Feasibility of GPU-assisted iterative image reconstruction for mobile C-arm CT

Yongsheng Pan, Ross Whitaker[a] and Arvi Cheryauka, Dave Ferguson[b]

[a]Scientific Computing and Imaging Institute, University of Utah, 72 S Central Campus Drive, 3750 WEB, Salt Lake City, UT, 84112;
[b]GE Healthcare - Surgery, 384 Wright Brothers Drive, Salt Lake City, UT, 84116

ABSTRACT

Computed tomography (CT) has been extensively studied and widely used for a variety of medical applications. The reconstruction of 3D images from a projection series is an important aspect of the modality. Reconstruction by filtered backprojection (FBP) is used by most manufacturers because of speed, ease of implementation, and relatively few parameters. Iterative reconstruction methods have a significant potential to provide superior performance with incomplete or noisy data, or with less than ideal geometries, such as cone-beam systems. However, iterative methods have a high computational cost, and regularization is usually required to reduce the effects of noise. The simultaneous algebraic reconstruction technique (SART) is studied in this paper, where the Feldkamp method (FDK) for filtered back projection is used as an initialization for iterative SART. Additionally, graphics hardware is utilized to increase the speed of SART implementation. Nvidia processors and compute unified device architecture (CUDA) form the platform for GPU computation. Total variation (TV) minimization is applied for the regularization of SART results. Preliminary results of SART on 3-D Shepp-Logan phantom using using TV regularization and GPU computation are presented in this paper. Potential improvements of the proposed framework are also discussed.

Keywords: C-arm CT, FDK, SART, GPU, CUDA, TV

1. INTRODUCTION

Computed tomography (CT)[1] has been widely used in modern medical and industrial applications. It is usually applied to reconstruct a three-dimensional (3D) object from a series of two-dimensional (2D) X-ray projections. CT reconstruction methods can be roughly categorized as analytical reconstruction methods such as filtered backprojection (FBP) methods and iterative reconstruction methods such as algebraic methods.

FBP methods are based on the Fourier slice theorem,[1] which means the Fourier transform of a parallel projection of a 2D object taken at a specific angle is equal to the 2D Fourier transform of the object along a line rotated by that angle. FBP methods are implemented by calculating the Fourier transform of each projection, multiplying the results using a weighting function, and backprojecting the weighting results over the image plane. In practice an FBP algorithm specifically designed to approximate cone beam geometries, proposed by Feldkamp,[2] is used by most manufacturers for efficiency.

Further author information: (Send correspondence to Ross Whitaker)
Ross Whitaker: E-mail: whitaker@cs.utah.edu, Telephone: 1 801 587 9549
Arvi Cheryauka: E-mail: Arvi.Cheryauka@med.ge.com, Telephone: 1 801 536 4653

Medical Imaging 2009: Physics of Medical Imaging, edited by Ehsan Samei, Jiang Hsieh,
Proc. of SPIE Vol. 7258, 72585J · © 2009 SPIE
CCC code: 1605-7422/09/$18 · doi: 10.1117/12.812162

FBP methods, however, require sufficient projection data with low noise level. This, combined with the approximations used for cone-beam acquisitions, results in reconstruction artifacts. Algebraic reconstruction methods have a significant potential to provide superior performance with incomplete and noisy data, particularly in the context of cone-beam acquisitions. These algebraic or iterative methods formulate the reconstruction problem as finding an array of unknowns using algebraic equations from the projection data. Specific algorithms, such as the algebraic reconstruction technique (ART) and the simultaneous algebraic reconstruction Technique (SART),[1] differ in the formulation of the objective and update strategy. Algebraic reconstruction methods require less data than FBP methods,[3] and they are more robust to the effects of noise. Furthermore, these methods can include nonlinear regularization terms—the equivalent of the linear filters used in FBP.

