4

Local approximation of advanced regularized iterative methods

4.1 Introduction

The goal of tomography is to reconstruct an object given its projections for different angles. Using tomography, it is possible to nondestructively examine the interior of objects, which makes it useful for many applications. Examples of tomography in practice include computed tomography in medicine and electron tomography in materials science. Because of its practical usefulness, many algorithms have been developed to perform tomographic reconstruction. An overview of past research on tomography can be found in [KS01; Nat01; Buz08]. Two types of reconstruction methods are commonly used: *analytical* methods, which discretize a continuous inversion formula of the problem, and *algebraic* methods, in which a linear system that represents the problem is solved.

In many applications of tomography, it is impossible to acquire a large number of noise-free projections. For example, when scanning live animals, there is a limit on the total dose deposited on the animal during the experiment [Lov+13]. In electron

This chapter is based on:

D. M. Pelt and K. J. Batenburg. "A method for locally approximating advanced regularized iterative tomographic reconstruction methods". *IEEE Transactions on Image Processing* (Submitted for publication).

tomography, the scanned sample is damaged by the electron beam, which leads to a limit on the number of projections that can be acquired [MDG95]. In these cases, standard reconstruction methods often fail to produce reconstructions with adequate quality for further analysis [Lov+13]. For analytical methods, the reason for this is that the continuous inversion formulas on which they are based assume that noise-free projections are available for *all* angles. In algebraic methods, the linear system that is solved is typically both underdetermined and ill-conditioned, which can make it difficult to find accurate reconstructions when the available projection data is limited and/or noisy.

Recently developed advanced reconstruction methods aim to improve reconstruction quality by exploiting prior knowledge about the scanned object or scanning system. Often, these methods add additional terms to the objective function that is minimized in standard algebraic methods. Methods of this type will be called *regularized iterative methods* in this chapter. For example, if it is known beforehand that the physical quantity that is reconstructed cannot be negative, a nonnegativity constraint can be added to the objective function to improve the reconstruction quality. If it is known that the scanned object has a sparse boundary, total variation minimization can be applied by adding a term that minimizes the gradient of the reconstructed image [SP08]. If the added prior knowledge is appropriate for the acquired data, regularized iterative methods can be extremely successful in reconstructing objects from (highly) limited data [BS11; Kos+13].

One of the main disadvantages of regularized iterative methods is their computational cost, which is typically very high. A high computational cost of a reconstruction method can be prohibitive for its application in practice. For example, in ultrafast tomographic experiments at synchrotrons, the computation time of the reconstruction method has to match the high speed of the acquisition of projection data [Mok+13]. An additional problem is that regularized iterative methods often have a number of tunable parameters that influence the reconstruction quality greatly. In many cases, values for these parameters are chosen by trial-and-error, which can be very time-consuming for methods with a high computational cost. These problems are especially important in cases where a large object is scanned, but the features of interest are only located in a small region of the object. Since regularized iterative methods, and the algebraic methods they are based on, minimize a *global* objective function, they typically need to compute the entire volume during reconstruction, which may not fit into the available memory of the graphic processing units used to perform the reconstruction [XM05].

Analytical methods, on the other hand, can be evaluated *locally*: if one is only interested in a small subvolume of the reconstruction, only that subvolume has to be reconstructed. When reconstructing large volumes, analytical methods can divide the reconstruction volume into subvolumes that do fit into the available memory, and reconstruct each subvolume separately, resulting in an efficient method to compute the full reconstruction volume. This property is one of the reasons that in many applications of tomography, standard analytical methods are still the most popular reconstruction methods instead of regularized iterative methods [PSV09].

In this chapter, we present a novel method for approximating a computationally expensive regularized iterative method in a (small) subvolume of the full reconstruction

volume. The proposed method only performs computations in the chosen subvolume, ensuring low computational and memory requirements. If one is only interested in part of the scanned object, the new method can significantly reduce the time needed to reconstruct that part compared to existing regularized iterative methods. If one wants to reconstruct the entire object, the proposed method also allows for significant reduction of computation time by enabling parallel computation of different subvolumes, and it enables regularized iterative reconstruction of large datasets that do not fit completely into the available memory. In addition, the method can be used to quickly estimate parameters of a slow regularized iterative method by estimating them in a small subvolume.

The proposed method is based on approximating standard algebraic methods by a modified analytical method. In recent years, several methods have been proposed that achieve this by modifying the filter that is typically used in analytical methods. In one study, an angle-independent filter is calculated based on analytic analysis of the algebraic SIRT method [Zen12]. An extension of the method for noisy projection data is given in [ZZ13]. In Chapter 2 of this thesis, a method of calculating a data-dependent filter is given. Finally, an angle-dependent and geometry-dependent filter is calculated by repeated application of the SIRT method in [BP12]. A faster method of calculating similar filters for the algebraic SIRT method is proposed in Chapter 3 of this thesis. None of these methods, however, allow for inclusion of popular advanced prior knowledge terms, such as total variation minimization, which can limit their usefulness in practice. In this chapter, we first show the application of the filter of Chapter 3 to locally approximate the algebraic SIRT method. Then, we extend the method to allow for local approximation of a regularized iterative method as well. Finally, we demonstrate that the proposed method is able to produce local reconstructions that are very similar to reconstructions of global regularized iterative methods for various types of exploited prior knowledge.

This chapter is structured as follows. In Section 4.2, we introduce the notations we use throughout the chapter, and formally define the tomographic reconstruction problem and the standard analytical and algebraic approaches. The main contribution of this chapter is given in Section 4.3, where we first apply the method proposed in Chapter 3 to approximate SIRT locally. We then extend this approximation by including prior knowledge in the reconstruction of a subvolume, and give some details on how to implement the resulting method in practice. The experiments we performed to study the new method are explained in Section 4.4, and the results of those experiments are shown in Section 4.5. We conclude in Section 4.6 with a brief summary of the chapter and some final remarks.

4.2 Notation and concepts

In this section, the mathematical notation that we use throughout the chapter is introduced, and a formal definition of the tomographic reconstruction problem is given. The standard analytical and algebraic approaches to the problem are explained, and their mathematical definitions are given. Finally, we explain how prior knowledge can

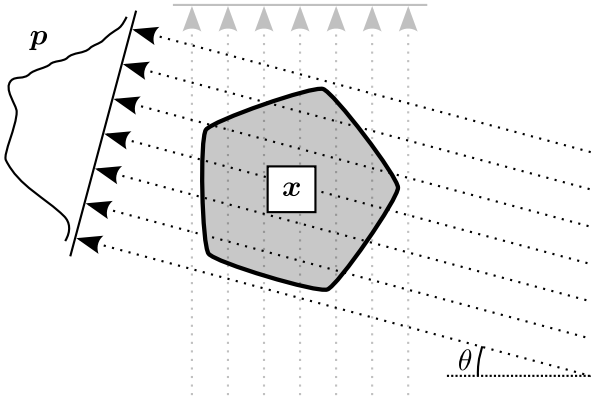


Figure 4.1: The two-dimensional parallel-beam geometry used in this chapter. Parallel lines, rotated by angle θ, pass through the object f. A line $l_{\theta,t}$ has the characteristic equation $t = x\cos\theta + y\sin\theta$, and a projection $P_\theta(t)$ of f is given by the line integral of f over the line $l_{\theta,t}$.

be exploited in algebraic methods by extending their objective functions, resulting in regularized iterative methods.

Notation and problem definition

We focus on two-dimensional parallel-beam tomographic reconstruction problems with a single rotation axis. Note that in many cases it is possible to convert other tomographic geometries, such as cone-beam or spiral tomography, to a parallel-beam geometry by rebinning [GKP00; KSK00]. Parallel-beam projection data are acquired by rotating an array of detectors around the object (or, equivalently, rotating the object), with the detectors of the array located on a straight line. This acquisition scheme is shown graphically in Fig. 4.1. If the number of detectors in the array is denoted by N_d, and the number of rotation angles for which data are acquired is denoted by N_θ, we can write the measured line integrals as a vector $\boldsymbol{p}$ with $N_d N_\theta$ elements, one for each combination of detector element and rotation angle. The reconstructed image is represented as a vector $\boldsymbol{x}$ with N^2 elements, one for each pixel of the $N \times N$ pixel grid on which the reconstruction is calculated. The main problem in tomographic reconstruction is to find the unknown image $\boldsymbol{x}$, given the acquired projection data $\boldsymbol{p}$.

