Algebraic filters for filtered backprojection

Plantagie, L.

2

FAST APPROXIMATION OF ALGEBRAIC RECONSTRUCTION METHODS FOR TOMOGRAPHY

Abstract – Most reconstruction algorithms for transmission tomography can be subdivided in two classes: variants of Filtered Backprojection (FBP) and iterative algebraic methods. FBP is very fast and yields accurate results when a large number of projections are available, with high signal-to-noise ratio and a full angular range. Algebraic methods require much more computation time, yet they are more flexible in dealing with limited data problems and noise. In this chapter we propose an algorithm that combines the best of these two approaches: for a given linear algebraic method, a filter is computed that can be used within the FBP algorithm. The FBP reconstructions that result from using this filter strongly resemble the algebraic reconstructions and have many of their favorable properties, while the required reconstruction time is similar to standard-FBP. Based on a series of experiments, for both simulation data and experimental data, we demonstrate the merits of the proposed algorithm.

This chapter has been published with minor modification as: K. J. Batenburg and L. Plantagie. Fast Approximation of Algebraic Reconstruction Methods for Tomography. *IEEE Trans. Image Process.* 2012; 21(8): 3648–3658. This publication is available through http://dx.doi.org/10.1109/TIP.2012.2197012.

2.1 INTRODUCTION

Transmission tomography is a mature imaging technique, for which both the engineering aspects of image acquisition and the mathematical and computational aspects of image reconstruction are well understood [1–4]. Although tomography is nowadays often used as an off-the-shelf technique, it is important to realize that the resulting image depends quite strongly on the employed reconstruction algorithm. When comparing reconstruction methods, various aspects must be balanced, such as the quality of the reconstructed image, the required reconstruction time, and robustness to noise.

During the past decades, reconstruction algorithms for transmission tomography have been developed among different chains [5]. The derivation of *analytical algorithms* departs from an idealized continuous representation of the image reconstruction problem, for which analytical solutions can be obtained. The resulting inversion formulas are then discretized and transformed into a reconstruction algorithm. On the other hand, *algebraic algorithms* depart from a discretized model of the tomographic imaging setup, which is represented by a system of linear equations. Although a least-squares solution of this system could in principle be calculated directly, this is impractical due to the enormous size of the corresponding matrix. Instead, iterative methods are used, as they can deal effectively with such large sparse systems.

The *Filtered Backprojection* algorithm (FBP) is among the most popular analytical reconstruction methods. It is capable of computing accurate reconstructions with high computational efficiency, provided that high quality projections have been acquired for a sufficiently large number of angles, distributed evenly between 0° and 180°. For this reason, variants of FBP have dominated clinical CT practice for many years. In particular, the related Feldkamp, Davis, and Kress (FDK) algorithm for cone-beam reconstruction [6] has demonstrated the ability to combine fully 3-D image acquisition with accurate and efficient reconstruction.

The limitations of FBP become apparent when only a small number of projection images can be acquired, when the angular range of the projections is limited or irregular, or when the measured data is noisy. In such cases, the reconstruction quality of FBP degrades and artefacts hamper subsequent image processing tasks, such as segmentation.

Compared to FBP, iterative algebraic methods have several key advantages: 1) limited data problems (i.e., small number of projection angles, limited angular range) can be modeled accurately by adjusting the system of equations, whereas in FBP the underlying analytical model is based on the assumption of having projections available from all angles; 2) noise in the projections can be effectively averaged, by seeking a least-squares solution of the equation system; 3) certain types of prior knowledge, such as nonnegativity of the attenuation coefficients, can be incorporated in the reconstruction algorithm; and 4) physical properties of the imaging system, such as a spatially varying point-spread function can be modeled by adjusting the system of equations.

Although the analytical (continuous) formula that is the basis of the FBP algorithm is exact, interpolation errors are introduced if the assumptions made in this formula (an infinite number of projections must be available across a full angular range) are not satisfied. Therefore, the discretized version of the formula does not yield an exact solution of the reconstruction problem, not even if the projection data is noiseless. In contrast, iterative methods minimize the residual projection error by departing from a discretized model that only incorporates the data that is really available.

Despite these advantages, the high computational cost of iterative methods, which can easily be one or two orders of magnitude larger than the computational load of FBP, is a major obstacle toward widespread practical use of such algorithms. The reconstruction time for iterative algebraic methods can be strongly reduced by parallelizing the computations, in particular when combined with the massive parallelism of modern graphic processing units (GPUs) [7, 8]. Still, the reconstruction of large volumes by iterative methods takes a long time when compared with FBP, which can be accelerated similarly by parallel computation.

Connections between algebraic and analytical reconstruction methods have been explored by several authors. In [9], it was shown that a least-squares matrix formulation of the discretized FBP operator provides a connection between the analytic and algebraic reconstruction approach. Any linear, shift invariant reconstruction algorithm is equivalent to FBP with a particular filter, which was demonstrated in [10]. In [11], a formula was derived for a filter kernel that can be used to express the image quality of SIRT.

In this chapter, we propose an actual reconstruction *algorithm* that combines the favorable properties of iterative algebraic methods with the computational advantages of FBP: for a given linear algebraic method, an angle-dependent filter is computed that can be used within the FBP algorithm. The FBP reconstructions that result from using this filter strongly resemble the algebraic reconstructions, while the required reconstruction time is similar to that of standard-FBP. In particular, for one selected image pixel, the reconstruction result of the algebraic method is identical to the result of applying FBP with the proposed filter, independently of the projection data.

The filter computation has a high computational cost, which is much larger than the cost of reconstructing a single image by an algebraic method. However, as long as the imaging geometry (i.e., projection angles, detector size, etc.) is fixed, the same filter can be used for all subsequent reconstructions, as the filter does not depend in any way on the scanned object. In the vast majority of commercial CT-scanners, only a few different acquisition schemes are used for a particular scanner. The filters for these schemes can be computed once, using a separate computer system, or even a large cluster. Subsequently, reconstructions can be computed from an arbitrary number of tomography datasets at the same speed as standard-FBP. The computed filters can be directly incorporated in existing FBP implementations.

This chapter is structured as follows. In Section 2.2, we briefly revisit the analytical model behind the FBP algorithm and its discretization. Section 2.3 introduces a general model for linear algebraic reconstruction algorithms. The key contribution of this article is made in Section 2.4, where an expression is derived for *algebraic filters* (AFs): filters for the FBP algorithm that are extracted from a linear algebraic method. We also discuss how such filters can be computed. In Section 2.5, we describe a set of experiments that have been performed to compare the reconstruction quality of our approach with alternative reconstruction algorithms. The results of these experiments are presented in Section 2.6. Section 2.7 concludes this chapter.

2.2 THE FILTERED BACKPROJECTION ALGORITHM

In this chapter, we focus on a *parallel beam* scanning geometry, using a single rotation axis. For several other scanning geometries, including

the fan-beam geometry and the circular cone-beam geometry with small cone angle, the approach presented here can also be applied, after rebinning of the projection data.

We start by revisiting the Radon transform and its analytical inversion. Let $f : \mathbb{R}^2 \to \mathbb{R}$ be a finite and integrable function with bounded support for which the Radon transform

$$\begin{aligned} p(\theta, t) &= (\mathcal{R}f)(\theta, t) \\ &= \int_{-\infty}^{\infty} f(t\cos\theta - s\sin\theta, t\sin\theta + s\cos\theta)\, ds \end{aligned} \tag{2.1}$$

is defined almost everywhere in $\theta \in [0, \pi)$, $t \in \mathbb{R}$. We refer to [12] for details on the analytical properties of the Radon transform. The variables θ and t denote the angle with the vertical axis and the signed distance between the projected line and the origin of the coordinate system, respectively. We refer to $p(\theta, \cdot)$ as the *projection* of f for angle θ. For $\theta \in [0, \pi)$, define $P(\theta, u) = \int_{-\infty}^{\infty} p(\theta, t) e^{-2\pi i u t}\, dt$ and $q(\theta, t) = \int_{-\infty}^{\infty} P(\theta, u) G(\theta, u) e^{2\pi i u t}\, du$, where G represents a *filter*. Thus, $q(\theta, \cdot)$ is obtained from $p(\theta, \cdot)$ by applying the filter $G(\theta, \cdot)$ in the Fourier domain. Alternatively, the filtering of the projection data can be formulated as a convolution in real space as $q(\theta, t) = \int_{-\infty}^{\infty} p(\theta, \tau) g(\theta, t - \tau)\, d\tau$, where $g(\theta, \cdot)$ denotes the inverse Fourier Transform of $G(\theta, \cdot)$.

For the choice $G(\theta, u) = |u|$, known as the *Ramp filter*, the filtered projections can be used to obtain an exact inversion formula for the Radon transform

$$f(x, y) = \int_0^{\pi} q(\theta, x\cos\theta + y\sin\theta)\, d\theta. \tag{2.2}$$

In practice, the function $p(\theta, t)$ can only be measured for a finite set $\Theta = \{\theta_1, \ldots, \theta_d\}$ of projection angles, and at a finite set $T = \{t_1, \ldots, t_\ell\}$ of *detector positions*. Algorithms that require evaluation of $p(\theta, t)$ at other detector positions, such as the FBP algorithm, typically employ some form of interpolation to approximate these projection values. For the sake of clarity, we will assume that $T = \{-R, -R + 1, \ldots, R - 1, R\}$ with R a positive integer, which corresponds to an array of $2R + 1$ contiguous detectors with unit spacing, centered around $t = 0$.