The iterative nature of algebraic methods poses a computational challenge, and the regularization of the solutions further increases the computation time. Many improvements on algebraic methods[456789] have recently been proposed. The methods proposed in[457] apply regularization methods to image reconstruction. The methods in[89] utilize graphics processing unit (GPU) to speed up the SART method. While Cg[10] is applied in[8] and OpenGL texture mapping[11] is utilized in[9] for speed-up, both platforms are nontrivial to implement.

The recently proposed API for graphics cards, called compute unified device architecture (CUDA), is more general for GPU computing, and is used in this paper to speed up the SART algorithm. Furthermore we use total variation (TV) minimization to regularize the SART algorithm. Preliminary results of the proposed method are provided in this paper.

This paper is organized as follows. Section 2 introduces the background information. The proposed method is shown in Section 3. Experimental results are provided in Section 4. Section 5 contains the summary and the future work.

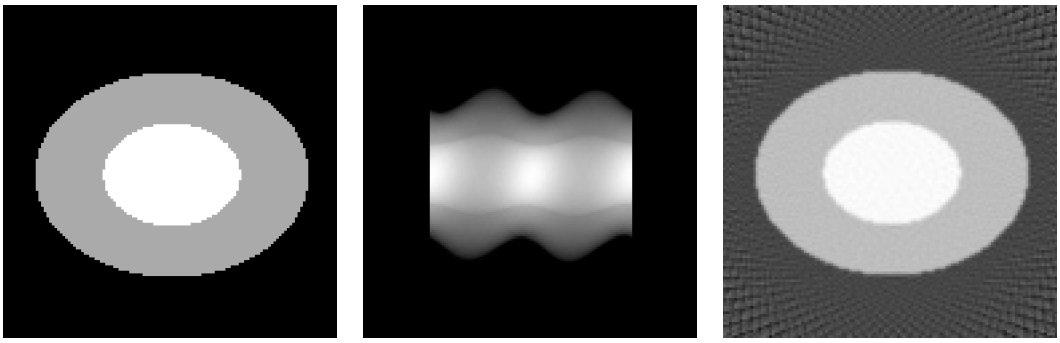
2. BACKGROUND

2.1 FBP

FBP methods reconstruct an object using Fourier transforms. They generally entail three steps. First, the Fourier transform $S_\theta(w)$ for each projection $P_\theta(t)$ is calculated. Second, the Fourier transformation $S_\theta(w)$ is multiplied with the weighting function $2\pi|w|/K$ in the frequency domain, where $w \in [-W_m, W_m]$ represents the bandlimited frequency and K is the number of projections. W_m is the maximum frequency related to the sampling interval during data acquisition. This step is usually implemented by a convolution in the time domain. Third, the inverse Fourier transforms of the filtered projections are summarized over the image plane (the backprojection process).

The reconstruction results from FBP methods are affected by the imaging beam types: parallel beam, equiangular fan beam, and equidistant fan beam. Fig. 1 shows an example of FBP reconstruction of a 2D object using the projections from the equidistant fan beam. It can be seen from Fig. 1(c) that the reconstruction is good except for some artifacts in the background.

Three-dimensional cone beam geometry is a 3D generalization of the 2D equidistant fan beam geometry, and results when a 3D object is irradiated with a point source and the x-ray transmission is measured on a plane using 2D equidistant fan beams. This mechanism is quite efficient. The Feldkamp algorithm (FDK)[2] is a practical FBP algorithm designed for 3D cone beam reconstruction. It is fast to compute, but only approximate, and results in reconstruction artifacts off of the center axis.



(a) (b) (c)

Figure 1. Illustration of FBP reconstruction. (a) Original synthetic image. (b) Projection of (a) using equidistant fan beam. (c) FBP reconstruction using (b).

2.2 SART

Algebraic reconstruction methods are designed to solve the following simultaneous equation system

$$p_i = \sum_{j=1}^{N} w_{ij} v_j \tag{1}$$

where p_i represents the ith projection, w_{ij} represents the weight which the voxel v_j contributes its value to the projection i. Reconstruction is achieved by finding v_j from the equation system (1).

