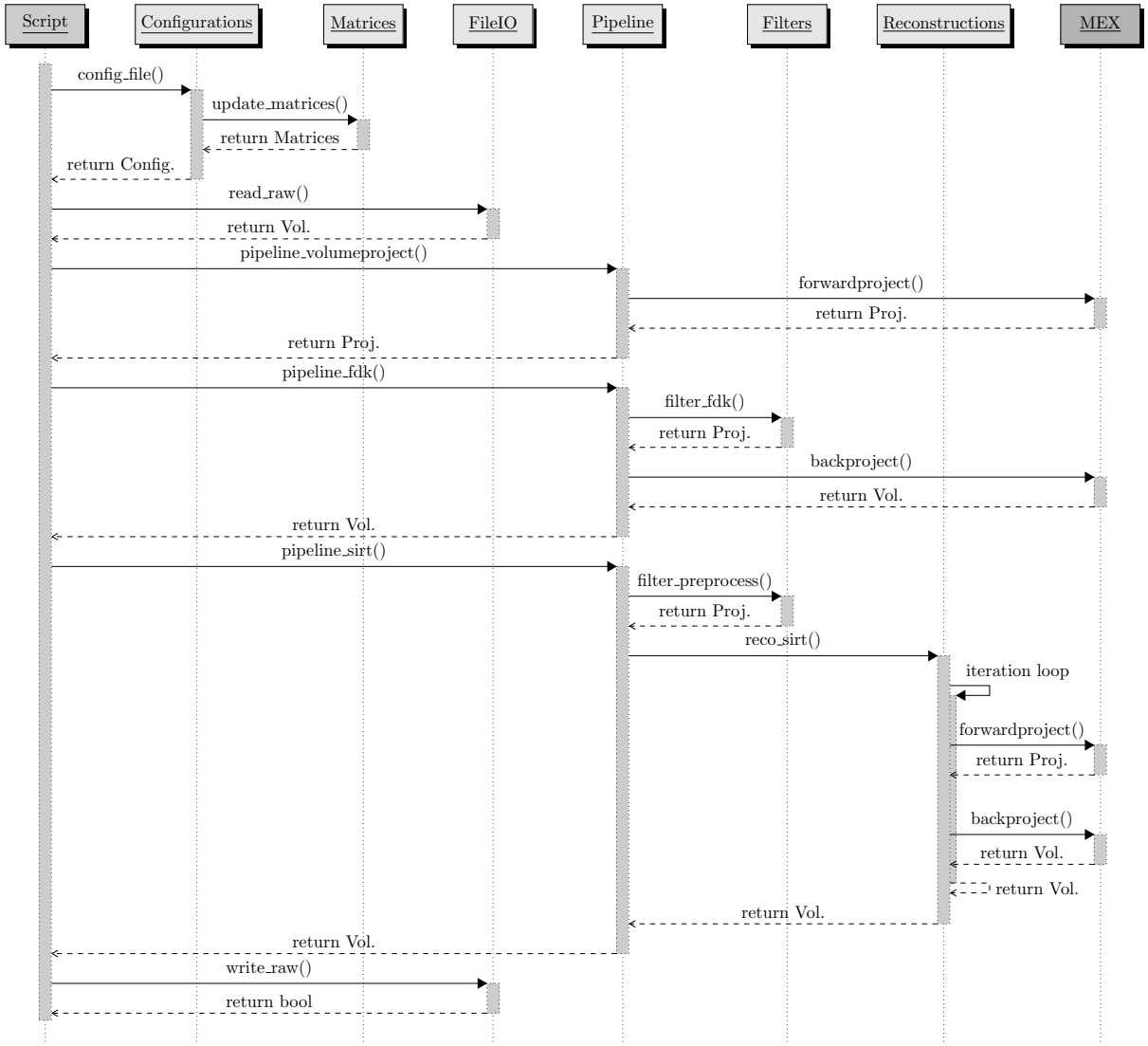


Figure 2.2: Call sequence diagram of a MATALB script using the CBCT tomography framework for the loading of a volume, the simulation of projections and two consecutive reconstructions of volumes using the FDK and the SIRT algorithm. The involved modules are depicted at the top of the diagram and the bottom layer the framework (C/C++ and CUDA) has been excluded from the call stack for reasons of clarity and comprehensibility.

can be used in conjunction with Katsevich's reconstruction algorithm [13, 19]. The maximal pitch can then be used to calculate the parameters *startZ* and *deltaZ*.

Once a system setup has been parameterized, the framework can use the supplied values to compute the transformation matrices which are needed by the projection operators. The calculation of the matrices for each projection is handled by the *updated_matrices* function automatically when the configuration function of a previously defined system setup is called. This behavior is also depicted in the upper part of Figure 2.2. The resulting matrices are stored with the configuration structure in accordance to the field names shown in the lower right part of Table 2.1. For the case that some geometric parameters have to be changed after the initial calculation of the matrices, the *Configurations* module provides special update functions (e.g.

Listing 2.1: Simplified version of the FDK filerting routine in MATLAB

```
function P = filter_fdk(P, H)

% compute size of the padding
[sizeV, sizeU, sizeS] = size(P);
pad = length(H)-sizeU;
prePad = floor(pad*0.5);
postPad = pad-prePad;

H = repmat(H, [sizeV 1]);
for i=1:sizeS % projection loop
    % replicate border of current projection
    I = P(:,:,i);
    I = padarray(I, [0 prePad ], 'replicate', 'pre');
    I = padarray(I, [0 postPad], 'replicate', 'post');

    % fast frequency domain filtering
    I = real(ifft(fft(I, [], 2) .* H, [], 2));

    % truncate the filtered projections
    P(:,:,i) = I(:,prePad+1:prePad+sizeU);
end
```

update_config_projection_scale, *update_config_volume_set_size*, ...), to modify these parameters while keeping the transformation matrices always up-to-date.

In addition to the parameters that define the system and the volume geometry, the configuration structure contains fields that are used by the *FileIO* module during the import of projection images. These fields are displayed in the upper right part of Table 2.1. They are interpreted together with the system parameters by the file loading routines to read the different projection image formats into MATLAB correctly. Then the imported images are preprocessed by a rescaling and by correcting their original orientation to the specified one if needed. As the configuration structure contains all necessary information about the CT system and since it can be easily extended with additional fields anywhere throughout the framework many of the internal routines follow a syntax that expects only this structure and a data supplying array (i.e. projections or volumes) as in- and output. The resulting syntax for a method *do_something* that awaits the configuration structure *C*, the input array *D_in* and returns its results to the array *D_out* can generally be written as: *D_out = do_something(D_in, C)*.

Filters: For the computation and the application of filters to projection data the *Filters* module inside the *filters* directory of the framework comes with to separate classes of methods: (1) functions to compute filters in the spatial or in the Fourier domain and (2) routines that apply the resulting filter to a set of projections. Functions of the first category have the prefix *compute_*, whereas function from the second category start with *filter_* in their name. A simplified version of the FDK filtering is depicted in Listing 2.1, which shows how to carry out the in-place linewise filtering of the projection *P* using the fast Fourier transform (FFT) with a precomputed

filter H. The filter H has been computed by the *compute_fdk_filter* function. The *Filters* module can be easily extended by additional m-files which follow the aforementioned syntax.

Simulations & Reconstructions: This module can be found in the *simreco* folder inside the MATLAB layer. It provides implementations for the simulation of projections from numerical phantoms (i.e. balls, wires, Shepp-Logan, ...) and from discrete volumetric datasets. In addition to this, the module contains common non-iterative and iterative reconstruction techniques used in the area of CT. By the introduction of operator concept[1], which reflects the system matrix of the inverse problem $Af = g$ through the use of forward- and backprojection functions provided by the MEX layer in conjunction with a given configuration structure, the reconstruction techniques are decoupled from the modeling of the CT system. This approach has basically two advantages: (1) the reconstruction techniques can be run with different forward- and backprojection models and (2) additional reconstruction and simulation techniques are more straightforward to implement since the operators provide a common interface to the forward- and backprojection routines without the overhead of passing geometric system parameters around.

Pipelines: The *Pipeline* module can be seen as logical grouping of methods to support easy and quick access to the lower level functionality of the framework. A pipeline can be thought of as a sequence of processing steps. Under the application of each step a pipeline computes the output dataset from the input data with respect to the supplied configuration. Additionally to the configuration structure, the input and output parameters of a pipeline are volumes and sets of projection images. A single step carried out inside of a pipeline can range from basic calls to MATLAB's internal functions to the use of filters, simulation and reconstruction techniques implemented in the other modules of the framework. Like depicted in Figure 2.2 the framework comes with several predefined pipelines (e.g. *pipeline_volumeproject* and *pipeline_fdk*) which can be used as templates to define new ones inside the *pipelines* folder.

Scripts: Scripts are the top level routines provided with the framework. Scripts are free from function definitions and their contained commands are processed sequentially. As an example, Figure 2.2 shows a call sequence diagram for a script that first initializes a CT system setup by the use of the *Configurations* module, then imports a volume dataset from a binary file, generates simulated projections, reconstructs images using the FDK and SIRT algorithms and finally writes the results in a binary file to the hard disk. More exemplary scripts can be found under the *scripts* folder of the MATLAB layer.

Wrapper: The *Wrapper* module in the MEX layer provides the connection between the algorithms implemented in native code (C/C++ and CUDA) and the MATLAB environment. Therefore it has been divided in to two parts. The first part of the module contains the compiled MEX

[1]For the implementation of linear-operators the *Spot* toolbox has been used. The toolbox can be found online at github.com/mpf/spot with a complete tutorial under www.cs.ubc.ca/labs/scl/spot.

functions under the *bin* folder of the MATLAB layer. These binaries have been generated with and without debug information which are stored in the *Debug* and *Release* subdirectories, respectively, and can be called directly from the within MATLAB. In addition to this, the second part of the wrapper provides a convenience layer with the functions in the *mex* directory that allow an easier interaction between MATLAB and the MEX binaries. These functions handle the preprocessing of data arrays and MALTAB structures, pass them to the MEX methods and post process the resulting data. To gain an interactive feedback (e.g. error and warning messages or a progress indicator) from the native code layer inside the MATLAB environment the techniques described in Appendix B have been used.

In addition to MEX layer described above, the framework provides an alternative use of the native code: a dynamic-link library (DLL) which bundles the functionality of the C/C++ and CUDA layer can be built from the definitions and implementations in the *lib* folder (see Figure 2.1b). This DLL-file can then be loaded in MATLAB's environment and used to make direct calls to its included functions from within m-files using MATLAB's internal library methods *loadlibrary* and *calllib*, respectively. Furthermore, this DLL can be used to integrate the low level functionality of the framework into other libraries and applications. However, the MEX wrapper used to access the C/C++ and CUDA functionally showed slight performance advantages over the DLL-solution.

Volume & Projection Processing: This module has been fully realized in C/C++ and provides functions that were not available in the MATLAB environment and for which a native implementation resulted in a superior performance. This is why routines that operate on full sets of projections (e.g. *filter_helix*, *filter_virtualdetector*, *metric_mattes_mutual_information*, ...) and volume datasets (e.g. *resize3*) have been realized on the bottom layer inside this module. Further details on the implementation of these functions will be given at the corresponding locations in the Chapters 3 to 5.

Forward- & Backprojectors: In addition to the projection and volume processing, the forward- and backprojection methods have been realized in a module using CUDA. The framework provides two different implementations for projection operators: (1) sampling-based ray casting as presented by Weinlich et al. [20] and (2) the use of separable footprints as suggested in [21, 22]. In addition to this, a stochastic forward projection operator has been implemented on that basis of the ray casting approach and will be described in more detail in Section 4.3.

Unit Tests: For the validation of the correctness of the implemented tools and algorithms inside the framework unit test have been written in MATLAB. These tests cover not only the MATLAB routines of the framework as they also test the functions realized in C/C++ and CUDA by calling them through the MEX wrapper. In particular, the forward- and backprojection methods have been tested for their correctness by paying special attention to the projected locations of the involved pixels and voxels, such that artifacts caused by errors in the implementation of the

projection operators were detected and removed. Unit testing greatly reduced the number of bugs and allowed an effortless integration of some components into VGStudio MAX already.

All the modules described above are loaded into the MATLAB environment by a single call the *devenv* script which resides in the MATLAB module path. This script can also be seen as a configuration file, which defines the modules to be loaded during initialization of the frame work and it can be easily extended to include novel additional or existing external modules into the framework.

What we observe is not nature itself, but nature exposed to our method of questioning.

Werner Heisenberg

3

Circular and Helical Cone-Beam CT

Depending on the physical properties of the scanned object the requirements on the mechanical setups of computed tomography (CT) systems differ with their usage in the medical and industrial area. To encompass the needs of those fields of interest the employed scan trajectory constitutes a key factor, as the specific X-ray source positions and the detector locations restrict the types of objects that can be scanned. Two basic types of system geometries are widely used in both areas: acquisition systems (1) with a circular and (2) with a helical scanning trajectory.

On the one hand a simple geometric system setup, where the source and the detector move on a circular path relative to the scanned item, lowers the demands and the costs during the manufacturing process of the CT machine. This is one of the main reasons why circular CBCT (C-CBCT) has been used intensively in medical and industrial applications, so that for the processing and the reconstruction of data from these types of systems a broad spectrum of efficient and stable algorithms have become available. On the other hand the data acquisition and analysis of large objects, especially in the industrial sector, require more complex geometrical configurations of the CT system. In particular, the increasing interest in the scanning of long mechanical objects, e.g. helicopter main rotor blades [23], has leveraged the advances made in helical computed tomography.

From the algorithmic point of view, the scanning trajectory used for the data acquisitions is one important factor for the selection of applied reconstruction method. In practice, data acquired from circular scanning path is commonly reconstructed using the Feldkamp, Davis and Kress (FDK) algorithm [12]. Since this approach is based on a filtered backprojection (FBP)-scheme it can be realized in a two-step implementation (i.e. filtering and backprojection), which has been long studied, well documented and optimized in tomography literature [24, 25]. The breakthrough for helical

CBCT (H-CBCT) has been achieved by Katsevich [13, 19] in 2002 with his first presentation of a theoretically exact reconstruction formula that can be realized using a FBP-type method.

When looking at the field of industrial CT the reconstruction accuracy and the reliability of measurement are vitally important for metrology and non-destructive testing (NDT) applications. However, these aspects are influenced by many system intrinsic and algorithmic parameters. The first set of parameters is mainly governed by the employed CT system, whereas the latter set depends on the chosen reconstruction method. To achieve an optimal image quality and to reduce the measurement errors to a minimum, a combined parameter configuration has to be selected specifically for each CT system and algorithmic setup. With respect to the FDK algorithm the crucial factors have been studied intensively by Barrett and Keat [26] and it has been shown that for cone-beam computed tomography (CBCT) the circular scan trajectory can produce major artifacts in the images. These inaccuracies occur because the amount of data acquired from the circular path is not sufficient for an exact and artifact-free reconstruction. Consequently, only an approximation to the real object of interest is possible. This approximation is more inaccurate at locations where the sampling provided by the projection data is sparse, such that artifacts become more prominent, especially at the top and bottom of the object of interest (compare with Muders et al. [1]).

On the contrary, the method proposed by Katsevich allows the reconstruction of helical projection data without the introduction of artifacts at the axial borders of the computed object. Because of the theoretical exactness of this approach it can be expected to deliver more accurate results than the standard FDK method. However, Katsevich's algorithm is non-trivial to realize on current parallel high performance hardware due to the complexity of its filtration and backprojection steps (compare [12] with [13]). Although various implementations [27–30] of Katsevich's exact formula have been reported in literature, these studies suggest that the specific details of each of these implementations influence the resulting reconstructions. Thus, the implementation details form an important factor together with the geometric and other algorithmic parameters in relation to the image quality which can be achieved by Katsevich's helical reconstruction approach. These numerous degrees of freedom can be seen as one of the main reasons why the exact helical FBP is not frequently used in the area of industrial CT, where only few studies have been conducted with real projection data [23, 31, 32] so far. But to gain reliable dimensional measurements from datasets reconstructed with Katsevich's method, it needs to well implemented and tested.

Therefore, this chapter has several aims. First, it will provide a detailed discussion and study of the implementation of Katsevich's algorithm for H-CBCT to point out the key aspects which have to be taken care of when realizing the method on current parallel hardware. Further, the goal of this chapter is to demonstrate the applicability of Katsevich's approach in industrial CT by investigating its strengths and weaknesses in comparison to the conventional FDK method. The here presented study will use simulated and real projection data from circular and helical trajectories to compare C-CBCT and H-CBCT reconstructions of a calibrated cubical object with spherical caps, i.e. a calotte cube (Kugelkalottenwürfel) (KKW), to identify and to analyze the most important factors that influence the overall image accuracy and the measurement uncertainty in terms of length and form devia-

tions. Therefore, the reconstructed images will be analyzed in VGStudio MAX[1]by using coordinate measurement techniques followed by a statistical evaluation to detect differences in their gray values and in their geometrical shape. The exactness of the conducted measurements will be quantified and compared for both algorithms by simultaneous and systematic tuning of the input parameters.

As a result, the presented experiments will outline the measurement uncertainty that can be expected and will highlight the limitations of both reconstruction techniques in relation to the field of industrial CT. The resulting statistics of the simulated projections will show that Katsevich's exact helical is able to achieve superior image quality and that it can theoretically lead to more precise measurements than the circular FDK method. In contrast to this, the measurements performed on the real helical datasets suffer from spiral shaped artifacts. These imprecisions are a result of a misaligned scanning geometry and show how sensible Katsevich's algorithm is to geometric system parameters. This fact makes a precise calibration during the data acquisition phase inevitable.

This chapter is organized as follows. At first, Section 3.1 will give an overview of the current state-of-the-art in CBCT with emphasis on the advances in exact helical reconstruction techniques. Then the implementation details of the FDK method and Katsevich's algorithm on multi-core hardware based on the software framework presented in the previous chapter will be given in Section 3.2. Additionally, this section will provide an overview of CBCT scanning systems with the geometric parameters used to generate the projection data. Afterwards, the coordinate measurement methods used to evaluate and to compare the circular and helical scans will be introduced, followed by a description of the conducted experiments and their results in Section 3.3. Finally, in Section 3.4 the results of these evaluations will be discussed in the context of measurement uncertainty and conclusion will be drawn. It has to be considered that the author of this thesis has discussed parts of the material described in this chapter in [1]. Consequently, some tables and figures within this chapter have been taken over or have been adapted from this publication.

3.1 Related Work

3.1.1 Circular CBCT Reconstruction

In the field of circular CBCT with respects to FBP reconstruction techniques much research has been conducted over the past 30 years. One fundamental publication in this area presented the method of Feldkamp, Davis and Kress (FDK) [12] for the reconstruction from projection data acquired from a circular trajectory using FBP. By the incorporation of a priori knowledge into an additional correction term the FDK method has been extended by Hui [33] to provide more accurate reconstructions than Feldkamp's original approach. In order to compensate missing projection information due to the circular scanning path Mori et al. [34] introduced a weighting scheme to reduce cone-beam artifacts and to enhance image quality, such that it cannot only be applied in medical multi-slice CT, but also for flat-panel detector data acquire with wide cone-angles. The density drop artifact along

[1]VGStudio MAX is a product of Volume Graphics GmbH which can be found online at www.volumegraphics.com

the dimension of the cone-angle has been reduced even further by Zheng et al. [35] using a heuristic weighting function which corrects the angle mismatch of the reconstructed plane from the midplane. The results of this approach show a significant improvement in relation to the reconstruction artifacts and demonstrate that this method can also be used to increase the resolution of the FDK algorithm. Moreover, the past developments in the area of flat-detector computed tomography (FDCT) have been summarized and reviewed by Kalender and Kyriakou [36] with a focus on the technical and performance critical aspects of these techniques in comparisons to standard clinical CT. They concluded that FDCT has superior spatial resolution properties, but implies a lower dose efficient, a reduced field of view and less resolution in the time-domain. However, the review points out the practical advantages of FDCT and describes its application areas, e.g. in interventional radiology for immediate imaging during surgeries.

Furthermore, Turbell [25] gives an overview of alternative algorithms used for circular CBCT and helical CBCT in comparison to the traditional FDK approach, while he also formulates a version of the FDK algorithm that runs in $\mathcal{O}(N^3 \log N)$ steps instead of $\mathcal{O}(N^4)$ needed by the originally proposed variant. A runtime performance in the same order of magnitude has been achieved by Xiao et al. [37] using a recursive hierarchical decomposition of the backprojection operator into sub-volumes, such that a 7-fold speedup has been measured for a 128^3-voxel volume. In addition to these algorithmic improvements, the reconstruction from circular cone-beam data has recently become possible in real-time through the use of additional hardware. Stsepankou et al. [24] suggested to speed up the backprojection step by up to a factor of 6 using field-programmable gate arrays (FPGAs), while the image quality of the reconstruction is not degraded by the limited precision of the hardware. Moreover, Xu and Mueller [38] demonstrated how to exploit the programmable shading pipeline of commodity graphics processing units (GPUs) devices to compute FDK reconstruction results immediately after the image acquisition. On a PC equipped with a single GPU, their techniques were able to process 40-50 projections per second with respect to a reconstruction of a 512^3-voxel volume. Instead of using graphics hardware, the backprojection algorithms presented by Knaup et al. [39], Scherl et al. [40], and Kachelrießet al. [41] have been performance optimized for the Cell Broadband Engine Architecture (CBEA). By realizing the filtration and backprojection step in a parallel manner on the Cell-processor, they were able to achieve a full run of the FDK method in 6.2 seconds for a 512^3 cube, whereas a full reconstruction of their central processing unit (CPU) implementation took 3.2 minutes. For a standard clinical scenario these speed suffices to present the reconstructed image in real-time, since the times for data acquisition are usually much higher than the reconstruction times. With the emergence of NVIDIA's Compute Unified Device Architecture (CUDA), a C language environment and programming model for parallel computing on GPUs, many researchers [42–48] have proposed similar algorithms to accelerate the FDK filtered backprojection algorithm on graphics hardware. In comparison to the CBEA-based techniques the realizations using CUDA have a lower implementation effort (see Scherl et al. [43]), while through the use of off-chip memory access reduction, loop unrolling and multi-threading over multiple GPUs a reconstructed image has been computed in under 5 seconds for a 512^3-voxel volume in [47]. A more general overview on

the recent use of graphics processing units (GPUs) in the area of medical physic, encompassing image reconstruction techniques, has been presented by Pratx and Xing [49].

With respect to the FDK algorithm the factors that influence the overall quality of the reconstructed image have been studied intensively in literature. Soimu et al. [50], for example, have conducted experiments to identify and quantify the effects of several acquisition problems, such as large cone angle, projection undersampling, limited angular range and truncated projection data. Since the data sufficiency condition is not fulfilled by the circular cone-beam data, the Feldkamp algorithm has been shown to perform worse than iterative reconstruction methods with respect to cone-beam artifacts, although its computational costs are lower than those of other approaches [51]. A full review of the most prominent artifacts in today's 3D CBCT that occur due to the discrepancies between the theoretically derived system model and the actual physics behind the imaging process has been presented recently by Schulze et al. [52] with a focus on the application of the standard FDK algorithm in the medical area. In the industrial field, a tool for the optimal placement of the scanned specimen has been suggested by Amirkhanov et al. [53] to reduce the amount of image artifacts significantly. Moreover, a survey with respect to important aspects of industrial CT meteorology, such as accuracy of the unit of length and measurement uncertainty has been conducted by Kruth et al. [54].

3.1.2 Helical CBCT Reconstruction

In 2001 and 2002 the analytical derivation by Katsevich [13, 55] of a theoretically exact inversion formula which can be expressed as a filtered backprojection algorithm formed a breakthrough in the area of helical CBCT. By the using a shift-invariant filtering of the derivative of the cone-beam projection data followed by a backprojection operation, Katsevich's approach solves the long object problem. In the initially proposed algorithm the filtration part can be carried out by two different schemes (see [56]): (1) using a single filtering step requiring differentiation between adjacent projections and (2) employing a more complicated filtering step with projection independent filtering, but having the requirement to backproject the filtered data twice with different weighting functions. The mathematical properties of Katsevich's inversion formula and the numerical precision of its implementation have been studied further in [57]. Within this article it has also been shown that the 2D Radon transform becomes a special case of Katsevich's formula when the helical pitch vanishes. In addition to this, Katsevich [58] has provided a general scheme for the derivation of FBP-type reconstruction algorithms for source trajectories that fulfill Tuy's sufficiency condition [59]. He also showed how to choose the weights during the construction of the inversion formula, such that the long object problem is solved.

On the basis of this research, Katsevich et al. [60] have functionally extended the original algorithm to a helical scanner setup where the pitch varies over time. In addition to this, an improved version of the exact FBP method has been proposed in [19], which, in comparison to the earlier variants, requires a smaller detector, puts no restrictions on the scanned object inside the gantry and is by a factor of two faster. Similar properties have also been achieved by an alternative FBP algorithm which

has been suggested by Zou and Pan [61] and which also performs a theoretically exact reconstruction for specific regions of interest from a minimal amount of projection data to reduce the radiation dose. By construction and similar to Katsevich's algorithm, Zou and Pan's method is also based on the concepts of π-lines and π-segments (see Section 3.2.1.1) which need to be computed by numerical optimization methods (compare [62, 63]) during the reconstruction phase. This fact can become a performance critical factor during the implementation of these methods. Therefore, Yang et al. [62] have developed the cone beam cover (CBC) method to realize Katsevich's inversion algorithm without the computation of π-segments (compare Section 3.2.1.5). While the aforementioned algorithms only used projection data from within a 1π-window, the method presented by Katsevich [64] uses redundant information from a three times larger detector region to improve the quality of the produced images. A summary and a comparison between the results from the 1π and 3π approaches can be found in [65].

An additional extension of the Katsevich's algorithm has been constructed in [66] with the requirements of novel electron-beam CT in mind, such that the inversion formula can be applied to variable radius spiral CBCT systems. The proposed reconstruction technique has been formulated in the form of a backprojected filtration (BPF) as well as in FBP-format. Yu et al. have shown that their BPF-based implementation requires less projection data, whereas the FBP approach delivers a superior image quality. An even more general approach to exact image reconstruction from cone-beam projections acquired along rather arbitrary scanning curves has been developed by Ye and Wang [67]. They have also discussed conditions for filtering directions and proposed a more convenient scheme than it has been provided by the previous publications of Katsevich. As a result, the method of Ye and Wang can be applied to non-standard spirals with non-constant radii and pitches as wells as to saddle trajectories. Later, Katsevich and Kapralov [68] also presented an exact and efficient FBP inversion formula for a general class of smooth, non-self-intersecting curves and generalized the selection of π-lines to achieve an acceptable image quality. A major difference between algorithms outlined above lies in the selection of filtering lines on the detector. A unifying framework for these approaches has been described in [69], such that a new 2D FBP inversion formula is obtained which can be utilized to construct FBP and BPF algorithms for general scanning paths. In addition to this, the works of Chen et al. [70] and Yarman [71] have demonstrated that by the design and the incorporation of appropriate weighting functions the development of novel FBP reconstruction algorithms for arbitrary source trajectories can also be achieved. Moreover, an optimized Katsevich-type algorithm for helical CBCT systems with a fractional pitch has been proposed by Katsevich et al. [72] by solving a least square problem to keep the weights of the Radon planes close to 1. Further applications of exact helical FBP algorithms to dual- and triple-source CT have been conducted in [73] and [74], respectively.

One important aspect that influences the resolution and image quality of the theoretically exact CBCT reconstruction is the implementation of the differentiation prior to the filtration phase. As mentioned above and as described in [56], the differentiation can be carried out in a view-dependent and in a view-independent manner, whereby the latter version implies more complex filter and backprojection operations. Therefore, several view-dependent schemes have been suggested and studied

in literature. Noo et al. [75] have developed a differentiation technique that can be employed with different sampling geometries. They have compared it to a direct differentiation and a chain-rule-based scheme. Additionally, Katsevich [76] presented a formula to compute the derivative by minimizing the conducted interpolation steps, such that resolution and noise properties are similar the method described by Noo et al. [75]. However, his differencing scheme is simpler to implement and computationally more efficient. By exploring the properties of Katsevich's algorithm in the limit of vanishing pitch, Faridani et al. [77] have derived another promising discretization scheme for the derivatives. At the same time, they have shown that a shift of the filtering kernel can be employed to remove ringing artifacts in the reconstructed images. Only quite recently, Wang [78] used a polynomial interpolation function to increase the precision of the numerical derivative computations and thus achieved sharper and more accurate images.

The first concrete implementation of Katsevich's algorithm has been described by Noo et al. [27] for the curved and flat detector geometry. With respect to both setups they have derived equations for finitely sampled data from their continuous forms, so that the resulting method realizes Katsevich's analytical inversion formula efficiently and accurately. Additional details and an implementation of this approach in MATLAB can be found in the master thesis of Wunderlich [79]. For the flat detector setup, Yu and Wang [80] followed a quite similar approach, but they pointed out several important differences to the aforementioned implementation which influence the overall image quality: (1) the necessity of a continuous handling of the endpoints of the π-segments, (2) the superior performance of a 2-point difference formula over an 8-point one, (3) the relative insensitivity of the reconstruction quality to the number of filtering lines within a certain interval and (4) the optimality of the rectangular window function with respect to Hilbert transform. An alternative implementation, which however did not prove as computationally efficient as the two previous ones, exploits invariance of the reconstruction geometry and has been derived by Weber [81] in his Diploma thesis. In the context of the aforementioned techniques, the computation of π-lines for each voxel inside the volume of interest can become a bottleneck of the algorithms. However, the fast algorithm presented by Izen [63] uses Newton's method to achieve at least a quadratic rate of convergence for π-interval computations.

The increasing algorithmic complexity in exact helical CBCT algorithms poses a big challenge on the effective solution of the reconstruction problem. Since a major part of the computation time is spent in the backprojection operation, Yang et al. [28] have proposed to parallelize this step in a cluster environment using Message Passing Interface (MPI) in combination with the CBC method, such that each node can process cone-beam projections independently. Deng et al. [82] have extended this technique to run both parts, the filtering and the backprojection, on a high performance computing (HPC) cluster achieving significant speedups. The resulting performance increase of their approach has been analyzed in detail in [83, 84] by the use of estimations from an analytical model. In contrary to a cluster-based solution other groups have exploited geometric symmetry properties together with optimized data structures to realize Katsevich's algorithm on multi-core PC hardware in a multi-threaded single-instruction multiple-data (SIMD) fashion [29, 85]. For their solutions an overall performance improvement of up to a factor of 40 has been reported. However, Fontaine

and Lee [29] suppose that the scalability of the algorithm is limited by the memory bandwidth. For this reason, Xue et al. [86] has designed a novel ray traversal strategy for Katsevich's algorithm which reduces the total memory consumption significantly by reading projection values only once. In addition to this, the precomputation of lookup tables (LUTs) for detector line coordinates and voxel weights allowed a full vectorization of the filtering and backprojection steps [87] that performed well on the CBEA [88]. This technique allowed helical CT reconstructions for a typical clinical scanner within a few seconds, such that the resulting images can be made available in almost real-time. An even higher speedup has been achieved by the implementation of Bi et al. [89] where the convolution and the backprojection steps have been fully realized on the GPU. In comparison to a pure CPU implementation the runtime of the GPU version was about 20-100 times faster. Another implementation which used the graphics hardware to accelerate Katsevich's backprojection step, while filtering on the CPU, has been proposed by Yan et al. [90] on the basis of the OpenGL standard and the Cg shading language. This publication also presented a novel volume blocking scheme based on an analytically derived overscan formula, which can be used to reconstruct large volume datasets efficiently. Alternative implementations of Katsevich's algorithm using NVIDIA's CUDA [30][2]or FPGA hardware [91] have also reported similar speedups. Note that Kachelrieß [87] has compared several of the aforementioned methods in terms of runtime performance against each other.

In addition to the numerical studies with simulated data conducted in many of the publications above, other researchers have investigated the image quality of Katsevich's method. Zhu et al. [92] have compared it to Feldkamp-type algorithms on helical trajectories and found out that for small cone-angles both methods deliver a similar quality. However, when the cone angle increases or when sharp density changes occur along the axial direction the exact FBP is preferable. A comparison of four different artifact types that can depend on the numerical realization of the individual steps of Katsevich's algorithm have been evaluated in [93] and it has been shown how to reduce and remove these inaccuracies. More theory on the occurrence of reconstruction artifacts in the context of exact helical backprojection algorithms has been provided by Hass and Faridani [94].

Beside the application of Katsevich's algorithm in standard clinical spiral CT, it has also been used for digital tomosynthesis in the area of breast imaging. The most recent developments for cone-beam mammo-computed tomography have been summarized in [95–97]. Furthermore, exact helical FBP has been applied in industrial applications [1, 23, 98], where with respect to the probing error a helical setup can be advantageous (compare Hiller et al. [32]).

3.2 Material and Methods

3.2.1 Reconstruction Algorithms

As it has been outlined above, there have been numerous improvements and implementations of Katsevich's algorithm. However, a full comparison of all the techniques used to realize and optimize

[2]The source code of the CUDA implementation can be found online at www.gpucomputing.net.

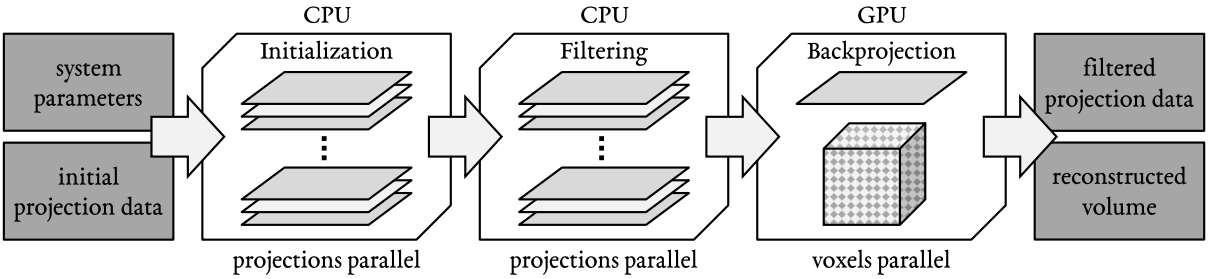


Figure 3.1: A parallel pipeline implementation scheme of Katsevich's algorithm. The acquired projection data and the geometric parameters are passed to the pipeline, projections are initialized and then filtered in parallel by the CPU. Finally, each projection is backprojected serially on the GPU using a voxel-parallel approach. The filtered projection data and the reconstructed volume is returned after pipeline completion.

the inversion formula has been yet missing and is also beyond the scope of this thesis. Although, the following sections will provide a detailed insight on how Katsevich's method can be implemented on current CPU and GPU architectures. The work presented below will serve as the basis for the experiments conducted in this and in the following chapters. Please note, that since the FDK algorithm is a standard method in CBCT, which is widely used and available through libraries and software packages[3]a detailed description will be omitted.

The implementation of Katsevich's FBP algorithm in a multi-threaded pipeline processing approach using C++ with Intel's Thread Building Blocks (TBB) library will be described in the following. The reconstruction pipeline is organized like depicted in Figure 3.1, such that in combination with the software framework presented in Chapter 2 data can be passed from MATLAB to the native layer and backwards to the MATLAB environment again. As input arguments the pipeline expects the geometric system parameters together with the acquired helical cone-beam projection data. The pipeline itself consists of mainly three processing steps, which are also called filters by the TBB library. Each of these filters can process projections sequentially or in parallel.

The first step of the pipeline initializes for each projection an individual structure that contains the corresponding acquisition data and the geometric information, i.e. projection size, index, and angle. The use of a structure that represents single projections instead of a data collection that stores all projection images in an array-like manner is preferred in the here presented pipeline implementation to enable to processing of very large projection sets which might not fit into the internal memory of the system. The use of individual projections in combination with the token-based scheduling of the TBB pipeline allows limiting the memory consumption of the whole algorithm while being able to process the work items in parallel (compare Reinders [99]). In addition to the setup of these data structures, the first filter initializes two pointers inside of each projection structure that store the memory locations of the neighboring projections, such that these links can be used during the

[3]Despite the FDK implementation inside VGStudio MAX and the one included in the MATLAB framework presented earlier, other versions that can be used in conjunction with MATLAB are available for downloaded from the MATLAB FileExchange at www.mathworks.com/matlabcentral/fileexchange or can be found within the Image Reconstruction Toolbox (IRT) at web.eecs.umich.edu/~fessler provided by Fessler (University of Michigan).

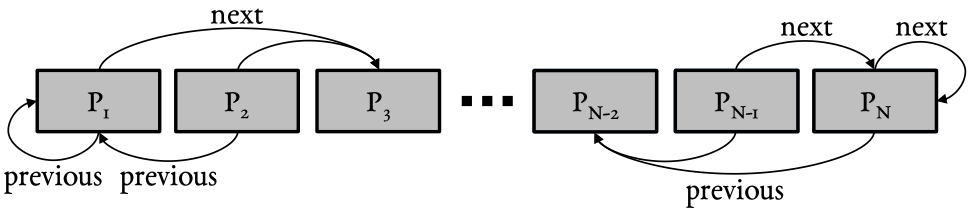


Figure 3.2: Linkage of the projections for derivative computations. Each projection stores a pointer to its neighboring projections, i.e. the previous and next one. The first and the last projection (P_1 and P_N) are linked such that their previous and next pointer link to the same projections as those of the second projection (P_2) and the second last projection (P_{N-1}), respectively.

computation of the derivatives by the second filter in the pipeline (see Section 3.2.1.3). The pointers to the previous and next projection are initialized in accordance the scheme displayed in Figure 3.2. The pointers of the first projection P_1 are set to the same references as those of the second projection P_2 and the pointers in the last projection P_N are initialized identically to the second last projection P_{N-1}. All interior projections $[P_2, ..., P_{N-1}]$ store pointers to their direct neighbors. The whole initialization step is performed by the CPU and is applied to the projections in parallel. However, as mentioned above, the number of active tokens, i.e. projections, which are initialized and processed by the pipeline concurrently, has been limited to four times the number of available threads. This ensures that for each thread running inside the pipeline sufficient projection structures are available for processing, but also limits the overall memory consumption.

After the initialization step, the second and the last filter of the pipeline perform the filtering and the backprojection step of Katsevich's algorithm, respectively. The filtering is computed on the CPU in a projection wise approach, while the backprojection is carried out sequentially for each projection structure on the GPU in a voxel-parallel implementation (see Section 3.2.1.5). Finally, the filtered projections and the reconstructed volume are returned from the pipeline.

3.2.1.1 Katsevich's Algorithm

Before the implementation of the filtration and the backprojection will be described, a review of Katsevich's algorithm [19] in accordance with the derivations in Noo et al. [27], Wunderlich [79] and Fontaine and Lee [29] will introduce the required notation. The here presented formulations and implementation details will be restricted to the flat detector geometry, although they can be easily transferred and applied to curved detector data by exploiting the relation given in equation (14) of Noo et al. [27].

Commonly, the scanning path used by the Katsevich's exact FBP is described by a helical trajectory which can be expressed as

$$y(\lambda) = \left[D_{SI} \cos(\lambda_0 + \lambda), D_{SI} \sin(\lambda_0 + \lambda), z_0 + P\frac{\lambda}{2\pi} \right], \tag{3.1}$$

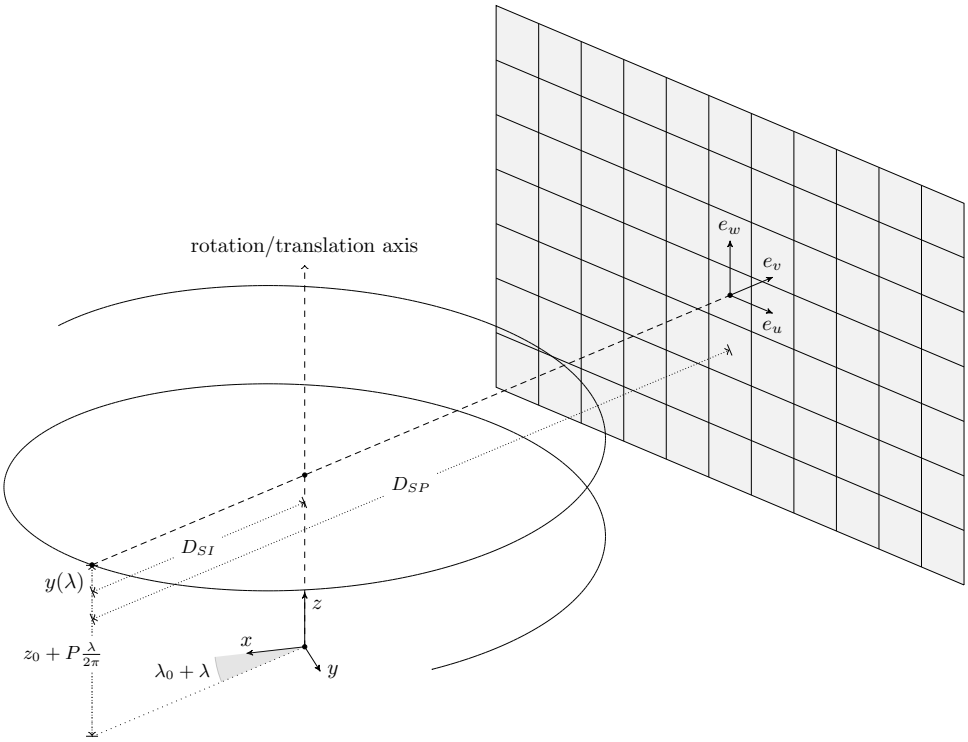


Figure 3.3: Flat detector geometry for helical cone-beam image acquisition. D_{SI} and D_{SP} are the distances from the source to the ISO-center and to the detector, respectively. The detector coordinate system is spanned by the unit vectors e_u, e_w and e_v. The helical trajectory of the X-ray source depends on the angular parameter λ and its starting point is defined by z_0 and λ_0 with respect to the z-axis and the angle, respectively. The coordinate space of the reconstructed volume is spanned by the vectors x, y and z.

where the X-ray source positions $y(\lambda)$ are parameterized by the angular variable λ. The helical pitch P defines the translation along the z-axis of the scanner during a single turn. Two additional constants, i.e. λ_0 and z_0, are used to define the starting point of the helix. Thus, the angle of the first acquired projection is defined by λ_0 and the initial position of the X-ray source along the z-direction is given by z_0. Additionally, the distance between the X-ray source and the center of rotation, i.e. the radius of the helix, is given by D_{SI} and the distance from the source to the X-ray detector is defined by D_{SP}. The parameterization of the source trajectory is depicted in Figure 3.3, which also visualizes the image acquisition with the flat detector geometry.

In general, a cone-beam projection g taken from the source position $y(\lambda)$ can be expressed by the cone-beam transform:

$$g(\lambda, u, w) = g(\lambda, \theta) = \int_0^\infty f(y(\lambda) + t\theta)dt, \tag{3.2}$$

where the vector θ points from the source position $y(\lambda)$ into the direction of the detector coordinates (u, w). As a consequence, the projections $g(\lambda, \theta(\lambda, u, w))$ are a function of (u, w) parameterized by the angle λ with

$$\theta(\lambda, u, w) = \frac{1}{\sqrt{u^2 + D_{SP}^2 + w^2}}(ue_u(\lambda) + D_{SP}e_v(\lambda) + we_w(\lambda)). \tag{3.3}$$

In equation (3.3) the unit vectors e_u, e_v and e_w span the detector coordinate system dependent on the projection angle λ, such that the center of the detector corresponds to $(u, w) = (0, 0)$ as shown in Figure 3.3.

In addition to the basic notation introduced above, the Katsevich's inversion formula relies on the concept of π-lines. A π-line is a line that connects two points on the helical source trajectory that lie not further than 360°, i.e. one helical turn, apart. A special property of π-line is that for a given point x inside the cylinder encompassed by the helical turns there exists only one π-line passing through x. Danielsson et al. [100] showed that the truncated cone-beam projections from the helical segment bounded by the endpoints of the π-line through x suffice to reconstruct $f(x)$. The extremities of the π-line are denoted with $y(\lambda_i(x))$ and $y(\lambda_o(x))$ and the interval $I_\pi(x) = [\lambda_i(x), \lambda_o(x)]$ with $\lambda_i(x) \leq \lambda_o(x)$ is called the π-segment, or π-interval, that corresponds to the π-line through x.

Now, the backprojection part of Katsevich's inversion formula for f at a point x inside the cylinder along the z-axis with radius D_{SI} can be constituted as

$$f(x) = -\frac{1}{2\pi}\int_{I_\pi(x)} \frac{1}{\|x - y(\lambda)\|} g^F\left(\lambda, \frac{x - y(\lambda)}{\|x - y(\lambda)\|}\right) d\lambda \tag{3.4}$$

where the integration uses the filtered projection data $g^F(\lambda, \cdot)$ from within the π-segment $I_\pi(x)$ as defined above.

For the computation of the filtered projections g^F the concept of κ-planes is used. A κ-plane has exactly three intersections with the helical source path, such that in terms of λ one intersection is halfway between the other two. As a consequence, for each point $y(\lambda)$ there exists a family of κ-planes which can be parameterized by an angular parameter ϕ, such that the plane $\mathcal{K}(\lambda, \phi)$ contains the points $y(\lambda)$, $y(\lambda + \phi)$ and $y(\lambda + 2\phi)$. From these three points the normal vector $n(\lambda, \phi)$ of $\mathcal{K}(\lambda, \phi)$ can be computed (see [27]).

The filtered projections $g^F(\lambda, \theta)$ can now be obtained by first taking the derivative $g'(\lambda, \theta)$ of $g(\lambda)$ with respect to λ at all directions θ (see Section 3.2.1.3). After this, the filtered data can be calculated as

$$g^F(\lambda, \theta) = \int_0^{2\pi} \frac{1}{\pi \sin\gamma} g'(\lambda, \cos\gamma\theta + \sin\gamma(\theta \times m(\lambda, \theta)))d\gamma \tag{3.5}$$

where $m(\lambda, \theta)$ is a vector normal to the κ-plane $\mathcal{K}(\lambda, \phi)$ that contains a line through $y(\lambda)$ in the direction of θ and which has the smallest absolute angle ϕ. Note that the $\times$ symbol refers to the vector cross product. As described by Noo et al. [27] the calculation of g^F can be expressed as a convolution, whereby a full proof for the underlying analytical inversion formula can be found in [13].

3.2.1.2 Discretization

In order to implement Katsevich's algorithm the equations introduced above have to be applied to the non-continuous case of discrete sampled data. Assuming that N_s projections have been measured in total and the initial angle λ_0 and the angular spacing $\Delta\lambda$ is known, then the discrete angles for the acquired projections are $\lambda_k = \lambda_0 + k\Delta\lambda$ where $k = 0, ..., N_s - 1$. Each projection has N_u columns and N_w rows with an evenly spacing Δu and Δw between the columns and rows, respectively. Therefore, the sample locations of the detector elements can be computed as $u_i = (i - (N_u - 1)/2)\Delta u$ and $w_j = (j - (N_w - 1)/2)\Delta w$ with $i = [0, ..., N_u - 1]$ and $j = [0, ..., N_w - 1]$. As a consequence, expressions of the form $g(\lambda_k, u_i, w_j)$ will be abbreviated with $g[k, i, j]$ throughout the further text where appropriate. The detector sample positions u_i and w_j were precomputed and stored into a LUTs, so that they can be efficiently accessed during the filtering process.

In addition to this, the X-ray source position along the helical path has been discretized in accordance with equation (3.1) as

$$y_k = y(\lambda_k) = \left[D_{SI} \cos(\lambda_k), D_{SI} \sin(\lambda_k), z_0 + P\frac{k\Delta\lambda}{2\pi} \right]. \tag{3.6}$$

3.2.1.3 Derivatives

The first calculation performed during the filtering phase in the pipeline involves the computation of the derivative of the cone-beam projection data at constant directions along the source trajectory. Since the derivatives $g'(\lambda, \theta)$ can be discretized in various ways, three different differentiation schemes have been implemented. The first implementation scheme $S1$ uses a simple chain rule scheme that utilizes central and forward differences similar to the 2-points difference formula described in [80]. The differentiation can be carried out as

$$\begin{aligned} g'_{S1}[k, i, j] = c_s(g[k + 1, i, j] - g[k - 1, i, j]) \\ + c_u(g[k, i + 1, j] - g[k, i, j]) \\ + c_w(g[k, i, j + 1] - g[k, i, j]) \end{aligned} \tag{3.7}$$

with the factors $c_s = 1/(2\Delta\lambda)$, $c_u = (D_{SP}^2 + u_i^2)/(D_{SP}\Delta u)$ and $c_w = u_i w_j/(D_{SP}\Delta v)$. Note that for the computation of the partial derivative along the λ-direction the central difference quotient is used, while for the directions parallel to the detector plane, i.e. the u- and w-direction, forward differences are employed.

The second differentiation scheme $S2$ was designed by replacing the 2-point central differences by a 6-point central differencing term while keeping the terms for the u- and w-direction equal to $S1$. The resulting formula for $S2$ is given by

$$
\begin{aligned}
g'_{S2}[k,i,j] = c_s(\,&(g[k+1,i,j] - g[k-1,i,j]) + \\
&(g[k+1,i+1,j] - g[k-1,i+1,j]) + \\
&(g[k+1,i,j+1] - g[k-1,i,j+1])) \\
&+ c_u(\,g[k,i+1,j] - g[k,i,j]) \\
&+ c_w(g[k,i,j+1] - g[k,i,j])
\end{aligned} \tag{3.8}
$$

with the factor $c_s = 1/(6\Delta\lambda)$, while c_u and c_w are the same as for the first differencing scheme.

The third derivative implementation g'_{S3} matches the one described by Katsevich [76] which uses a partially optimized chain rule scheme to improve the image resolution. Details on the implementation of this differentiation method have been presented in the note of Katsevich [76]. Moreover, further details and numerical evaluations of the here implemented differencing methods can be found in the publications of Noo et al. [75], Faridani et al. [77], Yu and Wang [80] and Katsevich [76].

The computations of derivatives are carried out as the first step during the filtration phase of the pipeline. As described earlier, these calculations (compare Figure 3.1), are performed in a projection parallel manner. Since each projection requires the originally acquired data from its neighboring projections ($g[k-1,\cdot,\cdot]$ and $[k+1,\cdot,\cdot]$) during the differencing step the derivatives cannot be computed in-place. As a consequence, the results of the derivative computations need to be stored in a temporary buffer. This projection buffer needs to be available for each thread that carries out a differencing operation. For this reason, a pool of projection buffers is initialized during the initialization phase of the pipeline. A projection buffer has the same size as an ordinary projection image ($N_u \times N_w$) and the number of buffers equals the maximal number of tokens which can be processed concurrently by the pipeline. References to all buffer objects are stored in a thread-safe container. Once a thread starts a derivative calculation, it fetches a buffer object from the container and removes its reference from it, such that two threads never use the same projection buffer. After a successful differencing operation, the buffer is released by the thread and its reference is inserted into the container again. This strategy reduces the runtime of the derivative computations by ensuring that the needed temporary memory is allocated only once and it does not need to be reallocated by each thread for each processed projection image.

3.2.1.4 Shift-Invariant Filtering

After the derivatives of the cone-beam projections have been computed successfully, the shift-invariant filtering is applied in accordance with equation (3.5) along κ-lines on the detector. A κ-line is defined by an angle ψ_l as the intersection between the κ-plane $\mathcal{K}(\lambda_k, \psi_l)$ and the detector plane. The lines of intersection can be described by the following function

$$
w[i,l] = w(u_i, \phi_l) = \frac{P}{2\pi D_{SI}}(\psi_l D_{SP} + \frac{\psi_l}{\tan\psi_l} u_i). \tag{3.9}
$$

where $\alpha_m = asin(R/D_{SI})$ is half the fan angle which depends on the object radius R and the helix radius D_{SI}. Given L as the number of κ-lines which are used to map the projection data onto the tilted lines, the separation between the lines can be computed as $\Delta\psi = (\pi + 2\alpha_m)/(L-1)$ and the discrete line angles are then calculated as $\psi_l = -\pi/2 - \alpha_m + l\Delta\psi$.

The rebinning of the differentiated projection data g' to the tilted κ-lines can now be carried out using linear interpolation similar to [80] by the calculation of $j_w = w[i,l]/\Delta w + (N_w - 1)/2$ and performing the following operation:

$$g_r[k,i,l] = g_r(\lambda_k, u_i, \phi_l) = c_l(t_t g'[k,i,j_t] + t_b g'[k,i,j_b]) \tag{3.10}$$

with interpolation indices $j_t = \lceil j_w \rceil$ and $j_b = \lfloor j_w \rfloor$ and with the weights $t_t = j_w - j_b$ and $t_b = j_t - j_w$. The rebinned images g_r are of the size $N_u \times L$, where the number of filtering lines L has been computed by the introduction of a factor f_r as $L = 2N_w f_r$. In accordance to [80] the rebinning factor f_r has been fixed to a value of 4.0 to ensure high precision during the rebinning step. The weighting factor c_l in equation (3.10) is a length correction term which needs to be applied because of the flat detector geometry and can be calculated as

$$c_l = \frac{D_{SP}}{D_{SP}^2 + u_i^2 + w[i,l]^2} \tag{3.11}$$

Different from other implementations of Katsevich's algorithm [27, 29, 79, 80, 90], the weighting is applied after the linear interpolation step to improve the accuracy of the length correction weighting. In addition to this, the term c_l is applied to the projection data in-place, such that no additional memory is required. In the case of the third derivative scheme $S3$ the forward height rebinning to κ-lines is slightly different (see [76]) from the here presented steps, but has also been realized using the same length correction weighting c_l after successful rebinning as described above.

Since the rebinned projection images g_r are by a factor of $2f_r$ larger than the original images g the mapping onto the tilted lines can also not be performed in-place. The temporary memory needed for the computations in the space of the κ-lines has been preallocated during the initialization phase of the pipeline by the same strategy that was used for the derivative computations, such that multiple threads can acquire image buffers from a thread-safe container and do not need to create their own buffering structures during runtime.

With the projection images rebinned to the tilted line space, a Hilbert transform is applied to each filtering line using the fast Fourier transform (FFT) (also see [79]). Therefore, the convolution kernel of the Hilbert transform has been computed in the spatial domain with sample positions located at $q_i = -(N_u - 1) + i - 0.5$ with $i = [0, ..., 2(N_u - 1) + 1]$ using the following formula:

$$h_H[i] = \frac{1 - cos(\pi(q_i))}{\pi(q_i)}. \tag{3.12}$$

Note that the offset of the sample positions q_i by half a detector cell has been introduced to compensate the shift of from derivative step and to eliminate ringing artifacts (compare [77, 94]). The

Hilbert kernel h_H has then been transformed to the Fourier domain and is then applied to the filtering lines by element-wise multiplication ($\star$). This operation can be expressed as

$$\begin{aligned} g_r[k,i,l] &= \mathcal{F}_i^{-1}(\mathcal{F}_i(g_r[k,i,l]) \star \mathcal{F}_i(h_H[i])) \\ &\forall\, k \in [0,...,N_s-1] \;\wedge\; l \in [0,...,L-1] \end{aligned} \tag{3.13}$$

where $\mathcal{F}_i$ and $\mathcal{F}_i^{-1}$ are the Fourier transform and the inverse Fourier transform along the rebinned detector rows, respectively. The resulting filtered projection data is stored in-place within g_r after the Hilbert transform has been calculated, such that no additional memory is consumed. The implementation of the Hilbert transform step uses the Intel Performance Primitives (IPP) library to exploit SIMD instructions of the CPU.

The final calculation during the filtering phase of the pipeline is a backward height rebinning that maps the κ-line data back to the detector rows. For this purpose, at each detector coordinate (u_i, w_j) an index l has been determined such that

$$\begin{aligned} w[i,l-1] \le w_j \le w[i,l] \qquad &\text{for } u_i < 0 \\ w[i,l] \le w_j \le w[i,l+1] \;&\text{for } u_i \ge 0. \end{aligned}$$

After this, linear interpolation coordinates and coefficients can be calculated as

$$\begin{aligned} &l_t = l, \quad &&l_b = l-1, \quad &&t_t = (w_j - w[i,l-1])/(w[i,l] - w[i,l-1]) \quad &&\text{for } u_i < 0 \\ &l_t = l+1, \quad &&l_b = l, \quad &&t_t = (w_j - w[i,l])/(w[i,l+1] - w[i,l]) \quad &&\text{for } u_i \ge 0 \end{aligned}$$

and $t_b = 1 - t_t \;\forall\, u_i$. These values are then used to remap the κ-lines to the detector rows by

$$g^F[k,i,j] = t_b g_r(k,i,l_b) + t_t g_r(k,i,l_t). \tag{3.14}$$

For reasons of runtime performance, this last step is carried out such that the filtered projections g^F are stored at the location of the original projection data g. By this means, the original data is overridden since it is no longer needed by the algorithm and no additional memory needs to be allocated.

One important property of the κ-lines described by $w[i,l]$ is that they are independent of the projection angle λ_k. As a consequence, the coefficients $w[i,l]$ are equal for all projections and can be precomputed and stored in a LUT before the projection processing starts. The coefficients $w[i,l]$ also have been used to precompute the linear interpolation coefficients for the forward and the backward rebinning step as described above. The interpolation weights are stored into two separate LUTs and used to speed up the rebinning process even further.

3.2.1.5 Backprojection

With the filtered data $g^F(\lambda, u, w)$ the volume f can be reconstructed according to equation (3.4). For a given voxel at location $x = (x^\star, y^\star, z^\star)$ its projected detector coordinates can be calculated by

$$v^\star(\lambda, x) = D_{SI} - x^\star cos(\lambda_0 + \lambda) - y^\star sin(\lambda_0 + \lambda) \tag{3.15}$$

$$u^\star(\lambda, x) = \frac{D_{SP}}{v^\star(\lambda, x)}(-x^\star sin(\lambda_0 + \lambda) + y^\star cos(\lambda_0 + \lambda)) \tag{3.16}$$

$$w^\star(\lambda, x) = \frac{D_{SP}}{v^\star(\lambda, x)}(z^\star - z_0 - \frac{P}{2\pi}\lambda). \tag{3.17}$$

The reconstruction of f at point x can then be performed with the formula

$$f(x) = \frac{1}{2\pi}\sum_{k=k_i}^{k_o} \frac{g^F(\lambda_k, u^\star(\lambda_k, x), w^\star(\lambda_k, x))}{v^\star(\lambda_k, x)}. \tag{3.18}$$

where the indices k_i and k_o correspond to the lower and upper bound of the π-interval $I_\pi(x)$ such that $\lambda_i(x) \leq \lambda_{k_i} < \lambda_{k_o} \leq \lambda_o(x)$. To access the projection memory the coordinates $(u^\star, w^\star)$ need to be converted to the corresponding raster indices. For this reason, two interpolation approaches are provided to compute the projection samples: (1) nearest neighbor [79] and (2) bi-linear interpolation [29].

