DART: A practical reconstruction algorithm for discrete tomography

Kees Joost Batenburg and Jan Sijbers

Abstract In this paper, we present an iterative reconstruction algorithm for discrete tomography, called DART (Discrete Algebraic Reconstruction Technique). DART can be applied if the scanned object is known to consist of only a few different compositions, each corresponding to a constant grey value in the reconstruction. Prior knowledge of the grey values for each of the compositions is exploited to steer the current reconstruction towards a reconstruction that contains only these grey values.

Based on experiments with both simulated CT data and experimental μCT data, it is shown that DART is capable of computing more accurate reconstructions from a small number of projection images, or from a small angular range, than alternative methods. It is also shown that DART can deal effectively with noisy projection data and that the algorithm is robust with respect to errors in the estimation of the grey values.

***Index Terms*—Discrete tomography; image reconstruction; segmentation; prior knowledge**

EDICS categories: COI-TOM, TEC-FOR

I. Introduction

Tomography is an important technique for non-invasive imaging with applications in medicine, industry, and science. It is applicable in scenarios where series of projection images of an object are available, acquired for a range of angles. A reconstruction of the object is subsequently computed from the projection images by a reconstruction algorithm.

A range of reconstruction algorithms are available, which differ in reconstruction accuracy, requirements on the projection geometry, computational load, etc. (see, e.g., [8], [15], [18], [22]). Classical Filtered Backprojection (FBP) techniques are still commonly used. Algebraic reconstruction methods, that are based on modelling the reconstruction problem as a large system of linear equations which is solved by iterative methods, are gradually becoming more common in tomography practice. Such algorithms can potentially yield more accurate reconstructions in some cases, at the expense of increased computation time.

K.J. Batenburg and J. Sijbers are members of IBBT-Vision Lab University of Antwerp, Belgium; K.J. Batenburg is also a researcher at CWI, Amsterdam; e-mail: joost.batenburg@cwi.nl

In many applications of tomography, it makes sense to exploit available prior knowledge of the unknown object. Incorporation of this knowledge in the reconstruction algorithm can potentially result in a reduction of the required number of projections, increased accuracy of the reconstruction, or an improved ability to deal with noisy projection data.

The problem of reconstructing images, or more general signals, from a small number of weighted sums of their values has recently attracted considerable interest in the field of Compressed Sensing [12], [13], [27], [28]. In particular, it was proved that if the image is sparse, it can be reconstructed accurately from a small number of measurements with very high probability, as long as the set of measurements satisfies certain randomization properties [10]. In many images of objects that occur in practice, the image itself is not sparse, yet the boundary of the object is relatively small compared to the total number of pixels. In such cases, sparsity of the gradient image can be exploited by Total Variation Minimization [9], [29].

In this paper, we consider a different type of prior knowledge, where it is assumed that the unknown object consists of a small number (i.e., 2-5) of different materials, each corresponding to a characteristic, approximately constant grey level in the reconstruction. Such prior knowledge is available in a wide range of tomography applications: when performing X-ray tomography of industrial objects, the compositions in these objects (e.g., aluminum, plastic, air) are often known in advance [24], [25]. If a bone is scanned (in-vitro) in a micro-CT scanner, one can sometimes assume that the bone has a single constant density [7]. As a third example we mention the reconstruction of homogeneous nanoparticles by electron tomography [6].

The problem of reconstructing images containing a small set of grey levels from their projections has been studied in the fields of *Discrete Tomography* and *Geometric Tomography*. Geometric tomography deals with the reconstruction of geometric objects from data about its sections, its projections, or both [14]. Images of such objects can be considered as binary images, where the first grey level (i.e., black) corresponds to the exterior of the object and the second grey level (white) corresponds to the interior. Much of the work on geometric tomography is concerned with rather specific objects, such as convex or star-shaped objects. According to [16], [17], the field of discrete tomography deals with the reconstruction of images from a small number of projections, where the set of pixel values is known to have only a few discrete values. The literature on discrete tomography contains some conflicting definitions of the field. Originally, the main focus was on the reconstruction of (typically binary) images for which the *domain* was a discrete set, inspired by applications in crystallography.

The focus of the algorithm described in this paper is somewhat different from both geometric and discrete tomography. Firstly, our approach deals not only with binary images, but also with images that contain three or more grey levels. There is no fixed upper bound on the number of grey levels. Yet, the proposed techniques will only be effective if the number of grey levels is small (i.e., 5 or fewer). Compared to discrete tomography, which focuses on reconstruction from a small number of projections (i.e., 4 or fewer), our approach is more general. If tens or even hundreds of projection images are available, prior knowledge of the grey levels in the reconstruction can still be used effectively to improve the quality of the reconstruction, in particular when the projection data are noisy.

A variety of reconstruction algorithms have been proposed for discrete tomography problems. In [26], a primal-dual subgradient algorithm is presented for reconstructing binary images from a small number of projections. This algorithm is applied to a suitable decomposition of the objective functional, yielding provable convergence to a binary solution. In [5], a similar reconstruction problem is modeled as a series of network flow problems in graphs, that are solved iteratively. Both [19] and [1] consider reconstruction problems that may involve more than two grey levels, employing statistical models based on Gibbs priors for their solution. For all these approaches, the required computation time becomes a major obstacles when dealing with image sizes used in practice.

Recently, a new reconstruction algorithm for discrete tomography, called DART (Discrete Algebraic Reconstruction Technique) was proposed. DART alternates iteratively between "continuous" update steps, where the reconstruction is considered as an array of real-valued unknowns, and discretization steps, which incorporate the prior knowledge of the grey levels in the image.

Application of this algorithm to experimental electron tomography data has already resulted in several important new insights in the properties of nanomaterials, as alternative techniques are not available at this scale [3], [4], [6], [30]. However, a full description of the algorithmic details has been lacking thus far. Also, DART is a heuristic algorithm without guaranteed convergence properties which calls for a thorough experimental validation of algorithm properties. In this paper, we provide a detailed presentation of the DART algorithm and validate this technique by extensive experiments based on simulated projection data, as well as real X-ray μCT data. We investigate its ability to reconstruct images from a small number of projections and from projections acquired along a small angular range, comparing DART with several alternative algorithms. We also present experimental results on the robustness of DART with respect to noise in the projection data and errors in the discrete grey levels used for reconstruction.

The outline of this paper is as follows. In Section II, mathematical notation is introduced to describe the tomographic reconstruction problem and the reconstruction problem for discrete tomography is stated formally. The Simultaneous Algebraic Reconstruction Technique (SART) algorithm for continuous tomography is briefly reviewed, as it is used as a subroutine in our implementation of DART. The DART algorithm is described in Section III. In Section IV, we discuss how this algorithm can be implemented efficiently. Section V presents the set of phantom images used in our simulation experiments and describes the experimental setup. Section VI reports on extensive experiments, comparing DART with three alternative reconstruction algorithms, investigating its robustness with respect to noise and errors in the grey level assumptions, and describing experimental convergence properties. Section VII concludes this paper.

II. Notation and concepts

A. Problem definition

This paper deals with an *algebraic* reconstruction algorithm, where the reconstruction problem is represented by a system of linear equations. Our description is restricted to the reconstruction of two-dimensional images from one-dimensional projections, but can be generalized to higher-dimensional settings in a straightforward manner. The reconstructed image is represented on a rectangular grid of size $n = w \times h$. Projections are measured as sets of detector values for various angles, rotating around the object. We denote the number of projection angles by d and the number of detector values for each projection by k. Hence, the total number of measured detector values is given by $m = dk$. Put $\mathbb{R}_{\geq 0} = \{x \in \mathbb{R} : x \geq 0\}$. Let $\boldsymbol{p} = (p_i) \in \mathbb{R}^m$ denote the measured data elements for all projections, collapsed into a single vector. The projection process in tomography can be modeled as a linear operator $\boldsymbol{W}$ that maps the image $\boldsymbol{x} = (x_i) \in \mathbb{R}^n$ (representing the object) to the vector $\boldsymbol{p}$ of measured data:

$$\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}. \tag{1}$$

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

This leads to the following standard reconstruction problem in tomography:

Problem 1: Let $\boldsymbol{W} \in \mathbb{R}_{\geq 0}^{m \times n}$ be a given projection matrix and $\boldsymbol{p} \in \mathbb{R}^m$ be a vector of measured projection data. Find $\boldsymbol{x} \in \mathbb{R}^n$ such that $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$.

In practice, the projection data often contains noise or other errors, in which case a solution is sought for which $||\boldsymbol{W}\boldsymbol{x} - \boldsymbol{p}||$ is minimal w.r.t. some norm $|| \cdot ||$.

In this paper, we consider the reconstruction of images that consist of only a few different grey levels, which are known a priori. This results in the following reconstruction problem for *Discrete Tomography*:

Problem 2: Let $\boldsymbol{W} \in \mathbb{R}_{\geq 0}^{m \times n}$ be a given projection matrix and $\boldsymbol{p} \in \mathbb{R}^m$ be a vector of measured projection data. Let $\ell > 0$ be the prescribed number of image grey levels and $R = \{\rho_1, \ldots, \rho_\ell\}$ denote the set of grey levels. Find $\boldsymbol{x} \in R^n$ such that $\boldsymbol{W}\boldsymbol{x} = \boldsymbol{p}$.

Note that the set R^n is not convex. As a consequence, many algorithms from convex optimization that can be used to solve the general algebraic reconstruction problem cannot be used directly for discrete tomography.

B. The SART algorithm

The DART algorithm that will be proposed in Section III alternates iteratively between "continuous" update steps, where the reconstruction is considered as an array of real-valued unknowns, and discretization steps, which incorporate the prior knowledge of the grey levels in the image. For the continuous step, a range of algebraic reconstruction methods can be used (e.g., ART, SART, SIRT). For the experiments in this paper, we have implemented a version of DART that uses the SART algorithm as a subroutine. Here, we briefly review the SART algorithm for continuous tomography.

In the SART algorithm [2], the current reconstruction is updated for each projection angle separately. Various ordering schemes can be used for the angle selection. The description given below related to our specific implementation of SART, which uses a randomized scheme.

The projection matrix $\boldsymbol{W}$ and vector $\boldsymbol{p}$ can be decomposed into d blocks of k rows as

$$\boldsymbol{W} = \begin{pmatrix} \boldsymbol{W}^1 \\ \vdots \\ \boldsymbol{W}^d \end{pmatrix}, \quad \boldsymbol{p} = \begin{pmatrix} \boldsymbol{p}^1 \\ \vdots \\ \boldsymbol{p}^d \end{pmatrix} \tag{2}$$

where each block $\boldsymbol{W}^t = (w_{ij}^t)$ represents the projection operator for a single angle and each block $\boldsymbol{p}^t$ represents the corresponding projection data.

For $j = 1, \ldots, n$ and $t = 1, \ldots, d$, put $\gamma_j^t = \sum_{i=1}^k w_{ij}^t$. For $i = 1, \ldots, m$ and $t = 1, \ldots, d$, put $\beta_i^t = \sum_{j=1}^n w_{ij}^t$. Furthermore, let S_d be the set of all permutations of the numbers $1, \ldots, d$ and let σ be a random element of S_d.

The SART algorithm starts with an initial guess $\boldsymbol{x} = \boldsymbol{x}^{(0)}$ and iteratively computes a new estimate $\boldsymbol{x}^{(s)}$ $(s = 1, 2, \ldots)$ from the previous estimate $\boldsymbol{x}^{(s-1)}$ by the update equation

$$x_j^{(s)} = x_j^{(s-1)} + \lambda \frac{1}{\gamma_j^{\sigma(s)}} \sum_{i=1}^{k} \frac{w_{ij}^{\sigma(s)} r_i^{(s)}}{\beta_i^{\sigma(s)}}, \qquad j = 1, \ldots, n \tag{3}$$

where $\boldsymbol{r}^{(s)} = \boldsymbol{p}^{\sigma(s)} - \boldsymbol{W}^{\sigma(s)} \boldsymbol{x}^{(s-1)}$ and λ is a relaxation factor. A single sweep through all projection angles, applying a sequence of d update steps, is referred to as a *SART iteration*.

III. The DART algorithm

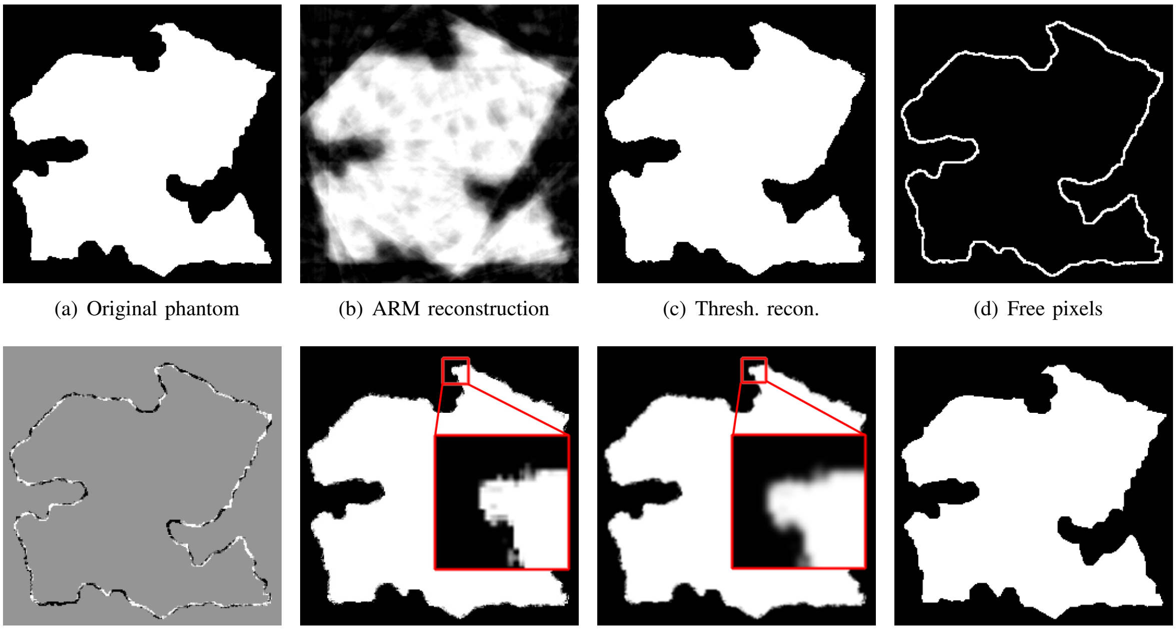
In this section, we describe the DART algorithm. DART utilizes a continuous iterative reconstruction algorithm, such as ART, SART or SIRT, as a subroutine. Within the general description of DART, we refer to the selected continuous method as ARM (Algebraic Reconstruction Method). In the examples and experimental results, SART will be used as the ARM. Before giving a concise description of the operations performed in the DART algorithm, we will first give a brief overview of the algorithmic ideas.