The SART algorithm solves the equation system by iteratively applying a correction array to each voxel v_j as follows

$$v_j^{k+1} = v_j^k + \lambda \frac{\sum_i \left\{ w_{ij} \frac{p_i - \sum_{m=1}^{N} w_{im} v_m^k}{\sum_{m=1}^{N} w_{im}} \right\}}{\sum_i w_{ij}} \tag{2}$$

where λ is a constant coefficient.

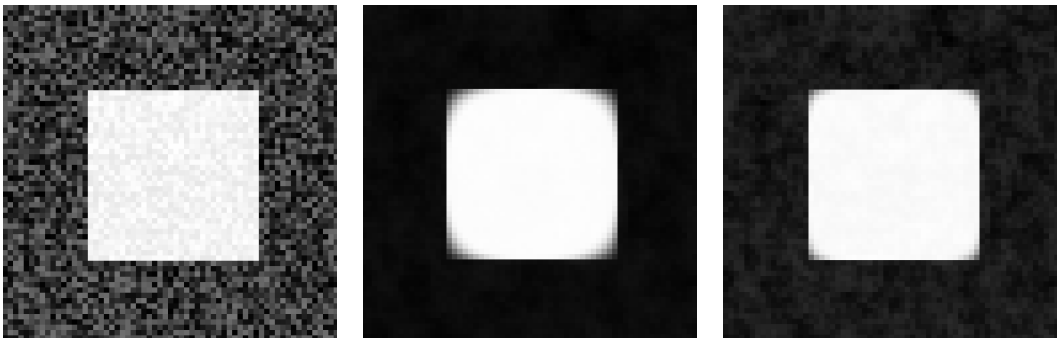
It has been proved that the SART algorithm converges globally to the solution of a weighted least square problem for $\lambda \in (0, 2)$.[12] An initial condition for the reconstruction v_j needs to be specified for iterative updates in the implementation.

2.3 Total Variation Minimization

Total variation minimization[13] is a nonlinear image denoising method that performs a gradient descent on the total variation of the image. Given an image f defined on domain Ω, this method seeks a smoothed image u which minimizes the following energy functional

$$F(u) = \int_\Omega \|\nabla u\| + \frac{\alpha}{2} \int_\Omega \|u - f\|^2 \tag{3}$$

where α is a constant coefficient. This penalty has the nice property of allowing (or preserving) straight, sharp edges, and thus allows solutions to have a piecewise flat property. It has been, very recently, tied to the more general method of *compressed sensing*.[14]



(a) (b) (c)

Figure 2. Illustration of TV smoothing. (a) Original image with noise. (b) Image smoothed using TV with $\alpha = 0.1$. (c) Image smoothed using TV with $\alpha = 0.5$.

The minimization of the energy functional (3) is numerically calculated using the following updating scheme

$$\frac{u_{n+1} - u_n}{\triangle t} = \nabla \cdot \frac{\nabla u_n}{\|\nabla u_n\|} - \alpha(u - f) \tag{4}$$

The minimization is iteratively processed until convergence.

Fig. 2 illustrates the smoothing effects of TV. Fig. 2(b) shows the TV results from the original image Fig. 2(a) with the coefficient $\alpha = 0.1$, and Fig. 2(c) shows the TV results with the coefficient $\alpha = 0.5$. It can be seen that the TV results contain less noise than the original image and that the results in Fig. 2(c) are closer to the original image with larger α.

3. PROPOSED METHOD

An integrated method is proposed in this paper to combine FBP, SART, TV, and GPU. Although SART is proved to converge, the initialization may make a difference on efficiency. In the proposed method, the reconstruction results from FDK are utilized as the initialization for SART. The TV method is applied after each iteration of SART to reduce the effects of noise. The proposed method takes the following steps for algebraic reconstruction.

Step 1. Perform FBP reconstruction for SART initialization,

Step 2. Perform TV on FBP results using (4),

Step 3. Compute image projection for each ray, i.e., $\hat{p_i} = \sum_{m=1}^{N} w_{im} v_m^k$,

Step 4. Compute correction image for each voxel, i.e., $(p_i - \hat{p_i}) / \sum_{m=1}^{N} w_{im}$,

Step 5. Perform image backprojection based on (2),

Step 6. Perform TV on SART results using (4),

where Steps 3-6 are iterated until convergence.

