

Studies on implementation of the Katsevich algorithm for spiral cone-beam CT

Hengyong Yu^a and Ge Wang^b

^a*College of Communication Engineering, Hangzhou Institute of Electronics Engineering, Hangzhou, Zhejiang, 310018, P.R. China*

E-mail: yuhengyong@hotmail.com

^b*Department of Radiology, University of Iowa, Iowa City, Iowa 52242, USA*

E-mail: ge-wang@uiowa.edu

Abstract. Recently, Katsevich proposed the first exact helical cone-beam reconstruction algorithm that takes longitudinally truncated data collected from part of a scanning turn. A key theoretical basis is that any point inside the scanning helix belongs to one and only one PI-line. Unfortunately, he only provided a general formulation and did not describe the implementation details. To make this complicated algorithm more transparent, we report here specifically how to program the Katsevich algorithm in the planar detector geometry. As an important component, we also analyze the effects of various implementation parameters on the image quality. Our results are intended to be valuable for those who are interested in evaluating, improving and extending the Katsevich approach.

Keywords: Computed tomography (CT), spiral cone-beam, Katsevich algorithm, filtered backprojection (FBP), image artifacts

1. Introduction

Spiral cone-beam computed tomography (CT) is a major area of medical imaging research. Generally speaking, the spiral cone-beam scanning mode offers significant advantages in image quality than other modes such as multiple circular scans for reconstruction of a long object. While original spiral cone-beam algorithms are approximate [1,2], in 2002 Katsevich proposed a theoretically exact filtered backprojection (FBP) algorithm for this purpose [3], and improved it to use data mainly from the Tam-Danielson window [4], and performed some analysis [5]. Katsevich's work is a breakthrough and has attracted increasing interests in the area of spiral cone-beam CT.

However, Katsevich only formulated his algorithm in the generalized detector geometry and gave little implementation details. In this paper, the Katsevich algorithm is specialized in the planar detector geometry, and effects of various implementation options are analyzed in terms of image quality. In the next section, the Katsevich formula is briefly reviewed. In the third section, the steps of the Katsevich algorithm are formulated in the continuous domain. In the fourth section, these steps are discretized with some numerical details clarified. In the fifth section, simulation results are presented using the 3D Shepp-Logan phantom. In the last section, some relevant issues are discussed.

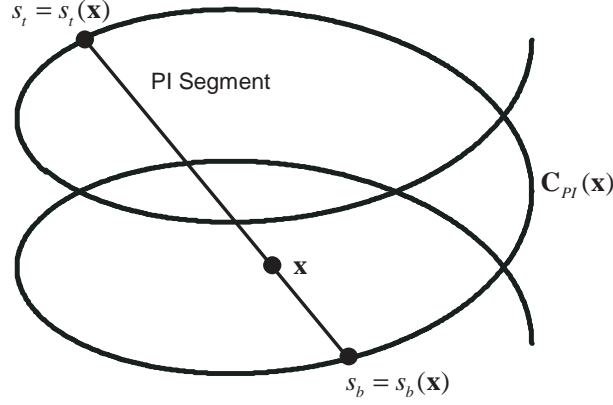


Fig. 1. Illustration of PI Segment.

2. Katsevich algorithm

Let a helical scanning locus be defined as

$$\mathbf{C} := \left\{ \mathbf{y} \in \mathbb{R}^3 : \mathbf{y}_1 = R \cos(s), \mathbf{y}_2 = R \sin(s), \mathbf{y}_3 = \frac{sh}{2\pi}, s \in \mathbb{R} \right\} \quad (1)$$

where $h > 0$. Let \mathbf{U} denote an open set strictly inside the helix

$$\overline{\mathbf{U}} \subset \{ \mathbf{x} \in \mathbb{R}^3, x_1^2 + x_2^2 < r^2 \} \quad 0 < r < R. \quad (2)$$

Let S^2 is the unit sphere in \mathbb{R}^3 and

$$D_f(\mathbf{y}, \boldsymbol{\beta}) := \int_0^\infty f(\mathbf{y} + t\boldsymbol{\beta}) dt, \boldsymbol{\beta} \in S^2 \quad (3)$$

$$\boldsymbol{\beta}(s, \mathbf{x}) := \frac{\mathbf{x} - \mathbf{y}(s)}{|\mathbf{x} - \mathbf{y}(s)|} \quad \mathbf{x} \in \mathbf{U}, \mathbf{y}(s) \in \mathbf{C}. \quad (4)$$

That is $D_f(\mathbf{y}, \boldsymbol{\beta})$, is the cone beam transform of f .