A straightforward approach to compute equation (3.18) is to first determine the interval I_π at point x and then backprojection the projections which fall into the range of $I_\pi(x)$. This technique is called the π-interval method [79]. In its original version this approach iterates over all projections $g^F[k, i, j]$ to reconstruct a single voxel x. To use the π-method in combination with the here presented pipeline-based processing architecture, where not all projections reside in the internal memory at the same time, the computation of the π-intervals have to be performed multiple times for the same point x. For this reason, the intervals $I_\pi(x)$ for all voxels inside the volume of interest are precomputed and buffered before the first projection is backprojected. The extremities λ_i and λ_o of the π-interval of $x = (x^\star, y^\star, z^\star)$ have been determined by finding the root of the function

$$F_\pi(\lambda_i) = \frac{P}{2\pi}\left[\left(\pi - 2atan\left(\frac{\beta_\pi(\lambda_i)}{\alpha_\pi(\lambda_i)}\right)\right)\left(1 + \frac{r^2 - D_{SI}^2}{2D_{SI}\alpha_\pi(\lambda_i)}\right) + \lambda_i\right] - z^\star \tag{3.19}$$

with $r = \sqrt{x^{\star 2} + y^{\star 2}}$, $\gamma_\pi = atan2(y^\star, x^\star)$, $\alpha_\pi(\lambda_i) = D_{SI} - rcos(\gamma_\pi - \lambda_i)$ and $\beta_\pi(\lambda_i) = rsin(\gamma_\pi - \lambda_i)$. Brent's method [101] has been employed to compute the root λ_i of F_π from an initial bracketing interval $[\frac{2\pi}{P}z^\star - \pi, \frac{2\pi}{P}z^\star]$. Then the upper boundary of the π-interval λ_o has been determined by

$$\lambda_o = \lambda_i + \pi - 2atan\left(\frac{\beta_\pi(\lambda_i)}{\alpha_\pi(\lambda_i)}\right) \tag{3.20}$$

where r, γ_π, $\alpha_\pi(\lambda_i)$ and $\beta_\pi(\lambda_i)$ are defined as above. A full proof for this approach can be found in Wunderlich [79]. The resulting π-intervals are stored in a voxel-dependent LUT and are reused during the backprojection operation.

To improve the final quality of the reconstruction it has been proposed in several publications [27, 29, 79, 80] to include the projections that lie slightly outside the π-interval into the backprojection by using a weighting scheme. In the here presented implementation the weighting $\rho(\lambda, x)$ of the π-interval endpoints has been computed similar to Noo et al. [27] and according to the following equations:

$$\rho(\lambda, x) = \begin{cases} 0 & \text{if } \lambda \leq \lambda_i(x) - \Delta\lambda \\ (1 + d_i)^2/2 & \text{if } \lambda_i(x) - \Delta\lambda < \lambda \leq \lambda_i(x) \\ 1/2 + d_i - d_i^2/2 & \text{if } \lambda_i(x) < \lambda \leq \lambda_i(x) + \Delta\lambda \\ 1 & \text{if } \lambda_i(x) + \Delta\lambda < \lambda \leq \lambda_o(x) - \Delta\lambda \\ 1/2 + d_o - d_o^2/2 & \text{if } \lambda_o(x) - \Delta\lambda < \lambda \leq \lambda_o(x) \\ (1 + d_o)^2/2 & \text{if } \lambda_o(x) < \lambda \leq \lambda_o(x) + \Delta\lambda \\ 0 & \text{if } \lambda > \lambda_o(x) + \Delta\lambda \end{cases} \tag{3.21}$$

where $d_i = (\lambda - \lambda_i(x))/\Delta\lambda$ and $d_o = (\lambda_o(x) - \lambda)/\Delta\lambda$. With the introduction of the endpoint weighting function ρ to the π-interval method the final contribution of each projection to $f(x)$ is then calculated as

$$\delta_\pi f(x) = \frac{\Delta\lambda}{2\pi} \frac{\rho(\lambda, x)}{v^\star(\lambda, x)} g^F(\lambda, u^\star(\lambda, x), w^\star(\lambda, x)). \tag{3.22}$$

Another approach to compute the projection contributions is the CBC method developed by Yang et al. [62]. This approach uses the Tam-Danielsson (TD) window [102] which defines the region of the detector which is required for an exact reconstruction. The top border of the TD-window can be computed for each discrete detector column u_i by the evaluation of

$$w_t(u) = \frac{P}{2\pi} \frac{D_{SP}}{D_{SI}} \frac{s_t(u)}{(1 - cos(s_t(u)))} \tag{3.23}$$

with $s_t(u) = 2\pi + 2atan(D_{SP}/u)$ for $u < 0$ and $s_t(u) = 2atan(D_{SP}/u)$ otherwise. Moreover, the bottom border of the TD-window can be calculated as

$$w_b(u) = \frac{P}{2\pi} \frac{D_{SP}}{D_{SI}} \frac{s_b(u)}{(1 - cos(s_b(u)))} \tag{3.24}$$

with $s_b(u) = 2atan(D_{SP}/u)$ for $u < 0$ and $s_b(u) = -2\pi + 2atan(D_{SP}/u)$ otherwise. Now, a point x belongs to the CBC of a projection, if its projected coordinates $(u^\star, w^\star)$ lie inside the borders of the TD-window. Additionally, Yang et al. [62] have proven that a projection belongs to the π-interval $I_\pi(x)$ if and only if the point x is contained in the CBC of the projection. Therefore, the

CBC can be used to reformulate the fractional contribution of the backprojection operation in equation (3.22) by replacing the weighting function ρ with

$$\chi(\lambda, x) = \chi(u^\star, w^\star) = \begin{cases} 0 & \text{if } w^\star > w_t(u^\star) + w_a \\ \frac{w_t(u^\star)+w_a-w^\star}{2w_a} & \text{if } w_t(u^\star) - w_a < w^\star \leq w_t(u^\star) + w_a \\ 1 & \text{if } w_b(u^\star) + w_a < w^\star \leq w_t(u^\star) - w_a \\ \frac{w^\star-w_b(u^\star)+w_a}{2w_a} & \text{if } w_b(u^\star) - w_a < w^\star \leq w_b(u^\star) + w_a \\ 0 & \text{if } w^\star \leq w_b(u^\star) - w_a \end{cases} \tag{3.25}$$

where $w_a = \xi \Delta w$ and ξ is a parameter used to control the amount and the smooth weighting of projection information from the area slightly outside the TD-window. For the studies performed within the later sections and chapters of this thesis the parameter ξ has been fixed to a value of 0.5, such that the information used from outside the border of the TD-window lies within half a detector element height.

The resulting fractional contribution of each projection to the reconstructed image f at the point x in accordance with the CBC method can be described similarly to equation (3.22) by

$$\delta_{cbc} f(x) = \frac{\Delta\lambda}{2\pi} \frac{\chi(\lambda, x)}{v^\star(\lambda, x)} g^F(\lambda, u^\star(\lambda, x), w^\star(\lambda, x)). \tag{3.26}$$

Note that the weighting functions χ and ρ and thus the CBC and the π-interval method will not produce exactly the same reconstruction results, since the weights are calculated with respect to different dimensions. On the one hand, the function χ is defined piecewisely along the w-direction of the detector to smoothly weight the borders of the TD-window. On the other hand, ρ has been designed to weight the endpoints of the π-segment in relation to the projection angle λ. However, it has been shown that both backprojection methods can produce accurate results, whereby the CBC method can be used to significantly improve the performance of Katsevich's algorithm when realized on parallel hardware [28, 29, 62]. The main reason for this is that the weighting function χ does not require any information about the π-segments. As a result, the time consumed by the numerical solution of equation (3.19) and equation (3.20) in the case of the π-interval method can be saved when employing the CBC method.

In addition to this, the TD-window is equal for all projection images, so that its upper and lower border defined by $w_t(u)$ and $w_b(u)$, respectively, can be precomputed. As a consequence, in there here presented implementation both boundaries have been calculated for each discrete detector column u_i and stored in the LUTs $w_t[i]$ and $w_b[i]$ index by the column index i before the backprojection process starts. When the nearest neighbor interpolation is chosen for the backprojection operation the projected coordinate $u^\star(\lambda, x)$ is rounded to the nearest sample position $u_{i^\star}$, such that the LUT-values $w_t[i^\star]$ and $w_b[i^\star]$ can be used directly within equation (3.25). In contrast to this, if a bi-linear interpolation strategy is used to compute the projection sample at the coordinates $(u^\star, w^\star)$, the border of

the TD-window is calculated by linear interpolation from the LUTs values. The linear interpolation weights are calculated from the LUT-values corresponding to the two detector columns adjacent to the projected coordinate $u^\star(\lambda, x)$ similar to [29].

Two additional weighting schemes, ρ' and χ' have been implemented during the course of this thesis. They can be seen as simplifications of the π-line weighting performed by ρ and of the CBC weighting scheme defined by χ, respectively, in the sense that both schemes forego the smooth weighting of the projection data at the boundaries. The weighting functions can be defined by

$$\rho'(\lambda, x) = \begin{cases} 0 & \text{if } \lambda < \lambda_i(x) \\ 1 & \text{if } \lambda_i(x) \leq \lambda \leq \lambda_o(x) \\ 0 & \text{if } \lambda_o(x) < \lambda \end{cases} \tag{3.27}$$

and by

$$\chi'(\lambda, x) = \chi'(u^\star, w^\star) = \begin{cases} 0 & \text{if } w^\star > w_t(u^\star) \\ 1 & \text{if } w_b(u^\star) \leq w^\star \leq w_t(u^\star) \\ 0 & \text{if } w^\star < w_b(u^\star) \end{cases}. \tag{3.28}$$

As mentioned above, the backprojection part of Katsevich's algorithm has been realized on the CPU and on the GPU using NIVDIA's CUDA [103]. Consequently, the four weighting functions ρ, ρ', χ and χ' have been realized such that they can be executed on both architectures. Additionally, the nearest neighbor and the bi-linear interpolation methods are available on the CPU and on the GPU. In accordance with equation (3.22) and with equation (3.26) the π-interval method and the CBC method have also been implemented as host (CPU) and device (GPU) code. However, for large volumes of interest the π-interval method can run out of memory, since the projection data is processed sequentially by the pipeline and, as a consequence, the π-intervals need to be precomputed and stored for each voxel. For this reason, the CBC method can be implemented more efficiently with respect to memory consumption on the GPU than the π-line-based method.

3.2.1.6 Slice Allocation on the GPU

For the backprojection of the χ-weighted projection data from the TD-window onto its corresponding CBC a single projection image and voxels of the reconstructed region of interest need to reside in the memory of the GPU. For extremely large[4]volume datasets, as they often occur in the industrial area, it can happen that the not all slices of the reconstructed volume fit into the global memory of the graphics device. Therefore, a novel slice allocation strategy has been developed which can handle volume sizes that exceed the memory capability of the GPU.

[4]An industrial dataset consisting of a 4K volume with 4096^3 each represented by $32 - bit$ floating point number requiring 4 bytes needs a final storage space of about 275 GB.

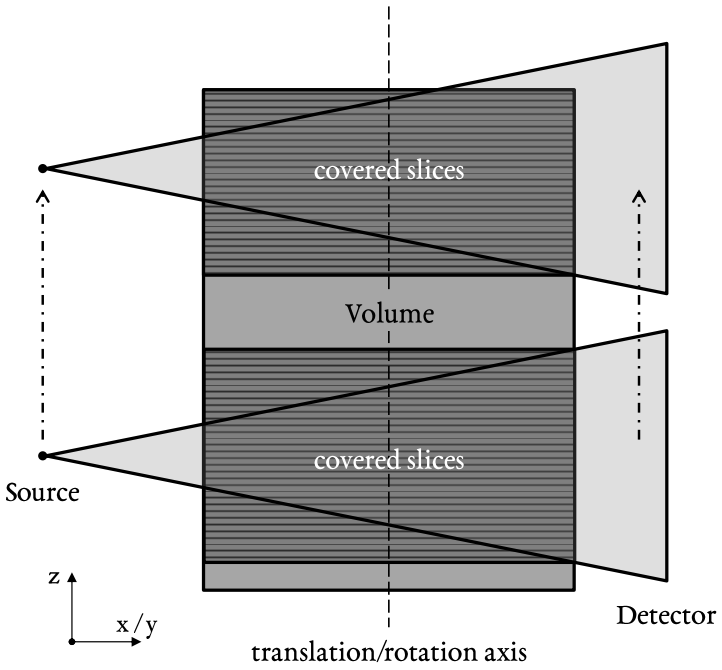


Figure 3.4: Slice allocation strategy used for GPU reconstruction of large datasets with Katsevich's algorithm. It is assured that all slices that lie inside the CBC are of the currently processed projection are allocated while the cone beam slides along the volume. Slices that exit the CBC from one to processing of the next projection are freed and copied back to their corresponding host memory location, while slices that enter the CBC are allocated by a pool-based strategy.

To realize this strategy, the volume is sliced up along the direction of the rotation axis into N_z individual slices (compare Figure 3.4). During the initialization of the GPU device an array with pointers to the volume slices is allocated whereby each pointer is initially set to $null$. This array of pointers is then checked before every projection is processed by the backprojector, such that the slices covered by the TD-window of the currently handled projection image are copied from the host memory to the global memory of the graphics device. In addition to this, slices that are no longer involved in the backprojection process are copied back to the host and freed from the GPU memory. Consequently, as the cone-beam and thus CBC corresponding to the acquired projections move along the volume of interest along the direction of the rotation axis, new slices need to be allocated on the GPU at the top of the CBC, while at the same time, slices at the bottom of the CBC are freed. This is why a pool-based allocation mechanism has been used, such that the reallocation of slice memory is carried out efficiently. The aforementioned slice management strategy is summarized and visualized in Figure 3.4.

The range of the slices that are contained in the CBC of the current projection along z-direction can be computed in the following way:

Table 3.1: Technical specifications of the employed CT scanner

CT scanner: v|tome|x s 225		
Manufacturer		Phoenix X-ray (GE Sensing & Inspection Technologies)
X-ray tube		Microfocus
Max. tube voltage		225 kV
Max. output		320 W
Detector		Perkin Elmer 840
Detector area		$204.8 \times 204.8\,\mathrm{mm}^2$
Pixel count	$(N_u \times N_w)$	512×512
Pixel Pitch	$(\Delta u = \Delta w)$	400 µm
Dist. Source-Detector	(D_{SP})	783.4191 µm
Dist. Source-Object	(D_{SP})	68.5492 µm

$$z_t = z_{src} + \frac{D_{SI} + R_{xy}}{D_{SP}} w_{t,max} \tag{3.29}$$

$$z_b = z_{src} + \frac{D_{SI} + R_{xy}}{D_{SP}} w_{b,min} \tag{3.30}$$

where z_t and z_b are the upper and lower boundary of the slice range, respectively. The X-ray source position along the z-axis is given by z_{src} and R_{xy} is the radius of the cylinder whose axis is parallel to the rotation axis and which intersects the corners of the volume slices. The maximal and minimal values of w_t and w_b with respect to the acquired projection images in accordance with equation (3.23) and equation (3.24) are given by $w_{t,max}$ and $w_{b,min}$, respectively. Furthermore, the calculated z-range $[z_t, z_b]$ can be converted to slice indices by

$$l_t = \lceil (z_t - z_{min})/\Delta z \rceil \tag{3.31}$$

$$l_b = \lfloor (z_b - z_{min})/\Delta z \rfloor \tag{3.32}$$

with z_{min} being z-coordinate of the lower corner of the volume bounding box and Δz defining the sampling distance between the volume slices along the z-axis. The indices l_t and l_b can then be used directly to check the integrity of the slice allocation as described above.

After it has been asserted that all covered slices reside in the memory of the graphics device, each projection is backprojected using a voxel driven and voxel parallel approach similar to the one used by Benquassmi et al. [30] to exploit the maximal capabilities of the GPU.

3.2.2 Projection Acquisition and Simulation

For the comparison of the FDK algorithm with Katsevich's method, as it will be presented in Section 3.3, a solid cubical object made from titan alloy has been used. The cube has been manufactured

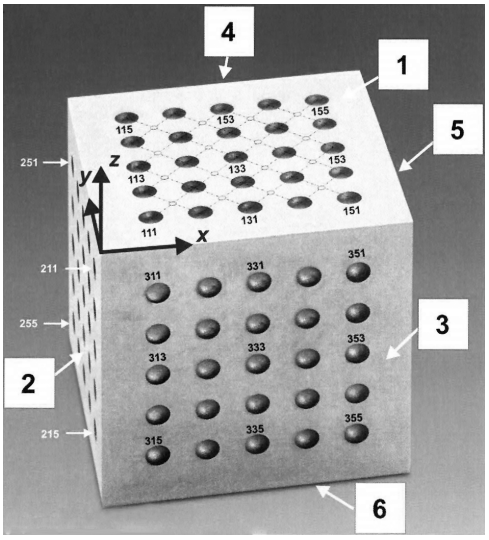


Figure 3.5: Numbering of the side planes and the caps of the KKW in accordance to the DKD calibration certificate. The numbers of the side planes are given in the boxes, while the numbers of the calottes are displayed close to their positions. Note that only the planes 1, 2 and 3 contain 25 spherical caps. Plane 5 contains a single calotte at its center which is solely used to determine the orientation of the cube. The image has been taken from the DKD calibration certificate (Calibration mark: 7455 DKD-K-25901 2007-7).

Table 3.2: Four parameter settings defining scanning and reconstruction geometries

Trajectory	**Number of Projections**	**Angular Range** $[°]$	**Pitch** $[\mathrm{mm}]$
Circular	2000	360.0	0.0
Helical	2000	720.0	12.5
Helical	1000	360.0	25.0
Circular	1000	360.0	0.0
Volume size	$512 \times 512 \times 512$		
Voxel size	$35\ \mu m$		

by FEINMESS GmbH & Co. KG[5]and has a size of $10\,\mathrm{mm} \times 10\,\mathrm{mm} \times 10\,\mathrm{mm}$. Each of three sides of the cube has 25 equidistantly spaced spherical caps, distributed over a 5×5 grid. Each cap has a radius of $0.4000 \pm 0.0008\,\mathrm{mm}$ and is drilled $0.4\,\mathrm{mm}$ deep into the cube. The precise edge lengths of the cube together with the positions and the sizes of each spherical cap have been calibrated by DKD using tactical measurements. Since the caps of the cube are also called calottes, this type of cube is also known as a calotte cube or KKW. The image displayed by Figure 3.5 illustrates the shape of the KKW, while additionally presenting the numbering of its side planes and calottes.

From the calibrated cube described above, real projection data has been acquired at the TPW Prüfzentrum in Neuss, Germany, using a phoenix X-ray v|tome|x s 225 industrial CBCT scanner. The v|tome|x s 225 is a high-resolution system which can be set up to acquire CT scans from various X-ray source trajectories. Table 3.1 presents the technical specifications of this CT machine together

[5]Additional details on FEINMESS GmbH & Co. KG and on the calotte cube (Kugelkalottenwürfel) (KKW) can be found online under www.feinmess.com.

with its employed geometrical configuration. For the studies carried out in this thesis the CT scanner was used to acquire three sets of projections. All the three scans use the detector geometry and the distances, i.e. source-to-object and source-to-detector, given in Table 3.1. The first projection set of the KKW was generated from a scan on a circular trajectory, where 2000 images have been captured over an angular range of 360°. During the next scan the KKW was imaged 2000 times using a helical scanning path with an angular range of 720°. The third scan only acquired 1000 projections over 360° in a single helical turn. Of both helical scans the pitch P was set to 25.0 mm, so that the movement of the detector relative to the scanned cube was equal for both acquisitions. Finally and as displayed in Table 3.2, a fourth dataset has been generated from the first scan by leaving out every second projection. As a result, the fourth projection sets contains 1000 images from an angular range of 360°, whereby the angular spacing between the projections is twice as large as for the first projection set. In total, this leads to four different acquisition schemes, two circular and two helical ones, for which Table 3.2 summarizes the scanning configurations.

Additionally to the real projection data acquired from the phoenix X-ray system, a digital reconstructed radiograph (DRR) algorithm was used to generate simulated projection data. This dataset was then used to verify the correctness of both implemented reconstruction methods and to demonstrate the theoretical optimum of the FDK algorithm in comparison to Katsevich's approach. The simulations of the projections have been calculated from a voxelized KKW model with a resolution of 512^3 voxels which has been generated from a polygonal mesh within VGStudio MAX. For the forward projection of the KKW volume the cone-beam transform given by equation (3.2) has been discretized and implemented on the GPU using a sampling based ray casting approach, following the derivations presented by Weinlich et al. [20]. The implementation is based on the software framework presented in Chapter 2 and is realized in native code with NVIDIA's CUDA Toolkit [103] in such a way that the geometric system parameters for the simulation are passed from MATLAB to the MEX layer and then to the CUDA routines. Finally, the generated projections are returned to MATLAB (compare Figure 2.2). The forward projection operation processes the projections sequentially and parallelizes the computations per detector pixel, i.e. per ray, in accordance with the following steps (compare with [20]):

1. **Initialization**: For each discrete detector element at (u_i, w_i) generate a ray that points from the X-ray source position $y(\lambda_k)$ into the direction $\theta(\lambda_k, u_i, w_i)$.

2. **Intersection**: Compute the intersection of each ray with the volume bounding box.

3. **Integration**: Sample the volume along each ray between the intersection points and accumulate the samples.

where k is the index of the currently processed projection, as introduced in Section 3.2.1.1.

All calculations during the projection simulation are carried out on the GPU in floating point precision. To improve the accuracy of the simulated projection sets the sampling distance along the ray has been set twice as high as the maximal edge length of the voxels. Additionally, the higher order

Listing 3.1: Ray casting using higher order intergation and Kahan summation

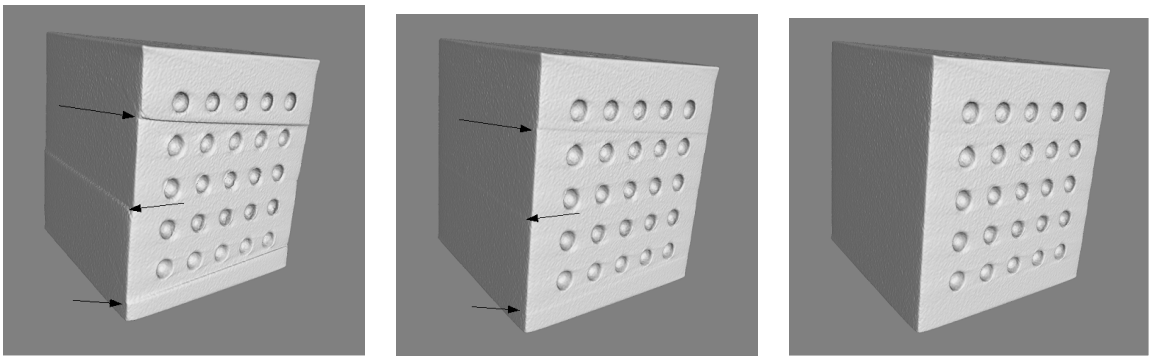
```
float integrateRay(Texture3f tex, ray3& ray, float tnear, float tfar)
{
  float t_step = cSamplingDistance;
  int ti = 0;
  float t_curr = tnear;

  // march along ray from front to back
  float sum1 = 0.0f;
  float sum2 = 0.0f;
  float c1 = 0.0f; // for error correction of sum1
  float c2 = 0.0f; // for error correction of sum2
  while (t_curr <= tfar)
  {
    // compute positions along the ray ...
          t_curr      = tnear + ti * t_step;
    float  t_next     = tnear + (ti + 1) * t_step;
    float3 rayPos_curr = ray.o + ray.d * t_curr + 0.5f;
    float3 rayPos_next = ray.o + ray.d * t_next + 0.5f;
    float3 rayPos_half = 0.5f * (rayPos_next + rayPos_curr);

    // ... sample the volume and accumulate the samples
    float y1 = tex3D(tex, rayPos_curr) - c1;
    float t1 = sum1 + y1;
    c1   = t1 - sum1 - y1;
    sum1 = t1;

    float y2 = tex3D(tex, rayPos_half) - c2;
    float t2 = sum2 + y2;
    c2   = t2 - sum2 - y2;
    sum2 = t2;

    ++ti;
  }

  // combine results using higher order integration
  return (sum1 + 2.0f * sum2) / 3.0f;
}
```

integration technique described by de Boer et al. [104] in combination with the Kathan summation algorithm [105] has been used to reduce the numerical error made during the sample accumulation process. The CUDA source code for integration along a single ray is presented in Listing 3.1.

3.2.3 Reconstruction with Detector Offset

The reconstructions from the simulated and the real projections have computed at a resolution of $512 \times 512 \times 512$ voxels with a sampling distance of $35\,\mu$m along the x-, y- and z-axis. As mentioned above, the projection sets acquired from the circular trajectory were reconstructed using the FDK method. During the filtration step of the FDK algorithm the Shepp-Logan filter [106] was applied to the projections. Unlike the circular datasets, the projections acquired from the helical source trajectory were reconstructed using the implementation of Katsevich's theoretical exact filtered backprojection algorithm as described above. For the computation of the derivatives of the cone-beam projections the scheme $S3$ was used (see Section 3.2.1.3) and the CBC method with the χ-weighting was chosen over the π-interval backprojection for performance reasons (compare Section 3.2.1.5). All backprojection steps were carried out on the GPU using bi-linear interpolation and the slice management strategy described in Section 3.2.1.6.

The initial reconstructions from the real projection data contained misalignment artifacts due to a horizontal shift of the detector panel during the acquisition phase. By the use of the geometric



(a) (b) (c)

Figure 3.6: Comparison of different choices for the horizontal detector offset during helical reconstruction from real data. A choice of $0.0\,mm$ (a) and of $-0.6\,mm$ as in the circular case (b) introduced step artifacts in the reconstructed volume. Manually tuning the parameter for the detector offset to $-0.76\,mm$ yielded the best results (c).

calibration capabilities of VGStudio MAX, the detector offset along the u-axis has been estimated from the projection sinograms for the circular data. The resulting horizontal shift of -0.6063 mm removed the geometry-induced artifacts from the reconstructed images. As for the circular scans, misalignment artifacts were also visible for the helical reconstructions, such that a horizontal detector shift had to be determined for this dataset as well. Although the circular and helical scans were carried out on the same the CT scanner with basically the same geometric configurations, the use of the offset determined in the circular case did not yield a reasonable result for the helical datasets. Therefore, the horizontal detector shift for the helical projections has been tuned heuristically by manual adjustments. Therefore, reconstructions of the helical projection were computed with various detector offsets to select the optimal shift. Figure 3.6 visualizes the reconstructed images of the helical scans from 2000 projections with different values for horizontal detector offset. Setting the horizontal shift of the detector to 0.0 mm and to -0.6063 mm, as determined for the circular reconstructions, resulted in step artifacts after the application of Katsevich's algorithm, like indicated by the arrows in Figure 3.6a and Figure 3.6b, respectively. A manually determined value of -0.76 mm for the detector shift yielded the best results for both helical scans (Figure 3.6c). Therefore, this value will be used during the evaluation throughout the following sections. Note that Chapter 4 will introduce a novel geometric calibration method which allows a fully automatic tuning of system parameters to compensate geometric misalignments of the CT scanner. Moreover, all simulated projection images the circular and helical setups were generated and reconstructed without any detector offsets.

3.2.4 Evaluation Methods

The comparison of the FDK algorithms with Katsevich's method was performed on the basis of noise statistics, visual inspection and shape measurements. After successful reconstruction of each of the four projection sets, the resulting volumes were normalized to the interval $[0, 1]$ to scale their gray

values to a comparable range. To analyze the noise characteristics of the computed images the signal-to-noise ratio (SNR) has been calculated using the following formula:

$$SNR(f) = \frac{\mu_{SIG}(f)}{\sigma_{BG}(f)} \tag{3.33}$$

where f in this context represents a vectorized version of reconstructed volume, μ_{SIG} is the mean gray value intensity of f and σ_{BG} is the standard deviation of the background signal which has been selected as the air in the inspected volumes. In addition to the SNR, the overall image quality of the resulting reconstructions has been compared by visual inspection using slice images and profile plots. Finally, the exactness of both reconstruction algorithms in terms of measurement uncertainty has been determined using the NDT-capabilities of VGStudio MAX. Therefore, a predefined measurement template of the KKW has been used. The template contained all nominal sizes and positions of all features, i.e. side planes and calottes, of the cube as they had been determined by the DKD. With this measurement template it was possible to apply the shape analysis equally to all reconstructions. To perform the measurements inside VGStudio MAX the following steps have been conducted for each dataset:

1. Reconstruction of the KKW using the FDK algorithm for the circular datasets and the exact FBP of Katsevich for projection data from helical scanning trajectories.

2. An "Advanced Surface Determination" has been computed for each resulting volume to define the surface of the calotte cube, such that the registration functionalities of VGStudio MAX become available for each dataset.

3. The predefined measurement template of the KKW was copied into the scene of interest and was fitted by hand against the reconstructed KKW. As a result, the boundaries of the template and the volume were roughly aligned.

4. The manual alignment was improved by copying and pasting the template under the volume in the scene tree. By this process VGStudio MAX automatically fits the reference objects, i.e. spheres, of the measurement template to the spherical caps of the reconstructed KKW, such that the template and volume become more precisely aligned.

5. To improve the precision of the alignment between the KKW and the template even further, the fitted reference objects of the volume were registered with the reference objects of the original imported measurement template.

6. By the deleting the fitted measurements under the volume scene node and repeating steps 4 and 5 the accuracy of the fitting between the reference objects was improved even more.

7. Finally, the resolution of the reconstructed volume was adjusted, so that the actual lengths of the cube fitted their nominal values in accordance with the information provided by the DKD

Table 3.3: Gray value statistics and SNR of the real datasets

Traj.	Proj.	μ_{SIG}	σ_{BG}	SNR
Circ.	1000	0.8161	0.0490	16.7
Circ.	2000	0.8312	0.0495	16.8
Hel.	1000	0.7875	0.0335	23.5
Hel.	2000	0.8075	0.0346	23.3

calibration certificate of the KKW. For the precise adjustment of the resolution nine features within the measurement template haven been selected. These features represent distances between eighteen spheres. For all features the quotient between their actual and nominal values has been calculated and averaged to yield the scaling factor for the correction of the volume resolution. This adjustment was repeated three times in an iterative process to gain a precisely scaled object of interest that matches the nominal size of the KKW.

After all these steps had been carried out successfully, the reference objects fitted to the reconstructed calotte cube was used in combination with the measurement template to calculate the actual center position and the actual surface form of each spherical cap. The positions were then used to calculate all distances between all reference objects followed by a comparison of their actual and nominal values. The surface form is a measure for the roughness of the reconstructed spherical caps. It has been computed directly within VGStudio MAX and can be defined in accordance with Smith [107] by

$$Z_t = Z_p - Z_v = \max_i Z_i - \min_i Z_i. \tag{3.34}$$

where the values Z_i are roughness measurements that represent the distance of the actual reconstructed spherical caps to the fitted reference spheres of the template. The peak height Z_p and valley depth Z_v are computed from these surface measurements by picking out the largest and smallest values, respectively. Consequently, the surface form Z_t quantifies the amount of distortion introduced by the employed scanning trajectory in combination with the used reconstruction approach.

3.3 Results

3.3.1 Noise Statistics and Visual Inspection

Table 3.3 gives a comparison of the noise characteristics of the two reconstruction techniques for the different projection sets. From the last column of the table it can be seen that for the KKW datasets Katsevich's FBP reconstruction algorithm has a higher SNR than the FDK method. For the FDK reconstructions the datasets containing 2000 projections resulted in a slightly higher SNR than the one with 1000 projection images. In contrast to this, the exact helical reconstruction algorithm delivered an SNR of about 23, where a small increase could be noticed for the lower projection count

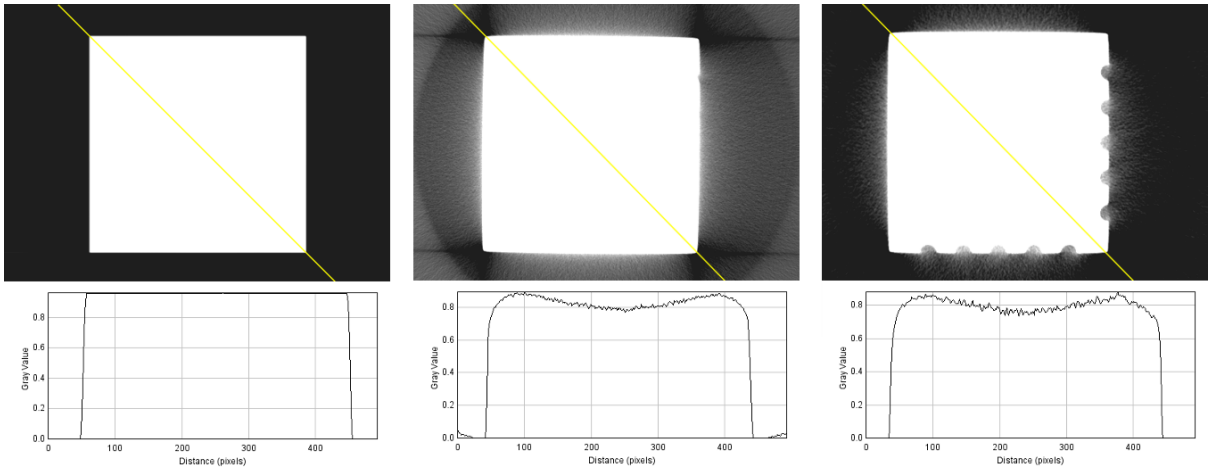
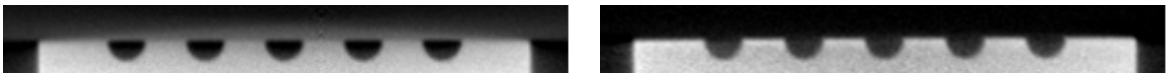


Figure 3.7: Central axial slices of the reconstructed calotte cube (window = 0.4, level = 0.15). The left image shows the reconstruction from the simulated helical projections. The middle image and the right image show the cube reconstructed from real data from the circular and the helical trajectory, respectively. The profiles below the slices show the gray value profiles along the yellow line in the slice images.

from 23.3 to 23.5. In addition to these facts, the gray value statistics for the circular cases show that with an increase of the number of projections the mean signal intensity μ_{SIG} raises together with the standard deviation of the background signal σ_{BG}. The same effect can be observed for the helical reconstructions. However, the mean gray value intensity resulting from Katsevich's algorithm was never large than the one of the FDK reconstructions, whereby the background variation follows the same relation. In summary, the SNR does not change significantly with the number of projections, while Katsevich's outperforms the FDK method with respect to the studied noise properties.

The observations made above become clearer with a visual inspection of the volumes reconstructed from the real projection data in comparison to the results computed from the simulated projections as displayed in Figure 3.7. The figure displays the central axial slices of reconstruction results of the KKW form the simulated helical, from the real circular and from the real helical projections from left to right. Hereby, the leftmost image of Figure 3.7 demonstrates how accurate Katsevich's algorithms work on the simulated projection data. The profile curve below the slice image is flat at the interior part of the calotte cube, the edges are sharp and show only little smoothing and the background is constantly zero without any noise. In contrast to this, the reconstruction from the FDK algorithm in the center of Figure 3.7 shows a slight cupping artifact towards the inside of the KKW, while the background noise is clearly visible. Additionally, the profile curve through the FDK volume reveals some noise at the interior part of the cube. As it can be seen from the rightmost images in Figure 3.7, the noise inside the KKW is even stronger for the reconstruction calculated with Katsevich's FBP. However, the background noise is lower than in the helical case than in the circular case, as it has been described above together with the SNR.

A closer look at the corners of all three reconstructed cubes suggests that these areas are undersampled in the real datasets and that this undersampling is more severe in the circular case and thus for the FDK method, than it is for the helical case in combination with the exact FBP (see central and



(a) (b)

Figure 3.8: Artifacts at the top of the reconstructed cube. In the FDK reconstructed object (b) cone beam artifacts occur at the top of the cube. When using the Katsevich's FBP algorithm no cone beam artifacts are visible (b).

Table 3.4: Differences of actual and nominal distances between the spherical caps in μm

Traj.	Data	Proj.	All planes	Plane 1 only	Plane 1 to 2 and 3
Circ.	Real	1000	9.1 ± 8.4	2.5 ± 1.9	11.6 ± 7.6
Circ.	Real	2000	9.0 ± 8.4	2.6 ± 2.0	11.5 ± 7.6
Circ.	Sim.	2000	10.4 ± 9.3	3.5 ± 4.2	17.9 ± 8.1
Hel.	Real	1000	10.9 ± 9.4	2.6 ± 2.1	12.4 ± 9.3
Hel.	Real	2000	10.6 ± 8.5	5.2 ± 3.3	11.4 ± 8.3
Hel.	Sim.	2000	5.7 ± 5.2	2.3 ± 3.8	7.8 ± 5.2

rightmost images in Figure 3.7). In addition to this, the areas close to the sides of the cube seem to be slightly blurred and noisier than the corners of the cube. The reconstructions of the simulated datasets do not show these artifacts. Because of this and due to the drop-off at the corners the sides of the cube appear slightly bent in the real circular and helical datasets. The reasons for this bend artifact could be X-ray scatter of the KKW, as these this effect has not been observed for the simulated projection sets where the sides of the calotte cube have been always perfectly flat. Furthermore, Figure 3.8 visualizes another artifact which only occurs in the Feldkamp reconstructions. When looking at the cube from the side it shows so-called Feldkamp artifacts at the top and at the bottom of the object of interest. These artifacts exist because the circular scanning geometry does not fulfill the Tuy's sufficiency condition [59], such that the acquired projection data does not provide a sampling of the frequency space which is sufficient for an exact reconstruction (compare with Barrett and Keat [26]).

3.3.2 Length Deviation

As mentioned earlier, three sides of the calotte cube contain 25 equally spaced spherical caps. To quantify the length deviations of the actual reconstruction and the calibrated nominal size of the scanned KKW the distances between all 75 calottes have been calculated for all real and simulated datasets. As a result, for each of the three side planes $(25^2 - 25)/2 = 300$ distances have been computed, while the number of distances calculated for the whole KKW with respect to all caps in all planes was $(75^2 - 75)/2 = 2775$. The differences between the actual and the nominal distances of the spherical caps were then compared to make statements about the measurement uncertainty.

For each of the six projection sets Table 3.4 presents the mean and the standard deviation of the absolute differences of the actual and nominal lengths. The calculation of the values in the fourth column of the table considers all 2775 distances between all calottes. The results in this column show

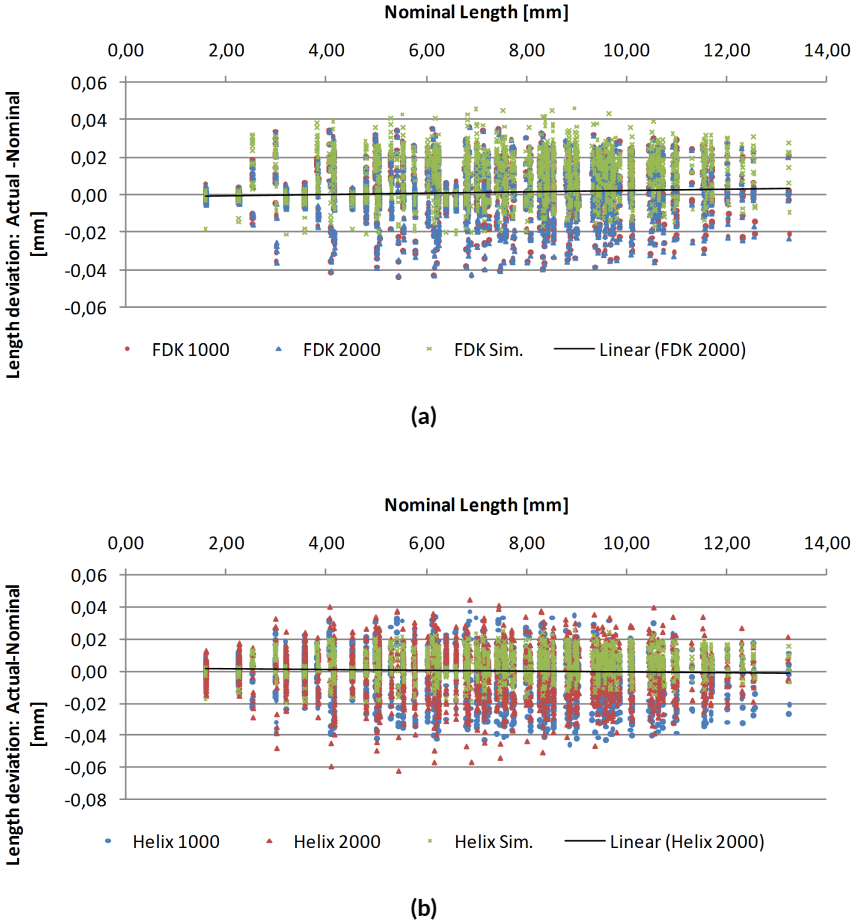


Figure 3.9: Deviations of the actual and the nominal distances between the spherical caps of the cube. The length deviations for all circular (a) and helical (b) reconstructions are plotted against their calibrated nominal lengths. For the reconstructions from the 2000 real projections a linear trend line has been fitted to the datasets.

that the mean of the absolute length deviation for the circular reconstructions is between 9 and 10 µm. This range is about one third of the voxel sampling distance of 35 µm. For the cube reconstructed from the simulated circular projections the average length deviation of 10.4 µm is slightly higher that from the real datasets (9.1 and 9.0 µm). Theoretically the simulation results (10.4 ± 9.3 µm) should provide a lower bound for the length deviations. However, for the circular reconstructions this is not the case. This behavior might be a result of the template fitting procedure used in VGStudio MAX, as described in Section 3.2.4. For the circular reconstructions the fit was more accurate in terms of the residual error with the real than with the simulated dataset. In contrast to this, the length deviations resulting from the reconstructions computed with Katsevich's exact helical FBP from the real projection data are higher than those of the simulated data. With a mean length deviation of 5.7 µm, which is about one sixth of the voxel size, the theoretical lower bound for Katsevich's algorithm is even lower than any results observed in the circular cases.

The length deviations in the fifth column of Table 3.4 have been calculated by considering the 25 spherical caps in plane 1 of the cube. With values in the range of 2.3 to 5.2 µm, the resulting deviations

are significantly lower than the ones computed in the previous column. Although, the real circular reconstructions still outperform their corresponding helical ones. For the simulation results similar observations as in the fourth column can be made: (1) the FDK reconstruction of the circular data performs worse than the real datasets and (2) the mean lower bound ($2.3\,\mu$m) given by the helical simulation study is not reached by the reconstruction from the real data. These two statements hold also true when considering the length deviations in the last column of Table 3.4 which have been computed between the top plane, i.e. plane 1 and the side planes, i.e. planes 2 and 3. The lengths within each of the three planes and between the caps of the two sides of the cube have not been included in the aggregations of the last column, so that a total number of $25^2 = 1250$ calotte distances have been considered. The results of these calculations show for all cases slightly larger values, from 7.8 to $17.9\,\mu$m, in comparison to the deviations listed in the fourth column where all spherical caps were considered. In contrast to the fourth and fifth column, the FDK and Katsevich's algorithm deliver almost equal length deviations ($11.4\,\mu$m for the helical and $11.5\,\mu$m for the circular trajectory) when the reconstruction was run with 2000 real projections. In accordance with the second statement made above, the theoretical lower bound of $7.8\,\mu$m resulting from the measurements performed on the simulated helical data is not reached by the other helical cases.

In Figure 3.9 the calibrated nominal lengths of the features have been plotted against the length deviations. The comparison shows that the reconstructions from 1000 and 2000 projections with the FDK method deliver almost the same results independent of the nominal length. As suggested by the linear trend fitted to the blue data points in Figure 3.9a, a small positive increase in the length deviations becomes visible with larger nominal lengths. However, this trend does stay below 0.01 mm. On the contrary, a positive bias in the length deviations can be observed for all nominal lengths in the reconstruction from the simulated projection set. This offset means that the actual reconstructed lengths are on average smaller than their corresponding nominal lengths. The reason for this effect could be that the template fit in VGStudio MAX performed not that accurate for the simulated datasets as for the real ones and, as a consequence, the iterative resolution adjustments were carried out less precisely.

In contrast to the FDK cases, Figure 3.9b shows no trend and no bias for any of the reconstructions with Katsevich's exact FBP algorithm. For the real data the helical trajectory delivered length deviations with quite equal distributions for the 1000 and 2000 projections, similar to the results of from the circular scanning path. As pointed out earlier for the simulated projections, the measurements carried out on the reconstruction of the helical data show a smaller variance, within the range of -20 to $20\,\mu$m, in comparison to ones made on the circular reconstructions which range from -20 to $40\,\mu$m (compare with Table 3.4).

3.3.3 Form Deviation

In addition to the length deviations, the differences between the actual and the nominal surface form of the spherical caps have been computed in accordance with equation (3.34). In Table 3.5) the mean

Table 3.5: Differences between actual and nominal surface form Z_t of the spherical caps in µm

Traj.	Data	Proj.	Plane 1	Plane 2	Plane 3	All planes
Circ.	Real	1000	11.5 ± 5.9	11.8 ± 5.1	13.5 ± 6.3	12.2 ± 5.8
Circ.	Real	2000	10.1 ± 4.9	10.6 ± 4.9	11.2 ± 6.4	10.6 ± 5.4
Circ.	Sim.	2000	7.8 ± 7.9	9.8 ± 3.7	5.6 ± 5.2	7.7 ± 6.1
Hel.	Real	1000	14.0 ± 8.0	19.7 ± 6.4	18.4 ± 8.1	17.3 ± 7.9
Hel.	Real	2000	13.4 ± 7.5	20.0 ± 6.9	21.5 ± 6.4	18.3 ± 7.7
Hel.	Sim.	2000	12.8 ± 8.3	10.4 ± 3.7	6.0 ± 5.2	9.7 ± 6.6

and standard deviations of the differences between the surface forms have been calculated over every single plane and over all planes. With respect to the real projection sets the mean form deviations presented in the table range from 10.1 µm, which is about one third of the voxel sampling distance, for the circular scan from the 2000 projections up to 21.5 µm for the helical reconstruction from the 2000 projections. In general, the surface from deviations computed from the circular trajectories are up to a factor of 2 more accurate than those of the helical cases, especially for plane 2 and 3 at the side of the cube (see columns 5 and 6). Additionally, the top plane, i.e. plane 1, shows the lowest differences in the surface from factor in comparison to the other planes (see fourth column). Moreover, Table 3.5 demonstrates that the surface form factors from the simulated projection sets are always lower than the ones of their corresponding real datasets, independently of the evaluated planes. However, the simulation studies suggest that the FDK algorithm is able to deliver more precise reconstructions of the surface form than Katsevich's algorithm for the here studied cases.

3.4 Discussion

On the one hand, the evaluations performed in the previous section demonstrate that the reconstructions from the real circular projection sets allowed more precise measurements and resulted in a lower measurement uncertainty with respect to both, length and form deviations. Although the reconstructions from the helical trajectory showed no cone-beam artifact, i.e. Feldkamp artifacts, at the bottom and top of the object of interest, they were outperformed by the FDK method in relation to the surface form factors. On the other hand, Katsevich's helical FBP algorithm had the smallest length deviations for the reconstructions of the simulated projection set in comparison to all other datasets. This gap between the processing of real and simulated projections suggests that in principle the exact method of Katsevich's as the potential to reach more accurate results for real datasets, which can become even better than those of than the FDK method.

The lower SNR in the FDK reconstructions can be explained by the following coherences. Since the Tuy-Smith sufficiency condition [59] is not fulfilled by the circular scanning trajectory, spatial resolution is lost in the direction of the rotation axis (compare Hiller et al. [32]). As a consequence the FDK method can only compute an approximate solution to the reconstruction problem and thus causes Feldkamp artifacts at the lower and upper boundaries of the cube. These artifacts raise the

background noise in the final image, such that the SNR decreases and is lower than for the exact FBP (see Table 3.3). Conversely, Katsevich's algorithm does not incorporate all the available detector information into the reconstruction process, since only data from inside the TD window is used (see Section 3.2.1.5). Although detector utilization is lower than for that of the FDK algorithm, the helical reconstructions of the cube show a higher SNR. The use of a technique which increases the detector utilization by the incorporation of additional and redundant detector data into the exact helical reconstruction, like suggested by Katsevich [64] and Schöndube et al. [108], would not only affect the SNR positively, but could also improve the accuracy of measured length and surface form together with a reduction of the measurement uncertainty.

In summary, it must be considered that the horizontal defector shift for the circular reconstructions was computed automatically from the sinogram using VGStudio MAX, whereas the detector offset for the helical projections has been determined heuristically. Therefore, the latter value is not as reliable as the automatically estimated one what in turn influences the uncertainty of the measurements conducted on the helical reconstructions. These circumstances in combination with the fact that the helical trajectory has an additional degree of freedom in comparison to the circular scanning path demonstrates that Katsevich's algorithm depends on a larger number of geometric parameters than the FDK method. These parameters influence the quality of the reconstructed image vitally and leave more space for measurement errors. As shown by the experiments presented in this chapter a precise knowledge of the geometric parameters and especially the fine tuning of the horizontal detector offset is inevitable to achieve accurate results, in particular for the length and form deviations. However, a heuristic method for the determination of geometric system parameters is not practicable, especially in industrial applications with large volumes of interest. Therefore, the next chapter will present research conducted in the field of automatic geometrical scanner calibration together with a novel strategy that can be employed to estimate the parameters of helical scanning trajectories.

In questions of science, the authority of a thousand is not worth the humble reasoning of a single individual.

Galileo Galilei

4 Geometric Calibration

The previous chapter has shown that a precise knowledge about the computed tomography (CT) system setup, in particular about the geometric parameters, is inevitable to keep image reconstruction artifacts at a minimum and to make accurate statements about the measurement uncertainty. Therefore, this chapter is going to introduce a novel geometric calibration technique for cone-beam computed tomography (CBCT). The method employs an iterative multi-resolution 2D-3D image registration approach to maximize the similarity between the originally acquired and the simulated reprojected projections. The method derived in the following is able to retrieve the geometric misalignment of a CBCT scanner precisely without the need of a dedicated calibration phantom or any other specific markers placed inside the object of interest. Consequently, the geometric system parameters can be estimated online from the acquired projections. Note that the author of this thesis has presented parts of the aforementioned approach in a publication (see [2]). Therefore, parts of this chapter overlap in terms of content, while other parts provide additional details and improvements. Moreover, figures and tables shown during this chapter are inherited or adjusted versions of those used by Muders and Hesser [2].

The here developed method has several differences to current state-of-the-art registration based calibration methods: (1) the approach evaluates the mutual information (MI) to maximize the similarity between the simulated and the original projection sets over multiple resolutional scales, (2) it uses a consistent strategy to keep the intermediately reconstructed volume up to date, (3) the employed forward- and backprojection operations have been implemented on the graphics card and a (4) stochastic ray sampling has been integrated into the projection simulation process. The combination of all those points ensures that the proposed algorithm retrieves the geometrical misalignment accu-

rately and that the overall runtime performance of the method is comparable to current registration-based calibration approaches. Furthermore, the conducted experiments will show that the achieved calibration results are as accurate as those of current phantom-based methods.

Additionally, the results in this chapter will demonstrate that the algorithm can be applied to circular and helical scanning paths with arbitrary cone angles. Independent of the underlying system geometry, the here presented approach converges stably to the correct solution, while being robust against the amount of projection noise. Moreover, the studies are conducted on different projection sets from simulated and real data to verify that the algorithm is able to deal with arbitrary imaging objects.

4.1 Motivation

Over recent years, the ability of CT to image internal structures of objects non-destructively has played an increasing in the medical and in the industrial sector. It has been used to gain precise information about patients and it improved the decision making with respect to diagnoses and treatment planning in the clinical area. Moreover, CT has driven forward major developments in the industrial field by supporting the quality assurance process through high-precision imaging methodologies, in particular for applications of non-destructive testing (NDT).

However and as pointed out in the previous chapter, the quality and the exactness of the reconstructed CT images depend on numerous factors, such that multiple overlapping effects can cause artifacts in the final results and decrease the precision of the subsequently performed analyses or measurements. The decrease in quality can basically be categorized into three groups of effects [26]. The first class includes effects, which are caused by the inadequate modeling of the imaging system. To this group belong beam hardening, X-ray scattering, photon starvation, partial volume effects, discretization and undersampling effects. Patient- and object-based influences during the image acquisition, such as object motion or truncated projection data, from the second category of effects. The third group of artifacts is caused by the calibration of the scanner, which is influenced by the shortcomings of the CT scanner. These imprecisions occur since a precise mechanical alignment of the imaging system can only be achieved up to a certain degree. Consequently, an ideal setup system cannot be realized in practical applications, whereas geometrical misalignment such as detector shift and tilt are likely to be observed.

For CBCT it is well known that only a slight divergence of the actual and the nominal geometrical system parameters can introduce visible artifacts into the reconstructed image, such that the image quality is degraded severely [109, 110]. Since due to this measurement uncertainty, the determination of the exact geometry of a CT system is challenging and a calibration phase is used to estimate the inaccuracies prior to the actual scan of the object of interest. During such a calibration procedure, the geometry of the scanner is determined to achieve a proper virtual alignment of the system. This process might employ projection images of a so-called calibration phantom, which can include several geometric markers as guidance to the calibration algorithm or it is carried out using only the data

acquired from the object of interest itself. The main goal of the calibration phase is to define a set of projection matrices that reflect the system geometry as precisely as possible. In order to estimate these matrices, a set of geometric parameters has to be known. As the parameter values can change with each individual data acquisition performed on the cone-beam system, the ideal case would be to perform a full system alignment prior to each scan. For these reasons, the calibration routine developed in this chapter models the projection matrices by a set parameters, such that a subset of the matrix values is considered as estimates of a parameter optimization procedure.

As a consequence, the objective of this chapter will be the development of a novel calibration strategy, which is robust in terms of projection noise and which employs an optimization scheme with a stable convergence against the true constellation of the geometric system parameters. The here proposed approach is similar to an iterative rigid 2D-3D registration process in the sense that the similarity between the original projections and a set of simulated projections is maximized by the use of the MI. Therefore, a preliminary reconstructed object is constantly updated from the initial set of projection images in a multi-resolution approach with changing geometric parameters, while the optimization process is driven by the projection simulation of the intermediate volume reconstruction. The projection operations have been implemented carefully on the graphics processing unit (GPU) in combination with a stochastic projection sampling strategy. This design allows the final algorithm to achieve runtimes between five and fifteen minutes in dependence of the underlying dataset. In contrast to algorithms presented in [111–114], which define an objective function in the space of the reconstructed object, the here developed method, with its estimation technique running in the projection space, can be extended to perform the needed optimizations per projection. Moreover, the approach presented in this chapter suggests a general scheme, which is applicable in various application areas of CBCT (e.g. medical and industrial cases) and which can be applied to different scanning geometries, such as circular and helical source trajectories.

This chapter is organized as follows. Section 4.2 will review existing CBCT calibration methods. In Section 4.3 a framework will be introduced which can be used to represent various setups of CBCT scanners. Furthermore, Section 4.3 will derive and explain the novel calibration strategy. Section 4.4 will describe the conducted experiments and present their results to demonstrate the stability and robustness of the method. The discussion given in Section 4.5 will then compare the here achieved results with other existing registration-based calibration approaches. A final perspective on future work in relation to the here presented algorithm will be given in the conclusions of Chapter 6.

4.2 Related Work

In the areas of cone-beam CT image reconstruction, geometrical scanner calibration and image registration a large amount of research has been conducted in the past, in particular over the recent years. The major advances in these fields will be described and summarized in the following sections.

4.2.1 Cone Beam CT Reconstruction

In applications with a practical orientation, filtered backprojection (FBP) approaches are often preferred for the image reconstruction from projection data acquired with a CBCT scanner. The reason for this is that the filtration and backprojection operations allow a straightforward implementation, which can be realized efficiently on current hardware to achieve a high computational performance [43].

For the image reconstruction in the medical area and for industrial CT the algorithm of Feldkamp, Davis and Kress (FDK) [12] is widely used for cone-beam projection data scanned with a circular X-ray source trajectory. However, for the case when the X-ray focal spot follows a helical scanning path, Hiller et al. [32] have demonstrated for an industrial setup that the FDK algorithm produces severe artifacts, as it has also been discussed in Section 3.3. Turbell [25] has additionally reviewed these image defects in a medical context. For the compensation of the so-called Feldkamp artifacts (see Section 3.4) various adaptations of the FDK algorithm have been developed by many researchers, as described extensively in Section 3.1.1 (compare with [25, 115]).

Although, these extensions of Feldkamp's method improve the image quality of the final reconstruction from the helical projection data, the breakthrough for helical CBCT (H-CBCT) has been made by Katsevich [13, 19] by the derivation of a theoretically exact FBP algorithm. Multiple implementations of this approach haven been presented and discussed by Noo et al. [27] and by Yu and Wang [80] as well as in the course of Chapter 3. Realizations of Katsevich's algorithm on parallel hardware have been suggested by Yang et al. [28], Fontaine and Lee [29], Yan et al. [90] and Benquassmi et al. [30] (also compare with Section 3.1.2). Despite from errors introduced by the discretization and interpolation steps, it has been shown in Chapter 3 for H-CBCT that the additional degree of freedom due to the translation stage makes a correct alignment of a helical scanner inevitable (also see [93]). In addition to this, the study by Hass and Faridani [94] underlines how sensitive Katsevich's algorithm reacts to incorrect assumptions made on the detector data and how this can lead to so-called "comet tail" artifacts.

4.2.2 Geometric Calibration

As pointed out in the introduction, misaligned scanner geometry or imprecise assumptions about the geometric setup of the scanner can lead to strong artifacts in the reconstructed image. The influence of the misalignment parameters on the image quality of the CBCT reconstructions have been studied intensively by many authors (see [26, 110, 116–118]). For example, Sun et al. [110] discuss the effects of each misalignment parameter on simple test objects, i.e. points, spheres and boxes. In [112, 113, 119, 120] the misalignment artifacts that occur in helical CT are displayed and in [1] a comparison between FDK and Katsevich's algorithm with respect to misalignment and image quality has been performed.

To solve the problem of misaligned scanner geometry and to reduce the caused artifacts, a multitude of parameter estimation strategies has been developed in literature. Many of these of geometric

calibration methods will be reviewed in the following and can be categorized into basically two classes: offline and online calibration methods.

4.2.2.1 Offline calibration

Approaches belonging to the class of offline calibration methods need a separate scan of a so-called calibration phantom. This phantom has to be specifically designed to guide the parameter estimation process. For example, the phantom can contain positional markers, i.e. small metal balls, which are tracked by the estimation algorithm to derive the system geometry from their movement. For this reason, offline methods are also often called "phantom-based" calibration techniques [121].

