Helical CT Reconstruction from Wide Cone-Beam Angle Data Using ART

Bruno M. Carvalho, Gabor T. Herman
Doctoral Program in Computer Science - The Graduate Center - City University of New York
365 5th Avenue, New York City, New York, 10036, USA
bruno_m_carvalho@yahoo.com, gherman@gc.cuny.edu

Abstract

We report on new results on the use of Algebraic Reconstruction Techniques (ART) for reconstructing from helical cone-beam Computerized Tomography (CT) data. We investigate two variants of ART for this task: a standard one that considers a single ray in an iterative step and a block version which groups several cone-beam projections when calculating an iterative step. Both algorithms were implemented using modified Kaiser-Bessel window functions, also known as blobs, placed on the body-centered cubic (bcc) grid. The algorithms were used to reconstruct a modified 3D Shepp-Logan phantom from data collected for the PI-geometry for two different maximum cone-beam angles ($\pm 9.46^\circ$ and $\pm 18.43^\circ$). Both scattering and quantum noise (for three different noise levels) were introduced to create noisy projections. The results presented here (for both noiseless and noisy data sets) point to the fact that, as opposed to filtered backprojection algorithms, the quality of the reconstructions produced by the ART methods does not suffer from the increase in the cone-beam angle.

1 Introduction

Algorithms for reconstructing CT data belong to the broader area of reconstruction from projections, which deals with the problem of reconstructing a function (image) from its line integrals. In CT reconstruction, a line integral of an object placed between an X-ray source and a detector is approximated by a relationship between the number of photons emitted by the X-ray source traveling toward the detector and the number of photons that reach that detector.

Introduced in the late 80s, helical CT consists of moving the patient table at a constant speed while the X-ray source and detector array rotate around the patient. Its development reduced the measurement acquisition time needed for full-body scanning from minutes to seconds, thus reducing motion artifacts and allowing the use of this technology in new applications, such as CT Angiography (CTA). Furthermore, the development of helical CT scanners also made CT more accessible: since the price of a CT scanner increased at a much slower rate than its speed, the improved throughput had the effect of lowering the cost associated with a CT exam.

The research in the field of 3D reconstruction algorithms for helical CT has been very active with several algorithms being published, but since many of them pre-weight the measurement values according to their cone angle, they tend to produce artifacts as the maximum cone-beam angle is increased, thus limiting the potential for increasing the acquisition speed by using bigger array detectors and maximum cone-beam angles. Since the Algebraic Reconstruction Techniques (ART) treat every measurement equally, as long as a sufficient number of measurements is collected for the desired resolution of the reconstruction these techniques should not be affected by an increase in the maximum cone-beam angle. Here we confirm this statement by presenting our first high-quality reconstructions from wide cone-beam helical CT data from low-contrast objects using such algorithms.

2 Algebraic Reconstruction Techniques

The ART algorithms belong to the class of reconstruction algorithms called series expansion methods. In this class of algorithms, the reconstruction problem is discretized at its beginning by assuming that the image f to be reconstructed can be approximated by a linear combination of N known basis functions b_j weighted by appropriate coefficients c_j, i.e.,

$$f(x,y,z) \approx \sum_{j=1}^{N} c_j b_j \left(\sqrt{(x-x_j)^2 + (y-y_j)^2 + (z-z_j)^2} \right), \quad (1)$$

where $\{(x_j, y_j, z_j)\}_{j=1}^{N}$ is the set of grid points in the three-dimensional (3D) Euclidean space to which the basis functions are shifted.

The problem of reconstruction from projections can then be formulated as the estimation of a column image vector c

Proceedings of the XVI Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'03)
1530-1834/03 $17.00 © 2003 **IEEE**

that must satisfy the system of approximate equalities:

$$Ac \approx Y, \tag{2}$$

given a measurement vector Y and a matrix A containing the contributions to the measured rays of the shifted basis functions. Thus, each individual measurement y_i of Y, for $1 \leq i \leq I$ (the total number of measurements), can be expressed as

$$y_i \approx \sum_{j=1}^{N} a_{i,j} c_j, \tag{3}$$

where $a_{i,j}$ denotes the contribution to the ith measurement of the jth basis function. One alternative notation for (3) is $y_i \approx \langle A_i^T, c \rangle$, where A_i^T denotes the transpose of the ith row of A. It is easy to see that this formulation in general will result in a very sparse projection matrix A in (2).

2.1 Standard ART

ART or algebraic reconstruction techniques form a family of iterative reconstruction algorithms, whose idea is to produce a sequence of vectors $c^{(0)}, c^{(1)}, c^{(2)}, \ldots c^{(k)}$, such that the sequence converges to a satisfactory solution of (2).

