Numerical studies on Palamodov and Generalized Feldkamp algorithm for general cone-beam scanning

Kai Zeng\textsuperscript{a,}\textsuperscript{*}, Hengyong Yu\textsuperscript{b}, Erwei Bai\textsuperscript{c} and Ge Wang\textsuperscript{b}

\textsuperscript{a}Biomedical Engineering Dept, University of Iowa, 132 Hawkeye court, Iowa city, IA 52246, USA
\textsuperscript{b}Biomedical Imaging Division, VT-WFU School of Biomedical Engineering & Science, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, USA
\textsuperscript{c}Department of Electrical and Computer Engineering, University of Iowa, Iowa city, IA 52246, USA

Received 25 January 2007
Revised 10 April 2007
Accepted 17 April 2007

Abstract. Recently, the Palamodov algorithm, which was formulated for exact cone-beam reconstruction from data collected along a continuous locus, has been proven to be an excellent approximate algorithm for general cone beam reconstruction. The filtration-backprojection framework of the Palamodov algorithm is efficient for sequential and parallel implementation because its filtration step only involves a 1-D shift-invariant filtering along the tangent direction of the scanning trajectory. On the other hand, the generalized Feldkamp algorithm proposed by Wang et al. also allows approximate general cone beam reconstruction. In this paper, we report a numerical study comparing these two approximate methods for the cases of helical and saddle curves. In this study, the image quality is evaluated in terms of mean square error (MSE), modulation transform function (MTF), etc. The results demonstrate that the Palamodov algorithm consistently performs similar or better than the generalized Feldkamp algorithm.

Keywords: Computed tomography (CT), palamodov algorithm, generalized feldkamp algorithm, approximate reconstruction, general cone-beam scanning

1. Introduction

In the CT field, cone-beam reconstruction algorithms have significantly improved since 1980s. Generally speaking, these algorithms can be classified into two categories: approximate and exact algorithms. Both the categories have their relative merits in terms of certain image quality indexes [1]. Ideally, the exact algorithms can reconstruct image without any error. However, in practice, the approximate algorithms are relatively simpler to implement especially for general curves and often produce satisfactory or even better results in the cases of noisy, inconsistent and/or incomplete data. Therefore, both exact and approximate algorithms have been extensively applied for numerous biomedical applications.

*Corresponding author: Kai Zeng, Tel.: +1 319 353 4311; Fax: +1 540 231 0970; E-mail: kzeng@engineering.uiowa.edu.
The most well-known approximate algorithm is the Feldkamp algorithm (FDK algorithm), which was proposed in 1984 [2] and works very well especially for small cone angles. Since then, several Feldkamp-type algorithms have been proposed for improved performance and new capabilities. While T-FDK (Tent-FDK) [3], Slant-FDK [4], ERB-FDK (Error Reduction Based FDK) [5] are still limited to a circular scan, Wang et al. generalized the Feldkamp algorithm into a 3D case to solve the so-called long object problem [6]. These FDK-type algorithms are efficient for sequential and parallel implementation due to their filtration-backprojection framework in which a shift-invariant 1D filtration is applied. However, these FDK-type algorithms also cause image artifacts, especially when a cone angle becomes large or the object varies quickly along the z-direction.

Parallel to the improvement of approximate algorithms, great efforts have also been made to develop exact algorithms for quantitative studies. The fundamental work on exact cone-beam reconstruction was established by Smith [7], Tuy [8] and Grangeat [9], who independently developed different schemes for general cone beam reconstruction. However, these schemes are not suitable for direct practical implementation. In 2002, an efficient exact helical cone-beam CT algorithm was developed by Katsevich [10, 11]. Then, two practical implementations based on this were reported [12,13]. Later on, the Katsevich algorithm was also extended for general cone-beam reconstruction [14-17]. In 2004, Palamodov [18] made the first attempt to perform exact cone-beam reconstruction with a general scanning locus. Unfortunately, it has been recently found that his algorithm is not exact [19]. Nevertheless, his algorithm can be used for approximate reconstruction [19]. The importance of Palamodov's algorithm lies in its filtration-backprojection structure, and also the filtering direction is much more straightforward than exact algorithms [10,17].