A. Overview of DART

Fig. 1 shows a flow-chart of DART. A continuous reconstruction is computed as a starting point, using the ARM. Subsequently, a number of DART iterations are performed.

Suppose that we want to reconstruct the binary image from Fig. 2(a) from only 12 projections. We assume that the two grey levels (black and white) are known in advance. The continuous SART algorithm is chosen as the ARM. Fig. 2(b) shows the ARM reconstruction after 10 iterations.

From the reconstructed image in Fig. 2(b), it is difficult to decide where the edges of the object are exactly. Yet, the thresholded reconstruction in Fig. 2(c) shows that if we look only at the interior of the object that is not too close to the boundary, the pixels



(a) Original phantom (b) ARM reconstruction (c) Thresh. recon. (d) Free pixels

(e) Free pixels after ARM (f) Total rec. after ARM (g) Smoothed Recon. (h) Final Recon.

Fig. 2. Reconstructing a phantom. The images indicate the various steps of the DART algorithm.

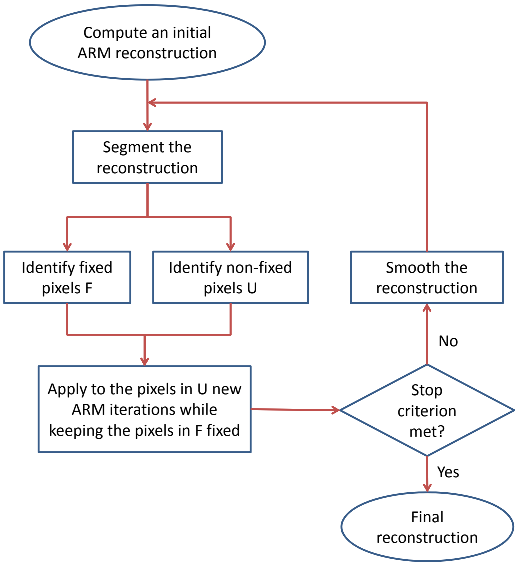


Fig. 1. Flow chart of the DART algorithm

in the thresholded image have the right grey level. The same holds for pixels in the background region that are far away from the object boundary. Next, we locate the boundary region U of the object in the thresholded image, which is defined as the set of all pixels that are adjacent to at least one pixel having a different grey level. The boundary is shown in Fig. 2(d). We now move back to the original grey level ARM reconstruction. All pixels that are not in U are assigned their thresholded value, either black or white. Next, several ARM iterations are performed again, while keeping the pixels that are not in U fixed at the assigned threshold values. That is, the only pixels that are updated by ARM are the pixels in U. In this way, the number of variables in the linear equation system in Eq. (1) is vastly reduced, while the number of equations remains the same. The result of the boundary reconstruction after one ARM iteration is shown in Fig. 2(e), where the gray levels have been scaled to show the range of gray levels present in the boundary pixels. In regions of the boundary where too many white pixels have been fixed, the surrounding boundary pixels have strongly negative pixel values, to compensate. The opposite occurs at parts of the boundary where the extent of the background has been overestimated in the first thresholded ARM reconstruction. In this way, the values of the boundary pixels indicate how the boundary should be adapted in a new estimate of the object. Fig. 2(f) shows the complete reconstruction obtained by merging the boundary with the fixed interior and background.

In the ARM step, each of the boundary pixels is allowed to vary independently, which may result in large local variations of the pixel values. In experiments, we observed that smoothing must be applied to the boundary after the ARM step. Fig. 2(g) shows the result of this smoothing operation. This completes the *DART iteration*.

Subsequently, a thresholded version of the image is computed again, and each of the steps just described is repeated iteratively. As a consequence of the boundary update step, the set of boundary pixels will change between subsequent iterations, allowing for movement of the object boundary.

The final result of this procedure, after four iterations, is shown in Fig. 2(h). It is nearly identical to the original phantom image.

The approach of fixing all pixels that are not on the boundary works well for the reconstruction of single objects that contain no holes. Fig. 3 shows a case where the algorithm fails to compute an accurate reconstruction, again from 12 projections. The phantom in Fig. 3(a) contains a small hole in the interior of the object. Thresholding the continuous reconstruction yields a solid object without any holes. In each iteration of the DART algorithm, the interior part of the object is fixed completely. All changes in the object occur only at the boundary. Therefore, the hole does also not occur in later iterations. The resulting reconstruction is shown in Fig. 3(b).

To allow for the formation of new boundaries that are not connected to the current boundary, a subset of the non-boundary pixels is selected in each iteration that is not fixed, and updated along with the boundary pixels; see Fig. 3(c). Fig. 3(d) shows the DART reconstruction after 10 iterations, where a random subset of 99% of all interior pixels (selected differently in each iteration) has been fixed. Allowing non-boundary pixels to be updated is also crucial for dealing with noisy projection data and grey level errors, as will be demonstrated in Sections VI-D and VI-E. If the boundary is relatively small compared to the image size, the noise from the projection data will be concentrated in the narrow boundary. Selecting a random subset of non-boundary pixels to be updated in each DART-iteration (up to 50%, or even more), largely maintains the capability to reconstruct an image from few projections, while greatly increasing the accuracy in case of noisy data.

B. Algorithm definition

In this section we will formally define the DART algorithm. For a fixed projection geometry, the input of DART consists of the vector $\boldsymbol{p}$ of measured projection data (see Eq. (1)) and the set $R = \{\rho_1, \ldots, \rho_\ell\}$ of grey levels in the reconstructed image. Fig. 4 shows a pseudo-code representation DART.

The first approximate reconstruction $\boldsymbol{x}^0$ is computed using the ARM. After computing the start solution, DART enters an iterative procedure. In each iteration, the following steps are carried out:

1) Segmentation: The current reconstruction is segmented to obtain an image that has only grey levels from the set $R = \{\rho_1, \ldots, \rho_\ell\}$. For the experiments in Section VI, we used a simple global threshold scheme for the segmentation as defined below. Alternative, more advanced segmentation techniques may lead to improved convergence or more accurate reconstruction results in some cases.

Compute a start reconstruction $\boldsymbol{x}^{(0)}$ using ARM;
$t := 0$;
while (stop criterion is not met) **do**
begin
  $t := t + 1$;
  Compute the segmented image $\boldsymbol{s}^{(t)} = r(\boldsymbol{x}^{(t-1)})$;
  Compute the set $B^{(t)}$ of boundary pixels of $\boldsymbol{s}^{(t)}$;
  Compute the set $U^{(t)}$ of free pixels of $\boldsymbol{s}^{(t)}$;
  Compute the set $F^{(t)} = \{1, \ldots, n\} \backslash U^{(t)}$ of fixed pixels;
  Compute the image $\boldsymbol{y}^{(t)}$ from $\boldsymbol{x}^{(t-1)}$ and $\boldsymbol{s}^{(t)}$, setting $y_i^{(t)} := s_i^{(t)}$ if $i \in F^{(t)}$ and $y_i^{(t)} := x_i^{(t-1)}$ otherwise;
  Using $\boldsymbol{y}^{(t)}$ as the start solution, compute the ARM reconstruction $\boldsymbol{x}^{(t)}$, while keeping the pixels in $F^{(t)}$ fixed;
  Apply a smoothing operation to the pixels in $U^{(t)}$;
end

Fig. 4. Basic steps of the algorithm.

Let $\boldsymbol{x}^{(t-1)}$ be the current reconstruction at the start of iteration t of the DART algorithm. A segmented reconstruction $\boldsymbol{s}^{(t)} \in R^n$ is computed from $\boldsymbol{x}^{(t-1)}$, where each pixel $s_i^{(t)}$ is assigned one of the gray values $\rho_1, \ldots, \rho_\ell$ according to a thresholding scheme using thresholds $\tau_1, \ldots, \tau_{\ell-1}$, where

$$\tau_i = \frac{\rho_i + \rho_{i+1}}{2}. \tag{4}$$

Define the *threshold function* $r : \mathbb{R} \to R$ as

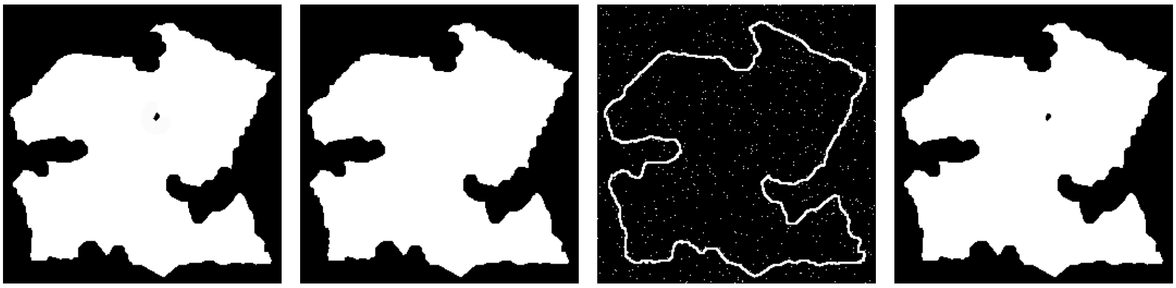
$$r(v) = \begin{cases} \rho_1 & (v < \tau_1) \\ \rho_2 & (\tau_1 \leq v < \tau_2) \\ \quad\vdots & \\ \rho_\ell & (\tau_{\ell-1} \leq v) \end{cases}. \tag{5}$$

As a shorthand notation we also define the threshold function of an *image* $\boldsymbol{x} \in \mathbb{R}^n$:

$$r(\boldsymbol{x}) = \begin{pmatrix} r(x_1) & r(x_2) & \ldots & r(x_n) \end{pmatrix}^T. \tag{6}$$

2) Selection of free pixels: The set $B^{(t)} \subset \{1, \ldots, n\}$ of *boundary pixels* is computed from the segmented reconstruction $\boldsymbol{s}^{(t)}$. We denote the neighborhood of pixel i by $N(i) \subset \{1, \ldots, n\}$. Various connectivity definitions can be used here. We used the 8-connected neighborhood for the experiments in this paper. A pixel $s_i^{(t)}$ is called a boundary pixel if $s_j^{(t)} = s_i^{(t)}$ for all $j \in N(i)$.

The set of *free pixels* $U^{(t)} \subset \{1, \ldots, n\}$ that will be subjected to a DART update, is composed by starting with $U^{(t)} = B^{(t)}$ and augmenting $U^{(t)}$ with non-boundary pixels in a randomized procedure. Let



(a) Original phantom | (b) DART reconstruction, fixing all non-boundary pixels | (c) Free pixels | (d) DART reconstruction, fixing 99% of boundary pixels

Fig. 3. Reconstructing a phantom. The images indicate the various steps of the DART algorithm.

$0 < p \leq 1$ be the *fix probability*. Each element of the non-boundary pixels is included in $U^{(t)}$ with probability $1-p$ independently. Note that the random selection process will be different in the computation for each new DART update. This allows for changes in image areas that are not near any of the boundary pixels.

3) ARM with fixed pixels: Consider the system of linear equations

$$\begin{pmatrix} | & & | \\ \boldsymbol{w_1} & \cdots & \boldsymbol{w_n} \\ | & & | \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \boldsymbol{p}, \tag{7}$$

where $\boldsymbol{w}_i$ denotes the ith column vector of $\boldsymbol{W}$. We now define the operation of *fixing* a variable x_i at value $v_i \in \mathbb{R}$. It transforms the system in Eq. (7) into the new system

$$\begin{pmatrix} | & & | & | & & | \\ \boldsymbol{w}_1 & \cdots & \boldsymbol{w}_{i-1} & \boldsymbol{w}_{i+1} & \cdots & \boldsymbol{w}_n \\ | & & | & | & & | \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_{i-1} \\ x_{i+1} \\ \vdots \\ x_n \end{pmatrix} = \boldsymbol{p} - v_i \boldsymbol{w_i}. \tag{8}$$

The new system has the same number of equations as the original system, whereas the number of variables is decreased by one.

Let $F^{(t)} = \{1, \ldots, n\} \backslash U^{(t)}$ be the set of *fixed pixels*. In each iteration of the DART algorithm, all pixels $i \in F^{(t)}$ are fixed at their values $s_i^{(t)}$, reducing the number of variables from n down to $n - |F^{(t)}|$. The resulting system $\tilde{\boldsymbol{W}}\tilde{\boldsymbol{x}} = \tilde{\boldsymbol{p}}$ is then solved using a constant number of iterations of the ARM. If the fixed pixels have been assigned the "correct" values with respect to the unknown original object, solving the remaining linear system will provide better values for the remaining unfixed pixels, compared to solving the original underdetermined system. When solving underdetermined reconstruction problems, the first few iterations of the DART algorithm will often fix a numerous pixels at incorrect values. As demonstrated in Section VI, the algorithm still demonstrates convergence towards the unknown original object, even if some of the fixed pixels are assigned incorrect values in one or more iterations.