Here, we make use of the notation of [2], because it is appropriate for both algorithms to be described. Let M denote the number of times the X-ray source is pulsed while it moves along its helical path and L denote the number of measured rays in the cone-beam for a single pulse ($I = LM$). Using this notation, the traditional ART algorithm can be represented by the iterations:

$$\begin{array}{ll} c^{(0)} & \text{is arbitrary (often the zero vector),} \\ c_j^{(n+1)} = c_j^{(n)} + \lambda^{(n)} \frac{y_i - \langle A_i^T, c^{(n)} \rangle}{\sum_{k=1}^{N} a_{ik}^2} a_{ij} & \text{for } 1 \leq j \leq N, \\ n = 0, 1, \ldots, & i = n \bmod I + 1, \end{array} \tag{4}$$

where $\lambda^{(n)}$ is a relaxation parameter such that $0 < \lambda^{(n)} < 2$. We call a sequence of I iterations a *cycle*. The relaxation parameter $\lambda^{(n)}$ controls the rate at which the solution $c^{(n)}$ is updated in a single iterative step, with values closer to 1 making the algorithm to fit the individual measurement better than when using values closer to 0 (or 2). Although this algorithm has a mathematically well-defined limiting behavior [3], due to computational costs we would tend to run it for only a few cycles.

As can be seen in (4), the core procedure of ART consists in tracing the rays while measuring their current accumulated values (forward projection), and distributing the appropriate correction back to the intersected basis functions (back-projection). This can be implemented without the necessity of any data interpolation. In fact, the same ART algorithm can be used for reconstructing images from data acquired on systems with different geometries, as long as the tracing of the rays correspond to the locations of the measured rays. Moreover, the same ART algorithm can be used with different reconstruction grids, like the face-centered cubic (fcc) grid or the body-centered cubic (bcc) grid, provided that the new geometrical arrangement of basis functions is correctly incorporated in the ray tracing.

ART has been previously implemented with success for reconstructing volumes from Positron Emission Tomography (PET) [7] and Electron Microscopy (EM) [11] data, with only one cycle through the data. However, there was no publication reporting the use of ART for reconstructing from helical cone-beam CT data until our report of high-quality reconstructions of low-contrast objects in [5]. In [5] we also proposed alternative approaches to parallelizing the algorithm, as well as the use of block-ART algorithms for reconstructing from helical cone-beam CT data; results obtained by such block-ART algorithms were first published in [1].

2.2 Block-ART

The basis functions positioned closer to the helical X-ray source path have their coefficients updated more frequently in (4) than those that are positioned farther from it. We believe that this contributes to the slowness of the convergence process and/or to errors in the reconstructions produced by (4).

This problem was also addressed in [15], in the context of circular cone-beam CT data. There it was suggested that a certain alteration of ART leads to improved reconstructions. However, besides an illustration of its performance, no properties (such as limiting behavior of the algorithm) was given.

In [5] we suggested the use of block-ART algorithms, whose convergence behavior was analyzed in [2], to compensate for the non-uniformity in the updating process of (4). The projection data are divided into blocks[1] with the forward projection of the rays belonging to a block performed before any back-projection (update), which is now done for a block after all forward projections for it have been computed. The general block-ART algorithm is given by:

$$\begin{array}{ll} c^{(0)} & \text{is arbitrary (often the zero vector),} \\ c^{(n+1)} = c^{(n)} + A_i^T \Sigma^{(n)} \left(Y_i - A_i c^{(n)} \right), & \\ n = 0, 1, \ldots, & i = n \bmod M + 1, \end{array} \tag{5}$$

where $\Sigma^{(n)}$ is an $L \times L$ relaxation matrix, Y_i is the L-dimensional vector of those measurements which form the ith block and A_i is the correspondent submatrix of A.

[1] The theory presented in [2] is general in the sense that it covers all kinds of block groupings, even fully simultaneous algorithmic schemes.

One of the ideas proposed in [5] is to compute an additional weighting of the updates in the block-ART algorithm in such a way that if we had taken the projection data of a uniform object filling the whole reconstruction region (i.e. an object defined by (1) with all the c_j having the same value), then the update step of a single block would also produce a uniform object, assuming that the reconstructed object was initialized to zero. One way to perform this is by introducing in (5) a second $(N \times N)$ relaxation matrix $\Delta^{(n)}$ in front of the A_i^T that will enforce a component-dependent weighting. Assuming that the $\Sigma^{(n)}$ is the identity matrix, then we can achieve the aim mentioned above by choosing $\Delta^{(n)}$ to be a diagonal matrix whose jth entry is proportional to

$$\frac{1}{\sum_{l=1}^{L} a_{[(i-1)L+l]j} \left(\sum_{k=1}^{N} a_{[(i-1)L+l]k} \right)}. \quad (6)$$

In order for this to work we have to ensure that the value of the denominator in (6) is never zero. This usually demands the forming of blocks that correspond to more than one pulse of the X-ray source.