The Filtered Backprojection algorithm is obtained by discretizing Eq. (2.2)

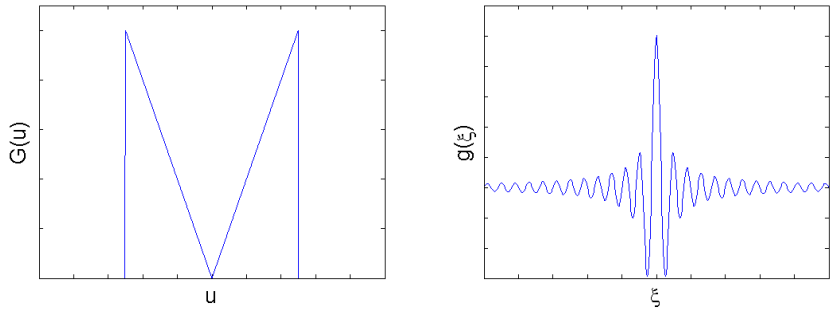
$$\begin{aligned} f(x,y) &\approx \frac{\pi}{d} \sum_{\theta\in\Theta} \int_{-\infty}^{\infty} p(\theta,\tau) g(\theta, x\cos\theta + y\sin\theta - \tau)\, d\tau \\ &\approx \frac{\pi}{d} \sum_{\theta\in\Theta} \sum_{\tau\in T} p(\theta,\tau) g(\theta, x\cos\theta + y\sin\theta - \tau) \\ &= \sum_{\theta\in\Theta} \sum_{\tau\in T} p(\theta,\tau) \tilde{g}(\theta, \tau - x\cos\theta - y\sin\theta), \end{aligned} \tag{2.3}$$

where $\tilde{g}(\theta,t) = (\pi/d)g(\theta,-t)$.

Various discrete approximations of the ideal Ramp filter $G(\theta,u) = |u|$ are used in practice. The Ram-Lak filter, for example, is a windowed Ramp-filter (see Fig. 2.1). Although the Ram-Lak filter can result in accurate reconstructions if high quality projection data are available, its amplification of high frequencies results in low reconstruction quality for noisy data. This effect can be reduced by applying a smooth window-function to the original filter. Several common filters, such as the Shepp-Logan and Cosine filter, are based on this principle. Still, their design is intrinsically heuristic, aimed at optimizing the visual quality of the reconstructed image.

Instead of designing filters in the Fourier domain as described above, the filters can also be designed in the spatial domain (i.e., the *detector* domain). In [13], new filters are created based on approximations of the Ramp filter in the spatial domain. In [14], the method of approximate inverse is applied to obtain filters in the spatial domain that are not derived from the Ramp filter. In [15] and [16], a method is described to create filters for tomosynthesis based on iterative reconstructions of certain test objects.

The effect of applying a particular filter to the projection data can be studied both in the Fourier domain and in the spatial domain, by considering the functions G and g, respectively. In this chapter, we mainly focus on the spatial domain, where the filter operation can be interpreted according to Eq. (2.3) as follows: the value of the reconstructed image at point (x,y) is formed by taking the dot-product of the discretized projection data with a *weight vector* that is formed by evaluating the function $\tilde{g}$ at consecutive discrete detector points. Therefore, the filter $\tilde{g}$ determines the weight of the contribution of each detector position to the reconstructed value.



(a) Fourier domain (b) Spatial domain

Figure 2.1: Filter formed by multiplying $G(\theta, u) = |u|$ with a window function.

2.3 ALGEBRAIC RECONSTRUCTION ALGORITHMS

In *algebraic* reconstruction algorithms, the reconstruction problem is represented by a system of linear equations. The reconstructed image is represented on a grid consisting of n pixels. Let $\mathbf{p} = (p_i) \in \mathbb{R}^m$ denote the measured data elements for all projections, collapsed into a single vector, where $m = d\ell$. Every entry p_i ($i = 1, \ldots, m$) corresponds to a pair $(\theta, t) \in \Theta \times T$, denoting the angle and detector position for that particular measurement. As an alternative notation, we refer to this entry as $p_{\theta t}$.

In the case of noiseless projection data, the projection process in tomography can be modeled as a finite linear transformation $\boldsymbol{W}$ that maps the image $\boldsymbol{\nu} = (\nu_i) \in \mathbb{R}^n$ (representing the object) to the vector $\mathbf{p}$ of measured data

$$\boldsymbol{W}\boldsymbol{\nu} = \mathbf{p}. \tag{2.4}$$

The $m \times n$ matrix $\boldsymbol{W} = (w_{ij})$ is called the *projection matrix* and the product $\boldsymbol{W}\boldsymbol{\nu}$ is referred to as the *forward projection* of $\boldsymbol{\nu}$. The entries of $\boldsymbol{\nu}$ correspond to the pixel values of the reconstruction. The entry w_{ij} determines the weight of the contribution of pixel j to measurement i, which usually represents the length of the intersection between the pixel and the projected line.

Several algebraic reconstruction algorithms, including the well-known ART, SART and SIRT algorithms [2], belong to the class of *linear reconstruction methods*. This means that their application to given projection data $\mathbf{p} \in \mathbb{R}^m$, yielding a reconstructed image $\mathbf{u} \in \mathbb{R}^n$, can be modeled as a linear transformation $S : \mathbb{R}^m \to \mathbb{R}^n$. We identify this transformation with the corresponding matrix $\mathbf{S} \in \mathbb{R}^{n \times m}$, called the *reconstruction matrix*, yielding the following expression describing the input-output relation of the reconstruction algorithm

$$\mathbf{u} = \mathbf{S}\mathbf{p}. \tag{2.5}$$

Example 1. For given projection data $\mathbf{p}$, one can apply a range of SIRT-like reconstruction algorithms to obtain a solution of Eq. (2.4). Taking $\mathbf{u}^{(0)} = \mathbf{0}$ as the start solution and denoting the reconstruction after k iterations by $\mathbf{u}^{(k)}$, the iteration step of this family of algorithms can be described by

$$\mathbf{u}^{(k+1)} = (\mathbf{I}_n - \omega \mathbf{C}\boldsymbol{W}^T\mathbf{R}\boldsymbol{W})\mathbf{u}^{(k)} + \omega \mathbf{C}\boldsymbol{W}^T\mathbf{R}\mathbf{p}, \tag{2.6}$$

where $\mathbf{C} = (c_{ij}) \in \mathbb{R}^{n \times n}$ is a diagonal matrix such that $c_{jj} = \alpha(\sum_{i=1}^{n} |w_{ij}|)$ for a certain scalar function α, $\mathbf{R} = (r_{ij}) \in \mathbb{R}^{m \times m}$ is a diagonal matrix such that $r_{ii} = \beta(\sum_{j=1}^{m} |w_{ij}|)$ for a certain scalar function β, $\mathbf{I}_n \in \mathbb{R}^{n \times n}$ denotes the identity matrix, and ω is a relaxation parameter [17].

Let $\mathbf{M} \in \mathbb{R}^{(m+n)\times(m+n)}$ be the *iteration-matrix* given by

$$\mathbf{M} = \begin{pmatrix} (\mathbf{I}_n - \omega \mathbf{C}\boldsymbol{W}^T\mathbf{R}\boldsymbol{W}) & \omega \mathbf{C}\boldsymbol{W}^T\mathbf{R} \\ \emptyset & \mathbf{I}_m \end{pmatrix}, \tag{2.7}$$

and define

$$\mathbf{S}_K = \begin{pmatrix} \mathbf{I}_n & \emptyset \end{pmatrix} \mathbf{M}^K \begin{pmatrix} \emptyset \\ \mathbf{I}_m \end{pmatrix}. \tag{2.8}$$

Then choosing $\mathbf{S} := \mathbf{S}_K$ gives the reconstruction matrix corresponding to K iterations of the SIRT algorithm.

2.4 ALGEBRAIC FILTERS

In this section, we will demonstrate how the reconstruction of a single pixel by a linear algebraic method can be interpreted as reconstruction of that same pixel by FBP, using a particular filter that is determined by the reconstruction matrix of the algebraic method. By using this filter within FBP to reconstruct the *entire* image, a reconstruction algorithm is obtained, which yields reconstructed images that are very similar to the results of the algebraic method.

Let $\mathbf{S}$ be a reconstruction matrix for a certain linear algebraic method S. We will now focus on a single pixel $c \in \{1, \ldots, n\}$ of the reconstructed image. Let $(x_c, y_c) \in \mathbb{R} \times \mathbb{R}$ denote the coordinates of the center of this pixel.

Denote the cth row of $\mathbf{S}$ by $\mathbf{s}^{(c)}$. Each entry of $\mathbf{s}^{(c)}$ corresponds to an entry in the right hand side of Eq. (2.5), and therefore to a pair $(\theta, \tau) \in \Theta \times T$, which we denote by $s^{(c)}_{\theta\tau}$. Substituting this notation in Eq. (2.5) yields

$$u_c = \sum_{\theta \in \Theta} \sum_{\tau \in T} p_{\theta\tau} s^{(c)}_{\theta\tau}. \tag{2.9}$$

For $\theta \in \Theta$, put $t^{(\theta)}_c = x_c \cos\theta + y_c \sin\theta$. We now introduce a function $h^{(c)}$, which is defined for $\tau \in T - t^{(\theta)}_c$, where the minus sign denotes element-wise subtraction

$$h^{(c)}(\theta, \tau) = s^{(c)}_{\theta(\tau + t^{(\theta)}_c)}. \tag{2.10}$$

Substituting Eq. (2.10) into Eq. (2.9) yields

$$u_c = \sum_{\theta \in \Theta} \sum_{\tau \in T} p_{\theta\tau} h^{(c)}(\theta, \tau - x_c \cos\theta - y_c \sin\theta). \tag{2.11}$$

All required evaluations of $h^{(c)}$ in this expression are defined according to Eq. (2.10). Comparing Eqs. (2.3) and (2.11), we see that for the selected pixel c, the result of applying the linear algebraic method S is equivalent to applying the FBP algorithm with the angle-dependent filter $\tilde{g} = h^{(c)}$. We refer to such a filter as an *algebraic filter* (AF).