4) Smoothing operation: Reducing the number of variables by fixing a subset of pixels can cause heavy fluctuations in the values of the pixels that are not fixed: the ARM will attempt to match noise in the projection data, as well as errors that result from pixels that are fixed at incorrect values, by adjusting just the values of the free pixels. As a means of regularization, a Gaussian smoothing filter of radius 1 is applied to the boundary pixels after applying the ARM.

5) Termination criterion: As DART is a *heuristic algorithm*, we cannot provide a formal statement of the conditions under which the algorithm will converge. Our experimental results demonstrate that for a variety of relevant images the algorithm converges rapidly to an accurate reconstruction of the original object that was used to obtain the projections; see Section VI-E.

As a termination criterion, either the *total projection error* $E : \mathbb{R}^n \to \mathbb{R}$, defined as

$$E(\boldsymbol{x}) = ||\boldsymbol{W}\boldsymbol{x} - \boldsymbol{p}||_2 \tag{9}$$

can be used, or a fixed number of DART iterations can be performed.

IV. Implementation

We have implemented the DART algorithm in C++. Instead of keeping the projection matrix $\boldsymbol{W}$ in memory as a sparse matrix, all entries are computed when they are needed by the algorithm. In the literature, both ray-driven and pixel-driven (or voxel-driven, in the 3D case) schemes have been proposed for computing the projections of an image [20], [21]. In ray-driven schemes, the projection lines that pass through the image are visited sequentially. For each line, all pixels lying on that line are visited, adding their respective projection to the total projection for that line. In a pixel-driven scheme, the pixels are visited one by one while collecting the total projections for all lines simultaneously. For the DART algorithm, a pixel-driven implementation is superior. Using a pixel-driven implementation, the projection of a subset of the pixels can be computed very fast by just iterating over all pixels in this subset. This operation has to be performed very often in the DART algorithm, when solving the continuous reconstruction problem while a large number of pixels is kept fixed.

In a pixel-driven implementation, the projection $\boldsymbol{q}_i \in \mathbb{R}^m$ of each pixel i is computed as

$$\boldsymbol{q}_i = v_i \boldsymbol{w}_i, \tag{10}$$

where $v_i \in \mathbb{R}$ denotes the value of pixel i and $\boldsymbol{w}_i \in \mathbb{R}^m$ denotes the ith column vector of the projection matrix $\boldsymbol{W}$. The vector $\boldsymbol{w}_i$ typically contains only a constant number of nonzero entries for each projection direction. Therefore, the total contribution of pixel i to the projection data for all angles can be computed in $O(d)$ time. Since we use a pixel-driven implementation, the projections of any set of u pixels in all d directions can be computed in $O(ud)$ time. If the size of F is large compared to the total number of pixels n, this will result in a vast reduction of the running time of the ARM iterations.

V. Experiments

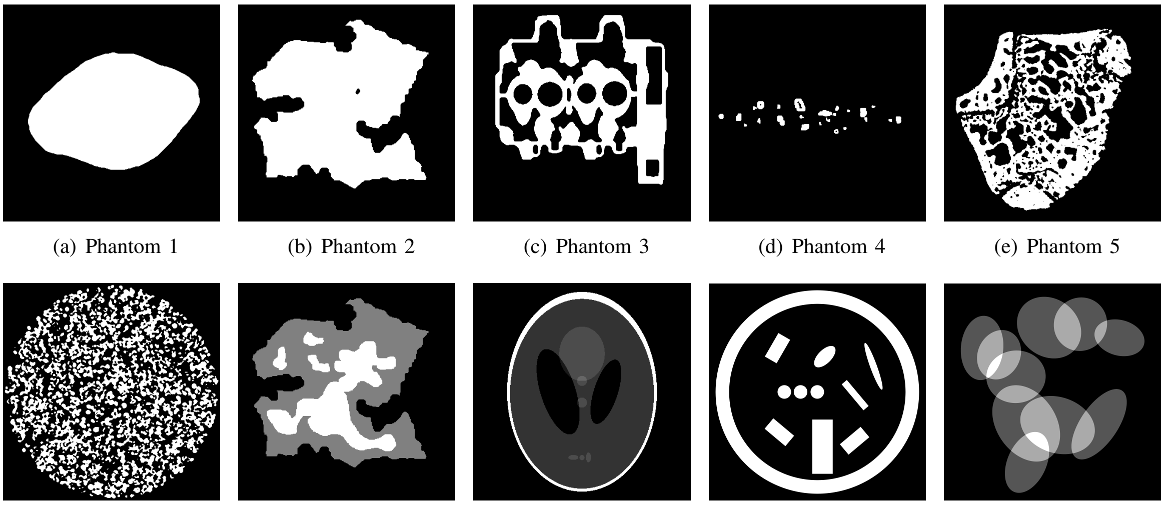
In this section, we describe a series of experiments, for both simulation data and experimental μCT data, that were carried out to evaluate the reconstruction performance of DART and to compare its performance with commonly used reconstruction methods.

A. Phantom images

The simulation experiments were based on ten phantom images, shown in Fig. 5. Phantoms 1-8 are pixel-based phantoms, represented on a pixel grid. The first six phantoms are binary with varying complexity, whereas phantoms 7 and 8 contain three or more grey levels. The last two phantoms, 9 and 10, are *geometric* phantoms that are defined as a superposition of geometric objects and cannot be represented exactly on a pixel grid.

- Phantom 1 represents a very simple, convex shaped object.
- Phantom 2 represents an object with a more complex boundary. Also, the object is not convex and the boundary is fairly complex.
- Phantom 3 represents a cross-section of a cylinder head in a combustion engine. It contains many holes and, as will become apparent from the results, it is more difficult to reconstruct accurately.
- Phantom 4 represents a cross-section of an electron microscopy sample containing homogeneous nanoparticles, taken from [6].
- Phantom 5 was constructed from a micro-CT image of a rat bone, acquired with a SkyScan 1072 cone-beam micro-CT scanner.
- Phantom 6 represents a foam with a very high complexity. It is far more complex than the other phantoms. Hence, it is expected that the number of required projections to accurately reconstruct this phantom will be much larger compared to the other phantoms.
- Phantom 7 represents an object with the same boundary complexity as in Phantom 2 but in which the interior is represented with three grey values.
- Phantom 8 is the well-known Shepp-Logan phantom (6 grey levels), which is commonly used to benchmark reconstruction algorithms for continuous tomography. It is clear that this phantom will not provide a fair comparison between algorithms for continuous and discrete tomography, but we include the results for the sake of completeness.
- Phantom 9 is a binary geometric phantom that consists of a set of ellipses and rectangles of varying sizes, rotated along varying angles. Projections have been computed based on the (non-discretized) geometric shapes of these components (see, e.g., Chapter 3 of [18]). The parameters describing an ellipse are the radius a along the longest axis (where the width and height of the image are both 1), the radius b along the shortest axis, the angle θ between the longest axis and the horizontal axis (in degrees, counterclockwise) and (c_x, c_y), denoting its center of mass. The ellipse parameters are provided in Table I. Similarly, the parameters of the rectangles are the width w and height h, the angle θ between the longest axis and the horizontal axis (counterclockwise), and the center of mass (c_x, c_y). The rectangle parameters are provided in Table II.
- Phantom 10 is a geometric phantom (4 grey levels) that consists of a superposition of randomly placed ellipses of varying sizes, rotated along varying angles. Projections have been computed based on the (non-discretized) geometric shapes of these components. The parameters describing the ellipses are provided in Table III.

The size of phantoms 1-8 is 512×512 pixels, an image size that is also common in practical CT applications. This is also the image size used for



(a) Phantom 1 (b) Phantom 2 (c) Phantom 3 (d) Phantom 4 (e) Phantom 5

(f) Phantom 6 (g) Phantom 7 (h) Phantom 8 (i) Phantom 9 (j) Phantom 10

Fig. 5. Phantom images that were used for the simulation experiments.

a	b	θ	c_x	c_y	value
0.4587	0.4587	0	0.5000	0.5000	1
0.4062	0.4062	0	0.5000	0.5000	-1
0.0628	0.0324	45	0.5337	0.3362	1
0.1145	0.0317	110	0.7575	0.3813	1
0.0313	0.0313	0	0.3477	0.500	1
0.0313	0.0313	0	0.4226	0.500	1
0.0313	0.0313	0	0.5000	0.500	1

TABLE I

PHANTOM 9: PARAMETERS OF THE ELLIPSES

h	w	θ	c_x	c_y	value
0.0625	0.0312	60	0.3147	0.2969	1
0.0782	0.0156	130	0.6740	0.5146	1
0.1250	0.0469	90	0.5253	0.7500	1
0.0625	0.0313	40	0.6728	0.7241	1
0.0625	0.0313	140	0.3264	0.6836	1

TABLE II

PHANTOM 9: PARAMETERS OF THE RECTANGLES

a	b	θ	c_x	c_y	value
0.191406	0.136719	124.318	0.378906	0.644531	1
0.117188	0.082031	164.964	0.808594	0.250000	1
0.148438	0.097656	84.9946	0.175781	0.296875	1
0.148438	0.089844	65.9028	0.382812	0.824219	1
0.187500	0.085937	55.6183	0.714844	0.648438	1
0.125000	0.121094	49.0257	0.628906	0.187500	1
0.175781	0.128906	161.814	0.523438	0.652344	1
0.136719	0.121094	3.97535	0.332031	0.429688	1
0.156250	0.132812	142.252	0.484375	0.203125	1
0.117188	0.097656	64.5288	0.253906	0.328125	1

TABLE III

PHANTOM 10: PARAMETERS OF THE ELLIPSES

the reconstructions. For all phantoms, including the geometric phantoms 9 en 10, the projection for each angle consists of 512 detector values, where the length of the detector is equal to the width (and height) of the image. For phantoms 1-8, this implies that the spacing between consecutive detectors is equal to the pixel size of the phantom. In all simulation experiments reported in this paper, a parallel beam geometry was used. However, the approach can be extended in a straightforward manner to any other acquisition geometry by using a different projection matrix.

B. Quantitative evaluation of reconstruction algorithms

Various simulation experiments were run in which the reconstruction accuracy of DART was compared to other well known reconstruction methods. In particular, a comparison was performed between the following four algorithms:

FBP A standard implementation of FBP was used that performs linear interpolation in the projection domain and uses a Ram-Lak filter.

SART A variant of the SART algorithm as described in Section II-B, performing 200 iterations. This number is large enough to ensure that convergence has been nearly reached. For noiseless projection data, performing so many iterations does not result in degraded reconstruction quality, as is common for high noise levels. We observed that the reconstruction result improves if a positivity constraint is incorporated, setting negative pixel values to zero after each update step. We report on the results obtained by this variant of SART, as it yields better results than without the constraint in all testcases.

TVMin Chambolle's algorithm for Total Variation Minimization (TVMin) was used, as described in [11]. The output of this algorithm depends on several parameters, for which appropriate settings were determined manually. We used $\lambda = 0.02$ (regularization parameter),

$\tau = 0.25$ (descent step) and 20 subiterations. We refer to the original article for details about the method and its parameters.

DART The DART algorithm, using the SART algorithm as described in Section II-B as the ARM. The main loop was repeated 200 times, typically more than enough to obtain convergence. In each iteration, 3 iterations of SART were performed, updating only the pixels in U. For the experiments in Sections VI-A and VI-B, the fix probability was kept constant at $p = 0.85$.

These experiments were based on perfect projection data that was not perturbed by noise or other errors. In particular, the reconstruction accuracy of DART in comparison to alternative approaches was studied

1) as a function of the number of projections, with the projection angles regularly distributed between 0 and 180 degrees.
2) as a function of the angular range of the projections.

In a second series of experiments, the robustness of DART was studied with respect to the assumptions made about the projection data and the object to be reconstructed. Real-world projection data always contains a certain amount of experimental noise. Also, DART assumes the grey levels in the phantoms to be known a priori. In practical applications, these grey levels are often only known approximately. Experiments have been performed to assess the

1) robustness of DART with respect to noise in the projection data.
2) robustness of DART with respect to errors in the input grey levels.

In all experiments, the total number K of pixels from the reconstructed image that differ from the original phantom image was used as a performance metric. We refer to this number as the *pixel error* of a reconstruction.

To compare the results of algorithms that yield greylevel images with the results of DART, the reconstructed images were segmented using the well known Otsu segmentation [23], yielding the required discrete set of grey levels.

C. Experiments for experimental μCT data

A diamond was scanned at 70 kVp in a Scanco μCT 40 X-ray scanner with a circular cone beam geometry. Projections were acquired at 266 angles between 0 and 187 degrees, using a 1024×56 (transaxial$\times$axial) pixel detector. A series of circular cone beam scans was performed at equally spaced axial positions, to cover the length of the diamond. After the scan, the data was rebinned to the parallel beam geometry, yielding a 1024×256 sized sinogram per slice with projection angles distributed equally between 0 and 180 degrees. Although the complete diamond spanned 1221 slices in the axial direction, the diamond extends beyond the field of view of the CCD (so-called *truncation*) from slice 300 and onwards. To allow for reliable reconstruction, only the first 260 slices were used for reconstruction. For this dataset, the reconstruction quality of DART was compared to that of SART for a small number of 15 projections, in which the SART reconstruction based on all 250 projections was used as a reference. A similar comparison between DART and SART was carried out in a limited-angle experiment, based on a subset of 51 projections, with angles distributed equally along an interval of 108 degrees.