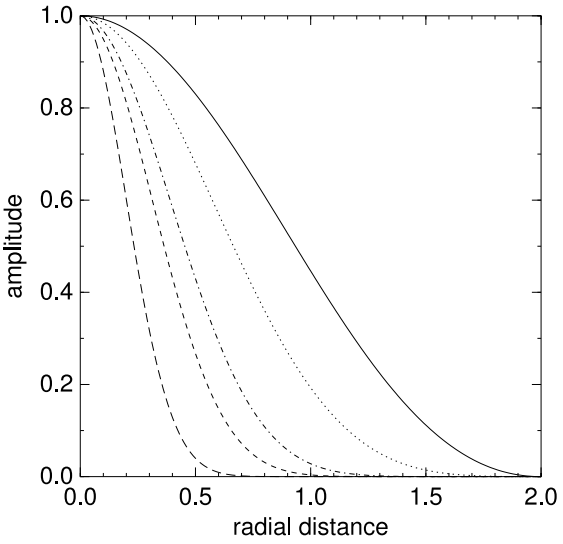

Figure 1. Radial profiles of blobs with the same support a (2.0) and m (2) but different α values. The values for α, ordered from left to right, are 100.0, 40.0, 25.0, 10.44 and 3.68.

3 Implementation

Traditionally, reconstruction algorithms use cubic voxels as basis functions, i.e., cubes with value 1 inside and 0 outside. However, the structure of ART algorithms makes possible the use of alternative basis functions for image reconstruction without requiring substantial changes. The use of modified Kaiser-Bessel window functions (also known as *blobs*) for digital image modeling, introduced in [9, 10], proved to be a better choice of basis functions than cubic voxels in several iterative algorithm implementations [12, 13].

There are several characteristics of blobs that make them attractive as basis functions. Although they are spatially localized, their Fourier transforms are effectively localized (i.e., they are almost band-limited). There are analytical formulas for their projections, Fourier transforms, gradients and Laplacians (see [9]), and the fact that they are rotationally symmetric makes their evaluation efficient. Finally, blobs can be constructed to have any finite number of continuous derivatives and they can act as low-pass filters, suppressing high-frequency noise.

Since blobs are functions with rotational symmetry, they can be represented as a function of a single variable, the distance r from the origin, by

$$b_{m,a,\alpha}(r) = \frac{\left[\sqrt{1-(r/a)^2}\right]^m I_m\left[\alpha\sqrt{1-(r/a)^2}\right]}{I_m(\alpha)}, \quad (7)$$

if $0 \leq r \leq a$ and 0 otherwise, where I_m denotes the modified Bessel function of order m, a is the radius of the support of the blob and α is a parameter controlling the blob shape. A volume is represented as a superposition of N scaled and shifted versions of the same blob, as in (1), with each b_j defined as the same $b_{m,\alpha,a}$. Figure 1 shows five profiles of blobs with the same m (2) and a (2.0), but having different values of α.

In the experiments reported in this article, as in the literature of image reconstruction using blobs mentioned above, the parameter m is set to 2, the resulting blobs are twice differentiable everywhere. However, the optimal values of a and α are dependent on the image and data characteristics, as well as on the grid that defines the placement of the blobs (1). One way to select these parameters is described in [13], where the authors produced a contour map reporting on the error (as a function of a and α) of approximating a uniform 3D function using blobs. An issue that has to be taken into account when choosing the blob parameters is that as the support of the blob increases, the more computationally expensive will be the reconstruction process.

Since blobs are rotationally symmetric functions, the integral of a ray intersecting a blob can be defined as a function of the distance from the center of the blob to the ray. This characteristic of the blobs can be used when producing computationally efficient algorithms. In practice, the values of the integrals of rays intersecting a blob are pre-calculated for a very fine sampling of the distance from the center (r) and stored in a lookup table, so that the computation of the integral is reduced to a memory access followed to by a