In this paper, we describe numerical experiments for comprehensive evaluations of the Palamodov algorithm against the generalized Feldkamp algorithm. All the cone-beam data are acquired from a standard helix, which is the most popular in modern medical CT, and a saddle curve, which may be used for future 4D cardiac CT, with and without noise. The work reported in the paper will have a direct impact on several applications including the development of adaptive computed tomography for angiography [20]. Two types of reconstruction algorithms are sought in angiography. One is fast but less accurate and is used to provide real-time contrast information required for control. The other is an off-line reconstruction algorithm which can be relatively slow but must be more accurate.

The paper is organized as follows. The scanning geometry, and the two approximate cone-beam algorithms are briefly summarized in Section 2. Numerical experiments and results are presented in Section 3. The related issues are discussed in the last section.

2. Geometry and Algorithms

2.1. Scanning geometry

Let $S^2$ denote the unit sphere in $\mathbb{R}^3$. Assume that $\Gamma \subset \mathbb{R}^3$ is a 1st order differentiable curve parameterized by $y(s)$, $s \in \mathbb{R}$, and $f$ a second-order differentiable function with a compact $\Omega \subset \mathbb{R}^3 \setminus \Gamma$. Then, a cone-beam projection of $f$ along $\Gamma$ is defined by:

$$D_f(y, \beta) = \int_0^\infty f(y + t\beta) dt, \ (y, \beta) \in \Gamma \times S^2. \quad (1)$$
As shown in Fig. 1, a chord $L$ is defined as a line-segment whose two endpoints $y(s_b)$ and $y(s_t)$ are on $\Gamma$ with $s_b < s_t$. $I_{P_f}(x) = [s_b, s_t]$ denotes the chord parametric interval for $x \in L$.

For any $y(s)$, a local planar detector coordinate system is defined whose $u$ and $v$ axes are parallel to the unit vectors $u = (-\sin s, \cos s, 0)$ and $v = (0, 0, 1)$ respectively. The $u - v$ plane contains the $z$ axis of the global coordinate system. The origin $O'$ of the detector is on the $z$ axis with the $z$ level same as that of the current X-ray source. Based on such a local detector coordinate system, a cone beam projection $D_f(y, \beta)$ can be expressed by $D_f(\gamma, (u, v))$. For fixed $y$ and $\beta$, the corresponding $(u, v)$ is uniquely determined as the intersection between the line $\{x | x = y + t\beta, t \in \mathbb{R}\}$ and the detector plane, that is,

$$u(x, s, \varphi) = \frac{R(s) (\beta \cdot u)}{\beta \cdot \varphi},$$

(2.1)

$$v(x, s, \varphi) = \frac{R(s) (\beta \cdot v)}{\beta \cdot \varphi},$$

(2.2)

where $\varphi(s) = (\cos s, \sin s, 0)$.

To compare the Palamodov and generalized FDK algorithms, two important loci are used for numerical experiments: spiral and saddle curves. Without loss of generality, we define a general scanning trajectory as

$$y(s) = (R(s) \cos s, R(s) \sin s, h(s)), \ a \leq s \leq b.$$

(3)

It gives a helix (Fig. 2a) when $R(s) = R$ and $h(s) = h_0 s / 2\pi$ with constant $R$ and $h_0$. It generates a saddle curve (Fig. 2b) when $R(s) = R$ and $h(s) = h_0 \cos(2s)$. 