The *forward projection* operator $\mathscr{W} : \mathbb{R}^{N^2} \to \mathbb{R}^{N_d N_\theta}$ is the operator that, for a given projection geometry, corresponds to the discretized line integrals of an object represented on a $N \times N$ pixel grid. Using the above notation, we can write this operator as a $N_d N_\theta \times N^2$ matrix $\boldsymbol{W}$, with element w_{ij} giving the contribution of pixel j to detector i. The transpose of this operator, $\boldsymbol{W}^T$, is called the *backprojection* operator. Typically, a forward projection of an image $\boldsymbol{x}$ is calculated on-the-fly by calculating its line integrals directly [PBS11]. Similarly, multiplying $\boldsymbol{p}$ by $\boldsymbol{W}^T$ is done implicitly by backprojecting $\boldsymbol{p}$ on-the-fly. The advantage of this approach is that the matrix $\boldsymbol{W}$, which can be very large, never has to be stored in memory. Furthermore, forward

projections and backprojections can be computed very efficiently on graphic processor units (GPUs) [XM05; MXN07].

Our novel approach aims to reconstruct only a *local* part $\mathscr{L}$ of the entire reconstruction grid. Here, $\mathscr{L}$ is a subset of all N^2 pixels of the entire reconstruction grid, usually ordered in a $N_{\mathscr{L}} \times N_{\mathscr{L}}$ grid as well. Let $\boldsymbol{M}_{\mathscr{L}}$ be a diagonal matrix with a value 1 on the diagonal of row i if pixel i is inside $\mathscr{L}$, and 0 everywhere else. In other words, $\boldsymbol{M}_{\mathscr{L}}$ keeps all pixels of an image that are inside $\mathscr{L}$, and zeros all other pixels. Similarly, we define a matrix $\boldsymbol{M}_{\mathscr{F}}$ that zeros all pixels inside $\mathscr{L}$, and keeps all other pixels. Using these, we can define local operators $\boldsymbol{W}_{\mathscr{L}}$ and $\boldsymbol{W}_{\mathscr{L}}^T$, and outer operators $\boldsymbol{W}_{\mathscr{F}}$ and $\boldsymbol{W}_{\mathscr{F}}^T$:

$$\begin{aligned} \boldsymbol{W}_{\mathscr{L}} &= \boldsymbol{W}\boldsymbol{M}_{\mathscr{L}} \\ \boldsymbol{W}_{\mathscr{L}}^T &= \boldsymbol{M}_{\mathscr{L}}\boldsymbol{W}^T \\ \boldsymbol{W}_{\mathscr{F}} &= \boldsymbol{W}\boldsymbol{M}_{\mathscr{F}} \\ \boldsymbol{W}_{\mathscr{F}}^T &= \boldsymbol{M}_{\mathscr{F}}\boldsymbol{W}^T \end{aligned} \tag{4.1}$$

Since $\boldsymbol{M}_{\mathscr{L}} + \boldsymbol{M}_{\mathscr{F}} = \boldsymbol{I}$ by construction, we have that the sum of $\boldsymbol{W}_{\mathscr{L}}$ and $\boldsymbol{W}_{\mathscr{F}}$ is equal to $\boldsymbol{W}$:

$$\boldsymbol{W} = \boldsymbol{W}_{\mathscr{L}} + \boldsymbol{W}_{\mathscr{F}} \tag{4.2}$$

Note that local forward projections and backprojections can be computed significantly faster than full forward projections and backprojections, since many rows and columns of $\boldsymbol{W}_{\mathscr{L}}$ and $\boldsymbol{W}_{\mathscr{L}}^T$ are zero.

Common reconstruction methods

Using the above definitions, we can write one of the most popular reconstruction methods, the analytical filtered backprojection (FBP) method, as:

$$FBP(\boldsymbol{p}, \boldsymbol{h}) = \boldsymbol{W}^T \boldsymbol{C}_{\boldsymbol{h}} \boldsymbol{p} \tag{4.3}$$

Here, $\boldsymbol{C}_{\boldsymbol{h}}$ is a convolution operator that convolves each 1D array of detector values, taken at a single rotation angle, with the 1D filter $\boldsymbol{h}$ [KS01]. Note that this 1D filter can be different for each rotation angle. Several fixed angle-independent filters are commonly used in practice, such as the Ram-Lak (ramp), Shepp-Logan, and Hann filters [Far+97]. One reason for the popularity of FBP is its computational efficiency: the filtering step can be performed very efficiently in Fourier space, and only one backprojection has to be computed during reconstruction. Another advantage of the filtered backprojection method compared to other methods is that we can calculate its values inside the local part $\mathscr{L}$ by simply exchanging $\boldsymbol{W}^T$ by $\boldsymbol{W}_{\mathscr{L}}^T$ in Eq. (4.3):

$$FBP_{\mathscr{L}}(\boldsymbol{p}, \boldsymbol{h}) = \boldsymbol{W}_{\mathscr{L}}^T \boldsymbol{C}_{\boldsymbol{h}} \boldsymbol{p} \tag{4.4}$$

A different approach to solving the reconstruction problem is the algebraic approach. Here, we form a linear system $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$, and solve for $\boldsymbol{x}$. Most algebraic methods find a solution $\boldsymbol{x}_{alg}$ by minimizing the difference, in some vector norm, between the forward

projection of the solution and the measured projection data. This difference is called the *projection error*. In the case of the ℓ_2-norm, we can write this as:

$$\boldsymbol{x}_{alg} = \underset{\boldsymbol{x}}{\operatorname{argmin}} \|\boldsymbol{p} - \boldsymbol{W}\boldsymbol{x}\|_2^2 \tag{4.5}$$

Since the matrix $\boldsymbol{W}$ is often very large, Eq. (4.5) is usually not solved directly. Instead, an iterative optimization method is typically used to iteratively decrease the projection error. Implicit regularization of the solution can be included by stopping the iteration process early, which is needed because $\boldsymbol{W}$ is usually ill-conditioned and noise is often present in $\boldsymbol{p}$.

Different iterative optimization methods can be used to minimize the projection error, leading to different algebraic methods. The CGLS method, for example, is based on a conjugate gradient method [Bjö96]. Another popular algebraic method is the simultaneous iterative reconstruction technique (SIRT) [KS01]. The SIRT method belongs to the class of Landweber iteration methods [Lan51], and uses a specific Krylov subspace method to minimize the projection error iteratively. A single iteration of the SIRT method can be viewed as a gradient-descent step on the projection error, and can be written as:

$$\boldsymbol{x}_s^{k+1} = S(\boldsymbol{x}_s^k) = \boldsymbol{x}_s^k + \alpha \boldsymbol{W}^T \left(\boldsymbol{p} - \boldsymbol{W}\boldsymbol{x}_s^k\right) \tag{4.6}$$

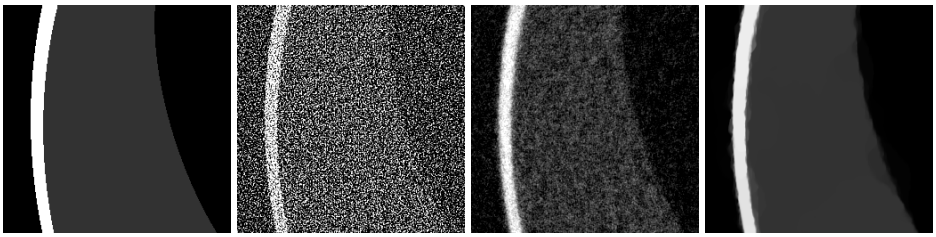
Note that in algebraic methods, we are not able to simply exchange $\boldsymbol{W}$ by $\boldsymbol{W}_{\mathscr{L}}$ to find the reconstruction inside $\mathscr{L}$, since then we would be solving the linear system $\boldsymbol{W}_{\mathscr{L}}\boldsymbol{x} = \boldsymbol{p}$, which will have a completely different solution than $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$ if the scanned object is nonzero outside $\mathscr{L}$.

Regularized iterative methods

A common way of including prior knowledge in algebraic methods is to add additional constraints to the objective function of Eq. (4.5). In this chapter, we distinguish two types of constraints that are commonly used: *domain constraints*, which restrict the domain of possible solutions, and *penalty constraints*, which penalize undesired solutions in the objective function. The resulting regularized iterative methods can be written as:

$$\boldsymbol{x}_{reg} = \underset{\boldsymbol{x}\in D}{\operatorname{argmin}} \left[\|\boldsymbol{p} - \boldsymbol{W}\boldsymbol{x}\|_2^2 + \lambda g(\boldsymbol{x})\right] \tag{4.7}$$

Here, D is a restricted domain for the possible solutions $\boldsymbol{x}$, and $g : \mathbb{R}^{N^2} \to \mathbb{R}$ is a penalty function that penalizes solutions that do not fit with the assumed prior knowledge. The λ term controls how strongly the penalty function is weighted compared to the projection error term. The domain D is used to specify domain constraints, for example when adding a nonnegativity constraint on the values of $\boldsymbol{x}$ by using $D = \{\boldsymbol{x} \in \mathbb{R}^{N^2}; x_i \geq 0,\ i = 1,\ldots,N^2\}$. The cost function $g(\boldsymbol{x})$ is used to specify penalty constraints. For example, if we assume that the scanned object is sparse in some wavelet basis, we can set $g(\boldsymbol{x}) = \|\boldsymbol{B}\boldsymbol{x}\|_1$, where $\boldsymbol{B}$ is the wavelet decomposition operator. Similarly, if we assume that the gradient of the scanned object is sparse, we set $g(\boldsymbol{x}) = \|\nabla\boldsymbol{x}\|_1$ to perform total variation minimization, where ∇ is a discrete



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

Figure 4.2: Zoomed-in reconstructions of the Shepp-Logan head phantom (a), showing the resulting images of three different reconstruction methods: (b) FBP, (c) SIRT, and (d) total variation minimization. The images were reconstructed on a 1024×1024 pixel grid, using projection data acquired with $N_d = 1024$ detectors and $N_\theta = 256$ projection angles, equally distributed in the interval $[0, \pi]$, and additional Poisson noise applied.

gradient operator. Several algorithms exist that are able to find solutions to Eq. (4.7), such as the popular fast iterative shrinkage-thresholding algorithm (FISTA) [BT09a], Chambolle-Pock algorithms [CP11], and adaptive steepest descent projection onto convex sets algorithm (ASD-POCS) [SP08]. A comparison of reconstructions obtained using FBP, SIRT, and total variation minimization from noisy projection data is shown in Fig. 4.2.