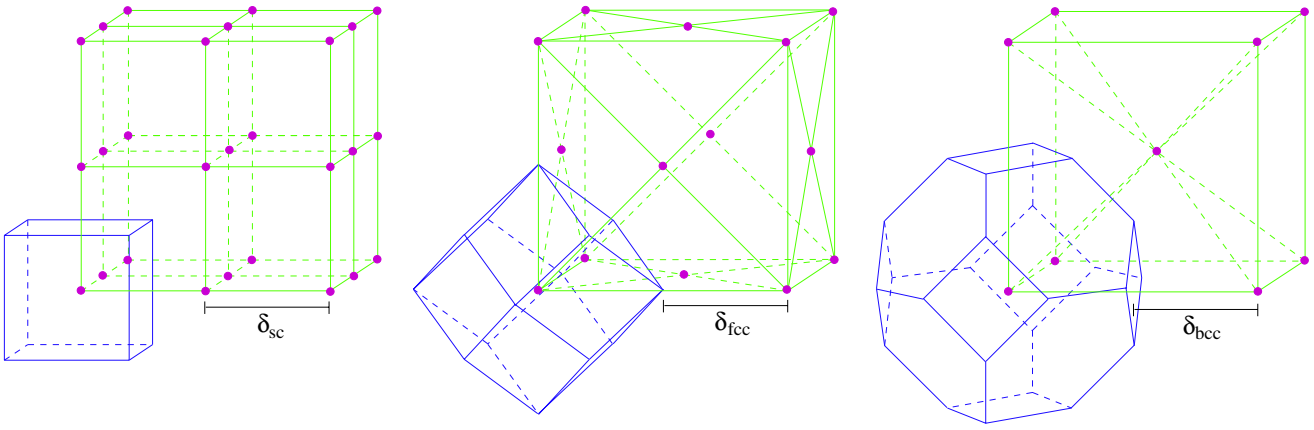

Figure 2. Three grids with the Voronoi neighborhood (voxel) of one of their grid points. From left to right, the simple cubic (sc) grid, the face-centered cubic (fcc) grid and the body-centered cubic (bcc) grid.

scaling.

As we can see from (1), there is no restriction on the grid at which the blobs are placed, and (4) and (5) can be easily adapted to work on any grid. The aim while choosing a reconstruction grid is to represent a three-dimensional image with similar accuracy to that provided by the traditional simple cubic grid but with fewer elements. Here we discuss the use of three different reconstruction grids, the traditional cubic grid, that we refer to as simple cubic (sc) grid, the face-centered cubic (fcc) grid and the body-centered cubic (bcc) grid.

Using $\mathbb{Z}$ to denote the set of all integers and δ a positive real number (that determines the sampling distance), we define the simple cubic grid (S_δ), the face-centered cubic grid (F_δ) and the body-centered cubic grid (B_δ) by:

$$S_\delta = \{(\delta c_1, \delta c_2, \delta c_3) \,|c_1, c_2, c_3 \in \mathbb{Z}\}\,, \tag{8}$$

$$F_\delta = \{(\delta c_1, \delta c_2, \delta c_3) \mid c_1, c_2, c_3 \in \mathbb{Z} \text{ and } c_1+c_2+c_3 \equiv 0 \,(\text{mod } 2)\}\,, \tag{9}$$

$$B_\delta = \{(\delta c_1, \delta c_2, \delta c_3) \mid c_1, c_2, c_3 \in \mathbb{Z} \text{ and } c_1 \equiv c_2 \equiv c_3 \,(\text{mod } 2)\}\,. \tag{10}$$

From the definitions above, the fcc and bcc grids can either be seen as one simple cubic grid without some of their grid points or as a union of shifted simple cubic grids (4 for the fcc and 2 for the bcc). Let G be any set of points in $\mathbb{R}^3$, then the Voronoi neighborhood of an element g of G is defined as

$$N_G(g) = \left\{v \in \mathbb{R}^3 \,|\forall h \in G, \|v-g\| \leq \|v-h\|\right\}. \tag{11}$$

In Figure 2, we illustrate the simple, the face-centered and the body-centered cubic grids and the Voronoi neighborhood [4] of their front-lower-left grid point. There are various approaches that can be used to compare the efficiency of three-dimensional grids. One of them is crystallography [8] that studies how atoms can be packed into periodic structures; it shows that both the fcc and the bcc have a better packing efficiency than the sc, i.e., the percentage of the total space filled by the maximal spheres fitting into the Voronoi neighborhoods of the fcc and the bcc grids is higher than for the sc. It was shown in [12] that ART reconstructions of PET data using blobs as basis functions produced 3D images with similar accuracy when using the simple cubic grid, the face-centered cubic grid or the body-centered cubic grid. However, 22.5% and 29.1% fewer grid points were needed for the fcc and bcc, respectively, than for the sc. Based on the experiments performed in [12] we have selected the bcc grid as the reconstruction grid and we have selected $a = 2.0$ and $\alpha = 10.444$ (see Figure 1).