Fig. 1. Geometry of a cone-beam scan along a general curve.
2.2. Generalized FDK algorithm

In our local detector coordinate system,

\[ \alpha(s) = u = (-\sin s, \cos s, 0), \]

\[ \beta(x, s) = \frac{y(s) - x}{\|y(s) - x\|}, \]

\[ e(x, s) = \frac{\alpha(s) - (\alpha(s) \cdot \beta(x, s)) \beta(x, s)}{\|\alpha(t) - (\alpha(s) \cdot \beta(x, s)) \beta(x, s)\|}, \]

\[ \theta(s, x, \phi) = \beta(x, s) \cos \phi + e(x, s) \sin \phi. \]
The generalized FDK can be reformulated into the following equation (Appendix A) [6]:

\[
F(x) = \frac{1}{2} \int_0^{2\pi} \int_{-\pi}^{\pi} D_f(y(s), \theta) \text{filt} \left( \frac{R(s)(\beta \cdot u)}{\beta \cdot \varphi} - \frac{R(s)(\theta \cdot u)}{\theta \cdot \varphi} \right) \left( \frac{[(\beta \cdot u)(e \cdot \varphi) - (e \cdot u)(\beta \cdot \varphi)]}{(\theta \cdot \varphi)(\beta \cdot \varphi \cos \phi + e \cdot \varphi \sin \phi)^2} \right) d\phi ds
\] (5)

This algorithm is usually referred as a full-scan generalized FDK algorithm. To reduce the dose and the detector size, a half-scan version of the generalized FDK algorithm was proposed [21,22]. The formula for half-scan helical cone-beam reconstruction uses a weighting function to normalize the projection data [21]:

\[
F(x) = \frac{1}{2} \int_0^{\pi + \text{FanAngle}} \int_{-\pi}^{\pi} \text{filt} \left( \frac{R(s)^2}{\|y(s) - x\|^2} \right) \left( \frac{R(s)(\beta \cdot u)}{\beta \cdot \varphi} - \frac{R(s)(\theta \cdot u)}{\theta \cdot \varphi} \right) \left( \frac{[(\beta \cdot u)(e \cdot \varphi) - (e \cdot u)(\beta \cdot \varphi)]}{(\theta \cdot \varphi)(\beta \cdot \varphi \cos \phi + e \cdot \varphi \sin \phi)^2} \right) d\phi ds
\] (6)

Technically, generalized FDK algorithm can be implemented as follows: The 2D flat panel projection data are first weighted. Then 1D ramp filter is applied on these weighted projection data along the filtration plane. Finally, the volumetric data is reconstructed by 3D backprojection of the filtered projection data.

2.3. Palamodov algorithm

Let us denote

\[
\alpha(s) = \frac{y'(s)}{\|y'(s)\|},
\] (7.1)

\[
\beta(x, s) = \frac{y(s) - x}{\|y(s) - x\|},
\] (7.2)

\[
\theta(s, x, \phi) = \beta(x, s) \cos \phi + e(x, s) \sin \phi,
\] (7.3)

\[
e(x, s) = \frac{\alpha(s) - (\alpha(s) \cdot \beta(x, s)) \beta(x, s)}{\|\alpha(t) - (\alpha(s) \cdot \beta(x, s)) \beta(x, s)\|}.
\] (7.4)

The Palamodov algorithm can be reformulated as follows [19]:

\[
F(x) = \frac{1}{2\pi^2} \int_{s_b}^{s_t} \int_{\|y(s) - x\|}^{2\pi} \frac{\partial D_f(y(q), \theta)}{\partial q} \left( \frac{d\phi}{\sin \phi} \right) ds
\] (8)

Technically, the Palamodov algorithm can be implemented as follows. The 2D flat panel projection data are first differentiated. Then, the data are filtered using the 1D Hilbert transform along the filtration direction (trajectory tangential direction) (Fig. 1). Finally, a volumetric image is reconstructed by 3D backprojection of the filtered data from the PI interval \(I_{PI}(x) = [s_b, s_t] \).
Fig. 3. Images of the FORBILD phantom (window setting: [1.0, 1.1]). (a) The slice at $z = 0$, with typical structures being labeled, (b) the slice at $z = 0$, (c) the edge for MTF measurement, and (d) the homogenous region for noise measurement.

3. Numeric experiments

3.1. Geometry parameters

The simulation parameters for cone-beam scans along spiral and saddle curves are shown in Tables 1 and 2.