Just as in FBP, the AF determines the weight of the contribution of each detector position to the reconstructed value. This brings up the

question if, similar to FBP, the same filter $h^{(c)}$ can also be used to reconstruct pixels other than c. Note that as $h^{(c)}(\theta,\tau)$ is only defined for $\tau \in T - t_c^{(\theta)}$, this will require the domain of $h^{(c)}(\theta,\cdot)$ to be extended by interpolation.

One may expect that at least for pixels j that are near c, we have

$$u_j \approx \sum_{\theta\in\Theta}\sum_{\tau\in T} p_{\theta\tau} h^{(c)}(\theta, \tau - x_j\cos\theta - y_j\sin\theta). \tag{2.12}$$

This approximation can be interpreted as follows: suppose that both the reconstruction region and the detector (for all angles) are shifted such that their relative position with respect to pixel j is the same as the relative position of the original geometry with respect to pixel c. Then Eq. (2.12) is an exact equality.

We point out that the filter $h^{(c)}$ depends on the particular pixel c, and may vary throughout the image domain, which may offer an advantage for the algebraic method compared to the approximation given in Eq. (2.12), based on a single filter. In particular, algebraic methods have the capability (by their very definition) to confine all the intensity of the object within the reconstruction grid, which cannot be accomplished by FBP methods. Still, as will be demonstrated in Section 2.5, even a single filter can already approximate the reconstruction properties of the underlying algebraic method quite accurately.

To compute the AF for a given linear algebraic method S for a particular pixel c, the cth row of the reconstruction matrix $\mathbf{S}$ must be computed, which comes down to determining the impulse response of pixel c for all detector positions. Let $\boldsymbol{e}_{\theta\tau} \in \mathbb{R}^m$ denote the unit vector which has a value of 1 for the entry corresponding to (θ,τ). Then $s_{\theta\tau}^{(c)}$ is given by

$$s_{\theta\tau}^{(c)} = [S(\boldsymbol{e}_{\theta\tau})]_c. \tag{2.13}$$

Therefore, the method S must be applied separately for each $(\theta,\tau) \in \Theta\times T$ to compute all filter coefficients $s_{\theta\tau}^{(c)}$.

In the experiments that will be presented in the next section, we will focus on the AF for the *central pixel* of the reconstruction grid, i.e., a

pixel centered at the origin. For this pixel c, we have $t_\theta^{(c)} = 0$ for all θ and therefore

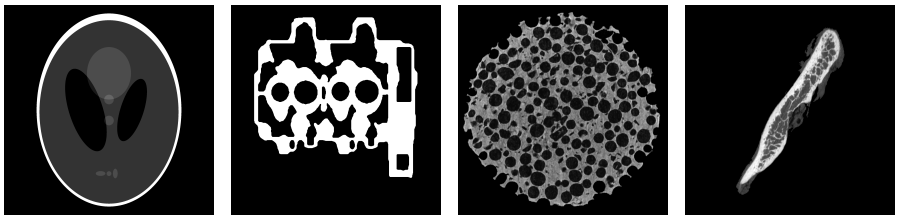
$$h^{(c)}(\theta, \tau) = s_{\theta\tau}^{(c)} \qquad \text{for all } \theta \in \Theta, \tau \in T. \tag{2.14}$$

To use this filter for projection angle θ in an FBP implementation, the discrete representation of $h^{(c)}(\theta, \cdot)$ must first be zero-padded, after which the Discrete Fourier Transform is applied. The resulting filter in the Fourier domain $H^{(c)}(\theta, \cdot)$ can then be applied to the Fourier representation of the projection data in exactly the same way as the Ramp filter, or other common filters. Note that the AF is different for each projection angle, while most common filters do not depend on the angle. This property does not have a significant impact on the running time of the filtering operation.

When computing the AF for a single pixel c, the required (sequential) computation time is mV, where V is the computation time of a single run of the algebraic method. For large image sizes, with many projections, the computational cost of computing a single filter can be substantial and several orders of magnitude larger than the cost of computing a single algebraic reconstruction. However, the resulting filter does not depend in any way on the scanned object. If the geometrical parameters (projection angles, detector size and position) of the scanning device are fixed, the same filter can be used for reconstructing an arbitrarily large number of datasets. In the vast majority of commercial CT-scanners, only a few different acquisition schemes are used for a particular scanner. The filters for these schemes can be computed once, using a separate computer system, or even a large cluster. Subsequently, reconstructions can be computed at the same speed as standard-FBP.

2.5 EXPERIMENTS

A set of experiments has been performed to compare the accuracy of reconstructions computed by AF-FBP, using an algebraic filter based on the iterative SIRT algorithm, with FBP using a standard filter and with SIRT. Note that no priors are used for the SIRT reconstructions or for the derivation of the SIRT-based filters. In this section we will describe the setup of the experiments.



(a) Phantom 1 (b) Phantom 2 (c) Phantom 3 (d) Phantom 4

Figure 2.2: Phantom images of size 2044×2044 used for the experiments.

2.5.1 *General Design of the Experiments*

Four phantom images have been used for the experiments; see Fig. 2.2. The first phantom is the well-known Shepp-Logan phantom. Phantom 2 represents a cross-section of a cylinder head in a combustion engine. Phantom 3 represents a metal foam, and Phantom 4 is a part of a human mandible. Phantoms 3 and 4 are slightly adjusted reconstructions of experimental μCT data sets. The size of all phantoms is 2044×2044 pixels. Note that the phantom size is a multiple of four. In fact, the reconstructions are computed using a pixel size that is four times as large as the phantom pixel size, in both directions. The features of the phantoms are not aligned with the coarse reconstruction grid, such that partial volume effects can be observed in the reconstructions. The detector consists of 511 bins, each having a width of four image pixels. The size of the detector therefore equals the width of the phantom images. Parallel beam projections are simulated using a ray-driven projector based on the Joseph kernel to determine the contribution of an image pixel to each ray [18]. Per detector bin four rays are traced, thereby ensuring that each image pixel participates with strictly positive weight. Unless stated otherwise, the projection angles of the parallel beam projections are regularly distributed between $0°$ and $180°$. The number of projection angles, denoted by d, may vary during the experiments.

2.5.2 *Quantitative Evaluation of the Reconstruction Algorithms*

The four phantoms are reconstructed using three different reconstruction algorithms. Several series of experiments are performed

to examine the relative reconstruction accuracy of different reconstruction algorithms. The following three reconstruction methods are compared.

1. FBP-RL: FBP with the standard Ram-Lak filter; see Section 2.2.

2. SIRT: The reconstruction method described by Eq. (2.6), with $\omega = 1$, $K = 200$, and $\alpha(\cdot)$ and $\beta(\cdot)$ given by $\alpha(x) = \beta(x) = 1/x$ for $x \in \mathbb{R}$. It converges to a weighted least-squares solution of the system $\boldsymbol{W}\boldsymbol{v} = \mathbf{p}$ of minimal norm. We used the version of SIRT that is described in [17].

3. SIRT-FBP: Filtered Backprojection with an angle-dependent algebraic filter based on SIRT. For every projection angle, the filter coefficients for the central pixel are obtained using Eq. (2.13), where the reconstruction algorithm S corresponds with 200 SIRT iterations.

A square reconstruction grid of $z \times z$ pixels is used during the experiments, where each pixel has the same width and height as a detector bin. Preliminary experiments have shown that in some cases the reconstruction accuracy of SIRT-FBP improves if the filters are created using a reconstruction grid with a size z that is larger than the number of bins of the detector. We denote the number of detector bins by z_0.

The quality of the reconstructed images is computed in the image domain by comparing the reconstruction with the phantom image, and in the projection domain by comparing the projections of the reconstructed image with the projections of the phantom. For a reconstruction $\mathbf{u} \in \mathbb{R}^{z^2}$ of size $z \times z$ (with $z \geqslant z_0$), let $\tilde{\mathbf{u}} \in \mathbb{R}^{z_0^2}$ denote the subimage of $\mathbf{u}$ of size $z_0 \times z_0$ with the same central pixel as $\mathbf{u}$. Furthermore, let $\hat{\mathbf{u}}$ be a high resolution version of $\tilde{\mathbf{u}}$, obtained by replacing every pixel in $\tilde{\mathbf{u}}$ by 4×4 small pixels with the same intensity. Note that the image $\hat{\mathbf{u}} = (\hat{u}_{ij})$ has the same pixel resolution as the phantom image $\boldsymbol{v} = (v_{ij})$. Define the *mean reconstruction error* E_r by

$$E_r = \frac{\sum_{i,j} |\hat{u}_{ij} - v_{ij}|}{\sum_{i,j} v_{ij}}. \tag{2.15}$$

Denote the forward projection of $\mathbf{u}$ by $\mathbf{q} = (q_{\theta t})$. Then the *mean projection error* E_p is defined by

$$E_p = \frac{\sum_{\theta \in \Theta} \sum_{t \in T} |q_{\theta t} - p_{\theta t}|}{\sum_{\theta \in \Theta} \sum_{t \in T} p_{\theta t}}. \tag{2.16}$$

When comparing FBP-RL with SIRT and SIRT-FBP, the FBP-RL and SIRT-FBP reconstruction are always computed on a grid of size $z_0 \times z_0$, whereas the grid size $z \times z$ for SIRT can vary. When computing the projection error, the full $z \times z$ reconstruction is used to determine the forward projection of the SIRT reconstructions.