VI. Results and discussion

In this section, we present the results of a series of experiments, comparing the reconstructions computed by DART with alternative approaches, and investigating the dependency of the results on the *fix probability*. We also present reconstruction results of a 3D volume, based on experimental μCT data of a raw diamond.

A. Varying the number of projections

We first consider the reconstruction accuracy of DART as a function of the number of projections, where it is assumed that the projection angles are regularly distributed between 0 and 180 degrees.

Fig. 6 shows the pixel error as a function of the number of projections for phantoms 1-8, for the FBP, SART, TVMin and DART algorithms. The results show that DART consistently yields more accurate reconstructions than FBP and SART. The pixel error for DART is only rarely larger than for TVMin, and in many cases it is much lower (e.g., Phantom 3 with 10 projections, Phantom 7 with 8 projections, Phantom 8 with 40 projections).

As an illustration of the results, Fig. 7 shows DART reconstructions of Phantoms 3, 5, and 7 for various projection numbers. Although the reconstruction gradually improves as the number of projections is increased, there appears to be a certain minimum number of projections for each phantom that is required to obtain an almost perfect reconstruction. For Phantom 3, 5, and 7, the number of projections for which the DART reconstruction is nearly perfect, was 10, 20, and 8, respectively. These DART reconstructions are shown in the last column of Fig. 7. As a comparison, the corresponding FBP, SART, and TVMin reconstructions are shown in column 1, 2, and 3 of Fig. 8, respectively. This figure also demonstrates an important feature of DART, and discrete tomography algorithms in general: the resulting reconstruction is already a segmented image that does not require additional segmentation steps.

B. Limited angle problems

In the previous series of experiments, we considered reconstruction problems that can also be solved accurately by continuous reconstruction methods such as FBP and SART, as long as sufficiently many projections are available. This is not the case for *limited angle problems*, which occur frequently in electron tomography and industrial tomography, and also in some medical applications.

In this section, we present reconstruction results of DART from a limited angular range of projections. Fig. 9 shows the pixel error for phantoms 1-8 as a function of the angular range, for the FBP, SART, TVMin and DART algorithms. Here, 180 degrees constitutes a full angular range and projections are sampled at 1 degree intervals. Therefore, the number of projections increases linearly with the angular range.

The results show that, with a few exceptions, DART consistently yields more accurate reconstructions than the three alternative methods. As an illustration of the resulting reconstructions, Fig. 10 shows results for the four methods applied to Phantom 1, 3, and 5, using varying angular ranges. The strong prior knowledge imposed by DART appears to be very powerful for dealing with limited angle problems, as was already demonstrated in several practical electron tomography problems [6].

C. Geometrical objects

The simulation experiments described above were performed with the pixelized phantoms 1-8. In practical situations, the objects scanned are of course not pixellized. In order to test the impact of the discretization of the phantom objects on a regular grid on the performance of DART, additional simulations experiments with continuous phantoms were set up. To this end, continuous phantom studies were performed with FBP, SART, TVmin and DART based on Phantom 9 and 10 for a varying number of projections as well as for a limited angular range of projections.

Fig. 11 shows the continuous pixel error K' as a function of the number of projections used in the reconstruction of the geometric phantoms 9 and 10. This pixel error is computed analytically from the intersection of the continuous phantom image with the rasterized and segmented reconstruction, and corresponds to the total area where the reconstruction and phantom are different (taking the area of a pixel as 1). Fig. 12 shows the continuous pixel error K' as a function of the number of the angular range for FBP, SART, TVMin, and DART, based on an angular step of 1 degree between the projections. Both experiments show that DART performs well compared to FBP, SART and TVMin in terms of the continous pixel error K'.

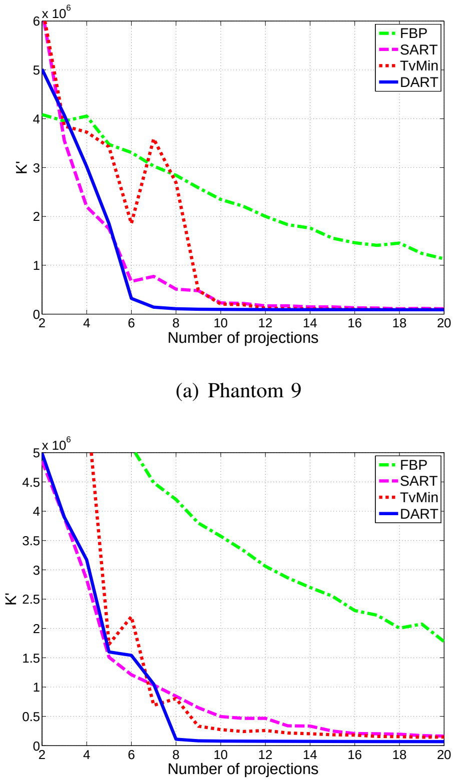
In addition, an experiment was performed based on the pixelized Phantom 3, to evaluate the quality of DART reconstructions as a function of the number of projections, where the original phantom was shifted over half a pixel in both directions before computing the projection data. The shift was performed analytically, representing each pixel as a square of constant grey level. Note that the shifted phantom cannot be represented exactly on the pixel grid used for reconstruction. The reconstructed image was shifted back, again analytically, and then compared to Phantom 3. Fig. 13 shows the pixel error K' of the DART reconstructions, as a function of the number of projections. Note that there is no significant difference between the shifted and the non-shifted reconstruction for a number of projections smaller than 10. From $d = 10$ and onwards, the difference between the shifted and non-shifted reconstruction becomes noticeable at the border area, as can be observed from Fig. 14. Nevertheless, it is clear that DART performs well, even for non-pixelized objects.

So far, we have compared the reconstruction quality for FBP, SART, TVMin and DART, based on perfect, noiseless simulations. Also, we assumed that the set of grey levels to be used in DART is perfectly known. In the next sections, we will turn our attention exclusively to DART and investigate the robustness of DART with respect to noise on the projection data and with respect to errors in the assumptions on the grey levels. The parameter p, that determines the fraction of non-boundary pixels that is kept fixed in the ARM iterations, plays an important role in these cases, and it will be varied in the experiments.

D. Noisy projection data

From the phantom images, CT projections were simulated as follows. First, the Radon transform of the images was computed, resulting in a sinogram for which each data point represents the line integral of attenuation coefficients. Then, (noiseless) CT projection data were generated where a mono-energetic X-ray beam was assumed[1]. The projections were then polluted with Poisson distributed noise where the number of counts per detector element I_0 was varied

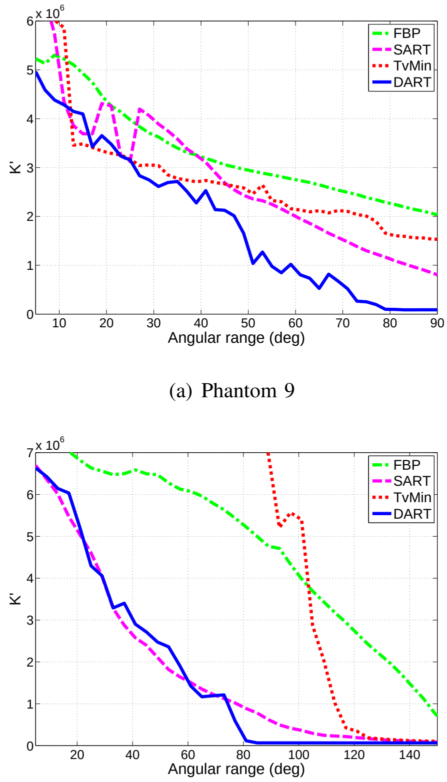
[1]More advanced CT simulation experiments, for example, taking into account scatter and beam-hardening, could as well have been performed, but would, to our view, unnecessarily complicate the discussion of the experimental results.



(a) Phantom 9

(b) Phantom 10

Fig. 11. The pixel error K' as a function of the number of projections used in the reconstruction using equally distributed projection angles for FBP, SART, TVMin and DART.



(a) Phantom 9

(b) Phantom 10

Fig. 12. Limited angle experiments for the geometric objects phantoms: pixel error K' as a function of the projections' angular range for FBP, SART, TVMin, and DART.

from $5 \times 10^3 - 6 \times 10^4$. Next, the noisy sinogram of the attenuation coefficients was obtained by dividing the CT projection data by the maximum intensity and computing the negative logarithm. In this way, simulated projection images were obtained for varying signal-to-noise ratios. Finally, the simulated, noisy CT images were reconstructed.

Fig. 15 shows the pixel error K as a function of the number of counts for various values of the fix probability p, for phantoms 1-8. From that figure, it can be concluded that for low SNR (low number of counts) the pixel error will in general be smaller if p is small, e.g. $p = 0.5$. For high SNR (high number of counts), choosing a high value of p (e.g. $p = 0.99$) yields more accurate reconstructions, but still p must be less than 1 to obtain optimal results for some of the phantoms, due to the inability to create new boundaries if p is set to 1. The observation that for high noise levels a low fix probability yields the best results can be explained by the fact that during the ARM iterations, all noise will be distributed between the free pixels. If there are too few free pixels, the value of these pixels will be determined mainly by the noise, resulting in inferior reconstructions.

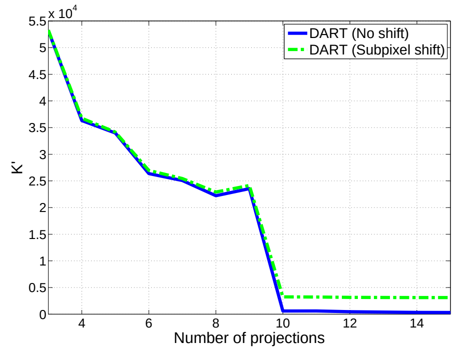
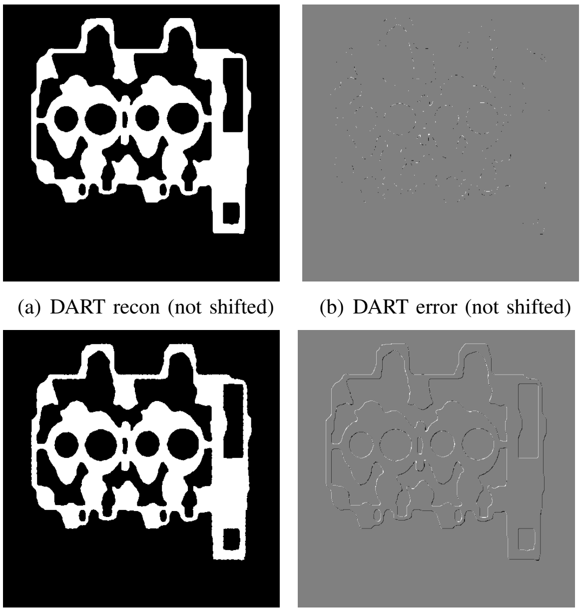


Fig. 13. Pixel error K' as a function of the number of projections for DART reconstructions of Phantom 3 and for DART reconstructions of Phantom 3 shifted by 0.5 pixels in both directions.

E. Prior knowledge on the grey levels

DART requires prior knowledge of the grey levels to be used in the reconstruction. In practical applications, these grey levels are often only known approximately. Therefore, experiments have been performed to assess the robustness of DART with respect to errors in the grey levels used for the reconstruction.

Fig. 16(a) shows the pixel error K of the DART



(a) DART recon (not shifted) (b) DART error (not shifted)

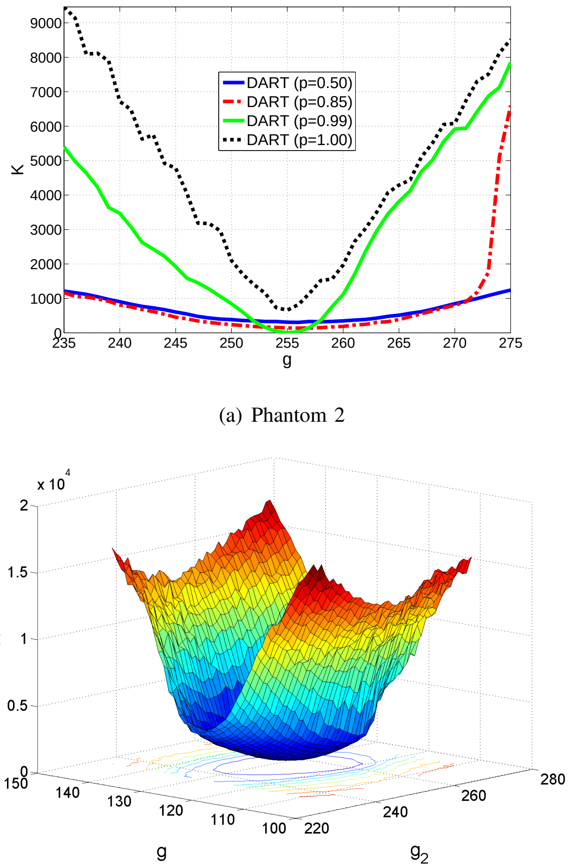
(c) DART recon (shifted) (d) DART error (shifted)

Fig. 14. DART reconstructions of Phantom 3 from 10 projections; (a) reconstruction based on non-shifted phantom; (b) difference between phantom and non-shifted reconstruction; (c) reconstruction based on shifted phantom; (d) difference between phantom and shifted reconstruction

reconstruction as a function of the assumed grey value g of the object for Phantom 2. If the assumed grey level g is over- or underestimated, the projection error is redistributed over the set of free pixels. Clearly, the smaller the number of free pixels is, the higher the update contribution per pixel will be, which will result in a large over- or undershoot of the updated pixel. If the under- or overshoot is large enough to cross the threshold used in the DART segmentation step, the DART reconstruction will be affected at that position. This can visually be observed in Fig. 17 where the DART reconstructions are shown for $g = 263$, 271, and 275 (the true grey value of the phantom image was $g_0 = 255$). Fig. 17(a-c) show the DART error images for $p = 0.99$. These figures show that, with increasing offset $|g - g_0|$ from the true grey level g_0, the number of incorrectly reconstructed pixels K at the border as well as in the interior part steadily and significantly increases.