As shown in Fig. 1, any point \mathbf{x} strictly inside the helix belongs to one and only one PI line [6]. A PI line is a line segment with its endpoints on one helical turn. Let $s_b = s_b(\mathbf{x})$ and $s_t = s_t(\mathbf{x})$ denote parametric values corresponding to these endpoints, we name $\mathbf{I}_{PI}(\mathbf{x}) := [s_b(\mathbf{x}), s_t(\mathbf{x})]$ as the PI interval, and denote the part of the helix corresponding to $\mathbf{I}_{PI}(\mathbf{x})$ as $\mathbf{C}_{PI}(\mathbf{x})$.

Choose any $\psi \in C^\infty([0, 2\pi])$ with the properties

$$\begin{cases} \psi(0) = 0 & 0 < \psi'(t) < 1 \quad t \in [0, 2\pi] \\ \psi'(0) = 0.5 & \psi^{(2k+1)}(0) = 0 \quad k \geq 1 \end{cases} \quad (5)$$

Suppose that s, s_1, s_2 are related by

$$s_1 = \begin{cases} \psi(s_2 - s) + s & s < s_2 < s + 2\pi \\ \psi(s - s_2) + s_2 & s - 2\pi < s_2 < s \end{cases} \quad (6)$$

For example, the function $\psi(t) = t/2$ satisfies condition Eq. (5) and leads to

$$s_1 = (s + s_2)/2 \quad s - 2\pi < s_2 < s + 2\pi \quad (7)$$

Let us also denote

$$\mathbf{u}(s, s_2) = \begin{cases} \frac{(\mathbf{y}(s_1) - \mathbf{y}(s)) \times (\mathbf{y}(s_2) - \mathbf{y}(s))}{|(\mathbf{y}(s_1) - \mathbf{y}(s)) \times (\mathbf{y}(s_2) - \mathbf{y}(s))|} \text{sgn}(s_2 - s) & 0 < |s_2 - s| < 2\pi \\ \frac{\dot{\mathbf{y}}(s) \times \ddot{\mathbf{y}}(s)}{|\dot{\mathbf{y}}(s) \times \ddot{\mathbf{y}}(s)|} & s_2 = s \end{cases} \quad (8)$$

Fix $\mathbf{x} \in \mathbf{U}$ and $s \in \mathbf{I}_{PI}(\mathbf{x})$, find $s_2 \in \mathbf{I}_{PI}(\mathbf{x})$ such that the plane through $\mathbf{y}(s)$, $\mathbf{y}(s_2)$ and $\mathbf{y}(s_1(s, s_2))$ containing \mathbf{x} . More precisely, we have to solve the following equation for s_2 :

$$(\mathbf{x} - \mathbf{y}(s)) \cdot \mathbf{u}(s, s_2) = 0, \quad s_2 \in \mathbf{I}_{PI}(\mathbf{x}) \quad (9)$$

It was shown [4] that this construction defines a smooth function $s_2 := s_2(s, \mathbf{x})$.

Consequently, $\mathbf{u}(s, \mathbf{x}) := \mathbf{u}(s, s_2(s, \mathbf{x}))$. The following theorem is Katsevich's main result [4]:

Theorem 1. For $f \in C_0^\infty(\mathbf{U})$ one has

$$f(\mathbf{x}) = -\frac{1}{2\pi^2} \int_{\mathbf{I}_{PI}(\mathbf{x})} \frac{1}{|\mathbf{x} - \mathbf{y}(s)|} \int_0^{2\pi} \frac{\partial}{\partial q} D_f(\mathbf{y}(q), \Theta(s, \mathbf{x}, \gamma))|_{q=s} \frac{d\gamma}{\sin \gamma} ds \quad (10)$$

where $\Theta(s, \mathbf{x}, \gamma) := \cos \gamma \beta(s, \mathbf{x}) + \sin \gamma \mathbf{e}(s, \mathbf{x})$ and $\mathbf{e}(s, \mathbf{x}) := \beta(s, \mathbf{x}) \times \mathbf{u}(s, \mathbf{x})$.