The proposed method is implemented using both CPU and GPU. CUDA is utilized for the GPU implementation on the Nvidia Geforce 8800 hardware. Allocation of the SART and TV algorithms on Nvidia's GT200 GPUs is in process. Because Step 4 has to be implemented in

CPU,[9] Step 3 and 5 are implemented using GPU. In the GPU implementation of Step 3, the 3D projection data are divided into smaller blocks, and the image projections on all these small blocks are computed simultaneously. The 3D volume data are divided into small blocks for parallel backprojection in Step 5.

4. EXPERIMENTAL RESULTS

Experimental results from the proposed method are shown in this section. The 3D Shepp-Logan head phantom, whose parameters are shown in Table 1, is used for performance evaluation. The head phantom comprises of 10 ellipsoids. The locations and the sizes of the ellipsoids are taken as those in.[1] The gray levels are revised to enhance the contrast between ellipsoids.

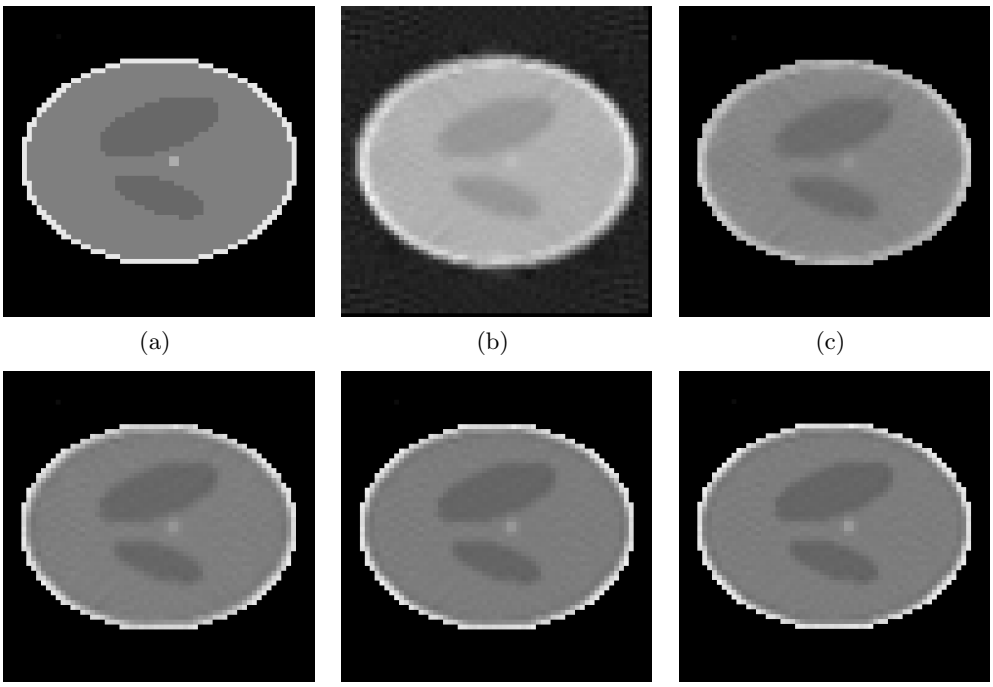
Table 1. Parameters of the revised 3D Shepp-Logan head phantom. The locations and the sizes of the ellipsoids are the same as those in.[1] The gray levels are revised to enhance the contrast between ellipsoids.