However, if the fix probability is lowered ($p = 0.5$; $p = 0.85$), the dependency of K as a function of K decreases. Note that, for $p = 0.5$ or $p = 0.85$, less than 0.5% of the pixels is misclassified, even if the offset of the assumed grey value from the true grey value deviates up to 10% from the true grey value. This is mainly because the interior pixels are not affected as long as the smoothing and thresholding DART step results in correctly classified pixels. The classification of the border pixels, are less affected by the smoothing step. Once the smoothing is insufficient to classify even the interior pixels correctly, bumps appear in the interior area and a sudden increase of K is noticed (e.g. at $g = 272$ for $p = 0.85$). This classification behavior is visualized in Fig. 17(d-f), where the DART error images are shown for $p = 0.85$.

Hence, for appropriate values of the fix probability p, DART was observed to be robust with respect to the prior knowledge on the true grey level of the object. Similar experiments where run for objects with more than one grey level, as in Phantom 7. In Fig. 16(b), K is shown as a function of g_1 and g_2, which are the assumed grey levels of Phantom 7 (the true values where 127 and 255, respectively). The 3D plot also indicates that DART is, within reasonable range, robust against errors in the prior knowledge on the true grey levels.



(a) Phantom 2

(b) Phantom 7 ($p = 0.85$)

Fig. 16. The pixel error K as a function of greylevel(s) g that was/were used as prior info for DART.

F. Convergence

A relevant question about any iterative scheme is its convergence behavior and computational stability, since it not only affects the reconstruction time but often the quality of the reconstructed image as well.

For Phantom 2, 3, 5 and 7, the total projection error E as well as the total pixel error K was computed as a function of the number of iterations for fixed probability levels of 0.50, 0.85, 0.99, and 1.00, based on noiseless projection data. Fig. 18(a), 18(b), 18(c), and 18(d) show the convergence rate of the total projection error E for Phantom 2, 3, 5, and 7, respectively. The number of projections used was $d = 6, 10, 50$ and 10, respectively. From Fig. 18, it can be observed that DART converges in a smooth way, although convergence to a solution that satisfies the projection data cannot be guaranteed. From the Fig. 18, it is clear that the *fix probability* p plays an important role in the convergence behavior of DART. For all experiments, setting p close to (but not equal to) 1.0, resulted in the highest convergence rate. Recall that fixing all non-boundary pixels (i.e., $p = 1.0$) would prevent the creation of holes in the object during the iterations. Hence, tiny holes, if missed in the segmentation step of the first iteration, such as in Phantom 2, would never be found, resulting in a relatively large projection error for $p = 1.0$ after convergence (see for example Fig. 18(a)). On the other hand, the smaller the fix probability, the larger the number of pixels is over which the projection error is redistributed during the ARM operation and the smaller the probability that a pixel is changed after thresholding, resulting in a slow convergence.

Fig. 19(a), 19(a), 19(b), and 19(d) show the convergence rate of the phantom error K (i.e., the number of pixel errors in the reconstruction) for Phantom 2, 3, 5, and 7, respectively. All figures show a monotonically decreasing pixel error as a function of the number of iterations.

The total number of iterations required for convergence is significantly larger than for classical iterative reconstruction algorithms, such as SART, where often just two iterations are used in practice. However, the fact that DART can reduce the *number of required projections* significantly, as well as the fact that only a subset of the pixels is updated by the ARM, will result in faster individual iterations. As actual *reconstruction times* are highly implementation dependent, we merely give an indication of the running times: for our experiments based on phantoms of size 512×512, the reconstruction time on a single modern CPU core varied between 10 s (Phantom 1, 5 projections, $p = 0.99$) and 20 minutes (Phantom 6, 50 projections, $p = 0.50$).

The experiments in Sections VI-A and VI-B demonstrate that DART converges to an accurate reconstruction of the original phantom for a broad range of phantoms, provided that a minimal, but sufficient number of projections are available. Yet, there is no absolute guarantee that the reconstruction computed by DART will accurately represent the original object, or even that its projections correspond closely to the original projection data. A compromise between the attractive features of DART and favorable formal convergence properties can be found by applying a postprocessing step to DART. Applying an algorithm for continuous tomography that *does* guarantee convergence in the sense of minimal total projection error, such as SIRT, while using the DART reconstruction as the initial reconstruction, will result in a grey level reconstruction that may not be entirely discrete, but is likely to be close to the DART reconstruction.

G. Experimental data

Five reconstructions have been computed for the experimental μCT diamond dataset described in Section V-C:

- **SART-250.** A SART reconstruction from 250 projections, using 4 iterations over all 250 angles.
- **SART-15.** A SART reconstruction from 15 projections, using 35 iterations over all 15 angles. The angles were selected by approximating constant angular steps between 0 and 180 degrees, each time choosing the nearest available projection angle.
- **DART-15.** A DART reconstruction from the same 15 projections as the SART-15 reconstruction, using $p = 0.60$ and 20 DART iterations. The grey level for the interior of the diamond was determined from the SART-250 reconstruction.
- **SART-A.** A SART reconstruction from limited-angle projection data based on 51 projections with angles distributed equally along an interval of 108 degrees, using 10 iterations over all 51 angles.
- **DART-A.** A DART reconstruction from the same 51 projections as the SART-A reconstruction, using $p = 0.60$ and 20 DART iterations.

Fig. 20(a) shows a 3D surface rendering of a clipped section from the reconstructed volume, based on the SART-250 reconstruction. As it is not obvious to assess reconstruction quality based on the surface rendering, we opted for an alternative visualization, based on three orthogonal slices through the reconstruction. For the five reconstructions, Fig. 20(b-f) each show three partial orthogonal slices through the reconstructed volume in a 3D frame. The partial cross-sections are more suitable for visual comparison with the SART-250 reconstruction.

The results show that although the DART-15 and DART-A reconstructions are not perfect, they approximate the SART-250 reconstruction quite well, and much better than the SART reconstructions for the corresponding subsets of projections. We expect that the accuracy of the presented DART reconstructions is mainly limited by beam hardening effects in the projection data, which could in principle be compensated for to further improve reconstruction quality. beam hardening effects,

VII. Conclusions

In this paper, we have presented the DART algorithm, which can be used for tomographic reconstruction if the scanned object is known to consist of only a few different compositions, each corresponding to a constant grey value in the reconstruction. DART has already been applied successfully to a range of experimental datasets, but a full description of the algorithmic details as provided in this paper has been lacking thus far. As DART is a heuristic algorithm, we have presented a thorough experimental validation of algorithm properties, comparing the resulting reconstruction accuracy to several alternative methods, and investigating the robustness of DART with respect to noise and grey level errors. The results show that DART yields more accurate reconstructions than the alternative methods in most of the experiments. Robustness is largely determined by the fix probability, that can be set according to the specific properties of a reconstruction problem at hand. Lowering the fix probability parameter results in an algorithm that is robust with respect to noise and errors in the set of grey levels used in the reconstruction. Various steps in the presented algorithm, such as the segmentation step and determination of the set of free pixels, can potentially be improved upon, which we will investigate in future research.

Acknowledgements

The authors are grateful to Diamcad NV for providing the experimental μCT diamond dataset.

References

[1] A. Alpers, H. F. Poulsen, E. Knudsen, and G. T. Herman, "A discrete tomography algorithm for improving the quality of 3DXRD grain maps," *J. Appl. Crystall.*, vol. 39, pp. 582–588, 2006.

[2] A. H. Andersen and A. C. Kak, "Simultaneous Algebraic Reconstruction Technique (SART): a superior implementation of the ART algorithm," *Ultrason. Img.*, vol. 6, pp. 81–94, 1984.

[3] S. Bals, K. J. Batenburg, D. Liang, O. Lebedev, G. Van Tendeloo, A. Aerts, J. A. Martens, and C. E. A. Kirschhock, "Quantitative three-dimensional modeling of Zeotile through discrete electron tomography," *J. Am. Chem. Soc.*, vol. 131, no. 13, pp. 4769–4773, 2009.

[4] S. Bals, K. J. Batenburg, J. Verbeeck, J. Sijbers, and G. Van Tendeloo, "Quantitative 3D reconstruction of catalyst particles for bamboo-like carbon-nanotubes," *Nano Letters*, vol. 7, no. 12, pp. 3669–3674, 2007.

[5] K. J. Batenburg, "A network flow algorithm for reconstructing binary images from continuous X-rays," *J. Math. Im. Vision*, vol. 30, no. 3, pp. 231–248, 2008.

[6] K. J. Batenburg, S. Bals, J. Sijbers, C. Kübel, P. A. Midgley, J. C. Hernandez, U. Kaiser, E. R. Encina, E. A. Coronado, and G. Van Tendeloo, "3D imaging of nanomaterials by discrete tomography," *Ultramicroscopy*, vol. 109, no. 6, pp. 730–740, 2009.

[7] K. J. Batenburg and J. Sijbers, "Discrete tomography from micro-CT data: application to the mouse trabecular bone structure," in *Proc. of SPIE Medical Imaging*, vol. 6142, 2006, pp. 1325–1335.

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

[9] E. J. Candès and J. Romberg, "Practical signal recovery from random projections," in *Wavelet Applications in Signal and Image Processing XI, Proc. SPIE*, vol. 5914, 2004.

[10] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information." *IEEE Trans. Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[11] A. Chambolle, "An algorithm for total variation minimization and applications," *J. Math. Im. Vision*, vol. 20, pp. 89–97, 2004.

[12] G.-H. Chen, J. Tang, and S. Leng, "Prior image constrained compressed sensing (PICCS): A method to accurately reconstruct dynamic CT images from highly undersampled projection data sets," *Medical Physics*, vol. 35, no. 2, pp. 660–663, 2008.

[13] D. Donoho, "Compressed sensing," *IEEE Trans. Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[14] R. J. Gardner, *Geometric Tomography, 2nd edition*. Cambridge University Press, 2006.

[15] G. T. Herman, *Fundamentals of Computerized Tomography*. Springer, 2009.

[16] G. T. Herman and A. Kuba, Eds., *Discrete Tomography: Foundations, Algorithms and Applications*. Birkhäuser, Boston, 1999.

[17] ——, *Advances in Discrete Tomography and Its Applications*. Birkhäuser, Boston, 2007.

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

[19] H. Y. Liao and G. T. Herman, "A coordinate ascent approach to tomographic reconstruction of label images from a few projections," *Discr. Appl. Math.*, vol. 151, pp. 184–197, 2005.

[20] S. Matej and R. M. Lewitt, "Practical considerations for 3-D image reconstruction using spherically symmetric volume elements," *IEEE Trans. Med. Imaging*, vol. 15, no. 1, pp. 68–78, 1996.

[21] K. Mueller, R. Yagel, and J. J. Wheller, "Fast implementations of algebraic methods for the 3D reconstruction from cone-beam data," *IEEE Trans. Med. Imaging*, vol. 18, no. 6, pp. 538–547, 1999.

[22] F. Natterer, *The Mathematics of Computerized Tomography*. SIAM, 2001.

[23] N. Otsu, "A threshold selection method from gray level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. 9, pp. 62–66, 1979.

[24] S.-J. Peng and Z.-F. Wu, "Research on the CT image reconstruction of steel tube section from few projections," *NDT & E Interational*, 2009, doi:10.1016/j.ndteint.2009.01.014.

[25] B. Schillinger, "Proposed combination of CAD data and discrete tomography for the detection of coking and lubricants in turbine blades or engines," *Electron. Notes Discrete Math.*, vol. 20, pp. 493–499, 2005.

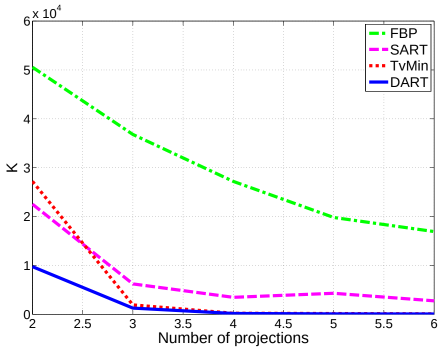
[26] T. Schüle, C. Schnörr, S. Weber, and J. Hornegger, "Discrete tomography by convex-concave regularization and D.C. programming," *Discr. Appl. Math*, vol. 151, pp. 229–243, 2005.

[27] E. Y. Sidky, M. A. Anastasio, and X. Pan, "Image reconstruction exploiting object sparsity in boundary-enhanced X-ray phase-contrast tomography," *Optics Express*, vol. 18, no. 10, pp. 10 404–10 422, 2010.

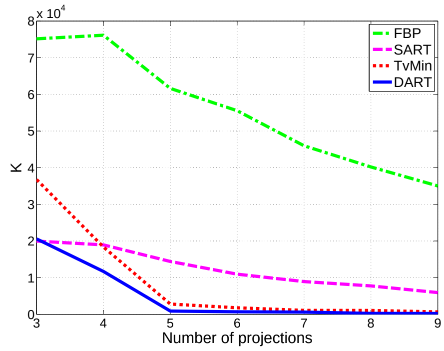
[28] E. Y. Sidky, C. M. Kao, and X. Pan, "Accurate image reconstruction from few-views and limited-angle data in divergent-beam CT," *J. X-ray Sci. Tech.*, vol. 14, pp. 119–139, 2006.