In the first series of experiments, the effect of the size of the reconstruction grid on the reconstruction accuracy of the algorithms is examined. In the second series of experiments the size of the reconstruction grid is kept fixed and the accuracy of the reconstructions from the three methods is examined as a function of the number of projection angles d, where the angles are regularly distributed between $0°$ and $180°$.

In certain tomography applications, notably electron tomography, the angular range of the projections is limited. Contrary to standard FBP, algebraic methods can easily be adapted to such limited-angle geometries by adjusting the projection matrix. In the third series of experiments, the dependence of the reconstruction accuracy on the angular range is examined for the three methods. In these experiments, the step between consecutive angles is kept fixed at $0.5°$ and an angular range of $180°$ corresponds to full angular range with $d = 360$. This means that the angular range and the *number* of projection angles vary simultaneously.

The experiments described above are based on noiseless projection data. In the fourth series of experiments the robustness of the reconstruction algorithms with respect to noise is examined. First, noiseless projection data are computed. Poisson distributed noise with varying I_0 (number of counts per detector element, measured without an object) is applied to this data. The reconstruction quality of FBP with the standard Ram-Lak is known to degrade for high noise levels. Various alternative filters have been described in the literature that are more robust to noise, including the Shepp-Logan (SL), Cosine (Cos), Hamming (Ham), and Hann (Hann) filters [4, 19]. The reconstruction

results for SIRT and SIRT-FBP are compared with the results for FBP using this range of filters.

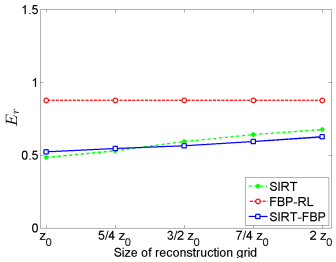
The AFs used in the previous experiments can vary throughout the projection angles. In the fifth set of experiments, we consider an *angle independent* filter, computed by taking the average of the filters for all projection angles. This way of filtering the projection data enables an implementation very similar to FBP with a standard filter. The corresponding reconstruction algorithm will be denoted by av-SIRT-FBP and it is compared with SIRT, SIRT-FBP and with FBP for several standard filters.

Finally, we examine the accuracy of the three reconstruction methods when reconstructing an experimental μCT data set. A diamond was scanned using a Skyscan 1172 cone-beam μCT scanner. From the experimental data, 200 parallel beam projections of the central slice were determined by rebinning the corresponding fan-beam projection data. Reconstructions computed from these projections using FBP-RL, SIRT and SIRT-FBP are compared.

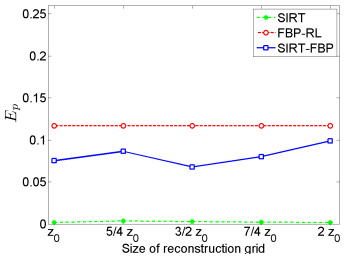
As noted before, computing an AF is highly computationally intensive. An optimized GPU implementation of the SIRT algorithm was used to generate the filters, running on a Tesla S1070 quad-GPU system. As an example of the running time, we mention computing a filter for an image of size 511×511, using 64 angles and a detector of size 511. The running time for 200 iterations of the SIRT algorithm on a single GPU is around 0.5 s. To compute the AF for all angles, this computation must be carried out 64×511 times, resulting in a total running time of 4.5 h.

2.6 RESULTS

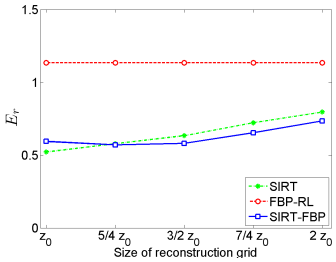
In this section we provide an overview of the results of the experiments. In addition to the quantitative results concerning the reconstruction and projection errors, a selection of the reconstructed images is shown, which is chosen such that it illustrates the behavior of the reconstruction methods.



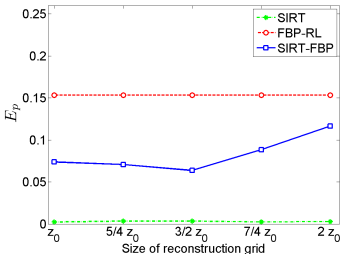
(a) Phantom 1: E_r



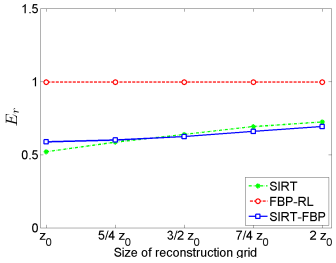
(b) Phantom 1: E_p



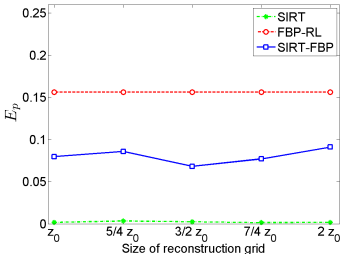
(c) Phantom 2: E_r



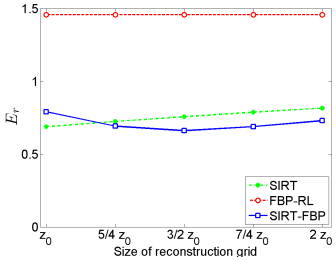
(d) Phantom 2: E_p



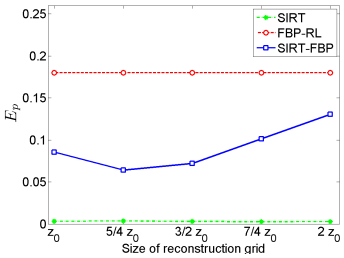
(e) Phantom 3: E_r



(f) Phantom 3: E_p

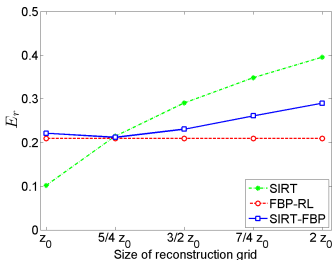


(g) Phantom 4: E_r

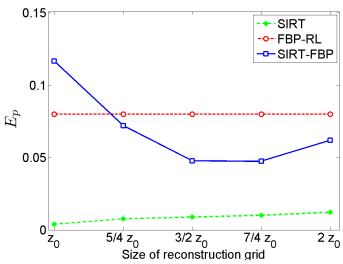


(h) Phantom 4: E_p

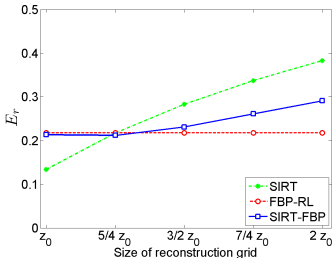
Figure 2.3: Mean reconstruction and projection error as a function of the size of the reconstruction grid, with $d = 32$.



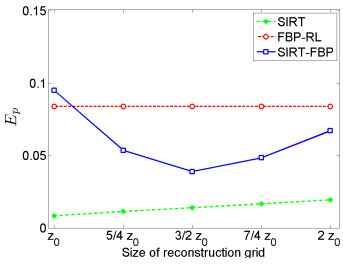
(a) Phantom 1: E_r



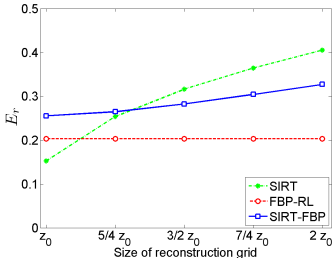
(b) Phantom 1: E_p



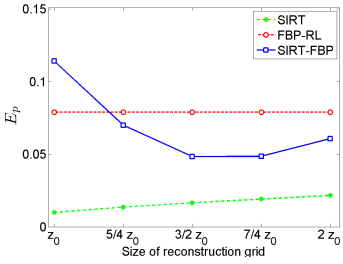
(c) Phantom 2: E_r



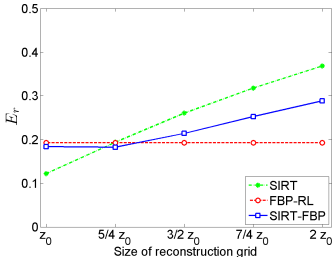
(d) Phantom 2: E_p



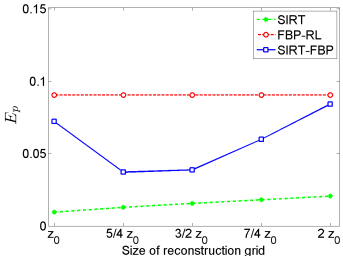
(e) Phantom 3: E_r



(f) Phantom 3: E_p



(g) Phantom 4: E_r



(h) Phantom 4: E_p

Figure 2.4: Mean reconstruction and projection error as a function of the size of the reconstruction grid, with $d = 256$.