Ellipsoid	Center Coordinates	Axis Lengths	Rotation angle	Gray Level
a	(0, 0, 0)	(0.69, 0.92, 0.9)	0	2.0
b	(0, 0, 0)	(0.6624, 0.874, 0.88)	0	-0.9
c	(-0.22, 0, -0.25)	(0.41, 0.16, 0.21)	108	-0.2
d	(0.22, 0, -0.25)	(0.31, 0.11, 0.22)	72	-0.2
e	(0, 0.1, -0.25)	(0.046, 0.046, 0.046)	0	-0.2
f	(0, 0.1, -0.25)	(0.046, 0.046, 0.046)	0	0.2
g	(-0.8, -0.65, -0.25)	(0.046, 0.023, 0.02)	0	0.1
h	(0.06, -0.065, -0.25)	(0.046, 0.023, 0.02)	90	0.1
i	(0.06, -0.105, 0.625)	(0.56, 0.04, 0.1)	90	0.2
j	(0, 0.1, -0.625)	(0.056, 0.056, 0.1)	0	-0.2

Fig. 3 shows the experimental results for SART reconstruction of the object (64×64×64) in Fig. 3(a). 100 rotations are taken in this experiment, and 169×169 projections are generated in each rotation. The FBP results shown in Fig. 3(b) are utilized as initializations for SART. Fig. 3(c)-Fig. 3(f) show the SART results after 3, 20, 35, and 50 iterations respectively. It can be seen that the SART results are much closer to the original object than the FBP results.

The plots in Fig. 4 show the convergence rate of SART. Fig. 4(a) shows the difference between the original object and the reconstructed object with respect to iterations. Fig. 4(b) shows the changes of the differences between the SART results. The differences of real projections and projections computed from the SART results are shown in Fig. 4(c). It can be seen that the differences change rapidly within the first five iterations and that they change gradually thereafter. This can be justified by the fact that no big changes can be observed perceptually in Fig. 3(d)-Fig. 3(f). The SART results after three iterations may be good enough in real applications.

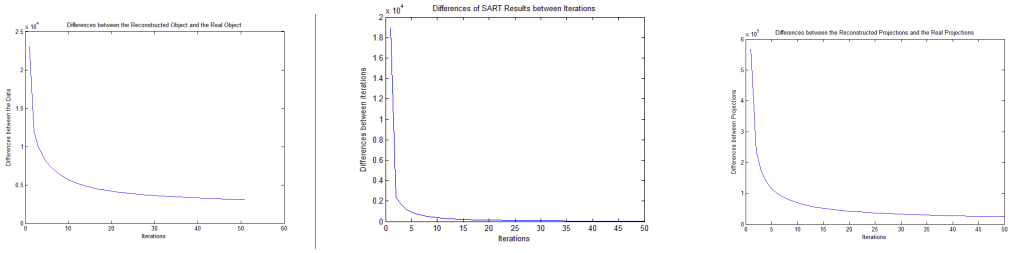
Fig. 5 demonstrates the effects of TV regularization on SART results. Fig. 5(c) shows the TV regularization of Fig. 5(b), which represents the FBP results of the original object Fig. 5(a). It can be seen that the noise is largely reduced after TV regularization. Fig. 5(e), Fig. 5(g), and Fig. 5(i) show the TV regularizations for Fig. 5(d), Fig. 5(f), and Fig. 5(h) respectively. The



(a) (b) (c)

(d) (e) (f)

Figure 3. Illustration of 3D SART using FBP as initializations. (Object volume: 64×64×64; Projection data: 100×169×169) TV regularization is not applied in this experiment. (a) The 25th slice from the phantom. (b) The reconstruction results from FBP. (c) The reconstruction results from SART after 3 iterations. (d) The reconstruction results from SART after 20 iterations. (e) The reconstruction results from SART after 35 iterations. (f) The reconstruction results from SART after 50 iterations.



(a) (b) (c)

Figure 4. Illustration of the convergence of 3D SART. (a) The difference between the original data and the reconstructed data w.r.t. iterations. (b) The difference between the SART results w.r.t iterations. (c) The difference of real projections and projections from the SART results.