Gullberg et al. [122] have initially proposed the basic concept of offline calibration as a parameter optimization problem carried out in three steps: (1) measure the projected location of one or multiple point objects, i.e. markers, on the detector, (2) then determine a set of non-linear equations to express the found locations by the unknown geometric parameters and (3) finally solve the equations using an iterative method. In the case of [122] the Levenberg-Marquardt algorithm was used. Li et al. [116] suggested an extension of Gullberg's method by including mechanical offsets into the parameter estimation process. Later, Rougée et al. [123] introduced constraints to hold the parameters in a certain range to make the iterative solution more stable. In addition to this, they used more than one point object for the calibration to improve the precision of the calibration. Although Rougée et al. have achieved more precise results than the previous methods, the manufacturing of the calibration phantom required a very high accuracy. Moreover, in [116, 122, 123] assumptions were made on the parameters in a sense that some of them have only a minor influence on the reconstruction and can be set to zero. In contrast to this, Rizo and Grangeat [124] proposed a method where the complete scanner could be misaligned. They showed that seven parameters are enough to calibrate circular CBCT (C-CBCT) and that correlations in the simultaneous estimation of all of them can be avoided by a splitting into intrinsic and extrinsic parameters followed by a step-wise optimization.

All the above methods try to estimate the geometric parameters by solving a highly non-linear optimization problem that can run into local minima depending on the order in which the parameters are estimated and on their initial values. To avoid these problems, Noo et al. [109] introduced a set of intermediate parameters, which lead to the straightforward problem of fitting ellipses to the projected locations of the point objects. Their estimation process showed to be robust, can be expressed by explicit analytic formulas, is easy to implement and uses a simple phantom with two steel balls. As a result, the calibration of six unknown parameters can be achieved from a minimum of six views. With a similar calibration phantom Bronnikov [125] estimates two parameters from two views that are $180°$ apart. However, Noo et al. [109] observed that additional steel balls make the calibration more stable and that more than two markers are needed to calibrate for rotations around the horizontal detector axis. Results of their approach on a helical scanning path have been also published in [119]. For the use of ellipse fitting on complex trajectories and on nine parameters for each projection Cho et al. [126] presented a technique that uses a calibration phantom of 24 steel balls. Chetley

Ford et al. [127] improved this implementation to make it more robust against phantom fabrication imprecision and inaccurate placement of the phantom. By a novel ellipse fitting that employs singular value decomposition presented in [128], the minimal limits on the design of calibration phantom have been reduced even further.

In contrast to this, for the accomplishment of clinical quality assurance of X-ray systems in the area of image-guided radiotherapy a simple ball-bearing phantom is commonly used to estimate varying detector offsets [129]. Moreover, Sawall et al. [120] have proposed a geometry estimation algorithm for micro-CT, which does not require a dedicated calibration phantom, such that their approach is able to calibrate the imaging system geometrically from a sequence of acquired projections of a single metal bead. Their estimation strategy works for spiral, sequential and circular cone-beam micro-CT and uses a genetic algorithm to find the global minimum. Further studies [110, 130] have successfully retrieved the geometrical setup of the imaging system from multiple steel balls. In the area of computed laminography Yang et al. [131] have described the estimation of the position of the X-ray focal spot by the combination of projections images acquired over multiple views of spherical objects. Another contrary approach, which uses Fourier analysis of the projection orbit data to find a solution for six system parameters, has been developed and analyzed by von Smekal et al. [117],

Nevertheless, all offline calibration techniques reviewed above require a more or less accurately manufactured calibration phantom. The needed precision of the employed phantom depends largely on the used method. Therefore, the precision of the knowledge about the locations of the markers in combination with the selected parameter estimation strategy influences and can determine the resulting accuracy of the system calibration.

4.2.2.2 Online calibration

Contrary to offline calibration techniques, a separately performed image acquisition of a dedicated phantom is not required for the geometric alignment of a CT scanner with an online calibration approach to find a solution of the parameter estimation problem. This is why online approaches are often called ”self-calibration” [132] or ”automatic” [133] calibration techniques, because they calculate a certain set of geometric conditions from the acquired projection data itself.

In this context, Panetta et al. [134] has proposed and online CBCT calibration approach, which defines an objective function using a particular group of 3D objects by exploiting the redundancies of the cone-beam projection data on the circular trajectory. In their experiments, they have proven that their optimization technique can be used to estimate the most critical geometric misalignment parameters of a C-CBCT system. Additionally, they have verified that their cost function can be used for objects that do not belong to the class which has been used during the design phase of the algorithm. In a similar manner, Patel et al. [133] have determined with respect to a plane parallel to the detector the actual angle of the rotation axis relative to the ideal y-axis. Their algorithm uses a registration of antiposed projection images in combination with a sinogram analysis to correct the actual horizontal offset of the rotational axis. By a registration-based optimization of a cost function based on the

cross correlation using a least-square minimization scheme in the projection space Mayo et al. [135] and Sasov et al. [4] have determined the vertical and horizontal shifts of the CBCT detector. The application of this method in the area of micro-CT achieved a system alignment in the order of sub-micrometers. A similar method for calibration of circular cone-beam geometry has been proposed by Wein et al. [132]. However, they used simultaneous iterative reconstruction technique (SIRT) for the image reconstruction in combination with a separate parameter optimization loop which minimizes the reprojection error in terms of the sum of absolute differences (SAD). The application of Wein's method for a dental CT case showed a precise projection-wise scanner alignment with a significant reduction of artifacts in the corrected images, while a full optimization terminated within a few minutes.

In contrast to the calibration methods above, which define their objective functions in detector domain; other approaches have been proposed which design cost functions not over the projection images, but in the space of the reconstructed volume. One of these approaches has been developed by Kyriakou et al. [136, 137] and uses an iterative correction scheme based on the minimization of the volume histogram entropy. For the optimization, they employ a multi-parameter simplex algorithm, whereby the backprojection operation was carried out efficiently on the graphics card. Reconstructing only a single volume slice from a down-sampled source trajectory and fitting a circle to it, they were able to reduce the overall computational complexity and achieved a runtime of 1-3 seconds for a complete calibration. Similarly to this approach, a technique inspired by the optical auto-focusing mechanisms has been suggested by Kingston et al. [111, 112]. They maximized the sharpness of the reconstructed image in terms of the L_2 norm of the image gradient to estimate the unknown parameter constellation of a circular system geometry. Their method has also been modified and applied to H-CBCT in [113]. In addition to this, Meng et al. [114] described an online calibration method, which exploits the symmetry property of the sum of projections (SOP). By the derivation of an error function from the SOP, they obtained the crucial geometric parameters of a C-CBCT system using a global optimization method. With their method, they were able to achieve accurate calibration results even for noisy projection data, whereby an image quality comparable to that of existing offline calibration methods was reached. Only quite recently, Wicklein et al. [138] have analyzed and evaluated various image features, such as the histogram entropy, the total variation and other texture-based image characteristics in relation to their capabilities to quantify a scanner misalignment in the context of the backprojection mismatch (BPM).

4.2.3 2D-3D Registration

In principle, the geometric scanner calibration problem can be interpreted as a rigid 2D-3D image registration operation, which iteratively aligns an approximately reconstructed intermediate volume with a given set of CT projections. Under the assumption that a preliminary image can be reconstructed from the initially known geometric system configuration, an iterative procedure usually optimizes a parameterized projection matrix that steers a forward projection operator. This operator

generates a digital reconstructed radiograph (DRR) from a 3D volume, which is then compared to the actual measured 2D projection.

Over recent years, major advances have been achieved in the field of image registration. Numerous measures have been suggested to compute the similarity between images, even for datasets with different modalities. In addition to this, specific optimization and regularization schemes have been developed to improve the stability of image registration algorithms. As a result, image registration has become a well-researched and understood field [139, 140]. Maintz and Viergever [141] and Zitová and Flusser [142] give a full overview on the specific registration strategies while pointing, explaining and comparing the required key components.

Detailed comparisons of various image similarity metrics, which have been used for 2D-3D image registration, have been given in the publications of Penney et al. [143] and Wein [139]. Fast and easy to compute similarity metrics are the SAD and the sum of squared differences (SQD) [139, 140], which can either be computed on two individual images or on two sets of images. Depending on the given data, these two metrics are not always the preferred choice. Other similarity measures have been suggested to make the optimization process more robust, to cope with different image modalities and to improve the precision of the final registration result. The normalized cross-correlation (NCC) is a very often used [139, 143] similarity function, since it is not affected by variations in the contrast and brightness of the registered images. Additionally, the mutual information (MI) [143–145] has proven to be a stable measure for multi-modal image registration [146, 147].

In addition to the similarity measures above, van der Bom et al. [148] has compared the different optimization strategies, which have been proposed for the estimation of the parameters defining the 2D-3D correspondence. Moreover, Viola and Wells [149, 150] have demonstrated that the employment of a random ray sampling during the DRR generation process suffices to steer the optimization procedure and to achieve reliable registration results. In [151] the approach of Viola and Wells has also been shown to be effective when applied to a random line sampling. Müller et al. [146, 147, 152] have reduced the computational complexity of the 2D-3D registration process by the stochastic sampling even further, such that the calibration parameters were estimated within less than a minute.

The important building blocks to accelerate and to successfully implement image registration on multi-core central processing units (CPUs) and high performance GPUs are compared in depth by Fluck et al. [140] and by Shams and Sadeghi [153]. With the use of modern stream processing hardware, even real-time image registration has recently become possible [154].

4.3 Materials and Methods

The basis of the calibration technique that will be presented in the following sections is a flexible framework that can be used to model various CBCT systems and their geometric misalignment by using homogeneous coordinates. For this reason, the notation that will be used to describe the system setup and its misalignment will be introduced first.

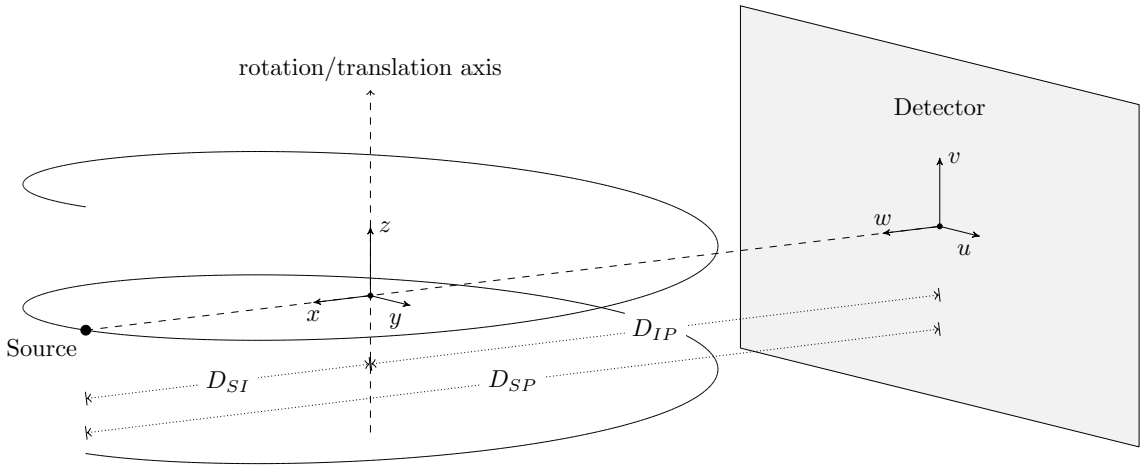


Figure 4.1: System setup for a H-CBCT scanner. D_{SI} is the distance from the X-ray source to the rotation/translation axis, D_{SP} is the distance from source to the detector and D_{IP} is the distance from the rotation axis to the detector. The world/object coordinate system is defined by the vectors x, y and z. The coordinate space of the detector is spanned by the vectors u, v and w.

4.3.1 System Setup

In principle, there exist two kinds of setups for CBCT scanners: (1) imaging systems where the X-ray source and detector rotate around and translate along the scanned object of interest and (2) CT machines where the scanned item is rotated and translated while the source and the detector are stationary. Without loss of generality, the derivations throughout this chapter will refer to the moving source-detector scanner. Additionally, the assumption will be made that all detector elements have the same height and width and are equally spaced in an absolutely flat plane, so that the physical detector does not have any distortions. For the geometries that will be discussed below, the X-ray source rotates in a counter-clockwise direction with a radius of D_{SI} around the z-axis on a circular or on a helical path. The detector is positioned at a distance of D_{SP} from the radiation source, whereby the distance from the rotation/translation axis to the detector can be calculated straightforwardly as $D_{IP} = D_{SP} - D_{SI}$. For the case of a H-CBCT scanner, the source moves along the translation axis in the positive z-direction. The system setup for the helical case is depicted in Figure 4.1. The C-CBCT system setup is similar to the one shown in Figure 4.1 with the only difference that the helical pitch vanishes to zero and the trajectory becomes a circle.

To describe the ideally aligned setup of different cone-beam scanners right-handed coordinate systems are introduced similar to [126]. The coordinate space C_w defines the world coordinate system, in which the scanned object has its origin at $(0, 0, 0)$. The z-axis of this coordinate system is identical with the rotation and translation axis of the scanner. Relative to C_w the X-ray source position is defined as $s_w := (x_{sw}, y_{sw}, z_{sw})$. The coordinate system C_a has its origin at the center of the virtually aligned detector. It is spanned by the unit vectors u, v and w, where the latter axis of C_a is pointing away from the detector towards the X-ray source. To use volume and projection data in the framework described below two additional coordinate systems are defined. The first coordinate space C_v

has the same orientation as the world coordinate system C_w, but its origin and scaling are different, so that voxel indices can be used directly in the projection operators, as it will be described below. The second additional coordinate system C_r defines pixel coordinates on the projection in a raster space with the same orientation as the detector coordinate system C_a.

The calculation of the projection of any point inside the volume of interest onto the detector employs matrices in homogeneous coordinates. These transformation matrices are tightly integrated into the forward projection and backprojection operations by following the descriptions of Galigekere et al. [18]. For the application of the matrices to any other source trajectory, apart from the circular and helical scanning paths, the derivations given below can be easily adapted. The inclusion of additional misalignment parameters can also be realized straightforwardly (compare with Section 4.3.2). Please note that the central dot symbol ($\cdot$) represents matrix multiplication in all equations throughout this and the following section, i.e. Section 4.3.2.

In the derivations below, the object of interest will be represented by a discrete volume, which contains $N_x \times N_y \times N_z$ voxels. The edge length of each of the volume elements amounts to Δx, Δy and Δz along the three axes, respectively. The voxels are arranged on a 3-D grid, whereby a single element can be accessed through its index $x_v = (i, j, k)$. Consequently and for a fixed source position, the projection of a voxel index x_v to pixel coordinates on the perfectly aligned detector, denoted by $x_a = (m, n)$ relative to C_a, can be expressed by a transformation matrix M_{va}. The decomposition of this matrix is given by

$$\begin{aligned} x_a &= M_{va} \cdot x_v \\ &= M_{pr} \cdot P_a \cdot M_{wa} \cdot x_w \\ &= M_{pr} \cdot P_a \cdot M_{wa} \cdot M_{vw} \cdot x_v. \end{aligned} \tag{4.1}$$

The first operation applied to the voxel index x_v in last row of equation (4.1) is a multiplication by the matrix M_{vw} which transforms the 3-D index in to a point of the world coordinate system C_w, which is given by $x_w = (x_w, y_w, z_w)$. The volume-to-world matrix can be defined by

$$M_{vw} = T(min_x, min_y, min_z) \cdot S(\Delta x, \Delta y, \Delta z) \tag{4.2}$$

where the point (min_x, min_y, min_z) defines the location of the lower corner of the axis-aligned bounding box of the volume of interest relative to the world coordinate space C_w. Additionally, the matrices $T(\cdot)$ and $S(\cdot)$ represent a translation and a scaling, respectively.

The second row of equation (4.1) describes the projection of the voxel in world coordinates x_w onto its corresponding detector raster coordinates x_a. Assuming that the X-ray source position is given by s_w, as defined above, relative to the world space C_w, a point in C_w can be expressed in the coordinates of the aligned detector space C_a by the application of the transformation matrix

$$M_{wa} = B \cdot T(D_{IP}, 0, -z_{sw}) \cdot R_z(-\phi). \tag{4.3}$$

In equation (4.3), the matrix R_z is carrying out a rotation by $-\phi$ around the z-axis. Thereby, the value of ϕ is given by the angle between the x-axis and the detector normal vector w. The application of the translation matrix T results in a shift of D_{IP} along the x-axis and in an additional translation along the z-axis of $-z_{sw}$. The final step in equation (4.3) preforms a reordering of the coordinate axes by the matrix B. This rearrangement is carried out, such that the y-axis becomes the u-axis and the z-axis is mapped onto the v-axis (compare with Figure 4.1). As a result, the matrix B can be expressed as

$$B = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}. \tag{4.4}$$

The explicit definitions of the transformation matrices T, S and R_z can be found in Appendix A. Additionally, note that the indices of each matrix above have been chosen accordingly to the coordinate spaces involved in corresponding transformation.

In the case of an ideal scanning geometry without any misalignments the X-ray source lies at $s_a = (0, 0, D_{SP})$ relative to the detector coordinate system C_a. As a consequence, the projective transformation that maps a given point from the coordinate space C_a onto the aligned detector can be defined as

$$P_a = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1/D_{SP} & 1 \end{pmatrix}. \tag{4.5}$$

The discrete detector data is stored in a projection image containing $N_u \times N_v$ pixels. The rows of the projections have a height of Δv, while the columns have a width of Δu. With respect to these detector properties, the projected coordinates, which result from the application of P_a in equation (4.1), can finally be mapped to raster indices x_a by the following transformation matrix:

$$M_{pr} = T(h_u, h_v, 0) \cdot S(1/\Delta u, 1/\Delta v, 1). \tag{4.6}$$

The matrix S in equation (4.6) applies a scaling by the inverses of the detector element width and height. In addition to this, the matrix T performs a translation by the offsets $h_u = (N_u - 1)/2$ and $h_v = (N_v - 1)/2$. By the substitution of the matrices M_{vw}, M_{wa}, P_a and M_{pr} from the equations (4.2), (4.3), (4.5) and (4.6) into the last row of equation (4.1) the projection matrix M_{va} can be calculated. This matrix can then be used to transform a voxel index directly from C_v to a raster coordinate in the virtually aligned projection space C_r.

4.3.2 Misalignment Parameters

In addition to an aligned scanning geometry, the transformation framework described above can also be employed to model the misalignment of a CBCT imaging system. Sun et al. [110] have demonstrated that the geometrical deviations of a CT scanner can be expressed solely as parameters of the detector position and orientation. In accordance with Noo et al. [109], Rizo and Grangeat [124] and Kingston et al. [112], the following seven parameters suffice to describe all possible misalignments of a C-CBCT scanner:

- Three translations of the detector: (1) the horizontal offset t_u, (2) the vertical shift t_v and (3) the translation along the direction of the detector normal, denoted by t_w.
- Three rotations of the detector: (1) the in-plane rotation around the detector normal vector by r_w and the two out-of-plane rotations around (2) the horizontal u-axis by r_u and (3) the vertical v-axis by r_v.
- The distance between source and sample: D_{SI}

Consequently, these seven parameters can be used to model all geometrical degrees of freedom for the circular case. Even if other researches have used a different parameterization for the system geometry, their expressions can be transformed to the parameters above by calculations similar to the ones proposed in [110]. Note that, for example, the translational movement of the detector along its normal vector, i.e. t_w, can be directly related to D_{SP}, so that the effective source-detector distance is modeled correctly in both the aligned and the misaligned detector case.

In contrast to the C-CBCT case, a helical scanner incorporates an additional translation along the axial direction, so that two further degrees of freedom are introduced. Therefore, the geometrical deviation of a H-CBCT system can be expressed by a total of nine misalignment parameters (see [113]):

- the seven detector parameters from the C-CBCT case
- two rotations of the translation axis relative to the rotation axis: r_x, r_y

In accordance with the discussion presented by Varslot et al. [113] about geometric calibration in relation to helical scans of long and short objects, the following two assumptions can be made: (1) the rotation axis and the translation axis stay approximately parallel during the scan of short objects and (2) the calibration problem for the scan of long objects can be split up into several parts, such that for each scanned segment the first assumption applies. As a result, the two rotations of the translation stage, namely r_x and r_y, can be set to zero and will be ignored in the following considerations to reduce the degrees of freedom that need to be estimated by the calibration algorithm. Additionally, it has been demonstrated by many researches [112, 114, 122, 133] that the distance D_{SI} mainly influences the magnification of the reconstructed volume and that it does not severely degrade the overall image quality. Therefore, the complexity of the parameter estimation procedure can be reduced even further

Table 4.1: Optimal Units of the misalignment parameters

Parameter	Optimal Unit	Simulations	Parameter	Optimal Unit	Simulations
t_u	$\frac{\Delta_u}{1+sin(\alpha_{fan})}$	$1.209\,mm$	r_u	$asin\left(\frac{4\cdot D_{SP}/(N_v^2\cdot\Delta_v)}{1+sin(\alpha_{cone})}\right)$	$0.991°$
t_v	$\frac{\Delta_v}{2\cdot tan(\alpha_{fan})}$	$2.344\,mm$	r_v	$asin\left(\frac{4\cdot D_{SP}/(N_u^2\cdot\Delta_u)}{1+sin(\alpha_{fan})}\right)$	$0.991°$
t_w	$\frac{D_{SP}}{N_u\cdot tan(\alpha_{fan})}$	$6.867\,mm$	r_w	$asin\left(\frac{2\cdot\Delta_u}{N_v\cdot\Delta_v\cdot(1+sin(\alpha_{fan}))}\right)$	$0.338°$

by the exclusion of D_{SI} from the statement of the problem. Consequently, the above simplifications leave six degrees of freedom for both C-CBCT and H-CBCT systems.

Furthermore, the results of Patel et al. [133] show that out-of-plane detector rotations, denoted by r_u and r_v, have only a small effect on the quality of the reconstructed image. In addition to this, von Smekal et al. [117] has found out that for the estimation of the two out-of-plane rotation parameters an error of about $50\,\%$ can be expected with offline calibration strategies in comparison to the in-plane rotation, i.e. r_w, for which the error was significantly lower. Moreover, the results of the analysis presented by Cho et al. [126] suggest that the overall precision that can be achieved by a CT calibration procedure is dependent from the magnitude of the rotations r_u and r_v. These finding imply that the accuracy of the parameter estimation decreases the larger the misalignment of the CT scanner with respect to the out-of-plane rotations. Yang et al. [130] have made additional observations and summarized them with the ones made above: (1) in practical applications it is quite difficult to obtain the two out-of-plane rotations, (2) the angles r_u and r_v have only a small influence on the artifacts inside the reconstructed volume in comparison to the remaining calibration parameters and (3) both angles can be kept small enough by a good mechanical scanner design, such that their effect on the image quality is negligible.

The observations pointed out above can be reaffirmed by looking at the influence that each parameter has on the final reconstructed volume with reference to the amount of artifacts and the image degradation that it introduces. For this purpose, a common scale of influence can be introduced by defining Optimal Units (OUs), denoted by the entity ou, for each parameter in accordance with Kingston et al. [112] and Varslot et al. [113]: with respect to the volume of interest the change in a single parameter value that is needed to introduce a perturbation that is not larger than one voxel can be described approximately by 1 ou. Table 4.1 gives the formulas for computing the OUs with respect to each misalignment parameter, whereby the fan angle and the cone angle of the imaging system are defined by α_{fan} and α_{cone}, respectively, in radians. Note that the publication of Varslot et al. [113] describes the full derivation of these formulas. For the simulation experiments conducted in Section 4.4.1, the third and the last column of Table 4.1 show the corresponding values for the OUs of each parameter. Note that larger OUs stand for a smaller influence of the related misalignment on the reconstructed image, since a larger variation of the parameter is required to introduce a perturbation

of one voxel. In contrast to this, parameters with a smaller OU have a stronger effect on the final volume. Again, it can be verified that the out-of-plane detector rotations introduce only a small amount of artifacts into the image due to their relatively large OUs. Note that a scheme for the comparison of translation and rotation parameters is given in Appendix C. Based on the observations made above, a reasonable criterion for the selection of a subset of misalignment parameters, which should be included into the estimation process during the calibration procedure, is to keep only those with the smallest OUs. By this means, those parameters are selected which have the largest influence on the overall image quality. For these reasons, the two out-of-plane rotations r_u and r_v are fixed to zero and the four parameters selected for the here presented calibration method are: t_u, t_v, t_w and r_w.

To include these four detector deviations into the projection framework, a new coordinate system C_m is defined to represent the misaligned physical detector. Given C_a, the aligned detector can be translated horizontally along the u-axis by t_u, vertically along the v-axis by a value of t_v and along the detector normal w by applying a shift of t_w. Additionally, the detector plane can be rotated around its normal w by a value of r_w. The application of these four transformations to the unit vectors u, v and w creates the vectors that span C_m. Consequently, to transform a voxel index x_v into the misaligned detector coordinate space C_m the matrix M_{wa} in equation (4.1) has to be replaced by

$$\begin{aligned} M_{wm} &= R_w(r_w) \cdot T(t_u, t_v, t_w) \cdot M_{wa} \\ &= R_w(r_w) \cdot T(t_u, t_v, t_w) \cdot B \cdot T(D_{IP}, 0, -z_{sw}) \cdot R_z(-\phi), \end{aligned} \tag{4.7}$$

where T applies a translation along the three axes of the detector by t_u, t_v and t_w, respectively, and R_w performs a rotation by r_w around the detector w-axis. Note that $R_w(r_w)$ is equal to the expression $R_z(r_w)$, since the coordinates have been reordered by the matrix B, and that M_{wa} is a special case of M_{wm} where the four misalignment parameters have been set to zero.

To perform a projection operation in the space C_m and accordingly to [134] and [155], the projection matrix P_a has to be changed to

$$P_m = \begin{pmatrix} 1 & 0 & -u_m/w_m & 0 \\ 0 & 1 & -v_m/w_m & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -1/w_m & 1 \end{pmatrix}, \tag{4.8}$$

where the source position relative to the misaligned detector space C_m can be expressed as $s_m := (u_m, v_m, w_m) = M_{wm} \cdot s_w$ where s_w is the X-ray source in world space C_w as defined in the previous section. A complete derivation of the projection matrix P_m for a misaligned scanner using homogeneous coordinates can be found in [134].

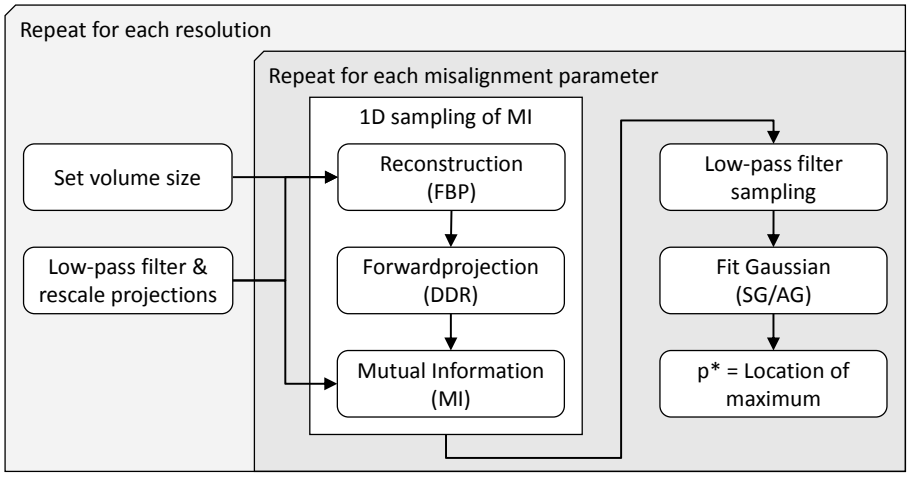


Figure 4.2: A schematic view of the novel calibration approach: for each resolution the projections and the volume are appropriately scaled, then for each parameter the optimal value $p^\star$ is estimated by a 1D sampling of the MI followed by a smoothing of the sample points and a shape dependent least-square fitting procedure.

The above replacements lead to the final transformation, which defines how to project voxel indices directly from C_v to raster coordinates x_m on the misaligned detector by using the following equation:

$$\begin{aligned} x_m &= M_{vm} \cdot x_v \\ &= M_{pr} \cdot P_m \cdot M_{wm} \cdot M_{vw} \cdot x_v. \end{aligned} \tag{4.9}$$

Using the transformation framework presented above, the novel online calibration procedure will be derived in the following section.

4.3.3 The Calibration Technique

The calibration method derived in this section steers a multi-resolution 2D-3D registration procedure by the optimization of an image similarity function. The similarity is evaluated over the set of the original acquired projections g_0 and the simulated reprojected projections g_i. Within this process the reprojections g_i are generated from a preliminary intermediately calculated volume of interest f_i. This temporary image is constantly kept up to date during each iteration, denoted by i, of the parameter optimization procedure. Figure 4.2 gives a schematic overview of the here developed algorithm. A detailed description of the approach will be presented below.

First, the set of geometric parameters is defined as $\mathcal{P}_i = (t_u, t_v, t_w, r_w) \in \mathbb{R}^4$. This set will be iteratively refined by the calibration process in each iteration step to find an optimal constellation $\mathcal{P}^\star = (t_u^\star, t_v^\star, t_w^\star, r_w^\star)$ that maximizes the reprojection similarity. Given $\mathcal{P}_i$, the simulated reprojections can be computed as

$$g_i(\mathcal{P}_i) = \mathcal{X}(\mathcal{P}_i) \cdot f_i = \mathcal{X}(\mathcal{P}_i) \cdot \mathcal{X}^{-1}(\mathcal{P}_i) \cdot g_0, \tag{4.10}$$

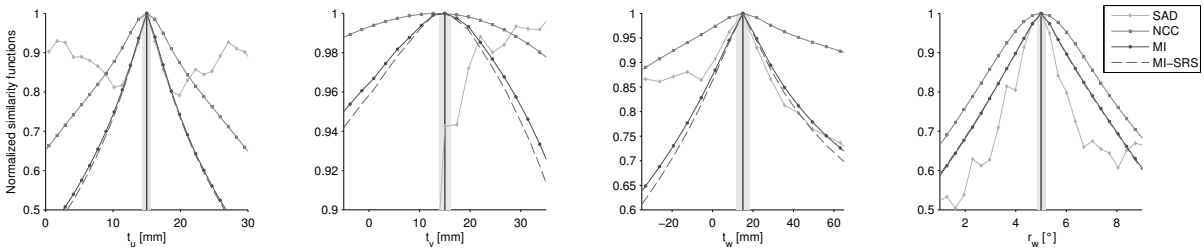


Figure 4.3: Comparison of different similarity functions for each of the four misalignment parameters : SAD, NCC, MI with full ray sampling (MI) and MI with stochastic ray sampling (MI-SRS). The functions have been sampled by varying only a single parameter at a time, while keeping all others fixed at their true value.

where the X-ray transform, i.e. cone-beam transform (see equation (3.2)), is denoted by the forward projection operator $\mathcal{X}$ which has been implemented on the GPU using a trilinear grid-interpolated ray casting approach similar to the descriptions given by Galigekere et al. [18], Xu and Mueller [156] and Weinlich et al. [20] and in Section 3.2.2. Further along, the reconstructed image f_i in equation (4.10) is computed by the operator $\mathcal{X}^{-1}$ which represents a filtered backprojection algorithm. In accordance with the descriptions and discussions presented in Chapter 3, the FDK algorithm [12] in combination with a ramp filter, also known as Ram-Lak filter, has been employed for the reconstruction of projection data acquired from the circular trajectory. For helical cone-beam projection data the implementation of Kastsevich's exact FBP algorithm [13, 27] presented in Section 3.2.1 has been used. Note that the two operators $\mathcal{X}$ and $\mathcal{X}^{-1}$ in equation (4.10) depend on the set of misalignment parameters $\mathcal{P}_i$, as they are both required the projection matrix $M_{vm}(\mathcal{P}_i)$ during the forward projection and backprojection process, respectively.

Given the original projections g_0 and the simulated reprojections g_i, different similarity metrics can be evaluated. Figure 4.3 shows a comparison of three similarity measures as functions of the four misalignment parameters: (1) the sum of absolute differences (SAD), (2) the normalized cross-correlation (NCC) and (3) the mutual information (MI). The plots were generated from the H-CBCT setup with the Shepp-Logan (SL) phantom as described in Section 4.4.1. It can be seen that the SAD, which has been used by Wein et al. [132], has a jagged shape and does not lead to the correct solution for all misalignment parameters. In comparison to the NCC, which has been employed in [4, 135], the MI shows the most defined and steepest peaks in relation to the parameter constellation of the aligned detector. For the other datasets studied during the experiments in Section 4.4, the same observations have been made for the three similarity metrics with respect to each parameter. For this reason, the derivation of the calibration procedure uses the MI.

Following the descriptions of Viola and Wells [149] and Thévenaz and Unser [157], the Parzen window method can be employed for the estimation of the joint probability density of two projection

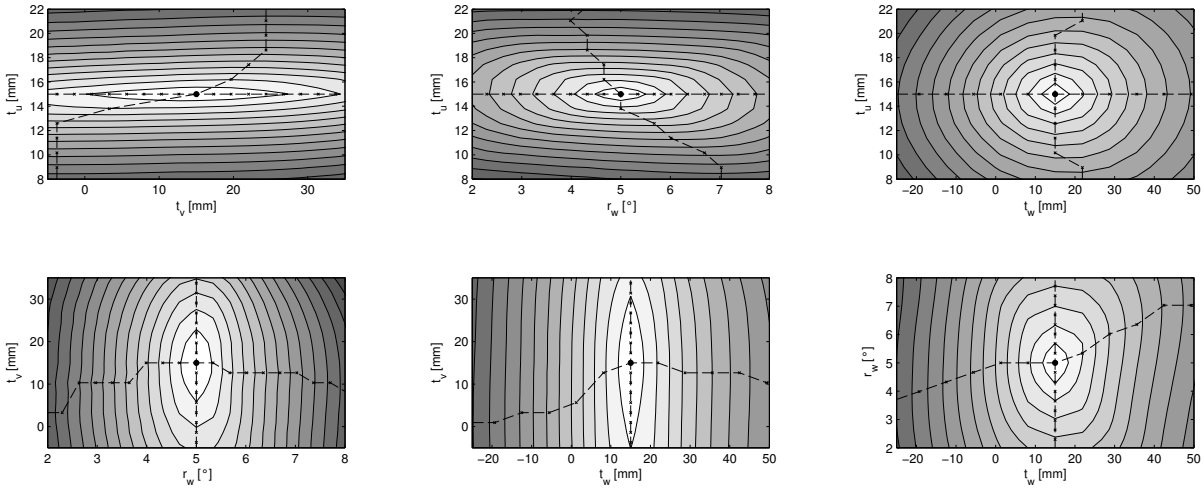


Figure 4.4: Contour plots of the MI as a function of two misalignment parameters. The true parameters values (black dot), the system setup and the projection data are the same as in Figure 4.3. Dashed lines mark the crests of the objective functions with respect to one parameters.

sets g_a and g_b. Given the two discrete sets G_a and G_b containing the gray values associated with the projections g_a and g_b, respectively, the discrete probability can be computed by

$$p(u, v) = \alpha \sum_{x_i \in V} \mathcal{W}(u - g_a(x_i))\mathcal{W}(v - g_b(x_i)), \tag{4.11}$$

where $u \in G_a$ and $v \in G_b$ and V is the domain of the projection sets, and where α is a normalization factor that ensures that $\sum p(u, v) = 1$. The weighting function $\mathcal{W}(\xi)$ is a Parzen window modeled by a cubic B-spline similar to [145, 157] and which fulfills $\mathcal{W}(\xi) \geq 0, \forall \xi$. For the two projection sets, the marginal probabilities can be defined and calculated by

$$p_a(u) = \sum_v p(u, v) \quad \text{and} \quad p_b(v) = \sum_u p(u, v). \tag{4.12}$$

With the definitions provided by the equations (4.11) and (4.12), the mutual information (MI) of the projection sets g_a and g_b can be computed as

$$MI(g_a, g_b) = \sum_u \sum_v p(u, v) log \frac{p(u, v)}{p_a(u)p_b(v)} \tag{4.13}$$

For helical system setup in combination with the SL phantom, which was used for the simulation studies in Section 4.4.1, the MI has been sampled as a function of each combination between two misalignment parameters. During the sampling of each parameter pair, the two remaining parameters were fixed to their true values. Figure 4.4 shows the resulting contour plots of each combination of parameters. These images can be used to analyze the behavior of the objective function. In the first row of the plots the crest of the MI follows a straight horizontal line with respect to the parameter t_u. This suggests that t_u is not affected by the values and by changes of any of the other parameters.

Equal observations for the horizontal detector shift have been made with respect to constellations where the values of the other three parameters differed from the ones used in Figure 4.4. Also for the datasets studied in Section 4.4, the parameter t_u showed a similar behavior. As a consequence, it can be assumed that the value of t_u can be estimated independently from the values of the other misalignment parameters. The last two plots in the second row of Figure 4.4 show that the optimal value of the detector shift t_w is changing neither with t_v nor with r_w, since the crest of the MI forms a straight vertical line. This fact suggests that t_w does not depend on the vertical detector misalignment t_v and not on the in-plane rotation of the detector. In addition to this, the first plot in the second row demonstrates that the optimal value of r_w can be found independently of the choice of the vertical detector offset t_v. Similar 2D contour plots of the MI have been analyzed during the development of the here presented algorithm with respect to various imaging system setups, for other objects of interest and at different resolutional scales. All plots showed similar characteristics and suggest that the optimal values of the studied misalignment parameters can be optimized independently from the values of the others. The above assumption must not hold to true for any combination of geometric system setup and considered objects, but in this thesis, it worked successfully for all studied datasets (see Section 4.4). Consequently, a CBCT calibration can be conducted by solving the following one-dimensional optimization problem of each parameter from $\mathcal{P}_i$ sequentially:

$$p^\star = \arg\max_p \; MI(g_0, g_i(p)) \tag{4.14}$$

where the parameter p is estimated individually while all other parameters in $\mathcal{P}_i$ are kept fixed. Note that consequently the reprojections $g_i(p)$ and thus the objective function $MI(g_0, g_i(p))$ vary only with the changes of the currently optimized parameter p.

Since the OUs provide a comparable scale, the order in which the calibration procedure estimates the misalignment parameters can be chosen by comparing the OUs in Table 4.1 with each other. To compare the rotation parameter r_w, which is given in degree, with the three translation parameters, the maximal shift that is introduced to the detector pixels by a rotation of r_w can be computed as described in Appendix C. As a result, the influence of r_w on the reconstructed image and on the objective function lies between t_u and t_v. Since a parameter with a smaller OU will influence the reconstruction more than a parameter with larger OU, the optimization order has to be chosen such that the parameters with the largest influence on the image quality are estimated first. This yields the following order:

1. Horizontal detector translation t_u
2. Rotation around the detector normal r_w
3. Vertical detector translation t_v
4. Detector translation along its normal t_w

As each evaluation of the objective function MI needs a backprojection and a reprojection step, it is not efficient to process the dataset only at the highest quality. Instead, an optimized multi-resolution 1D grid search similar to the one proposed by Kingston et al. [112] is employed to estimate the unknown parameter constellation stably and to make a solution of the optimization problem computationally feasible. As a consequence, equation (4.14) is first solved on a coarse scale with the original projections, the volume and the simulated reprojections sampled at a lower resolution. For this approach, a significant performance improvement can be expected with respect to the coarser scales. Together with this speedup, the multi-resolution strategy allows the calibration algorithm to scan a wider range of parameter values to find larger geometrical misalignments. Additionally, it has been observed in image registration literature (see Thévenaz and Unser [158] and Cole-Rhodes et al. [159]) and for the cases studied within this thesis that an optimization over multiple resolutions decreases the risk of running into local optima. These non-global solutions to the optimization problem can exist especially at higher resolution scales and are a result of the approximate and discrete nature of the involved reconstruction and reprojection implementations. As a result, the objective function can have artifacts at a finer scale, which do not exist at the coarser resolution [159]. In addition to the robustness against local optima, the employed 1D search strategy has the advantage that it can be executed in parallel over different compute nodes. As each calculation of the MI at a specific parameter constellation is independent from other configurations, samples can be calculated at the same time. However, as mentioned above, the forward and the backprojection exploit the capability of the GPU, such that for the concurrent evaluation of the objective function at multiple locations the number of graphics card devices needs to scale with the amount of parallelism and performance needed by the specific application. Note that in the studies performed within this thesis the 1D grid search has been carried out on a single GPU, such that only one sample of MI is computed at a time.

The suggested 1D search technique computes the samples $MI(g_0, g_i(\mathcal{P}_i))$ at each resolution scale by varying only a single parameter. The sampling distance for the 1D scanning of each misalignment from $\mathcal{P}_i$ is chosen so that the parameter variations are on a common scale. This means that the effects of a parameter change on the reconstructed image and on the objective function should be in a normalized range and comparable when looking at different parameters. For this reason, the changes from one to the next sample are selected to have a value of 1 ou in correspondence with the scanned parameter (see Table 4.1). Consequently, for the scanning of the function $MI(p_i)$ given by equation (4.14) the sampling distance and the sample locations can be computed by $\Delta p = OU_p$ and $p_i = \Delta\, p(-2\beta + i)$, respectively, where OU_p equals 1 ou of the sampled parameter p calculated by the formulas in Table 4.1 and $i \in [0, ..., 4\beta]$. The factor β defines the range of the sampled interval and is chosen accordingly to the currently processed resolution scale. At the highest scale β has a value of 1 and with each lower resolutional level β is increased by 1, such that it corresponds to the binning of projection and volume data carried out by the downsampling process at each resolution. This scanning technique is similar in terms of the employed sampling distance to the one used for sharpness optimization by Kingston et al. [111, 112] with the difference that the here proposed method evaluates MI as objective function.

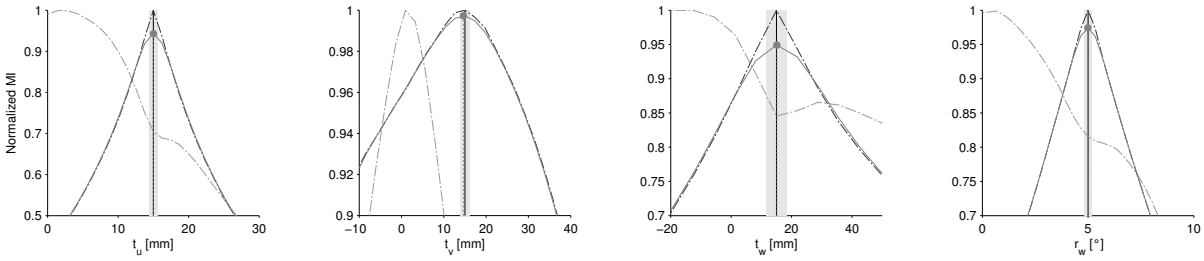


Figure 4.5: Two volume update strategies are compared with respect to the MI: (1) update after $p^\star$ has been estimated (light gray dashed line) and (2) continuous updates for each evaluation of MI (dark gray dashed line).For the second strategy the samples have been low-pass filtered (gray solid line) and the center (gray circle) has been computed by the fitting of a Gaussian function. The true values of the misalignment parameters (black solid line) are $t_u = t_v = t_w = 15\,mm$ and $r_w = 5°$. The range from half an optimal unit below and above the true parameter value is also shown (light gray area).

In addition to the common scale, the OUs can be employed as a stopping criterion for the optimization procedure, since they provide a rough limit on how accurate the calibration can get for a given geometrical system setup. In accordance with Kingston et al. [112], sub-voxel precision can be achieved by a calibrated reconstruction of the object of interest, if all differences between the calibrated nominal and the actual values of the misalignment parameters are smaller than half of an OU. The experiments in Section 4.4 suggested that the optimization of a single parameter p can be terminated when the change of its value from one to the next performed 1D scan is less than a quarter of OU_p. This criterion ensured sufficient calibration accuracy for all datasets studied within this thesis and it has been observed that once the parameter change dropped below the threshold of $1/4 OU_p$ convergence was guaranteed (compare with Figure 4.10).

The next step of the calibration algorithm applies a low-pass filter to the 1D sampling at the locations p_i of the objective function MI for each parameter. Therefore, a 1D Gaussian function is fitted to the peak of the samples $MI(p_i)$ to estimate its center. In principle, the similarity function MI can show a symmetric or non-symmetric behavior around the optimal values $p^\star$ for certain misalignment parameters. Figure 4.5 demonstrates the symmetry of the MI for t_u and r_w, while the parameters t_v and t_w do not behave symmetrically around their central peak, i.e. the optimal value. Similar observations for the shape of the objective function in relation to the four parameters have been made for all studied datasets. For this reason, a symmetric Gaussian (SG) and an asymmetric Bi-Gaussian (AG) is fitted to all four parameters inside $\mathcal{P}_i$, whereby the AG function is given by the following expression:

$$AG_{a,c,b,\sigma,\tau}(x) = a \cdot exp\left(-0.5\frac{(x-c)^2}{s^2}\right) + b \tag{4.15}$$

where a, b and c define the amplitude, the center and a constant background value, respectively. Moreover, the left and right standard deviations are given by $s = \sigma$ for $x < c$ and $s = \tau$ for $x \geq c$, respectively. Note that the SG function becomes a special case of the AG when $\sigma = \tau$. The fitting procedure itself has been carried out in a least-square sense by the application of the Levenberg–

Marquardt algorithm. After this, the location of the center c of the Gaussian peak has been determined from the fit with the lower residual norm. The found center position is used as the optimal value $p^\star$ of the currently estimated parameter and is stored in the set $\mathcal{P}_i$. This step finishes the processing of a single parameter on the current resolution scale. Afterwards, the remaining geometrical parameters are estimated sequentially using the same 1D optimization approach. In summary, the calibration method becomes more robust against projection noise and against local optima through the low-pass filtering of the samples and the shape dependent fitting has the advantage that the non-symmetric behavior of the objective function is approximated more precisely. As it will be demonstrated in Section 4.4.1, the techniques described above let the algorithm estimate the true parameter values stably and exactly (see Section 4.4.1).

Once the calibration algorithm has determined all misalignment parameters of the set $\mathcal{P}_i$ on the currently processed scale it switches to the next higher resolutional scale to repeat the optimization procedure until the highest resolution is reached. As it will be demonstrated by the results presented in Section 4.4, the proposed optimization strategy refines the parameters successively and iteratively over multiple scales, such that they converge against the true solution of the calibration problem. Moreover, Le Moigne et al. [160] have shown that in the context of image registration a similar search strategy can be applied successfully, although they used a multi-resolution wavelet-based decomposition in combination with a cross correlation measure to design their objective function.

The intermediate volume f_i in equation (4.10) can be kept up-to-date by basically two approaches. With the first volume update strategy, f_i is kept constant during the optimization of equation (4.14). The volume is only updated after a full 1D scanning and fitting has performed, such that $p^\star$ has been estimated for a single parameter. The second update scheme synchronizes the volume updates with the evaluations of $MI(g_0, g_i)$, such that for the calculation of each 1D sample of the objective function an updated version of the volume f_i is reconstructed. The comparison of both approaches in Figure 4.5 demonstrates that the first update strategy can lead to a faulty calibration result, where the final estimates of the misalignment parameters are more than 1 ou off from their true values. In contrast to this, the second update scheme shows a similarity function, which has its peak around the true parameter constellation. The main reason for this behavior is that the updated volume f_i influences the simulated reprojections g_i through equation (4.10). As a consequence, the objective function $MI(g_0, g_i)$ is also affected by changes of f_i, such that a consistently updated volume, produces more precise reprojections, which lead to a superior registration and calibration result. For this reason and in contrast to the techniques developed by Sasov et al. [4], Mayo et al. [135] and Wein et al. [132], the here derived calibration approach employs the second update strategy. In accordance with the overall calibration scheme displayed in Figure 4.2, a FBP reconstruction is performed during each evaluation of the MI prior to each forward projection step. Although the more frequent updates require additional computations and thus consume more time than the first strategy, the use of the same set of parameters $\mathcal{P}$ during the reconstruction and the reprojection process promises more precise calibration results.

In the context of performance optimization, a stochastic forward projection has been implemented to speed up the calibration process. The developed approach follows the ideas of Viola and Wells [149] and Müller et al. [152], such that during the reprojection process only a subset of the detector pixels is computed. Therefore, the ray casting algorithm is only executed for half of the detector elements at random, such that the rays are distributed uniformly over the simulated projection images. For this reason, a first step of the DRR generation process assigns a unique index to each of the $N_u \cdot N_v$ detector pixels. Then a random permutation of these indices is generated and stored in an array. Finally, the first $N_u \cdot N_v/2$ indices are picked from this list and rays are traced through the volume of interest to the corresponding detector elements. In addition to this, the selected permuted indices are stored for each projection, so that they can be reused during each execution of the forward projection operation. On the one hand, this projection-wise index buffer enables the algorithm the calculation of the compute-intensive index permutation only once at the initialization of the forward projector. On the other hand, it ensures that the random selection of the detector pixels is the same for all evaluations of the MI with respect to a single resolution scale. The latter property is very important for the estimation of the optimal parameter value, since a constantly varying random selection of detector pixels from one to the next calculation of MI could introduce strong artifact into the sampling of the objective function and lead to an incorrect solution of equation (4.14). A result of this stochastic sampling is that the time spent for the generation of the simulated reprojections is reduced by a factor of two. In addition to this, for the evaluation of MI only half of the detector pixels have to be considered, so that the overall time spent for the calculation of the projection similarity can be also approximately halved. Moreover and as demonstrated in Figure 4.3, the introduced stochastic ray sampling (SRS) does not modify the general behavior of the similarity function, nor does it shift the location of its maximum. Although a stochastic sampling can be implemented to improve the performance of the backprojection operator $\mathcal{X}^{-1}$, it has not been implemented in the context of this thesis, such that the backprojection operation still leaves room for further optimizations.

Putting all the above steps together, the 1D multi-resolution search (1D-MRS) calibration technique can be summarized with the following steps (compare Figure 4.2):

1. Initialize all parameters $p \in \mathcal{P}$ to zero.
2. Select a resolution scale factor S to start with.
3. Resize projections g_0 by the factor S.
4. Repeat steps (a) to (c) for each $p \in \mathcal{P}$
   (a) Perform a 1D sampling of $MI(g_0, g_i)$.
   (b) Low-pass filter the samples.
   (c) Find $p^\star$ by the fitting of Gaussian functions.
5. Set S to the next finer resolution and rerun the calibration from step 3.

A more detailed description of this algorithm can be found in Appendix D.

The influence of the out-of-plane detector rotations r_u and r_v on the above observations has not been studied within this thesis, so that further investigation are needed before these two misalignment parameters can be integrated into the method. For the case that the 1D search strategy described above does not work for additional parameters, the proposed estimation scheme can be exchanged by a multidimensional optimization method. To demonstrate that the here designed calibration approach works well in conjunction with a different optimization algorithm, the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) algorithm [148] will be used and compared to the 1D-MRS in Section 4.4.

4.3.4 Implementation

The calibration algorithm has been implemented as a combination of C, C++ and Compute Unified Device Architecture (CUDA) [103] routines that are wrapped and called from MATLAB. The forward projection has been optimized with ray casting using projection matrices like described in [18] to run completely on the GPU.

For C-CBCT and for H-CBCT filtered backprojection algorithms are used. With data from a circular trajectory the backprojection step of the FDK algorithm has been realized on CUDA according to [24, 43], where the filtering step is carried out in MATLAB using its internal fast Fourier transform (FFT) routines. The implementation of Katsevich's helical reconstruction algorithm follows the descriptions of [27, 28] as described in Section 3.2.1 and is split up into two steps. The first routine is performing the filtering along tilted lines on the CPU using the Intel Performance Primitives (IPP) library and parallelizes the process per projection. Afterwards, a π-segment based backprojection is computed in a second routine on the GPU using CUDA. This allows the algorithm to exploit the compute capacities of the CPU and GPU fully by filtering and backprojecting in parallel. Katsevich's FBP and the FDK algorithm are used in the calibration technique for C-CBCT and H-CBCT, respectively. For the computation of MI the methods provided by the Insight Segmentation and Registration Toolkit (ITK) [145, 158, 161, 162] have been integrated into MATLAB using the techniques described in Chapter 2.

After a set of misalignment parameters has been estimated, a FBP reconstruction with the corrected geometry has to be computed. For the FDK algorithm, the authors in [155] have shown that the misalignment parameters only have a minor influence on the filtering step. According to this, the FDK filter is not modified when the misalignment parameters change, so that the parameters estimated during the calibration are only used within the backprojection step and the projection data does not need to be filtered multiple times. In addition to this, the projection matrix M_{vm} for the misaligned detector geometry from equation (4.9) is used to perform the backprojection directly from the misaligned projections of the C-CBCT system.

In contrast to the FDK case, Katsevich's algorithm needs a perfectly aligned geometry (see [94]). To transform misaligned projection data into virtually aligned detector data a perspective rebinning has

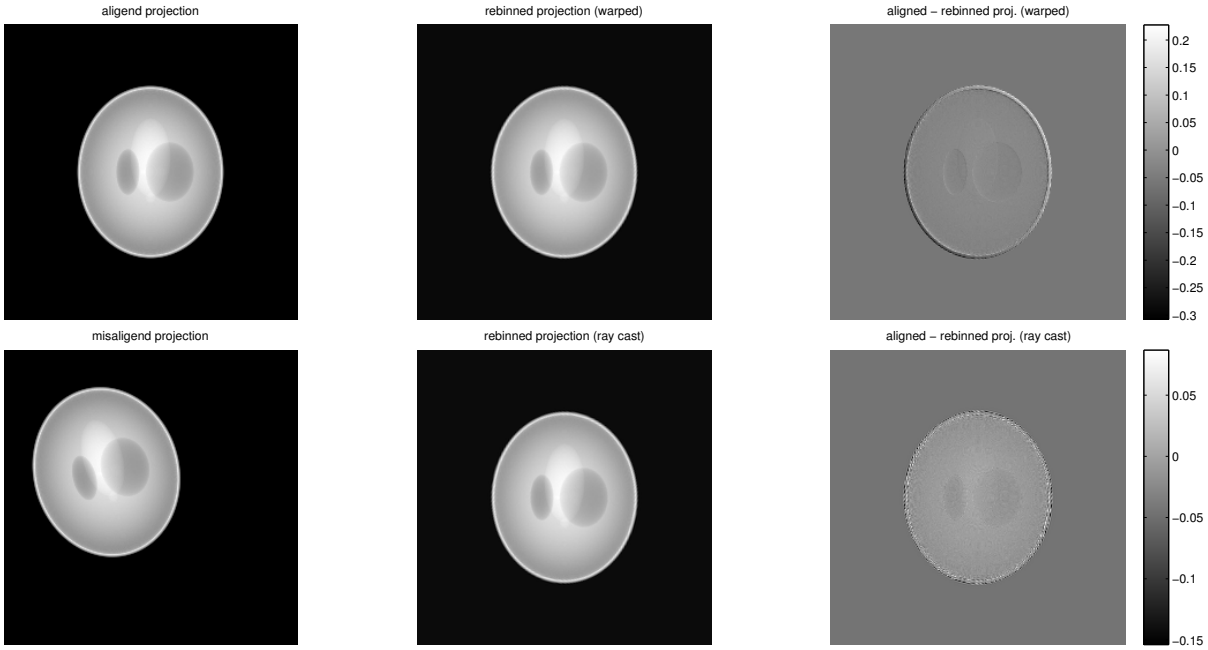


Figure 4.6: A comparison of the two virtual detector rebinning strategies. A projection of the SL phantom has been simulated using an aligned system setup (top left) and a a misaligned geometry (bottom left). The perspective warping (top center) and the ray casting approach (bottom center) have been used to correct the misaligned projection. The rebinned projections (central column) have been compared to the aligned projection by computing the difference (last column). For generation of the projection data the helical system setup described in Section 4.4.1 has been used and for the misaligned projections the geometric parameters have been set to the following values: $t_u = 50\,mm, t_v = 50\,mm$ and $r_w = 5°$.

to be applied. For this purpose, two different approaches have been compared. The first technique uses perspective warping to map the projection images of the misaligned detector onto the aligned detector plane. It has been realized by using MATLAB's image transformation routines (*maketform* and *imtransform*), which are accelerated by the IPP internally. The second approach uses a ray casting strategy to resample the misaligned projection data by rays that intersect the pixels of the aligned detector. The basic idea of this rebinning implementation can be summarized in three steps: (1) cast a ray from the source to each element of the perfectly aligned virtual detector, (2) find the intersection of the ray with the misaligned detector and (3) finally store the interpolated value from the intersection at the misaligned detector to the aligned virtual detector. A complete formulation of the ray casting strategy for virtual detector rebinning in conjunction with the matrix transformation framework can be found in Appendix E. This algorithm has been efficiently implemented using CUDA, so that the computations for each ray are processed in parallel.

The projection images of the SL phantom in Figure 4.6 compare both perspective rebinning strategies and show that the ray casting based approach results in a smaller error with respect to the aligned detector data. The differences between the aligned and rebinned projection image using ray casting are smaller than 0.154 in their absolute values, whereas as the errors for the perspective warping range from -0.307 up to 0.228. When comparing the whole projection set of the ray casting approach with

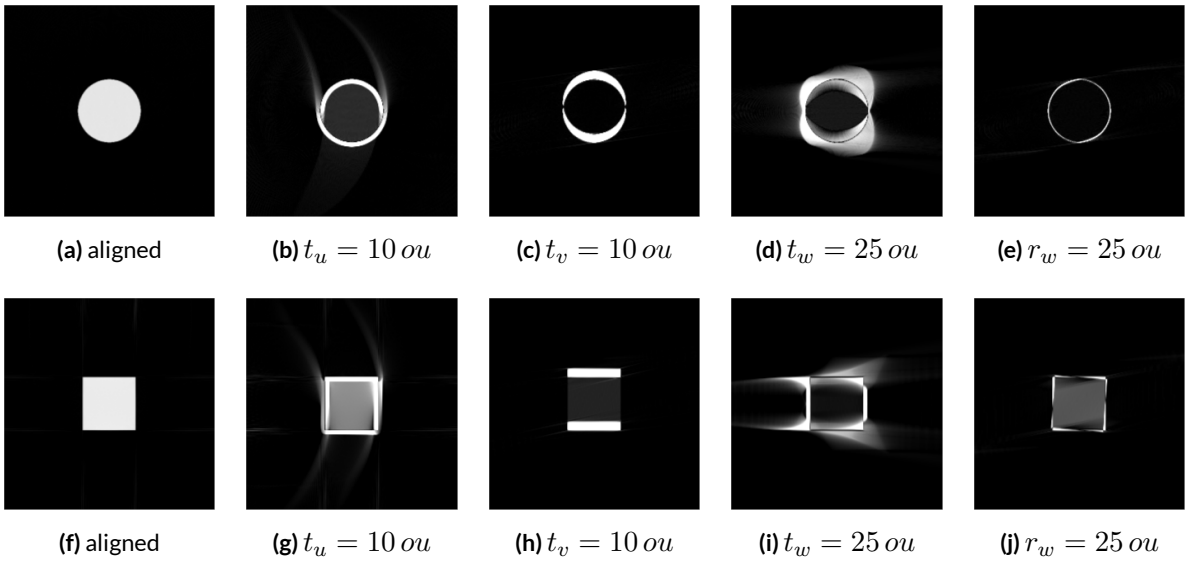
the aligned detector data the root-mean-square error (RMSE) is 5.95 %. In contrast to this, the RMSE for the warping-based rebinning is higher than 7.0 %. With other simulated dataset, different geometries and various constellations of the misalignment parameters similar results have been observed, so that in general the ray casting based perspective correction algorithm delivers a slightly superior projection image quality after rebinning. As consequence, the volumes reconstructed with this approach have a smaller deviation from the original image. Moreover, for the study in Figure 4.6, the rebinning of the whole projection sets using the ray casting approach took only 0.09 seconds, whereas the perspective warping took 0.461 seconds. Also for larger projection sets, a similar performance increase by up to a factor of 5 has been measured using the ray casting based rebinning. Because of its better overall computational performance, the ray casting algorithm will be used for all further experiments and discussions throughout this chapter.