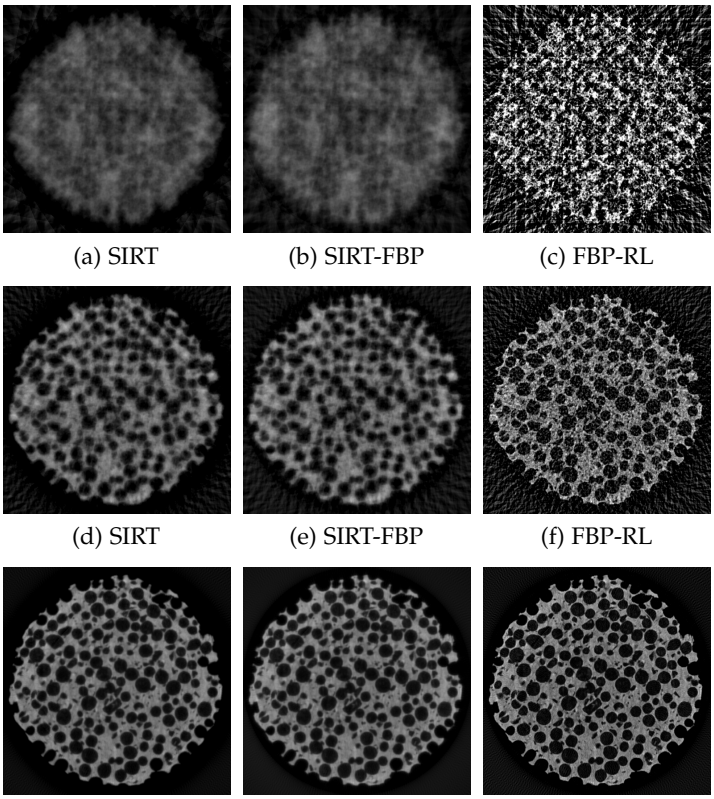
2.6.1 *Varying the Size of the Reconstruction Grid*

As the SIRT reconstruction depends on the size of the reconstruction grid z, the filters that are based on SIRT reconstructions also depend on z, thereby affecting the quality of the SIRT-FBP reconstructions. In the first series of experiments we examine the accuracy of the reconstructions as a function of z. During the first run, the relatively low number of 32 projection angles is used, while for the second run a larger set of 256 projection angles is used.

Reconstruction errors are shown in Fig. 2.3 and Fig. 2.4, for $d = 32$ and $d = 256$ angles, respectively, where the mean reconstruction error is plotted in the first column and the mean projection error in the second column. As expected, the errors of the FBP-RL reconstructions are independent of z. Since the number of unknowns of the equation system that is solved by SIRT increases with increasing z, while the number of equations remains the same, the system becomes increasingly underdetermined. This results in decreasing reconstruction accuracy of SIRT as a function of z. The results show that the reconstruction error for SIRT-FBP is minimal when z is between z_0 and $\frac{3}{2}z_0$. For larger grid sizes the decreasing quality of the SIRT reconstructions determines the behavior of the errors of the SIRT-FBP reconstructions. The minimal mean projection error occurs close to $z = \frac{3}{2}z_0$, and the error of SIRT-FBP for this grid size is significantly less than the mean projection error of FBP-RL. Furthermore, the quality of the SIRT-FBP reconstructions with $d = 32$ exceeds that of FBP-RL and SIRT on grids with z between $\frac{3}{2}z_0$ and $2z_0$. Such results are not to be expected for $d = 256$, since for a sufficiently large number of angles FBP-RL is known to outperform SIRT. Still we see that SIRT-FBP has a significantly smaller reconstruction error than SIRT on these grid sizes.

2.6.2 *Varying the Projection Angles*

In the second series of experiments the number of projection angles d is varied between 16 and 256 angles, while z is kept fixed at $z = \frac{3}{2}z_0$, based on the results of the previous experiments. Some reconstructions of Phantom 3 are shown in Fig. 2.5. Fig. 2.6 shows that, for all considered numbers of projection angles, the mean projection error of SIRT-FBP is smaller than the mean projection error of the FBP-RL re-



(a) SIRT (b) SIRT-FBP (c) FBP-RL

(d) SIRT (e) SIRT-FBP (f) FBP-RL

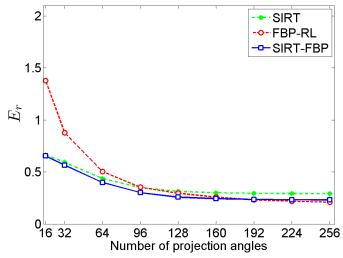
(g) SIRT (h) SIRT-FBP (i) FBP-RL

Figure 2.5: Reconstructions of phantom 3 with $z = (3/2)z_0$ and varying number of projection angles. Top row: $d = 16$. Middle row: $d = 64$. Bottom row: $d = 256$.

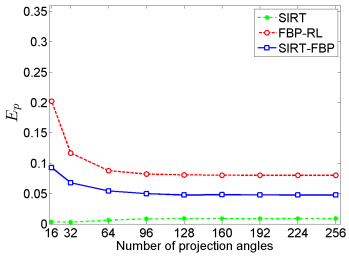
constructions. For most phantoms, the projection error is also smaller for SIRT-FBP than for FBP-RL, as long as the number of angles is not too large (i.e., at most 192).

2.6.3 *Varying the Angular Range*

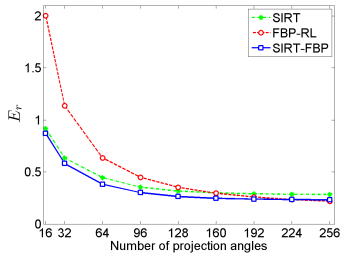
In this series of experiments the angular range is varied between 5° and 170°. Fig. 2.7 shows the difference between Phantom 1 and its reconstructions for a selection of the considered limited angular ranges, i.e. 35°, 80°, 125° and 70°, comparing SIRT, SIRT-FBP and FBP-RL. The values of the difference plots range between black and white, where a



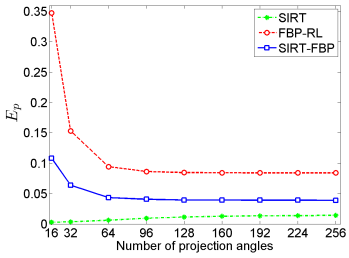
(a) Phantom 1: E_r



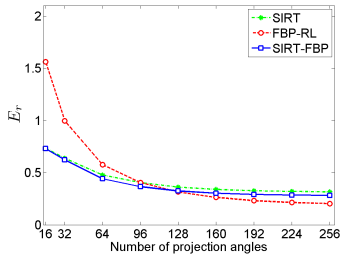
(b) Phantom 1: E_p



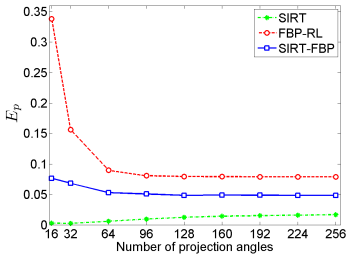
(c) Phantom 2: E_r



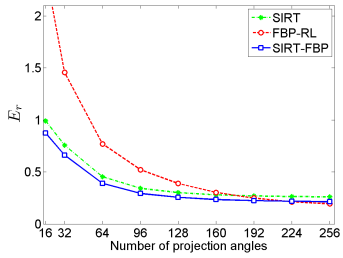
(d) Phantom 2: E_p



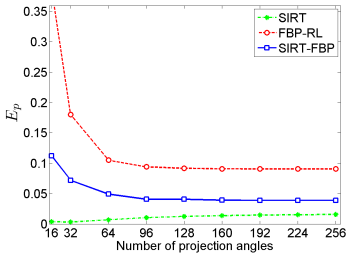
(e) Phantom 3: E_r



(f) Phantom 3: E_p

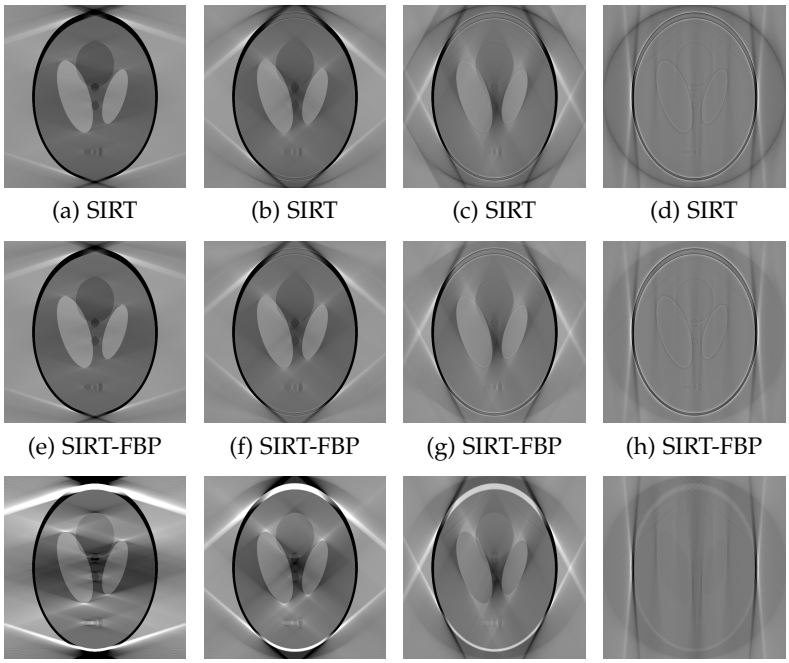


(g) Phantom 4: E_r



(h) Phantom 4: E_p

Figure 2.6: Mean reconstruction and projection error as a function of the number of projection angles with $z = (3/2)z_0$.