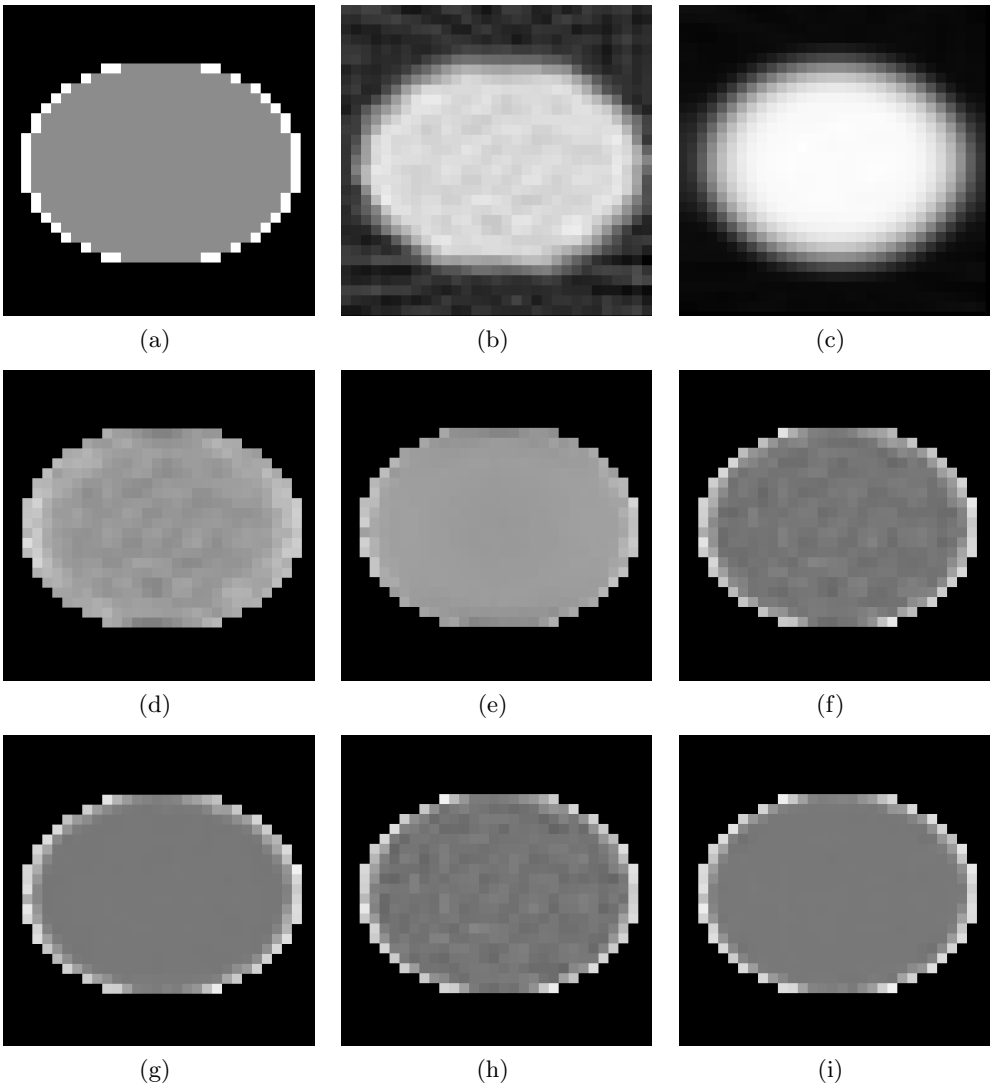


Figure 5. Illustration of 3D SART using TV regularizations. (Object volume: 32×32×32, Projection data: 26×43×43) (a) The 21st slice from the phantom. (b) Results from FBP. (c) Results from FBP after TV regularizations . (d) Results from SART after 3 iterations. (e) Results of (c) after TV regularizations. (f) Results from SART after 20 iterations. (g) Results of (f) after TV regularizations. (h) Results from SART after 35 iterations. (i) Results of (h) after TV regularizations.

artifacts are largely reduced after TV regularization. This demonstrates the potential of TV regularization for SART reconstruction.