Many regularized iterative methods use a scheme that alternates between gradient-descent steps on the projection error $\|\boldsymbol{p} - \boldsymbol{W}\boldsymbol{x}\|_2^2$, steps that minimize the penalty function $g(\boldsymbol{x})$, and steps that enforce the domain constraints D. Since a single iteration of the SIRT method is identical to a single gradient-descent step on the projection error, these regularized iterative methods can be viewed as a combination of SIRT iterations and some additional steps incorporating the prior knowledge. As an example, one can include box constraints on the values of the reconstruction pixels of the form $l \leq x_i \leq r$, which is a domain constraint with $D = \{\boldsymbol{x} \in \mathbb{R}^{N^2}; l \leq x_i \leq r,\ i = 1, \ldots, N^2\}$ by using the following iterations for pixel i of the reconstruction:

$$\boldsymbol{x}_i^{k+1} = \begin{cases} l & : \text{ if } S(\boldsymbol{x}^k)_i < l \\ r & : \text{ if } S(\boldsymbol{x}^k)_i > r \\ S(\boldsymbol{x}^k)_i & : \text{ otherwise} \end{cases} \tag{4.8}$$

An example of using a penalty constraint is the ISTA method [DDD04] for ℓ_1-norm minimization of a representation of the reconstructed image in a wavelet basis. In this case, a single iteration of the method can be written as:

$$\boldsymbol{x}^{k+1} = \boldsymbol{B}^{-1}\mathscr{P}_\lambda(\boldsymbol{B}S(\boldsymbol{x}^k)) \tag{4.9}$$

where $\boldsymbol{B}$ is the wavelet decomposition operator, and $\mathscr{P}_\lambda$ the soft thresholding operator with threshold λ:

$$\mathscr{P}_\lambda(\boldsymbol{y})_i = \begin{cases} \operatorname{sgn}(\boldsymbol{y}_i)(|\boldsymbol{y}_i| - \lambda) & : \text{ if } \boldsymbol{y}_i > \lambda \\ 0 & : \text{ otherwise} \end{cases} \tag{4.10}$$

In this chapter, we propose a method to locally approximate regularized iterative reconstruction methods that are a combination of SIRT iterations and additional steps that incorporate the prior knowledge.

4.3 Method

In this section, we introduce the major contribution of this chapter: a local approximation method for regularized iterative reconstruction methods. We first explain the method introduced in Chapter 3 to approximate the algebraic SIRT method by FBP with a specific geometry-dependent filter, and show how this approach can be used to approximate SIRT locally as well. Afterwards, we extend the approximation to include prior knowledge, improving the reconstruction quality. Finally, we give details on how we implemented the resulting method for the experiments of Section 4.4.

Local approximation of SIRT

Recall that a single iteration of the SIRT method can be written as:

$$\boldsymbol{x}_s^{k+1} = S(\boldsymbol{x}_s^k) = \boldsymbol{x}_s^k + \alpha \boldsymbol{W}^T \left(\boldsymbol{p} - \boldsymbol{W}\boldsymbol{x}_s^k\right) \tag{4.6}$$

Here, $\alpha \in \mathbb{R}$ is a parameter that influences the stability and rate of convergence of the method. In the rest of this chapter, we use $\alpha = (N_\theta N_d)^{-1}$.

To find an approximation method for the SIRT method, we start by rewriting the equation of a single SIRT iteration (Eq. (4.6)) in a matrix format:

$$\boldsymbol{x}_s^{k+1} = (\boldsymbol{I} - \alpha \boldsymbol{W}^T \boldsymbol{W})\boldsymbol{x}_s^k + \alpha \boldsymbol{W}^T \boldsymbol{p} \tag{4.11}$$

This is a recursion equation of the form $\boldsymbol{x}^{k+1} = \boldsymbol{A}\boldsymbol{x}^k + \boldsymbol{b}$, which has the following solution for iteration n:

$$\boldsymbol{x}_s^n = \boldsymbol{A}^n \boldsymbol{x}_s^0 + \alpha \left[\sum_{k=0}^{n-1} \boldsymbol{A}^k\right] \boldsymbol{W}^T \boldsymbol{p} \tag{4.12}$$

where $\boldsymbol{A} = \boldsymbol{I} - \alpha \boldsymbol{W}^T \boldsymbol{W}$. Often, the initial image of the SIRT method is set to the zero image ($\boldsymbol{x}_s^0 = \boldsymbol{0}$), in which case we end up with:

$$\boldsymbol{x}_s^n = \alpha \left[\sum_{k=0}^{n-1} \boldsymbol{A}^k\right] \boldsymbol{W}^T \boldsymbol{p} \tag{4.13}$$

Now, we want to find a method that can approximate Eq. (4.13). In order to find such a method, we look at the FBP method, and note that, in parallel-beam tomography, convolving a sinogram with a filter and backprojecting the result is identical to backprojecting the sinogram and convolving the resulting image with the backprojected filter:

$$FBP(\boldsymbol{p}, \boldsymbol{h}) = \boldsymbol{H}_{h'} \boldsymbol{W}^T \boldsymbol{p} \tag{4.14}$$

Algorithm 4.1 Compute an FBP filter that approximates n iterations of SIRT

Require: $\boldsymbol{W} \in \mathbb{R}^{N_d N_\theta \times N^2}$, $n \in \mathbb{Z}^+$, $\alpha \in \mathbb{R}$
  $\boldsymbol{q}_0 \leftarrow \boldsymbol{0}$
  $\boldsymbol{c} \leftarrow [0, \ldots, 0, 1, 0, \ldots, 0]^T$
  for $k = 1$ **to** n **do**
    $\boldsymbol{q}_k \leftarrow \boldsymbol{q}_{k-1} + \boldsymbol{c}$
    $\boldsymbol{c} \leftarrow \boldsymbol{c} - \alpha \boldsymbol{W}^T \boldsymbol{W} \boldsymbol{c}$
  end for
  $\boldsymbol{u}_n \leftarrow \alpha \boldsymbol{W} \boldsymbol{q}_n$
  return $\boldsymbol{u}_n$

Here, $\boldsymbol{H}_{\boldsymbol{q}}$ is a 2D convolution with filter $\boldsymbol{q}$, and $\boldsymbol{h} = \boldsymbol{W}\boldsymbol{h}'$.

Note the similarities between the rewritten SIRT equation (Eq. (4.13)) and the rewritten FBP equation (Eq. (4.14)), which suggest that we can approximate the SIRT equation by approximating $\sum_{k=0}^{n-1} \boldsymbol{A}^k$ by a 2D convolution operation with filter $\boldsymbol{q}_n$:

$$\boldsymbol{x}_s^n \approx \alpha \boldsymbol{H}_{\boldsymbol{q}_n} \boldsymbol{W}^T \boldsymbol{p} \tag{4.15}$$

A good approximating filter $\boldsymbol{q}_n$ can be found by taking the impulse response of $\sum_{k=0}^{n-1} \boldsymbol{A}^k$:

$$\boldsymbol{q}_n = \sum_{k=0}^{n-1} \boldsymbol{A}^k [0, \ldots, 0, 1, 0, \ldots, 0]^T \tag{4.16}$$

In other words, we apply $\boldsymbol{A}$ to an image $n-1$ times, starting with an image with only the central pixel set to 1, and sum the resulting images to obtain the 2D filter $\boldsymbol{q}_n$.