3.2. Phantom

Here we use the FORBILD head phantom to evaluate these algorithms. The phantom consists of a simple representation of anatomical structures which are important in evaluating CT image quality, such as calotte, frontal sinus and surrounding bones, homogeneous brain matter with low contrast objects, etc. Figure 3 shows the phantom images at $z = 0$ and $y = 0$. To measure spatial resolution, we computed the modulation transfer function (MTF) from the edge along the dash line in Fig. 3c. To measure image noise, we measured standard deviation for a homogenous area at $z = -0.5$. 
3.3. Experiments results

In this study, the Palamodov and Generalized FDK algorithms were evaluated according to the following criteria:

(a) Visualization of image quality: the reconstructed images at $z = 0$ and $y = 0$ were shown together with the same gray window (Figs 4–7). The visualization of the labeled structures were compared, as summarized in Tables 3 and 5.

(b) Reconstruction accuracy: the mean square errors (MSE) were computed using Eq. (9). (Tables 3 and 4).

$$MSE = \frac{1}{N} \sum_{i=0}^{N-1} (F_{\text{phantom}} (x_i) - F_{\text{recon}} (x_i))^2$$ (9)
Fig. 5. Reconstructed images from a helical scan at $z = 0$ (window setting: [1.0, 1.1]). The images in the first column were obtained with the Palamodov, full-scan and half-scan generalized Feldkamp algorithms, while that in the second column were obtained from noisy data respectively.

(c) Sensitivity to noise: the image noises were measured by the variance of the voxel values within the homogenous region (Tables 3 and 4). Number of photons was kept the same for each of the algorithms since the noise in reconstructed images is closely related to it. The total number of photons was uniformly distributed to each view (Tables 1 and 2). The Poisson noise was generated from the ideal projection data based on the photon numbers.

(d) Spatial resolution: By applying Fourier transformation to edge responses, MTF curves can be calculated. Here the edge responses are measured along the dashed line in Fig. 3c (Figs 6 and 9).

(e) Reconstruction speed: the computation time for each algorithm was recorded. All the programs were written in Visual C++. Our computer was Pentium 4 2.8 GHz having 3 GB of memory with Windows 2003 Server operating system.

In the case of helical cone-beam CT, all the three methods introduced artifacts around high contrast structures (Figs 4, 5). However, the results obtained using the Palamodov algorithm contained less streak artifacts around the bone structures (1st column in Figures 4 and 5 and Table 3). For Feldkamp-type algorithms, the ripple artifacts around structure 2 are stronger than those from Palamodov's algorithm.
Fig. 6. Spatial resolution analysis across the edge shown in Fig. 3c in the helical cone-beam CT case. (a) Edge profiles obtained using the Palamodov, full-scan and half-scan generalized FDK algorithms from a helical scan, and (b) the corresponding MTF curves.

The results obtained using the half-scan generalized FDK algorithm were superior as compared to the full-scan counterpart. The overall reconstruction accuracy was also characterized by MSE. Palamodov algorithm produced the smallest error, while the full-scan generalized FDK yielded the largest error. Besides, if the projection data contained Poisson noise, the reconstructed images became quite noisy (2nd column in Figs 4 and 5). The Palamodov algorithm seemed not as sensitive to noise as the Feldkamp-type as summarized in Table 3 and Figs 4 and 5. The Palamodov algorithm produced the worst spatial resolution since the edge reconstructed using it looked smoother than that from the Feldkamp-type algorithms. This is also apparent from the MTF curves. As far as the reconstruction speed is concerned, the Palamodov and half-scan Feldkamp ran much faster than the full-scan counterpart since they only processed half-scan data.

In the case of saddle curve cone-beam CT, all the three methods introduced artifacts more or less (Figs 7, 8) similar to those in the results from the helical scan, Palamodov's results produced images
with much fewer artifacts. And Feldkamp-type algorithms introduced very strong artifacts, making the structure 3 almost invisible (Fig. 7). The noise and spatial properties are pretty similar to helical locus. Unlike in helical scans, half-scan generalized Feldkamp algorithm introduced much stronger artifacts when compared to its full-scan counterpart as can be seen from Fig. 8.