4 Experiments

The helical CT geometry used in the experiments reported in this article is the *PI-geometry* [17]. This geometry has the detector array fitting exactly between two consecutive turns of the X-ray source trajectory. Named after its inventor, this detector is known as the Tam window, but it is also referred as the *PI-detector*. Figure 3 shows an example of the PI-geometry, where a ray is indexed by the angles β, γ and κ, and P indicates the pitch of the helical trajectory (the distance in the z direction traversed over one full turn of the X-ray source).

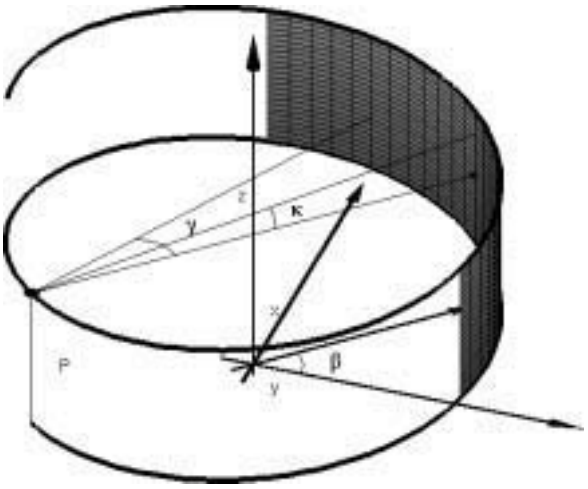

Figure 3. PI-geometry.

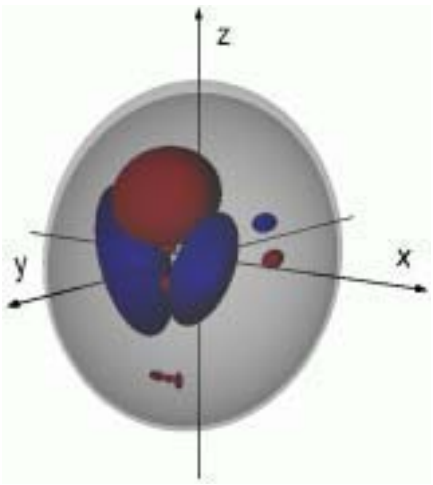

Figure 4. The 3D Shepp-Logan phantom.

The phantom used in the experiments presented here is the modified version of the 3D Shepp-Logan phantom described in [6]. The phantom is generated by adding the densities of twelve ellipsoids, resulting in an ellipsoid centered at the origin with a higher attenuating shell (density 2.0) than its interior (with densities in the range of $[1.00, 1.04]$) and totally enclosed within a cube with edge size of 2 units. Figure 4 shows a view of the modified 3D Shepp-Logan phantom with the ellipsoids rendered in blue having smaller attenuation coefficients (1.0) than the background (1.02) inside the high attenuating (2.0) outer shell and the red ellipsoids having greater attenuation coefficients (1.03 or 1.04) than the background.

We collected the data using two instances of the PI-geometry (Figure 3). Both data sets acquired over 2 full turns of the helix of radius 3 with 300 views (angles) per turn and the fan angle range $\gamma = [-21°, 21°]$. For the first data set, the cone-beam angle range $\kappa = [-9.46°, 9.46°]$ with an array of 64×128 detectors and a pitch of 2 units. For the second data set, the cone-beam angle range was $\kappa = [-18.43°, 18.43°]$ with an array of 128×128 detectors and a pitch of 4 units. (The number of turns of the helix is the same for both data sets because the program that computes the raysums generates them only for an integer number of turns.) Each raysum was obtained by averaging four equally spaced raysums over the detector area.

The phantom was discretized on a $128 \times 128 \times 128$ simple cubic grid by averaging the phantom on 27 points equally spaced inside each voxel.

Figure 5 shows the xz-slice at $y = -0.25$ of the phantom (a) and the reconstructions using ART ((b) and (c)) and block-ART ((d) and (e)) of the data sets mentioned above. ART was executed for 10 cycles with $\lambda^{(n)} = 0.024$ while block-ART was executed for 10 cycles using the $\Delta^{(n)}$ as defined by (6) but multiplied by 0.1. The blocks were formed by 8 views $90°$ apart from each other for both data sets, even though we could have formed blocks using less views for the data set with the wider cone-beam angle.

We also used data sets corrupted with noise that is similar to that of CT data collection. The number of photons emitted by the X-ray source over a period of time is a Poisson random variable (ξ). Because the number of photons emitted is large enough we can represent the distribution of this random variable as a normal distribution with mean ξ and standard deviation $\sqrt{\xi}$. We have introduced three different levels of noise by selecting ξ so that the minimum number of photons detected at any detector was $500,000$, $100,000$ and $10,000$. For these three levels, ξ is equal to $3,275,967$, $655,193$ and $65,519$, respectively, for the dataset with $\kappa_{max} = \pm 9.46°$ and $3,304,030$, $660,806$ and $66,080$ for the dataset with $\kappa_{max} = \pm 18.43°$. After this process, we simulated scattering by assigning 1% of the total number of detected photons of each detector to its 8 neighbors.