Furthermore, to generate high quality rebinned projections with minimal artifacts from the misaligned detector data cubic B-spline interpolation (CI) provided by [163] has been integrated into the rebinning process. After the rebinning has been completed, Katsevich's algorithm can reconstruct the image from the virtually aligned data. The CI can also be enabled in the reprojection and in the backprojection process, but preliminarily experiments have shown no improvement of projected and reconstructed images in comparison to bilinear interpolation. Therefore, the use of the CI inside the projection operators did not show any superior calibration results and is therefore not further considered.

4.4 Results

Before the results of the calibration algorithm described above will be presented, the artifacts that the four different misalignment parameters (t_u, t_v, t_w and r_w) generate in the reconstructed image will be discussed. For C-CBCT, these effects have been described using simple test objects by Sun et al. [110]. The following study will also use simple test objects to demonstrate the misalignment artifacts for the H-CBCT case.

For this purpose, helical cone beam projections of a single sphere and a simple cube are generated. The system setup used for this simulation is composed of a set of parameters that resemble an average system geometry with respect to CBCT configurations that are used for the studies of real data in Section 4.4.2 (compare Table 4.2). The setup consists of a flat detector geometry with $N_u = 512$ columns and $N_v = 512$ rows each having a size of $\Delta u = \Delta v = 0.8$ mm. Hereby, the detector is placed at $D_{SP} = 600$ mm away from the X-ray source, which is rotating around the object of interest three times acquiring 1080 projections in total. The helical pitch is chosen to have an optimal value of $P = 50.56$ mm, so that the largest possible amount of data from the projections can be used during the reconstruction process (compare [113]). During this procedure, the volume that is computed with Katsevich's algorithm is placed with its center at a distance of $D_{SI} = 50.0$ mm from the source and has a size of $N = 256^3$ voxels with each having an edge length of $\Delta x = \Delta y = \Delta z = 0.1$ mm.



(a) aligned **(b)** $t_u = 10\,ou$ **(c)** $t_v = 10\,ou$ **(d)** $t_w = 25\,ou$ **(e)** $r_w = 25\,ou$

(f) aligned **(g)** $t_u = 10\,ou$ **(h)** $t_v = 10\,ou$ **(i)** $t_w = 25\,ou$ **(j)** $r_w = 25\,ou$

Figure 4.7: Perturbations introduced by the four detector mismanagement parameters: t_u, t_v, t_w and r_w. The top row shows the reconstruction from an aligned detector (a) and the absolute differences (b)-(e) between the reconstructions from aligned and misaligned detector data of a sphere. The bottom row shows the corresponding images for a cube. All reconstructed volumes in the first column have been normalized to $[0, 1]$ and for the difference images in the last four columns the grey value window has been set to $[0, 0.5]$.

The resulting perturbations of each misalignment parameter are displayed in Figure 4.7. The axial central slices in the first column of Figure 4.7 show the objects reconstructed from the aligned detector data. The last four columns show the artifacts introduced by the misalignment parameters. Figure 4.7b and 4.7g illustrate the absolute differences in the central slices of the reconstruction from the aligned detector and from a detector that has been horizontally shifted by 10 ou $= 6.05$ mm. The horizontal movement of the detector shrinks the sphere and the cube on one half and expands them on their other half, so that the largest deviations between the aligned and misaligned reconstructions occur at the border of the objects of interest. When visualizing these differences in 3D one can see that the edge where the shrinking changes over to the expansion of the object has a spiral shape. Similar artifacts, so-called "comet trails", have also been described by Hass and Faridani [94] while studying the effect of different implementations strategies for the derivative computation in Katsevich's algorithm. They showed that the effect of a horizontal detector shift of only half the width of a projection pixel can be recognized in the final reconstruction. These facts demonstrate how sensitive Katsevich's algorithm is for variations in the parameter t_u.

The absolute differences in central coronal slices between the reconstruction from an aligned and a vertically shifted detector ($t_v = 10$ ou $= 11.72$ mm) are displayed in Figure 4.7c and 4.7h. A vertical displacement of the detector essentially moves the object in the reconstructed image into the direction of the detector shift, so that the largest differences occur at the bottom and at the top of the object. Beside of this, minor wing shaped artifacts can be recognized at the lower and upper corners of the sphere and the cube. However, the overall shapes of the objects reconstructed from the misaligned

projections matches those of the objects computed from the aligned data. The main difference in the volumes introduced by the parameter t_v lies in the vertical positioning of the objects.

In contrast to the two parameters above, the parameter t_w has for the given system setup a relatively large OU with 1 ou $= 3.433$ mm. As a result, a change in its value has a smaller effect on the resulting reconstructed volume as the variations of t_u and t_w. The artifacts, which are introduced to the sphere and to the cube by a shift of the detector along its normal ($t_w = 25$ ou $= 85.83$ mm), are shown in the fourth column of Figure 4.7. The misalignment in t_w creates streaking artifacts mainly at the bottom and the top of the objects. The effect drops off towards the center of the object. This transition can have the effect that the final reconstructed object looks tapered. Note that large shifts, as the one has been used here to make the artifacts of t_w visible, are very unlikely to appear in practice, since for most CT scanners the distance D_{SP} can be measured very accurately and large deviations can be avoided by a precise manufacturing of the CT system.

The coronal central slices in Figure 4.7e and 4.7j show the differences between a reconstruction from an aligned detector and from a detector rotated by $r_w = 25$ ou $= 4.23°$ around its normal. On the one hand, a detector rotation introduces effects that are similar to the wing shaped artifacts like they have been shown for a misalignment of the parameter t_v. On the other hand, a rotation of the object itself can be noticed.

The above overview helps to categorize different types of detector misalignments in H-CBCT by looking at the reconstructed images, because the shape of the artifacts varies with the misalignment parameters. Additionally, this study can be employed as a rough visual guide to the strength of the influence that each detector parameter has on the final image. The major results from this study can be summarized as follows: (1) the parameter t_v mainly introduces a shift to the object, (2) the value of t_w only has a minor influence on reconstruction quality because of its relatively large OU and (3) the detector shift t_u and the rotation around the detector normal r_w introduce the most serve perturbations to the shape of the reconstructed object.

4.4.1 Simulated Data

The stability and robustness of the calibration method developed in Section 4.3 will be verified by simulations generating synthetic projection data during the following experiments. These studies will demonstrate the convergence of the parameter estimation process and that it approximates the correct solution. Moreover, the robustness of the algorithm will be illustrated by adding artificial noise to the projections. As commonly known, the X-ray photons are approximately following a Poisson noise distribution. However, for the practical simulation of projection data, it will be assumed the X-ray image acquisition procedure behaves according to a normal distribution when a large number of photons are involved into the process. Additionally, it can be observed during real data experiments (see [164]) that the standard deviation of the noise is not independent from the attenuation that the X-ray beam experiences. Consequently, the standard deviation is not constant over a whole projection image. Hence, the following simulations will be carried out using two different noise models based on

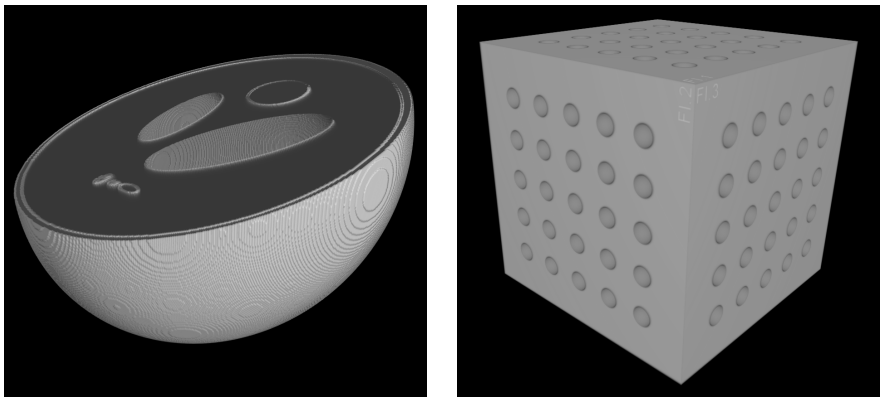
Table 4.2: Initial system setups for the calibration from simulated and real data

	Simulations	**Pen**	**TP09**	**Stent**	**KKW**	**Head&Neck**
Trajectory	Circular | Helical	Circular	Helical	Circular	Helical	Circular
Num. of Projections	1080	800	1370	600	1000	360
N_u	512	512	512	1848	512	512
N_v	512	512	512	1480	512	512
$N_{x,y,z}$	512	512	512	512	512	512
$\Delta_u\ [mm]$	0.800	0.800	0.800	0.127	0.400	0.8
$\Delta_v\ [mm]$	0.800	0.800	0.800	0.127	0.400	0.8
$\Delta_{x,y,z}\ [mm]$	0.047	0.030	0.030	0.030	0.030	0.48
$D_{SP}\ [mm]$	600.000	759.392	759.392	425.095	783.419	1536
$D_{SI}\ [mm]$	50.000	87.649	207.543	59.185	68.549	1000
incr. Z $[mm/proj.]$	0.000 | 0.140	0.000	0.139	0.000	0.025	0.000
incr. Angle $[°/proj.]$	0.333 | 1.000	0.450	0.450	0.600	0.360	1.000

Gaussian distributions. In the first model zero-mean Gaussian noise will be added to a projection pixel with value p, whereby the standard deviation was chosen to be signal-dependent (SIG) by attaining a value of $\sqrt{p}$. The assumption that the attenuation of the X-ray beam is approximately equal for all rays passing through the object of interest is made in the second model. Consequently, a constant (CON) additive Gaussian noise model is chosen to carry out the second run of the simulations, while a standard deviation of one tenth of the full projection intensity range is selected. The results presented in the following will show that the calibration technique works for both noise models.

The studies of the simulated data will be performed on a circular and on a helical scanning trajectory. For both setups the first columns of Table 4.2 summarizes the employed geometrical system configurations. To show that the novel calibration approach is able to deal with different types of geometrical shapes, two objects with differing structures have been selected: (1) the 3D modified SL phantom described first by Toft [165] and (2) an artificially constructed dataset of a calotte cube (Kugelkalottenwürfel) (KKW). The latter volume dataset has been generated by the voxelization of a polygonal model of the KKW, which corresponds to the one used in Section 3.3 and in [1] to compare Katsevich's reconstruction algorithm with the FDK method. Figure 4.8 shows a 3D visualization of both studied objects. The clipped volume rendering of the SL phantom reveals its internal structures and shows its plain spherical surface. In contrast to this, the KKW is internally solid, of cubic shape and has concave spherical calottes on the outside.

During the simulations, 20 sets of projections with differing random detector misalignments are generated for each combination of the trajectories, objects and noise levels. Therefore, the four misalignment parameters values were drawn independently and randomly from a uniform distribution. The scales of the misalignments were chosen, such that no parameter value fell outside the corresponding ranges listed in the second column of Table 4.3. For this reason, the absolute value of the three translation parameters never exceeded 1.5 cm. Furthermore, the magnitude of the rotation angle r_w was kept below $5.0°$. Then, to retrieve the values of the misalignment parameters from the



(a) (b)

Figure 4.8: The two objects that are used to simulate the projections. A volume rendering with a clipping plane through a central slice of the modified SL phantom is given in (a) and a full view of the KKW is displayed in (b).

simulated sets of projections the proposed calibration algorithm was applied twice. First it was used in conjunction with the 1D-MRS algorithm and afterwards with the L-BFGS optimizer. Thereafter, the resulting estimations of the parameter values were compared to their true initial values. Hence, the mean errors between the estimated and the reference values of the 20 simulation runs have been computed together with their corresponding standard deviations.

The following scheme summarizes the steps carried out for the evaluation of the calibration method on the simulated datasets:

- For each trajectory, for each object and for each noise level do the steps below
    1. Draw 20 random misalignment parameter constellations according to Table 4.3.
    2. Simulate 20 sets of projections from the generated parameter configurations
    3. Apply the calibration algorithm to each of the 20 simulations twice, once using 1D-MRS and once using L-BFGS
    4. For each optimizer compare the estimated parameter values to their reference values, by computing the mean and the standard deviation over the 20 results.

Table 4.3 presents a full overview of the results of the experiments with respect to the mean parameter deviations. Besides the comparison of the errors, the table gives a criterion for the precision of the calibration approach with respect to each parameter. For this purpose, OUs were used. Parameters for which the deviations between estimated and reference value were not larger than their corresponding OU have for any ray back-projected through the volume of interest a maximum perturbation that is approximately smaller than one voxel. Consequently and as discussed by Kingston et al. [112], the volume can be reconstructed with accuracy in the sub-voxel range if the estimation error of a misalignment parameter is not larger than half of the OU. For this reason, the criterion to judge the calibration precision of a single parameter is chosen to be half of 1 ou and is listed accordingly in the last column of Table 4.3.

Table 4.3: Results from simulations with and without noise

Parameter	Range	Unit	Algorithm	Noise	C-CBCT SL	C-CBCT KKW	H-CBCT SL	H-CBCT KKW	$^1/_2$ OU
t_u	$[-15, ..., 15]$	$[mm]$	1D-MRS	NONE	0.014 ± 0.015	0.011 ± 0.009	0.024 ± 0.018	0.018 ± 0.018	≤ 0.605
			L-BFGS	NONE	0.027 ± 0.024	0.028 ± 0.014	0.036 ± 0.025	0.048 ± 0.025	
			1D-MRS	CON	0.019 ± 0.020	0.010 ± 0.008	0.030 ± 0.026	0.017 ± 0.010	
			1D-MRS	SIG	0.028 ± 0.027	0.009 ± 0.006	0.031 ± 0.020	0.022 ± 0.016	
t_v	$[-15, ..., 15]$	$[mm]$	1D-MRS	NONE	0.337 ± 0.179	0.451 ± 0.344	0.455 ± 0.310	0.118 ± 0.126	≤ 1.172
			L-BFGS	NONE	0.566 ± 0.346	0.703 ± 0.146	0.588 ± 0.285	0.738 ± 0.325	
			1D-MRS	CON	0.397 ± 0.230	0.560 ± 0.307	0.378 ± 0.274	0.121 ± 0.075	
			1D-MRS	SIG	0.506 ± 0.283	0.643 ± 0.341	0.464 ± 0.274	0.120 ± 0.113	
t_w	$[-15, ..., 15]$	$[mm]$	1D-MRS	NONE	1.806 ± 0.327	0.685 ± 0.476	1.042 ± 0.041	0.353 ± 0.036	≤ 3.433
			L-BFGS	NONE	0.679 ± 0.797	1.522 ± 0.914	0.199 ± 0.111	0.229 ± 0.128	
			1D-MRS	CON	1.843 ± 0.603	0.927 ± 0.560	1.095 ± 0.025	0.393 ± 0.050	
			1D-MRS	SIG	1.765 ± 0.289	1.297 ± 1.050	1.127 ± 0.144	0.398 ± 0.065	
r_w	$[-5, ..., 5]$	$[°]$	1D-MRS	NONE	0.096 ± 0.055	0.045 ± 0.022	0.040 ± 0.013	0.037 ± 0.023	≤ 0.169
			L-BFGS	NONE	0.121 ± 0.024	0.027 ± 0.017	0.019 ± 0.022	0.022 ± 0.018	
			1D-MRS	CON	0.087 ± 0.052	0.045 ± 0.023	0.042 ± 0.027	0.044 ± 0.013	
			1D-MRS	SIG	0.087 ± 0.045	0.046 ± 0.025	0.060 ± 0.023	0.062 ± 0.011	

4.4.1.1 Calibration Accuracy

The errors made for the estimation of the parameter t_u are displayed in the first row of Table 4.3. Comparing these errors with the last column of the table demonstrates that the corresponding OU is an order of magnitude larger than the average deviations between the estimated and the true parameter value. Additionally, the self-calibration procedure was able to retrieve the horizontal detector shift t_u in the case of noisy projection data with an error consistently less or equal than 0.031 ± 0.020 mm. Consequently, as for the noiseless projections, these deviations are considerably lower than 0.605 mm, which is half of the OU of t_u. In general, the calibration results achieved with respect to t_u independently from the employed scanning trajectory are slight less precise for the SL phantom than for the KKW projection data. Moreover, the lowest average estimation error for t_u with 0.009 ± 0.006 mm was reached for the KKW dataset using the C-CBCT system setup while adding SIG noise during the simulation of the projections. With respect to the detector pixel width of $\Delta u = 0.8$ mm, it has been found that the precision of the calibrated horizontal detector shift t_u was 0.064 pixels in the worst case and 3.75×10^{-3} pixels in the best case.

Contrary to the translation of the detector along the horizontal u-axis, the vertical misalignment parameter t_v shows less accurate results, as it can be seen in the second row of Table 4.3. Nevertheless, for all simulations the mean calibration errors were smaller than half of the corresponding OU of 1.172 mm. Moreover and in accordance with t_u, the deviations between the true and estimated parameter values of t_v were never larger than the height of the detector elements, that is $\Delta v = 0.8$ mm. In this context, the reformulation of half of the OU in terms of the detector element size yields $(1.172\,\text{mm}/0.8\,\text{mm}) = 1.465$ pixels. From this value it becomes clear that a relatively large vertical translation of the detector along the v-axis is needed for the introduction of misalignment artifacts that create a perturbation larger than one voxel. With respect to the pixel height Δv, the largest deviation for t_v was 1.23 pixels and the smallest estimation errors was 0.057 pixels, such that for all simulations the estimation errors were smaller than the 1.465 pixels. A comparison of

the calibration result in relation to the KKW datasets showed that the parameter estimation of t_v from the noisy projection sets (SIG) using the circular scanning geometry resulted in the highest errors. Contrary to this, the smallest estimation errors have been registered for the noiseless projection data generated from the helical source path. For the latter case the deviations were never larger than 0.118 ± 0.126 mm.

A quite similar behavior can be observed for the calibration of the translation along the detector normal vector t_w: the estimation process with respect to t_w delivers for the helical scanning trajectory consistently lower calibration errors than for the circular system setup. Furthermore and similar to t_u and t_v, the achieved estimation errors of t_w are less than half of its OU, that is 3.433 mm. This result has been noticed independently from the type of the scanning trajectory, such that sub-voxel accuracy with respect to t_w was reaches for all helical and circular cases. Hereby, the parameter calibration procedure performed significantly less accurate for the projections of the SL phantom simulated using the circular geometry, while all other cases achieved more precise results. Accordingly, the simulations of the KKW dataset using the H-CBCT setup produced the lowest estimation errors for t_w. However, a comparison of the absolute calibration errors of t_w with those of the other two detector shifts reveals that the overall precision of t_w is inferior to t_u and t_v.

For the calibration of the rotational parameter r_w the errors showed a similar behavior and accuracy as those of the parameter t_u. Consequently, all parameter deviations for r_w were smaller than half of the corresponding OU of $0.169°$. The lowest calibration errors were obtained with values of $0.037 \pm 0.023°$ for the simulated noiseless projections of the KKW using the H-CBCT system setup. Contrary to this, the highest deviations by the estimation procedure were made with values of $0.096 \pm 0.055°$ for the noiseless simulations of the SL phantom using the circular source trajectory.

Comparing the calibration results obtained from the two employed noise models to each other shows that the estimation errors of the constant (CON) noise model are slightly smaller for almost every simulated projection set than the errors of the signal-dependent (SIG) noise model. However, for both models it has been observed that the deviations of the parameters were consistently smaller than half of their corresponding OUs. Similar observations have been made with respect to the calibration using the L-BFGS optimization scheme: the convergence was for all simulated cases correct, such that the true parameter constellation was approached, and sub-voxel accuracy was reached in the sense that the resulting estimation error of each parameter was lower than half of the OU. Additionally, the errors of the L-BFGS optimizer in relation to the detector translations t_u and t_v were in principle higher than those achieved of the 1D-MRS approach. On the contrary, for almost every case, in particular for the H-CBCT datasets, the L-BFGS optimization scheme reaches in relation to the detector shift along and the rotation around the w-axis a calibration accuracy superior to that achieved by the 1D-MRS. The exceptions are formed by the C-CBCT KKW dataset for the parameter t_w and the C-CBCT SL case with respect to r_w, where the 1D-MRS algorithm performed better than L-BFGS. Nevertheless, the L-BFGS optimizer was able to reach the smallest estimation errors for the parameters t_w and r_w from the helical projection sets. Hereby, the smallest errors of t_w and r_w were

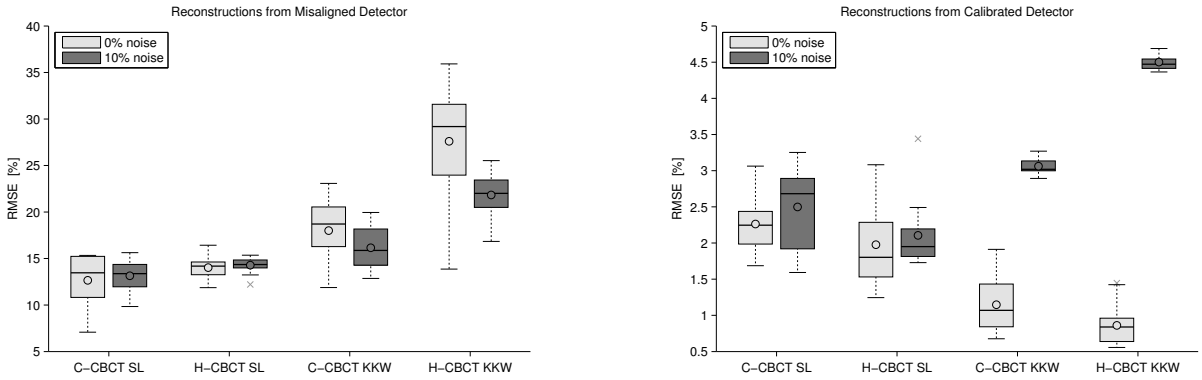


Figure 4.9: Box-Whisker-Plots visualizing the relationship between the RMSE and the trajectory, object of interest and the noise level. Each box was generated from 20 simulation and calibration runs. The reconstructed volumes have been normalized to the range $[0, 1]$ before the RMSE between the reconstruction from aligned and misaligned detector data (left) and between aligned and calibrated detector data (right) have been computed. Dots are the average RMSE, central black line is the median, lower and upper edges of the boxes are the 25% and the 75% quartiles, respectively. The fences show the range of the RMSE excluding mild outliers, which are marked by a cross.

0.199 ± 0.111 mm and $0.019 \pm 0.022°$, respectively, when calibrating the helical projections of the SL phantom.

The aforementioned observations can be generalized to derive some general characteristics about the calibration algorithm that apply to the studies made above and which have been fulfilled during preliminary evaluations: the here developed approach

1. achieves more precise calibration results for the KKW dataset than for the SL phantom,
2. estimates the parameters more accurately for the H-CBCT than for the C-CBCT case
3. and its accuracy is not considerably depending on noise in the initial projections.

4.4.1.2 Reconstruction Quality

By taking a closer look at the relationship between the reconstructions from the misaligned and the calibrated detector data by means of the RMSE the latter two properties can be reaffirmed. For this purpose, for each object, trajectory and noise-level the RMSE has been calculated in relation to the original phantom for the volume reconstructed from the aligned detector data and for the reconstruction computed from the misaligned projections. Note that all volumes have been normalized to the range $[0, 1]$ before the RMSE has been computed. The statistics about the RMSE from the 20 simulation runs are visualized in the box-and-whisker plots of Figure 4.9.

From this figure, it can be observed that random variations of the four misalignment parameter values within the intervals specified for the simulations (see Table 4.3), generate on average a reconstruction error larger than 12 % for the C-CBCT case and higher than 16 % for the H-CBCT geometry. The RMSE for uncalibrated reconstructions was never smaller than 5 %, whereby the smallest mean RMSEs were obtained for the SL phantom reconstruction from the circular trajectory and the largest

errors were reached for the KKW dataset reconstructed from the helical projection data (see left box plot of Figure 4.9). Furthermore, a look at the RMSEs calculated with respect to the calibrated projection data reveals that the estimation of the misalignment parameters achieves a reduction of reconstruction errors larger than 35 % to under 1 % on average for the H-CBCT KKW dataset. In general, the develop method is able to reduce the initial errors of the misaligned datasets, which were larger than 5 %, below this threshold. According to the second algorithmic property formulated above, for the noiseless case the calibrated reconstructions from the helical trajectory resulted consistently in lower RMSEs than those calculated from the circular scanning path. In contrast to this, the results in Figure 4.9 demonstrate the high sensitivity of Katsevich's exact helical FBP algorithm towards noise that increased the RMSE significantly. A main reason for this is differentiation step involved in the reconstruction process, which can amplify noisy data. A detailed discussion about this behavior can be found in [80, 93]. However, when comparing the uncalibrated and calibrated reconstructions from the noisy data for each case, the box plots show that the calibration algorithm is able to reduce the RMSE to fewer than 3 % for the SL phantom and to fewer than 5 % for the KKW dataset. This fact underlines that the algorithm is stable under the presence of noise and supports the third statement made above.

Contrary to these observations, the first characteristic of the calibration algorithm pointed out above cannot be reaffirmed by the sole interpretation of the RMSEs, although for the noiseless cases the errors of the KKW dataset were always smaller than 2 %, whereas those of the SL phantom were larger. Put another way, in the absence of noise the RMSEs of the calibrated SL reconstructions were always higher than those of the corresponding KKW simulations. Hereof, it has to be considered that the intrinsic features of both objects influence the RMSE differently, such that the errors observed for the noiseless cases not only depending on the accuracy of the calibration. Consequently, for an equal detector misalignment an object with many internal structures, as the SL phantom, will result in a larger RMSE as one that is mostly homogeneous, as the KKW. Therefore, due to the differing object compositions, imprecisions of the calibration algorithm will lead to a higher amount of artifacts (compare also with Figure 4.11) and thus to a larger RMSE for the SL phantom than for the KKW dataset. As a result, the RMSE depends on both, the calibration precision and the structure of the object of interest, and is therefore not suited to compare the results of the KKW with those of the SL datasets.

4.4.1.3 Runtime Performance

For the measurements of the runtime performance of the CBCT calibration algorithm a workstation personal computer (PC) equipped with a 2.80 GHz Intel Core i7 860 CPU and a NVIDIA GeForce GTX TITAN has been used. During each run of the calibration on the 20 noiseless simulations with the 1D-MRS and the L-BFGS approach the overall times of execution have been recorded, while the time spent in the most compute-intense subroutines has been tracked additionally. Note that to gain a better overview, operations that took less time during the calibration have been omitted from the

Table 4.4: Time performance of the calibration method

Functions	Unit	C-CBCT SL		C-CBCT KKW		H-CBCT SL		H-CBCT KKW	
		1D-MRS	L-BFGS	1D-MRS	L-BFGS	1D-MRS	L-BFGS	1D-MRS	L-BFGS
◇ Calibration	[min]	6.519	16.205	5.546	16.819	11.607	13.260	11.364	14.924
▷ Objective	[sec]	1.051	2.108	0.975	2.085	1.936	2.792	1.925	2.686
▷ Forward projection	[sec]	0.463	0.528	0.447	0.519	0.154	0.149	0.150	0.149
▷ Backprojection	[sec]	0.459	0.512	0.443	0.506	1.381	1.600	1.375	1.511
▷ Mutual Information	[sec]	0.168	0.278	0.132	0.274	0.163	0.256	0.159	0.230
Objective Func. calls		318	416	319	437	313	262	308	306

following presentation, since the overall performance of the procedure is only marginally influenced by these subroutines. After the measurements, the runtimes have been averaged over each of the 20 executions. Moreover, these time recordings have been conducted for both objects and trajectories. The resulting runtimes of the most time consuming task, i.e. function calls, are listed in Table 4.4, where the hierarchy of the calls (see Section 4.3.3 and Section 4.3.4) is visualized by the indentation of the tasks.

From the table it can be observed that the 1D-MRS approach takes for the calibration of a circular projection set between 5 and 7 minutes, while the parameter estimation for a helical dataset takes significantly longer, with about 11 minutes. One of the main reasons for this deviation is the differing execution times of calls to the similarity function. In the C-CBCT case a single objective function call needs about half the time required for the execution of the helical similarity function. This behavior in turn can be explained by the times spent in the forward and backprojection operators. On the one hand, the circular backprojection process using the FDK is considerably faster than the helical one using Katsevich's algorithm, which consists of more complex procedures than the FDK method (compare Chapter 3). On the other hand, for the helical case the forward projection operation outperforms the one of the circular simulations, since for some helical projections the scanned object is not fully inside the field of view of the detector, so that cast rays have not to be traced through the volume, terminate earlier and less time is spent during their computation. Nevertheless, the time measurements show that the latter effect is not strong enough to compensate the former one. In other words, the time spent in the helical backprojection is too long, even longer than 1 second on average, so that in accumulation with the shorter runtime of the helical forward projection the overall performance is inferior to that of the C-CBCT case.

The average number of calls to the objective function has also been recorded and is shown in the last row of Table 4.4. For all four executions of the 1D-MRS these number are almost equal. However, it can be noticed that the H-CBCT calibrations have a slightly reduced number of objective function calls. The reason for this is that for the helical setups more runs of the calibration procedure terminated earlier for some misalignment parameters, since they fulfilled the optimality criterion, as described in Section 4.3.3, at a coarser resolutional scale (compare with Figure 4.10). Consequently, these parameters were not optimized further and calls to the objective function were saved.

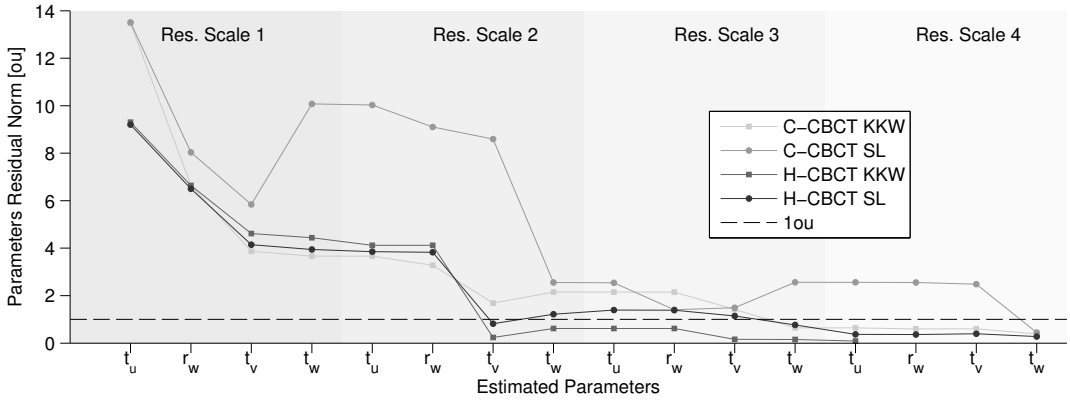


Figure 4.10: Convergence of the calibration method over multiple resolution scales. For each of the four datasets the residual norm of the parameters has been computed after the 1D scanning of each parameter at each scale finished. For easier interpretation the parameter values have been scaled by their corresponding OUs.

In contrast to the 1D-MRS approach, the calibration using the L-BFGS optimization scheme showed significantly longer times of execution for all studied objects and trajectories. For the calibration of projections from a circular trajectory the L-BFGS was even up to three times slower than the 1D-MRS, while for the helical scanning path the parameter optimization with L-BFGS took only few minutes longer than with the 1D grid search. As an explanation to the longer runtimes of L-BFGS, it has been observed in comparison to the 1D search strategy that the optimizer made fewer calls to the objective function on the coarser resolutional scales, whereas a higher number of calls have been registered at the finer more compute-intense resolutions. As a consequence, even the L-BFGS approach has a lower total number of objective function calls for the H-CBCT setup, it was not able to achieve the performance of the 1D-MRS optimizer proposed within this thesis.

4.4.1.4 Convergence

The convergence of the 1D-MRS calibration algorithm with respect to a single initial constellation of the misalignment parameters is visualized for each of the studied objects and trajectories in Figure 4.10. For the generation of the line plots the true values of the four detector parameters were set once randomly to $t_u = -7.889\,\text{mm}$, $t_v = -14.770\,\text{mm}$, $t_w = -8.781\,\text{mm}$ and $r_w = -4.005°$. With this parameter constellation, denoted in the following by $\mathcal{P}_{true}$, the misaligned projection sets were simulated for all four datasets, while no artificial noise was added during the generation process. Then the calibration procedure was run for each projection set. After each parameter within the set $\mathcal{P}_i$ had been determined by the optimization process in accordance with equation (4.14) on each resolutional scale the residual norm $|\hat{\mathcal{P}}_i - \hat{\mathcal{P}}_{true}|$ has been computed. Hereby, the sets $\hat{\mathcal{P}}_i$ and $\hat{\mathcal{P}}_{true}$ contain normalized parameters, such that each value has been divided by its corresponding OU. The lines in Figure 4.10 reflect the residual norms of these sets with respect to each dataset and are plotted against the parameter names.

The convergence plot demonstrates that the calibration technique converges stably for all four cases to the correct solution and shows that in the end the residual norm of the final parameter set lies below 1.0 ou. Moreover, it is shown that the optimization process is able to recover estimation errors from a previous resolution scale: even if a single parameter estimate is not correct on one resolution scale, its value is refined and approaches the true solution on the next finer scale. For example during the calibration on the C-CBCT SL dataset, the parameter estimate of t_w on the first scale increased the residual norm considerably, while its estimation on the second scale led to the correct solution. Note that similar observations in relation to the convergence have been made during the other calibration runs performed in the previous and following sections for the different studied datasets, whereby all optimizations converged to the correct parameter constellation.

A comparison of the two system geometries with respect to the convergence of the calibration algorithm reveals for most of the iterations that the residual norms of the C-CBCT cases are higher than those of the H-CBCT dataset. This observation underlines the behavior discussed in the previous sections (compare Table 4.3): the parameter optimization achieves for helical trajectories an accuracy that is superior to that of circular ones. Additionally, for the H-CBCT KKW dataset it can be noticed in Figure 4.10 that the calibration process terminates after 13 iterations, such that the parameters r_w, t_v and t_w are not optimized on the last scale. In contrast to this, all 16 iterations are needed by the 1D-MRS to calibrate the other three datasets. Nevertheless, these results are in accordance with the discussion on the objective function calls in one of the previous paragraphs, where the number of calls was lower for H-CBCT than for C-CBCT.

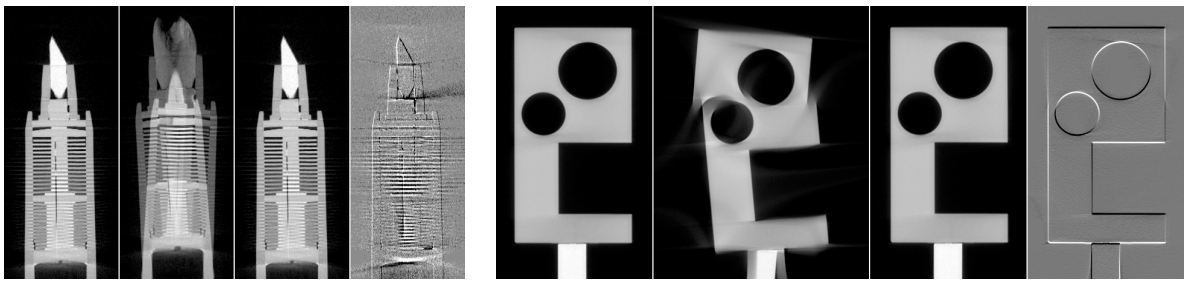
4.4.2 Real Data

The second set of experiments, which is used to verify the correctness of the calibration algorithm, uses projection data from various CT scanners with different geometric setups. The last four columns of Table 4.2 give an overview of the datasets that will be discussed in the following section.

The Pen and the TP09 were both scanned on a μCT scanner at the Fraunhofer Development Center for X-ray Technology (EZRT) in Fürth (Germany). The TP09 is an aluminum test-phantom consisting of 12 planes and 2 drill holes. It has been developed in [166] and used in recent studies on industrial CT metrology [167, 168]. For both of these datasets the CT-scanner was calibrated using the offline calibration procedure described in [169], so that the measurement error of the distances and of the detector position was minimized. As a result, the resolution limit, which has been computed as $R = 1/4 \cdot \Delta u \cdot D_{SI}/D_{SP}$, lies at $R_{Pen} = 0.023$ mm for the Pen dataset and at $R_{TP09} = 0.055$ mm for the TP09 dataset. For this reason, the true values of the misalignment parameters can be assumed to be all zero. Therefore, the values $t_u = t_v = t_w = 0.0$ mm and $r_w = 0.0°$ will be used as the references for the estimated geometric parameters. Note that since the distances and the detector size are the same for both datasets, their scan geometries have equal OUs, although the Pen has been scanned by a C-CBCT and the TP09 dataset with a helical setup. This can be validated by using the geome-

Table 4.5: Calibration results from real data

Parameter	Unit	Pen	TP09	OU	Stent	KKW	Head&Neck	OU
t_u	mm	0.059	0.115	≤ 0.635	5.665	0.574	0.044	≤ 1.209
t_v	mm	1.849×10^{-5}	0.480	≤ 1.483	-13.813	4.101	0.641	≤ 2.344
t_w	mm	0.087	1.036	≤ 5.500	-8.681	14.559	1.732	≤ 6.867
r_w	°	0.006	0.057	≤ 0.178	0.099	0.003	0.132	≤ 0.338

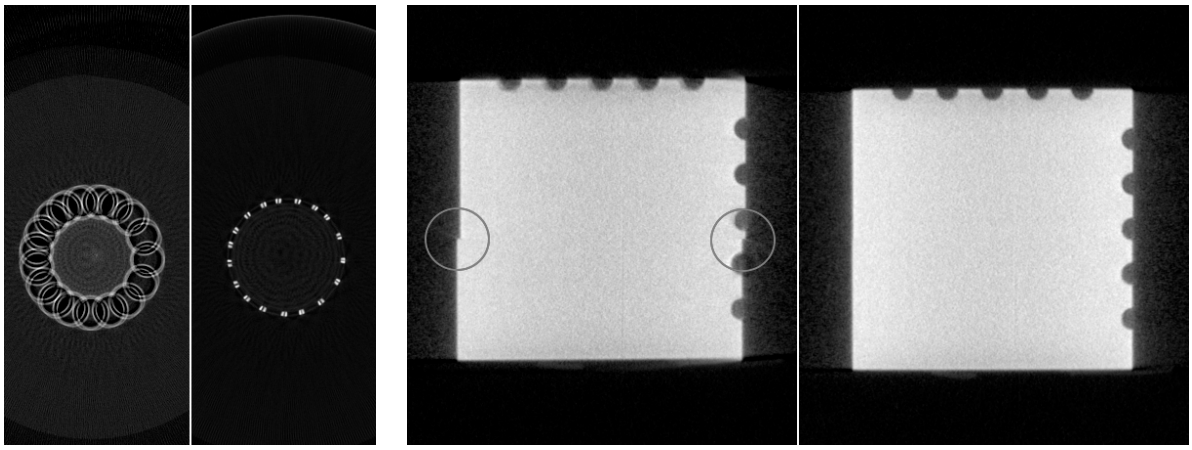


(a) (b)

Figure 4.11: Reconstruction of the Pen and the TP09 dataset. In (a), the leftmost image has been generated with all four misalignment parameters set to zero, in the second image the misalignment parameters were randomly chosen, the image right from the center shows the reconstruction from the projections with the corrections from the calibration algorithm and the rightmost image displays the signed differences between the first and the third image using a gray-value window of $[-0.08, 0.09]$. The images in (b) show the reconstructions of the TP09 dataset: parameters set to zero (left), misalignment parameters chosen randomly (left of center), after online calibration (right of center) and differences (window = $[-0.09, 0.13]$) between first and third image (right).

try from Table 4.2 in the formulas of Table 4.1. The resulting OUs are shown in the fifth column of Table 4.5.

To test the calibration algorithm on the two offline calibrated datasets, first a reconstruction is performed from the original projections without any alignment corrections. During the next reconstruction the values of the detector parameters were randomly chosen and finally the calibration method was applied to estimate these four parameters followed by a FBP using the calibrated system. For the Pen dataset the misalignment parameters were set to $t_u = 8.399\,\text{mm}$, $t_v = -14.587\,\text{mm}$, $t_w = 12.546\,\text{mm}$, $r_w = 2.905°$ and for the TP09 dataset they were fixed to $t_u = 9.578\,\text{mm}$, $t_v = -2.334\,\text{mm}$, $t_w = -3.255\,\text{mm}$, $r_w = 0.258°$. All reconstructed volumes were normalized to the range $[0, 1]$ and are shown in Figure 4.11. The leftmost images of each study show the reconstructions of the Pen and the TP09 from the projection data, which has been aligned by the offline calibration. The images second from the left demonstrate the strength of the influence that the detector misalignment has on the final reconstruction. For the helical TP09 dataset, the misalignment artifacts appear not only as double edges and as object deformations like for the circular scan of the Pen, additional wing-shaped stripes are introduced to the object as they have been discussed in the beginning of Section 4.4. The differences between the aligned and the calibrated reconstructions oc-



(a) (b)

Figure 4.12: Reconstructed slices of the Stent (a) and the KKW (b). The left images show the uncalibrated reconstructions and the right ones show the reconstruction using the results of the calibration algorithm. Because the slice in (a) is positioned slightly above the crossing of the stent wires, the double points in the right image are no miscalibration artifacts. In the uncalibrated reconstruction of the KKW the step artifacts have been marked (circled).

cur mainly at the edges of both objects, but do not significantly influence the image quality since the mean absolute difference for the Pen is 0.007 ± 0.011 and 0.002 ± 0.005 for the TP09.

When looking at the numerical results of the two datasets in Table 4.5, it can be seen that the absolute error (third and fourth column for the Pen and TP09, respectively) between the reference values and the estimated parameter values is always lower than the corresponding OU. The errors are even below half the OU, so that sub-voxel precision is reached after the calibration method has been applied. A comparison of both datasets reveals that for the Pen dataset the calibration delivered more precise results than for the TP09. For the vertical detector translation t_v the lowest absolute deviation of 1.849×10^{-5} mm is reached. A reason for this result can be that the Pen object has more internal structures with nonzero derivative along the z-axis than the mostly homogeneous TP09. These inhomogeneities help the MI-based registration approach to find the unknown parameters more accurately. The largest error relative to the OU is reached for the TP09 dataset, where the deviation of parameter t_v is almost one third of the OU. This deviation can also be seen in the difference image of Figure 4.11b where the largest differences occur at the horizontal edges as a result of the vertical detector shift.

In contrast to the EZRT datasets, the Stent and the KKW dataset were scanned on a μCT scanner, for which no data from any offline calibration technique was available. This means that for those two datasets no reference values for the misalignment parameters exist. Because of this, the visual quality of the reconstructed images is used to evaluate the correctness of the online calibration. The system setups for the Stent and the KKW scans are listed in Table 4.2. In Figure 4.12 the reconstructions from misaligned projections are compared to the ones that have been corrected by the calibration algorithm. The estimated values for the misalignment parameters are displayed in the sixth and sev-

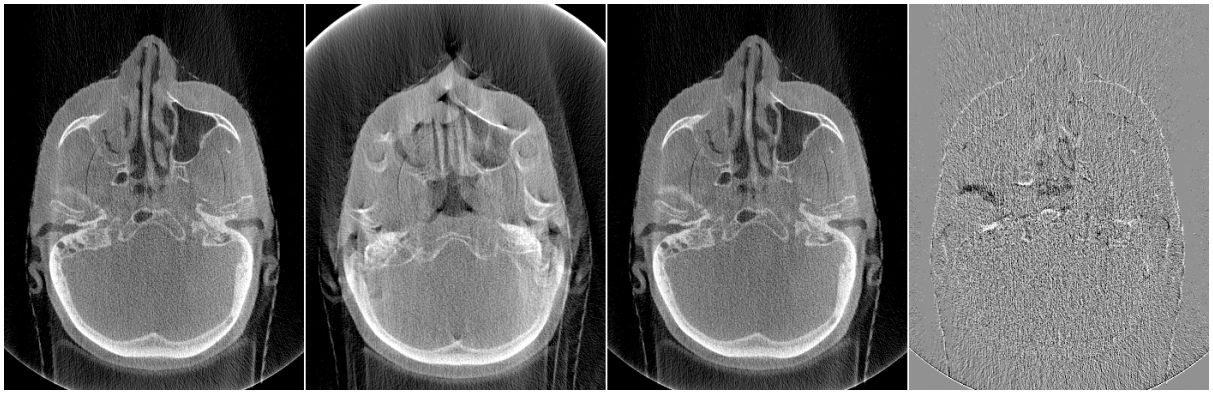


Figure 4.13: Reconstructed slices of the Head&Neck dataset. The images (window = $[0.0, 0.25]$) show the reconstructions from the offline calibrated (left), misaligned (second from the left) and from the calibrated projections (second from the right). The rightmost image displays the signed differences between the first and the third image using a gray-value window of $[-0.03, 0.03]$.

enth column of Table 4.5. For both datasets, the estimated rotation angle r_w was smaller than its corresponding optimal OU. According to this, it can be assumed that with respect to the rotation around the detector normal the CT scanner was well aligned during the acquisition. For the Stent dataset in Figure 4.12a the ringing artifacts are completely gone after the calibration succeeded and each wire of the stent is clearly visible and distinguishable from the others. The step artifacts in the KKW dataset in Figure 4.12b have also been completely removed by the alignment procedure, so that the calibrated KKW has a clean edge.

In addition to the datasets presented above, the online calibration method has been tested on a real clinical dataset. The Head&Neck dataset was acquired on an Elekta Synergy radiation therapy system equipped with an X-ray tube for on-line kV imaging. The system was calibrated by a flexmap using the offline method described in [170]. The images in Figure 4.13 were generated using the same procedure that was used for the EZRT datasets. For the leftmost image all parameters were assumed to be zero and for the reconstruction from the misaligned projections (second from the left image) the parameters were set to $t_u = -8.569\,\text{mm}, t_v = 12.780\,\text{mm}, t_w = 2.277\,\text{mm}$ and $r_w = -2.910°$. This parameter constellation was then estimated by the online calibration with the estimation errors shown in the second last column of Table 4.5. The lowest error was reached for the horizontal detector shift t_u with $0.044\,\text{mm}$. As it has been observed for the other datasets, the errors increase with larger OUs, but stay below half of an OU, so that sub-voxel accuracy was achieved for the calibrated reconstruction. The images in Figure 4.13 demonstrate again that the calibration method is able to recover the calibration parameters (third image) from a setup with a relatively large misalignment (second image). At the same time the errors (rightmost image) between the aligned and the calibrated reconstruction were smaller than 3 % in their absolute value.

4.5 Discussion

4.5.1 Robustness and Accuracy

The previously conducted experiments demonstrate with their results that the novel registration-based calibration technique converges against the true parameter constellation. This behavior has been verified for various system setups and with different objects of interest, whereby simulated and real data has been studied on circular and helical scanning trajectories. Hereby, for the different reconstruction techniques, i.e. the FDK and Katsevich's method, that have been used with respect to the C-CBCT and H-CBCT geometries, respectively, it has been observed that the final calibration error within the estimated parameters are higher for the circular scanning path than for calculations on the corresponding the helical trajectory. An explanation for this effect can be given by the accuracy that the reconstruction methods that are employed during the calibration process are able to achieve for the intermediately updated volume (compare with Section 4.3.3). Since the implementation of Katsevich's algorithm is based on a theoretically exact inversion formula for H-CBCT (see Section 3.1.2) and the method of FDK can only compute an approximate solution to the reconstruction problem for the circular trajectory (see Section 3.1.1), the reconstructed volumes and thus the simulated reprojections are more accurate for the helical case. In general, the assumption can be made that a more precise reconstruction technique in combination with the suggested calibration algorithm will deliver superior parameter estimates. As a consequence, the reduction of artifacts that are inherent in the reconstruction algorithm, especially in the CBCT case, will likely result in an improvement of the precision of the calibration procedure and needs further attention in future studies.

Additionally, the results obtained from the simulation experiments revealed that the calibration method is able to achieve sub-voxel precision for the parameters t_u, t_v, t_w and r_w. This is true for cases where the originally simulated projections were contaminated by artificial noise and where the detector before calibration was misaligned, such that the true values of the four parameters were an order of magnitude larger than their corresponding OUs. Moreover, it has been observed that the parameter t_u was estimated more accurately than the two other translation parameters, t_v and t_w. This behavior is caused by the different impacts of the artifacts, which these three translations can introduce during the reconstruction process if they are not calibrated. For example, a vertical shift of the detector, as it has been demonstrated in the beginning of Section 4.4 within Figure 4.7c and Figure 4.7h, in principle only translates the object of interest in relation to the reconstructed volume, but does not create a larger amount of additional artifacts. Furthermore, a shift of the detector position along the w-axis by t_w results in a change of the source-to-detector distance and therefore modifies mainly the magnification, i.e. the effective size, of the studied object within the volume. As a result, variations of the parameters t_v and t_w have only a minor influence on the calculated reprojections and affect the objective function given by equation (4.14) accordingly. In contrast to this, it has also been shown in Section 4.4 that a horizontal shift of the detector by t_u introduced serve artifact into reconstructed volumes. This effect is underlined by the fact that the OUs of the two translation parameters

t_v and t_w are significantly larger than that of t_u. Therefore, it can be concluded that once the artifacts generated through additional influences, e.g. noise, sampling, discretization and interpolation, on the reprojection process become larger than the perturbations caused by t_v and t_w, then the MI-driven registration procedure will become unable to determine the correct parameter constellation.

An even more general formulation is suggested by the behavior observed above:

> A change in the value of a misalignment parameter will have a larger effect on the reconstructed volume when it's corresponding OU is relatively small. Vice versa, for parameters with larger OUs a similar variation will introduce fewer perturbations. If for the latter case the effect on the volume decreases, the objective function will no longer be steered by the misalignment parameter and can become dominated by the inaccuracies of the reprojection procedure. Consequently, the parameters are unlikely to be estimated correctly by the calibration technique.

As a result, it can be assumed that there exists a boundary up to which the calibration algorithm will work precisely and that this threshold depends on the OU of each parameter. However, such extreme situation did not occur in any of the conducted experiments, so that the actual parameter constellations were determined successfully. In any case, further studies to determine the factors that limit the parameter estimation process will have to be performed.

From the discussion above, it becomes evident that further studies have to be conducted to determine the influences which additional system parameters, such as the two out-of-plane rotations r_u and r_v, have on the reconstructions from misaligned projection data, on the similarity function and on the calibration procedure. By such means, a successful integration of these parameters into the current estimation algorithm can be guaranteed. In this context, the definition of OUs can provide reasonable indicators for the order in which the additional parameters can be arranged during the optimization process with respect to the already integrated ones (compare with Section 4.3.3). Nevertheless, it has to be considered that the selected reconstruction and projection generation algorithms can produce artifacts that superimpose the effects introduced by the calibration parameter changes. For this reason, further simulations that quantify these coherences are inevitable, whereby the inclusion of exact analytical and approximate reconstruction techniques into these studies could lead to further improvements.

4.5.2 Comparison to other Methods

The comparison presented in the following section will highlight the differences and common aspects of the currently existing registration-based online calibration methods. Therefore, the advantages of the method that has been described in Section 4.3 will be contrasted to the strategies proposed by Wein et al. [132] and Sasov et al. [4], which are also using registration approaches. Principally, online calibration methods for CBCT that are based on a 2D-3D registration framework come with several benefits:

- Image registration is a well-understood field of research. Many similarity metrics, optimizers, forward projectors and samplers have been suggested for registration purposes and provide a well-documented basis for further developments in the area of registration-based self-calibration approaches (see Section 4.2.3).

- There exist several full-featured open-source registration frameworks[1] that can be used as a foundation for the implementation of calibration methods (see [140] and [153]).

- The implementation of the 2D-3D matching is straightforward and can be realized in a projection-wise manner. Therefore, a calibration algorithm can be easily mapped to different scanning trajectories and to a projection-dependent calibration scheme.

- From the implementation perspective, additional system parameters can be added to the framework relative uncomplicated, although for achieving reliable calibration results further work is necessary (see Section 4.5.1).

In contrast to the here presented technique, which uses the MI, Sasov et al. [4] has compared the sum of squared differences (SQD) with the NCC in an iterative FDK approach, whereas the method described by Wein et al. [132] employs a modified ordered subsets SIRT (OS-SIRT) minimization of the sum of absolute differences (SAD). However, in Figure 4.3 it has been shown that the SAD can lead to incorrect estimates for the misalignment parameters when minimizing the reprojection error, because the resulting objective function showed multiple local optima and had its global optimum not always at the true parameter value. Furthermore, it has been shown that the methods of Sasov et al. and Wein et al. work stable for C-CBCT achieving precise calibration results, but a verification that any of these approaches can be successfully applied to the case of helical tomography is still unresolved. On the contrary, for the self-calibration method derived within this thesis it has been demonstrated during the experiments in Section 4.4 that an application to circular and helical scanning paths is possible and stable. In addition to it, the proposed method could be extend to handle more complex scanning trajectories, whereby it has been shown that the calibration accuracy can be improved by employing more precise reconstruction techniques (see Section 4.4 and Section 4.5.1). For this reason, the calibration algorithm should be well applicable to classes of trajectories for which a theoretically exact reconstruction algorithm exist.

Furthermore, the use of the Amoeba direct search strategy can lead to the case that Wein's algorithm is trapped inside local minima. As it has been discussed in Section 4.3.3 and shown in Figure 4.3, local optima are caused by inaccuracies during the reprojection process and can influence the behavior of the similarity function (also compare Section 4.5.1). In the case of the here developed method, these local extrema are studiously avoided by the 1D-MRS technique, which uses the described combination of sampling and function fitting to overcome the problem. By this means the calibration procedure reaches a reconstruction precision that is up to the sub-voxel scale exact. These

[1] For example: the *elastix* toolbox available from elastix.isi.uu.nl, the *Orfeo* Toolbox at www.orfeo-toolbox.org and the *Plastimatch* software, which can be found under plastimatch.org

results are comparable to those reached by current state-of-the are offline calibration methods (see Section 4.2.2.1). However, a detail numerical comparison of the here proposed method and the approaches of Sasov et al. and Wein et al. with offline calibration techniques has not been conducted so far. Nevertheless, the results from Section 4.4 supplement those of [132] and [4] in the sense that they have all demonstrated that a CBCT calibration based on the optimizations of a reprojection and alignment procedure can be successfully used to estimate detector parameters with relatively large initial misalignments. Moreover, the experiments in Section 4.4.1 on simulated data have proven that such a scheme can be robust against projection noise.

Contrary to the calibration method constructed in Section 4.3, the method developed by Wein et al. [132] allows the determination of an additional parameter that represents a shearing of the detector. Additionally, their method is able to carry out a per projection calibration of misalignments with respect to the acquisition angle. To incorporate such a projection-wise optimization into the MI-based calibration framework, such that it becomes able to estimate a set of parameters that affect every individual frame, there exist in principle two strategies :

Average and Refine: In a first run the proposed calibration algorithm can be used, exactly as in the previous sections, to estimate an average value for each projection-dependent misalignment parameter. After the mean parameter values have been determined an appropriate 2D-3D registration approach can be applied in a second phase, such that each projection is aligned individually with the volume of interest that has been reconstructed during the first step.

Refine directly: All misalignment parameters, frame-dependent and frame-independent, could be estimated in a single run of a large-scale optimization approach, like the Levenberg-Marquardt algorithm. Note that a similar technique using a simplex optimizer has been introduced and successfully applied for medical flat-panel CT by Dennerlein and Jerebko [171].

To make both strategies computationally feasible, it has to be remembered that the multi-resolution and the stochastic sampling techniques could be optimized even further. Additionally, a combination of both approaches could lead to a divide-and-conquer optimization strategy, which could be formulated as follows:

Divide and Conquer: During a first run the frame-independent and the average values of all frame-dependent parameters are determined, as in the "Average and Refine" approach. Then the set of projections is split in two parts and the parameter estimation is repeated for both projection sets. The splitting and optimization process is carried on until all misalignment parameters have been calculated for each individual frame. At some point during this process a switch of the employed optimizers, from a small-scale to a large-scale optimization scheme (e.g. from the 1D-MRS to the L-BFGS or to the Levenberg-Marquardt algorithm), could become advantageous.

In contrast to the employed optimization approaches, the here developed calibration algorithm is comparable in terms of runtime performance to that of Wein et al.. The 1D-MRS method takes between 5 and 12 minutes to determine the misalignment parameters depending on the dataset, whereas Wein et al.'s OS-SIRT approach requires in dependence of the number of estimated parameters 1 to a few minutes to execute a full calibration run. Note that the stochastic sampling has led to a significant speedup of the forward projection operator and of the calculation of the overall objective function (see Section 4.3.3). Up to now, the backprojection operator has not been optimized by means of a sparser sampling, although this could additionally improve the runtime performance of the calibration algorithm. Moreover, a further decrease of the number of random rays cast in the forward projector would speed up the method even more. Nevertheless, to guarantee a stable and robust calibration in conjunction with the aforementioned optimizations additional studies will be necessary.

In contrast to registration-based calibration, the online method presented by Kingston et al. [111, 112] is based on an auto-focus approach maximizing the sharpness in several dedicated slices of the reconstructed image to estimate a set of misalignment parameters. As the method presented in this thesis, their technique has been demonstrated to work on circular [112] and on helical trajectories [113] using Katsevich's algorithm. Additionally, it has been shown that their method is robust against projection noise with similar properties as the method presented in this thesis. On the contrary, the auto-focus methods has lower runtimes than the 1D-MRS since it only operates on several single slices and can perform a complete calibration faster than a full reconstruction of the dataset.