Table 2 shows the computational efficiency of CPU SART and GPU SART. The parameters in Table 1 are used to define the 3D object with volume size 32×32×32. 26 rotations are applied for projection generation. Results from each rotation contains 43×43 projections. Sij in the table represents the computation for Step i on the jth iteration. The results in Table 2 show that the GPU implementation is more than 100 times faster than the CPU implementation on the average for Step 3 and Step 5 of the SART algorithm. It demonstrates the potential of CUDA GPU on CT reconstruction.

Table 2. Comparison of SART computation using CPU and GPU.

	S31	S51	S32	S52	S33	S53	S34	S54	S35	S55	Total
CPU (ms)	7.674	0.343	7.992	0.388	7.751	0.396	10.042	0.421	9.801	0.411	45.219
GPU (ms)	0.091	0.042	0.026	0.023	0.036	0.022	0.034	0.022	0.026	0.024	0.346
Speed-up	84	8	307	17	215	18	295	19	377	17	130

5. SUMMARY

Analytical CT reconstruction methods are restricted to the choice of acquisition geometries. They can not tolerate limited-angle or sparse acquisitions, and they are sensitive to the effects of noise. Therefore, iterative reconstruction methods, which are superior in such cases, may have large potentials in real applications. This paper presents the results of SART algebraic reconstruction on 3-D synthetic images. FBP is chosen as the initialization for SART, and TV regularization is applied to reduce the effects of noise. CUDA GPU is utilized to speed up the computation. Results on real CT data will be presented in later publications.

ACKNOWLEDGMENTS

The authors are grateful to Nvidia for instrumental GPU support.

REFERENCES

[1] Kak, A. and Slaney, M., [*Principles of Computerized Tomographic Imaging*], IEEE Press, New York (1999).

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

[3] Mueller, K., *Fast and Accurate three-dimensional reconstruction from cone-beam projection data using algebraic methods*, PhD thesis, the Ohio State University (1998).

[4] Villain, N., Goussard, Y., Idier, J., and Allian, M., "Three-dimensional edge-preserving image enhancement for computed tomography," *IEEE Trans. on Medical Imaging* **22**(10), 1275–1287 (2003).

[5] Yu, D. and Fessler, J., "Edge-preserving tomographic reconstruction with nonlocal regularization," *IEEE Trans. on Medical Imaging* **21**(2), 159–173 (2002).

[6] Hsieh, J. and Tang, X., "Tilted cone-beam reconstruction with row-wise fan-to-parallel rebinning," *Physics in Medicine and Biology* **51**, 5259–5276 (2006).

[7] Sidky, E. and Pan, X., "Few-view, cone-beam CT image reconstruction by GPU-accelerated total variation minimization," in [*9th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*], Kacheriess, M., Beekman, F., and Muller, K., eds., 60–63 (2007).

[8] Tita, R. and Lueth, T., "Online iterative reconstruction with the use of the Graphical Processing Unit(GPU)," in [*9th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*], Kacheriess, M., Beekman, F., and Muller, K., eds., 72–75 (2007).

[9] Mueller, K. and Yagel, R., "Rapid 3-D cone-beam reconstruction with the Simultaneous Algebraic Reconstruction Technique (SART) using 2-D texture mapping hardware," *IEEE Trans. on Medical Imaging* **19**(12), 1227–1237 (2000).

[10] Fernando, R. and Kilgard, M., [*The Cg Tutorial: The Definitive Guide to Programmable Real-Time Graphics*], Addison-Wesley Professional (March 2003).

[11] Shreiner, D., Woo, M., Neider, J., and Davis, T., [*OpenGL Programming Guide*], Addison-Wesley Professional, 5 ed. (2006).

[12] Jiang, M. and Wang, G., "Convergence of the Simultaneous Algebraic Reconstruction Technique (SART)," *IEEE Trans. on Image Processing* **12**, 957–961 (August 2003).

[13] Rudin, L. I., Osher, S., and Fatemi, E., "Nonlinear total variation based noise removal algorithms," *Physica D* **60**, 259–268 (1992).

[14] Candes, E., Romberg, J., and Tao, T., "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Information Theory* **52**, 489–509 (2006).