4. Discussions and conclusion

It has been found in our experiments that the results with the Palamodov algorithm contained lower noise and smoother edge. Such a smoothing effect can be attributed to the bi-linear interpolation in the filtration step when the filtration was performed along a tangential direction. This interpolation also decreased spatial resolution. More sophisticated interpolation methods may be used to improve the spatial resolution.

Furthermore, the Palamodov algorithm requires the existence of the PI segment within the reconstruction region, which prevents it from working in a number of important cases such as a circular scan, a
Fig. 8. Reconstructed images from a saddle curve scan at $y = 0$ (window setting: $[1.0, 1.1]$). The images in the first column were obtained with the Palamodov, full-scan and half-scan generalized Feldkamp algorithms, while that in the second column were obtained from noisy data respectively.

scan consisting of disconnected loci, and so on. Future work is needed to generalize the Palamodov algorithm into more general cases.

In conclusion, we have numerically demonstrated that as compared to the generalized Feldkamp algorithm the Palamodov algorithm provides equally good or better image quality for cone beam reconstruction along helical and saddle curves. As compared with exact algorithms, such as Katsevich-type algorithms [15,23], the Palamodov algorithm is easier to implement and computationally efficient. However, it is not as widely applicable as the generalized Feldkamp algorithm.

Acknowledgment

This work is partially supported by NIH EB002667, NIBIB EB004287, NSF ECS-0555394 grants and a University of Iowa Mathematical and Physical Sciences Program Award.
Appendix A. Formulation of the generalized Feldkamp algorithm

In the local planar detector coordinate system shown in Fig. 1, the generalized Feldkamp algorithm can be expressed as [6]:

\[
F(x) = \frac{1}{2} \int_{0}^{2\pi} \int_{-\infty}^{\infty} \frac{R(s)}{(R(s) - r)^2} D_f(y(s), (u, v)) \text{filt}\left(\frac{R(s)t}{R(s) - r - u}\right) \frac{R(s)}{\sqrt{R(s)^2 + u^2 + v^2}} \, duds \quad (A.1)
\]
Table 1
Simulation parameters for helical cone-beam CT

<table>
<thead>
<tr>
<th></th>
<th>Palamodov</th>
<th>Full-scan FDK</th>
<th>Half-scan FDK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detector size ((W \times H))</td>
<td>240 \times 80 \text{mm}²</td>
<td>240 \times 160 \text{mm}²</td>
<td>240 \times 80 \text{mm}²</td>
</tr>
<tr>
<td>Detector matrix ((N_{col} \times N_{row}))</td>
<td>300 \times 100</td>
<td>300 \times 200</td>
<td>300 \times 100</td>
</tr>
<tr>
<td>Detector pixel size</td>
<td>0.8 \times 0.8 \text{mm}²</td>
<td>0.5 \times 0.8 \text{mm}²</td>
<td>0.5 \times 0.8 \text{mm}²</td>
</tr>
<tr>
<td>Scanning radius ((R))</td>
<td>500 mm</td>
<td>500 mm</td>
<td>500 mm</td>
</tr>
<tr>
<td>Number of projections per turn</td>
<td>600</td>
<td>600</td>
<td>600</td>
</tr>
<tr>
<td>Total number of turns</td>
<td>3</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Pitch ((h_0))</td>
<td>110 mm</td>
<td>110 mm</td>
<td>110 mm</td>
</tr>
<tr>
<td>X-ray source Z position</td>
<td>-165 mm, 165 mm</td>
<td>-165 mm, 165 mm</td>
<td>-165 mm, 165 mm</td>
</tr>
<tr>
<td>Photons per pixel (noisy data)</td>
<td>(1.0 \times 10^6)</td>
<td>(0.5 \times 10^6)</td>
<td>(1.0 \times 10^6)</td>
</tr>
<tr>
<td>Reconstruction matrix</td>
<td>256 \times 256 \times 256</td>
<td>256 \times 256 \times 256</td>
<td>256 \times 256 \times 256</td>
</tr>
<tr>
<td>Voxel size</td>
<td>0.78 \times 0.78 \times 0.78 \text{mm}³</td>
<td>0.78 \times 0.78 \times 0.78 \text{mm}³</td>
<td>0.78 \times 0.78 \times 0.78 \text{mm}³</td>
</tr>
</tbody>
</table>