Since backprojecting a sinogram and convolving the resulting image is the same as convolving the sinogram with the forward projected filter and backprojecting the result, we can write this as:

$$\begin{aligned} \boldsymbol{x}_s^n &\approx \boldsymbol{W}^T \boldsymbol{C}_{\boldsymbol{u}_n} \boldsymbol{p} \\ \boldsymbol{u}_n &= \alpha \boldsymbol{W} \boldsymbol{q}_n \end{aligned} \tag{4.17}$$

Here, $\boldsymbol{C}_{\boldsymbol{h}}$ is the same convolution operator as in Eq. (4.3), and $\boldsymbol{u}_n$ is the corresponding angle-dependent filter. Comparing Eq. (4.3) and Eq. (4.17), we conclude that the SIRT method with n iterations can be approximated by the FBP method with a special filter $\boldsymbol{u}_n$:

$$\boldsymbol{x}_s^n \approx FBP(\boldsymbol{p}, \boldsymbol{u}_n) \tag{4.18}$$

To summarize, the algorithm to compute an approximating filter is given in Algorithm 4.1. For more information on implementing this method, and results for non-local tomographic reconstruction, we refer to Chapter 3.

One advantage of this approximation is that, after calculating the filter, the final reconstruction method is identical to standard FBP. Therefore, we can use the same approach as for FBP to evaluate it locally: simply exchanging $\boldsymbol{W}^T$ with $\boldsymbol{W}_{\mathscr{L}}^T$:

$$\boldsymbol{x}_s^n \approx FBP_{\mathscr{L}}(\boldsymbol{p}, \boldsymbol{u}_n) \tag{4.19}$$

Results for locally approximating SIRT with this approach are given in Section 4.5.

Including regularization

As explained in Section 4.2, many regularized iterative methods include a SIRT step in their iterative equations. In Section 4.3, we showed that we can approximate these SIRT steps locally by using the proposed filter method. However, to locally approximate the complete regularized iterative methods, we need to perform some extra steps. We start by explicitly splitting the reconstruction image at iteration k into two parts: a standard SIRT image $\boldsymbol{x}_s^k$ and a prior-based correction term $\boldsymbol{y}^k$:

$$\boldsymbol{x}^k = \boldsymbol{x}_s^k + \boldsymbol{y}^k \tag{4.20}$$

Furthermore, we rewrite the equation for a single iteration of these methods, such that it consists of a single SIRT step on the previous iteration, and an additional correction term $\boldsymbol{d}$ that incorporates the prior knowledge:

$$\boldsymbol{x}^{k+1} = S(\boldsymbol{x}^k) + \boldsymbol{d}^{k+1} \tag{4.21}$$

Note that it is usually straightforward to rewrite a regularized iterative method that uses SIRT to this form, although one would typically not use such a formulation in practice. For example, SIRT with box constraints (Eq. (4.8)) can be written in this form by taking:

$$\boldsymbol{d}_i^{k+1} = \begin{cases} l - S(\boldsymbol{x}^k)_i & : \text{ if } S(\boldsymbol{x}^k)_i < l \\ r - S(\boldsymbol{x}^k)_i & : \text{ if } S(\boldsymbol{x}^k)_i > r \\ 0 & : \text{ otherwise} \end{cases} \tag{4.22}$$

As another example, iterations of the ISTA method with a wavelet basis (Eq. (4.9)) can be written in the form of Eq. (4.21) by taking:

$$\boldsymbol{d}^{k+1} = \boldsymbol{B}^{-1}\mathscr{P}_\lambda(\boldsymbol{B}S(\boldsymbol{x}^k)) - S(\boldsymbol{x}^k) \tag{4.23}$$

Now, we aim to find a local approximation to Eq. (4.21). If we apply a single SIRT iteration to $\boldsymbol{x}^k$, we get:

$$\begin{aligned} S(\boldsymbol{x}^k) &= \boldsymbol{A}\left(\boldsymbol{x}_s^k + \boldsymbol{y}^k\right) + \alpha \boldsymbol{W}^T \boldsymbol{p} \\ &= \boldsymbol{A}\boldsymbol{x}_s^k + \alpha \boldsymbol{W}^T \boldsymbol{p} + \boldsymbol{A}\boldsymbol{y}^k \\ &= S(\boldsymbol{x}_s^k) + \boldsymbol{A}\boldsymbol{y}^k \end{aligned} \tag{4.24}$$

By combining Eq. (4.21) and Eq. (4.24), we see that:

$$\boldsymbol{x}^{k+1} = S(\boldsymbol{x}_s^k) + \boldsymbol{A}\boldsymbol{y}^k + \boldsymbol{d}^{k+1} \tag{4.25}$$

Using the definition of Eq. (4.20), we can take:

$$\begin{aligned} \boldsymbol{x}_s^{k+1} &= S(\boldsymbol{x}_s^k) \\ \boldsymbol{y}^{k+1} &= \boldsymbol{A}\boldsymbol{y}^k + \boldsymbol{d}^{k+1} \end{aligned} \tag{4.26}$$

In order to locally approximate Eq. (4.21), we need to find local approximations for $\boldsymbol{x}_s^{k+1}$ and $\boldsymbol{y}^{k+1}$.

The iterations of $\boldsymbol{x}_s^{k+1}$ are identical to SIRT iterations, for which we already derived a local approximation in Section 4.3:

$$\boldsymbol{x}_s^{k+1} \approx FBP_{\mathscr{L}}(\boldsymbol{p}, \boldsymbol{u}_{k+1}) \tag{4.27}$$

Furthermore, we can choose to only apply the prior knowledge inside the local part $\mathscr{L}$. In this case, the prior-based correction term $\boldsymbol{d}^{k+1}$ is only nonzero for pixels inside $\mathscr{L}$. To find a local approximation to $\boldsymbol{A}\boldsymbol{y}^k$, we expand $\boldsymbol{A}$, and use the definition of the local and outer projection operations Eq. (4.2):

$$\begin{aligned}\boldsymbol{A}\boldsymbol{y}^k &= \left(\boldsymbol{I} - \alpha \boldsymbol{W}^T \boldsymbol{W}\right)\boldsymbol{y}^k \\ &= \boldsymbol{y}^k - \alpha\left(\boldsymbol{W}_{\mathscr{L}}^T + \boldsymbol{W}_{\mathscr{F}}^T\right)\boldsymbol{W}\boldsymbol{y}^k \\ &= \boldsymbol{y}^k - \alpha \boldsymbol{W}_{\mathscr{L}}^T \boldsymbol{W}\boldsymbol{y}^k - \alpha \boldsymbol{W}_{\mathscr{F}}^T \boldsymbol{W}\boldsymbol{y}^k\end{aligned} \tag{4.28}$$

We approximate Eq. (4.28) locally by simply ignoring the term $\alpha \boldsymbol{W}_{\mathscr{F}}^T \boldsymbol{W}\boldsymbol{y}^k$ which affects the pixels outside $\mathscr{L}$. By ignoring this term, we ignore the effect that the local prior has on the pixels outside $\mathscr{L}$, which can affect the pixels inside $\mathscr{L}$ in later iterations. Since we are, in the end, only interested in the reconstruction inside $\mathscr{L}$, this approximation is usually sufficiently accurate in practice. Another result of this approximation is that $\boldsymbol{y}^k$ will be zero outside $\mathscr{L}$ for any iteration k, and therefore we can substitute $\boldsymbol{W}_{\mathscr{L}}$ for $\boldsymbol{W}$ in the forward projection as well:

$$\boldsymbol{A}\boldsymbol{y}^k \approx \boldsymbol{y}^k - \alpha \boldsymbol{W}_{\mathscr{L}}^T \boldsymbol{W}_{\mathscr{L}} \boldsymbol{y}^k \tag{4.29}$$

To summarize, we have derived a method to approximate a regularized iterative method inside $\mathscr{L}$. Starting with $\boldsymbol{y}^0 = \boldsymbol{0}$, we use the following iterations:

$$\begin{aligned}\boldsymbol{x}_s^{k+1} &= FBP_{\mathscr{L}}(\boldsymbol{p}, \boldsymbol{u}_{k+1}) = \boldsymbol{W}_{\mathscr{L}}^T \boldsymbol{C}_{\boldsymbol{u}_{k+1}} \boldsymbol{p} \\ \boldsymbol{y}^{k+1} &= \boldsymbol{y}^k - \alpha \boldsymbol{W}_{\mathscr{L}}^T \boldsymbol{W}_{\mathscr{L}} \boldsymbol{y}^k + \boldsymbol{d}^{k+1} \\ \boldsymbol{x}^{k+1} &= \boldsymbol{x}_s^{k+1} + \boldsymbol{y}^{k+1}\end{aligned} \tag{4.30}$$

Note that every projection operation in Eq. (4.30) is local, and can therefore be computed efficiently. The needed filters $\boldsymbol{u}_k$ for all iterations can be precomputed for a certain projection geometry with a single run of Algorithm 4.1 by returning a filter for each iteration. The method is based on three approximations to a standard regularized iterative method:

1. Iterations of SIRT are approximated by FBP with specific filters.
2. The prior knowledge is only applied inside $\mathscr{L}$.
3. The effect of the local prior on pixels outside $\mathscr{L}$ is ignored.