Figure 6 shows the same slice shown in Figure 5 of the ART and block-ART reconstruction for the three different noise levels for both datasets. Because the data were corrupted by noise, experience tells us that we should execute ART and block-ART for more cycles using a smaller relaxation parameter than in the noiseless case. For all the reconstructions from noisy data we executed ART and block-ART for 15 cycles using $\lambda^{(n)} = 0.005$ for ART and $\Delta^{(n)}$ as defined by (6) but multiplied by 0.05 for block-ART. The blocks used for these data sets were the same ones as used in the noiseless case.

Table 1 reports on the sums of the squared differences between the phantom and the ART and block-ART reconstructions. The sums were computed for the volume formed

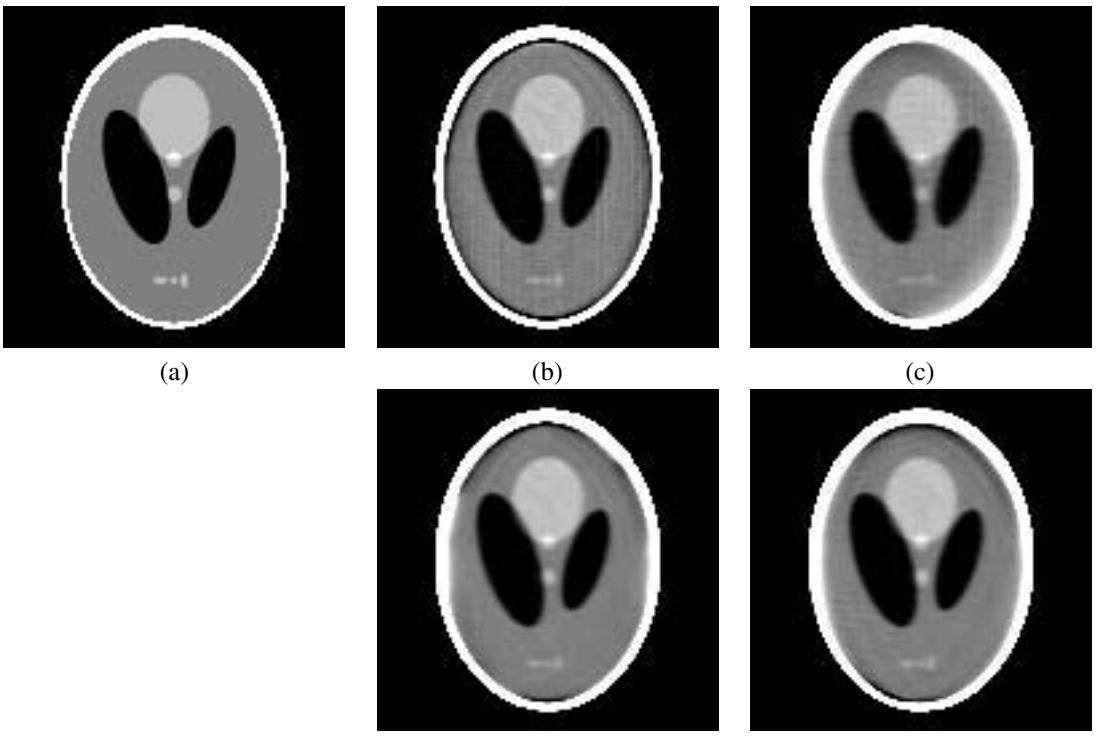

Figure 5. The xz-slice at $y = -0.25$ of the 3D Shepp-Logan phantom is shown in (a). The other images in the top row show the same slice of the ART reconstructions from perfect cone-beam data for $\kappa_{max} = \pm 9.46°$ (b) and for $\kappa_{max} = \pm 18.43°$ (c). The bottom row shows the block-ART reconstructions from perfect cone-beam data for $\kappa_{max} = \pm 9.46°$ (d) and for $\kappa_{max} = \pm 18.43°$ (e). (The gray-scale window range used was $[1.00, 1.04]$.)