Table 2
Simulation parameters for a saddle scan

<table>
<thead>
<tr>
<th></th>
<th>Palamodov</th>
<th>Full-scan FDK</th>
<th>Half-scan FDK</th>
</tr>
</thead>
<tbody>
<tr>
<td>Detector size ((W \times H))</td>
<td>240 \times 480 \text{mm}²</td>
<td>240 \times 480 \text{mm}²</td>
<td>240 \times 480 \text{mm}²</td>
</tr>
<tr>
<td>Detector matrix ((N_{col} \times N_{row}))</td>
<td>300 \times 600</td>
<td>300 \times 600</td>
<td>300 \times 600</td>
</tr>
<tr>
<td>Detector pixel size</td>
<td>0.8 \times 0.8 \text{mm}²</td>
<td>0.8 \times 0.8 \text{mm}²</td>
<td>0.8 \times 0.8 \text{mm}²</td>
</tr>
<tr>
<td>Scanning radius (R))</td>
<td>800 mm</td>
<td>800 mm</td>
<td>800 mm</td>
</tr>
<tr>
<td>(h_0)</td>
<td>110 mm</td>
<td>110 mm</td>
<td>110 mm</td>
</tr>
<tr>
<td>Total number of projections</td>
<td>800</td>
<td>800</td>
<td>800</td>
</tr>
<tr>
<td>Angular Range</td>
<td>([-\pi, \pi])</td>
<td>([-\pi, \pi])</td>
<td>([-\pi, 0.2978])</td>
</tr>
<tr>
<td>Photons per pixel (noisy data)</td>
<td>(1.0 \times 10^6)</td>
<td>(1.0 \times 10^6)</td>
<td>(1.0 \times 10^6)</td>
</tr>
<tr>
<td>Reconstruction matrix</td>
<td>256 \times 256 \times 256</td>
<td>256 \times 256 \times 256</td>
<td>256 \times 256 \times 256</td>
</tr>
<tr>
<td>Voxel size</td>
<td>0.78 \times 0.78 \times 0.78 \text{mm}³</td>
<td>0.78 \times 0.78 \times 0.78 \text{mm}³</td>
<td>0.78 \times 0.78 \times 0.78 \text{mm}³</td>
</tr>
</tbody>
</table>

Table 3
Visualization of the labeled structures for helical cone-beam CT

<table>
<thead>
<tr>
<th></th>
<th>Palamodov</th>
<th>Full-scan Feldkamp</th>
<th>Half-scan Feldkamp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noise free data</td>
<td>Clear</td>
<td>Clear</td>
<td>Clear</td>
</tr>
<tr>
<td>1</td>
<td>Clear</td>
<td>Not clear in y plane</td>
<td>Not clear in y plane</td>
</tr>
<tr>
<td>2</td>
<td>Clear</td>
<td>Not clear with density drop</td>
<td>Not clear with density drop</td>
</tr>
<tr>
<td>3</td>
<td>Clear</td>
<td>Clear</td>
<td>Clear</td>
</tr>
<tr>
<td>4</td>
<td>Not seen</td>
<td>Not seen</td>
<td>Not seen</td>
</tr>
<tr>
<td>Hardly seen</td>
<td>Not seen</td>
<td>Not seen</td>
<td>Not seen</td>
</tr>
<tr>
<td>Noisy data</td>
<td>Can be seen</td>
<td>Hardly seen</td>
<td>Hardly seen</td>
</tr>
<tr>
<td>1</td>
<td>Clear</td>
<td>Not clear in y plane</td>
<td>Not clear in y plane</td>
</tr>
<tr>
<td>2</td>
<td>Can be seen</td>
<td>Hardly seen</td>
<td>Hardly seen</td>
</tr>
<tr>
<td>3</td>
<td>Clear</td>
<td>Not clear in y plane</td>
<td>Not clear in y plane</td>
</tr>
<tr>
<td>Hardly seen</td>
<td>Not seen</td>
<td>Hardly seen</td>
<td>Hardly seen</td>
</tr>
<tr>
<td>Not seen</td>
<td>Not seen</td>
<td>Not seen</td>
<td>Not seen</td>
</tr>
</tbody>
</table>