Results from Section 4.5 will show that despite these approximations, reconstructions computed by our method are of significantly higher quality than either local FBP or global SIRT reconstructions, and visually similar to global regularized iterative reconstructions. The method is summarized in Algorithm 4.2.

Algorithm 4.2 Compute a local approximation to a regularized iterative method

Require: $\boldsymbol{p} \in \mathbb{R}^{N_d N_\theta}$, $\boldsymbol{W} \in \mathbb{R}^{N_d N_\theta \times N^2}$, $n \in \mathbb{Z}^+$, $\alpha \in \mathbb{R}$
  $\boldsymbol{y}^0 \leftarrow \boldsymbol{0}$
  for $k = 1$ **to** n **do**
    $\boldsymbol{x}_s^k \leftarrow FBP_{\mathscr{L}}(\boldsymbol{p}, \boldsymbol{u}_k)$
    $\boldsymbol{y}^k \leftarrow \boldsymbol{y}^{k-1} - \alpha \boldsymbol{W}_{\mathscr{L}}^T \boldsymbol{W}_{\mathscr{L}} \boldsymbol{y}^{k-1} + \boldsymbol{d}^k$
  end for
  return $\boldsymbol{x}_s^n + \boldsymbol{y}^n$

Algorithm 4.3 Compute a local approximation to FISTA minimizing $\|\nabla \boldsymbol{x}\|_1$

Require: $\boldsymbol{p} \in \mathbb{R}^{N_d N_\theta}$, $\boldsymbol{W} \in \mathbb{R}^{N_d N_\theta \times N^2}$, $n \in \mathbb{Z}^+$, $n_{FGP} \in \mathbb{Z}^+$, $\alpha \in \mathbb{R}$
  $t^0 \leftarrow 1$
  $\boldsymbol{x}_{\mathscr{L}}^0 \leftarrow \boldsymbol{0}$
  $\boldsymbol{x}^0 \leftarrow \boldsymbol{0}$
  for $k = 1$ **to** n **do**
    $\boldsymbol{x}_s \leftarrow FBP_{\mathscr{L}}(\boldsymbol{p}, \boldsymbol{u}_k)$
    $\boldsymbol{q} \leftarrow \boldsymbol{x}_{\mathscr{L}}^{k-1} - \alpha \boldsymbol{W}_{\mathscr{L}}^T \boldsymbol{W}_{\mathscr{L}} \boldsymbol{x}_{\mathscr{L}}^{k-1}$
    $\boldsymbol{x}^k \leftarrow FGP(\boldsymbol{x}_s + \boldsymbol{q}, n_{FGP})$
    $t^k \leftarrow (1 + \sqrt{1 + 4t^{k-1}})/2$
    $\boldsymbol{r} \leftarrow \boldsymbol{x} + (t^{k-1} - 1)\boldsymbol{x}^k / (t^k \boldsymbol{x}^{k-1})$
    $\boldsymbol{x}_{\mathscr{L}}^k \leftarrow \boldsymbol{r} - \boldsymbol{x}_s^k$
  end for
  return $\boldsymbol{x}^n$

The term $\boldsymbol{d}$ in Algorithm 4.2 is the term in which the prior knowledge is exploited, and depends on which regularized iterative method is used. Often, in actual implementations, a different formulation can be used that is more natural to that specific regularized iterative method than the one shown in Algorithm 4.2. As an example, Algorithm 4.3 shows an implementation of the method when using FISTA to minimize the ℓ_1 norm of the gradient of the reconstructed image. Here, we use similar notation to [BT09b], and $FGP(\boldsymbol{x}, n_{FGP})$ refers to the FGP method of [BT09b] with n_{FGP} iterations, applied to the image $\boldsymbol{x}$.

Implementation details

In this section, we will discuss a few details on implementing the proposed method. Specifically, we will discuss how to prevent certain reconstruction artifacts from appearing and how to improve the computation time of the method in repeated applications.

Using some forms of prior knowledge, artifacts can appear in the reconstructed image near the edges of the reconstruction grid. For example, the gradient in a total

variation constraint is often defined differently for pixels on the edge of the reconstruction grid compared to pixels in the interior, which can affect the reconstruction near the edges. For global regularized iterative methods, the interesting features of the reconstructed object are usually situated relatively far from the edge, in which case the artifacts near edges can simply be ignored. In the proposed local method, however, interesting features may be located near or on the edge of the chosen local part. A simple but effective way of reducing the effect of edge artifacts in these cases is to increase the size of the local part slightly, and crop the resulting reconstruction to the chosen local part. In the rest of this chapter, we increase the size of the local part by padding it with $\frac{1}{8}$ of the height/width of the local part on each side.

The reconstruction quality of the filter-based approximation of the SIRT method given in Section 4.3 depends on the discrete implementations of the projection operators, as explained in Chapter 3. Specifically, the method is based on approximating the combined $\boldsymbol{W}^T\boldsymbol{W}$ operator by a shift-invariant convolution operation. The discrete projection operations can be implemented in different ways [XM06], and the accuracy of the approximation depends on the chosen implementation. In practice, most artifacts resulting from the errors in the approximation are found in the low frequencies of the reconstructed image, similar to the artifacts that can occur when discretizing the Ram-Lak filter of the FBP method [KS01, Fig. 3.13]. By using implementations of the projection operators that minimize the approximation error that is made, reconstruction artifacts can be limited, and are typically invisible to a human observer. In this chapter, we use an additional preprocessing step to further reduce these artifacts. Before each reconstruction with the local approximation method, we subtract from the projection data the forward projection of a disc, centered on the rotation axis, with a diameter N and a constant gray value. The gray value is chosen such that the ℓ_2-norm of the zero-frequency components of all projections are minimized after subtraction. By reducing the low-frequency components of the projection data with this procedure, the artifacts resulting from the approximation error are reduced as well. After reconstruction, the same disc is added back to the reconstructed image. In practice, this procedure ensures that artifacts resulting from errors made in approximating SIRT by filtered backprojection are minimal.

As explained in Section 4.3, all projection operations of the proposed method can be computed locally, and are therefore efficient to compute. When the local part is much smaller than the number of detector pixels ($N_{\mathscr{L}} \ll N_d$), however, the convolution operation in $FBP_{\mathscr{L}}$, which scales with N_d instead of $N_{\mathscr{L}}$, can become a significant part of the total computation time. In many cases, however, one will perform repeated applications of the local method, for example when finding optimal parameters for the applied prior knowledge term, or when reconstructing multiple local parts at different locations. In these cases, the convolution of the projection data with the different filters $\boldsymbol{u}_k$ for each iteration can be precomputed once and reused for the different local reconstructions, improving reconstruction time significantly.

4.4 Experiments

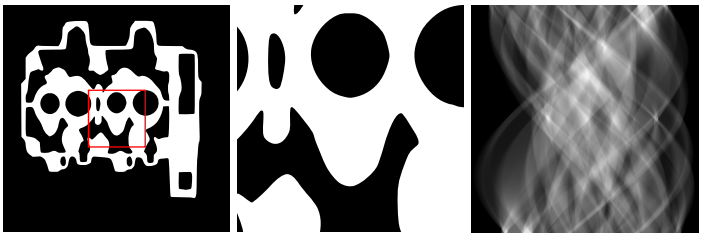
To investigate the properties of the proposed method, we implemented it in Python, version 3.4.3, using the ASTRA toolbox [Aar+15] to perform all tomographic projection operations, which enables the use of optimized GPU-based computations [PBS11]. All experiments were performed on a machine running Fedora Linux 21, with an Intel Xeon E5-2623 processor, 13 GB of memory, and a NVIDIA GeForce GTX TITAN Z GPU using CUDA version 7.0.

We present results for three different forms of prior knowledge about the reconstructed object: one domain constraint and two penalty constraints. For the domain constraint we use box constraints on the pixel values by specifying $D = \{\boldsymbol{x} \in \mathbb{R}^{N^2}; l \leq x_i \leq r,\ i = 1, \ldots, N^2\}$ in the objective function of Eq. (4.7). For the penalty constraints, we use ℓ_1 minimization of the reconstruction in a Haar wavelet basis, i.e. specifying $g(\boldsymbol{x}) = \|\boldsymbol{B}\boldsymbol{x}\|_1$, and ℓ_1 minimization of the gradient of the reconstructed image (TV minimization), i.e. specifying $g(\boldsymbol{x}) = \|\nabla \boldsymbol{x}\|_1$. We use Eq. (4.8) to find solutions in the case of box constraints on the pixel values, and the FISTA method in the case of both ℓ_1 penalty functions. In all cases, we compare the locally approximated reconstructions with global reconstructions of the full object exploiting the same prior knowledge on the full volume, and with the popular analytical FBP method and algebraic SIRT method, which do not explicitly exploit any prior knowledge.