[29] E. Y. Sidky and X. Pan, "Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization," *Phys. Med. Biol.*, vol. 53, pp. 4777–4807, 2008.

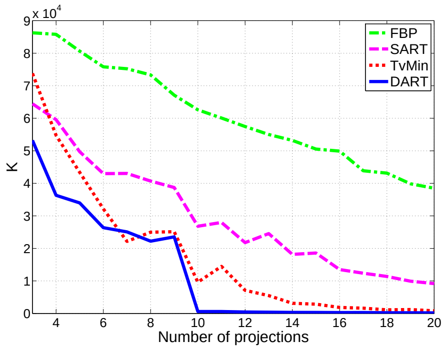
[30] S. Turner, S. M. F. Tavernier, G. Huyberechts, E. Biermans, S. Bals, K. J. Batenburg, and G. Van Tendeloo, "Assisted spray pyrolysis production and characterisation of ZnO nanoparticles with narrow size distribution," *J. Nanoparticle Research*, vol. 12, no. 2, pp. 615–622, 2009.



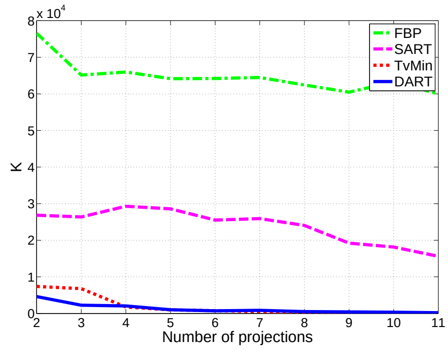
(a) Phantom 1



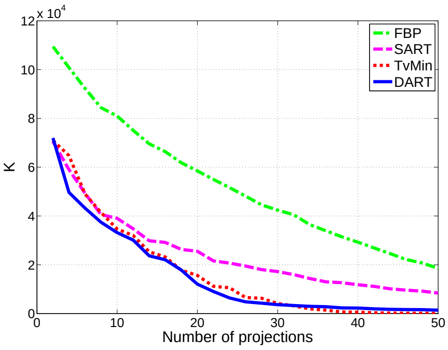
(b) Phantom 2



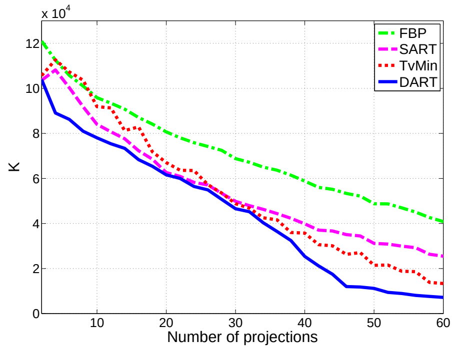
(c) Phantom 3



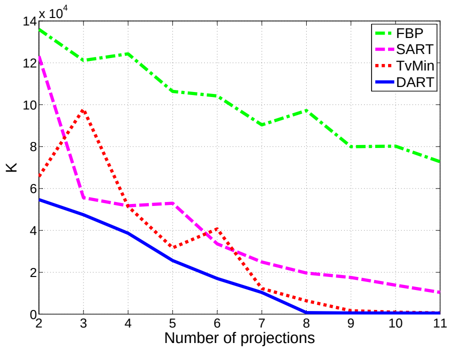
(d) Phantom 4



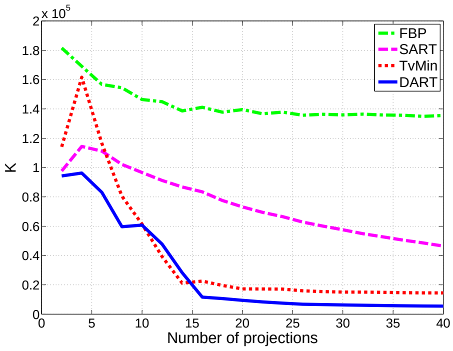
(e) Phantom 5



(f) Phantom 6

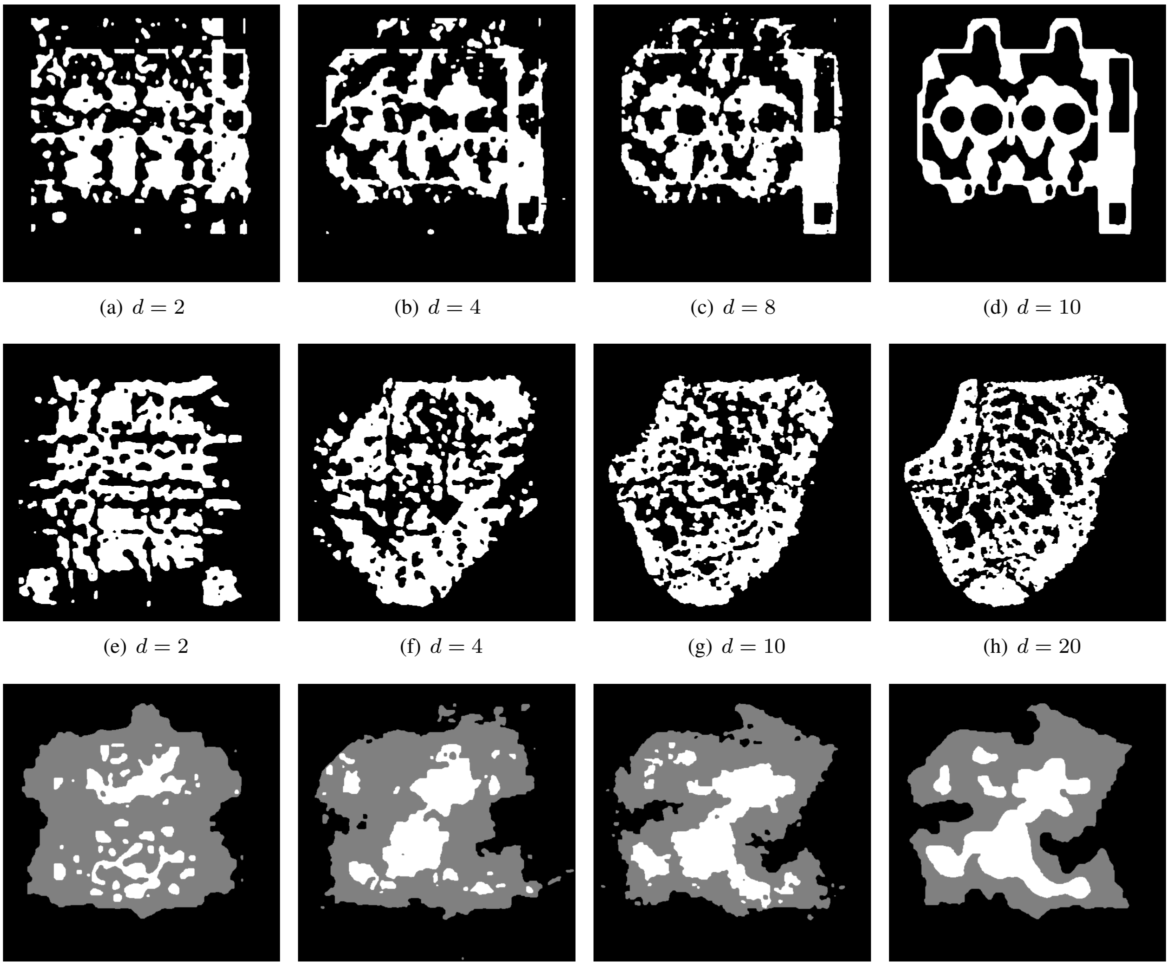


(g) Phantom 7



(h) Phantom 8

Fig. 6. The pixel error K as a function of the number of projections used in the reconstruction.

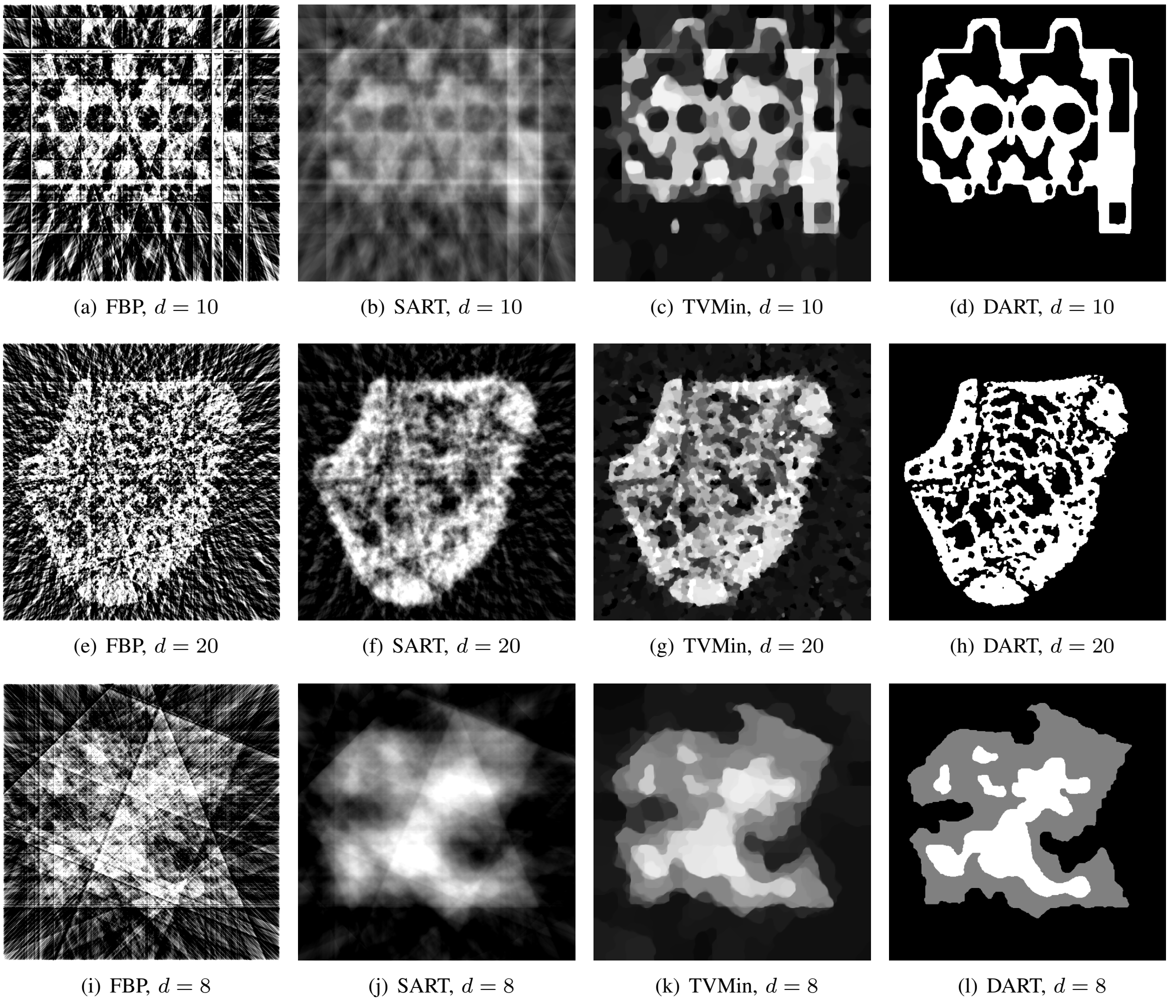


(a) $d = 2$ (b) $d = 4$ (c) $d = 8$ (d) $d = 10$

(e) $d = 2$ (f) $d = 4$ (g) $d = 10$ (h) $d = 20$

(i) $d = 2$ (j) $d = 4$ (k) $d = 6$ (l) $d = 8$

Fig. 7. DART reconstructions of Phantom 3 (top row), Phantom 5 (middle row), and Phantom 7 (bottom row) for various projection numbers.