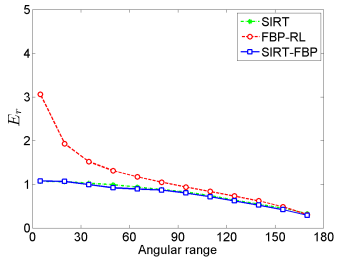
(a) SIRT (b) SIRT (c) SIRT (d) SIRT

(e) SIRT-FBP (f) SIRT-FBP (g) SIRT-FBP (h) SIRT-FBP

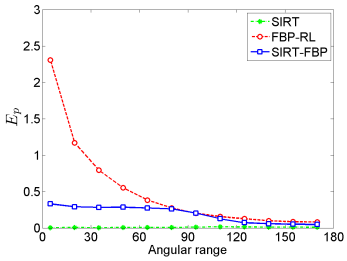
(i) FBP-RL (j) FBP-RL (k) FBP-RL (l) FBP-RL

Figure 2.7: Difference between the original image and reconstructions of phantom 1 with $z = (3/2)z_0$ and varying angular range; limited angular range per column (from left to right): 35°, 80°, 125°, 170°. Dark (light) pixels correspond to reconstruction values that are higher (smaller) than those of the original phantom.

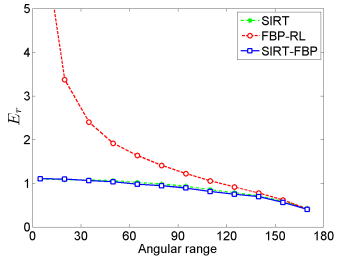
pixel is black if the corresponding reconstruction pixel has a value of at least $(3/2)$ the value of the phantom pixel, and a pixel is white if the corresponding reconstruction pixel has a value of at most $(1/2)$ the value of the phantom pixel. The difference plots for SIRT and SIRT-FBP are very similar, while FBP-RL results in quite different reconstructions. The results in Fig. 2.8 show the accuracy of the reconstructions when the angular range is limited, for the three methods. The projection error is calculated using only those projections that are included in the angular range. The reconstruction errors of SIRT-FBP are similar to those of SIRT reconstructions, while for an angular range below 150° the reconstruction errors of SIRT-FBP are much smaller than the errors of FBP-RL for all phantoms.



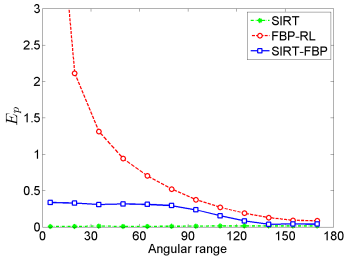
(a) Phantom 1: E_r



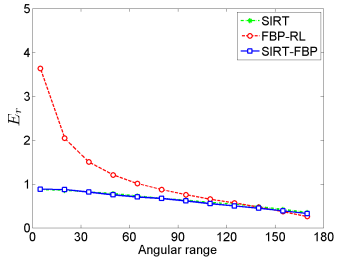
(b) Phantom 1: E_p



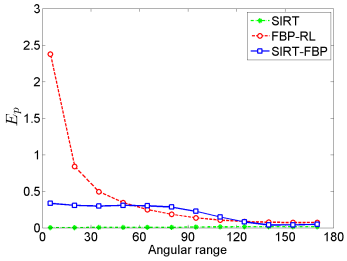
(c) Phantom 2: E_r



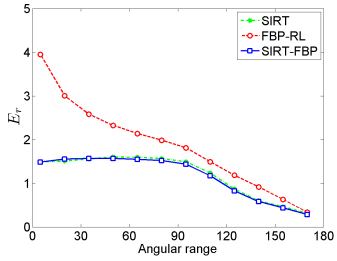
(d) Phantom 2: E_p



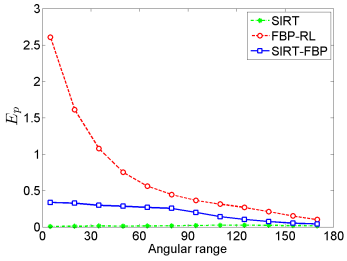
(e) Phantom 3: E_r



(f) Phantom 3: E_p



(g) Phantom 4: E_r



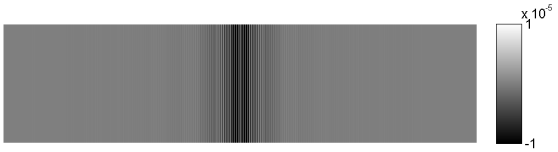
(h) Phantom 4: E_p

Figure 2.8: Mean reconstruction and phantom error as a function of the angular range with $z = (3/2)z_0$.

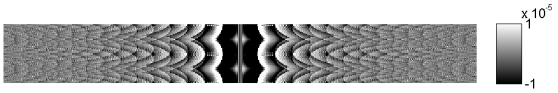
The projection error of SIRT-FBP reconstructions is smaller than that of FBP-RL for almost all test cases. Since SIRT-FBP is an approximation of SIRT, some discrepancy exists between the SIRT and SIRT-FBP reconstructions. The magnitude of this effect depends on the particular image, which can lead to an inferior reconstruction quality of SIRT-FBP compared to FBP-RL in certain cases. An example is Phantom 3 in Fig. 2.8 for the limited angular range between 60° and 120°. The magnitude of this effect depends on the particular image. The results also demonstrate that for the cases where iterative reconstruction methods perform well, i.e. few projection angles or highly limited angular range, SIRT-FBP clearly outperforms FBP-RL.

2.6.4 *Variations in the Filters*

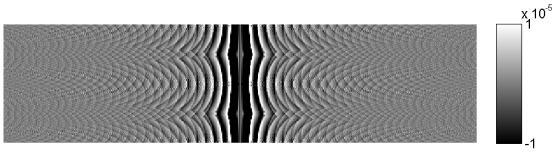
Since the operation of the SIRT algorithm depends on both the size of the reconstruction grid and the set of projection angles, the corresponding filters also depend on these parameters. Fig. 2.9 shows a selection of filters computed for the central pixel for a varying number of projection angles and varying grid size, respectively. Every row in these grayscale figures represents a filter. Consecutive rows correspond to consecutive angles and the first row corresponds to the filter for an angle of 0°. The standard Ram-Lak filter, which does not depend on the projection angle, is also shown for comparison. The SIRT-based filters corresponding to an angle of 0° for $d = 64$ and $d = 256$ are also shown as line plots in Fig. 2.10 together with the standard Ram-Lak filter. Note that the algebraic filters depend on the particular projection angle. As the reconstruction grid in our experiments is a square, the algebraic reconstruction problem is not rotationally invariant, not even when pixel discretization effects are neglected. In the grayscale figures that show the SIRT-based filters some irregularities appear in the form of lines. These may be attributed to aliasing effects due to discretization of both the detector and the image domain. For the case of limited angle tomography, we found that presenting a thorough, yet compact overview of the filter variations is not straightforward. The filters may vary significantly depending on the particular projection angle.



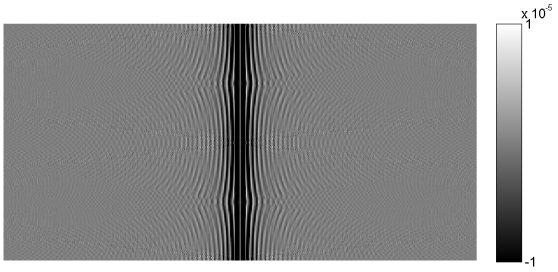
(a) Ram-Lak filter, $z = \frac{3}{2}z_0$, $d = 128$



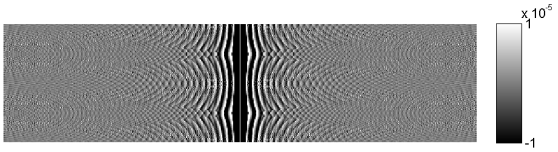
(b) SIRT-based filters, $z = \frac{3}{2}z_0$, $d = 64$



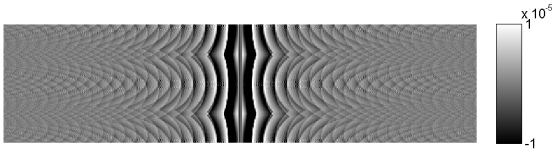
(c) SIRT-based filters, $z = \frac{3}{2}z_0$, $d = 128$



(d) SIRT-based filters, $z = \frac{3}{2}z_0$, $d = 256$

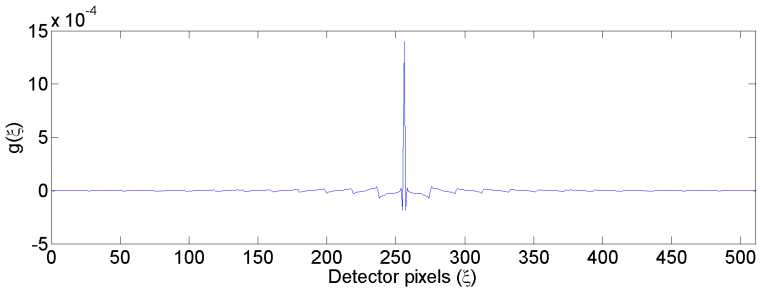


(e) SIRT-based filters, $z = z_0$, $d = 128$

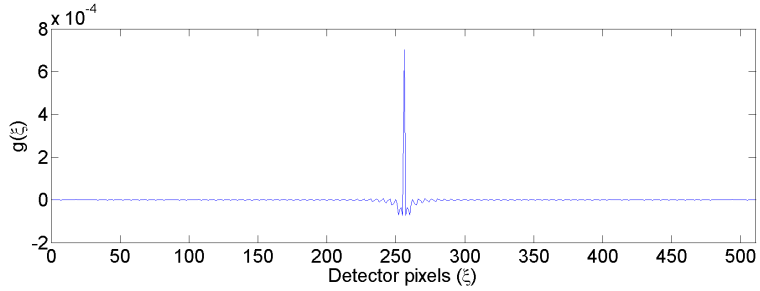


(f) SIRT-based filters, $z = 2z_0$, $d = 128$

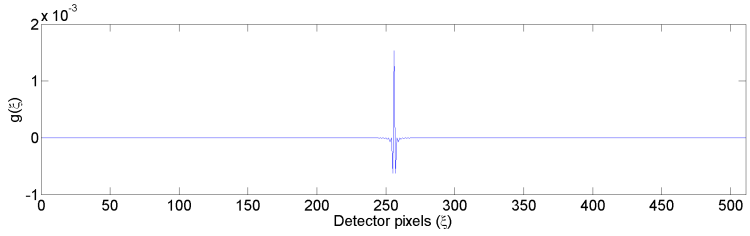
Figure 2.9: Grayscale representations of various filters, where each row represents a filter for some projection angle. In all cases, a detector with 511 bins was used (z_0). Both the size of the reconstruction grid z and the number of projection angles d is varied. The range of the gray scale is $[-1, 1] * 10^{-5}$.