The phantom that is used in most experiments in this chapter is shown in Fig. 4.3. This phantom was chosen because it is suitable for all three forms of prior knowledge that we exploit. It consists of two materials: a background with a value of zero and a foreground with a value of one. Therefore, box constraints can be effectively exploited by setting $l = 0$ and $r = 1$. Since the phantom has a sparse boundary, TV minimization and a Haar wavelet basis can also be used to improve reconstruction quality. In addition to the phantom shown in Fig. 4.3, we also present some results for the Shepp-Logan head phantom, shown in Fig. 4.9a, which has a relatively sparse boundary as well.

For each reconstruction, we report the mean squared error (MSE) of the reconstructions inside the region of interest, compared to a known ground truth image. We also report the structural similarity index (SSIM) [Wan+04] of the reconstructions inside the region of interest compared to the ground truth, which is a metric that is designed to be closer to the human visual system than the mean squared error. For methods where a parameter needs to be chosen, i.e. λ in Eq. (4.7), we perform two reconstructions each time: one with the value that minimizes the *MSE* and one with the value that maximizes the *SSIM*. In each case, we find the optimal parameter value using the Nelder-Mead method [NM65]. Note that the optimal parameter value can depend on the dimensions of the reconstruction grid, and therefore, the optimal values can be different for the global regularized iterative reconstructions compared to the locally approximated reconstructions. For all iterative methods, we use 200 iterations to compute each reconstruction, and we use 100 FGP iterations in the FISTA method for TV minimization [BT09b].

In most experiments, we use a 4096×4096 pixel image of the phantom, and generate projection data for 4096 detector pixels. Afterwards, the projection data is



(a) (b) (c)

Figure 4.3: The phantom used for most experiments in this chapter. In (a), the entire phantom is shown with a red square indicating the local part (b) that is used in most experiments. In (c), the sinogram of the phantom is shown for 1024 detector pixels and 1024 projections equally distributed in $[0, \pi]$.

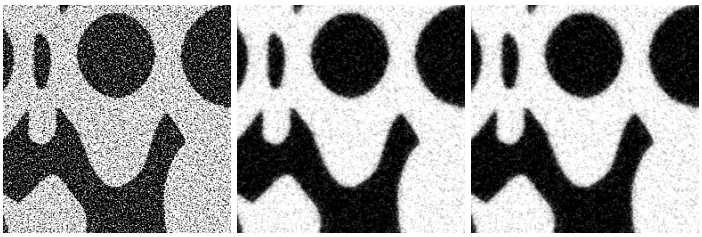
resampled to 1024 detector pixels, and reconstructions are computed on a 1024×1024 pixel grid, or a local part of that grid. These reconstructions are compared to the original 4096×4096 pixel phantom, resampled to a 1024×1024 pixel grid. In most cases, additional Poisson noise is applied to the projection data to simulate experimental conditions. The amount of applied Poisson noise is indicated by a variable I_0, with lower values corresponding to higher amounts of applied noise. Specifically, the noise is applied by first transforming the simulated projections to virtual photon counts, in which the largest photon count out of all detector pixels is set to I_0. For each detector pixel, a new photon count is sampled from a Poisson distribution with the original photon count as the expected value. Finally, the resulting noisy photon counts are transformed back to noisy line integrals of the phantom.

4.5 Results

In this section, we present the results of the experiments that we performed to investigate the properties of the proposed local approximation method, and discuss these results.

Local SIRT approximation

In Fig. 4.4, reconstructions are shown for the local part of the phantom, computed by standard FBP, standard SIRT, and the local approximation of SIRT (Eq. (4.19)). Note that the global SIRT reconstruction and its local approximation are visually very similar. The difference between the computation times is significant, however: the local reconstructions take 28 milliseconds to compute each, while the global SIRT reconstruction takes 2.6 seconds. The *MSE* of the FBP, SIRT, and local approximation are 0.245, 0.016, and 0.016, respectively, and the *SSIM* values are 0.07, 0.25, and 0.27.



(a) (b) (c)

Figure 4.4: Reconstructions of a 256×256 pixel local part of the motor phantom (Fig. 4.3a), using projection data of 1024 detector pixels with $N_\theta = 512$ projection angles, equally distributed in the interval $[0, \pi]$, and with Poisson noise applied. In (a) the local FBP reconstruction is shown, in (b) the global SIRT reconstruction cropped to the local part, and in (c) the locally approximated SIRT reconstruction.

Local regularized iterative approximation

In Fig. 4.5, the mean squared error and structural similarity index are shown as a function of the amount of applied Poisson noise I_0, for standard FBP, standard SIRT, and global and locally approximated reconstructions using various types of prior knowledge. The results show that by exploiting prior knowledge, reconstruction quality can be significantly improved compared to standard FBP and SIRT reconstructions. For this phantom, exploiting total variation minimization yields reconstructions with the lowest *MSE* and highest *SSIM* values. The results also show that for all tested types of prior knowledge, the quality metrics of the locally approximated reconstructions are very close to those of the global regularized iterative reconstructions. For unknown reasons, the quality metrics of the local approximations are slightly better than the global regularized iterative reconstructions. Similar results can be seen in Fig. 4.6, where the quality metrics are shown as a function of the number of projections angles.

The mean squared error and structural similarity index are shown as a function of the size of the local part $\mathscr{L}$ in Fig. 4.7. For all three prior knowledge types, the reconstruction quality of the local approximations is only significantly lower compared to the global regularized iterative methods when the local size is $N_\mathscr{L} = 32$ or smaller, at which point the number of pixels of the local part is less than 0.1% of the number of pixels in the global reconstruction grid. For larger local sizes, the reconstruction quality is almost independent of the local size. These results suggest that, even for reasonably small local parts, the approximations that are made by the proposed local method do not influence the reconstruction quality significantly.

Reconstructed images of a local part with 256×256 pixels are shown in Fig. 4.8, for projection data of 1024 detector pixels and 512 equiangular projections with Poisson noise applied. The images show that the local approximations are visually almost identical to the global regularized iterative reconstructions for all three prior knowledge types. The results also show how the different prior knowledge types can help improve certain image characteristics compared to standard FBP and SIRT reconstructions. In Fig. 4.9, reconstructed images are shown for a smaller local part (128×128 pixels) of the Shepp-Logan head phantom. Similar to the previous results,

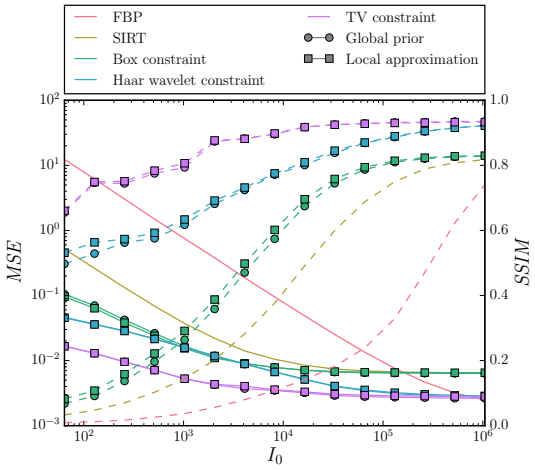


Figure 4.5: Mean squared error (*MSE*, solid lines) and structural similarity index (*SSIM*, dashed lines) of reconstructions of a region (256 × 256 pixels) of the motor phantom (Fig. 4.3a) for various amounts of applied Poisson noise I_0 and types of prior knowledge. The reconstructions are computed using projection data of 1024 detector pixels and 512 projections equally distributed in the interval $[0, \pi]$.

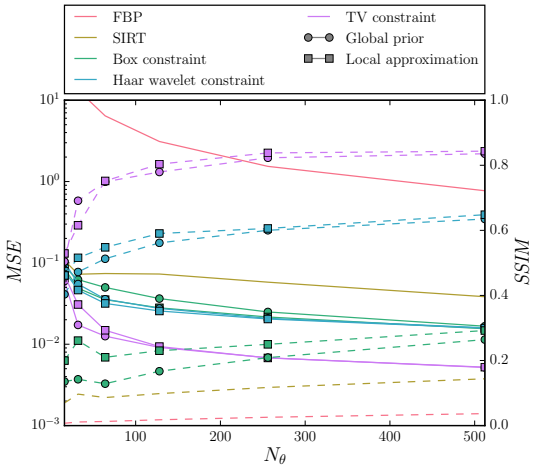


Figure 4.6: Mean squared error (*MSE*, solid lines) and structural similarity index (*SSIM*, dashed lines) of reconstructions of a region (256 × 256 pixels) of the motor phantom (Fig. 4.3a) for various numbers of projection angles N_θ (equally distributed in the interval $[0, \pi]$) and types of prior knowledge. The reconstructions are computed using projection data of 1024 detector pixels, with applied Poisson noise.