where \(t\) and \(r\) are defined as

\[
\begin{align*}
  t &= x \cos s + y \sin s \\
  r &= -x \sin s + y \cos s,
\end{align*}
\]

\[
\theta = \frac{R(s)z(s)}{R(s) - r},
\]

\[
\tilde{z}(s) = z - h(s).
\]
According to the relationship between $D_f(y, (u, v))$ and $D_f(y, \beta)$, we have

$$u(x, s, \phi) = \frac{R(s)(\theta \cdot u)}{\theta \cdot \phi}, \quad (A.4.1)$$

$$\frac{R(s)t}{R(s) - r} = \frac{R(s)(\beta \cdot u)}{\beta \cdot \varphi}, \quad (A.4.2)$$

$$v(x, s, \phi) = \frac{R(s)(\theta \cdot v)}{\theta \cdot \varphi} = \frac{R(s)(\beta \cdot v)}{\beta \cdot \varphi}, \quad (A.4.3)$$

The change of $\theta$ to $\beta$ in A 5.3 is based on that $\theta \in \alpha, \beta$ and the intersection of this plane and the detector plane is a line parallel to the $u$ axis. Hence, all $\theta$ have the same $v$.

Substituting A 4 into A 1, we have

$$F(x) = \frac{1}{2} \int_0^{2\pi} \int_{-\pi}^{\pi} D_f(y(s), \theta) \text{filt} \left( \frac{R(s)(\beta \cdot u)}{\beta \cdot \varphi} - \frac{R(s)(\theta \cdot u)}{\theta \cdot \varphi} \right)$$

$$\frac{R(s)}{R(s)(\theta \cdot \varphi)} d\phi ds$$
The inner derivative term can be simplified as follows:

\[
\begin{align*}
    d\left( R(s) (\theta \cdot u) \right) &= d\left( R(s) \frac{\partial R(s) \theta \cdot u}{\theta \cdot u} \right) d\phi \quad d\psi = R(s) \frac{\partial R(s) \theta \cdot u}{\theta \cdot u} d\phi d\psi \\
    &= R(s) (\beta \cdot u) (e \cdot u) \left( \frac{\beta \cdot u}{(\beta \cdot u)^2} \right) d\phi \\
    &= R(s) (\beta \cdot u) (e \cdot u) \left( \frac{\beta \cdot u}{(\beta \cdot u)^2} \right) d\phi \\
    &= \frac{1}{2} \int_0^{2\pi} D_f(y(s), \theta) \left( \begin{array}{c} R(s) (\beta \cdot u) \\
    (\beta \cdot u) (e \cdot u) \left( \frac{\beta \cdot u}{(\beta \cdot u)^2} \right) \\
    \end{array} \right) d\phi ds
\end{align*}
\]

Therefore, the generalized FDK algorithm can be re-written as:

\[
F(x) = \frac{1}{2} \int_0^{2\pi} \frac{R(s)^2}{|y(s) - x|^2} d\phi ds \\
\int_\pi \varphi F(y(s), \theta) \left( \begin{array}{c} R(s) (\beta \cdot u) \\
    (\beta \cdot u) (e \cdot u) \left( \frac{\beta \cdot u}{(\beta \cdot u)^2} \right) \\
    \end{array} \right) d\phi ds
\]

References