(a) Filter for $\theta = 0°$, $z = (3/2)z_0$, $d = 64$



(b) Filter for $\theta = 0°$, $z = (3/2)z_0$, $d = 256$

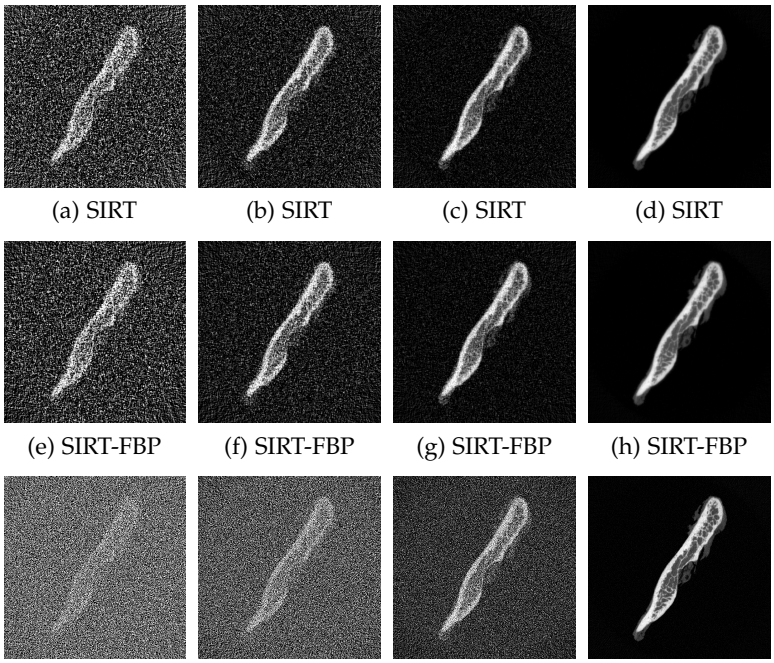


(c) Ram-Lak filter

Figure 2.10: Two AFs used in SIRT-FBP and the standard Ram-Lak filter in the spatial domain.

2.6.5 *Noise*

All experiments so far were performed using noiseless projection data. FBP-RL is known to produce poor quality reconstructions from projection data that are highly polluted with noise, while SIRT handles this data relatively well. In this series we examine the accuracy of SIRT-FBP compared to FBP with several common filters and SIRT in case of noisy projection data, where the detector count I_0 ranges from $10^2 - 10^6$. Some reconstructions are shown in Fig. 2.11. The re-



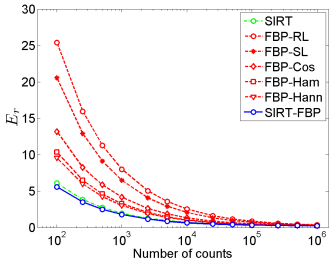
(a) SIRT (b) SIRT (c) SIRT (d) SIRT

(e) SIRT-FBP (f) SIRT-FBP (g) SIRT-FBP (h) SIRT-FBP

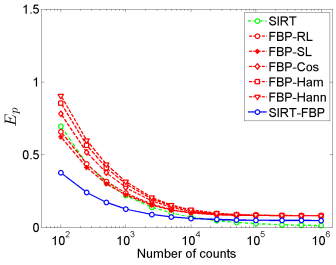
(i) FBP-RL (j) FBP-RL (k) FBP-RL (l) FBP-RL

Figure 2.11: Reconstruction of phantom 4 with $z = (3/2)z_0$ and varying noise levels, I_0 per column (from left to right): 250, 1000, 5000, 10^6.

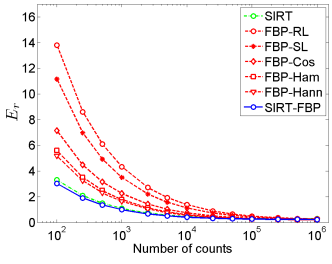
sults in Fig. 2.12 show that the reconstruction errors of SIRT-FBP and SIRT are very similar and that the reconstruction and projection errors of SIRT-FBP are significantly lower than the corresponding errors of FBP reconstructions with any standard filter used in this experiment. Surprisingly, the projection errors for SIRT-FBP are smaller than for SIRT for very high noise levels. This can be attributed to the fact that for SIRT, noise can accumulate in corners of the reconstruction grid, where the intersection between projected lines and the image domain is very small for certain projection angles. These effects may cause numerical instabilities, resulting in a somewhat larger projection error, which does not occur for SIRT-FBP as only the angle-dependent filter for the central pixel is employed there. We remark that one cannot say that SIRT-FBP is more robust to noise than FBP in general, as SIRT-FBP itself is just a variant of FBP with an appropriately chosen filter.



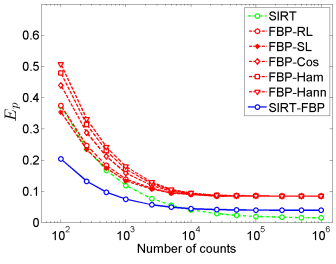
(a) Phantom 1: E_r



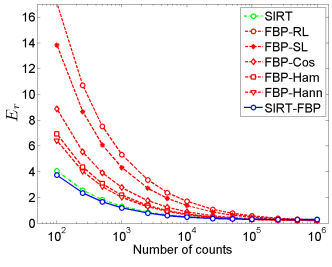
(b) Phantom 1: E_p



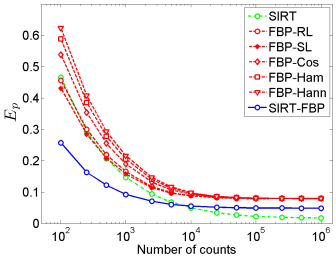
(c) Phantom 2: E_r



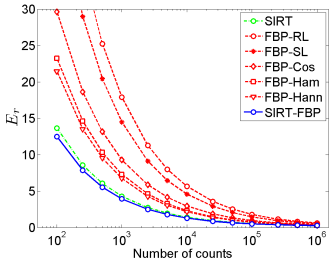
(d) Phantom 2: E_p



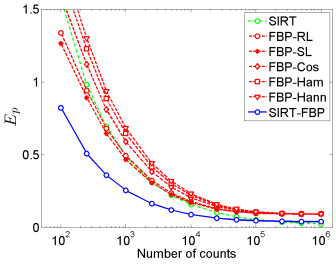
(e) Phantom 3: E_r



(f) Phantom 3: E_p

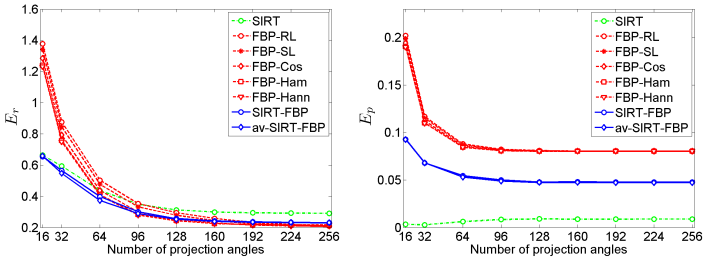


(g) Phantom 4: E_r

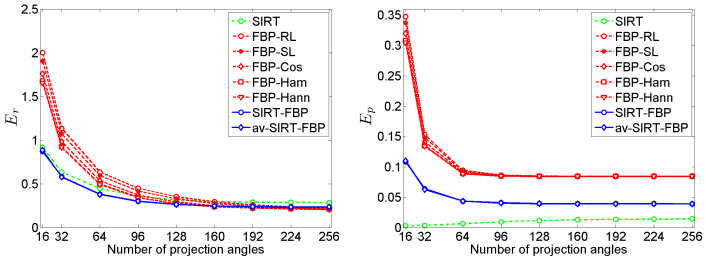


(h) Phantom 4: E_p

Figure 2.12: Mean reconstruction and projection error as a function of the noise level with $z = (3/2)z_0$ and $d = 256$.