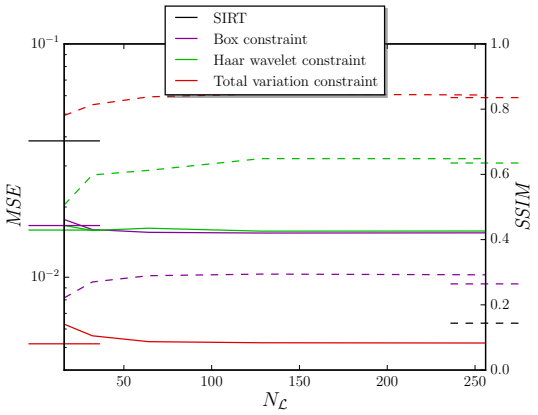


Figure 4.7: Mean squared error (*MSE*, solid lines) and structural similarity index (*SSIM*, dashed lines) of reconstructions of a region (256×256 pixels) of the motor phantom (Fig. 4.3a) for various sizes of the local part $N_{\mathscr{L}}$ and types of prior knowledge. The reconstructions are computed using projection data of 1024 detector pixels and 512 projections equally distributed in the interval $[0, \pi]$, with applied Poisson noise. For $N_{\mathscr{L}} < 256$, multiple local reconstructions are tiled to create a reconstruction of 256×256 pixels, to enable comparison between different local sizes. The partial horizontal lines on each axis indicate the *MSE* and *SSIM* of global SIRT and global regularized iterative reconstructions, cropped to the same 256×256 pixels.

the local approximations are visually almost identical to the global regularized iterative reconstructions.

Computation time

The computation time of the proposed local reconstruction method is shown in Fig. 4.10 as a function of the size of the local part $\mathscr{L}$. Also shown is the computation time of the standard global regularized iterative method. For the local method, computation times are shown both for the first application, as well as for subsequent applications, in which the convolution results of the first application can be reused to decrease the needed computation time (see Section 4.3). For all of types prior knowledge, the local method requires significantly less computation time than the global regularized iterative methods.

If one is only interested in a local part of the object, the local method can be used to compute advanced regularized reconstructions in a few seconds instead of the several minutes it costs to compute the global reconstruction. In cases where the same regularized iterative method is computed multiple times for the same projection data, for example when estimating the λ parameter, the proposed local method requires even less computation time, leading to a significant reduction of processing time in practice. Finally, since each local reconstruction is independent of the other local reconstructions, different local parts can be reconstructed in parallel and combined afterwards to compute a larger part of the scanned object in short time. An example of such a reconstruction is shown in Section 4.5.

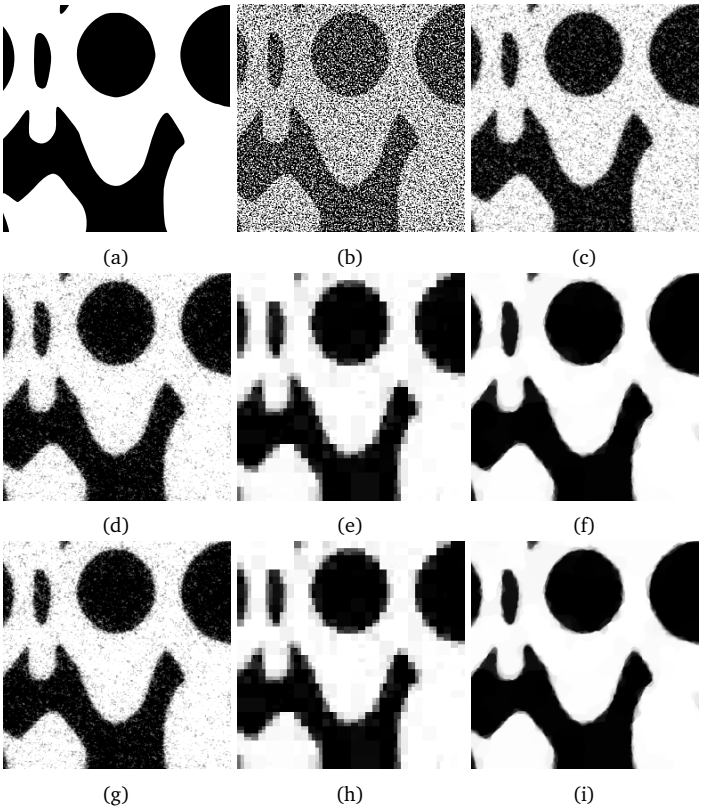


Figure 4.8: Reconstructions of a local part (256×256 pixels) of the motor phantom (a) from projection data of 1024 detector pixels and 512 projections equally distributed in the interval $[0, \pi]$, with Poisson noise applied, using various reconstruction methods: (b) local FBP, (c) global SIRT cropped to local part, (d)-(f) global regularized iterative method cropped to local part, with (d) box constraint, (e) Haar wavelet constraint, and (f) TV constraint, and (g)-(i) the proposed local method, with (g) box constraint, (h) Haar wavelet constraint, and (i) TV constraint. The local reconstructions are shown with a gray-level window in which black corresponds to the minimum value and white to the maximum value of the phantom inside the local part.

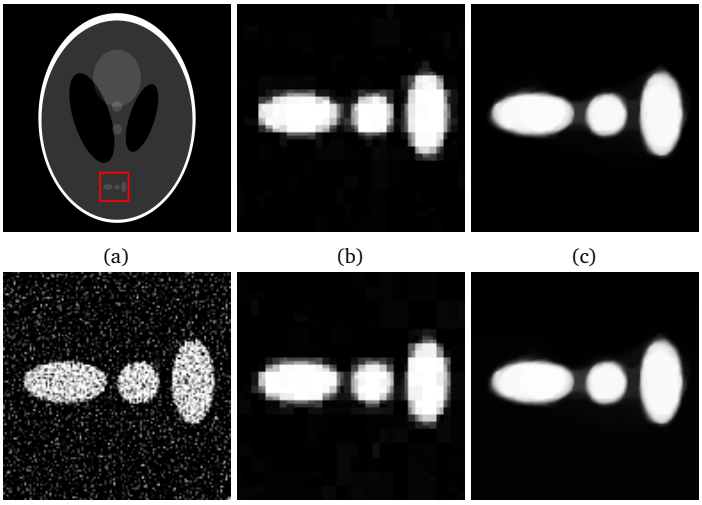


Figure 4.9: Reconstructions of a local part (128 × 128 pixels) of the Shepp-Logan head phantom, indicated by the red square in (a). The reconstructions are computed from projection data of 1024 detector pixels and 512 projections equally distributed in the interval $[0, \pi]$, with Poisson noise applied, using various reconstruction methods: (b)-(c) global regularized iterative method cropped to local part, with (b) Haar wavelet constraint and (c) TV constraint, (d) local FBP, and (e)-(f) the proposed local method, with (e) Haar wavelet constraint and (f) TV constraint. The local reconstructions are shown with a gray-level window in which black corresponds to the minimum value and white to the maximum value of the phantom inside the local part.

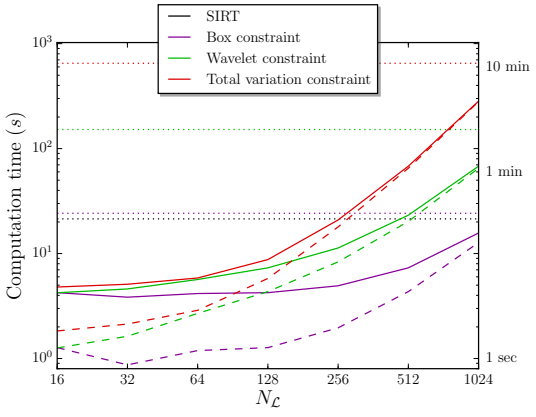
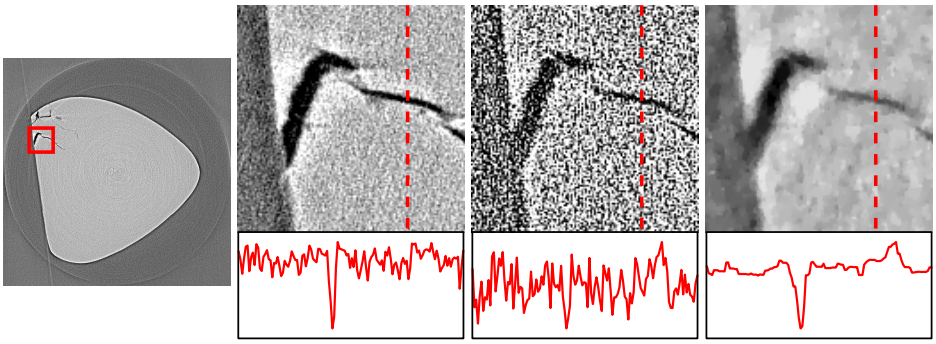


Figure 4.10: Reconstruction time of the global regularized iterative methods (dotted) and the proposed local method for various sizes of the local part $N_{\mathcal{L}}$ and constraint types, using data of 2048 detector pixels and 512 projections. Solid lines show the reconstruction time for a single application of the local method, and dashed lines show the reconstruction time for subsequent applications, where the convolution results of an earlier reconstruction can be reused (see Section 4.3).