It is clear from Eq. (9) that $s_2(s, \mathbf{x})$ actually depends on only s and $\beta(s, \mathbf{x})$. Therefore, we have [4]

$$\mathbf{u}(s, \beta) := \mathbf{u}(s, s_2(s, \beta)), \quad \mathbf{e}(s, \beta) := \beta \times \mathbf{u}(s, \beta), \quad (11)$$

$$\Psi(s, \beta) := \int_0^{2\pi} \frac{\partial}{\partial q} D_f(\mathbf{y}(q), \cos \gamma \beta + \sin \gamma \mathbf{e}(s, \beta))|_{q=s} \frac{1}{\sin \gamma} d\gamma, \quad (12)$$

$$f(\mathbf{x}) := -\frac{1}{2\pi^2} \int_{\mathbf{I}_{PI}(\mathbf{x})} \frac{1}{|\mathbf{x} - \mathbf{y}(s)|} \Psi(s, \beta(s, \mathbf{x})) ds. \quad (13)$$

Fix $s_2 \in [s - 2\pi + \Delta, s + 2\pi - \Delta]$, $s_2 \neq s$, $\Delta = 2 \cos^{-1}(r/R)$, and let $\Pi(s_2)$ denote the plane through $\mathbf{y}(s)$, $\mathbf{y}(s_2)$ and $\mathbf{y}(s_1(s, s_2))$. If $s = s_2$, $\Pi(s_2)$ is determined by continuity and coincides with the plane through $\mathbf{y}(s)$ and parallel to $\dot{\mathbf{y}}(s)$ and $\ddot{\mathbf{y}}(s)$. Then, we have a family of lines $L(s_2)$ obtained by intersecting $\Pi(s_2)$ with the detector plane. By construction, for a given $\mathbf{x} \in \mathbf{U}$ with $\beta(s, \mathbf{x})$ contained in $\Pi(s_2)$, s_2 thus determined is the same as that found by solving Eq. (9).

Since $\mathbf{e}(s, \beta) \bullet \beta = 0$, $|\mathbf{e}(s, \beta)| = 1$, we have

$$\beta = (\cos \theta, \sin \theta), \mathbf{e}(s, \beta) = (-\sin \theta, \cos \theta), \beta, \mathbf{e}(s, \beta) \in \Pi(s_2), \quad (14)$$

$$\Psi(s, \beta) := \int_0^{2\pi} \frac{\partial}{\partial q} D_f(\mathbf{y}(q), (\cos(\theta + \gamma), \sin(\theta + \gamma)))|_{q=s} \frac{1}{\sin \gamma} d\gamma, \beta \in \Pi(s_2). \quad (15)$$

This implies that the algorithm is in a filtered-backprojection format.