Nevertheless, the slice selection can become one of the main bottlenecks of Kingston's method, since it may not work correctly if the object of interest does not reside or too little information about the object is contained within the chosen slices. Theoretically their method can also fail if a misalignment parameter does not influence the sharpness in these slices. This can be the case for a vertical detector shift as it has been described in the previous section. Like discussed above, a registration-based approach can also lead to wrong calibration results in those cases, but since the full volume is used to compute the reprojection and to perform the 2D-3D matching it will likely act more robust than the auto-focus computations on single slices. Another drawback of Kingston's method is that for the estimation of parameters per projection more complex models for the objective function would be needed (see [112]) in comparison to those that have been proposed for registration-based approaches. However, a full comparison of auto-focus and registration-based calibration techniques on similar datasets and geometries is beyond the scope of this thesis, but needs further attention in future studies.

In comparison to offline calibration methods, the here presented self-calibration procedure does not require an additional scan of a specifically designed calibration phantom. Consequently, no additional time and effort have to be invested, as the online approach computes the detector parameters directly from the measured projection sets. However, it has to be noticed that without the information from such a reference scan any online calibration technique can in the best case only allow for the determination of relative geometric parameters. For the estimation of absolute parameters a calibration phantom scan is inevitable, such that for online methods a global scaling factor remains

undetermined. This factor can only be computed by a suitable offline calibration approach or by fitting the reconstructed object of interest against a reference template, like it has been done in Section 3.2.4 (compare with [1]). Having determined the global scaling by such means, the registration-based 1D-MRS calibration technique can be employed in industrial CBCT applications, e.g. for NDT and CT metrology, to gain precise quantitative measurements.

4.5.3 Performance optimization

The combination of native code with the MATLAB scripting language allows rapid changes of the here discussed method on the one hand, but comes with certain computational costs on the other hand. Additional overhead is accepted when calling the C-routines from MATLAB and the copying of projection and volume data to and from the GPU has to be carried out more frequently than in a pure native implementation. Therefore, an overall increase in performance of the calibration procedure can be expected, when porting it to a pure C++/CUDA implementation.

In addition to this and since the stochastic sampling approach has been used for the forward projection operator, there is still room for further optimization. Future simulations will have to show how the stochastic sampling can be improved by the further reduction of the amount of cast rays to optimize the overall performance of the calibration process. Additionally, the integration of a stochastic backprojection operator may improve the performance of the algorithm even further.

As regards intellectual work it remains a fact, indeed, that great decisions in the realm of thought and momentous discoveries and solutions of problems are only possible to an individual, working in solitude.

Sigmund Freud

5

Filter Optimization for Limited-Angle CT

The focus of the previous chapters was on the analysis and the development of a geometrical correction method for circular and helical cone-beam computed tomography (CBCT) using an optimization strategy. In addition to this, the following chapter will introduce, evaluate and discuss a novel approach for the design of filters, which can be used in limited-angle CBCT applications. The basis of the here presented method is formulated in the framework of the Approximate Inverse (AI), which uses a so-called reconstruction kernel to compute a regularized solution to a given inverse problem. For the case of the here derived filter optimization approach it will be shown that this reconstruction kernel is view-dependent, such that it is different for each projection angle. The reconstruction kernel will be computed using a numerical optimization technique, i.e. the simultaneous iterative reconstruction technique (SIRT) algorithm. For the use of this kernel in a filtered backprojection (FBP)-type reconstruction method an intermediate filter design step will generate a set of 2D filters by the application of a window function to the kernel. The resulting filters will have the important properties that they are independent of the acquired projection data. Therefore, a set of filters needs to be calculated for a given system geometry only once and can be reused on all datasets scanned with the same setup. The novel method will be called angle-optimized FBP (AO-FBP). The here developed filter optimization approach has also been described by the author in [3], even though the here given presentation is intended to provide a deeper insight. Consequently, figures and tables depicted during the derivation of Section 5.3 and in the experiments of Section 5.4 are taken over or are adapted from this publication.

Moreover, the developed filter optimization strategy is so far the only approach which combines the AI with SIRT in the area of limited-angle CBCT. Consequently, it is not derived from an analyt-

ical inversion scheme and does not rely on the existence of such a formula. In comparison to other filter optimization methods [172, 173], the approach does require additional measurements of system intrinsic parameters estimated from the scan of a wire-phantom. Nevertheless, the reconstruction results achieved with the here derived filtering strategy are better than those of standard FBP methods. Furthermore, the derivation of the filtering approach by means of the AI suggests a general principle, which can be used to construct filters in the same way for other scanning trajectories and system setups. On top of this, the AI allows the calculation of more sophisticated filters that can be used to implement image reconstruction and analysis in a single step.

The experiments conducted in this chapter will deal with simulated and real datasets to verify the correctness and to examine the quality of the filter optimization approach. The comparison of AO-FBP with standard FBP, i.e. the Feldkamp, Davis and Kress (FDK) method, and SIRT with respect to the reconstruction quality will reveal that the optimized filters result in a lower root-mean-square error (RMSE) and a higher contrast-to-noise ratio (CNR) than the standard filtering used by the FDK algorithm. Additionally, an analysis of the three algorithms using the artifact-spread function (ASF) will demonstrate that AO-FBP can be used to improve the image quality perpendicular to the in-focus plane, i.e. the central slice along the z-axis, by the reduction of ghosting artifacts in the final reconstructions when the angular range is low.

5.1 Motivation

Limited-angle computed tomography (CT) has basically two application areas: (1) tomosynthesis, which is used particularly in medicine for breast cancer diagnosis (see [174] for an overview) and (2) laminography for the quick inspection of large and flat industrial workpieces within the field of non-destructive testing (NDT) [175–177]. The main goal of the developments in digital breast tomosynthesis (DBT) applications is the detection of different kinds of lesions at the lowest possible radiation dosage. Therefore, the number and the angular range of projections has to be limited such that an optimal distribution of X-source positions [178–180] yields a reasonable compromise between X-ray radiation delivery and the detectability of microcalcifications. In contrast to tomosynthesis, the developments in the area of laminography aim to the realization of in-production non-destructive analysis of the assembled products. Hereby, the inspection is carried out either right before the assembly process, while considering every single complement of the final workpiece, or right after the construction with an analysis of the resulting product itself. Within this context, quality measurements are conducted either on a per sample basis, where only few items are drawn and tested from the overall production, or by an inline-CT inspection of every product directly after its assembly. For the scan and analysis of every produced item, the scan and reconstruction times are crucial parameters, which need to be kept at a minimum to make inline-NDT feasible. In addition to this, the geometric dimensions of the workpieces can place high demands on the mechanical setup of the CT scanner. As a consequence, the requirements of laminography only allow the acquisition of a limited number of projections from a restricted range of X-ray source positions. For example, Dobbins III

and Godfrey [174] have proposed a limited-angle CT setup for the scanning and the analysis of large flat workpieces where the source is located above the assembly line and the detector is right beneath the conveyor, such that the objects of interest are translated past the fixed scanner. In summary, the medical and industrial limited-angle CT applications end up with similar restrictions on the projection acquisition, although the former ones are motivated by optimal dose-utilization and the latter ones try to reduce the overall measurement times. However, one has to consider that in the industrial area the amount of projection noise can be greatly reduced by the increase of the X-ray tube current, at least if the inspected item is unaffected by the exposed radiation. On the contrary, highly noisy projections due to a dose reduction can lead to poor image quality in the area of tomosynthesis. Nevertheless, for both application areas the final reconstruction quality requires to achieve a certain accuracy, such that reasonable, precise and reliable medical diagnosis or industrial measurements can be obtained.

The main problem in limited-angle CT is the decrease in image quality due to the data insufficiency caused by the incomplete angular sampling, which leaves large areas of the frequency space unmeasured. Over the recent years, many researchers (see Section 5.2.1) have proposed methods to overcome this problem and to improve the image quality of limited-angle tomography. On one side, many iterative reconstruction techniques have been developed, but as their computational costs are higher than those of FBP methods they are not widely used in practical applications [173, 181, 182]. In particular their use in industrial scenarios with high resolution projection data is fairly limited. On the other side, FBP techniques have been adopted to the limited-angle case by filter modifications and different weighting schemes. The main advantages of a FBP algorithm in comparison to iterative methods is that it can be realized more efficiently on current multi-core hardware, such that real-time reconstructions become possible [24]. However, it has been demonstrated (see [173, 174]) that the resulting image quality of iterative approaches is superior to the one achieved by FBP approaches. To overcome this drawback, Kunze et al. [172] have proposed a method that iteratively constructs an optimal view-dependent set of projection filters from modulation transfer function (MTF) measurements of a limited-angle CT scanner. Their filters can be used in a FBP-type algorithm, whereby they achieve an image quality comparable to that of iterative methods and superior to that of standard FBP. However, their approach needs a separate scan of a wire phantom carried out on the CT system of interest. The analytical method of Nielsen et al. [183] does not require such a reference scan to compute a set of object independent filters, but reaches similar results as Kunze's method. However, none of these two filter optimization methods have been applied to an industrial CT setup so far.

The filter optimization process developed within this chapter will allow the computation of a set of object independent projection-wise 2D filters. These filters can be precomputed for different geometric setups, such that various angle configurations can be addressed and such that they can be reused for the reconstruction of different projections scanned with the same configuration. Moreover, the resulting projection filters are general in the sense that they do not require any prior knowledge about scanned object of interest. The foundation of the here derived method is formed by the theory of the Approximate Inverse (AI), introduced by Louis [184] and first used for filter design in 3D CBCT

in [185]. This theoretical basis is employed to numerically approximate an optimal set of projection-dependent shift-invariant filters for a given geometrical system configuration by the use of simultaneous iterative reconstruction technique (SIRT). Contrary to the filter optimization strategies used by Kunze et al. [172], Mertelmeier et al. [186] and Ludwig et al. [173], the here developed approach does not require any additional reference measurements or a priori knowledge for the determination of system intrinsic parameters of the CT scanner (e.g. impulse response and MTF). Furthermore, since the algorithm approximates the filters for a given system geometry numerical by an iterative reconstruction technique, it does not rely on the existence of an analytical inversion formula. Additionally, the computed set of optimized filters can be used in a FBP-type algorithm, which can be easily realized by replacing the standard filter of the FDK method (see Figure 2.2) with the calculated 2D projection-wise filters (see Section 5.3.4). The experiments conducted with simulated and real data will demonstrate the resulting set of optimized filters can reach an overall image quality, which is comparable to current state-of-the-art iterative reconstruction techniques. In comparison to the FDK algorithm [12] the here developed optimized FBP method shows a superior reconstruction quality with a significant improvement in relation to image contrast and artifact reduction. On top of all this, the here presented procedure for the construction of the optimized filters can be employed as a general scheme for filter design in CBCT. It can be used to construct more complex and application specific filters for other types of system setups with different scanning trajectories. For example, Louis and Weber [187] has recently incorporated image reconstruction and analysis into a single set of filters derived from the AI.

This chapter is structured as follows. In Section 5.2 the related work that has been conducted on limited-angle tomography and the AI will be described. Section 5.3 will introduce the new approach for filter design and present its incorporation into a FBP algorithm. In Section 5.4 the conducted experiments will be described and their result will be presented and visualized. Finally, Section 5.5 will discuss the results and a conclusion with perspectives on future work will be given in Chapter 6.

5.2 Related Work

5.2.1 Reconstruction Methods

As mentioned above, tomosynthesis and laminography applications only incorporate a limited angular range and a limited number of projections into the reconstruction process. Consequently, information about the object of interest is unmeasured and parts of the Fourier space are left unsampled. For this reason the inverse problem in limited-angle CT is highly ill-conditioned [188] and the inversion of the impulse response of the system is problematic, as described in [174]. However, many reconstruction algorithms have been suggested in literature to overcome this problem. An overview of these techniques can be found in the reviews of Dobbins III and Godfrey [174], Sechopoulos [189] and Males et al. [190]. In principle, the reconstruction algorithms proposed for limited-angle to-

mography can be categorized into three groups: (1) FBP methods, (2) deblurring techniques and (3) iterative reconstruction approaches.

The first group of algorithms contains FBP methods and their extensions. For linear limited-angle tomography Grant [191] proposed in 1972 the traditional shift-and-add algorithm (SAA) method, which computes reconstructed planes through the object of interest by a simple backprojection operation. Niklason et al. [192] have extended this fundamental method for a tomosynthesis geometry where the X-ray source moves on an arc above a stationary detector. To facilitate the selection of filters Edholm et al. [193] suggested the so-called Ectomography. Based on the observations made with respect to the undersampled frequency space in the case of limited-angle CT, their technique constructed inverse projection filters such that a constant sampling density is achieved in the 3D Fourier space of the reconstructed volume of interest. Further investigations of inverse filtering techniques for the development of improved FBP methods have been conducted by Matsuo et al. [194] and led to a reduction of the artifact spread with respect to planes outside the focused slice. Moreover, the attempts of Lauritsch and Härer [195] to enforce a uniform depth response over a wider range of frequencies and the modifications of the blurring function proposed by Stevens et al. [196] resulted in even further improvements of the reconstructed images.

The angular range from which the projection images are acquired and the total number of sampled projections are two of the major differences between geometric setups of tomosynthesis devices. For this reason, Zhao and Zhao [197] have developed a cascaded linear system model to study the effects of the X-ray source position distribution and the influences of different reconstruction filters on the presampling MTF and on the noise power spectrum (NPS) using a computer simulation platform Zhou et al. [198]. By the use of the linear system model, the investigations conducted by Hu et al. [178] allowed a quantitative prediction of the 3D point spread function (PSF) of the CT scanner. At the same time, they were able to reduce image artifacts caused by microcalcifications to achieve an increased detection rate of lesions. Additionally, Zhao et al. [199] have combined a slice thickness (ST) filter with a Hanning window function to limit high-frequency components along the transaxial direction. As a result, they greatly reduced noise aliasing, optimized the 3D detective quantum efficiency (DQE) and significantly improved the detection performance of low-contrast, large-area mass lesions. Furthermore, Mertelmeier et al. [186] and Orman et al. [200] developed and analyzed a modification of the standard FBP technique, which has been used in [201] to demonstrate that the visibility of low-frequency image features can be considerably improved by an increase of the angular sampling range. By this means, their method also enhanced the resolution of the reconstructed image along the transaxial direction.

Another optimized FBP reconstruction technique has been developed by Kunze et al. [172]. For a given geometric system setup, their approach computes angle-dependent filters, which can be used in a FBP-type algorithm. The filters are calculated numerically from a reference scan of a wire phantom using a reformulation of the SIRT algorithm, which approximates the impulse response of the system by updates in the projection space. This is the reason why they call their filter design approach corrected projections simultaneous iterative reconstruction technique (P-SIRT). Moreover, it has been

shown that reconstructions with the optimized set of filters can reach an overall image quality that is comparable to that of iterative reconstruction techniques. In addition to this, Kunze et al. [172] have shown in an application of their optimized FBP (OFBP) method to real data from a digital breast tomosynthesis (DBT) system that the resulting images do not suffer from overshoots at the borders of the breast and that the intensity distribution in the images is more homogeneous than in the reconstructions computed with standard FBP. An extension of Kunze's method has been suggested by Ludwig et al. [173], where the P-SIRT-optimized filters are modeled by an approximation through a polynomial function. On clinical data it has been shown that the application of this technique preserves the contrast of low-frequency image features better than standard FBP. Ren et al. [202] have reported similar image properties for numerically optimized filters by the use of other iterative reconstruction techniques, like maximum-likelihood expectation maximization (ML-EM) and simultaneous algebraic reconstruction technique (SART). In the area of circular and limited-view CBCT, Batenburg and Plantagie [203] and Shtok et al. [204, 205] have proposed similar iterative optimization strategies for the construction of a geometry-adaptive set of filters that can be used in a FBP scheme. However, none of these techniques has been studied in conjunction with a limited angular projection range. In contrast to the iterative approaches described above, [183] derived a set of effective filters for the limited-angle case analytically by exploiting the properties of a specific acquisition geometry.

The second group of reconstruction algorithms used in the area of limited-angle CT is made up by blurring methods. The approach of Matrix inversion tomosynthesis (MITS) belongs to this group and has been proposed by Ghosh Roy et al. [206]. MITS removes blurring artifacts from adjacent planes by the use of a set of coupled equations. These equations exploit the knowledge about the blurring functions to solve exactly for the distortion generated by planes immediately adjacent to the in-focus plane. An extension of this method to the entire set of conventionally reconstructed planes has been developed by Dobbins III [207]. They have demonstrated that from the complete set of tomosynthesized slices an exact reconstruction of in-plane structures is possible, while the amount of out-of-plane blurring artifacts were reduced significantly. In addition to this, their improved approach can be realized efficiently with respect to runtime performance. However, [208] have shown for the improved MITS approach that noise at the very lowest spatial frequencies is considerably amplified. For this reason, Chen et al. [209] introduced a Gaussian frequency blending (GFB), which combines the advantages of the high-pass filtered MITS approach with those of low-pass FBP algorithms. A comparison between GFB and a pure MITS implementation has demonstrated that the former approach has a superior performance in relation to small-scaled structures and for the reduction of high-frequency noise. Moreover, the images reconstructed with the GFB showed more low-frequency content than MITS and standard FBP. Further developments in the area of deblurring methods, in particular constrained iterative restoration techniques for the separation of blur from in-plane image features, have been summarized in the review of Dobbins III and Godfrey [174].

The third category of techniques to solve the problem of limited-angle CT encompasses iterative reconstruction methods like SIRT and the classical algebraic reconstruction technique (ART) algo-

rithm (see Colsher [210]). Bleuet et al. [211] have extended ART with half quadratic regularization scheme for linear digital tomosynthesis, such that a significant improvement of the vertical resolution is achieved in combination with a noise and artifact reduction superior to normal ART. Additionally, Wu [212] presented an application and an analysis of the EM-algorithm for low-dose digital breast tomosynthesis (DBT) with respect to different acquisition protocols. They found out that for a radiological diagnosis 8 to 10 iterations deliver generally reconstructions with sufficient image details and feature contrast. However, the discussion by Kolehmainen et al. [181] states that even for a low number of projection views, the computational costs of iterative reconstruction algorithms are higher than those of FBP techniques, whereby a slowdown of a factor between 20 and 100 can be expected. In addition to this, iterative methods based on statistical optimization using the concept of Bayesian smoothing [213, 214] have been compared to simple backprojection, restoration-based deblurring and expectation maximization by Suryanarayanan et al. [213, 214]. Similar to this study, [181] has shown that for the limited-angle case iterative approaches achieve an image quality superior to FBP. Nevertheless, the comparison of the SAA algorithm, a FBP method and an iterative maximum-likelihood (ML) technique performed by Wu et al. [215] demonstrated the following three key aspects: (1) the SAA method showed the best noise reduction capabilities with respect to low-contrast masses, whereby details in the images were lost, (2) the FBP procedure generated the sharpest images, although the quality of the masses was the lowest and (3) only with the iterative ML reconstruction the microcalcifications and the masses were restored equally well. Note that for the field of DBT additional comparisons and evaluations of iterative reconstruction techniques along with further discussions about their clinical potentials have been presented in the publications of Dobbins III and Godfrey [174], Badea et al. [216], Chen et al. [217] and Sechopoulos [189]. Furthermore, an overview of current state-of-the-art hardware and software platforms used by tomosynthesis manufactures has been given in the review of Sechopoulos [180]. This publication also reveals that iterative methods have quite recently been implemented in DBT systems by several companies.

5.2.2 Laminography

Besides tomosynthesis, the field of industrial quality assurance is an increasing application area for limited-angle CT. As the demands on quality of modern printed circuit boards (PCBs) raise more and more, especially in the automotive and communication industry, mass production of PCBs is pushed, whereby the reliability of electronic components becomes increasingly important. Because it is challenging to produce PCBs with low defect levels, better detection strategies are needed to increase and guarantee quality standards. For NDT systems 2D X-ray inspection is a commonly used technique. Although X-ray inspection has low scan times and it is fairly easy to integrate into the production process, it has the major drawback due to the projection technique that features are superimposed in the acquired image. As a consequence, it is not possible to get any information about the depth of the object of interest. Especially in PCB inspection, this fact hinders the analysis of double-sided boards

with a high density of components. As scan times are crucial in the inline-inspection of assemblies, the main goal of an effective testing solution is to get real 3D data out of a single fast measurement.

Classical X-ray laminography was the first step into this direction. The basic principle of laminography has been proposed by [218], where the source and the detector rotate synchronously 180 degree out of phase while the inspected object remains stationary. Alternatively, the detector and the source can be translated into opposite directions [175]. All collected images are averaged, so that the parts of the object of interest that lie within the focal plane become prominent and out-of-focus features only contribute to the background in a blurred way. The main disadvantages of classical laminography are that the background intensity reduces contrast and that for the acquisition of multiply layers of the object it needs to be shifted with respect to the focus plane, resulting in long scan times. Additionally, the complicated mechanical scanning system yields to image distortions which need to be corrected by image post-processing [219].

To overcome the limitations of classical laminography, digital computed laminography (DCL) has been proposed [220], where digital detectors are used to store the projections. With DCL data it has been justified that algorithms from the fields CT and tomosynthesis can be applied to reconstruct a 3D dataset of the object of interest [221]. Furthermore, it has been shown that the 3D reconstruction problem that results from laminography, where the object is moved in a straight line across the X-ray beam with a fixed source and detector, is equivalent to that of a tomographic scan with limited angular access [221]. But in contrast to its classical counterpart in digital computed laminography all object layers can be obtained by one scan followed by a reconstruction procedure that first rebins the projections to a parallel beam geometry and then uses an iterative reconstruction algorithm like ART [174, 222, 223]. All in all, computed laminography has a simpler mechanical setup and yields a sharper image with a higher contrast resolution than the classical approach, because features do not get superimposed. The main advantage of computed laminography over CT is that large flat objects can be scanned and that the artifacts that exist in CT due to the lateral attenuation through large objects are significantly lower.

Digital computed laminography has been used in the NDT area, especially for the examination of electronic components like ball-grid-arrays (BGAs) [224] with an automatic inspection of solder joints [225] where the dimensions of the detail is vitally important. The introduction of a new approach for X-ray geometry with a minimum of moving parts in micro-laminography [226, 227] has shown that a separation of structures by $10\,\mu m$ is possible. A closer study of this technique [176] demonstrated the location of defects in integrated circuit (IC) packages, but also pointed out that cracks smaller than $5\,\mu m$ where not detectable. In [177] a digital tomosynthesis technique has been applied successfully to a multilayer PCB and evaluated with respect to various operation parameters showing that for a reliable image quality the scan angle should be larger than $60°$ with a step angle as narrow as possible.

Recently, the use of synchrotron radiation (SR) in CT area made it possible to distinguish weakly contrasted materials with a high spatial resolution down to the sub-micrometer scale. As CT is not suited for PCB inspection, the application of SR to computed laminography has achieved a spatial res-

olution down to the scale of 1 μm [228, 229]. This lower limit is mainly dominated by the mechanical precision of the system and due to laminographic artifacts that are related to the sample structure itself [230].

5.2.3 Approximate Inverse

In 1996 Louis [184] has introduced the AI for the regularized solution of linear problems. The great benefit of the method of AI is that a so-called reconstruction kernel can be precomputed from an auxiliary problem, which can then be used to calculate an approximate solution to the inverse problem. This approximation can be expressed as a scalar product of the precomputed kernel with the data of interest. Contrary to the technique presented in [231], the AI has the advantage that missing information is compensated by the reconstruction kernel and is not reconstructed from the known measured data, which might lead to wrong results in unsampled regions due to data incoherence. Consequently, the main advantage of the AI is that the reconstruction kernel can be precomputed independently from the measured data, such that it can be reused for different datasets, whereby the correction term, i.e. the kernel, itself acts independently from the underlying data and from the measurement errors. For this reason, the AI is less prone to data inaccuracies, e.g. projection noise, and data insufficiency, e.g. projection undersampling, than the method described in [231].

The PhD thesis of Dietz [232] described the first application of the AI to 2D circular fan-beam CT and for 3D circular CBCT (C-CBCT). The techniques derived by Dietz are quite similar to the FBP method developed earlier by Davison and Grünbaum [233], whereby their derivation was more complex and did not exploit the relation between the AI and the inverse problem of tomography. Although Dietz and Davison and Grünbaum constructed the filters by analytical inversion of the Radon transform, the approach of Davison and Grünbaum made a priori the assumption that the computed reconstruction filters vary with each projection angle. Therefore, the work presented by Dietz can be seen as an extension of [233] to a more general scheme. Additionally, Dietz [232] performed a comparison between the AI and the standard FDK method. His results demonstrate the behavior of both algorithms for varying noise levels. It can be observed that the FDK method achieves a lower reconstruction error than the AI when the noise level on the projections is low. However, as the amount of noise on the projections is increased the AI yields an image quality superior to that of the FDK algorithm.

Since for some applications the computation of the reconstruction kernel can become instable, Rieder and Schuster [234] have developed an alternative method based a combination of the AI with a singular value decomposition of the underlying system operator. This approach has been applied to 2D fan-beam CT and a detailed error analysis of their method in the context of FBP has been given. In 2003, Louis [185] has presented a further generalization of the AI with the derivation of an inversion formula for 3D cone-beam CT, whereby the implementation of the final algorithm was made computationally feasible by the exploitation of symmetries of the underlying scanning geometry. On reconstructions from real projection data, Louis has shown that the method can efficiently

produce reasonable results and that the noise in the final reconstructions is reduced to a minimum. A summary of the aforementioned techniques and a complete analysis of the AI with respect to 3D CBCT in comparison to the FDK algorithm have been presented in the thesis of Mohr [235]. This work also gives additional details on the implementation of the reconstruction kernel computation and on its use in a FBP-type algorithm. Furthermore, the AI has been used by Louis and Weber [187] for the development of rapid algorithms, which provide a combination of image reconstruction and image analysis in a single step. As a result of this approach, the authors have shown that direct edge detection can be performed within the reconstruction step, such that no additional post-processing is required. Even further developments in the area of the AI with respect to optimizations of the kernel calculation have been discussed by Louis et al. [236], Louis [237] in the context of industrial and medical imaging. First major steps into the direction of 3D helical CBCT supported by the theory of the AI have been conducted in the PhD thesis of Weber [238]. Weber concluded that the formulas resulting from his derivations were too complex to allow a concrete calculation of a reconstruction kernel for the helical CBCT (H-CBCT) case. Additionally, Hu and Zhang [239] have shown how to reduce the complexity of Katsevich's FBP algorithm by rewriting it in the form of the AI and by providing specific expressions for the required reconstruction kernels. Nevertheless, a concrete discrete implementation of the AI in the context of 3D helical CT has not been presented so far.

As an extension to the FDK approach, Feng et al. [240] have presented a kernel-based method, which obtains the reconstruction kernels as a combination of the AI with the FDK algorithm. The resulting reconstructions have been improved by the introduction of a version of the kernel that has compact support. Moreover, they have shown that the computation of the kernel is carried out in a moderate time. By the integration of an anisotropic diffusion model into their kernel-based approach, Feng and Zhang have demonstrated in [241] that a truncated kernel can significantly reduce noise in reconstructions computed from a limited number of views of a circular trajectory. Besides this application of the AI to undersampled data, the case of 2D limited-angle CT in relation to the AI has been analyzed by Dietz [232]. Nevertheless, the calculation of reconstruction kernels for 3D limited-angle CBCT in the areas of tomosynthesis and laminography is still outstanding.

5.3 Method

In the following the derivation of the novel optimization method will be presented. Therefore, the next section will start with an introduction of the notation and a description of the geometric CBCT system setup that will be assumed throughout the rest of this chapter.

5.3.1 Geometric Setup

As already explained in Section 4.3.1 there exist in principle two types of geometric setups for CBCT image systems. For the first kind of geometries the object of interest is rotated between a fixed source-detector setup. In the second kind of systems the X-ray detector and the source are moving around

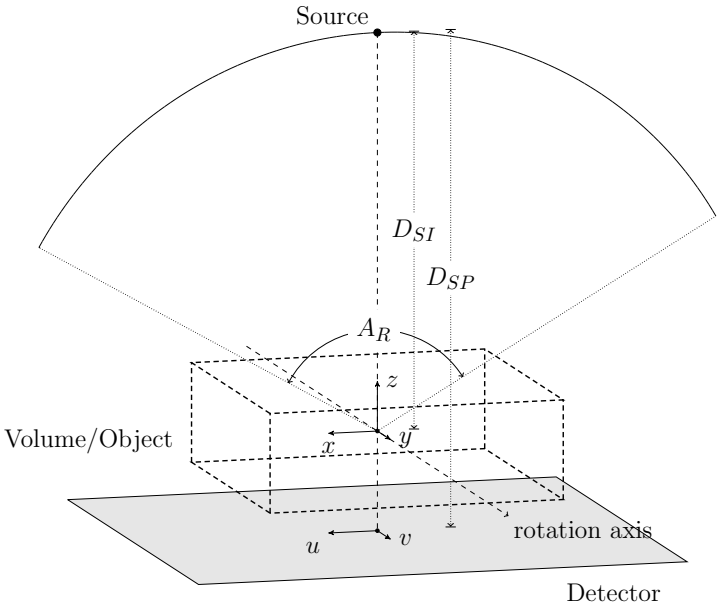


Figure 5.1: The general geometric setup of the used limited-angle CBCT scanner. The distance from the X-ray source to the rotation axis is given by D_{SI}. D_{SP} is the source-to-detector distance and D_{IP} is the distance from the ISO-center to the detector. Note that all projection images used during the experiments were acquired on a μCT scanner with a circular source trajectory. First a full scan of $360°$ was carried out. Afterwards the projection sets were generated by restricting the angular range to A_r.

the inspected item. Without loss of generality, this chapter will refer to a geometric setup where the source and the detector are moving. The here derived limited-angle reconstruction algorithm will be formulated by means of this system setup, although the CT scanner, which has been used for the data acquisition in the experiments of Section 5.4 uses a setup with a moving object. However, note that a mapping between both geometric setups onto each other can be achieved by a simple change of reference systems. Additionally and in accordance with Chapter 4, it will be assumed the CT scanner has a planar physical detector. Consequently, this ideal detector has no distortions, such that all of its elements are equally spaced and have the same dimensions.

For the here used limited-angle CT system the general geometric setup is display in Figure 5.1. The scanned object, and thus the reconstructed region of interest, is centered on the ISO-center at the point $\overline{x} = (0, 0, 0)$ where the ray from the X-ray source to the detector center intersects the rotation axis, i.e. the y-axis. The X-ray source rotates on a circular path with a radius of D_{SI} around the object, while the physical detector is positioned at a distance of D_{SP} from the source. Furthermore, the coordinate system of the detector is spanned by the vectors u and v, whereby the center of the detector is located at $(u, v) = (0, 0)$. Additionally, the coordinate space of the scanned object and of reconstructed volume is defined by the x-, y- and z-axis. The projection angle λ is given as the angle between the z-axis and the central ray of the projection (i.e. the ray from the source, through the ISO-center, to the center of the detector). At a projection angle of $\lambda = 0°$ the xy-plane is parallel to the detector and the z-axis points into the direction of the X-ray source. The total scan angle

range is denoted by A_r, such that the smallest and the largest projection angle can be expressed as $\lambda_{min} = -A_r/2$ and $\lambda_{max} = A_r/2$, respectively.

5.3.2 The Approximate Inverse

As mentioned in Section 5.2.3 the basic idea of the AI has been introduced by Louis [184]. Although the approach has been applied to the case of non-limited circular CBCT by several researchers [185, 187, 232, 235–237, 240, 241], efforts in the direction of 3D limited-angle CT have not been reported so far. In the following, a set of angle-dependent 2D projection filters will be constructed, which can be integrated into a FBP reconstruction framework to compute an approximate solution of the inverse problem in 3D limited-angle CBCT. For the design phase the method of Approximate Inverse (AI) will be used to derive numerically optimized and regularized filters from an intermediate reconstruction kernel.

Given the attenuation coefficients of the object of interest by $f \in \mathbb{R}^N$ and the projections collected by the image acquisition system as $g \in \mathbb{R}^M$, the inverse problem in tomography can be expressed by the following equation:

$$Af = g \tag{5.1}$$

where $A : \mathbb{R}^N \rightarrow \mathbb{R}^M$ is an operator that projects the volume of interest to the projections, i.e. the forward projection operator. This operator is also commonly known as the system matrix. In the case of limited-angle CT where projection data is acquired only from a restricted angular range the inverse problem formulated by equation (5.1) is a notoriously ill-posed one (compare with [188] and [242]). One common approach to solve ill-posed problems is by the use of the concept of regularization operators. Note that more details on the fundamentals of regularization methods can be found in the book of Tikhonov and Arsenin [243], while their applications in the field of CT have been summarized by Titarenko et al. [244].

To solve the ill-pose inverse problem given by equation (5.1) the regularization method of the AI uses a so-called mollifier, denoted by e_γ. The principle idea of mollification for the solution of improperly posed problems has been introduced by Hegland and Anderssen [245] and a detailed introduction and discussion of the mollifier concept in the field of CBCT can be found in the publications of Louis et al. [236], Louis [237]. The AI uses the mollifier to compute an approximation $\tilde{f}$ of the original image f by the evaluation of $\tilde{f}(\overline{x}) = \langle f, e_\gamma(\overline{x}, \cdot)\rangle$ at each location $\overline{x} \in \mathbb{R}^3$ inside the region of interest. Thereby, the mollifier $e_\gamma(\overline{x}, \overline{y})$ can be interpreted as convolution operator that applies a certain amount of smoothing to the image f. Assuming a fixed point $\overline{x}$ that defines the location at which the center of the mollifier is placed, the term e_γ becomes a function of $\overline{y} \in \mathbb{R}^3$. For this reason, the mollifier at a fixed location $\overline{x}$ will be denoted by $e_{\gamma,\overline{x}}(\cdot) = e_\gamma(\overline{x}, \cdot) \in \mathbb{R}^N$ throughout the following derivation. The amount of smoothing that is applied during the reconstruction procedure is controlled by the regularization parameter γ. The behavior of the mollifier has to be chosen such

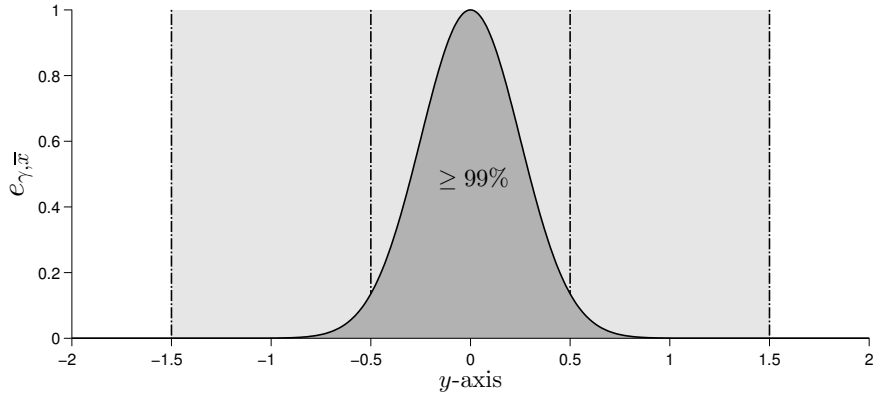


Figure 5.2: A 1D profile plot along the central y-axis through the 3D Gaussian mollifier defined in equation (5.3). The position of the mollifier has been fixed to $\overline{x} = (0, 0, 0)$ and the standard deviation of the Gaussian has been set 0.25 voxels. Therefore, 99 % of the support of the mollifier (dark gray) are within a range of 3 voxels (light gray).

that for a fixed point $\overline{x}$ and $\gamma \to 0$ the function $e_{\gamma,\overline{x}}(\cdot)$ converges against the delta distribution $\delta_{\overline{x}}(\cdot)$ at $\overline{x}$ defined by

$$\delta_{\overline{x}}(\overline{y}) = \begin{cases} +\infty, & \overline{y} = \overline{x} \\ 0, & \overline{y} \neq \overline{x} \end{cases}. \tag{5.2}$$

An example of a mollifier that fulfills the aforementioned properties is the Gaussian function. In the studies conducted by Louis et al. [236], Feng et al. [240], Feng and Zhang [241], Hu and Zhang [246] this function has been used successfully to integrate the AI with CBCT and it has been shown that a mollifier modeled by a Gaussian function reduces the projection noise significantly and is well suited to preserve important image structures. For these reasons, a 3D Gaussian mollifier has been chosen to perform the subsequent filter calculations. Consequently, assuming a fixed point $\overline{x}$, the 3D mollifier can be expressed as a function of $\overline{y}$ as in the equation below:

$$e_{\gamma,\overline{x}}(\overline{y}) = (2\pi)^{-\frac{3}{2}} \frac{1}{\gamma^3} e^{-\frac{\|\overline{x}-\overline{y}\|^2}{2\gamma^2}}. \tag{5.3}$$

To limit the support of the mollifier the standard deviation of the 3D Gaussian reflected by the parameter γ has been fixed to a value of 0.25. As a result, more than 99 % of the support of the resulting 3D function are within a region of $3 \times 3 \times 3$ voxels with respect to the system setup used in the experiments of Section 5.4. This choice for the regularization parameter γ lets the Gaussian approximate the delta distribution close enough, so that the blurring caused by the mollifier in the resulting reconstructed volume is kept at a minimum. With a further reduction of γ towards 0 no significant changes and improvements have been observed neither in the computed filters nor in the resulting reconstructions. Additionally, Figure 5.2 depicts a 1D profile curve plotted through the rotationally symmetric Gaussian mollifier and visualizes the voxels lying inside the support of the function.

With the selection of the mollifier in accordance with equation (5.3) it has been demonstrated in [184] that equation (5.1) can be solved for f approximately by $\tilde{f} = \langle f, e_\gamma \rangle$. Additionally, since for limited-angle CBCT the operator A in equation (5.1) can be assumed to be linear, the solution $\tilde{f}$ can be calculated by an approximation of the mollifier e_γ through a so-called reconstruction kernel ψ_γ, which lies in the range of the adjoint operator $A^* : \mathbb{R}^M \to \mathbb{R}^N$. Consequently and in accordance with the presentations of Louis [185], Louis et al. [236], Louis [237], an auxiliary problem can be defined that facilitates the computation of the reconstruction kernel. Given the mollifier $e_{\gamma,\overline{x}}(\cdot)$ and fixing its center at the location $\overline{x}$, the auxiliary problem can be expressed as

$$A^* \psi_\gamma(\overline{x}, \cdot) = e_{\gamma,\overline{x}}(\cdot). \tag{5.4}$$

In equation (5.4) the transposed of the system matrix A is represented by the operator A^*. Additionally, the reconstruction kernel at the point $\overline{x}$ is given by $\psi_\gamma(\overline{x}, \cdot) \in \mathbb{R}^M$ and is also denoted by $\psi_{\gamma,\overline{x}}(\cdot)$ in the further course of this thesis. Note that the reconstruction kernel $\psi_{\gamma,\overline{x}}(\cdot)$ has the same dimensions and thus resides in the same domain as the projections g. Knowing the kernel at the fixed point $\overline{x}$, the approximate solution $\tilde{f}$ of equation (5.1) can be calculated using the following derivation:

$$\begin{aligned} \tilde{f}_\gamma(\overline{x}) &= \int_{\mathbb{R}^N} f(\overline{x}) e_{\gamma,\overline{x}}(\overline{y}) d\overline{y} = \langle f, e_{\gamma,\overline{x}}(\cdot) \rangle_{\mathbb{R}^N} \\ &= \langle f, A^* \psi_\gamma(\overline{x}, \cdot) \rangle_{\mathbb{R}^N} = \langle Af, \psi_{\gamma,\overline{x}}(\cdot) \rangle_{\mathbb{R}^M} \\ &= \langle g, \psi_{\gamma,\overline{x}}(\cdot) \rangle_{\mathbb{R}^M} =: \mathcal{S}_\gamma g(\overline{x}), \end{aligned} \tag{5.5}$$

whereby $\langle \cdot, \cdot \rangle$ represents the scalar products in the appropriate spaces. The first step of the derivation shows how the approximate solution $\tilde{f}$ at $\overline{x}$ can be expressed as a convolution of the image f with the mollifier $e_{\gamma,\overline{x}}$. This integral operation can be reformulated as a scalar product, which can be rewritten in terms of the projection data g and the reconstruction kernel $\psi_{\gamma,\overline{x}}$ corresponding to the mollifier. Finally, in the last line of equation (5.1) the operator $\mathcal{S}_\gamma$ is defined as the AI for approximating the solution $\widetilde{(f)}$ of $Af = g$ at $\overline{x}$. The reconstruction technique formulated by this operator has two important properties: (1) the reconstruction kernels $\psi_{\gamma,\overline{x}}$ are independent of the underlying projection data g and can therefore by precomputed without any knowledge of g by solving the auxiliary problem given by equation (5.4); (2) as shown in the literature on the AI [185, 232, 233, 235, 236, 238], invariances, e.g. symmetries, of the mollifier can be exploited to speed up the computation of the reconstruction kernels and to integrate them into a shift-invariant FBP framework. Note that these two properties have been proven for the non-limited angle case of circular CBCT by Dietz [232]. Although the here presented filter optimization strategy is formulated for the limited-angle tomography, it will make use of both of the above proprieties to efficiently compute and use the derived set of filters. Since for the derivation of the AI in equation (5.5) no assumptions on the geometry of the imaging system are required, the first property from above can be trivially transfered to the limited-

angle case. To apply the second property to limit-angle CT a closer knowledge about the behavior of the reconstruction kernels with respect to the mollifier positioning is required. Therefore, a detailed analysis within this context will be discussed in Section 5.3.3.

As presented by the related work conducted on the method of AI in Section 5.2.3, there exist several methods to calculate the reconstruction kernel from a given mollifier: kernels can be obtained

- by derivation from an analytical reconstruction formula for C-CBCT, e.g. the Radon transform or Tuy's formula, like it has been presented in [185, 187, 232, 235–237, 240, 246] or for H-CBCT from Katsevich's exact inversion formula as discussed in [238, 239],
- by the analysis of the singular value decomposition associated with the operator of the inverse problem as described by Louis [184] and Dietz [232]
- or by using an iterative numerical optimization strategy as it has been employed and analyzed for 2D CT with a parallel scanning geometry by Rieder and Schuster [234].

None of the above approaches has been applied to the problem of cone-beam limited-angle tomography so far and only Dietz [232] has shown briefly how a singular value decomposition can be used to obtain kernels for the 2D limited-angle case. However, iterative optimization can provide a general approach for the design of filters on the basis of the AI without the need of an analytical inversion formula and without complex derivations performed by singular value decomposition (see [234]). For this reason, the AI reconstruction kernels $\psi_{\gamma,\overline{x}}$ will be computed numerically using a suitable optimization technique, such that the auxiliary problem introduced by equation (5.4) will be solved by a minimization operation at each point $\overline{x}$. Consequently, the resulting optimization problem has to be solved at N different locations that lie at the center of each voxel inside the volume to be reconstructed. This fact suggests the following general optimization approach:

Step 1: Position the 3D Gaussian mollifier $e_{\gamma,\overline{x}}$ given by equation (5.3) at a fixed point $\overline{x}$

Step 2: Compute the minimum norm solution of $\|A^*\psi_{\gamma,\overline{x}} - e_{\gamma,\overline{x}}\|_R$ to solve equation (5.4)

Step 3: Repeat steps 1 and 2 for each voxel to be reconstructed

Note that the second step represents a weighted least-square problem, which can be solved by iterative approximation using SIRT, whereby R is a positive definite diagonal matrix containing the row sums of the adjoint system matrix A^* and can be defined by

$$R := diag\left(\frac{1}{\sum_{j=1}^{N} a_{1j}^*}, ..., \frac{1}{\sum_{j=1}^{N} a_{Mj}^*}\right). \tag{5.6}$$

Hereby, the element in the ith row and jth column of the matrix A^* are given by a_{ij}^*. Gregor and Benson [247] have conducted an analysis of classical normalized SIRT and in accordance with their

notation the calculation of a reconstruction kernel $\psi_{\gamma,\overline{x}}$ can be carried out by using the following update expression in vector-matrix notation:

$$\psi_{\gamma,\overline{x}}^{(k+1)} = \psi_{\gamma,\overline{x}}^{(k)} + CAR(e_{\gamma,\overline{x}} - A^{*}\psi_{\gamma,\overline{x}}^{(k)}), \tag{5.7}$$

whereby the index k denotes the current iteration of SIRT, R is given by equation (5.6) and the diagonal matrix C is positive definite and contains the inverse column sums of the adjoint system matrix A^{*}. Consequently, C can be expressed as

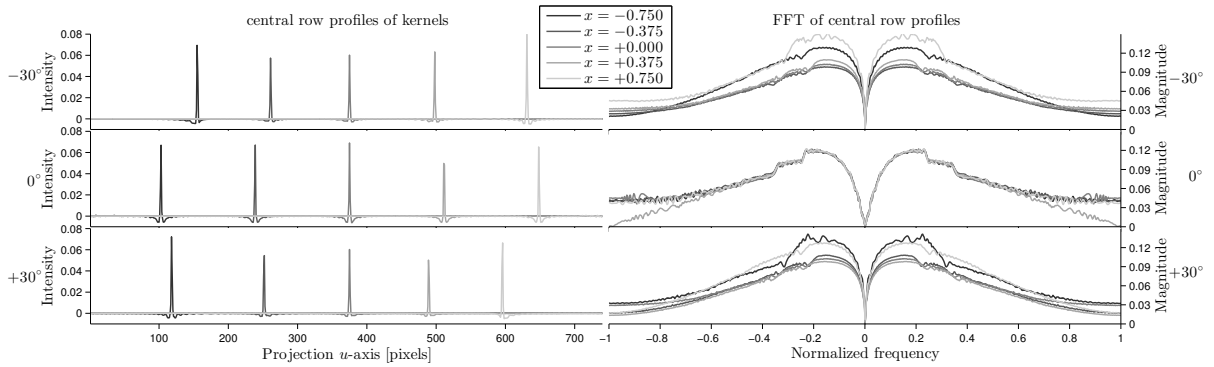
$$C := diag\left(\frac{1}{\sum_{i=1}^{M} a_{iN}^{*}}, ..., \frac{1}{\sum_{i=1}^{M} a_{iN}^{*}}\right). \tag{5.8}$$

In equation (5.7) the matrix R merely serves as a preconditioner for the associated normal equations. In relation to this, a scalar based preconditioning with a reduced memory consumption and a smaller memory footprint has been proposed in [247]. They have shown that this simplified preconditioning scheme can achieve similar result as the classical one and that it can be used in highly distributed implementations, e.g. on the graphics processing unit (GPU). However, since the approximation of the kernels was not performance critical and it had to be carried out only once during the precomputation phase, SIRT has been implemented straightforwardly using equation (5.7).

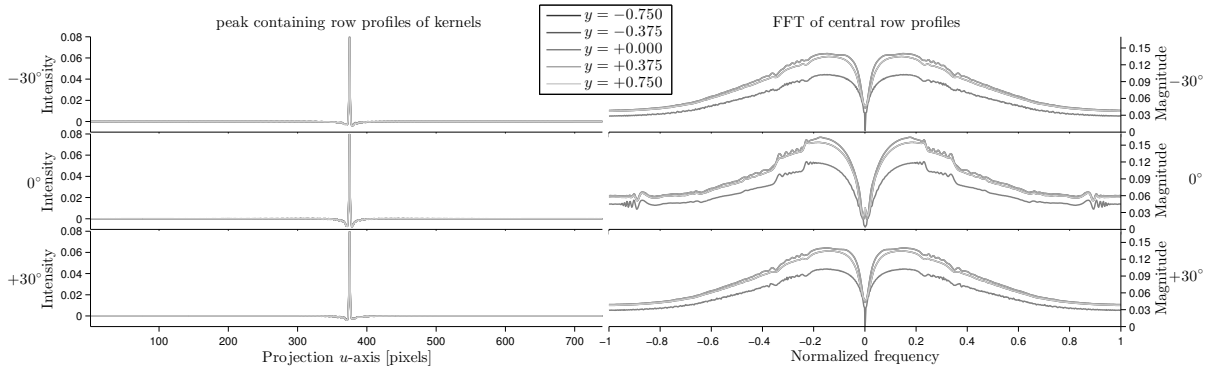
Contrary to this, the implementation of the forward projection operator A and the backprojection operator A^{*} has been realized on the GPU using Compute Unified Device Architecture (CUDA). Both operators employ the technique of separable footprints, which has been developed by Long et al. [21], Wu and Fessler [22]. Moreover, this forward projection and backprojection methods have been tightly integrated into the software framework introduced in Chapter 2, so that they are accessible from all layers, in particular from MATLAB. Therefore, the update scheme from equation (5.7) has been implemented in MATLAB to allow future modifications, while the projection operator performs more efficiently in native GPU code.

5.3.3 Reconstruction Kernel Approximation

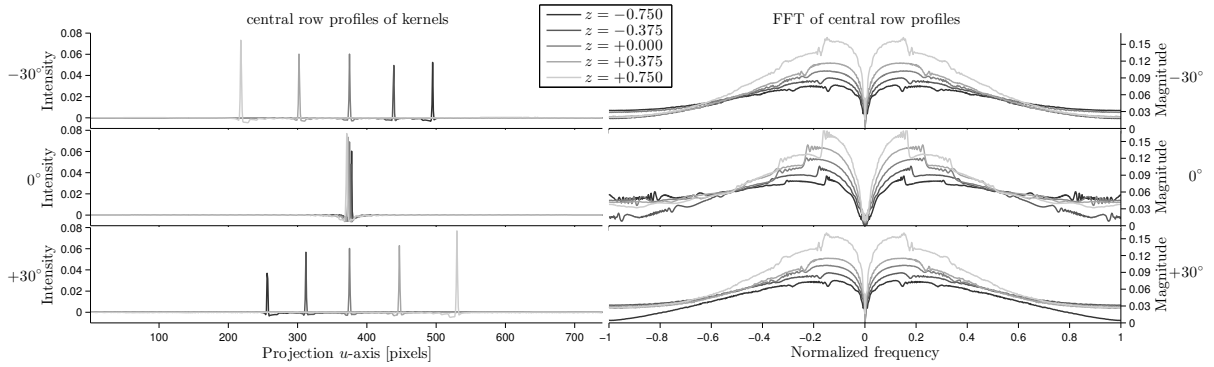
The steps 1 to 3 of the general optimization scheme discussed in the previous section constitute a naive approach for the computation of the kernels. This procedure can quickly become computationally impracticable as the size of the dataset to be reconstructed increases. The reason for this is that accordingly to equation (5.5) the reconstruction kernels depend of the choice of the point $\overline{x}$ that defines the mollifier position. Consequently, for each voxel inside the volume of interest a separate kernel $\psi_{\gamma,\overline{x}}$ has to be calculated by the use of SIRT. Each of these kernels requires the same storage capacity as the original set of acquired projections. Hence, the overall time to compute all reconstruction kernels and the total storage requirements can easily be exceeded as the size of the processed datasets becomes larger. In addition to this, it has been discussed in previous section that for an AI-based reconstruction in accordance with equation (5.5) invariance of the computed kernels can be exploited to employ them in a FBP algorithm. For this reason, the following studies will examine the behavior



(a) mollifier translated along central x-axis



(b) mollifier translated along central y-axis



(c) mollifier translated along central z-axis

Figure 5.3: Comparison of the reconstruction kernels computed from mollifiers placed along the central axes of the volume of interest. For the calculation of all kernels the 3D Gaussian mollifier has been used with $\gamma = 0.5$ and SIRT was run for 100 iterations. Note that the bounding box of the volume of interest has been normalized to $[-1, 1]^3$. The mollifiers have been placed along the central x-, y- and z-axis in (a), (b) and (c), respectively. The line plots on the left depict for each kernel the profile through its central row for with respect to the projection at $-30°$ (top), $-0°$ (middle) and $30°$ (bottom), while their corresponding frequency spectra are displayed on the right.

of reconstruction kernels $\psi_{\gamma,\overline{x}}$ in relation to their corresponding mollifiers $e_{\gamma,\overline{x}}$ for different choices of $\overline{x}$.

For the study of the invariances Figure 5.3 shows line plots that compare several reconstruction kernels. These kernels have been calculated from the 3D Gaussian mollifier located at differing posi-

tions along the three coordinate axes of volume of interest. For the computation of the kernels the geometric setup with an angular scan range of 60° from Section 5.4 has been used. The line plots have been generated with respect to three different projection angles: $-30°$ in the plots at the top, 0° in the plots in the middle and 30° in the plots at the bottom. The left part of each graph displays the profiles of the row containing the peak of the kernel. At the right side of the plots the Fourier spectra corresponding to the kernel rows are depicted. Note that the peak of the kernels generated from the x-translated and from the z-translated mollifiers fall onto the central row of the projections space, which corresponds to the central row of the kernels since $\psi_{\gamma,\overline{x}} \in \mathbb{R}^M$. On the contrary, the y-translated mollifier produces kernels translated accordingly along the v-axis of the detector.

In Figure 5.3a the kernels are displayed that have been computed from mollifiers translated along the x-axis. It can be observed that at $-30°$ and at 30° the profile plots of the central kernel rows look similar in their shape and distribution, with the only difference that they are mirrored with respect to the kernels computed from the mollifier at $x = 0$. Consequently, the frequency spectra at $\pm 30°$ have similar shapes when comparing them in a reversed order. Moreover, the profiles of the kernels corresponding to the projection angle of 0° only show minor changes in their shape for the x-translated mollifiers, whereby a single kernel ($x = 0.375$) has a peak height differing from the other ones. The Fourier spectrum of the central row of this kernel also shows differences to the other kernels at higher frequency ranges. For lower frequencies all reconstruction kernels show a similar behavior. Additionally, kernels from mollifiers placed at positions intermediate to the ones displayed in Figure 5.3a have been inspected. For these kernels shapes, peak heights and Fourier spectra corresponded to the behavior observed in the aforementioned line plots. Variations in the peak height at 0° can be explained by inaccuracies (e.g. accumulating forward- and backprojection artifacts) caused by numerical approximations made during the kernel optimization with SIRT. In summary, it can be concluded that the reconstruction kernels referring to the x-axis show a symmetric behavior with respect to the projections angle.

In contrast to this behavior observed along the x-axis, the central row profiles of the kernels calculated from the y-translated mollifiers do not show significant difference in their shape and peak height for different y-positions. A closer look at the negative side lobes of the kernels at $-30°$ and 30° shows that these minima are mirrored with respect to the projection angle and that their amplitude is smaller than for the 0° case. Consequently, for the different projection angles the corresponding Fourier spectra of the kernel rows reveal a symmetric behavior in their shapes. Only for the kernel at $y = 0$ the spectrum needs to be scaled to match those of the other reconstruction kernels. Additionally, the kernels computed from the mollifiers positioned along the z-axis indicate a symmetry for the minimal and maximal projection angle. As for kernels of the x-translated mollifiers the row profiles are mirrored with respect to the scan angle, what is once more reflected by the according frequency spectra. The profile curves at 0° show slight variations of the peak height, which are caused by the geometry of the used system setup and will be studied in detail in Section 5.4.4. Moreover, for the 0° case the Fourier spectra of the reconstruction kernels of the z-translated mollifiers have quite similar, vary in their scaling and differ from the spectra of the kernels at the other two projection angles.

Eventually, the above observations suggest that the principle shape of the reconstruction kernels is similar for differing mollifier positions. Moreover, it has been demonstrated that the profile curves and the corresponding frequency spectra vary symmetrically with a change in the projection angle. As a result of the above analysis, the following approximations will be assumed to make the calculation of the kernels computationally feasible and to rigorously reduce the storage requirements of the optimization scheme proposed in the previous section:

1. The overall appearance and thus the frequency spectra of the reconstruction kernels are independent from a translation of the mollifier along the x-, y- and z-direction

2. and the parts of a specific kernel corresponding to a projection image vary with the scan angle.

To put it differently, a reconstruction kernel does not depend on the mollifier position, but on the projection angle. Note that for the application of the AI to 2D limited-angle CT the latter behavior has also been observed by Dietz in his PhD thesis [232]. As a result of the above assumptions, the numerical kernel optimization will be simplified by the calculation of only a single reconstruction kernel ψ_γ^0 from the Gaussian mollifier placed at the ISO-center $\overline{x} = (0, 0, 0)$. Nevertheless, through the made simplifications one has to consider and accept a certain approximation error, which will influence the finally reconstructed image. A closer look at this error will be taken in the analysis presented by Section 5.4.4.

5.3.4 Filter Optimization

Once the central reconstruction kernel ψ_γ^0 has been computed by means of the optimization methods described in the previous sections, a set of 2D image filters $\omega_\gamma^0 \in \mathbb{R}^M$ can be designed that depends on the angle of the acquired projection. For these filters it will be shown that they can be used in a shift-invariant FBP reconstruction algorithm to compute an approximation of f from the measured projections g, whereby an evaluation of the resulting image quality will be presented in Section 5.4.

For the design of the filters, the required notations will be introduced first. Although some of the following formulas have been employed in the previous chapters of this thesis (compare Chapter 3 and Chapter 4), they will be redefined within this section to achieve a consistent notation that avoids misunderstandings in the derivation of the here presented method. As described in Section 5.3.1, the parameter λ defines the acquisition angle between the central ray of projection and the z-axis. Consequently, the positions of the X-ray source can be expressed in dependence of λ by the expression

$$\overline{s}(\lambda) = (D_{SI} \sin\lambda, 0, D_{SI} \cos\lambda), \tag{5.9}$$

where $\lambda \in [\lambda_{min}, \lambda_{max}] = [-A_r/2, A_r/2]$ in accordance with Section 5.3.1. Thereby, a parameterization of the set of projection images g can be formulated by the cone-beam transform:

$$g(\lambda, u, v) = g(\lambda, \overline{\theta}) = \int_0^\infty f(\overline{s}(\lambda) + t\overline{\theta})dt, \tag{5.10}$$

with the vector $\overline{\theta}$ that represents the direction from the location of the X-ray source to the detector coordinates.

Additionally, the angular projection range A_r is sampled at L discrete points given by the set $\Lambda = \{\lambda_0, ..., \lambda_L\}$, which is index by the variable l. Hence, a discrete projection angle $\lambda_l \in \Lambda$ defines a source location $\overline{s}_l := \overline{s}(\lambda_l)$ in accordance with equation (5.9). The projection image acquired with the source at $\overline{s}_l$ is denoted by $g_l = g(\lambda_l, \cdot)$, whereby each projection has N_v rows and N_u columns. Consequently, the total number of elements in the acquired set of projections g can be calculated as $M = N_u \times N_v \times L$. The size of the elements in a projection image are given by the values Δu and Δv, which define the detector column width and the row height, respectively. By means of the element count and size, the locations of the detector pixels can be derived as $u_n = (n - (N_u - 1)/2)\Delta u$ and $v_m = (m - (N_v - 1)/2)\Delta v$, whereby the columns and row indices are within the ranges $n = [0, ..., N_u - 1]$ and $m = [0, ..., N_w - 1]$, respectively. According to this sampling, the detector center lies at $(u, v) = (0, 0)$ and the vector pointing from the source position $\overline{s}(\lambda_l)$ to the discrete detector coordinates (u_n, v_m) will be denoted by $\overline{\theta}_{n,m}(\lambda_l)$. Moreover, the part of the calculated central reconstruction kernel ψ^0_γ that corresponds to a single projection image g_l is parameterized accordingly by $\psi^0_{\gamma,l} = \psi^0_\gamma(\lambda_l, \cdot)$. Similarly, the parameterization of the set of 2D filters represented by ω^0_γ will contain for each g_l a separate filter that is denoted by $\omega^0_{\gamma,l} = \omega^0_\gamma(\lambda_l, \cdot)$.