(a) Phantom 1: E_r (b) Phantom 1: E_p

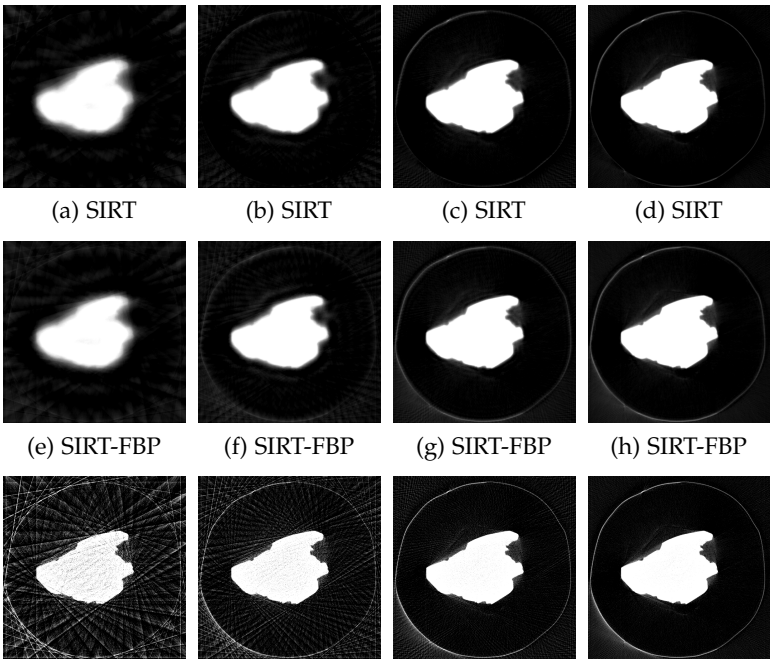


(c) Phantom 2: E_r (d) Phantom 2: E_p

Figure 2.13: Mean reconstruction and projection error of FBP with several standard filters, SIRT, SIRT-FBP, and av-SIRT-FBP, as a function of the number of projection angles with $z = (3/2)z_0$.

2.6.6 *Experiments with an angle independent filter*

In this series of experiments, av-SIRT-FBP is compared with SIRT-FBP, SIRT and with FBP based on various standard filters. Although the implementation of av-SIRT-FBP is similar to FBP with a standard filter, the quality of av-SIRT-FBP reconstructions is comparable to that of SIRT-FBP reconstructions, as can be seen in Fig. 2.13 for Phantoms 1 and 2. The reconstructions of Phantoms 3 and 4 show similar patterns and are therefore not included. We remark that in some cases, av-SIRT-FBP actually results in a smaller reconstruction error compared to both SIRT and SIRT-FBP. As all experiments presented here deal with underdetermined systems of linear equations, even SIRT cannot be expected to converge to the phantom image. As both SIRT-FBP and av-SIRT-FBP are approximations to SIRT, they sometimes perform better and sometimes worse than SIRT.



(a) SIRT (b) SIRT (c) SIRT (d) SIRT

(e) SIRT-FBP (f) SIRT-FBP (g) SIRT-FBP (h) SIRT-FBP

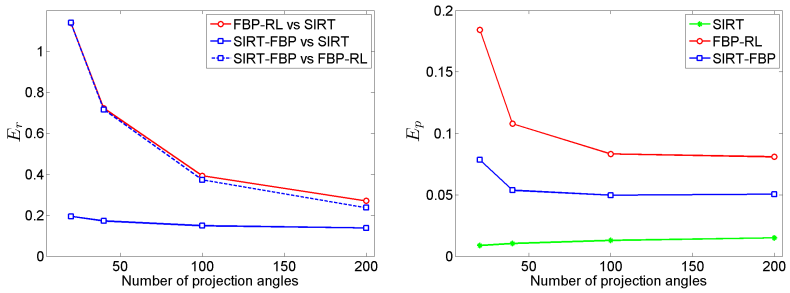
(i) FBP-RL (j) FBP-RL (k) FBP-RL (l) FBP-RL

Figure 2.14: Diamond reconstruction with $z = (3/2)z_0$ from experimental μCT data; d per column (from left to right): 20, 40, 100, 200.

2.6.7 *Experimental μCT data*

For the final experiments, an experimental μCT data set is used. The dataset was acquired by scanning a raw diamond using a Skyscan 1172 μCT scanner. Cone-beam projection data were acquired for 400 angles in a full 360° angular range, using an angular step size of 0.9°. The camera pixel size was 41μm. The projection data for the central slice are effectively fan-beam data. Since the FBP algorithm and filter derivation used in this chapter are based on a parallel beam geometry, the projection data were rebinned to a parallel beam geometry, forming a dataset of 200 parallel beam projections equally distributed between 0° and 180°. Each of the projections consists of 511 detector values.

Since the ground truth data are not available, the reconstructions are compared with *each other* to analyze the reconstruction behavior for the different algorithms. The results in Fig. 2.15 show that the error



(a) Diamond: E_r (b) Diamond: E_p

Figure 2.15: Reconstruction errors as a function of the number of projection angles for the experimental μCT diamond data with $z = (3/2)z_0$. As there is no ground truth data available, a comparison is made between pairs of reconstructions computed by different algorithms.

of the SIRT-FBP reconstructions behaves very similar to that of SIRT reconstructions, while the difference between SIRT-FBP and FBP-RL reconstructions is comparable to that between SIRT and FBP-RL reconstructions. The reconstructions in Fig. 2.14 show the same behavior patterns. For low numbers of projection angles the mean projection error of SIRT-FBP is much less than that of FBP-RL.

2.7 CONCLUSIONS

In this chapter, we have presented an algorithmic approach for computing *AFs* that can be used within the framework of the well-known FBP algorithm. The resulting AF-FBP reconstructs objects with the computational efficiency of FBP, while maintaining the more robust reconstruction properties of the chosen algebraic reconstruction technique. We have presented a formal description of the AFs and examined their properties in several series of experiments for the algebraic reconstruction method SIRT. The results showed that SIRT-FBP reconstructions are very similar to SIRT reconstructions. Therefore, AF-FBP could be applied in situations, where FBP with standard filters was known to produce low quality reconstructions and algebraic reconstruction techniques yield superior results, such as low dose tomography, limi-

ted angle tomography, etc. Computing the AF has a high computational cost. However, for a fixed scanning geometry and a fixed set of projection angles, this computation must be performed only once, as it does not depend on the scanned object. The filter computation should, therefore, be considered as a calibration step, which can be performed in an off-line setting. In further research, the properties of AF-FBP will be analyzed in more detail and generalizations to other common geometries, such as various cone-beam geometries will be considered.

ACKNOWLEDGEMENTS

The authors would like to acknowledge W. J. Palenstijn from the Vision Lab., University of Antwerp, Wilrijk, Belgium, for developing the high-performance GPU code used to generate the algebraic filters and assisting in the computations. They are grateful to J. Sijbers, also from the Vision Lab., for several useful discussions.

BIBLIOGRAPHY

[1] F. Natterer. *The Mathematics of Computerized Tomography*. Philadelphia: SIAM, 2001.

[2] A. C. Kak and M. Slaney. *Principles of Computerized Tomographic Imaging*. Philadelphia: SIAM, 2001.

[3] T. M. Buzug. *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT*. Berlin: Springer, 2008.

[4] G. T. Herman. *Fundamentals of Computerized Tomography: Image Reconstruction from Projections*. Berlin: Springer, 2009.

[5] X. Pan, E. Y. Sidky, and M. Vannier. Why do commercial CT scanners still employ traditional, filtered back-projection for image reconstruction? *Inv. Problems* 2009; 25(12): 123009.

[6] L. A. Feldkamp, L. C. Davis, and J. W. Kress. Practical cone-beam algorithm. *J. Opt. Soc. Am.* 1984; 1(A6): 612–619.

[7] F. Xu and K. Mueller. Accelerating Popular Tomographic Reconstruction Algorithms On Commodity PC Graphics Hardware. *IEEE Trans. Nucl. Science* 2005; 52(3): 654–663.

[8] D. Castano-Diez, H. Mueller, and A. S. Frangakis. Implementation and performance evaluation of reconstruction algorithms on graphics processors. *Journal of Structural Biology* 2007; 157(1): 288–295.

[9] J. K. Older and P. C. Johns. Matrix formulation of computed tomogram reconstruction. *Physics in Medicine and Biology* 1993; 38(8): 1051–1064.

[10] R. Clack. Towards a complete description of three-dimensional filtered backprojection. *Physics in Medicine and Biology* 1992; 37(3): 645–660.

[11] H. Kunze et al. Pre-calculation of the image quality of the Simultaneous Iterative Reconstruction Technique. *Proc. of SPIE Medical Imaging*. 2007: 6510: 65105E.

[12] A. Markoe. *Analytic Tomography*. New York: Cambridge University Press, 2006.

[13] J. Hsieh Y. Wei G. Wang. An intuitive discussion on the ideal ramp filter in computed tomography (I). *J. Comput. Math. Appl.* 2005; 49(5–6): 731–740.

[14] A. K. Louis and Th. Schuster. A Novel Filter Design Technique in 2D - Computerized Tomography. *Inverse Problems* 1995.

[15] H. Kunze et al. Filter determination for Tomosynthesis aided by iterative reconstruction techniques. *Proc. of Fully3D* 2007: 309–312.

[16] J. Ludwig et al. A Novel Approach for Filtered Backprojection in Tomosynthesis Based on Filter Kernels Determined by Iterative Reconstruction Techniques. *Proc. of IWDM* 2008; 5116: 612–620.

[17] J. Gregor and T. Benson. Computational analysis and improvement of SIRT. *IEEE Trans. Med. Imag.* 2008; 27(7): 918–924.

[18] P. M. Joseph. An improved algorithm for reprojecting rays through pixel images. *IEEE Trans. Med. Imag.* 1982; 1(3): 192–196.

[19] L. A. Shepp and B. F. Logan. The Fourier reconstruction of a head section. *IEEE Trans. Nucl. Science* 1974; 21(3): 21–43.