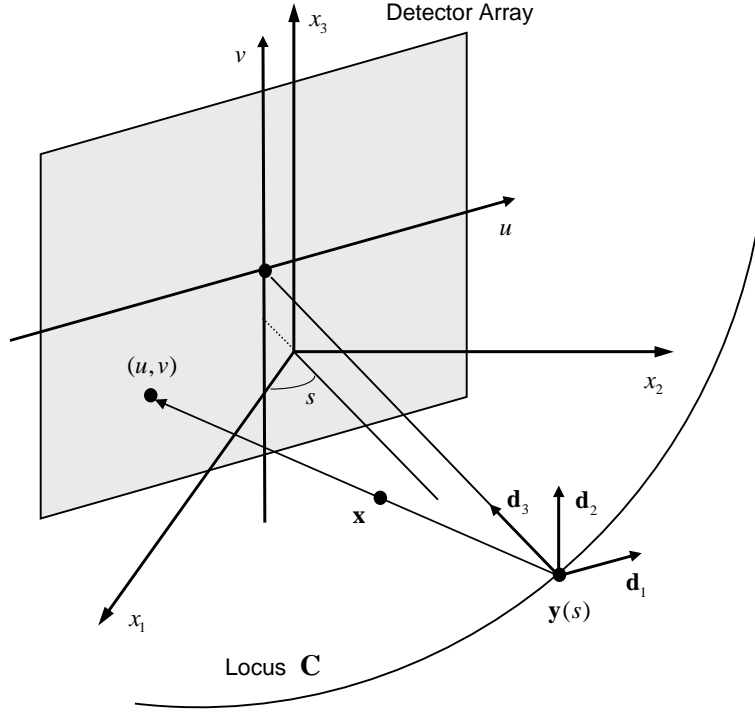


Fig. 2. Local coordinate system for cone-beam projection measurement on a planar detector.

3. Formulation for equispatal data

As shown in Fig. 2 let us define

$$\mathbf{d}_1 := (-\sin(s), \cos(s), 0) \quad (16a)$$

$$\mathbf{d}_2 := (0, 0, 1) \quad (16b)$$

$$\mathbf{d}_3 := (-\cos(s), -\sin(s), 0) \quad (16c)$$

A local coordinate system is formed with cone-beam data measured using a planar detector array parallel to \mathbf{d}_1 and \mathbf{d}_2 at a distance D from $\mathbf{y}(s)$. A detector position in the array is given by a pair of values (u, v) , which are signed distances along \mathbf{d}_1 and \mathbf{d}_2 respectively. Let $(u, v) = (0, 0)$ correspond to the orthogonal projection of $\mathbf{y}(s)$ onto the detector array. If s and D are given, (u, v) are determined by β . Since D is constant, the equispatal cone-beam projection $g(s, u, v) := D_f(\mathbf{y}, \beta)$. In this local coordinate system, the Katsevich algorithm is implemented in the three steps: (1) the first order derivatives of cone beam data are computed with respect to s while the orientations of involved X-rays are the same in the global coordinate system \mathbf{x} , (2) the shift-invariant filtering of the first derivative data is performed along a $L(s_2)$ line on the detector plane according to Eq. (12), and (3) the filtered data are backprojected in 3D according to Eq. (13).

3.1. Derivatives of cone-beam data

For a given $\mathbf{x} \in \mathbf{U}$, its projection position in the local coordinate system is expressed as:

$$u = \frac{D(\mathbf{x} - \mathbf{y}(s)) \bullet \mathbf{d}_1}{(\mathbf{x} - \mathbf{y}(s)) \bullet \mathbf{d}_3} \quad (17a)$$

$$\nu = \frac{D(\mathbf{x} - \mathbf{y}(s)) \bullet \mathbf{d}_2}{(\mathbf{x} - \mathbf{y}(s)) \bullet \mathbf{d}_3} \quad (17b)$$

Using the chain rule, the derivative of cone-beam data with respect to s is:

$$\frac{d}{ds}g(s, u, \nu) = \frac{\partial g}{\partial s} + \frac{\partial g}{\partial u} \frac{\partial u}{\partial s} + \frac{\partial g}{\partial \nu} \frac{\partial \nu}{\partial s} \quad (18)$$

with

$$\frac{\partial u}{\partial s} = \frac{D^2 + u^2}{D}, \quad (19a)$$

$$\frac{\partial \nu}{\partial s} = \frac{u\nu}{D}. \quad (19b)$$

The derivation of the formula Eq. (19) is in Appendix A.