The basic notation and the parameterization of the projections, kernel and filters provided above can now be used to take a closer look at the filter design from the central reconstruction kernel. Since ψ^0_γ is approximated numerically by an unregularized version of SIRT it can be observed in Figure 5.3 that the frequency spectra contain an unpredictable amount of ripples. These artifacts occur especially at higher frequencies and can be noticed in the spectral distributions of all parts $\psi^0_{\gamma,l}$. However, for an ideal solution of equation (5.4) one would expect the reconstruction kernel to have a smooth shape, like it has been shown for the analytically computed kernels in the AI literature (e.g. see the ones depicted in [232, 235, 236]). The ripples in the here calculated solution are caused by the SIRT algorithm, which inherently suffers from small numerical imprecisions that accumulate over the iterations to yield visible artifacts. This accumulation process can be avoided by the use of a suitable regularization method, which can be integrated directly into the SIRT algorithm. For the solution of the inverse problem in equation (5.1) many such regularization schemes have been proposed, also in relation to limited-angle CT (see [242, 248–252]). However, none of these techniques have been applied to solve the auxiliary problem stated in equation (5.4) in the context of the AI. Therefore and for the sake of simplicity, the unwanted artifacts will be removed from the frequency spectra by a low-pass filtering approach. For this purpose, a finite impulse response (FIR) filter is designed using the window method. As window function the 2D Kaiser window, also commonly known as Kaiser-

Bessel window, will be employed. Given a detector pixel with the indices (n, m), the Kaiser window can be defined in the domain of the projection images by the following equation:

$$w_\tau(n,m) = \begin{cases} \frac{I_0\left(\alpha\kappa(\frac{2n}{N_u-1})\right) I_0\left(\beta\kappa(\frac{2m}{N_v-1})\right)}{I_0(\alpha) I_0(\beta)} & \text{for } 0 \leq n \leq N_u - 1 \text{ and } 0 \leq m \leq N_v - 1 \\ 0 & \text{otherwise} \end{cases} \tag{5.11}$$

with I_0 representing the zeroth order Modified Bessel function of the first kind and the function $\kappa(\xi) := \sqrt{1-(\xi-1)^2}$. Moreover, a single control parameter $\tau \in [0, +\infty]$ is introduced, which determines the shape of the Kaiser window by acting consistently on the variables α and β in equation (5.11). Consequently, values of both have been calculated as $\alpha := N_u\tau$ and $\beta := N_v\tau$. By this approach the parameter τ can be interpreted as a window size, which in turn controls the smoothing of the Fourier spectrum of the windowed function, i.e. the image. For a value of $\tau = 0$ the Kaiser window becomes $w_\tau(n, m) = 1$ within the range of the first case in equation (5.11), while for increasing values of τ the window becomes narrower. As a result of this behavior, the application of the window function to an image will smooth its Fourier spectrum stronger for larger values of τ, while a value of $\tau = 0$ leaves the spectrum unchanged. Note that the Kaiser window has been defined independently of the source angle. Finally, the set of projection-wise optimized 2D filters can be calculated by the application of the 2D Kaiser window w_τ to the computed reconstruction kernel ψ^0_γ. This process can be expressed as

$$\omega^0_{\gamma,l} = w_\tau * \psi^0_{\gamma,l} \quad \forall \quad l \in [0, ..., L], \tag{5.12}$$

whereby the index l corresponds to the scan angle and thus to the source position according to equation (5.9). Note that $*$ denotes the element-wise multiplication operator for two projection images of the size $N_u \times N_v$. Figure 5.4 demonstrates the impact of the Kaiser window on the frequency distribution by comparing the central rows of the originally calculated reconstruction kernel ψ^0_γ, of the Fourier spectra of ψ^0_γ and of the spectra of the windowed kernel, i.e. the resulting optimized filters. It can be seen that the final filters have a smoothened appearance in the Fourier domain and that the ripples have been removed by the application of the window function.

For the use of the filters $\omega^0_{\gamma,l}$ computed from equation (5.12) in a FBP-style reconstruction algorithm, a unitary matrix $U^T_{\lambda,x}$ is introduced. Given a fixed source angle λ and a point $\overline{x}$, this matrix rotates the vector $(\overline{s} - \overline{x})/|\overline{s} - \overline{x}|$ onto the vector $\overline{s}/D_{SI}$ and can be defined through

$$U^T_{\lambda,\overline{x}} \left(\frac{\overline{s}(\lambda) - \overline{x}}{|\overline{s}(\lambda) - \overline{x}|} \right) = \frac{\overline{s}(\lambda)}{D_{SI}}, \tag{5.13}$$

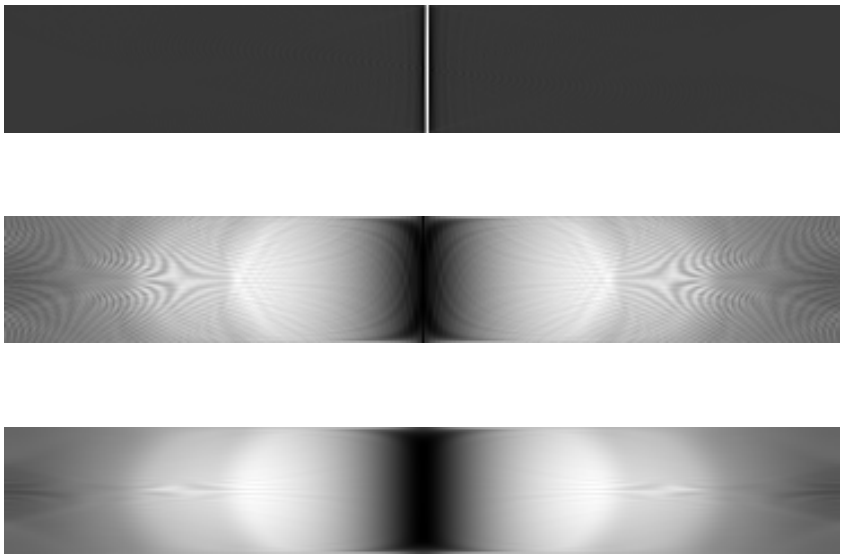


Figure 5.4: Reconstruction kernel visualization demonstrating the effect of the Kaiser window. The top image shows the central rows extracted and stacked row-wisely for each projection angle with respect to the reconstruction kernel ψ_γ^0. The middle image shows for each kernel row the corresponding Fourier spectra and the bottom image depicts the spectra after the Kaiser windows has been applied in the spacial domain.

whereby $\overline{s}(\lambda)$ describes the circular X-ray source path in accordance with equation (5.9). For the case of a circular trajectory and for the assumption that $U\overline{s}'(\lambda) \approx \overline{s}'(\lambda)$, Dietz [232] and Louis [185] have proven for the reconstruction kernels that the following approximate invariance holds true:

$$\psi_{\gamma,\overline{x}}(\lambda, \overline{\theta}) \approx \frac{D_{SI}^2}{|\overline{s}(\lambda) - \overline{x}|^2} \psi_\gamma^0(\lambda, U_{\lambda,\overline{x}}^T \overline{\theta}). \tag{5.14}$$

By the employment of this equation the central kernels ψ_γ^0 can be used to compute reconstruction kernels at arbitrary positions $\overline{x}$ inside the volume of interest. Consequently, only ψ_γ^0 is required for the evaluation of equation (5.5) to obtain $\tilde{f}$ for each desired location $\overline{x}$. Moreover, the works of Dietz [232, pp. 72–76] and Mohr [235, pp. 37–40] derived and proved that equation (5.5) can be reformulated as a shift-invariant FBP algorithm by exploiting the invariance given in equation (5.14). As a result, the AI operator $\mathcal{S}_\gamma$ with its involved scalar product can be rewritten as

$$\begin{aligned} \tilde{f}_\gamma(\overline{x}) &= \mathcal{S}_\gamma g(\overline{x}) = \langle g, \psi_{\gamma,\overline{x}}(\cdot) \rangle_{\mathbb{R}^M} \\ &\approx \frac{1}{8} \sum_{l=1}^{L} \sum_{n=1}^{N_u} \sum_{m=1}^{N_v} g(\lambda_l, \overline{\theta}_{n,m}) \psi_{\gamma,\overline{x}}(\lambda_l, \overline{\theta}_{n,m}) \\ &\approx \frac{1}{8} \sum_{l=1}^{L} \frac{D_{SI}^2}{|\overline{s}(\lambda_l) - \overline{x}|^2} \sum_{n,m} g(\lambda_l, \overline{\theta}_{n,m}) \psi_\gamma^0(\lambda_l, U_{\lambda_l,\overline{x}}^T \overline{\theta}_{n,m}). \end{aligned} \tag{5.15}$$

The last line of the above equation makes clear that the algorithm is of FBP-type, since the inner sum represents a 2D convolution of the projection image $g(\lambda, \cdot)$ with the corresponding part of the central reconstruction kernel $\psi^0_\gamma(\lambda, \cdot)$.

Since the optimized angle-dependent filters $\omega^0_{\gamma,l}$ are left unchanged with respect to projection geometry and are only subject to a smoothing of the frequency spectrum when computed from the corresponding part of the central kernel $\psi^0_{\gamma,l}$ in accordance with equation (5.12), the assumption can be made that equation (5.14) also holds true when replacing the kernels ψ by the set of filters ω. For this reason, the invariance of equation (5.14) also applies for the angle-optimized filters $\omega^0_{\gamma,l}$ and the final implementation of the limited-angle FBP approach will compute the approximation to f by

$$\tilde{f}_\gamma(\overline{x}) \approx \frac{1}{8} \sum_{l=1}^{L} \frac{D_{SI}^2}{|\overline{s}(\lambda_l) - \overline{x}|} \sum_{n,m} g(\lambda_l, \overline{\theta}_{n,m}) \omega^0_\gamma(\lambda_l, U^T_{\lambda_l, \overline{x}} \overline{\theta}_{n,m}). \tag{5.16}$$

Furthermore, the convolution step, i.e. the filtering, that needs to be performed in equation (5.16) is carried out in the Fourier domain using a projection-wise approach. Given the original acquired projection image g_l and the associated 2D filter $\omega^0_{\gamma,l}$, the filtered projection parameterized by $h_l = h(\lambda_l, \cdot)$ can be calculated in accordance with the following equation:

$$h_l(u_n, v_m) = \mathcal{F}^{-1}\{\mathcal{F} g_l(u_n, v_m) * |\mathcal{F} \omega^0_{\gamma,l}(u_n, v_m)|)\}, \tag{5.17}$$

whereby $\mathcal{F}$ is the 2D Fourier transform and $\mathcal{F}^{-1}$ its inverse. Specifically, for the evaluation of equation (5.17), the Fourier transforms of g_l and $\omega^0_{\gamma,l}$ are calculated first. Then from the angle-optimized filter the magnitude ($|\cdot|$) of the frequency spectrum is taken to remove undesirable phase information, which is caused by the numerical derivation of the filters using SIRT. Finally, the magnitude spectrum is multiplied element-wisely ($*$) by the Fourier transformed projection and the result is transformed back to the spatial domain to gain the filtered projection h_l. The evaluation of equation (5.17) has to be carried out for each of the L projections in g to obtain the full set of filtered projections h. Having successfully computed the set of images h, the backprojection operator A^* can be employed to calculate the reconstructed volume of interest in accordance with equation (5.15) by

$$\tilde{f}_\gamma(\overline{x}) = A^* h(\overline{x}). \tag{5.18}$$

The full algorithm which has been derived above is called angle-optimized FBP (AO-FBP). To improve the performance of the above computations, the processing has been parallelized, such that multiple projections are filtered concurrently. By this means, capabilities of current multi-core central processing units (CPUs) can be fully exploited, while the backprojection step is realized efficiently on the GPU.

Table 5.1: Angular ranges used for simulated and real data

	LOW (60°)	**HIGH (120°)**	**FULL (360°)**
Num. of Projections	60	120	1000
incr. Angle [°/*proj.*]	1.0	1.0	0.36

5.4 Experiments

As depicted in Figure 5.1, the geometric system setup used for the derivation of the method in the previous section looks similar to a typical breast tomosynthesis system (compare with Kunze et al. [172]). However, the following experiments with real datasets were carried out on an industrial μCT scanner with a circular scanning path, which also has been used for some experiments in Section 4.4. For the image acquisition the source-to-detector distance of the scanner was set to $D_{SP} = 451.267$ mm and the scanned object was placed at $D_{SI} = 111.656$ mm from the X-ray source. In accordance with Section 4.4, the detector has $N_v = 1848$ rows and $N_u = 1480$ columns, whereby the height and the width of detector elements equal $\delta_v = \delta_u = 0.127$ mm. Note that for the simulation of the projection data studied in this section and for the reconstructions from this data the same system setup has been used.

The results from the study of industrial limited-angle CT presented by Cho et al. [177] have demonstrated that the image quality for a total scan angle of $A_r \geq 60°$ can generate results comparable to a full scan of 360°. Therefore, an angular range of at least 60° has been used during the following experiments. Additionally, three different angular scan ranges will be compared by the experiments in accordance with the following acquisition protocols: (1) with the protocol denoted by LOW 60 projections have been acquired over a total scan angle of 60°; (2) during the scans with the HIGH protocol 120 projections are taken with an angular coverage of 120° and (3) the third protocol, denoted by FULL, uses 1000 projections from a full CT scan over 360°. Note that for the experiments performed on the real data, only the FULL protocol has been used to acquire projection images. Then the other two configurations, i.e. the LOW and HIGH protocol, have been generated from the full scan by selecting the according subset of projections. Additionally, the reconstructions computed from the scan of the FULL protocol will be employed as a reference for the results achieved by the other two setups. In contrast to this, the simulated projections have been generated directly with the LOW and the HIGH protocol. As a reference for the reconstruction from these simulations the original phantom images were used. Note that Table 5.1 gives an overview of the used angular projections ranges used throughout the following presentation.

5.4.1 Quality Measurement

Since the distribution of the attenuation coefficient is known a priori for the reconstruction from the simulated projection sets, the original phantoms can be used as reference for the resulting images.

Given the reference and the reconstructed dataset, the RMSE can be used as a measure for the quality of the different reconstruction algorithms. Additionally, in Section 5.4.4 the RMSE will be used to analyze the deviations between different sets of filtered projections.

In the general case, for two data vectors $\overline{a}, \overline{b} \in \mathbb{R}^K$, the RMSE can be computed as

$$RMSE(\overline{a}, \overline{b}) = \sqrt{\frac{\sum_{i=1}^{K}(a_i - b_i)^2}{K}}, \tag{5.19}$$

whereby K represents the number of vector elements a_i and b_i, which has to be equal for both vectors $\overline{a}$ and $\overline{b}$. Note that the two vectors can be either projection sets of volumes.

From the derivation of the filter optimization approach described in Section 5.3 it becomes evident that the AO-FBP is influenced by a number of parameters intrinsic to the algorithm. Two key factors that influence the image quality are: (1) the number of SIRT-iterations used to compute the reconstruction kernel and (2) the variable τ that controls the size of Kaiser window during the filter construction. The study of these parameters and of how they affect the final quality of the reconstructed images will be performed on the basis of the RMSE. Moreover, the RMSE is used to gain better understanding of the filters designed in the above optimization approach. The here developed method will be compare to the FDK algorithm, which has been implemented as the standard FBP approach presented by Feldkamp et al. [12] in combination with a Shepp-Logan (SL) filtering step. Additionally, the 2D Shepp-Logan (SL) phantom [165] is used for the generation of the simulated projection data. The 2D slices of the phantom have been arranged parallel to the xy-plane and stacked along the z-axis in accordance with the system setup described in Section 5.3.1. As a result, the 3D reference volume employed during the simulations is equal with respect to all xy-slices. The phantom projections are then generated using the LOW angular range setup (see Table 5.1). After this, volumes will be reconstructed with AO-FBP and FDK, whereby the voxel size is set to 0.179 mm along each dimension and the total number of reconstructed voxels is equal to $255 \times 255 \times 127$. Note that an odd number of voxels has been chosen along each coordinate axis, such that the mollifier can be positioned exactly at the location of the ISO-center, as it has been pointed out in Section 5.3.3. Furthermore, for the study of the algorithmic parameters a 2×2 binning has been applied to the projection data, so that 4 pixels were averaged into a single one. As a result, the number of detector rows and columns are given by $N_v = 924$ and $N_u = 740$, receptively. After the reconstructions have been calculated successfully, slice images and profile plots will be generated to visually inspect the image quality in relation to the in-focus xy-plane. The results with respect to SIRT-iteration count and Kaiser window size will be described in following sections.

5.4.2 Iteration Count

As the reconstruction kernel from which the optimized filters are constructed is computed numerically by SIRT in accordance with equation (5.7), the number of iterations used during the kernel optimization has an influence on the final image quality achieved by the algorithm. Figure 5.5 visu-

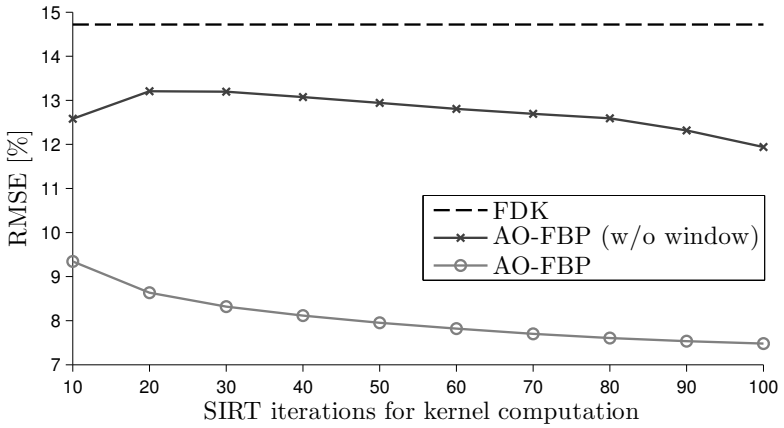
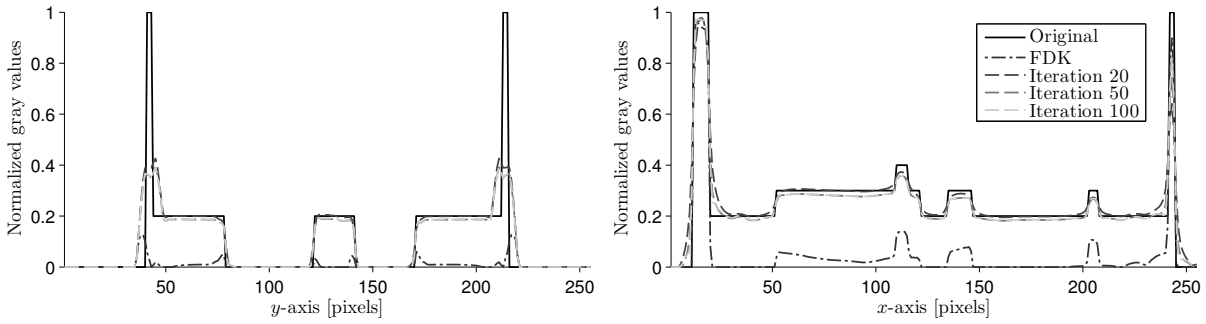


Figure 5.5: Comparison with respect to the RMSE between standard FDK (dahsed line), AO-FBP without application of a window function (line with crosses) and AO-FBP using the Kaiser window for filter design (line with circles). The number of SIRT iterations used to compute the reconstruction kernels is plotted against the RMSE of the original SL phantom and the reconstructed image. The window size of $\tau = 0.676$ was determined by hand-tuning and the mollifier size γ has been set to 0.25 voxels.

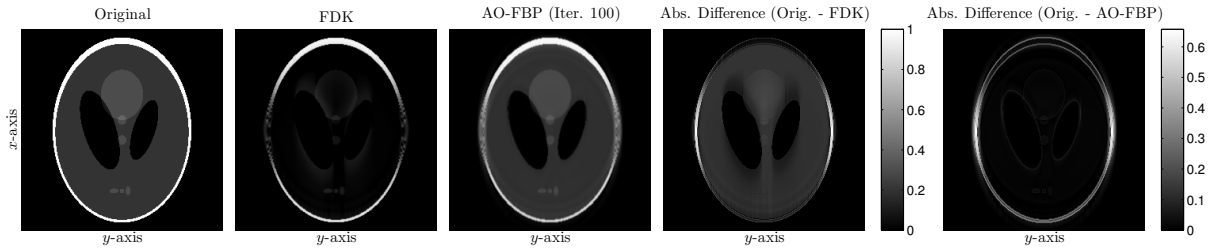
alizes this relation by comparing the number of SIRT iterations used to compute the reconstruction kernel with the achieved image quality in terms of RMSE. A look at the number of iterations in relation to the RMSE of the images reconstructed with AO-FBP reveals that with an increasing iteration count the RMSE drops from 9.34 % down to 7.48 % at 100 iterations. When using the reconstruction kernels without the application of the 2D Kaiser window the RMSE of the final reconstructions stayed above 11.9 % for all iterations. However, the FDK algorithm produced a higher RMSE of 14.7 % for the chosen setup.

For the proposed AO-FBP algorithm the change in the RMSE from 80 iterations to 100 is smaller than 0.06 %, so that an improvement in the image quality with further iterations can be expected to be even lower. This is why the number of SIRT iterations is fixed to 100 for all following experiments. The profile plots in Figure 5.6a underline this choice again. The displayed slice profiles of the central phantom row and column generated with AO-FBP using different SIRT iterations (dashed lines) show a visible improvement relative to the original profile from iteration 20 to 50, while the profile curve stays almost unchanged from iteration 50 to 100. In comparison to FDK, it can be seen that AO-FBP reconstructs the gray values of the original phantom more accurate, especially in the central regions of the SL phantom where the profiles of the FDK reconstruction are clearly below the original intensity values.

This behavior can also be verified by the images in Figure 5.6b which compare the central xy-slices of the FDK and AO-FBP reconstructions with the original SL phantom. In the difference images at the right of the figure it can be seen that the largest absolute deviations between the original and the AO-FBP reconstructed slice occur mainly at the edges of the phantom and are not larger than 0.66 in their absolute value. In contrast to this, the largest errors between the FDK and the original phantom occur at the left and right border and have a magnitude of 1.0. Additionally, the deviations between



(a)



(b)

Figure 5.6: In (a) the profile plots display the gray values of the central row (left) and central column (right) of the SL phantom to compare the original image with FDK and with AO-FBP using kernels computed from three different numbers of SIRT iterations. The images in (b) compare the reconstructions of the central slice of the SL phantom. The left image shows the original phantom, the image second from the left is a reconstruction using the FDK algorithm, the central image has been computed using AO-FBP with a filter set from a 100-iteration SIRT optimization. The image second from the right displays the differences between the original phantom and the FDK reconstruction. The rightmost image visualizes the deviations between the second and the third image. Note that all volumes haven been normalized to the range $[0, 1]$.

the FDK and original image at the interior part of the phantom (excluding the white borders) range up to 0.284 with a mean value of 0.153, whereas these deviations for the AO-FBP have only a maximal value of 0.103 with an average value of 0.012.

5.4.3 Window Size

As described in Section 5.3.4, the window size parameter τ controls the smoothing of the frequency spectrum of the reconstruction kernels during the filter construction. As a result, it also influences the final image quality of the AO-FBP algorithm. The plots in Figure 5.7 show how τ is related to the RMSE of the final reconstruction and how the gray values change with varying parameter values. For window sizes $\tau < 0.06$ the left plot shows a ripple artifact. This effect is due to the fact that Kaiser window at those parameter values becomes narrower than the central peak of the reconstruction kernel, such that the windowing changes the overall shape and behavior of the kernel. For this reason, the application of the Kaiser window makes little sense for values of $\tau < 0.06$ and this area is ignored when determining an optimal window size τ^*.

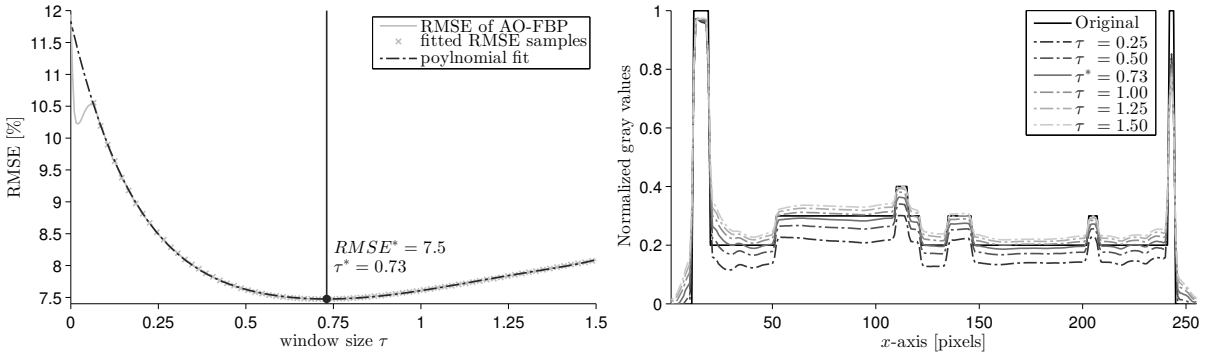
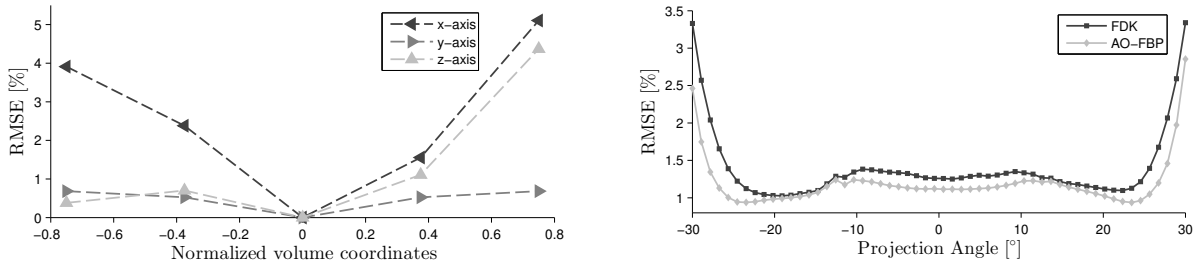


Figure 5.7: Comparison of different window sizes with respect to the RMSE using the SL phantom. In the left graph the window size parameter τ is plotted against the RMSE (solid light gray line). The optimal window size of $\tau^* = 0.73$ (dark gray dot) hs been determined by fitting a 6th degree polynomial (dashed dark gray line) to the samples. To ignore the artifacts introduced by window sizes smaller than 0.06, only samples with $\tau \geq 0.06$ (light gray crosses) have been considered during the estimation process. The profile plot at the right shows curves along the central column of the central slice of the SL phantom comparing the original gray values with those of the AO-FBP reconstructions at different window sizes.

The optimal value of τ^* is estimated using a straightforward linear line search strategy on the basis of the minimization of the RMSE. Therefore, the reconstructions from the simulated SL projection have been computed with AO-FBP using different windows size in the interval $[0.06, 1.5]$ with a sampling distance of $\Delta\tau = 0.0025$. For each τ the corresponding RMSE has been calculated with respect to the original phantom (see light gray crosses in Figure 5.7). After the sampling of the RMSE, a 6th degree polynomial function has been fitted to the samples to determine the location, i.e. the window size, with the minimal error. By the application of this procedure, the optimal window size of $\tau^* = 0.73$ has been estimated. The corresponding minimal RMSE was at $7.5\,\%$. The here found optimal value 0.73 for the window size parameter will be used in all subsequent experiments of this chapter.

Furthermore, the left graph in Figure 5.7 shows that the RMSE rises only slowly with a non-optimal value of τ and for parameter values $\tau^* \pm 0.5$ the error is still lower than $8\,\%$. This fact suggests that the here developed AO-FBP algorithm does not strongly depend on the precise selection of the window size τ, since values close to the optimum produce reasonable results. However, when looking at the profile plots in Figure 5.7 a quite contrary observation can be made: a significant deviation from original phantom can be seen for the reconstruction computed with window sizes of $\tau = 0.5$ and $\tau = 1$. Moreover, the differences for $\tau = 0.5$ and $\tau = 1$ occur particularly in regions at the interior part of the SL phantom and become stronger for the other choices of $\tau \in \{0.25, 1.5\}$. On the contrary, the optimal window size of $\tau^* = 0.73$ produces a profile curve that resembles the gray values of the original phantom most accurately. However, the relatively small variations in the RMSE for values of τ in the interval $[0.5, 1.0]$, as they have been observed in the left plot of Figure 5.7, can basically be explained by the magnitude of the gray value deviations caused by different parameter values. For window sizes of $\tau < 0.5$ the magnitude of these differences is relatively large in



(a) (b)

Figure 5.8: The line plot in (a) visualizes the RMSE between the filter set ω_γ^0 and the sets of filters generated from different mollifier positions along the central coordinate axes of the volume of interest. The coordinates of the volume have been normalized to the range $[-1, 1]$. A comparison between filtered projections with respect to each projection angle is shown in (b). For the FDK and for the AO-FBP algorithm the filtered projections have been compared with the images produced by P-SIRT after 100 iterations using the RMSE. In both plots the number of SIRT-iterations for the kernel computation were set to 100, while a window size of $\tau^* = 0.73$ was used during the reconstruction with AO-FBP. Note that both graphs have been generated using the SL phantom and an angular projection range of $60°$.

comparison to those generated by other choices of the parameter value. Moreover, one has to consider that smaller differences influence the RMSE proportionally less than larger ones, because the measure is based on squared differences (see equation (5.19)). Consequently, the squaring can amplify larger deviations and weight outliers heavily, such that even if the change in the RMSE is relatively small, the associated variations in the image intensities can introduce significant inaccuracies in the final reconstruction.

5.4.4 Filter Quality

Additionally to the analysis of the algorithmic parameters performed on the reconstructed images in the previous sections, the RMSE has been employed for the evaluation of the quality of the iteratively optimized filters. For this purpose, the geometric setup described in Section 5.4.1 has been used to simulate projections of the SL phantom. For the reconstruction process with AO-FBP the SIRT-iteration count and the optimal window size has been fixed to the values determined in Section 5.4.2 and in Section 5.4.3, respectively.

The initial derivation in Section 5.3.3 showed how the kernels inside the region of interest can be approximated for each point $\overline{x}$ by a single run of SIRT on a given mollifier position. However, the final algorithm makes the simplification that the kernels for all $\overline{x}$ are equal and can be computed by the exploitation of invariances from the central kernel ψ_γ^0 and thus from the central mollifier e_γ^0. By means of this simplification a certain error has to be accepted in the optimized filters and in the final reconstructions as well. To study this approximation error, the set of angle-dependent filters ω_γ^0 from equation (5.12) will be compared to filters constructed from kernels that correspond to mollifiers positioned along the central axes of the volume. For this purpose, the reconstruction kernels discussed in Section 5.3.3 and depicted in Figure 5.3 will be reused in the following evaluation. Equivalently

to equation (5.12) a set of filters, named $\omega_{\gamma,\overline{x}}$, will be computed for the kernels derived from the x-, y- and z-translated mollifiers. Then each of these filter sets will be compared with the central set of filters ω_γ^0 employing the RMSE.

Figure 5.8a shows the results of this comparison. It can be observed that the maximal error, which is made by the kernel approximation, is always smaller than 5.11 %. Notably, the RMSE with respect to the central filters ω_γ^0 and the filters computed from the y-translated mollifiers were never larger than 0.68 %. Moreover, these sets of filters shows a symmetric distribution of the RMSE around the center of the volume. The largest errors between ω_γ^0 and $\omega_{\gamma,\overline{x}}$ occurred with respect to a translation of the kernel optimization process along the x-axis. The RMSE for the x-translated filters were continuously higher than those translated along the other two axes, but it was never larger than 6 %. Moreover, the errors in relation to the x-axis show a slightly non-symmetric behavior, while they increase with further distance from the ISO-center. Contrary to this, a translation of the mollifier into the direction of the X-ray source, along the positive z-axis, increases the RMSE between the central and the translated set of filters. Hereby, the errors in relation to the z-axis approach those of the positive x-axis. However, a translation of the mollifier into the negative z-direction keeps the RMSE between ω_γ^0 and $\omega_{\gamma,\overline{x}}$ consistently smaller than 1 %.

The asymmetric behavior along x- and the z-axis is directly related to the shape of the corresponding shifted reconstruction kernels in relation to the central kernel ψ_γ^0. The variations of the kernels due to the translation of the mollifier depend on the system geometry. With respect to the here used setup, as it has been depicted in Figure 5.1, a translation of the mollifier along the y-axis has a symmetric effect on the resulting kernels and thus on the sets of filters with respect to positive and negative shifts. This symmetry can also be observed in the spectral analysis provided by Figure 5.3b. For a translation of the mollifier along the x-axis one would also expect a symmetric behavior in relation to the central kernel, because of the setup of the CT system with respect to the x-axis. But the Fourier spectrum in Figure 5.3a shows deviations for different mollifier translations, in particular for the projections at $-30°$ and $30°$. These variations could be due to the numerical computation of the x-shifted kernels. Consequently, for the corresponding filters the line plot in Figure 5.8a show an asymmetry with respect to the x-axis. Moreover, it has been shown in Section 5.3.3 that the kernels computed from the z-translated mollifiers behave non-symmetrical. Therefore, the resulting sets of filters and their deviations from the central filters ω_γ^0 result in an asymmetric line plot.

In addition to the study of the central and translated sets of filters, the quality of the filters ω_γ^0 will be quantified during the following analysis by comparing sets of filtered projection images on the basis of the RMSE. Therefore, the relation between the originally measured projections g and a set of "optimally" filtered projections will be exploited as it has been presented and discussed in Appendix F in conjunction with the P-SIRT algorithm. Using the LOW system setup, a set of filtered projections will be computed using the P-SIRT algorithm with 100 iterations as introduced by Kunze et al. [172] and also summarized in Appendix F. The resulting projections from the run of P-SIRT are denoted by $h_P := h_{100}$ and will be used as a reference for the projections filtered with FDK and AO-FBP.

For reasons of readability the projection sets resulting from the latter two algorithms will be named accordingly by h_F and $h_A := \omega_\gamma^0$, respectively.

Having computed the filtered projection sets of all three algorithms successfully, the quality of the filters was measured and studied by the comparison of $RMSE(h_P, h_F)$ and $RMSE(h_P, h_A)$ per projection. Figure 5.8b depicts the resulting values of the RMSEs plotted against the projection angle. The line plots in this figure reveal that the filtered projections generated with the AO-FBP approach have a smaller RMSE with respect to the P-SIRT filtered projection than those resulting from the FDK method. In particular, the deviations between $RMSE(h_P, h_F)$ and $RMSE(h_P, h_A)$ become evident for projection angles smaller than $25°$ and larger than $-25°$, whereby the angles $\lambda \in [-10°, 10°]$ also show clearly visible differences in the RMSE. Furthermore, the mean of the RMSE between the projections filtered with P-SIRT and FDK is $1.40\,\%$, while the average RMSE of h_P and h_A is lower with $1.19\,\%$. Finally, the maximal difference between $RMSE(h_P, h_F)$ and $RMSE(h_P, h_A)$ can be observed at the limits of the angular projection range with a deviation of $0.87\,\%$.

5.4.5 Contrast

In this section the image quality of the AO-FBP algorithm developed above will be compared in terms of contrast and object uniformity to that achieved by the standard FDK algorithm and by SIRT. For this purpose, the evaluation will be based on the CNR, which can be defined as

$$CNR = \frac{|\bar{\mu}_{ROI} - \bar{\mu}_{BG}|}{\sqrt{\sigma_{ROI}^2 + \sigma_{BG}^2}}, \tag{5.20}$$

whereby the average gray value intensities of pixels, or voxels, within a region of interest (ROI) and within a background region adjacent to that ROI are given by $\bar{\mu}_{ROI}$ and $\bar{\mu}_{BG}$, respectively. Moreover, the corresponding standard deviations of the gray values in these ROIs are denoted as σ_{ROI} and σ_{BG}. Commonly, the CNR is used to perform, studies on reconstruction data from noisy projection data, e.g. in [118, 202, 215, 253]. However, in the following studies the projections are simulated without any contribution of additional noise, unlike for example in the experiments conducted in Section 4.4.1. The reason for this is that the main aim of the CNR calculation is the quantification of the image quality of the reconstruction algorithms in the noise-free case. Therefore, it has to be considered that the standard deviations in the denominator of equation (5.20) provide a measure for the non-uniformity introduced by the reconstruction procedure instead of quantifying noise. As a result, the here performed CNR evaluation can be interpreted as a measurement of a contrast-to-artifact ratio.

For the following experiment, an artificially generated contrast-phantom has been employed to perform the evaluations of the CNR with respect to the three algorithms. The phantom contains 8 larger circle-shaped regions, which are positioned around the center of the phantom, and 10 smaller dot-like regions that lie closer to the center, whereby the larger circles have a higher contrast than the

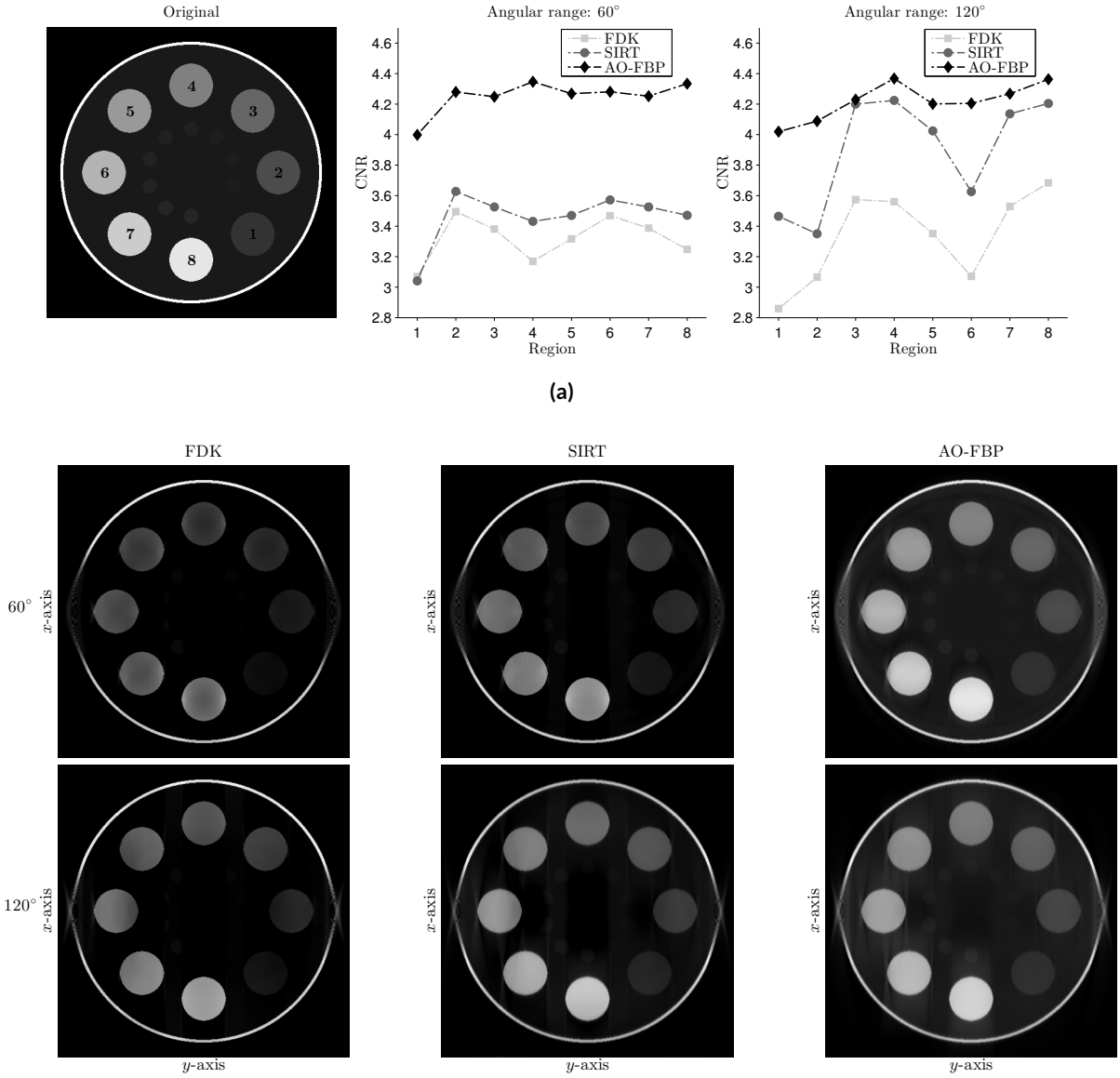


Figure 5.9: In (a), the larger inserts of the contrast-phantom (left) are numbered from 1 to 8 counterclockwise. The line plots in (a) visualize the CNR for each region of interest with respect to each algorithm. The simulations have been carried out for the LOW (central plot) and the HIGH (right plot) angular projection range. In (b), the slice images compare the reconstructions for FDK, SIRT and AO-FBP for both angular ranges using the contrast-phantom. Again, the reconstructed volumes have been normalized to $[0, 1]$ and are displayed with the same range.

smaller ones. The leftmost image in Figure 5.9a displays the central xy-slice of the contrast-phantom, for which the larger ROIs have been numbered in a counterclockwise order from 1 to 8 in the order of their gray value intensities. Similar to the simulations carried out on the SL phantom in the previous sections, the 2D slice image shown in Figure 5.9a has been stacked along the z-axis to generate a 3D volume dataset that is homogeneous with respect to the z-direction.

In accordance with the setups provided in Table 5.1, projections are simulated for the LOW and the HIGH angular scan ranges from the 3D contrast phantom. The detector binning used in the previous

simulations was abandoned to exploit the full resolution of the detector as described in the beginning of Section 5.4. The resolution of the volumes to be reconstructed by FDK, SIRT and AO-FBP was set to $511 \times 511 \times 255$ voxels, whereby the sampling distance between the voxels and thus their edge lengths were fixed to be 0.089 mm. After the reconstruction procedures, for each of the eight high-contrast regions the CNR was computed. Therefore, a circular ROI has been chosen within the each of the eight inserts to compute the corresponding average intensity $\bar{\mu}_{ROI}$ and the standard deviation of the gray values σ_{ROI}. For the calculation of $\bar{\mu}_{BG}$ and σ_{BG} a background area in the shape of a hollow circle has been selected, which has been placed such that it surrounds the eight ROIs during each computation of the CNR for each reconstruction algorithm. Contrary to the eight larger ROIs, the FDK and SIRT were not able to reconstruct the smaller low-contrast inserts of the contrast-phantom correctly. Consequently, these ROIs have not been considered in the numerical evaluation during CNR calculations and are only employed to inspect and to compare the three methods visually using the resulting slice images.

The diagrams at the center and at the right of Figure 5.9a visualize the result of the CNR evaluation for all algorithms and ROIs. The line plots reveal that the CNRs for almost all ROIs and for both angular projection ranges are consistently higher for the AO-FBP and SIRT algorithms than for the FDK method. Only the SIRT algorithm shows a slightly lower CNR than the FDK method for the LOW angular range setup with respect to ROI 1, which has the lowest gray value intensity out of the eight studied regions. Independently of the angular range and for all other regions the FDK algorithm performs inferior to SIRT with respect to image contrast. However, the following similarities between SIRT and the FDK method can be observed: (1) for the LOW system setup both algorithms achieve higher CNR for the ROIs 2 and 6, which are positioned along the y-axis, i.e. the axis of rotation, while (2) for the HIGH angular range both techniques deliver a superior contrast in regions that are not placed on the rotation axis. The latter observation holds true for almost all inserts in the 120° case, so that the CNRs of the ROIs 3, 4, 7 and 8 are significantly higher than those of the ROIs 2 and 6. The exception occurs for region 1, which lies not on the y-axis but does not consistently perform better than the axial regions 2 and 6. In principle, the lines connecting the CNRs of all ROIs plotted in Figure 5.9a show similar patterns for SIRT and the FDK method in relation to the studied low- and high-contrast regions. Consequently, it can be assumed that both algorithms react similar to different levels of contrast in the images. Nevertheless, it has to be kept in mind that SIRT generally delivers a superior CNR, particularly for the 120° case.

Comparing the quality measurements of the AO-FBP approach presented in Figure 5.9a with those of the FDK algorithm and SIRT shows that the optimized filters achieve a consistently higher CNR than the other algorithms. This observation can be made independently from the studied ROI and from the used angular scan range. Moreover, it can be seen for the AO-FBP that the values of the CNR lie closer to each other for different ROIs. Consequently, this behavior suggests that for the angle-optimized approach the resulting image contrast is not that strongly depending on the location of the regions of interest within the phantom. Additionally, the average CNR has been calculated over the eight regions together with the corresponding minima, maxima and standard deviations.

Table 5.2: Contrast phantom CNR-results accumulated over all ROIs

Ang. Range	**LOW (60°)**			**HIGH (120°)**		
Algortihm	**FDK**	**SIRT**	**AO-FBP**	**FDK**	**SIRT**	**AO-FBP**
Mean	3.32	3.46	4.25	3.34	3.90	4.22
Std. Dev.	0.15	0.18	0.11	0.30	0.36	0.12
Min	3.07	3.04	4.00	2.86	3.35	4.02
Max	3.49	3.63	4.35	3.68	4.22	4.37

The resulting statistics are presented in Table 5.2 and can be used to underline and verify the observations made on the line plots above numerically. The table shows that the FDK algorithm delivers the lowest average CNR for the LOW and the HIGH projection ranges. Slightly higher mean CNRs are reached by SIRT, whereby AO-FBP outperforms both algorithms significantly with average values higher than 4.2 for both system setups. In addition to this, the angle optimization approach has the smallest standard deviation with respect to the CNR. Hereby, the standard deviations of SIRT are even slightly higher than those resulting from the FDK algorithm. As a consequence, it can be noted that although the SIRT algorithm has a higher standard deviation with respect to the eight ROIs for both setups, its mean CNR is lower than that of the FDK algorithm. The last two rows of Table 5.2 also demonstrate that the here derived optimized FBP approach achieves CNRs consistently higher than a value of 4.0. Contrary to this, the minimal CNRs of SIRT and the FDK method are 3.04 and 2.86, respectively, while only SIRT reaches a CNR higher than 4.0 for some regions using the HIGH angular range (compare Figure 5.9a). Nevertheless, on the one hand, an increase in the calculated mean CNRs can be observed for FDK and SIRT when comparing the HIGH with the LOW angular projection range. On the other hand, the average CNR is almost the same for both system setups in the AO-FBP case. Furthermore, for all three approaches the standard deviations of the CNR increase with the angular range.

For the visual inspection of the image quality and the contrast achieved by the different approaches Figure 5.9b depicts the central slices of the contrast-phantom parallel to the xy-plane. The columns of the figure show the reconstructions of the FDK, SIRT and AO-FBP approaches, respectively, while the rows correspond to the LOW and HIGH angular projection range. As mentioned in the introduction of this section, the smaller low-contrast ROIs are used to visually judge the overall reconstruction quality. With respect to the FDK algorithm these regions are hardly visible in the reconstructed images for both angular setups. Anyhow, SIRT was able to restore parts of the smaller circles, while only the angle-optimized reconstruction technique produced images for both angular ranges that allowed distinguishing the low-contrast regions from the background of the phantom. In summary, it can be stated that the contrast and thus the visibility of the smaller circles increases from the FDK method, to SIRT and to the AO-FBP. Additionally, an image improvement can be registered from the $60°$ to the $120°$ cases. However, in images with higher contrast the prominence of streaking and shadowing artifacts increases. These artifacts are caused by the limited amount of angular information available

during the reconstruction process (see Section 5.2.1 and Section 5.3.2). These effects can be noticed particularly for the LOW reconstruction with AO-FBP and for the images computed from the HIGH setup using SIRT and AO-FBP. Note that the increased contrast in these cases has been demonstrated in Figure 5.9a and in Table 5.2.

5.4.6 Artifact Spread

It has been demonstrated in [177] that the total scan range A_r and the in-plane size of the feature of interest mainly influence the resolution in the z-direction perpendicular to the xy-plane. Consequently, the result of a limited angular projection range is that the reconstructed images are interfered by ghosting artifacts, in particular along the z-axis. Hereby, the slices right next or close to the inspected structures are more severely affected by the artifacts than slices further away from the studied feature. In the publication of [215] the ASF has been proposed for the quantification of the ghosting artifact distribution. The ASF can be calculated by the following expression:

$$ASF(z) = \frac{\bar{\mu}_A(z) - \bar{\mu}_{BG}(z)}{\bar{\mu}_F(z_0) - \bar{\mu}_{BG}(z_0)}, \tag{5.21}$$

whereby z_0 defines the location of the in-focus plan of the feature of interest along the z-direction and the parameter z represents the location of off-focused planes containing the artifacts caused by the feature due to the limited scan angle. In equation (5.21) the average gray values of the feature and the mean intensities of the background with respect to the in-focus plane z_0 are given $\bar{\mu}_F(z_0)$ and $\bar{\mu}_{BG}(z_0)$, respectively. For an off-focus plane at a location of z the average intensities of the ghosting artifacts and the background are denoted by $\bar{\mu}_A(z)$ and $\bar{\mu}_{BG}(z)$, respectively. Note that by its definition the ASF function value at z_0 equals 1, i.e. $ASF(z_0) = 1.0$, because $\bar{\mu}_A(z_0) = \bar{\mu}_F(z_0)$.

For the calculation of the ASF simulated projections were generated from a phantom which consists of a 5×5 grid of solid spheres, which are distributed in the xy-plane. The spacing between the spheres is equal for the x- and y-direction and amounts to 5.356 mm, while each sphere is having a diameter of 1.161 mm. The projection simulations have been carried out for the LOW and the HIGH angular range setup with respect to the system geometry specified in Section 5.3.1 and in Section 5.4. Moreover, the detector binning has also been disabled to employ its full resolution. Finally, volumes are reconstructed from simulated data using FDK, AO-FBP and SIRT. The geometric parameters for the volumes of interest are set to the same values as for the evaluation of the image contrast in the previous section. The central slices parallel to the xz-plane of the resulting reconstructions are depicted in Figure 5.10a. A visual inspection of this figure reveals that the ghosting artifacts for SIRT and AO-FBP have a more blurry appearance, although they have a similar overall shape as those produced by the FDK algorithm. Moreover, it can be seen that the spread of the artifacts for the $120°$ case is less wide with respect to the z-direction and less intensive in comparison to the $60°$ case.

In the upper left image of Figure 5.10a three of the five ball objects have been marked by B_1, B_2 and B_3, respectively. The evaluation of the ASF has been performed for each of the three spheres in relation to the six displayed combinations between employed reconstruction techniques and angular scan

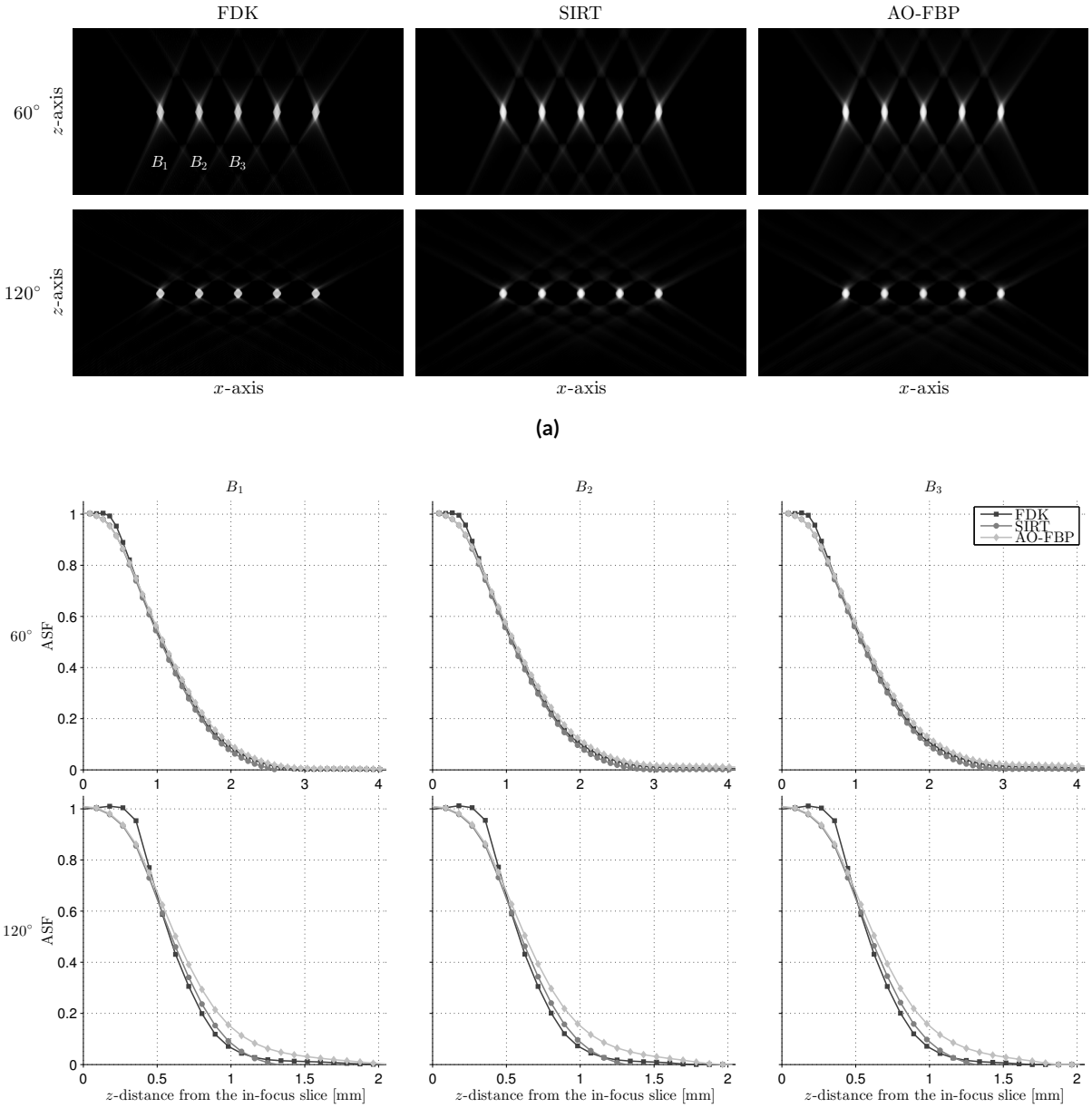


Figure 5.10: The slice images in (a) show the central xz-plane of the ball phantom reconstructed with FDK, SIRT and AO-FBP for the $60°$ and the $120°$ case. The ASF has been calculated for the three spheres marked with B_1, B_2 and B_3 in the upper left image of (a) in relation to each algorithm and both angular projection ranges. The plots in (b) show the resulting ASFs, whereby each of the three balls is plotted in the first second and third column, respectively. The $60°$ cases are shown in the first row and the $120°$ cases are displayed in the plots of the second row.

configuration. The mean values $\bar{\mu}_F(z_0)$ and $\bar{\mu}_A(z)$ were calculated accordingly to equation (5.21) from regions within the balls and their artifacts along the z-axis, respectively. The size of these ROIs was chosen to be 8×8 pixels, so that they did not exceed the boundaries of the spheres. Similar to the CNR calculation, regions in the shape of a hollow circle lying outside of the object of interest at z_0 and its ghosting artifacts were used to compute the corresponding average background inten-

sities $\bar{\mu}_{BG}(z_0)$ and $\bar{\mu}_{BG}(z)$. The plots in Figure 5.10b display the resulting ASFs for both angular projection ranges in relation to each of the studied balls B_1, B_2 and B_3.

The comparison of the plots of Figure 5.10b in relation to the three employed reconstruction techniques demonstrates that SIRT and AO-FBP result in an ASF, which is for smaller distances from the in-focus plane lower than that produced by the FDK algorithm. This behavior can be observed independently from the inspected sphere and is not changed by the angular projection range. In particular, for the LOW scan protocol the ASF of the FDK method is higher than that of SIRT and AO-FBP up to a value of $z \approx 0.75\,\text{mm}$. Furthermore, the ASF of the FDK reconstruction lies continuously above that of SIRT, even for values of $z > 0.75\,\text{mm}$. A contrary observation can be made for the 120° case: for values of $z < 0.5\,\text{mm}$ the ASFs of SIRT and AO-FBP lie below that of the FDK algorithm, while for values larger than $0.5\,\text{mm}$ FDK outperforms the other two reconstruction approaches. Nevertheless, for increasing distances z from the in-focus plane the ASF of the FDK reconstruction lies above that of AO-FBP and SIRT again (see bottom row of Figure 5.10b). Additionally, the largest deviations between the ASF curves can be noticed for the planes close to the in-focus object. In these intervals the AO-FBP algorithm performs almost equal to SIRT. Consequently, the distribution of the ghosting artifacts along the z-axis is similar for both algorithms and significantly narrower than that of the FDK method. In particular for the 60° case, the ASFs of all three algorithms behave quite similar for planes further away from the in-focus object. However, for the HIGH angular range setup SIRT performs worse than the FDK algorithm in the interval $0.5\,\text{mm} < z < 1.25\,\text{mm}$. In addition to this, the ASF of the AO-FBP lies above that of the FDK algorithm over an even wider range of $0.5\,\text{mm} < z < 2.5\,\text{mm}$. Moreover, it can be noticed for both angular range protocols that the ASFs of the three balls B_1, B_2 and B_3 are not showing any significant deviations with respect to the three algorithms. Note that evaluations of the ASF with respect to other ball objects inside the given phantom showed similar results to the ones pretested above.

Besides the observations above, the ASF curves in Figure 5.10b show an approximately equal full width at half maximum (FWHM) with respect to the three algorithms, whereby differences occur for the two angular scan ranges. For the LOW acquisition protocol a FWHM of about $1.1\,\text{mm}$ can be estimated for all the algorithms, whereas the FWHM for the HIGH protocol is approximately at $0.6\,\text{mm}$. Consequently, an increase of the angular projection range by a factor of 2 has almost halved the spread of the ghosting artifacts along the z-axis. Because of the facts mentioned above, it can be concluded that the here developed projection-wise filter optimization strategy has a superior artifact reduction for planes that lie close to the in-focus slice.