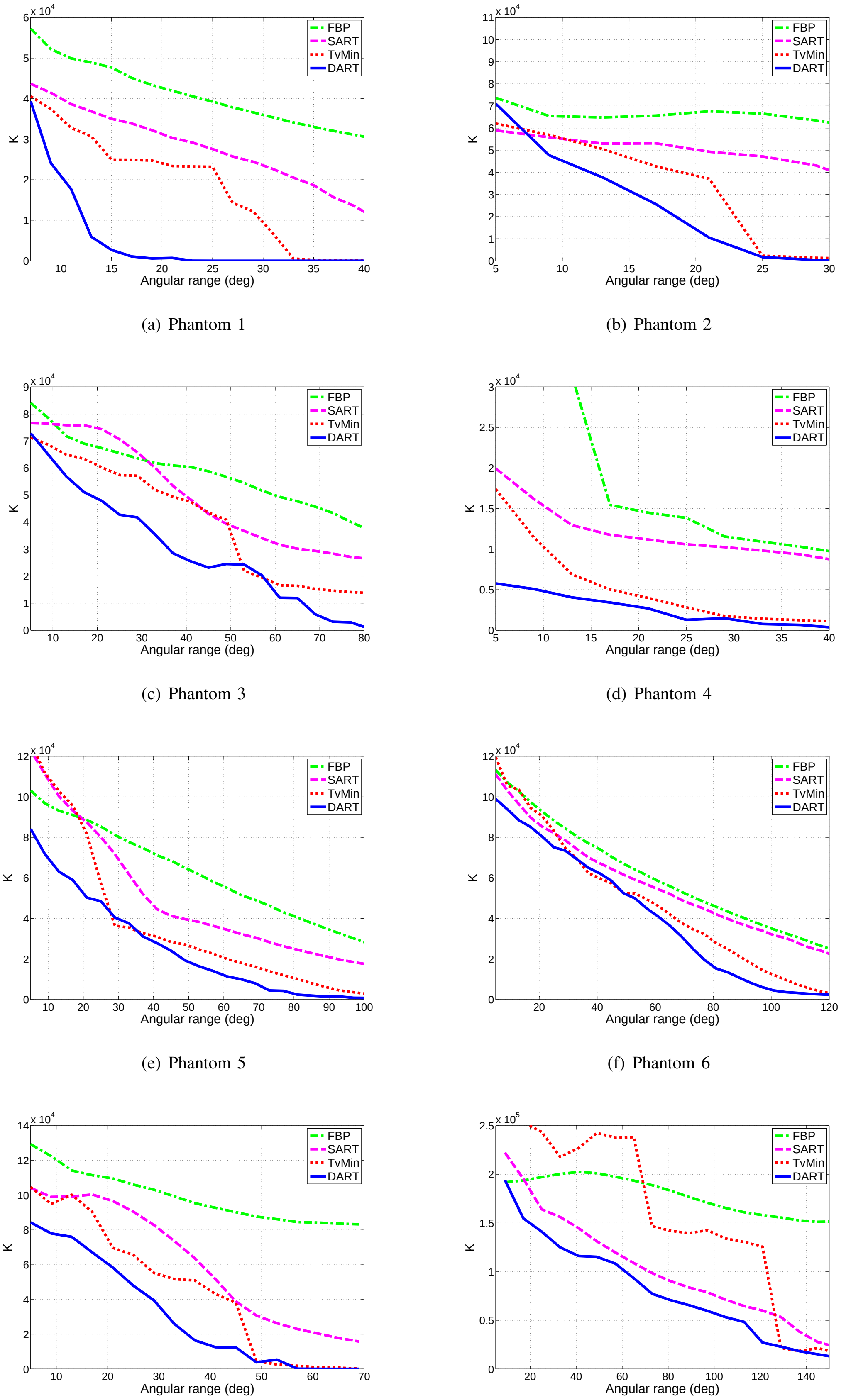


(a) FBP, $d = 10$ (b) SART, $d = 10$ (c) TVMin, $d = 10$ (d) DART, $d = 10$

(e) FBP, $d = 20$ (f) SART, $d = 20$ (g) TVMin, $d = 20$ (h) DART, $d = 20$

(i) FBP, $d = 8$ (j) SART, $d = 8$ (k) TVMin, $d = 8$ (l) DART, $d = 8$

Fig. 8. Comparison of FBP (column 1), SART (column 2), TVMin (column 3) and DART (column 4) for Phantom 3 using 10 projections (row 1), Phantom 5 using 20 projections (row 2), and Phantom 7 using 8 projections, respectively.



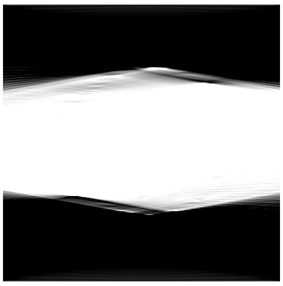
(a) Phantom 1 (b) Phantom 2

(c) Phantom 3 (d) Phantom 4

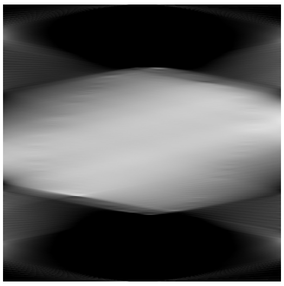
(e) Phantom 5 (f) Phantom 6

(g) Phantom 7 (h) Phantom 8

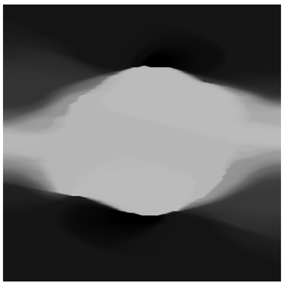
Fig. 9. Limited angle experiments: pixel error K as a function of the angular range of the projections for FBP, SART, TVMin, and DART.



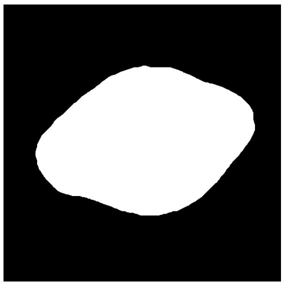
(a) FBP, $\alpha = 20^\circ$



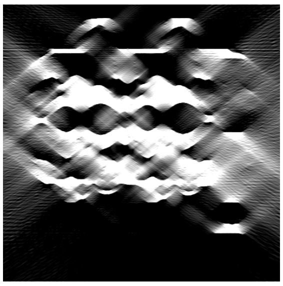
(b) SART, $\alpha = 20^\circ$



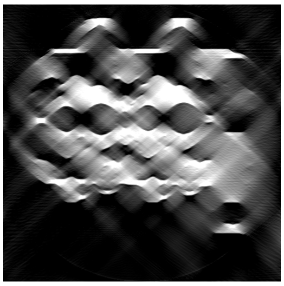
(c) TVMin, $\alpha = 20^\circ$



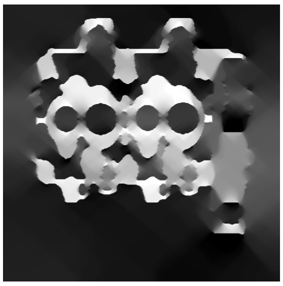
(d) DART, $\alpha = 20^\circ$



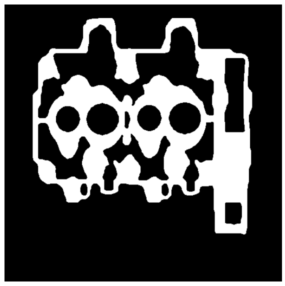
(e) FBP, $\alpha = 80^\circ$



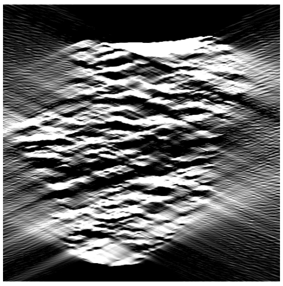
(f) SART, $\alpha = 80^\circ$



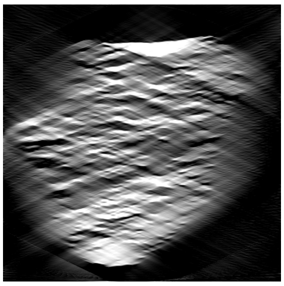
(g) TVMin, $\alpha = 80^\circ$



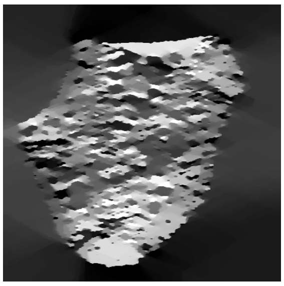
(h) DART, $\alpha = 80^\circ$



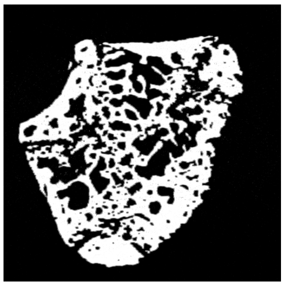
(i) FBP, $\alpha = 52^\circ$



(j) SART, $\alpha = 52^\circ$

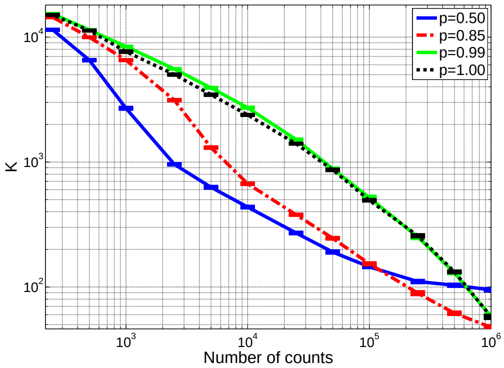


(k) TVMin, $\alpha = 52^\circ$

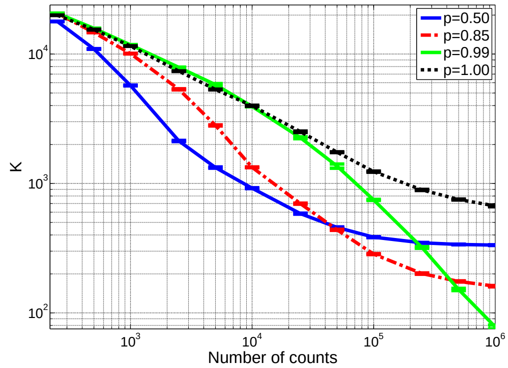


(l) DART, $\alpha = 52^\circ$

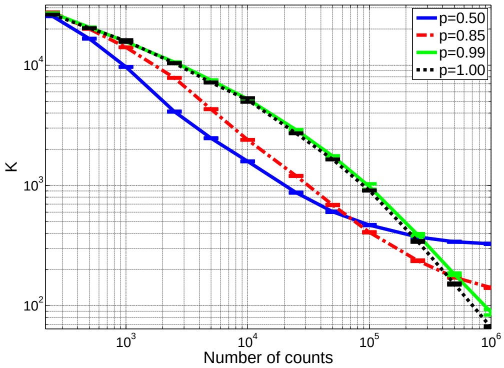
Fig. 10. Comparison of FBP (column 1), SART (column 2), TVMin (column 3) and DART (column 4) for Phantom 1 with an angular range of $\alpha = 20^\circ$ (row 1), Phantom 3 with an angular range of $\alpha = 80^\circ$ (row 2), and Phantom 5 with an angular range of $\alpha = 52^\circ$, respectively.



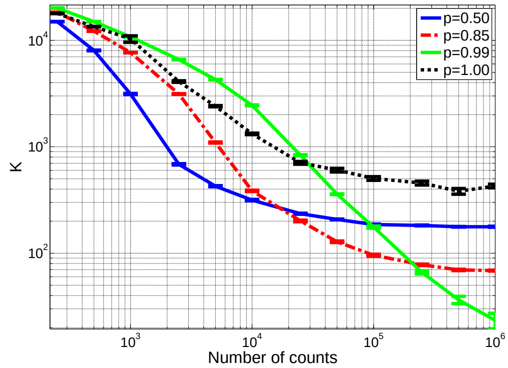
(a) Phantom 1 ($d = 10$)



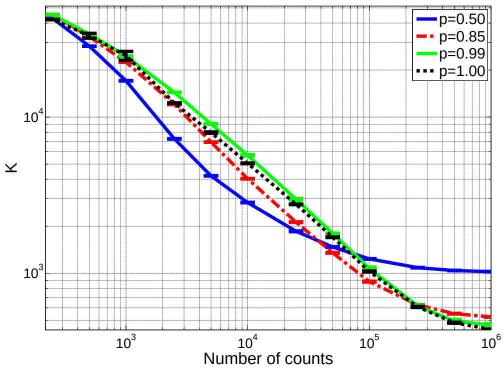
(b) Phantom 2 ($d = 15$)



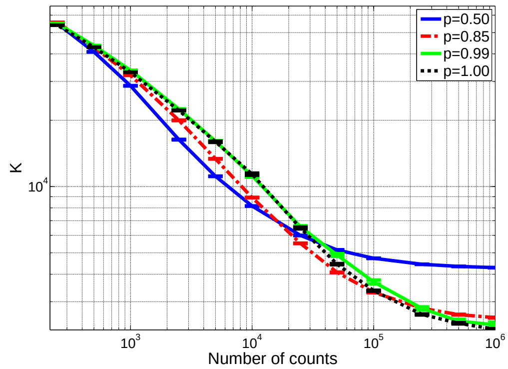
(c) Phantom 3 ($d = 25$)



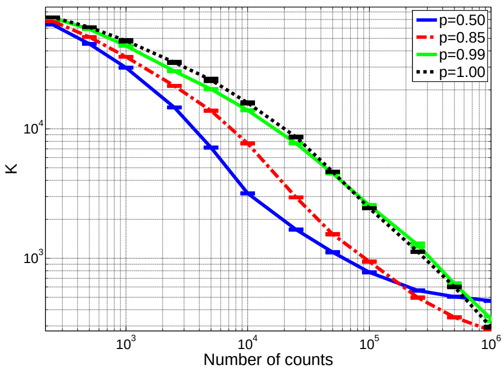
(d) Phantom 4 ($d = 25$)



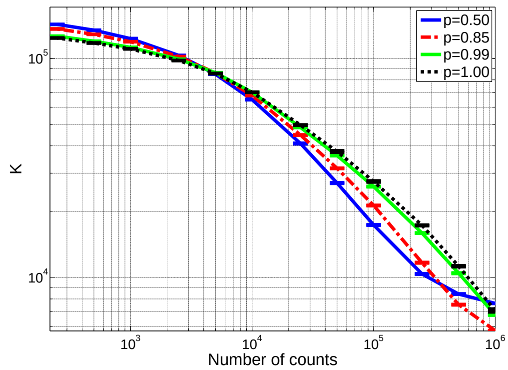
(e) Phantom 5 ($d = 100$)



(f) Phantom 6 ($d = 100$)