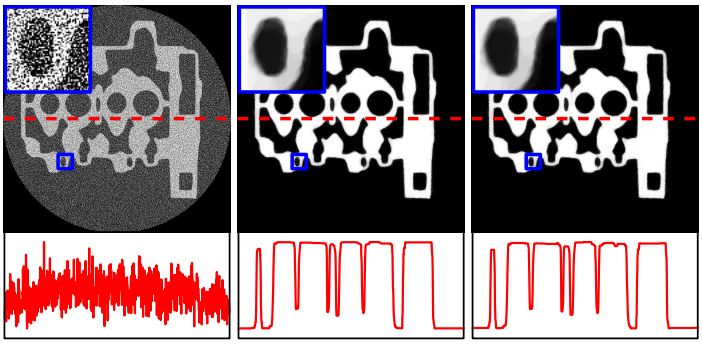
(a) (b) (c) (d)

Figure 4.11: Reconstructions of a local part (128 × 128 pixels) of experimental data of a small fatigue test sample made from Ti alloy VST 55531, acquired with 1200 detector pixels and 1500 projections equally distributed over the interval $[0, \pi]$. In (a) and (b), FBP reconstructions are shown using all 1500 projections, with the local part indicated by a red square in (a). The local FBP reconstruction using only 75 equiangular projections is shown in (c). In (d), a reconstruction is shown for the same 75 projections, using the local reconstruction method presented in this chapter with TV-minimization regularization by the FISTA method. Underneath each local reconstruction, the line profile of the column indicated by the dashed line is shown.

Experimental data

In Fig. 4.11, reconstructed images are shown for a local part of an experimental dataset. The experimental data was acquired for a small fatigue test sample made from Ti alloy VST 55531. The sample was scanned at beamline ID11 of the European Synchrotron Radiation Facility (ESRF), with a parallel, monochromatic (52 keV) synchrotron X-ray beam. The distance between the sample and detector was 40 mm, and 1500 projections were acquired, equally distributed in the interval $[0, \pi]$. The projections were acquired on a high resolution detector system, resulting in projections, after 2×2 binning, with 1200×1200 pixels and an effective pixel size of 0.56 microns.

Results are shown in Fig. 4.11 for a single slice of the reconstructed dataset, computed using FBP and the proposed local method with a TV minimization constraint. For FBP, we show results both when using all 1500 projections that were acquired, and when using only 75 projections, selected by taking every 20^{th} projection of the full dataset. For the local method, we show results for the same limited dataset of 75 projections. The results show that the local method can be successfully applied to an experimental dataset to exploit prior knowledge in the reconstruction. Compared to the FBP reconstruction using 75 projections, the local method is able to more clearly separate the formed crack from the sample itself, which is especially visible in the line profiles. Note that in this type of sample, a user would typically only be interested in the highly localized crack that is forming in the sample, which would make global regularized iterative methods waste significant amounts of computation time on parts of the sample that are not interesting. With the proposed local method, on the other hand, a user would be able to select and reconstruct only those parts of the sample that are interesting.



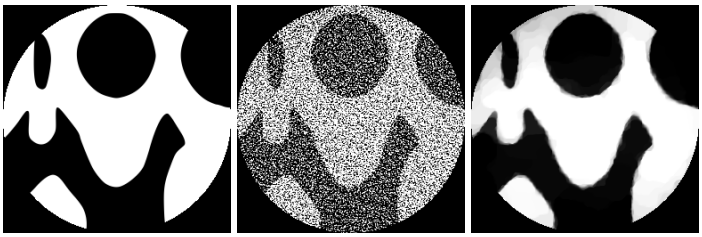
(a) (b) (c)

Figure 4.12: Reconstructions of the motor phantom using projection data of 1024 detectors and $N_\theta = 512$ projection angles, equally distributed in the interval $[0, \pi]$, with Poisson noise applied. The reconstructions are computed with (a) FBP, (b) global TV minimization by the FISTA method, and (c) local 128×128 pixel reconstructions tiled to the complete 1024×1024 pixel grid. The local reconstructions in (c) are computed using the local reconstruction method presented in this chapter with TV-minimization regularization by the FISTA method. Underneath each reconstruction, the line profile of the row indicated by the dashed line is shown. A small region, indicated by the blue square, is shown enlarged in the top-left corner of each reconstruction as well.

Tiling reconstructions

As explained before, one possibility of the proposed local method is to reconstruct different local parts of the image and combine them afterwards into a single reconstruction. One application of this approach would be to compute the different local parts in parallel, which can be parallelized efficiently since each local reconstruction is independent of the others. Another application would be to estimate reconstruction parameters such as the λ term of Eq. (4.7) only in a local part of the reconstruction, which would significantly reduce the time needed to estimate them. Afterwards, the complete image can be reconstructed by combining several local reconstructions using these parameters, which can be computed in parallel as well.

An example of a reconstruction that is computed by tiling several local reconstructions is shown in Fig. 4.12. In this case, we combined 64 local reconstructions of 128×128 pixels each to compute a single 1024×1024 pixel reconstruction, using TV-minimization as the prior knowledge term. The local reconstructions are tiled by simply placing them next to each other on the large reconstruction grid, without any overlapping regions. The results show that there are no visible artifacts from this tiling procedure. Furthermore, the tiled reconstruction is visually almost identical to a reconstruction computed by the global regularized iterative method. This shows that it is possible to significantly reduce the computation time of a global regularized iterative reconstruction method by approximating it with a tiling of local reconstructions computed in parallel.



(a) (b) (c)

Figure 4.13: Reconstructions of the motor phantom (a), using projection data of 1024 detectors truncated to the central 256 detector pixels, using $N_\theta = 512$ projection angles, equally distributed in the interval $[0, \pi]$, and with Poisson noise applied. The reconstructions of FBP (b) and the proposed method with TV-minimization by the FISTA method (c) are shown for the central disc with a width of 256 pixels. Constant padding is used in both reconstructions to reduce truncation artifacts.

Truncated projection data

In some applications of tomography, it is impossible to acquire projections that include the entire scanned object. In these cases, the acquired projection data are *truncated* at the edge of the detector. The resulting reconstruction problem is similar to local reconstruction: again, one is only interested in a subvolume of the entire scanned object. In this case, however, data for the object outside the subvolume is missing. Filtered backprojection is often used to reconstruct truncated data by simply padding the acquired data in order to reduce the artifacts caused by the truncation. Since the local method proposed in this chapter uses FBP to approximate the SIRT method, the same padding approach can be used to apply the method to truncated data. Reconstructions of truncated phantom data are shown in Fig. 4.13, for FBP and the proposed local method. The results show that the local method can be used to exploit prior knowledge in the case of truncated data to improve reconstruction quality.

4.6 Conclusions

In this chapter, we introduced a method to approximate regularized iterative tomographic reconstruction methods inside a region of interest. This method can be used to reduce computation time when one is only interested in the reconstruction inside the region of interest. The method is based on approximating the SIRT steps that are part of many regularized iterative methods by filtered backprojection with specific pre-calculated filters. The result is a reconstruction method in which all projection operations involve only the pixels that are inside the region of interest. The method can also be applied to truncated projection data by similar padding techniques as used for filtered backprojection.

To investigate the properties of the proposed method, we computed reconstructions using various types of prior knowledge about the reconstructed object: box constraints on the pixel values, ℓ_1 minimization of the reconstruction in a wavelet basis, and ℓ_1

minimization of the gradient of the reconstructed image. The results show that the proposed method is able to accurately approximate the reconstructions that would be the result of computing the regularized iterative methods on the full object. Compared to standard reconstruction methods such as FBP and SIRT, the proposed method is able to significantly improve reconstruction quality by exploiting prior knowledge.

One interesting application of the method is to use it to tile reconstructions of small subvolumes to obtain a reconstruction of the complete object. Using the proposed method, the reconstruction of each subvolume is completely independent of the other subvolumes. This enables parallel computation of the complete reconstruction, resulting in a significant reduction of computation time. The results of this chapter show that the reconstruction quality of such a tiling is comparable to the standard global regularized iterative reconstruction. The reduction of computation time might enable the use of more advanced types of prior knowledge that are too computationally expensive to apply globally. Another application is to quickly estimate the parameters of a slow regularized iterative method by estimating them in only a small subvolume.

The filter-based method of Chapter 3 on which the proposed method is based relies on the shift-invariance of the projection operations. Therefore, it is only applicable to parallel-beam tomography in its current form. How to apply a similar method to other acquisition geometries is subject to further research. It may be necessary to use additional approximations to derive filter-based methods in other geometries, in which case exploiting prior knowledge may actually help to reduce artifacts caused by the additional approximations.