5.4.7 Real Data

For the demonstration and the analysis of the results from the filter optimization algorithm on real projection data, a board of a graphics card including the chipset has been scanned with a μCT system. The scan was carried out using the FULL acquisition protocol in accordance with Table 5.1, whereby the full defector resolution was employed (see beginning of Section 5.4). Afterwards, the limited-

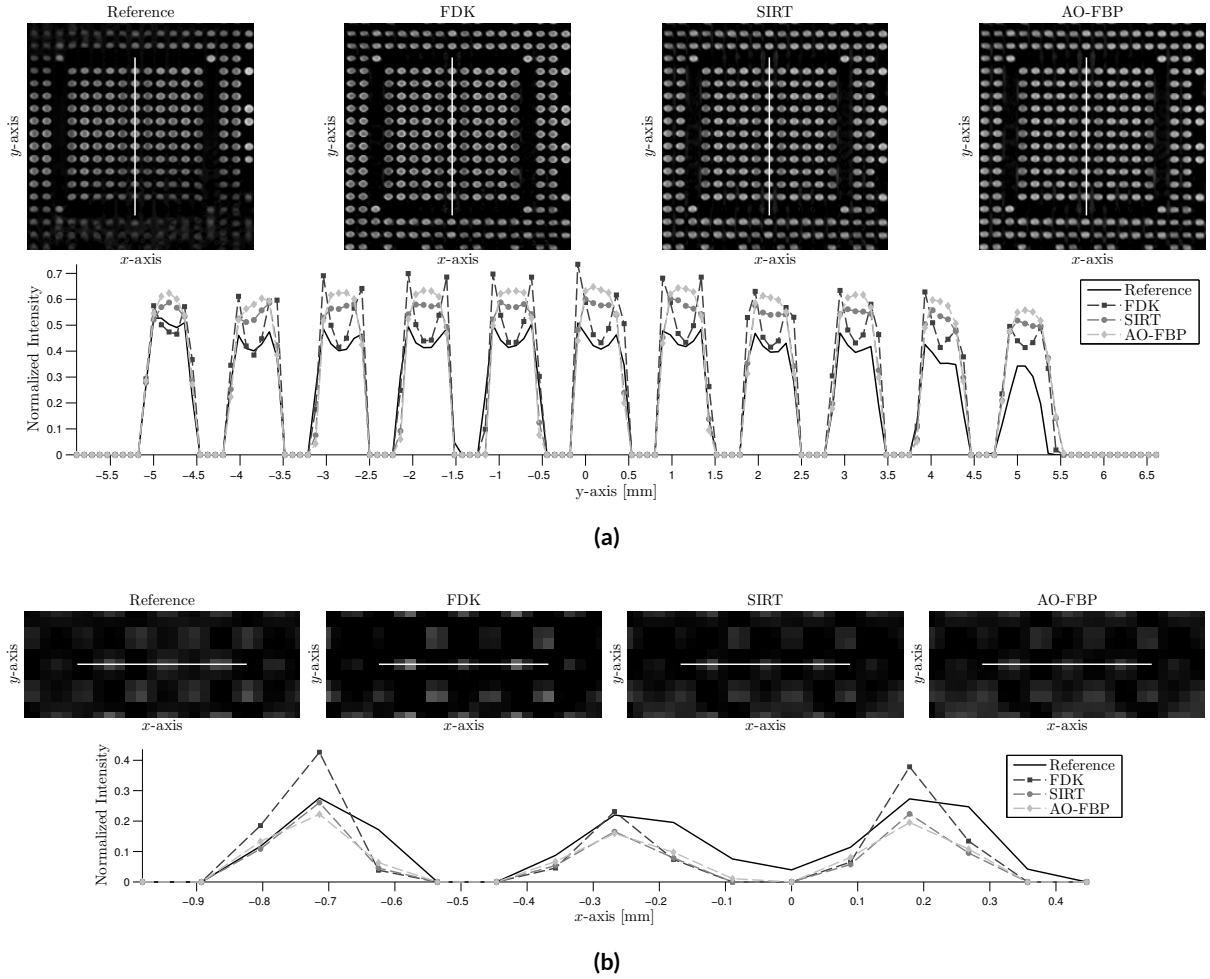


Figure 5.11: The images in (a) depict the central xy-slices of the volumes reconstructed with the FULL FDK, LOW FDK, LOW SIRT and LOW AO-FBP protocols. A comparison of the four reconstructions with respect to the white lines in the four slice images is given in the line plot of (a). Regions containing small point-like structures are depicted in the four images of (b) in relation to each protocol. The profile curves in (b) correspond to the white lines in the images and provide an indicator for the resolution that can be achieved by the four protocols. Note that all volumes are normalized and displayed in the range $[0, 1]$

angle projection data for the 60° and the 120° case was generated selecting the corresponding subsets of projection from the FULL dataset. Then, the reconstructions with FDK, SIRT and AO-FBP were computed from the LOW and HIGH projection sets. For these three reconstructions the full 360° scan reconstructed with the FDK algorithm has been used as a reference.

The results of this comparison with respect to the 60° projection data are shown in Figure 5.11a by the visualization of the central reconstructed xy-slices that lie orthogonal to the z-axis. The line profiles in the lower part of the figure correspond to the white lines drawn in the slice images. A look onto the profile curves reveals that the FDK reconstructions from the FULL and from the LOW acquisition protocol contain severe cupping artifacts at the boundaries of the solder joints inside the BGA. These artifacts are stronger for the limited-angle scan, whereas they do not occur in the reconstructions computed with SIRT and with the AO-FBP approach. Nevertheless, the images produced

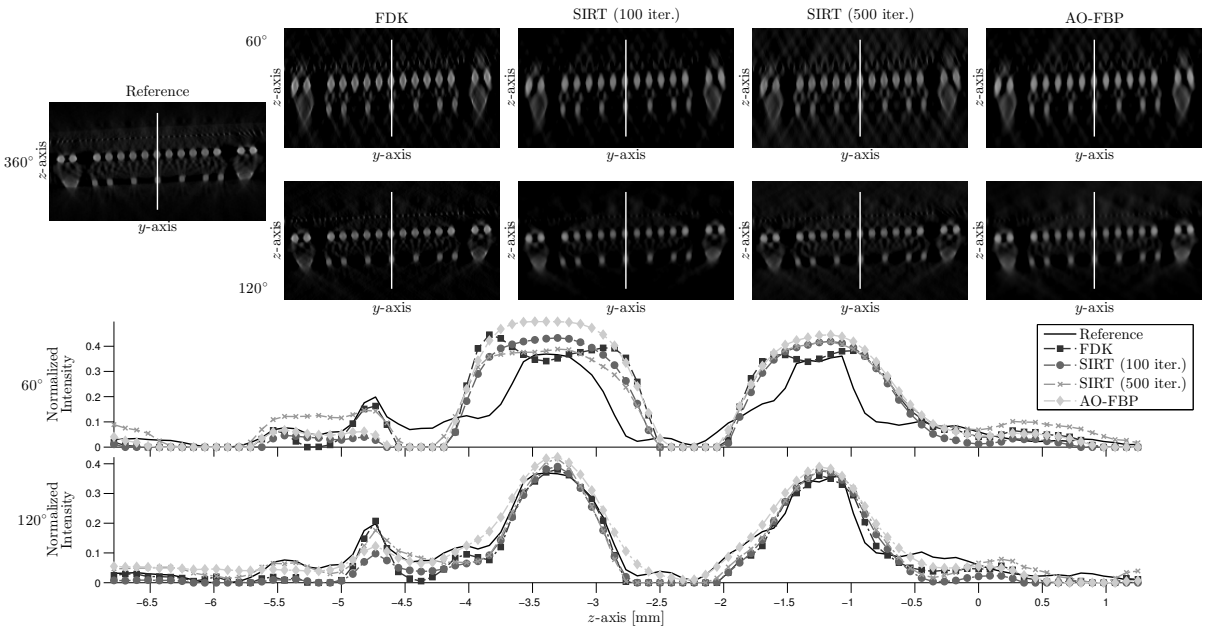


Figure 5.12: The images in the upper part show the central slices in the xz-plane of the reconstructions with the LOW (top row) and the HIGH (second row) protocols for the FDK, SIRT and AO-FBP. The reference image (upper left) has been reconstructed with the FULL FDK setup. For the $60°$ case (upper plot) and the $120°$ case (lower plot) the profile curves for each algorithm are drawn along the corresponding white lines in the slice images. All volumes are normalized to $[0, 1]$ and are displayed with this range.

by the two FDK protocols (i.e. Reference and LOW FDK) have a slightly sharper appearance than the central slice images reconstructed by SIRT and AO-FBP. This behavior can be verified when looking at the profile curves, where the AO-FBP has an even smoother shape than SIRT with respect to each ball. Additionally, since the object borders in the FDK reconstructions are amplified by the cupping effect, the impression that the FDK algorithm produces sharper results than the other two algorithms is even strengthened. However, the real BGA contains solid and homogeneous metal objects, for which a profile plot would show peaks with flat tops. In Figure 5.11a such a behavior can only be observed for some peaks in the reference reconstruction (at $y \approx -5.0\,\text{mm}$) and for SIRT, where balls closer to the center of the BGA show more flattened peaks.

Furthermore, Figure 5.11b provides a comparison of the three algorithms with respect to the resolution that they achieve for small contrast features lying in the xy-plane. For the $60°$ case the profile plots show that the reconstructions of the FDK algorithm result in the highest peaks with the steepest edges at all three feature locations. On the contrary, the profiles of SIRT and AO-FBP are even lower than those of the reference image, so that they appear more flat, but are quite similar in their shape. In contrast to this, it can be noticed that the FDK algorithm creates a profile plot that overshoots that of the $360°$ reference image for two of the studied features. Additionally, SIRT and AO-FBP produce lines plots that have sharper edges than that of the reference reconstruction.

Additionally to the evaluation of the reconstruction quality with respect to the xy-plane, an analysis of the projection-wise optimized FBP method has been conducted to study its image quality and

artifact behavior along the z-axis. This study has also been performed considering all three algorithms for both angular range setups, whereby the 360° scan served as reference again. The resulting slice images and profile curves are displayed in Figure 5.12, whereby additional results are depicted for SIRT after 500 iterations. From these images, it can be observed that the overall image quality of SIRT has been improved significantly from iteration 100 to iteration 500. This fact suggests that the SIRT approach has not been fully converged for the here studied GPU dataset after 100 iterations. Anyway, the slice images and profiles of the 100-iteration SIRT are shown here for the sake of comparison with the AO-FBP, since the filter optimization uses the same number of iterations to estimate the reconstruction kernel (see Section 5.4.2). For this reason, a comparison between the 100-iteration SIRT and AO-FBP seems to be reasonable and fairer.

For the LOW acquisition protocol, the FDK algorithm produces cupping artifacts with respect to the large structures (i.e. the solder joints of the BGA), which become visible in the slice and in the profile view. These artifacts have already been registered for the simulated datasets in the previous section. Additionally and similar to the simulation experiments, the reconstruction of the FDK appears sharper, while the slice views of SIRT and AO-FBP look more blurry. This behavior can also be noticed when comparing the profile plot of the three algorithms. These curves also reveal that after 500 iterations the SIRT algorithm approaches the reference lines at the larger masses more closely than the other line plots. However, the 100-iteration SIRT and the AO-FBP resemble the hull of the graphics chip more accurately than the other methods when comparing their reconstructed profiles against the reference curve in the interval $-6.0\,\text{mm} < z < -5.0\,\text{mm}$. In contrast to this, the AO-FBP and the SIRT algorithm after 100 iterations are not able to restore to wire structure in the area $-5\,\text{mm} < z - 4.5\,\text{mm}$ as precisely as the FDK and the 500-iteration SIRT.

For the HIGH angular range protocol, similar observation can be made. Again, the reconstructions of the FDK algorithms are sharper than those of SIRT and AO-FBP. Nevertheless, the SIRT method was capable to reconstruct small structures after 500 iterations. Additionally, the overall appearance of the reconstructions after 500 iterations became sharper for the 120° case and was similar to that of the FDK algorithm. In summary, it can be said that the projection-wise filter optimization strategy derived within this chapter has a behavior that is similar to that of SIRT. Consequently, the AO-FBP delivers superior results with respect to low intensity background areas, as it has been shown for the hull of the chip. Contrary to this, the FDK method is able to recover finer structures, such as the wires, more accurately. Additionally, it has to be kept in mind that SIRT is capable to produce sharper reconstruction results with smaller features becoming detectable, though the number of iterations has to be increased considerably. As a consequence, SIRT has a significantly longer runtime, but can approach the reference reconstruction more precisely than any other algorithm. For 120° setup this behavior has been observed after 500 iterations. A contrary observation has been made for AO-FBP in the preliminary studies (compare Section 5.4.2): a higher number of SIRT-iterations during the calculation of the central reconstruction kernel has not led to a significant quality improvement in the images reconstructed with AO-FBP.

One important fact has to be considered when looking at the above analysis, in particular with respect to Figure 5.11a: the FDK reconstructed image for the $60°$ case has an appearance superior to the reference image computed from the $360°$ projection data, although the former reconstruction uses less information. In principle, an explanation for this effect can be found by taking a look the yz-slice images in Figure 5.12, which correspond to the views of the xy-plane in Figure 5.11a. On the one hand, the FULL reference FDK reconstruction contains ball objects that are almost perfectly round with respect to the yz-plane. On the other hand, for the $60°$ protocol the FDK method produces ghosting artifacts, as discussed in Section 5.4.6, with respect to the z-axis. The profile plots of Figure 5.12 can be employed to verify and to quantify this behavior. From these plots, it can be noticed that the profiles of the solder beads for the limited-angle configuration are wider than those of the non-limited-angle case. Moreover, it becomes evident from slice images drawn parallel to the yz-axis that in the reconstructed volume the BGA is not exactly aligned with the horizontal xy-plane, because the solder spheres in Figure 5.12 are slightly angled and thus not parallel to the y-axis. Consequently, the minor misalignment of the BGA becomes also visible in the xy-slices displayed in Figure 5.11a, but is more difficult to interpret. Because the object of interest is slightly rotated around the x-axis, the solder beads are intersected by the central xy-slices of Figure 5.11a at different locations, such that balls located along the positive y-axis are sliced at their lower part and balls along the negative y-axis are cut by the slice at their upper part. Furthermore, with Figure 5.11a it has been demonstrated above that the limited-angle FDK reconstruction results in an artifact spread along the z-axis. As a result of the wider artifact distribution in the $60°$ case, the solder balls are intersected by the central xy-slice in conjunction with their ghosting artifacts, which let the beads appear more intensively in the slice images. This is the main reason why the solder balls seem to be reconstructed more precisely in images computed from the LOW acquisition protocol. As a consequence, the whole $60°$ FDK slice image Figure 5.11a looks superior to the reference reconstruction.

Note that the observations from the previous paragraph can also be verified by the profile plots of Figure 5.11a drawn along the y-axis, where the rotation of the BGA is reflected by the decreasing peak height for further distances from the ISO-center ($y = 0$). In summary, it can be stated that the appearance of the reference reconstruction seems to be inferior to the one of the LOW FDK algorithm for basically two reasons: (1) the solder beads of the GPU dataset are slightly misaligned with the y-axis and (2) the $60°$ limited-angle reconstruction contains ghosting artifacts that spread along the z-axis.

5.4.8 Runtime Performance

The runtime of the FDK, AO-FBP and of SIRT have been measured during the experiments conducted with the simulations and real projection data of the graphics board. The measurements were carried out on a workstation equipped with an Intel Core i7 3820 3.60 GHz processor and a NVIDIA GeForce GTX 680. All filter computations have been run inside MATLAB 2013b and the low level optimizations, such as those for the forward and backprojection operators, have been realized using NVIDIA's CUDA 4.2 (see Chapter 2 for further details).

For the GPU dataset a SIRT reconstruction with 100 iterations took on this machine an average time of 14.59 seconds with respect to the LOW angular range setup and 29.19-seconds for the HIGH image acquisition protocol. For the FDK and AO-FBP algorithm the runtimes were, as expected, significantly lower. A single run of the FDK method using MATLAB's *fft* and *ifft* functions for the 1D line-wise filtering and the CUDA-based backprojection took in the mean 0.55 seconds for the 60° case and averagely 1.11 seconds for the 120° case. Moreover, the runtime performance of the AO-FBP algorithm was even superior to that of the FDK approach. The execution of the AO-FBP on the 60 projections took only 0.11 seconds on average, while the reconstruction from 120 projections was carried out within 0.21 seconds. Since FDK and AO-FBP employ the same backprojection operator the main reason for the higher performance of AO-FBP has to lie within the filtering procedure.

For this reason, it has to be considered that the FDK filter uses the 1D fast Fourier transform (FFT) called in a loop for each line of each projection, whereas the AO-FBP employs the 2D FFT for Fourier domain filtering each projection. Although both FFT operations are optimized in MATLAB internally by the use of the Intel Performance Primitives (IPP), the loop over the projection lines in the FDK algorithm could be the cause for the performance drop, since MATLAB's strengths are matrix operations and not loop optimizations. Moreover, it has to be noticed that the FDK realization used for the experiments is filtering each projection row sequential, such that a further increase of the overall computational performance can be expected by exploiting the possible concurrency of these operations. With the parallelization of the FDK on multi-core hardware, it can be assumed that runtimes in the same order of magnitude as those of the AO-FBP method can be achieved.

Additionally, the calculation of the central reconstruction kernel using 100 iterations took the same amount of time as the execution of SIRT on the projection data. However, this kernel and thus the resulting filters can be reused for different datasets, whereas a full run of SIRT has to be performed on each projection set individually. Further speed improvements for the kernel computation could be reached by the employment of optimization methods different from SIRT. Nevertheless, a study of such approaches in conjunction with the here introduced AO-FBP algorithm is beyond the scope of this thesis and is planned for future work (see also Section 6.2).

5.5 Discussion

The analysis of the reconstruction results presented in Section 5.4 demonstrated the novel projection-wise filter optimization strategy based on the AI has the capability to improve the overall image quality in the area of limited-angle CBCT. Moreover, the selection of the algorithm-intrinsic parameters has been shown experimentally using the RMSE. By this means, the iteration count of SIRT during the kernel computations has been fixed to 100, since further iterations did not improve the image quality significantly (see Section 5.4.2). Additionally, the window size parameter τ, as introduced in Section 5.3.4, has been optimized using a line search in conjunction with a polynomial fitting procedure (see Section 5.4.3). Nevertheless, during these and further evaluations the AO-FBP algorithm showed a significantly smaller RMSE than the FDK method (compare Section 5.4.1 et seq.), whereby

the determination of the number of iterations to 100 and the selection of the optimal window size of reduced the RMSE even further.

The superior reconstruction performance of the AO-FBP approach for low-intensity regions is one of the main reasons for its lower RMSE. The comparison of FDK and AO-FBP conducted on slice and profile plots of the SL phantom verified this behavior (see Figure 5.6b). In particular at interior regions of the SL phantom, the AO-FBP reconstructions showed considerably smaller differences with respect to the reference image than the FDK method. The same behavior has also been observed for the reconstruction from the real projection sets of the GPU, where the angle-optimized filters were able to restore regions containing low gray value intensities (e.g. the hull of the chip) better than the FDK algorithm. To gain a deeper insight on the behavior of the constructed filters, Section 5.4.4 performed a comparison between the filters used by the FDK (i.e. the Ram-Lak filter) and the AO-FBP filters. The results of this comparison have demonstrated that the filters developed in Section 5.3 approximate the Moore-Penrose-pseudo-inverse of AA^* (see Appendix F for an introduction) better than the Ram-Lak filter. Consequently, these observations suggest that the projection-wise set of angle-dependent filters lies closer to the minimum of the L2-norm of equation (F.3) than the FDK-filter. This fact is also the reason why SIRT and AO-FBP share similar properties with respect to the quality of the reconstructed volumes.

During the further experiments studying the CNR (see Section 5.4.5), the reconstructions computed with AO-FBP showed a significantly improved performance in relation to image contrast and object uniformity in relation to FDK and SIRT. Hereby, the CNR of AO-FBP has shown to be consistently higher than those of the other two reconstruction techniques, whereby this behavior has been occurred independently from the employed acquisition protocol. The CNR of AO-FBP was never smaller than a value of 4.0, which was reached at ROI 1 for both angular projection ranges. This ROI has the lowest deviation from the background area inside the contrast phantom. In general, AO-FBP reached a higher CNR for ROIs 4 and 8 that lay further away from the rotation axis and have larger gray values than ROI 1 (see Figure 5.9a). On the contrary, for the $60°$ limited-angle reconstruction it has been noticed that SIRT and the FDK method achieved the highest CNRs for ROIs 2 and 6 that lie on the rotation axis. However, for the HIGH acquisition protocol they achieve superior CNRs for regions 3, 4, 7 and 8 that are not lying on the central y-axis. In addition to this, the visual inspection of the smaller ROIs with lower contrast, which are positioned closer to the center of the phantom, demonstrated that the AO-FBP algorithm reconstructed these structures more accurately than FDK and SIRT. Moreover, for the $120°$ case the slice images of SIRT and AO-FBP have more streak-like ghosting artifacts, which are caused by the limited amount of projection information due to the angular range, than the FDK algorithm. In summary, it registered the optimized reconstruction method proposed in Section 5.3 shows better contrast characteristics than the two other approaches. Nevertheless, the downside of the improvement of the image contrast, which is similar to SIRT, is that streaking artifacts are not canceled out as strong as with the FDK method and therefore lower the overall appearance of the reconstructed images (compare Figure 5.9b).

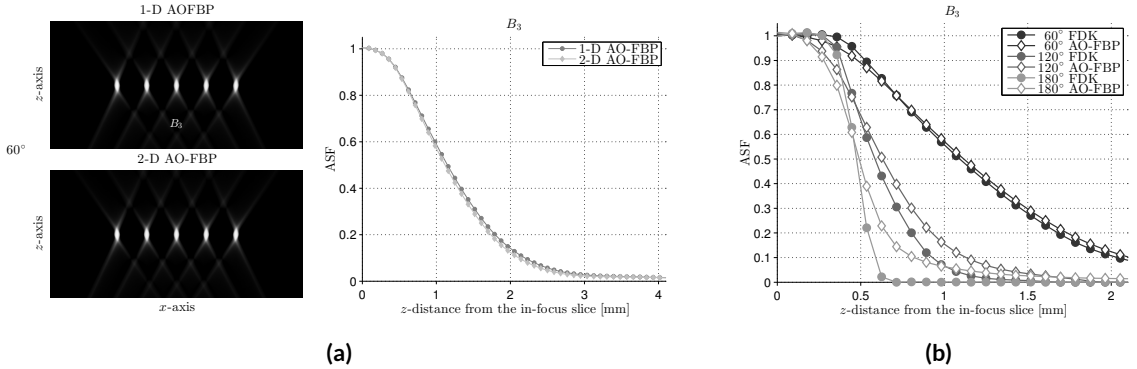


Figure 5.13: A comparison of the central slices in the xz-plane of reconstructions computed with 1D AO-FBP (upper) and with 2D AO-FBP (lower) is given in (b). The 1D filters have been constructed from the transaxial component of the 2D filters, whereby the angular projection range was fixed to $60°$. The ASF with respect to the central sphere of the phantom (B_3) is depicted in the plot on the right of (a) for the 1D and 2D AO-FBP. The plot in (b) compares the ASFs of FDK and AO-FBP with respect to B_3 for three different angular ranges: $60°$, $120°$ and $180°$.

In Section 5.4.6 the observations made for the CNR have been complemented by a quantification of the streaking artifacts along the z-axis by the use of the ASF. Moreover, the ASF of the AO-FBP at slices nearer to the in-focus plane ($z \lessapprox 0.5\,\text{mm}$) was for both acquisition protocols lower than that of the FDK, while its progression in this area was similar to that of SIRT. In the $60°$ case, the behavior of the three reconstruction approaches became more and more similar with increasing distances from the in-focus plane. However, in the $120°$ case, the FDK reconstruction showed a minor improvement of the ASF for $z > 0.5\,\text{mm}$. In Section 5.4.7 the reconstruction quality in relation to the z-direction has also been evaluated visually on the real projection sets. Hereby, SIRT and the AO-FBP approach had again a quite a similar behavior for the LOW system setup, where their profile plot along the z-axis were more flat at the peaks and represented the solder beads more accurately than those of the FDK method. Actually, the FDK algorithm produced cupping artifacts for the $60°$ system setup, whereas the other methods did not. Contrary to this, a better resemblance of the reference profile curve was achieved by the FDK when using 120 projections from the wider angular range. All the observations made for the artifact spread along the z-direction suggest that with a wider angular range SIRT and AO-FBP will become outperformed by the FDK algorithm with respect to the ASF. Nevertheless, for a decreasing angular projection range the here developed filter optimization approach will together with SIRT result in a superior reconstruction quality with a lower amount of ghosting artifacts. Since real projection data for a wider angular range is available because of the conducted FULL scan, Figure 5.13b visualizes the just proposed conjecture. It can be observed that for a reconstruction from $180°$ the FDK algorithm generates an ASF that has a significantly steeper edge as that achieved by AO-FBP.

Comparing other currently existing FBP approaches that exploit the use of projection-dependent filters for the optimization of limit-angle CBCT to the method developed within this chapter will point out the similarities and the differences between these techniques. As mentioned in the intro-

duction of this chapter and in Section 5.2, Kunze et al. [172] have developed a method that estimates the impulse response of the CT system from the scan of a thin wire phantom. They have shown that the resulting filters are dependent on the projection angle. A quite similar set of filters has been derived by Nielsen et al. [183], whereby their method did not require the scan of a point or wire like object, but calculated the filters by the analysis of the pseudo-inverse of a specific tomosynthesis system. Both approaches have in common that the resulting projection-wise filters are 1D and can be applied to each row of the detector equally within a FBP-type scheme. Contrary to this, the AO-FBP method computes a set of 2D filters from the reconstruction kernels. Other than the 1D filters in [172] and [183], the angle-dependent filters employed by the AO-FBP incorporate additional detector column information, which lies parallel to rotation axis, into the filtering operation. Therefore, the AO-FBP filters can be interpreted as the 2D impulse response of the limited-angle CT scanner. A comparison of slice images and ASFs that have been generated from volumes reconstructed with the standard 2D AO-FBP approach and with a modified version that employs a 1D set of filters generated from the transaxial component of the 2D filters is displayed in Figure 5.13a. For both methods the slice images have an almost equal appearance, whereby the 2D set of filters results in an ASF that lies slightly below that of the 1D filters. Consequently, it can be assumed that the axial filtering component that is present in 2D AO-FBP method improves the image quality with respect to artifact spread along the z-axis. Although this improvement is not very large, it is obtained automatically when constructing the 2D filters as described in Section 5.3. However, this short study makes clear that the here presented method can be employed to construct optimized sets of 1D and 2D filters for limited-angle CT. Further detailed studies will are needed to find out which role the axial and the transaxial components of the 2D filter play and in which situations the axial part of the filters are necessary and when they can be neglected.

Additionally, the optimization method of Kunze et al., called OFBP, requires an additional scan of a wire phantom to determine the 1D impulse response during the filter estimation process. In contrast to this, the AO-FBP method does not need a separate scan of a dedicated object and uses only the provided information about the system setup to calculate the kernel and the filters. On the one hand, this fact can constitute an advantage in situation where no wire- or point-like object is available at the time of scanning or where a reference scan of such a phantom has not been carried out, e.g. for projection data that has been scanned some time ago, such that an estimation of the impulse response using the method of [172] is not possible. On the other hand, the use of a real measurement of the wire phantom during the filter optimization process as suggested by Kunze et al. incorporates the intrinsic properties of the CT system automatically into the resulting filters, whereas the AO-FBP only allows the compensation of the aspects of the CT scanner simulated during the kernel calculation. Therefore, one can expect that the here introduced approach greatly benefits from improved forward and backprojection operators that model the CT system more accurately. Moreover, when the real measured projection data of the wire in the case of Kunze et al. is replaced by the simulated projections of the mollifier, it can be seen that the basic theory behind the AO-FBP and the OFBP are closely related. This fact also becomes evident when comparing the SIRT-update equations from

Section 5.3.3 and from Appendix F with the ones described in [172]. For this reason, it should be possible to apply the OFBP together with its filter calculation technique to simulated data, without the need of a wire scan. Nevertheless, such studies have not been conducted so far and are beyond the scope of this thesis, but can provide more details on the connections between both approaches. Additionally, the studies performed on clinical data using the OFBP algorithm in [172] and the technique of Nielsen et al. [183] achieved similar results to the ones presented in Section 5.4. In accordance with these findings, it has been demonstrated that a projection-wise optimized angle-dependent FBP resembles the results of SIRT quite closely and that these approaches can generate more homogeneous images than the standard FDK algorithm.

For the evaluations in the previous section, one has to consider that the SL phantom and the contrast-phantom have been generated by stacking equal 2D slice images parallel to the xy-plane along the z-axis that is orthogonal to the rotation axis, i.e. the y-axis (compare Section 5.3.1). This fact can be irritating, especially when looking at the slices presented in Figure 5.6b and in Figure 5.9b, because the quality of these reconstructions is by far superior to that shown in other limited-angle CT publications (see [242, 254–257]). Nevertheless, in this literature the phantoms are aligned orthogonally with the rotation axis of the system, such that in the case of the here used system setup the 2D slice images would have been stacked along the y-axis to construct the final 3D phantoms. As a result, the reason for the deviations between the images shown in the experiments and those reconstructions known from tomosynthesis or other clinical limited-angle CT applications are the use of different system geometries. The reason for aligning the slices of the phantoms with the xy-plane is that it is typically the plane of interest in common industrial applications. For example, the reconstructions of the GPU dataset in Section 5.4.7 demonstrate that the studied BGA is almost parallel with the xy-plane. Consequently, both artificially generated phantoms have been aligned similar to the real data for the sake of comparison. Moreover, a look at reconstructed images in the area of industrial laminography and NDT [175–177] reveals that the here visualized results are reasonable and comparable to those of the industrial studies.

It had long since come to my attention that people of accomplishment rarely sat back and let things happen to them. They went out and happened to things.

Leonardo da Vinci

6 Summary and Outlook

The last chapter of this thesis will summarize the work and the research contributions which have been discussed in the previous chapters. After drawing final conclusions, areas of future research and potential implementation aspects will be pointed out.

6.1 Conclusion

During this work, the presentations and discussions in Chapters 3 to 5 have demonstrated that the overall image quality of reconstruction from cone-beam computed tomography (CT) data depends on many factors. Since the implementation of novel algorithms in the area of tomography can become quite complicated due to the complex mathematical background of the subject, a set of standardized tools facilitates an easy introduction to this research field and supports the derivation of novel algorithms. Such a toolbox is given by the software framework suggested in Chapter 2 and can be used to assist future students and researchers in their development process. The software requirements for a computed tomography library have been discussed and a framework for applications in cone-beam computed tomography (CBCT) has been derived and explained. The presented framework is self-contained and has been implemented using a combination of the MATLAB scripting language and native low-level code (C/C++ and Compute Unified Device Architecture (CUDA)). This design allows a specific application to exploit the maximal available computational performance of the underlying hardware while it provides a good accessibility and extendability through the use of the scripting language.

On top of this framework, the realization of Katsevich's algorithm on multi-core hardware has been described in detail and the resulting implementation has been evaluated and compared to the standard Feldkamp, Davis and Kress (FDK) algorithm. It has been shown on simulated data that the helical reconstruction method of Katsevich has the potential to improve the resentment results in the area of non-destructive testing (NDT). However, the artifacts due to a misaligned scanning geometry are more severe in the helical reconstructions than in the circular ones. In summary, the discussion in the end of Chapter 3 has suggested that even if Katsevich's algorithm shows best results on simulated data, the conducted study demonstrated that in the case of real data the influence and the uncertainty of geometric parameters make a precise calibration of the CT scanner necessary to increase the accuracy of helical acquisitions and reconstructions. For this reason, it is necessary to apply a precise, stable and robust calibration method to obtain high quality reconstructions.

As a consequence, in Chapter 4 a novel geometrical calibration method for CBCT has been introduced. The proposed self-calibration algorithm has been designed to estimate the misalignment parameters of a CT scanner from the cone-beam projection data without the need of any additional measurements or reference scans. It combines a multi-scale 1D grid search optimization scheme with the maximization of the mutual information (MI) with respect to the reprojected volume to estimate a set of four unknown detector parameters. The internal optimization scheme is similar to a 2D-3D registration approach, whereby a novel volume update scheme in combination with a stochastic reprojection strategy has been integrated into the algorithm to achieve a reasonable runtime performance. After a run of the calibration, the estimated parameters can be used to correct the misalignment of the CT scanner.

In addition to this, it has been shown that this approach works successfully for circular and helical trajectories, that it is stable and robust in the presence of projection noise and that its time performance is comparable to current registration-based calibration techniques. In the experiments it has been demonstrated that the calibration is able to handle various types of objects and that sub-voxel precision can be reached. Additionally, the suggested calibration framework can be extended, so that additional misalignment parameters and scanning paths can be included and the optimization can be run per projection. It has been shown that the method works well with a different optimization algorithm even though a slight increase of the runtime can be observed. In summary, the discussed results have shown that the calibration methods introduced in this thesis reaches an acceptable accuracy and can compete with current state-of-the-art calibration approaches.

For the construction of filters in the area of limited-angle tomography a general scheme which uses the Approximate Inverse (AI) to compute optimized angle-dependent projection filters has been derived in Chapter 5. The approach constitutes the first application using the theory of the AI in the context of 3D limited-angle CBCT. The method of AI is used to derive a single angle-dependent 3D reconstruction kernel optimized by simultaneous iterative reconstruction technique (SIRT). This kernel allows then the design of a projection-wise set of 2D filters, which can be employed in a filtered backprojection (FBP) framework. The final algorithm does not require the existence of a theoretically exact inversion formula, nor does it need a separate reference scan of a point- or line-like structure.

Quite the contrary, the iterative estimation of the optimized filters can be carried out solely from the knowledge about the geometric properties of the CT scanner. Moreover, the realization of the proposed approach is straightforward, allows an easy integration into existing FBP algorithms and can be extend to other geometric setups and scanning trajectories by following the same scheme that has been used for the limited-angle case. Additionally, the general approach of using the AI allows the design of application specific filters, which combine image reconstruction and analysis into a single step. As it has been shown for a circular system in [187] such more advanced filters can be developed using the here suggested design principle and applying it not only to limited-angle CBCT. For this purpose, the resulting overall filter optimization approach can roughly be outlined by the following three steps: (1) design a mollifier that performs the required image processing steps on the volume of interest, i.e. a gradient operation, (2) compute reconstruction kernels from the given mollifier by solving the auxiliary problem given in equation (5.4) by the application of iterative reconstruction techniques, e.g. SIRT and (3) exploit invariances of the kernels or model them approximately to gain a set of filters, which can be used in a FBP-style algorithm.

Additionally to the theory above, sets of filters have been precomputed during the experiments for two angular range setups and where then reused on multiple datasets. The novel angle-optimized FBP (AO-FBP) approach has been compared to the standard FDK algorithm and to SIRT in Section 5.4. The evaluation of the three algorithms with respect to image quality has been conducted on simulated and on real projection data. Hereby, the overall performance of AO-FBP showed to be comparable to that of SIRT. Moreover, it the technique outperformed the standard FDK method with respect to image quality measured by the root-mean-square error (RMSE) and image contrast quantified by the contrast-to-noise ratio (CNR). Similar to SIRT, the AO-FBP significantly reduced the artifact spread for planes lying near to the in-focus features, while the characteristics of the AO-FBP and the FDK algorithm in relation to image resolution have been noticed to be in the same order of magnitude. Moreover, for each system configuration the experiments have clearly demonstrated that the numerical optimization approach can be employed to precompute a set of 2D filters that can be reused for different datasets, as it has been the case for the simulated datasets. Finally, it has been observed that the current implementation of AO-FBP has even shorter runtimes than the FDK algorithm. In summary, it can be stated that the here introduced filter optimization produces results comparable to those of SIRT with respect the reduction of reconstruction artifacts, whereby its runtime performance is similar to that of the FDK algorithm.

6.2 Future Work

In the context of this work only prototypes of the introduced framework and developed algorithms have been built. It has been shown on simulated and real data that these methods can be used efficiently to support and to improve the results gained from CBCT reconstructions.

Future plans consider extending the calibration approach by a projection-wise optimization, make further evaluations with medical datasets and pay special attention to the area of measurement un-

certainty in industrial CT. These studies will also allow determining the limiting factors of the calibration method even more precisely. Moreover, the implementation of the online calibration will be optimized for speed using the techniques mentioned in the previous chapters.

With respect to the filter optimization algorithm it has been planned an extension of the approach to different geometric setups, so that 2D sets of filters can be computed for arbitrary source trajectories. Furthermore, an integration of different optimization methods (e.g. conjugate gradient methods) and appropriate regularization techniques (e.g. total variation approaches) into the iteratively driven reconstruction kernel approximation process can lead to future improvements of the here proposed principal method. Additionally, further evaluations of the method will be conducted with medical and industrial datasets. In particular, the application of the filter calculation scheme to tomosynthesis and to other clinical applications can be a promising direction for future research. In addition to this, a full verification that the AO-FBP method can be applied for other geometric setups in the area of tomosynthesis and industrial CT with the same success needs to be performed in future studies.

In Section 5.2.1 it has been pointed out that several modifications of the standard FDK method have been proposed to improve the image quality of limited-angle CT reconstruction [178, 194–197, 199, 201]. However, the implementation of these approaches was beyond this dissertation and they are not implemented in the current state of the reconstruction framework. A future implementation of at least some of these methods would allow comparisons of these improved techniques with the AO-FBP algorithm and could lead to further enhancements of the filter optimization process. Moreover, a detailed numerical evaluation of the here introduced method with the ones presented by [172] and [183] is still outstanding due to the large implementation effort.

Besides all this, there have been plans to make parts of the software framework which has been used throughout the thesis as a basis for the 3D CBCT applications publicly available, such that it can be used and extended by other researchers in the field of CT.

Appendices

A

Transformation Matrices

For the definition of the geometric setup of a CBCT system transformation matrices are employed. For this purpose, the basic translation, scaling and rotation matrices in homogeneous coordinates are defined below. A translation along the x-axis by t_x, along the y-axis by t_y and along the z-axis by t_z can be expressed in matrix notation as

$$T(t_x, t_y, t_z) = \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

A scaling along the x-direction, y-direction and z-direction by the factors s_x, s_y and s_z, respectively, can be defined as

$$S(s_x, s_y, s_z) = \begin{pmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

In addition to this, a rotation around the x-axis, y-axis and z-axis by the given Euler angles α, β and γ can be decomposed as

$$
\begin{aligned}
R(\alpha,\beta,\gamma) &= R_x(\alpha)\cdot R_y(\alpha)\cdot R_x(\alpha) \\
&= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos\beta & 0 & \sin\beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\beta & 0 & \cos\beta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 & 0 \\ \sin\gamma & \cos\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.
\end{aligned}
$$

B

Message Passing from C/C++ to MATLAB

When using C/C++ code, compiling it to a MEX binary and using it from within the MATLAB environment the console output generated by functions like *printf* and the use of the standard console output *std::out* will not appear in the MATLAB command window. In general without the instantiation of a separate graphical user interface (GUI) context, it can be quite complicated to get some kind of visual feedback from a MEX function. For this reason, a helping class has been implemented in C++ to support the output of command line message issued from inside the MEX code to the MATLAB prompt. The code of the *MatlabStream* class which is a specialization of the stream buffer class *std::streambuf* is presented in Listing B.1. The constructor of the class replaces the standard console output stream buffer with the current instance of the *MatlabStream* class. Additionally, a reference to the original stream buffer is stored in a temporary variable so that it can be restored in the class destructor. Once an instance of the *MatlabStream* class has been created all C/C++ methods writing to *std::cout* will not use the original stream buffer, but the one provided by the implementation of the here presented class. The virtual methods *xsputn* and *overflow* will redirect all messages to the MATLAB command prompt by using the *mexPrintf* function and by issuing the *drawnow* command. The *drawnow* method causes MATLAB to flush the event queue and to update its user interface including the command window. Since the original stream buffer stays replaced by the *MatlabStream* object for the duration of its existence, it makes sense to create a *MatlabStream* instance at the beginning of each MEX function that needs to issue messages to the MATLAB command line. The technique described above can then be used to send any type message from C/C++ layer to MATLAB and has been used intensively throughout the tomography framework introduced in Chapter 2.

Listing B.1: Source code of the MatlabStream class

```
class MatlabStream : public std::streambuf {
  public:
    MatlabStream () : std::streambuf (), mStdStream (NULL) {
      mStdStream = std::cout.rdbuf(this);
    }

    ~MatlabStream () {
      if (NULL != mStdStream) std::cout.rdbuf(mStdStream);
    }

  protected:
    virtual std::streamsize xsputn(const char *s, std::streamsize n) {
      mexPrintf("%.*s",n,s);
      mexEvalString("drawnow;");
      return n;
    }

    virtual int overflow(int c = EOF) {
      if (c != EOF) {
        mexPrintf("%.1s",&c);
        mexEvalString("drawnow;");
      }
      return 1;
    }

  private:
    std::streambuf* mStdStream;
};
```

C

Effect of Rotation Parameters on the Detector

To compare the rotation r_w to the detector shifts (t_u, t_v and t_w), the maximal transformation that a rotation will apply to a detector pixel is calculated. Given a detector pixel (u, v), it is rotated by the angle α around the detector center at $(0, 0)$ and compute the difference to its original position by

$$\begin{pmatrix} \delta u \\ \delta v \end{pmatrix} = \begin{pmatrix} |u \cdot cos(\alpha) - v \cdot sin(\alpha) - u| \\ |u \cdot sin(\alpha) + v \cdot cos(\alpha) - v| \end{pmatrix}.$$

The deviations δu and δv become larger with increasing distance r of the pixel from the detector center. Therefore, pixels at the detector corner with

$$r_{max} = \sqrt{((N_u - 1)/2 \cdot \Delta u)^2 + ((N_v - 1)/2 \cdot \Delta v)^2}$$

have the largest deviations. When setting $(u, v) = (0, r_{max})$ the maximal difference $\delta(r_w)_{max} = max(\delta u_{max}, \delta v_{max})$ can be calculated using

$$\begin{pmatrix} \delta u_{max} \\ \delta v_{max} \end{pmatrix} = \begin{pmatrix} | - r_{max} \cdot sin(\alpha)| \\ |r_{max} \cdot (cos(\alpha) - 1)| \end{pmatrix}.$$

for all pixels with radius r_{max}. With the Optimal Unit (OU) of r_w given in Table 4.1 and the system setup from the first column of Table 4.2 the value $\alpha = 0.338°$ can be set and $r_{max} = 289.065\,\text{mm}$ can be computed. The maximal translation that is introduced by r_w results in $\delta_{max} = 1.705\,\text{mm}$. Comparing this value to the OUs of the other misalignment parameters in Table 4.1 shows that the influence of r_w on the reconstructed volume is between those of t_u and t_v.

For the other two rotation parameters a similar approach can be used, so that r_u results in

$$\begin{pmatrix} \delta v_{max} \\ \delta w_{max} \end{pmatrix} = \begin{pmatrix} |-r_{height} \cdot sin(\alpha)| \\ |r_{height} \cdot (cos(\alpha) - 1)| \end{pmatrix}.$$

and r_v in

$$\begin{pmatrix} \delta u_{max} \\ \delta w_{max} \end{pmatrix} = \begin{pmatrix} |-r_{width} \cdot sin(\alpha)| \\ |r_{width} \cdot (cos(\alpha) - 1)| \end{pmatrix}.$$

with $r_{width} = (N_u - 1)/2 \cdot \Delta u$ and $r_{height} = (N_v - 1)/2 \cdot \Delta v$. As a result, $\delta(r_u)_{max} = \delta(r_v)_{max} = 3.535$ mm is obtained.

D

Parameter Estimation Algorithm

The pseudocode listed below gives a detailed outline of the calibration algorithm as it has been described in Section 4.3.3. The calibration procedure expects the original set of projections g_0 and all parameters of the system setup summarized in a single structure C to compute the projection matrices during the calibration process as described in Section 4.3.1. Initially, the four parameters t_u, t_v, t_w and r_w are set to be active and all set to zero. The outer loop beginning in line 4 of the estimation algorithm implements the multi-resolution approach starting with a binning of 16 for the projections and the volume, so that the resolution on the first level is reduced by a factor of 0.0625. After downsampling the initial projection set each parameter is optimized individually during the inner loop from line 10 to 36. Note that the initial sample range on the first level of the multi-scale approach is set manually to a range r_{init} and a step size of 8 to allow the algorithm the estimation of larger geometric misalignments by scanning a wider range of the parameter space (line 12). During the optimization the 1D scan is performed using OUs by centering the scan range around the actual estimate of the currently optimized parameter (line 16). In the lines 17 to 29 the reprojection similarity is computed at each point in the scan range R_{scan} and the maximal similarity MI_{max} is recorded with its location R_{max}. Then the samples MI are low-pass filtered and fitted by the Gaussian functions symmetric Gaussian (SG) and asymmetric Bi-Gaussian (AG) in line 31 to find the new parameter estimate. Finally, line 32 checks if the current parameter optimization has converged.

Algorithm 1 Parameter Estimation Algorithm

Input: Initial projections g_0, system setup C, scan range of first level r_{init}
Output: Estimated parameters $P^\star$

$P_{active} \leftarrow \{t_u, t_v, t_w, r_w\}$ ▷ activate all parameters
$P^\star \leftarrow \{0, 0, 0, 0\}$ ▷ intialize parameters with zeros
$P_{ou} \leftarrow$ ComputeOptimalUnits(C) ▷ compute OUs of all parameters
for each $b \in \{16, 8, 4, 2, 1\}$ **do** ▷ **resolution level loop**
  if $P_{active} = \emptyset$ **then** ▷ check if any parameter left
    return $P^\star$ ▷ terminate algorithm
  end if

  $s \leftarrow 1/b$ ▷ compute scale factor
  $g_{scaled} \leftarrow$ Resize(g_0, s) ▷ downsample projections by factor s
  for each $p \in P_{active}$ **do** ▷ **parameter loop**
    if $b = 16$ **then** ▷ scan wider range on the first level
      $R_{scan} \leftarrow [-r_{init}, -r_{init} + 8, ..., r_{init} - 8, r_{init}]$
    else
      $R_{scan} \leftarrow [-2b, -2b + 1, ..., 2b - 1, 2b]$
    end if
    $R_{scan} \leftarrow P^\star[p] + P_{ou}[p] * R_{scan}$ ▷ offset scan range and convert by OUs

    $R_{max} = -1$
    $MI_{max} = -\infty$
    for each $R \in R_{scan}$ **do** ▷ **scan loop**
      $P_{curr} \leftarrow P^\star$
      $P_{curr}[p] \leftarrow R$ ▷ replace value of current parameter
      $f_i \leftarrow$ FBP(g_{scaled}, P_{curr}) ▷ reconstruct volume
      $g_i \leftarrow$ DRR(f_i, P_{curr}) ▷ simulate projections
      $MI[R] \leftarrow$ MI(g_0 , g_i) ▷ calculate and store sample of MI
      if $MI(R) > MI_{max}$ **then**
        $R_{max} = R$
        $MI_{max} = MI[R]$
      end if
    end for

    $MI[R] \leftarrow$ Smooth(MI) ▷ low-pass filter all samples
    $p_{new} \leftarrow$ CenterOfBestGaussianFit(MI, R_{max}) ▷ find best parameter value

    if $(P^\star[p] - p_{new}| < 0.25 P_{ou}[p]$ **then**
      $P_{active} \leftarrow P_{active} \setminus p$ ▷ deactivate parameter - precise enough
    end if
    $P^\star[p] \leftarrow p_{new}$ ▷ update estimated parameter
  end for
end for

return $P^\star$ ▷ return estimated parameters

E

Virtual Detector Rebinning

The algorithm below presents the ray casting approach for the virtual detector rebinning on a single projection. To compute a virtually aligned projection from misaligned detector data, the projection matrices M_{var} and M_{vmr} are needed for the aligned and misaligned geometric setup, like described in equation (4.1) and equation (4.9), respectively. Note that the notation in the pseudocode below is similar to the one used in [18].

Algorithm 2 Virtual Detector Rebinning

Input: Initial projections g_0, projection matrices M_{mvr}, M_{avr}, projection size $N_u \times N_v$
Output: Rebinned projection g_r

extract source position from aligned projection matrix
$P_{3,a} \leftarrow$ Extract first three columns from M_{var}
$p_{4,a} \leftarrow$ Extract last column of M_{var}
$S_a \leftarrow$ Compute source position by $-P_{3,a}^{-1} \cdot p_{4,a}$
extract source position from misaligned projection matrix
$P_{3,m} \leftarrow$ Extract first three columns from M_{vmr}
$p_{4,m} \leftarrow$ Extract last column of M_{vmr}
$S_m \leftarrow$ Compute source position by $-P_{3,m}^{-1} \cdot p_{4,m}$
compute three positions on the detector
$P1 \leftarrow S_m + P_{3,m}^{-1} \cdot (0, 0, 1)$
$P2 \leftarrow S_m + P_{3,m}^{-1} \cdot (N_u, 0, 1)$
$P3 \leftarrow S_m + P_{3,m}^{-1} \cdot (0, N_v, 1)$

compute misaligned detector plane

$P \leftarrow P1$ ▷ point on the plane

$N \leftarrow (P2 - P1) \times (P3 - P1)$ ▷ plane normal

for each pixel (u, v) of the aligned projection **do**

  $O \leftarrow S_a$ ▷ ray origin

  $D \leftarrow P_{3,a}^{-1} \cdot (u, v, 1)$ ▷ ray direction

  $I \leftarrow \textsc{IntersectRayPlane}(O, D, P, N)$ ▷ intersect ray with misaligned detector plane

  $(u_s, v_s) \leftarrow M_{vmr} \cdot I$ ▷ sample position

  $g_r(u, v) \leftarrow g_0(u_s, v_s)$ ▷ interpolate projection

end for

return G_r ▷ rebinned projection

F

Relation between Measured and Filtered Data

Let h be a set of appropriately filtered projections. Then, under the assumption that a FBP approach can be employed to reconstruct the attenuation coefficients of the volume of interest f, the following formula can be used to express this fact:

$$f = A^* h, \tag{F.1}$$

where A^* represents the backprojection operator. In addition to this, it is generally known that the originally measured projections g are obtained through

$$g = Af, \tag{F.2}$$

whereby A is defined as the forward projection operator, i.e. the system matrix (compare Section 5.3.2). Substituting equation (F.1) into equation (F.2) yields $g = AA^*h$ and demonstrates the relation between the measured and the filtered data. The defect

$$||g - AA^*h||_2 \tag{F.3}$$

can be minimized in order to find the "optimally" filtered data as $h = (AA^*)^+ g$. From this expression it becomes evident that the optimal filtering operator is the Moore-Penrose-pseudo-inverse ($^+$) of AA^*. Hereby, it has to be considered that in general the solution $(AA^*)^+$ by the minimization of the L2-norm is not independent of the underlying data g, such that the filter operator can be different for each dataset and for each system geometry. The method of Kunze et al. [172] uses projections of a wire phantom to demonstrate, which invariances can be assumed for an optimized set of filters.

Nevertheless, since Section 5.4.4 only evaluates the filter quality of AO-FBP and FDK with respect to a specific projection set and system geometry, invariances and generalizations for optimized sets of filters will not be discussed here in greater detail.

However, the derivation above can be employed to calculate the filtered projections h from the measured projection data g. To solve the minimization problem stated above, wide variety of methods can be employed. For the sake of simplicity and convenience, SIRT [247] is selected to minimize the L2-norm in equation (F.3). By the definition of the self-adjoint matrix $M := AA^*$, the update equation for SIRT can be formulated quite similar to equation (5.7) as

$$h_n = h_{n-1} + UMV(g - M^* h_{n-1}), \tag{F.4}$$

with $h_0 := 0$, n as the iteration index and $U = V$ as the diagonal matrices of inverse row and column sums with respect to the matrix M, respectively. Note that the update scheme above is equivalent to the corrected projections simultaneous iterative reconstruction technique (P-SIRT) algorithm described in [172]. Consequently, equation (F.4) leads to the same set of filtered projections as the P-SIRT approach and the resulting projections h_n will be called P-SIRT filtered projections.

Listing of Figures

Listing of Tables

Listing of Algorithms

Listing of Acronyms

1D-MRS	1D multi-resolution search
AG	asymmetric Bi-Gaussian
AI	Approximate Inverse
AO-FBP	angle-optimized FBP
ART	algebraic reconstruction technique
ASF	artifact-spread function
BGA	ball-grid-array
BPM	backprojection mismatch
BPF	backprojected filtration
CBCT	cone-beam computed tomography
CBC	cone beam cover
CBEA	Cell Broadband Engine Architecture
C-CBCT	circular CBCT
CI	cubic B-spline interpolation
CNR	contrast-to-noise ratio
CON	constant
CPU	central processing unit
CSV	comma-separated values
CT	computed tomography
CUDA	Compute Unified Device Architecture
DBT	digital breast tomosynthesis
DCL	digital computed laminography
DKD	Deutscher Kalibrierdienst
DLL	dynamic-link library
DQE	detective quantum efficiency
DRR	digital reconstructed radiograph
FBP	filtered backprojection
FDCT	flat-detector computed tomography
FDK	Feldkamp, Davis and Kress
FFT	fast Fourier transform
FIR	finite impulse response
FPGA	field-programmable gate array
FWHM	full width at half maximum
GFB	Gaussian frequency blending
GPU	graphics processing unit
GUI	graphical user interface
H-CBCT	helical CBCT
HPC	high performance computing
IC	integrated circuit
IGRT	image-guided radiation therapy
IMRT	intensity-modulated radiation therapy
IPP	Intel Performance Primitives
ITK	Insight Segmentation and Registration Toolkit
JSON	JavaScript Object Notation
KKW	calotte cube (Kugelkalottenwürfel)
kV	kilovoltage
L-BFGS	Limited-memory Broyden-Fletcher-Goldfarb-Shanno
LUT	lookup table
MITS	Matrix inversion tomosynthesis
MI	mutual information
ML-EM	maximum-likelihood expectation maximization
ML	maximum-likelihood
MPI	Message Passing Interface
MTF	modulation transfer function
MV	megavoltage

NCC	normalized cross-correlation
NDT	non-destructive testing
NPS	noise power spectrum
OFBP	optimized FBP
OU	Optimal Unit
OS-SIRT	ordered subsets SIRT
PC	personal computer
PCB	printed circuit board
PSF	point spread function
P-SIRT	corrected projections simultaneous iterative reconstruction technique
RMSE	root-mean-square error
ROI	region of interest
SAA	shift-and-add algorithm
SAD	sum of absolute differences
SART	simultaneous algebraic reconstruction technique
SG	symmetric Gaussian
SIG	signal-dependent
SIMD	single-instruction multiple-data
SIRT	simultaneous iterative reconstruction technique
SL	Shepp-Logan
SNR	signal-to-noise ratio
SOP	sum of projections
SQD	sum of squared differences
SRS	stochastic ray sampling
SR	synchrotron radiation
ST	slice thickness
TD	Tam-Danielsson
TBB	Thread Building Blocks
XML	Extensible Markup Language

References

[1] J. Muders, J. Hesser, A. Lachner, and C. Reinhart, "Accuracy Evaluation and Exploration of Measurement Uncertainty for Exact Helical Cone Beam Reconstruction Using Katsevich Filtered Backprojection in Comparison to Circular Feldkamp Reconstruction with Respect to Industrial CT Metrology," in *Proc. International Symposium on Digital Industrial Radiology and Computed Tomography*, Jun. 2011, pp. 1–8.

[2] J. Muders and J. Hesser, "Stable and Robust Geometric Self-Calibration for Cone-Beam CT Using Mutual Information," *IEEE Trans. Nucl. Sci.*, vol. 61, no. 1, pp. 202–217, Feb. 2014.

[3] ——, "Projection-wise filter optimization for limited-angle cone-beam CT using the Approximate Inverse," 2014, submitted to *IEEE Trans. Nucl. Sci.*, accepted and in print.

[4] A. Sasov, X. Liu, and P. Salmon, "Compensation of mechanical inaccuracies in micro-CT and nano-CT," in *Proc. SPIE X-Ray Tomography*, vol. 7078, Aug. 2008, pp. 70 781C–70 781C–9.

[5] S. J. Schambach, S. Bag, V. Steil, C. Isaza, L. Schilling, C. Groden, and M. A. Brockmann, "Ultrafast high-resolution in vivo volume-CTA of mice cerebral vessels." *Stroke; a journal of cerebral circulation*, vol. 40, no. 4, pp. 1444–50, Apr. 2009.

[6] H. Wertz, D. Stsepankou, M. Blessing, M. Rossi, C. Knox, K. Brown, U. Gros, J. Boda-Heggemann, C. Walter, J. Hesser, F. Lohr, and F. Wenz, "Fast kilovoltage/megavoltage (kVMV) breathhold cone-beam CT for image-guided radiotherapy of lung cancer." *Physics in medicine and biology*, vol. 55, no. 15, pp. 4203–17, Aug. 2010.

[7] F. Natterer and F. Wübbeling, *Mathematical Methods in Image Reconstruction*. Society for Industrial and Applied Mathematics, 2001.

[8] A. Faridani, "Introduction to the Mathematics of Computed Tomography," *Inside Out: Inverse Problems and Applications*, vol. 47, pp. 1–46, 2003.

[9] T. M. Buzug, *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT*, 1st ed. Springer, Jul. 2008.

[10] W. A. Kalender, *Computed tomography: fundamentals, system technology, image quality, applications*. Wiley, 2011.

[11] C. Shaw, *Cone Beam Computed Tomography*, ser. Imaging in medical diagnosis and therapy. Taylor & Francis, 2014.

[12] L. A. Feldkamp, L. C. Davis, and J. W. Kress, "Practical cone-beam algorithm," *J. Opt. Soc. Amer.*, vol. 1, no. 6, pp. 612–619, Jun. 1984.

[13] A. Katsevich, "Theoretically exact filtered backprojection-type inversion algorithm for spiral CT," *SIAM J. Appl. Math.*, vol. 62, no. 6, pp. 2012–2026, 2002.

[14] W. Mao, T. Li, N. Wink, and L. Xing, "CT image registration in sinogram space." *Med. Phys.*, vol. 34, no. 9, pp. 3596–3602, Sep. 2007.

[15] K. Martin and B. Hoffman, *Mastering CMake: A Cross-Platform Build System*. Kitware Inc, Jan. 2003.

[16] P. Luszczek, "Parallel Programming in MATLAB," *International Journal of High Performance Computing Applications*, vol. 23, no. 3, pp. 277–283, Aug. 2009.

[17] J. Kim, H. Guan, D. Gersten, and T. Zhang, "Evaluation of algebraic iterative image reconstruction methods for tetrahedron beam computed tomography systems," *Int. J. Biomed. Imaging*, vol. 2013, pp. 1–14, May 2013.