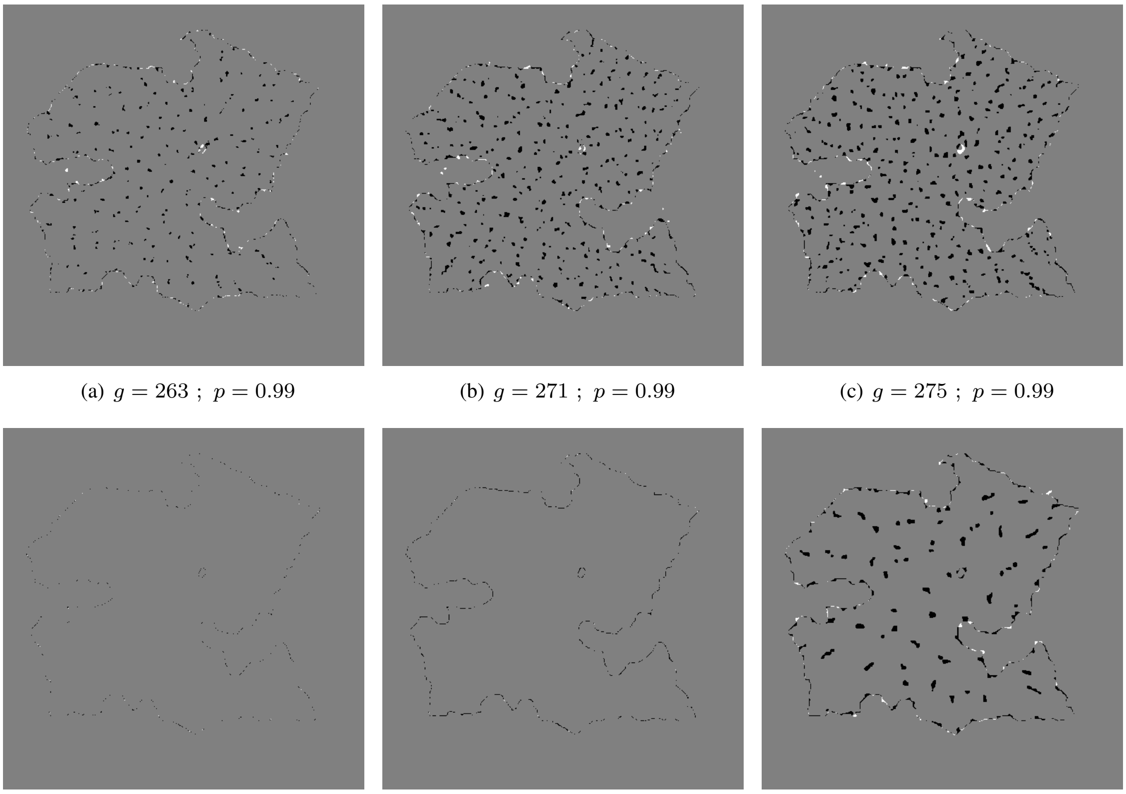


(g) Phantom 7 ($d = 25$)



(h) Phantom 8 ($d = 50$)

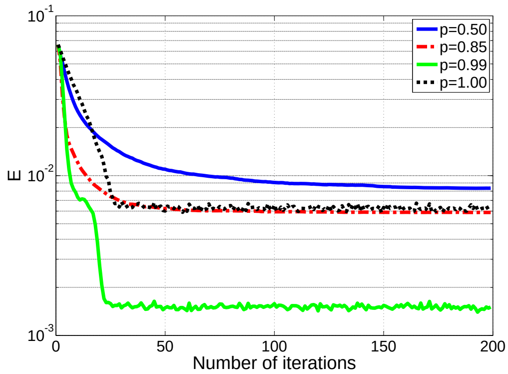
Fig. 15. The pixel error K as a function of the number of counts (SNR) for various values of the fix probability p.



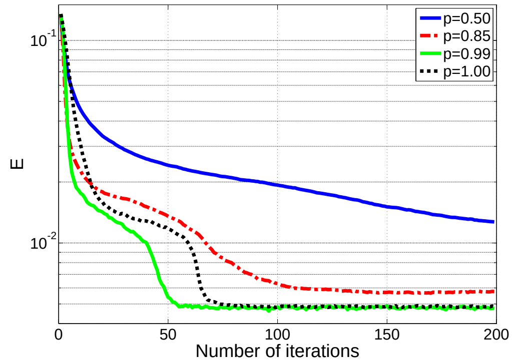
(a) $g = 263$; $p = 0.99$ (b) $g = 271$; $p = 0.99$ (c) $g = 275$; $p = 0.99$

(d) $g = 263$; $p = 0.85$ (e) $g = 271$; $p = 0.85$ (f) $g = 275$; $p = 0.85$

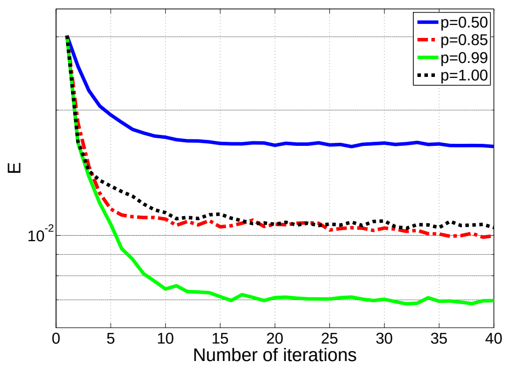
Fig. 17. Phantom 2: difference between the DART reconstruction ($d = 25$) and the original phantom when the grey level g is underestimated (a,d) or overestimated (b,c,e,f). The true grey level of the phantom was 255.



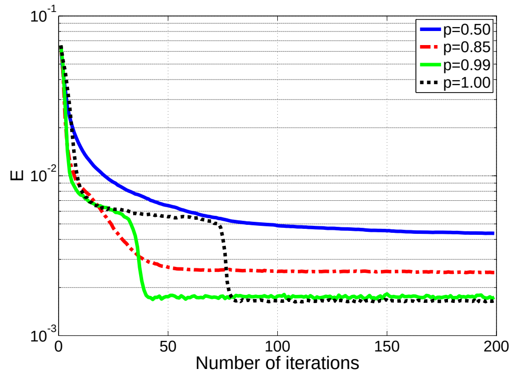
(a) Phantom 2 ($d = 6$)



(b) Phantom 3 ($d = 10$)

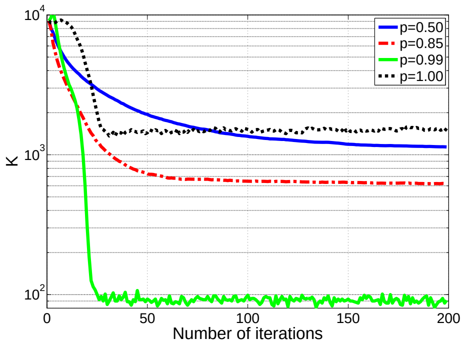


(c) Phantom 5 ($d = 50$)

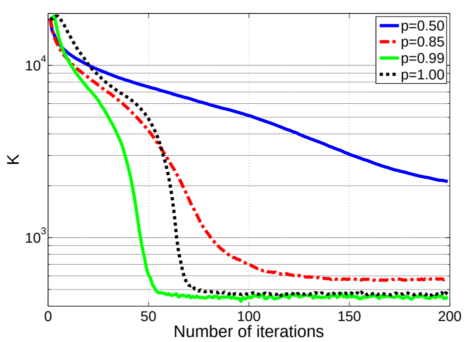


(d) Phantom 7 ($d = 10$)

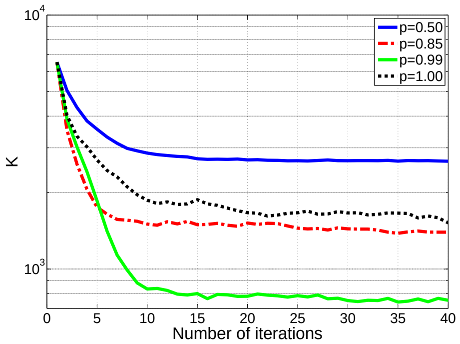
Fig. 18. The convergence rate: projection error as a function of the number of iterations.



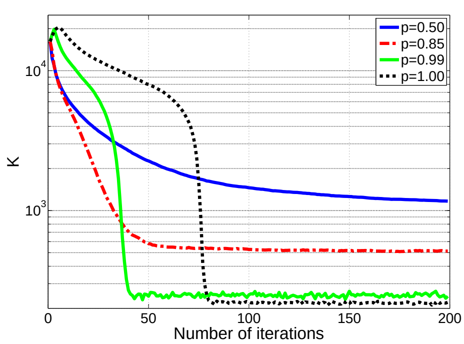
(a) Phantom 2 ($d = 6$)



(b) Phantom 3 ($d = 10$)

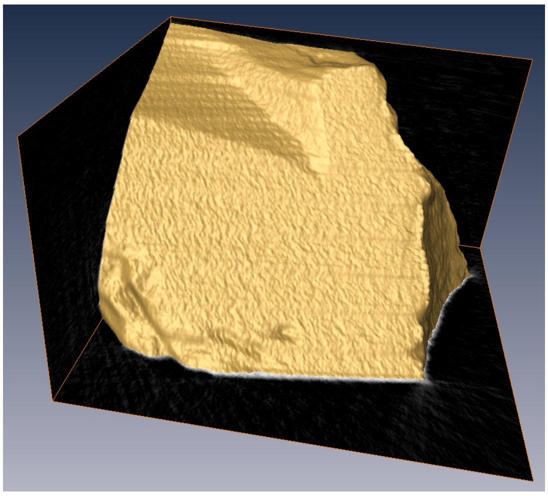


(c) phantom 5 ($d = 50$)

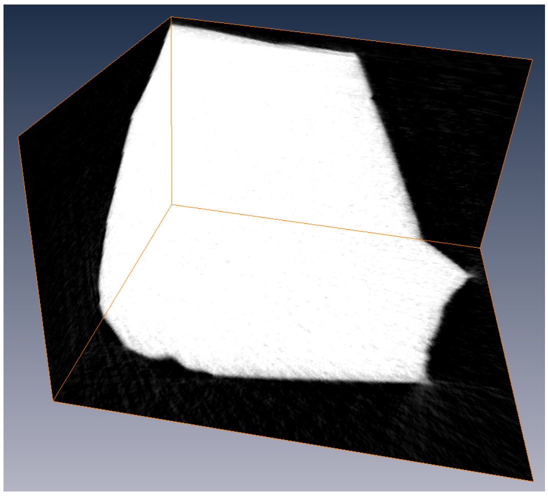


(d) Phantom 7 ($d = 10$)

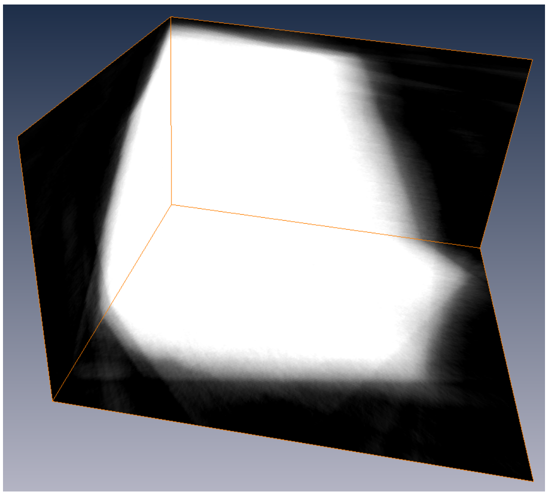
Fig. 19. The convergence rate: phantom error as a function of the number of iterations.



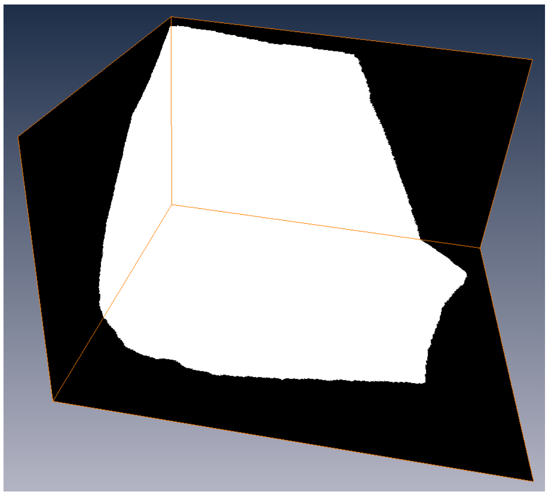
(a) SART-250: 3D model



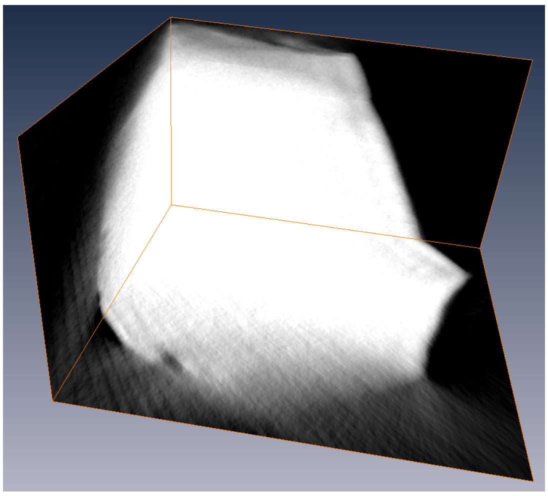
(b) SART-250: orthoslices



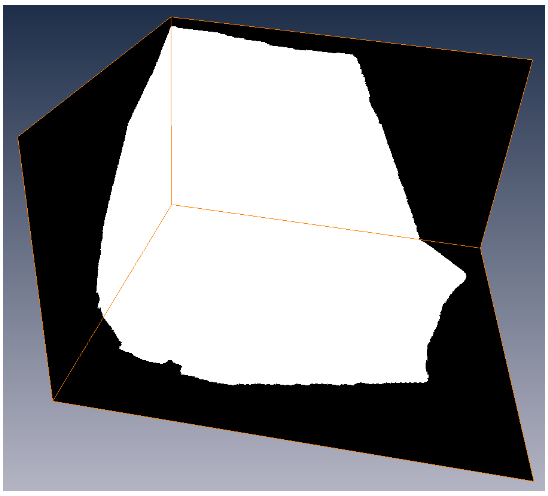
(c) SART-15: orthoslices



(d) DART-15: orthoslices



(e) SART-A: orthoslices



(f) DART-A: orthoslices

Fig. 20. Visualizations of reconstruction results for the experimental diamond μCT dataset. Each of the subfigures (b)-(f) show three partial orthogonal slices through the reconstructed volume.