Denote $D_g(s, u, \nu) = \frac{d}{ds}g(s, u, \nu)$, we have

$$D_g(s, u, \nu) = \left(\frac{\partial}{\partial s} + \frac{D^2 + u^2}{D} \frac{\partial}{\partial u} + \frac{u\nu}{D} \frac{\partial}{\partial \nu} \right) g(s, u, \nu). \quad (20)$$

3.2. Filtration of derivative data

Let $\psi(s) = s/2$, we have

$$s_1 = (s + s_2)/2, s_2 \in [s - 2\pi + \Delta, s + 2\pi - \Delta], s_1 \in [s - \pi + 0.5\Delta, s + \pi - 0.5\Delta] \quad (21)$$

For a given s , the equation for $L(s_2)$, also denoted as $L(s_1)$, in the detector plane is derived in Appendix B as follows:

$$\nu = \frac{Dh(s_1 - s)}{2\pi R} + \frac{h(s_1 - s)\text{ctg}(s_1 - s)}{2\pi R}u. \quad (22)$$

Figure 3 illustrates a family of $L(s_1)$. As shown in Fig. 4 and Appendix C, let $(\tilde{u}, \tilde{\nu})$ denote local coordinates of a variable point on the line $L(s_1)$ determined by (u, ν) , we have

$$\frac{d\gamma}{\sin \gamma} = \frac{\sqrt{D^2 + u^2 + \nu^2} d\tilde{u}}{(\tilde{u} - u)\sqrt{D^2 + \tilde{u}^2 + \tilde{\nu}^2}}. \quad (23)$$

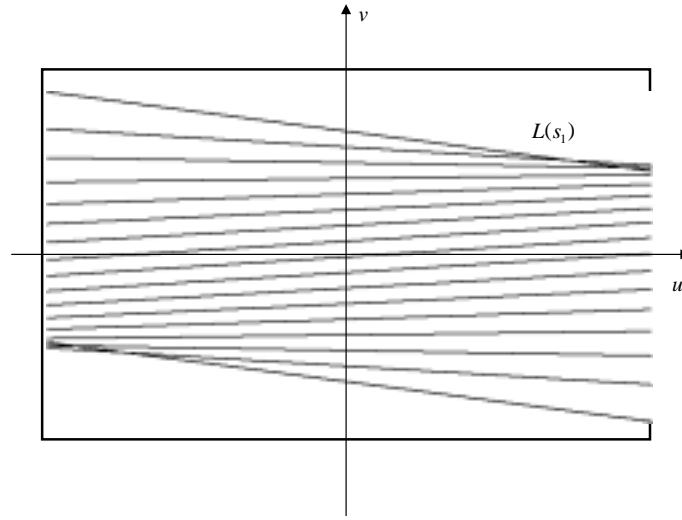
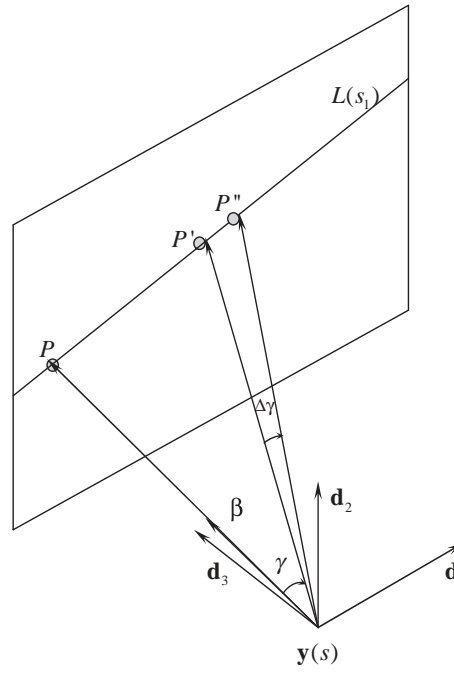
Fig. 3. Illustration of a family of $L(s_1)$.

Fig. 4. Proof of Eq. (23).

Therefore, Eq. (15) is simplified as:

$$\Psi(s, u, \nu) = \int_{-\infty}^{\infty} D_g(s, \tilde{u}, \tilde{\nu}) \frac{\sqrt{D^2 + u^2 + \nu^2}}{\sqrt{D^2 + \tilde{u}^2 + \tilde{\nu}^2}(\tilde{u} - u)} d\tilde{u}, \quad (24)$$

which is implemented in the following three steps:

$$\Psi^{(1)}(s, u, \nu) = D_g(s, u, \nu) \frac{D}{\sqrt{D^2 + u^2 + \nu^2}}, \quad (25a)$$

$$\Psi^{(2)}(s, u, \nu) = \int_{-\infty}^{\infty} \Psi^{(1)}(s, \tilde{u}, \tilde{\nu}) \frac{1}{(\tilde{u} - u)} d\tilde{u}, \quad (25b)$$

$$\Psi^{(3)}(s, u, \nu) = \frac{\sqrt{D^2 + u^2 + \nu^2}}{D} \Psi^{(2)}(s, u, \nu). \quad (25c)$$