Table 1. Squared differences between the phantom and the reconstructed volumes

Dataset		ART	Block-ART
Noiseless	± 9.46	9.7726	12.1361
	± 18.43	9.5543	14.0277
$500,000$	± 9.46	71.5268	31.3534
	± 18.43	74.1007	36.4828
$100,000$	± 9.46	71.7176	31.6540
	± 18.43	74.2386	36.7359
$10,000$	± 9.46	73.1572	53.4584
	± 18.43	75.6591	39.5138

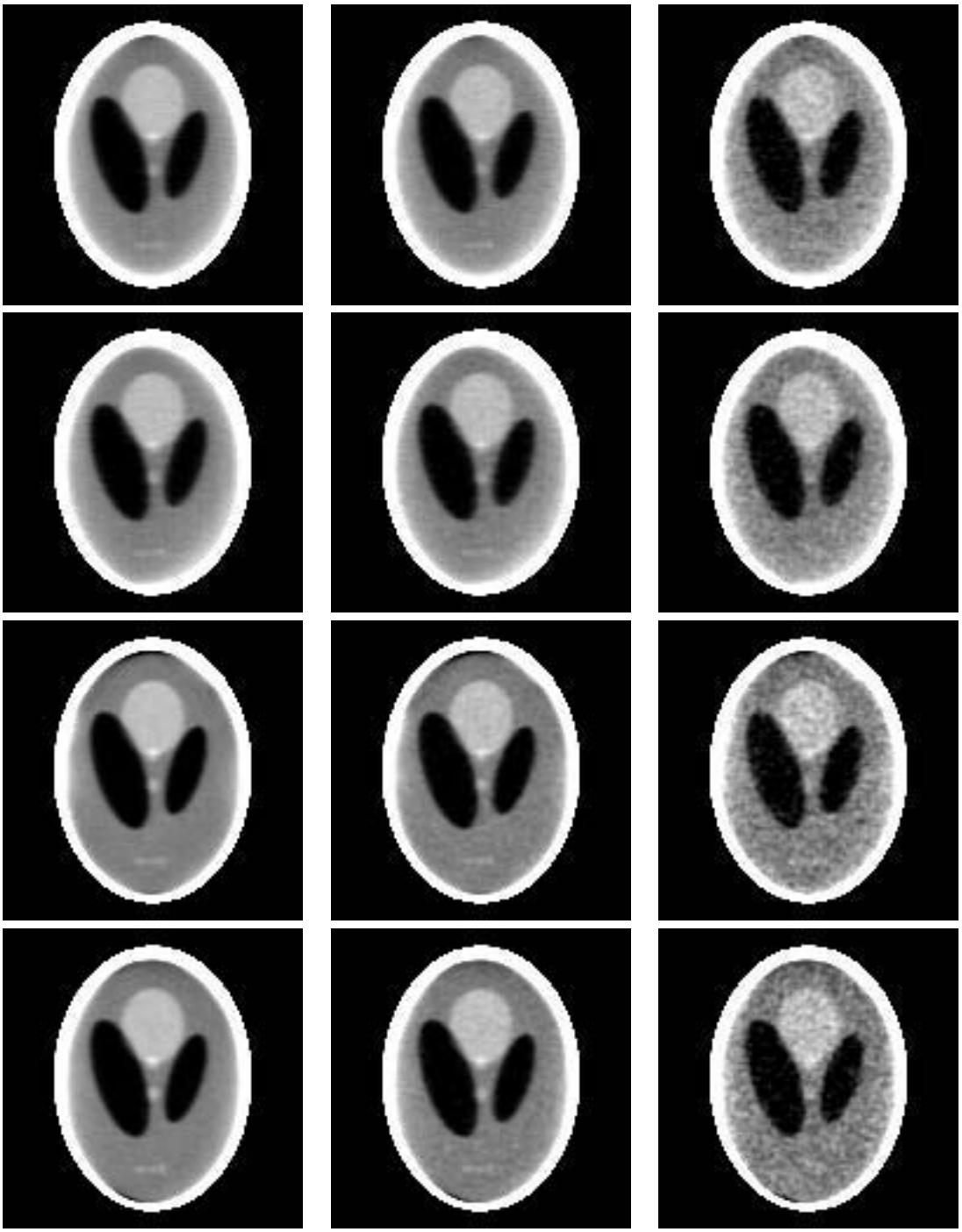

Figure 6. ART (top rows) and block-ART (bottom rows) reconstructions of the Shepp-Logan phantom from three noisy cone-beam data sets with $\kappa_{max} = \pm 9.46°$ (first and third rows) and $\kappa_{max} = \pm 18.43°$ (second and fourth rows) corrupted by different noise levels. The three columns correspond, from left to right, to the minimum number of detected photons of $500,000$, $100,000$ and $10,000$. (The gray-scale window range used was $[1.00, 1.04]$.)

by first selecting the voxels belonging to the background inside the high-attenuating shell and then eroding this volume using a $3 \times 3 \times 3$ cube as the structuring element. The total number of voxels for which the differences were computed was $495,400$. We can see from the reported differences that the block-ART algorithm is somewhat less affected by the simulated noise.