[18] R. R. Galigekere, K. Wiesent, and D. W. Holdsworth, "Cone-beam reprojection using projection-matrices," *IEEE Trans. Med. Imag.*, vol. 22, no. 10, pp. 1202–1214, Oct. 2003.

[19] A. Katsevich, "An improved exact filtered backprojection algorithm for spiral computed tomography," *Adv. in Appl. Math.*, vol. 32, no. 4, pp. 681–697, May 2004.

[20] A. Weinlich, B. Keck, H. Scherl, M. Kowarschik, and J. Hornegger, "Comparison of High-speed ray casting on GPU using CUDA and OpenGL," in *Proc. First International Workshop on New Frontiers in High-performance and Hardware-aware Computing*, Nov. 2008, pp. 25–30.

[21] Y. Long, J. A. Fessler, and J. M. Balter, "3D forward and back-projection for X-ray CT using separable footprints with trapezoid functions," in *Proc. First Intl. Mtg. on Image Formation in X-ray Computed Tomography*, 2010, pp. 216–219.

[22] M. Wu and J. A. Fessler, "GPU acceleration of 3D forward and backward projection using separable footprints for X-ray CT image reconstruction," in *Proc. Intl. Mtg. on Fully 3D Image Recon. in Rad. and Nuc. Med*, Jul. 2011, pp. 56–59.

[23] R. Oster, "Non-destructive testing methodologies on helicopter fiber composite components challenges today and in the future," in *Proc. 18th World Conference on Nondestructive Testing*, no. April, Apr. 2012, pp. 1–10.

[24] D. Stsepankou, K. Kommesser, J. Hesser, and R. Männer, "Real-time 3D cone beam reconstruction," in *IEEE Nuclear Science Symposium Conference Record*, vol. 6, Oct. 2004, pp. 3648–3652.

[25] H. Turbell, "Cone-beam reconstruction using filtered backprojection," PhD thesis, Linköping University, Sweden, Feb. 2001.

[26] J. F. Barrett and N. Keat, "Artifacts in CT: Recognition and Avoidance," *RadioGraphics*, vol. 24, no. 6, pp. 1679–1691, Nov. 2004.

[27] F. Noo, J. Pack, and D. Heuscher, "Exact helical reconstruction using native cone-beam geometries," *Phys. Med. Biol.*, vol. 48, no. 23, pp. 3787–3818, Nov. 2003.

[28] J. Yang, X. Guo, Q. Kong, T. Zhou, and M. Jiang, "Parallel Implementation of Katsevich's FBP Algorithm," *Int. J. Biomed. Imaging*, vol. 2006, no. 1, pp. 1–8, Feb. 2006.

[29] E. Fontaine and H.-H. S. Lee, "Optimizing Katsevich image reconstruction algorithm on multicore processors," in *Proc. International Conference on Parallel and Distributed Systems*, vol. 2, Dec. 2007, pp. 1–8.

[30] A. Benquassmi, E. Fontaine, and H.-H. S. Lee, "Parallelization of Katsevich CT Image Reconstruction Algorithm on Generic Multi-Core Processors and GPGPU," in *GPU Computing Gems Emerald Edition*. Morgan Kaufmann, 2011, ch. 41, pp. 659–678.

[31] T. Schön, I. Bauscher, T. Fuchs, U. Hassler, J. Hiller, and S. Kasperl, "Dimensionelles Messen mit Helix-Computertomographie," in *Proc. Anual Meeting of the DGZfP*, Feb. 2009, pp. 1–2.

[32] J. Hiller, S. Kasperl, T. Schön, S. Schröpfer, and D. Weiss, "Comparison of Probing Error in Dimensional Measurement by Means of 3D Computed Tomography with Circular and Helical Sampling," in *Proc. International Symposium on NDT in Aerospace*, Nov. 2010, pp. 1–7.

[33] H. Hui, "An improved cone-beam reconstruction algorithm for the circular orbit," *Scanning*, vol. 18, no. 8, pp. 572–581, Nov. 1996.

[34] S. Mori, M. Endo, S. Komatsu, S. Kandatsu, T. Yashiro, and M. Baba, "A combination-weighted Feldkamp-based reconstruction algorithm for cone-beam CT," *Phys. Med. Biol.*, vol. 51, no. 16, pp. 3953–3965, Aug. 2006.

[35] H. Zheng, Z. Chen, Y. Kang, and J. Liu, "A New Heuristic Weighting Function for FDK-based reconstruction of Cone Beam Tomography," in *Proc. Asian-Pacific Conference on Medical and Biological Engineering*, vol. 19, no. 1, Apr. 2008, pp. 206–209.

[36] W. A. Kalender and Y. Kyriakou, "Flat-detector computed tomography (FD-CT)," *Eur. Radiol.*, vol. 17, no. 11, pp. 2767–2779, Nov. 2007.

[37] S. Xiao, Y. Bresler, and D. C. J. Munson, "Fast feldkamp algorithm for cone-beam computer tomography," in *Proc. International Conference on Image Processing (ICIP)*, vol. 2, Sep. 2003, pp. II–819–22 vol.3.

[38] F. Xu and K. Mueller, "Real-time 3D computed tomographic reconstruction using commodity graphics hardware," *Phys. Med. Biol.*, vol. 52, no. 12, pp. 3405–3419, Jun. 2007.

[39] M. Knaup, S. Steckmann, O. Bockenbach, and M. Kachelrieß, "Tomographic image reconstruction using the cell broadband engine (CBE) general purpose hardware," in *Proc. SPIE Electronic Imaging*, 2007, pp. 64 980P–64 980P–10.

[40] H. Scherl, M. Koerner, H. Hofmann, W. Eckert, M. Kowarschik, and J. Hornegger, "Implementation of the FDK algorithm for cone-beam CT on the cell broadband engine architecture," in *Proc. SPIE Medical Imaging*, vol. 6510, Mar. 2007, pp. 651 058–651 058–10.

[41] M. Kachelrieß, M. Knaup, and O. Bockenbach, "Hyperfast parallel-beam and cone-beam backprojection using the cell general purpose hardware," *Med. Phys.*, vol. 34, no. 4, pp. 1474–1486, Apr. 2007.

[42] D. Riabkov, X. Xue, D. Tubbs, and A. Cheryauka, "Accelerated cone-beam backprojection using GPU-CPU hardware," in *Proc International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, Jul. 2007, pp. 4–7.

[43] H. Scherl, B. Keck, M. Kowarschik, and J. Hornegger, "Fast GPU-based CT reconstruction using the common unified device architecture (CUDA)," in *IEEE Nuclear Science Symposium Conference Record*, vol. 6, Nov. 2007, pp. 4464 – 4466.

[44] G. Yan, J. Tian, S. Zhu, Y. Dai, and C. Qin, "Fast cone-beam CT image reconstruction using GPU hardware," *J. Xray Sci. Technol.*, vol. 16, no. 4, pp. 225–234, Jul. 2008.

[45] M. Knaup, S. Steckmann, and M. Kachelriess, "GPU-based parallel-beam and cone-beam forward-and backprojection using CUDA," in *IEEE Nuclear Science Symposium Conference Record*, Oct. 2008, pp. 5153–5157.

[46] Y. Okitsu, F. Ino, and K. Hagihara, "Accelerating Cone Beam Reconstruction Using the CUDA-Enabled GPU," in *High Performance Computing (HiPC)*, ser. Lecture Notes in Computer Science, Dec. 2008, vol. 5374, pp. 108–119.

[47] ——, "High-performance cone beam reconstruction using CUDA compatible GPUs," *Parallel Comput.*, vol. 36, no. 2–3, pp. 129–141, Mar. 2010.

[48] P. B. Noël, A. M. Walczak, J. Xu, J. J. Corso, K. R. Hoffmann, and S. Schafer, "GPU-based cone beam computed tomography," *Comput. Methods Programs Biomed.*, vol. 98, no. 3, pp. 271–277, Jun. 2010.

[49] G. Pratx and L. Xing, "GPU computing in medical physics: A review," *Med. Phys.*, vol. 38, no. 5, pp. 2685–2697, 2011.

[50] D. Soimu, C. Badea, and N. Pallikarakis, "Image Quality in Cone-Beam 3D reconstructions using Feldkamp algorithm: Acquisition Problems," in *Proc. International Conference on Medical Informatics & Engineering (MEDINF)*, vol. 11, Oct. 2003, pp. 3–6.

[51] C. Maaß, F. Dennerlein, F. Noo, and M. Kachelrieß, "Comparing short scan CT reconstruction algorithms regarding cone-beam artifact performance," in *IEEE Nuclear Science Symposium Conference Record*, Oct. 2010, pp. 2188–2193.

[52] R. Schulze, U. Heil, D. Groß, D. D. Bruellmann, E. Dranischnikow, U. Schwanecke, and E. Schoemer, "Artefacts in CBCT: a review," *Dentomaxillofac. Radiol.*, vol. 40, no. 5, pp. 265–273, Jul. 2011.

[53] A. Amirkhanov, C. Heinzl, M. Reiter, and E. Gröller, "Visual optimality and stability analysis of 3DCT scan positions," *IEEE Trans. Vis. Comput. Graphics*, vol. 16, no. 6, pp. 1477–1486, Dec. 2010.

[54] J. P. Kruth, M. Bartscher, S. Carmignato, R. Schmitt, L. De Chiffre, and A. Weckenmann, "Computed tomography for dimensional metrology," *CIRP Ann. Manuf. Technol.*, vol. 60, no. 2, pp. 821–842, Jan. 2011.

[55] A. Katsevich, "Theoretically exact FBP-type inversion algorithm for spiral CT," in *Proc. Sixth International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, Nov. 2001, pp. 6–9.

[56] ——, "Improved exact FBP algorithm for spiral CT," *Adv. Appl. Math*, pp. 1–19, Dec. 2001.

[57] ——, "Analysis of an exact inversion algorithm for spiral cone-beam CT," *Phys. Med. Biol.*, vol. 47, no. 15, pp. 2583–2697, Jul. 2002.

[58] ——, "A general scheme for constructing inversion algorithms for cone beam CT," *Int. J. Math. Math. Sci.*, vol. 2003, no. 21, pp. 1305–1321, Sep. 2003.

[59] H. Tuy, "An Inversion Formula for Cone-Beam Reconstruction," *SIAM J. Appl. Math.*, vol. 43, no. 3, pp. 546–552, Jul. 1983.

[60] A. Katsevich, S. Basu, and J. Hsieh, "Exact filtered backprojection reconstruction for dynamic pitch helical cone beam computed tomography," *Phys. Med. Biol.*, vol. 49, no. 14, pp. 3089–3103, Jun. 2004.

[61] Y. Zou and X. Pan, "Exact image reconstruction on PI-lines from minimum data in helical cone-beam CT," *Phys. Med. Biol.*, vol. 49, no. 6, pp. 941–959, Mar. 2004.

[62] J. Yang, Q. Kong, T. Zhou, and M. Jiang, "Cone beam cover method: An approach to performing backprojection in Katsevich's exact algorithm for spiral cone beam CT," *J. Xray Sci. Technol.*, vol. 12, no. 4, pp. 199–214, Feb. 2004.

[63] S. H. Izen, "A fast algorithm to compute the π-line through points inside a helix cylinder," *Proc. Amer. Math. Soc.*, vol. 135, no. 1, pp. 269–276, Jan. 2007.

[64] A. Katsevich, "3PI algorithms for helical computer tomography," *Adv. in Appl. Math.*, vol. 36, no. 3, pp. 213–250, Mar. 2006.

[65] A. Katsevich, K. Taguchi, and A. Zamyatin, "Formulation of four Katsevich algorithms in native geometry," *IEEE Trans. Med. Imag.*, vol. 25, no. 7, pp. 855–868, Jul. 2006.

[66] H. Yu, Y. Ye, and G. Wang, "Katsevich-type algorithms for variable radius spiral cone-beam CT," in *Proc. SPIE X-Ray Tomography*, vol. 5535, Oct. 2004, pp. 550–557.

[67] Y. Ye and G. Wang, "Filtered backprojection formula for exact image reconstruction from cone-beam data along a general scanning curve," *Med. Phys.*, vol. 32, no. 1, pp. 42–48, Jan. 2005.

[68] A. Katsevich and M. Kapralov, "Filtered Backprojection Inversion of the Cone Beam Transform for a General Class of Curves," *SIAM J. Appl. Math.*, vol. 68, no. 2, pp. 334–353, Nov. 2007.

[69] S. Zhao, H. Yu, and G. Wang, "A unified framework for exact cone-beam reconstruction formulas," *Med. Phys.*, vol. 32, no. 6, pp. 1712–1721, Jun. 2005.

[70] G.-H. Chen, T.-L. Zhuang, S. Leng, and B. E. Nett, "Cone-beam filtered backprojection image reconstruction using a factorized weighting function," in *Proc. SPIE Optical Engineering*, vol. 46, no. 8, Aug. 2007, pp. 087 006–087 006–14.

[71] C. E. Yarman, "An Inversion Method for the Cone-Beam Transform," in *Proc. SPIE Medical Imaging*, vol. 6142, Mar. 2006, pp. 61 424C–61 424C–8.

[72] A. Katsevich, A. A. Zamyatin, and M. D. Silver, "Optimized Reconstruction Algorithm for Helical CT With Fractional Pitch Between 1PI and 3PI," *IEEE Trans. Med. Imag.*, vol. 28, no. 7, pp. 982–990, Jul. 2009.

[73] L. Zeng and X. Zou, "Katsevich-type reconstruction for dual helical cone-beam CT," *J. Xray Sci. Technol.*, vol. 18, no. 4, pp. 353–367, Jan. 2010.

[74] J. Zhao, Y. Jin, Y. Lu, and G. Wang, "A Filtered Backprojection Algorithm for Triple-Source Helical Cone-Beam CT," *IEEE Trans. Med. Imag.*, vol. 28, no. 3, pp. 384–393, Mar. 2009.

[75] F. Noo, S. Hoppe, F. Dennerlein, G. Lauritsch, and J. Hornegger, "A new scheme for view-dependent data differentiation in fan-beam and cone-beam computed tomography," *Phys. Med. Biol.*, vol. 52, no. 17, pp. 5393–5414, Sep. 2007.

[76] A. Katsevich, "A note on computing the derivative at a constant direction," *Phys. Med. Biol.*, vol. 56, no. 4, pp. N53–61, Feb. 2011.

[77] A. Faridani, R. Hass, and D. C. Solmon, "Numerical and theoretical explorations in helical and fan-beam tomography," in *J. Phys.: Conf. Ser.*, vol. 124, no. 1, Jul. 2008, p. 012024.

[78] L. F. Wang, "The Improvement of Katsevich Reconstruction Algorithm Based on Cone-Beam CT," *Appl. Mech. Mater.*, vol. 239–240, pp. 1148–1151, Jan. 2013.

[79] A. J. Wunderlich, "The Katsevich Inversion Formula for Cone-Beam Computed Tomography," Master of science thesis, Oregon State University, Sep. 2006.

[80] H. Yu and G. Wang, "Studies on implementation of the Katsevich algorithm for spiral cone-beam CT," *J. Xray Sci. Technol.*, vol. 12, no. 2, pp. 97–116, Jul. 2004.

[81] T. Weber, "Spiralgeometrie in der 3D-Computertomographie – Die Inversionsformel von Katsevich," Diplomarbeit, Universität des Saarlandes, Postfach 151141, 66041 Saarbrücken, Apr. 2004.

[82] J. Deng, H. Yu, J. Ni, T. He, S. Zhao, L. Wang, and G. Wang, "A Parallel Implementation of the Katsevich Algorithm for 3-D CT Image Reconstruction," *J. Supercomput.*, vol. 38, no. 1, pp. 35–47, Oct. 2006.

[83] J. Ni, J. Deng, H. Yu, T. He, and G. Wang, "Analysis of Performance Evaluation of Parallel Katsevich Algorithm for 3-D CT Image Reconstruction," in *First International Multi-Symposiums on Computer and Computational Sciences (IMSCCS)*, vol. 1, Jun. 2006, pp. 258–265.

[84] J. Ni, J. Deng, T. He, H. Yu, and G. Wang, "Speedup Performance Analysis of Parallel Katsevich Algorithm for 3-D CT Image Reconstruction," *Int. J. of Computational Science and Engineering*, vol. 6, no. 3, pp. 151–159, Aug. 2011.

[85] K. Zeng, E. Bai, and G. Wang, "A fast CT reconstruction scheme for a general multi-core PC," *Int. J. Biomed. Imaging*, vol. 2007, p. 9, Apr. 2007.

[86] G. Xue, Y. Zhang, and J. Li, "An improved backprojection algorithm for spiral cone-beam CT with an improved ray traverse algorithm," in *Proc. 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human*, Oct. 2009, pp. 790–795.

[87] M. Kachelrieß, "High Performance Exact Spiral Cone–Beam CT Image Reconstruction," in *Proc. 2nd Workshop on High Performance Image Reconstruction and the 10th International Meeting on Fully 3D Image Reconstruction*, Sep. 2009, pp. 9–12.

[88] S. Steckmann, M. Knaup, and M. Kachelrieß, "Algorithm for hyperfast cone-beam spiral backprojection," *Comput. Methods Programs Biomed.*, vol. 98, no. 3, pp. 253–260, Jun. 2010.

[89] W. Bi, Z. Chen, L. Zhang, and Y. Xing, "Accelerate helical cone-beam CT with graphics hardware," in *Proc. SPIE Medical Imaging*, vol. 6913, Mar. 2008, pp. 69 132T–69 132T–8.

[90] G. Yan, J. Tian, S. Zhu, C. Qin, Y. Dai, F. Yang, P. Wu, and D. Dong, "Fast Katsevich algorithm based on GPU for helical cone-beam computed tomography," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 4, pp. 1053–1061, Jul. 2010.

[91] P. Biswal and S. Banerjee, "Implementation of Katsevich algorithm for helical cone-beam computed tomography using CORDIC," in *International Conference on Systems in Medicine and Biology (ICSMB)*, Dec. 2010, pp. 313–317.

[92] J. Zhu, S. Zhao, H. Yu, Y. Ye, S. W. Lee, and G. Wang, "Numerical studies on Feldkamp-type and Katsevich-type algorithms for cone-beam scanning along nonstandard spirals," in *Proc. SPIE X-Ray Tomography*, vol. 5535, Oct. 2004, pp. 558–565.

[93] H. Yu and G. Wang, "Studies on artifacts of the Katsevich algorithm for spiral cone-beam CT," in *Proc. SPIE X-Ray Tomography*, vol. 5535, Oct. 2004, pp. 540–549.

[94] R. A. Hass and A. Faridani, "Regions of Backprojection and Comet Tail Artifacts for π-Line Reconstruction Formulas in Tomography," *SIAM J. Imaging Sci.*, vol. 5, no. 4, pp. 1159–1184, Oct. 2012.

[95] K. Zeng, H. Yu, L. L. Fajardo, and G. Wang, "Cone-beam mammo-computed tomography from data along two tilting arcs," *Med. Phys.*, vol. 33, no. 10, pp. 3621–3633, Oct. 2006.

[96] K. Zeng, H. Yu, S. Zhao, L. L. Fajardo, C. Ruth, Z. Jing, , and G. Wang, "Digital tomosynthesis aided by low-resolution exact computed tomography," *J. Comput. Assist. Tomogr.*, vol. 31, no. 6, pp. 976–983, Dec. 2007.

[97] D. Yang, R. Ning, and W. Cai, "Circle plus partial helical scan scheme for a flat panel detector-based cone beam breast X-ray CT," *Int. J. Biomed. Imaging*, vol. 2009, p. 637867, Jan. 2009.

[98] X. Zou and L. Zeng, "Scan and reconstruction of helical cone-beam industrial CT for big workpiece," in *Proc. World Conference on Nondestructive Testing*, Oct. 2008, pp. 25–28.

[99] J. Reinders, *Intel Threading Building Blocks: Outfitting C++ for Multi-core Processor Parallelism*. O'Reilly Media, 2007.

[100] P.-E. Danielsson, P. Edholm, J. Eriksson, and M. Magnusson, "Towards exact reconstruction for helical cone-beam scanning of long objects. A new detector arrangement and a new completeness condition," in *Proc. Meeting on Fully 3D Image Reconstruction in Radiology and Nuclear Medicine*, 1997, pp. 141–144.

[101] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes 3rd Edition: The Art of Scientific Computing*, 3rd ed. New York, NY, USA: Cambridge University Press, 2007.

[102] K. C. Tam, S. Samarasekera, and F. Sauer, "Exact cone beam CT with a spiral scan," *Phys. Med. Biol.*, vol. 43, no. 4, p. 1015, Jul. 1998.

[103] J. Nickolls, I. Buck, M. Garland, and K. Skadron, "Scalable parallel programming with CUDA," *Queue*, vol. 6, no. 2, pp. 40–53, Mar. 2008.

[104] M. W. de Boer, A. Gröpl, J. Hesser, and R. Männer, "Reducing artifacts in volume rendering by higher order integration," in *Proc. IEEE Conference on Visualization; Late Breaking Hot Topics*, Oct. 1997, pp. 1–4.

[105] W. Kahan, "Pracniques: Further Remarks on Reducing Truncation Errors," *Commun. ACM*, vol. 8, no. 1, pp. 40–48, Jan. 1965.

[106] L. Shepp and B. Logan, "The Fourier reconstruction of a head section," *IEEE Trans. Nucl. Sci.*, vol. 21, no. 3, pp. 21–43, Jun. 1974.

[107] G. T. Smith, *Industrial Metrology: Surfaces and Roundness*. Springer London, 2002.

[108] H. Schöndube, K. Stierstorfer, and F. Noo, "Accurate helical cone-beam CT reconstruction with redundant data," *Phys. Med. Biol.*, vol. 54, no. 15, pp. 4625–4644, Aug. 2009.

[109] F. Noo, R. Clackdoyle, C. Mennessier, T. A. White, and T. J. Roney, "Analytic method based on identification of ellipse parameters for scanner calibration in cone-beam tomography," *Phys. Med. Biol.*, vol. 45, no. 11, pp. 3489–3508, Nov. 2000.

[110] Y. Sun, Y. Hou, F. Zhao, and J. Hu, "A calibration method for misaligned scanner geometry in cone-beam computed tomography," *NDT&E Int.*, vol. 39, no. 6, pp. 499–513, Sep. 2006.

[111] A. M. Kingston, A. Sakellariou, A. P. Sheppard, T. K. Varslot, and S. J. Latham, "An auto-focus method for generating sharp 3D tomographic images," in *Proc. SPIE X-Ray Tomography*, S. R. Stock, Ed., vol. 7804, Aug. 2010, pp. 78 040J–78 040J–15.

[112] A. Kingston, A. Sakellariou, T. Varslot, G. Myers, and A. Sheppard, "Reliable automatic alignment of tomographic projection data by passive auto-focus," *Med. Phys.*, vol. 38, no. 9, pp. 4934–45, Sep. 2011.

[113] T. Varslot, A. Kingston, G. Myers, and A. Sheppard, "High-resolution helical cone-beam micro-CT with theoretically-exact reconstruction from experimental data," *Med. Phys.*, vol. 38, no. 10, p. 5459, Oct. 2011.

[114] Y. Meng, H. Gong, and X. Yang, "On-line geometric calibration of cone-beam computed tomography for arbitrary imaging objects," *IEEE Trans. Med. Imag.*, vol. 32, no. 2, pp. 278–288, Feb. 2013.

[115] J. Zhao, Y. Lu, Y. Jin, E. Bai, and G. Wang, "Feldkamp-type reconstruction algorithms for spiral cone-beam CT with variable pitch," *J. X. Ray. Sci. Tech.*, vol. 15, no. 4, pp. 177–196, Jan. 2007.

[116] J. Li, R. J. Jaszczak, H. Wang, K. L. Greer, and R. E. Coleman, "Determination of both mechanical and electronic shifts in cone-beam SPECT," *Phys. Med. Biol.*, vol. 38, no. 6, pp. 743–754, Jun. 1993.

[117] L. von Smekal, M. Kachelrieß, E. Stepina, and W. A. Kalender, "Geometric misalignment and calibration in cone-beam tomography," *Med. Phys.*, vol. 31, no. 12, p. 3242, Nov. 2004.

[118] D. Stsepankou, A. Arns, S. K. Ng, P. Zygmanski, and J. Hesser, "Evaluation of robustness of maximum likelihood cone-beam CT reconstruction with total variation regularization," *Phys. Med. Biol.*, vol. 57, no. 19, pp. 5955–5970, Oct. 2012.

[119] F. Noo and S. Schaller, "Image reconstruction from misaligned truncated helical cone-beam data," in *IEEE Nuclear Science Symposium Conference Record*, 1999, pp. 1028–1032.

[120] S. Sawall, M. Knaup, and M. Kachelrieß, "A robust geometry estimation method for spiral, sequential and circular cone-beam micro-CT," *Med. Phys.*, vol. 39, no. 9, pp. 5384–92, Sep. 2012.

[121] H. Miao, X. Wu, H. Zhao, and H. Liu, "A phantom-based calibration method for digital x-ray tomosynthesis," *J. X. Ray. Sci. Tech.*, vol. 20, no. 1, pp. 17–29, Jan. 2012.

[122] G. T. Gullberg, B. M. W. Tsui, C. R. Crawford, J. G. Ballard, and J. T. Hagius, "Estimation of geometrical parameters and collimator evaluation for cone-beam tomography," *Med. Phys.*, vol. 17, no. 2, pp. 264–272, Mar. 1990.

[123] A. Rougée, C. Picard, C. Ponchut, and Y. Trousset, "Geometrical calibration of X-ray imaging chains for three-dimensional reconstruction," *Computerized Medical Imaging and Graphics*, vol. 17, no. 4-5, pp. 295–300, Jul. 1993.

[124] P. Rizo and P. Grangeat, "Geometric calibration method for multiple-head cone-beam SPECT system," *IEEE Trans. Nucl. Sci.*, vol. 41, no. 6, pp. 2748–2757, Dec. 1994.

[125] A. V. Bronnikov, "Virtual alignment of x-ray cone-beam tomography system using two calibration aperture measurements," *Opt. Eng.*, vol. 38, no. 2, pp. 381–386, Feb. 1999.

[126] Y. Cho, D. J. Moseley, J. H. Siewerdsen, and D. A. Jaffray, "Accurate technique for complete geometric calibration of cone-beam computed tomography systems," *Med. Phys.*, vol. 32, no. 4, pp. 968–983, Mar. 2005.

[127] J. Chetley Ford, D. Zheng, and J. F. Williamson, "Estimation of CT cone-beam geometry using a novel method insensitive to phantom fabrication inaccuracy: Implications for isocenter localization accuracy," *Med. Phys.*, vol. 38, no. 6, pp. 2829–2840, Jun. 2011.

[128] X. Li, D. Zhang, and B. Liu, "A generic geometric calibration method for tomographic imaging systems with flat-panel detectors – A detailed implementation guide," *Med. Phys.*, vol. 37, no. 7, pp. 3844–54, Jul. 2010.

[129] K. R. Muralidhar, P. N. Murthy, and R. Kumar, "Commissioning and quality assurance of the X-ray volume Imaging system of an image-guided radiotherapy capable linear accelerator," *J. Med. Phys.*, vol. 33, no. 2, pp. 72–77, Apr. 2008.

[130] K. Yang, A. L. C. Kwan, D. F. Miller, and J. M. Boone, "A geometric calibration method for cone beam CT systems," *Med. Phys.*, vol. 33, no. 6, pp. 1695–1706, Jun. 2006.

[131] M. Yang, J. Zhang, M. Yuan, X. Li, W. Liu, F. Meng, S.-J. Song, and D. Wei, "Calibration method of projection coordinate system for X-ray cone-beam laminography scanning system," *NDT & E International*, vol. 52, pp. 16–22, Nov. 2012.

[132] W. Wein, A. Ladikos, and A. Baumgartner, "Self-calibration of geometric and radiometric parameters for cone-beam computed tomography," in *Proc. 11th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, no. 2, Jul. 2011, pp. 1–4.

[133] V. Patel, R. N. Chityala, K. R. Hoffmann, C. N. Ionita, D. R. Bednarek, and S. Rudin, "Self-calibration of a cone-beam micro-CT system," *Med. Phys.*, vol. 36, no. 1, pp. 48–58, Jan. 2009.

[134] D. Panetta, N. Belcari, A. Del Guerra, and S. Moehrs, "An optimization-based method for geometrical calibration in cone-beam CT without dedicated phantoms," *Phys. Med. Biol.*, vol. 53, no. 14, pp. 3841–3861, Jul. 2008.

[135] S. Mayo, P. Miller, D. Gao, and J. Sheffield-Parker, "Software image alignment for X-ray microtomography with submicrometre resolution using a SEM-based X-ray microscope," *J. Microsc.*, vol. 228, no. 3, pp. 257–263, Dec. 2007.

[136] Y. Kyriakou, R. M. Lapp, L. Hillebrand, D. Ertel, and W. A. Kalender, "Simultaneous misalignment correction for approximate circular cone-beam computed tomography," *Phys. Med. Biol.*, vol. 53, no. 22, pp. 6267–6289, Nov. 2008.

[137] ——, "Image-based Online Correction of Misalignment Artifacts in Cone–Beam CT," in *Proc. SPIE Medical Imaging*, E. Samei and J. Hsieh, Eds., vol. 7258, Feb. 2009, pp. 72 581V–72 581V–10.

[138] J. Wicklein, H. Kunze, W. A. Kalender, and Y. Kyriakou, "Image features for misalignment correction in medical flat-detector CT," *Med. Phys.*, vol. 39, no. 8, pp. 4918–31, Aug. 2012.

[139] W. Wein, "Intensity Based Rigid 2D-3D Registration Algorithms for Radiation Therapy," Diplomarbeit, Technische Universität München, Dec. 2003.

[140] O. Fluck, C. Vetter, W. Wein, A. Kamen, B. Preim, and R. Westermann, "A survey of medical image registration on graphics hardware," *Comput. Meth. Programs Biomed.*, vol. 104, no. 3, pp. e45–e57, Dec. 2011.

[141] J. B. Maintz and M. A. Viergever, "A survey of medical image registration," *Med. Image Anal.*, vol. 2, no. 1, pp. 1–36, Mar. 1998.

[142] B. Zitová and J. Flusser, "Image registration methods: a survey," *Image Vision Comput.*, vol. 21, no. 11, pp. 977–1000, Jun. 2003.

[143] G. P. Penney, J. Weese, J. a. Little, P. Desmedt, D. L. Hill, and D. J. Hawkes, "A comparison of similarity measures for use in 2-D-3-D medical image registration," *IEEE Trans. Med. Imag.*, vol. 17, no. 4, pp. 586–95, Aug. 1998.

[144] J. P. W. Pluim, J. B. A. Maintz, and M. A. Viergever, "Mutual-information-based registration of medical images: a survey," *IEEE Trans. Med. Imag.*, vol. 22, no. 8, pp. 986–1004, Aug. 2003.

[145] D. Mattes, D. R. Haynor, H. Vesselle, T. K. Lewellen, and W. Eubank, "PET-CT image registration in the chest using free-form deformations," *IEEE Trans. Med. Imag.*, vol. 22, no. 1, pp. 120–128, Jan. 2003.

[146] U. Müller, J. Hesser, and R. Männer, "Fast rigid 2D-2D multimodal registration," in *MICCAI 04.* Springer, 2004, pp. 887–894.

[147] ——, "Optimal parameter choice for automatic fast rigid multimodal registration," in *Proc. SPIE Medical Imaging*, vol. 5747, Apr. 2005, pp. 163–169.

[148] I. M. J. van der Bom, S. Klein, M. Staring, R. Homan, L. W. Bartels, and J. P. W. Pluim, "Evaluation of optimization methods for intensity-based 2D-3D registration in x-ray guided interventions," in *Proc. SPIE Medical Imaging*, vol. 7962, Feb. 2011, pp. 796 223–796 223–15.

[149] P. Viola and W. M. Wells, "Alignment by maximization of mutual information," *Int. J. Comput. Vis.*, vol. 24, no. 2, pp. 137–154, Jun. 1997.

[150] W. M. Wells, P. Viola, and H. Atsumi, "Multi-modal volume registration by maximization of mutual information," *Med. Image Anal.*, Mar. 1996.

[151] A. Bardera and M. Feixas, "Medical image registration based on random line sampling," in *Proc. IEEE International Conference on Image Processing*, Sep. 2005, pp. 1157–1160.

[152] U. Müller, S. Bruck, J. Hesser, and R. Männer, "Correction of C-arm projection matrices by 3D-2D rigid registration of CT-images using mutual information," in *WBIR 03*. Springer, 2003, pp. 161–170.

[153] R. Shams and P. Sadeghi, "A survey of medical image registration on multicore and the GPU," *IEEE Signal Process. Mag.*, vol. 27, no. 2, pp. 50 – 60, Mar. 2010.

[154] L. Xu and J. W. L. Wan, "Real-time intensity-based rigid 2D-3D medical image registration using RapidMind Multi-core Development Platform," in *Proc. 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Aug. 2008, pp. 5382–5385.

[155] M. Karolczak, S. Schaller, K. Engelke, A. Lutz, U. Taubenreuther, K. Wiesent, and W. Kalender, "Implementation of a cone-beam reconstruction algorithm for the single-circle source orbit with embedded misalignment correction using homogeneous coordinates," *Med. Phys.*, vol. 28, no. 10, pp. 2050–2069, Oct. 2001.

[156] F. Xu and K. Mueller, "A comparative study of popular interpolation and integration methods for use in computed tomography," in *Proc. IEEE International Symposium on Biomedical Imaging: Nano to Macro*, 2006, pp. 1252–1255.

[157] P. Thévenaz and M. Unser, "Spline pyramids for intermodal image registration using mutual information," in *Proc. SPIE Optical Science*, vol. 3169, no. Oct., 1997, pp. 236–247.

[158] ——, "Optimization of mutual information for multiresolution image registration," *IEEE Trans. Image Process.*, vol. 9, no. 12, pp. 2083–99, Dec. 2000.

[159] A. Cole-Rhodes, K. Johnson, J. LeMoigne, and I. Zavorin, "Multiresolution registration of remote sensing imagery by optimization of mutual information using a stochastic gradient," *IEEE Trans. Image Process.*, vol. 12, no. 12, pp. 1495–1511, Dec. 2003.

[160] J. Le Moigne, W. J. Campbell, and R. F. Cromp, "An automated parallel image registration technique based on the correlation of wavelet features," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 8, pp. 1849–1864, 2002.

[161] D. Mattes, D. R. Haynor, H. Vesselle, T. K. Lewellyn, and W. Eubank, "Nonrigid multimodality image registration," in *Proc. SPIE Medical Imaging*, vol. 4322, Feb. 2001, pp. 1609–1620.

[162] L. Ibanez, W. Schroeder, L. Ng, and J. Cates, *The ITK Software Guide*. Clifton Park, NY: Kitware, 2005.

[163] D. Ruijters and P. Thevenaz, "GPU Prefilter for Accurate Cubic B-spline Interpolation," *Comput. J.*, vol. 55, no. 1, pp. 15–20, Jan. 2010.

[164] H. Lu, I.-T. Hsiao, X. Li, and Z. Liang, "Noise properties of low-dose CT projections and noise treatment by scale transformations," *IEEE Nuclear Science Symposium Conference Record*, vol. 3, pp. 1662–1666, 2002.

[165] P. Toft, "The Radon Transform - Theory and Implementation," Ph.D. dissertation, Department of Mathematical Modelling, Technical University of Denmark, Jul. 1996.

[166] S. Kasperl, "Qualitätsverbesserungen durch referenzfreie Artefaktreduzierung und Oberflächennormierung in der industriellen 3D-Computertomographie," Dissertation, Universität Erlangenâ€"Nürnberg, 2005.

[167] C. Heinzl, J. Kastner, B. Georgi, and H. Lettenbauer, "Comparison of surface detection methods to evaluate cone beam computed tomography data for three dimensional metrology," in *Proc. International Symposium on Digital industrial Radiology and Computed Tomography*, 2007.

[168] C. Heinzl, J. Kastner, A. Amirkhamov, E. Gröller, and C. Gusenbauer, "Optimal specimen placement in cone beam X-ray computed tomography," *NDT & E International*, vol. 50, pp. 42–49, Sep. 2012.

[169] J. Hess, P. Kuehnlein, S. Oeckl, and T. Schoen, "An Acquisition Geometry-Independent Calibration Tool for Industrial Computed Tomography," in *Proc. International Symposium on NDT in Aerospace*, 2012, pp. 1–7.

[170] J. Renström, "Evaluation of the Elekta Synergy concept for patient positioning in image guided radiotherapy," Master of Science Thesis, Lund University, 2005.

[171] F. Dennerlein and A. Jerebko, "Geometric jitter compensation in cone-beam CT through registration of directly and indirectly filtered projections," *IEEE Nuclear Science Symposium and Medical Imaging Conference Record*, no. 3, pp. 2892–2895, 2012.

[172] H. Kunze, W. Härer, J. Orman, T. Mertelmeier, and K. Stierstorfer, "Filter determination for tomosynthesis aided by iterative reconstruction techniques," in *Proc. 9th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, 2007, pp. 309–312.

[173] J. Ludwig, T. Mertelmeier, H. Kunze, and W. Härer, "A Novel Approach for Filtered Backprojection in Tomosynthesis Based on Filter Kernels Determined by Iterative Reconstruction Techniques," in *Digital Mammography*, ser. Lecture Notes in Computer Science, E. A. Krupinski, Ed. Springer Berlin Heidelberg, 2008, vol. 5116, pp. 612–620.

[174] J. T. Dobbins III and D. J. Godfrey, "Digital x-ray tomosynthesis: current state of the art and clinical potential," *Phys. Med. Biol.*, vol. 48, no. 19, pp. R65–R106, Oct. 2003.

[175] S. Gondrom and S. Schröpfer, "Digital computed laminography and tomosynthesis - functional principles and industrial applications," in *Proc. International Symposium on Computerized Tomography for Industrial Applications and Image Processing in Radiology*, vol. 7, no. 2, 1999, pp. 75–81.

[176] T. D. Moore, D. Vanderstraeten, and P. M. Forssell, "Three-dimensional x-ray laminography as a tool for detection and characterization of BGA package defects," *IEEE Trans. Compon., Hybrids, Manuf. Technol.*, vol. 25, no. 2, pp. 224–229, Jun. 2002.

[177] M. K. Cho, H. Youn, S. Y. Jang, and H. K. Kim, "Cone-beam digital tomosynthesis for thin slab objects," *NDT & E Int.*, vol. 47, pp. 171–176, Apr. 2012.

[178] Y.-H. Hu, B. Zhao, and W. Zhao, "Image artifacts in digital breast tomosynthesis: Investigation of the effects of system geometry and reconstruction parameters using a linear system approach," *Med. Phys.*, vol. 35, no. 12, pp. 5242–5252, Dec. 2008.

[179] Y. Zhang, H.-P. Chan, M. M. Goodsitt, A. Schmitz, J. W. Eberhard, and B. E. H. Claus, "Investigation of Different PV Distributions in Digital Breast Tomosynthesis (DBT) Mammography," in *Digital Mammography*, ser. Lecture Notes in Computer Science, E. Krupinski, Ed. Springer Berlin Heidelberg, 2008, vol. 5116, pp. 593–600.

[180] I. Sechopoulos, "A review of breast tomosynthesis. Part I. The image acquisition process," *Med. Phys.*, vol. 40, no. 1, pp. 014 301–1–12, Jan. 2013.

[181] V. Kolehmainen, S. Siltanen, S. Järvenpää, J. P. Kaipio, P. Koistinen, M. Lassas, J. Pirttilä, and E. Somersalo, "Statistical inversion for medical x-ray tomography with few radiographs: II. Application to dental radiology," *Phys. Med. Biol.*, vol. 48, no. 10, pp. 1465–1490, May 2003.

[182] X. Pan, E. Y. Sidky, and M. Vannier, "Why do commercial CT scanners still employ traditional, filtered back-projection for image reconstruction?" *Inverse Probl.*, vol. 25, no. 12, p. 123009, Dec. 2009.

[183] T. Nielsen, S. Hitziger, M. Grass, and A. Iske, "Filter calculation for x-ray tomosynthesis reconstruction," *Phys. Med. Biol.*, vol. 57, no. 12, pp. 3915–3930, Jun. 2012.

[184] A. K. Louis, "Approximate inverse for linear and some nonlinear problems," *Inverse Probl.*, vol. 12, no. 2, pp. 175–190, Apr. 1996.

[185] ——, "Filter design in three-dimensional cone beam tomography: circular scanning geometry," *Inverse Probl.*, vol. 19, no. 6, pp. S31–S40, Nov. 2003.

[186] T. Mertelmeier, J. Orman, W. Härer, and M. K. Dudam, "Optimizing filtered backprojection reconstruction for a breast tomosynthesis prototype device," in *Proc. SPIE Medical Imaging*, vol. 6142, Feb. 2006, pp. 61 420F–61 420F–12.

[187] A. K. Louis and T. Weber, "Image Reconstruction and Image Analysis in Tomography: Fan Beam and 3D Cone Beam," in *Proc. Interdisciplinary Workshop on Mathematical Methods in Biomedical Imaging and Intensity-Modulated Radiation Therapy (IMRT)*, Oct. 2007, pp. 211–229.

[188] M. E. Davison, "The Ill-Conditioned Nature of the Limited Angle Tomography Problem," *SIAM Journal on Applied Mathematics*, vol. 43, no. 2, pp. 428–448, Apr. 1983.

[189] I. Sechopoulos, "A review of breast tomosynthesis. Part II. Image reconstruction, processing and analysis, and advanced applications," *Med. Phys.*, vol. 40, no. 1, pp. 014 302–1–17, Jan. 2013.

[190] M. Males, D. Mileta, and M. Grgic, "Digital breast tomosynthesis: A technological review," in *Proc. 53rd International Symposium ELMAR*, Sep. 2011, pp. 41–45.

[191] D. G. Grant, "TOMOSYNTHESIS: A Three-Dimensional Radiographic Imaging Technique," *IEEE Trans. Biomed. Eng.*, vol. BME-19, no. 1, pp. 20–28, Jan. 1972.

[192] L. T. Niklason, B. T. Christian, L. E. Niklason, D. B. Kopans, D. E. Castleberry, B. H. Opsahl-Ong, C. E. Landberg, P. J. Slanetz, A. A. Giardino, R. Moore, D. Albagli, M. C. DeJule, P. F. Fitzgerald, D. F. Fobare, B. W. Giambattista, R. F. Kwasnick, J. Liu, S. J. Lubowski, G. E. Possin, J. F. Richotte, C. Y. Wei, and R. F. Wirth, "Digital tomosynthesis in breast imaging," *Radiology*, vol. 205, no. 2, pp. 399–406, 1997.

[193] P. Edholm, G. Granlund, H. Knutsson, and C. Petersson, "Ectomography. A new radiographic method for reproducing a selected slice of varying thickness," *Acta Radiol. Diagn. (Stockh.)*, vol. 21, no. 4, pp. 433–442, 1980.

[194] H. Matsuo, A. Iwata, I. Horiba, and N. Suzumura, "Three-dimensional image reconstruction by digital tomo-synthesis using inverse filtering," *IEEE Trans. Med. Imag.*, vol. 12, no. 2, pp. 307–313, Jun. 1993.

[195] G. Lauritsch and W. H. Härer, "Theoretical framework for filtered back projection in tomosynthesis," in *Proc. SPIE Medical Imaging*, vol. 3338, Feb. 1998, pp. 1127–1137.

[196] G. M. Stevens, R. Fahrig, and N. J. Pelc, "Filtered backprojection for modifying the impulse response of circular tomosynthesis," *Med. Phys.*, vol. 28, no. 3, pp. 372–380, Mar. 2001.

[197] B. Zhao and W. Zhao, "Three-dimensional linear system analysis for breast tomosynthesis," *Med. Phys.*, vol. 35, no. 12, pp. 5219–5232, Dec. 2008.

[198] J. Zhou, B. Zhao, and W. Zhao, "A computer simulation platform for the optimization of a breast tomosynthesis system," *Med. Phys.*, vol. 34, no. 3, pp. 1098–1109, Mar. 2007.

[199] B. Zhao, J. Zhou, Y.-H. Hu, T. Mertelmeier, J. Ludwig, and W. Zhao, "Experimental validation of a three-dimensional linear system model for breast tomosynthesis," *Med. Phys.*, vol. 36, no. 1, pp. 240–251, Jan. 2009.

[200] J. Orman, T. Mertelmeier, and W. Härer, "Adaptation of Image Quality Using Various Filter Setups in the Filtered Backprojection Approach for Digital Breast Tomosynthesis," in *Digital Mammography*, ser. Lecture Notes in Computer Science, S. M. Astley, M. Brady, C. Rose, and R. Zwiggelaar, Eds. Springer Berlin Heidelberg, 2006, vol. 4046, pp. 175–182.

[201] T. Mertelmeier, J. Ludwig, B. Zhao, and W. Zhao, "Optimization of Tomosynthesis Acquisition Parameters: Angular Range and Number of Projections," in *Digital Mammography*, ser. Lecture Notes in Computer Science, E. A. Krupinski, Ed. Springer Berlin Heidelberg, 2008, vol. 5116, pp. 220–227.

[202] B. Ren, C. Ruth, Y. Zhang, A. Smith, C. Williams, B. Polischuk, and Z. Jing, "The CNR method in scan angle optimization of tomosynthesis and its limitations," in *Proc. SPIE Medical Imaging*, vol. 7258, Feb. 2009, pp. 72 585W–72 585W–10.

[203] K. J. Batenburg and L. Plantagie, "Fast Approximation of Algebraic Reconstruction Methods for Tomography," *IEEE Trans. Image Process.*, vol. 21, no. 8, pp. 3648–3658, Aug. 2012.

[204] J. Shtok, M. Elad, and M. Zibulevsky, "Adaptive filtered-back-projection for computed tomography," in *Proc. IEEE 25th Convention of Electrical and Electronics Engineers in Israel*, Dec. 2008, pp. 528–532.

[205] ——, "Direct adaptive algorithms for CT reconstruction," in *Proc. IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, Jun. 2009, pp. 181–184.

[206] D. N. Ghosh Roy, R. A. Kruger, B. Yih, and P. Del Rio, "Selective plane removal in limited angle tomographic imaging," *Med. Phys.*, vol. 12, no. 1, pp. 65–70, Jan. 1985.

[207] J. T. Dobbins III, "Matrix Inversion Tomosynthesis improvements in longitudinal x-ray slice imaging," US Patent Patent 4,903,204, Feb. 20th, 1990.

[208] R. J. Warp, D. J. Godfrey, and J. T. Dobbins III, "Applications of matrix inversion tomosynthesis," in *Proc. SPIE Medical Imaging*, vol. 3977, Feb. 2000, pp. 376–383.

[209] Y. Chen, J. Y. Lo, J. A. Baker, and J. T. Dobbins III, "Gaussian frequency blending algorithm with matrix inversion tomosynthesis (MITS) and filtered back projection (FBP) for better digital breast tomosynthesis reconstruction," in *Proc. SPIE Medical Imaging*, vol. 6142, Mar. 2006, pp. 122–130.

[210] J. G. Colsher, "Iterative three-dimensional image reconstruction from tomographic projections," *Comput. Graph. Image Process.*, vol. 6, no. 6, pp. 513–537, Dec. 1977.

[211] P. Bleuet, R. Guillemaud, and I. Magnin, "Resolution improvement in linear tomosynthesis with an adapted 3D regularization scheme," in *Proc. SPIE Medical Imaging*, vol. 4682, Feb. 2002, pp. 117–125.

[212] T. Wu, "Three-dimensional mammography reconstruction using low-dose projection images," PhD thesis, Brandeis University, 2002.

[213] S. Suryanarayanan, a. Karellas, S. Vedantham, S. J. Glick, C. J. D'Orsi, S. P. Baker, and R. L. Webber, "Comparison of tomosynthesis methods used with digital mammography." *Acad. Radiol.*, vol. 7, no. 12, pp. 1085–1097, Dec. 2000.

[214] S. Suryanarayanan, A. Karellas, S. Vedantham, S. P. Baker, S. J. Glick, C. J. D'Orsi, and R. L. Webber, "Evaluation of Linear and Nonlinear Tomosynthetic Reconstruction Methods in Digital Mammography," *Acad. Radiol.*, vol. 8, no. 3, pp. 219–224, Mar. 2001.

[215] T. Wu, R. H. Moore, E. A. Rafferty, and D. B. Kopans, "A comparison of reconstruction algorithms for breast tomosynthesis," *Med. Phys.*, vol. 31, no. 9, pp. 2636–2647, Sep. 2004.

[216] C. Badea, Z. Kolitsi, and N. Pallikarakis, "Image quality in extended arc filtered digital tomosynthesis." *Acta Radiol.*, vol. 42, no. 2, pp. 244–248, Mar. 2001.

[217] Y. Chen, J. Y. Lo, and J. T. Dobbins III, "Impulse response analysis for several digital tomosynthesis mammography reconstruction algorithms," in *Proc. SPIE Medical Imaging*, vol. 5745, Feb. 2005, pp. 541–549.

[218] B. G. Z. d. Plantes, "Eine Neue Methode Zur Differenzierung in der Röntgenographie (Planigraphies)," *Acta Radiol.*, vol. 13, no. 2, pp. 182–192, 1932.

[219] Y. J. Roh, W. S. Park, and H. Cho, "Correcting image distortion in the X-ray digital tomosynthesis system for PCB solder joint inspection," *Image Vision Comput.*, vol. 21, no. 12, pp. 1063–1075, Nov. 2003.

[220] S. M. Rooks, B. Benhabib, and K. C. Smith, "Development of an inspection process for ball-grid-array technology using scanned-beam X-ray laminography," *IEEE Trans. Compon., Packag., Manuf. Technol. A*, vol. 18, no. 4, pp. 851–861, Dec. 1995.

[221] S. Gondrom, J. Zhou, M. Maisl, H. Reiter, M. Kröning, and W. Arnold, "X-ray computed laminography: an approach of computed tomography for applications with limited access," *Nucl. Eng. Des.*, vol. 190, no. 1–2, pp. 141–147, Jun. 1999.

[222] U. Ewert, J. Robbel, C. Bellon, A. Schumpm, and C. Nockermann, "Digital laminography," *Materialprüfung*, vol. 37, no. 6, pp. 218–222, 1995.

[223] J. Zhou, M. Maisl, H. Reiter, and W. Arnold, "Computed laminography for materials testing," *Appl. Phys. Lett.*, vol. 68, no. 24, pp. 3500–3502, Jun. 1996.

[224] J. An, "A new approach to translational laminographic method for PCB inspection," *Circuit World*, vol. 24, no. 2, pp. 14–20, 1998.

[225] A. R. Kalukin and V. Sankaran, "Three-dimensional visualization of multilayered assemblies using X-ray laminography," *IEEE Trans. Compon., Packag., Manuf. Technol. A*, vol. 20, no. 3, pp. 361–366, Sep. 2002.

[226] A. Sassov, "High Resolution Microlaminography," in *Proc. 7th European Conference on Non-Destructive Testing*, vol. 3, no. 8, May 1998.

[227] A. Sassov and F. Luypaert, "X-ray digital microlaminography for BGA and flip-chip inspection," in *AIP Conf. Proc.*, vol. 507, no. 1, May 2000, pp. 239–244.

[228] L. Helfen, A. Myagotin, A. Rack, P. Pernot, P. Mikulík, M. Di Michiel, and T. Baumbach, "Synchrotron-radiation computed laminography for high-resolution three-dimensional imaging of flat devices," *Phys. Status Solidi (a)*, vol. 204, no. 8, pp. 2760–2765, Aug. 2007.

[229] M. N. Mavrogordato, P. Wright, L. Helfen, I. Sinclair, and S. M. Spearing, "Assessment of laminate damage micromechanisms using high resolution synchrotron radiation computed tomography & laminography," in *Proc. 14th European Conference on Composite Materials*, Jun. 2010.

[230] L. Helfen, T. Baumbach, P. Mikulik, D. Kiel, P. Pernot, P. Cloetens, and J. Baruchel, "High-resolution three-dimensional imaging of flat objects by synchrotron-radiation computed laminography," *Appl. Phys. Lett.*, vol. 86, no. 7, p. 071915, Feb. 2005.

[231] A. K. Louis and W. Törnig, "Picture reconstruction from projections in restricted range," *Math. Meth. Appl. Sci.*, vol. 2, no. 2, pp. 209–220, Jun. 1980.

[232] R. L. Dietz, "Die Approximative Inverse als Rekonstruktionsmethode in der Röntgen-Computertomographie," PhD thesis, Universität des Saarlands, Saarbrücken, Apr. 1999.

[233] M. E. Davison and F. A. Grünbaum, "Tomographic reconstruction with arbitrary directions," *Comm. Pure Appl. Math.*, vol. 34, no. 1, pp. 77–119, Jan. 1981.

[234] A. Rieder and T. Schuster, "The approximate inverse in action with an application to computerized tomography," *SIAM J. Numer. Anal.*, vol. 37, no. 6, pp. 1909–1929, Jan. 2000.

[235] J. Mohr, "Rekonstruktionsfilter in der 3D-Computertomographie," Diplomarbeit, Universität des Saarlands, Saarbrücken, Sep. 2002.

[236] A. K. Louis, T. Weber, and D. Theis, "Computing Reconstruction Kernels for Circular 3-D Cone Beam Tomography," *IEEE Trans. Med. Imag.*, vol. 27, no. 7, pp. 880–886, Jul. 2008.

[237] A. K. Louis, "Optimal reconstruction kernels in medical imaging," in *Optimization in Medicine*, ser. Springer Optimization and Its Applications, C. J. S. Alves, P. M. Pardalos, and L. N. Vicente, Eds. Springer New York, 2008, vol. 12, pp. 153–168.

[238] T. Weber, "Schnelle Rekonstruktionskernberechnung in der 3D-Computertomographie," PhD thesis, Universität des Saarlandes, Apr. 2008.

[239] H. Hu and J. Zhang, "Helical cone-beam CT reconstruction algorithm based on approximate inverse," *Comp. Eng. App.*, vol. 47, no. 21, pp. 199–201, 2011.

[240] J. Feng, J.-Z. Zhang, and B. Zhou, "Compact Support FDK Kernel Reconstruction Model Base on Approximate Inverse," *Math. Probl. Eng.*, vol. 2012, pp. 1–12, Nov. 2012.

[241] J. Feng and J.-Z. Zhang, "Improved kernel-based limited-view CT reconstruction VIA anisotropic diffusion," in *Proc. IEEE 18th International Conference on Image Processing*, Sep. 2011, pp. 1381–1384.

[242] X. Lu, Y. Sun, and Y. Yuan, "Optimization for limited angle tomography in medical image processing," *Pattern Recogn.*, vol. 44, no. 10–11, pp. 2427–2435, Oct. 2011.

[243] A. N. Tikhonov and V. I. A. Arsenin, *Solutions of ill-posed problems*, ser. Scripta series in mathematics. Winston, 1977.

[244] V. Titarenko, R. Bradley, C. Martin, P. J. Withers, and S. Titarenko, "Regularization methods for inverse problems in x-ray tomography," in *Proc. SPIE X-Ray Tomography*, vol. 7804, Aug. 2010, pp. 78 040Z–78 040Z–10.

[245] M. Hegland and R. Anderssen, "A mollification framework for improperly posed problems," *Numerische Mathematik*, vol. 78, no. 4, pp. 549–575, Feb. 1998.

[246] H.-L. Hu and J.-Z. Zhang, "Approximate inverse based implementation for Tuy's formula," in *Proc. 17th IEEE International Conference on Image Processing (ICIP)*, no. 2, Sep. 2010, pp. 621–624.

[247] J. Gregor and T. Benson, "Computational analysis and improvement of SIRT," *IEEE Trans. Med. Imag.*, vol. 27, no. 7, pp. 918–24, Jan. 2008.

[248] J. Tang, B. E. Nett, and G.-H. Chen, "Performance comparison between total variation (TV)-based compressed sensing and statistical iterative reconstruction algorithms," *Phys. Med. Biol.*, vol. 54, no. 19, pp. 5781–5804, Sep. 2009.

[249] Y. Pan, R. Whitaker, A. Cheryauka, and D. Ferguson, "TV-regularized iterative image reconstruction on a mobile C-ARM CT," in *Proc. SPIE Medical Imaging*, vol. 7622, Mar. 2010, pp. 76 222L–76 222L.

[250] B. Vandeghinste, B. Goossens, J. De Beenhouwer, A. Pizurica, W. Philips, S. Vandenberghe, and S. Staelens, "Split-Bregman-based sparse-view CT reconstruction," in *Proc. 11th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, 2011, pp. 431–434.

[251] X. Lu, Y. Yuan, P. Yan, and X. Li, "A novel alternative algorithm for limited angle tomography," in *Proc. IEEE 18th International Conference on Image Processing (ICIP)*, Sep. 2011, pp. 409–412.

[252] M. Kalke and S. Siltanen, "Adaptive frequency-domain regularization for sparse-data tomography," *Inverse Probl. Sci. En.*, vol. 21, no. 7, pp. 1099–1124, Aug. 2013.

[253] M. Koutalonis, H. Delis, G. Spyrou, L. Costaridou, G. Tzanakos, and G. Panayiotakis, "Contrast-to-noise ratio in magnification mammography: a Monte Carlo study," *Phys. Med. Biol.*, vol. 52, no. 11, p. 3185, Jun. 2007.

[254] A. H. Delaney and Y. Bresler, "Globally convergent edge-preserving regularized reconstruction: an application to limited-angle tomography," *IEEE Trans. Image Process.*, vol. 7, no. 2, pp. 204–221, Feb. 1998.

[255] E. Y. LaRoque, Samuel J.and Sidky and X. Pan, "Accurate image reconstruction from few-view and limited-angle data in diffraction tomography," *J. Opt. Soc. Amer.*, vol. 25, no. 7, pp. 1772–1782, Jul. 2008.

[256] Z. Chen, X. Jin, L. Li, and G. Wang, "A limited-angle CT reconstruction method based on anisotropic TV minimization," *Phys. Med. Biol.*, vol. 58, no. 7, pp. 2119–2141, Mar. 2013.

[257] Y. Yang, L. Li, and Z. Chen, "3D anisotropic total variation method for limitedangle CT reconstruction," in *Proc. IEEE Nuclear Science Symposium Conference Record*, Oct. 2013, pp. 1–4.

Colophon

This thesis was typeset using LaTeX, originally developed by Leslie Lamport and based on Donald Knuth's TeX. The body text is set in 12 point EB Garamond, designed by Georg Duffner, has been released under the SIL Open Font License (OFL), and can be found online at github.com/georgd or directly downloaded from bitbucket.org/georgd.

A template, which can be used to format a PhD thesis with a look and feel similar to this one, has been released under the permissive MIT (X11) license, and can be found online at github.com/suchow/ or from the author at suchow@post.harvard.edu.