5 Conclusion

In this article we have shown that ART can be used to reconstruct low-contrast objects from cone-beam helical CT data even when collected from a wide cone-beam angle (approximately 37°), confirming the idea that the quality of the reconstructions produced by these techniques do not suffer from the increase in the cone-beam angle of the collected data as do some 3D filtered backprojection algorithms.

In order to implement ART more efficiently, we have placed the blobs on the body-centered cubic grid, thus requiring a smaller number of blobs than if we had used the simple cubic grid. Other approaches that can be used to implement the algorithms described here even more efficiently include the investigation of other block-ART algorithms for speeding up the convergence of the iterative solution and the use of commercially available texture mapping hardware for implementing (4) and (5) [14].

Filtered backprojection algorithms are known to produce severe artifacts if the cone-beam angle is greater than a few degrees. There are current efforts for dealing with this problem in the filtered backprojection framework, for example, [16] uses a detector that extends over n consecutive turns of the X-ray source (where n is an odd number), thus decreasing the artifacts due to a wide cone-beam angle but increasing the dosage needed for producing the reconstructions.

Acknowledgements

This research has been supported by NIH grant HL740472 (BMC and GTH) and CAPES-BRAZIL (BMC).

References

[1] B. Carvalho, G. Herman, and S. Matej. ART for helical cone-beam CT reconstruction. In *3D-2001 - The Sixth International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, pages 249–252, Pacific Grove, California, USA, 2001.

[2] P. Eggermont, G. Herman, and A. Lent. Iterative algorithms for large partitioned linear systems with applications to image reconstruction. *Linear Algebra Appl.*, 40:37–67, 1981.

[3] G. Herman. *Image Reconstruction from Projections: The Fundamentals of Computerized Tomography*. Academic Press, New York, 1980.

[4] G. Herman. *Geometry of Digital Spaces*. Birkhäuser, Boston, MA, 1998.

[5] G. Herman, S. Matej, and B. Carvalho. Algebraic reconstruction techniques using smooth basis functions for helical cone-beam tomography. In D. Butnariu, Y. Censor, and S. Reich, editors, *Inherently Parallel Algorithms in Feasibility and Optimization and their Applications*, Studies in Computational Mathematics, chapter 17, pages 307–324. Elsevier Science, Amsterdam, The Netherlands, 2001.

[6] C. Jacobson. *Fourier Methods in 3D-Reconstruction from Cone-Beam Data*. PhD thesis, Department of Electrical Engineering, Linköping University, 1996.

[7] P. Kinahan, S. Matej, J. Karp, G. Herman, and R. Lewitt. A comparison of transform and iterative reconstruction techniques for a volume-imaging PET scanner with a large acceptance angle. *IEEE Trans. Nucl. Sci.*, 42:2281–2287, 1995.

[8] C. Kittel. *Introduction to Solid State Physics*. John Wiley & Sons, New York, 7th edition, 1996.

[9] R. Lewitt. Multidimensional digital image representations using generalized Kaiser-Bessel window functions. *J. Opt. Soc. Amer. A*, 7:1834–1846, 1990.

[10] R. Lewitt. Alternatives to voxels for image representation in iterative reconstruction algorithms. *Phys. Med. Biol.*, 37:705–716, 1992.

[11] R. Marabini, G. Herman, and J. Carazo. 3D reconstruction in electron microscopy using ART with smooth spherically symmetric volume elements (blobs). *Ultramicrosc.*, 72:53–65, 1998.

[12] S. Matej and R. Lewitt. Efficient 3D grids for image reconstruction using spherically-symmetric volume elements. *IEEE Trans. Nucl. Sci.*, 42:1361–1370, 1995.

[13] S. Matej and R. Lewitt. Practical considerations for 3-D image reconstruction using spherically symmetric volume elements. *IEEE Trans. Med. Imag.*, 15:68–78, 1996.

[14] K. Mueller and R. Yagel. Rapid 3-D cone-beam reconstruction with the simultaneous algebraic reconstruction technique (SART) using 2-D texture mapping hardware. *IEEE Trans. Med. Imag.*, 19:1227–1237, 2000.

[15] K. Mueller, R. Yagel, and J. Wheller. Anti-aliased three-dimensional cone-beam reconstruction of low-contrast objects with algebraic methods. *IEEE Trans. Med. Imag.*, 18:519–537, 1999.

[16] R. Proksa, T. Köhler, M. Grass, and J. Timmer. The *n*-PI-method for helical cone-beam CT. *IEEE Trans. on Med. Imag.*, 19:848–863, 2000.

[17] H. Turbell and P.-E. Danielsson. Helical cone-beam tomography. *Internat. J. Imag. Systems Tech.*, 11:91–100, 2000.