Clearly, Eq. (25a) performs a pre-weighting operation on the first order derivatives of cone-beam data, while Eq. (25b) can be regarded as a Hilbert filtering along the line $L(s_1)$, and Eq. (25c) a post-weighting procedure.

3.3. Backprojection of filtered data

The backprojection operation according to Eq. (13) is essentially the same as that in the well known Feldkamp-type reconstruction. The main issue is to determine $\mathbf{I}_{IP(\mathbf{x})}$, that is, $s_b(\mathbf{x})$ and $s_t(\mathbf{x})$. For any given $\mathbf{x}_0 = (x_{10}, x_{20}, x_{30}) \in \mathbf{U}$, there must be a unique triplet s_b, s_t, t with $t \in (0, 1)$ and $0 < s_t - s_b < 2\pi$ such that

$$x_{10} = Rt \cos(s_b) + R(1 - t) \cos(s_t), \quad (26a)$$

$$x_{20} = Rt \sin(s_b) + R(1 - t) \sin(s_t), \quad (26b)$$

$$x_{30} = \frac{h}{2\pi} t s_b + \frac{h}{2\pi} (1 - t) s_t. \quad (26c)$$

First, we consider a family of PI lines intersecting the line \mathbf{V}_0 that contains \mathbf{x}_0 and is parallel to the x_3 axis [6]. This family is parameterized by s_b . Using the notation $\mathbf{x}_0 = (r_0 \cos(\xi), r_0 \sin(\xi), x_{30})$ and solving the first two equations in Eq. (26), we have

$$t(x_{10}, x_{20}, s_b) = \frac{R^2 - r_0^2}{2R[R - r_0 \cos(\xi - s_b)]}, \quad (27a)$$

$$s_t(x_{10}, x_{20}, s_b) - s_b = 2 \cos^{-1} \left(\frac{r_0 \sin(\xi - s_b)}{\sqrt{R^2 + r_0^2 - 2Rr_0 \cos(\xi - s_b)}} \right). \quad (27b)$$

Each PI line in the family intersects the vertical line \mathbf{V}_0 through a point (x_{10}, x_{20}, x_3) , where

$$x_3(x_{10}, x_{20}, s_b) = \frac{h}{2\pi} s_b t(x_{10}, x_{20}, s_b) + \frac{h}{2\pi} s_t(x_{10}, x_{20}, s_b) [1 - t(x_{10}, x_{20}, s_b)]. \quad (28)$$

It was shown [6] that $\frac{dx_3}{ds_b} > 0$. Hence, there must be a unique s_b which satisfies $x_3(x_{10}, x_{20}, s_b) = x_{30}$. Denote $s_0 = \frac{2\pi x_{30}}{h}$, by Eq. (26c) we have

$$(s_t - s_0) = \frac{t}{t-1}(s_b - s_0). \quad (29)$$

Substitute Eq. (29) into (27b), we obtain

$$s_b - s_0 = 2(t-1) \cos^{-1} \left(\frac{r_0 \sin(\xi - s_b)}{\sqrt{R^2 + r_0^2 - 2Rr_0 \cos(\xi - s_b)}} \right). \quad (30)$$

With $\mu_0 = \frac{r_0}{R}$, $0 \leq \mu_0 < 1$, the following inequalities can be readily verified:

$$\cos^{-1}(\mu_0) \leq \cos^{-1} \left(\frac{r_0 \sin(\xi - s_b)}{\sqrt{R^2 + r_0^2 - 2Rr_0 \cos(\xi - s_b)}} \right) \leq \pi - \cos^{-1}(\mu_0), \quad (31)$$

$$\frac{-\mu_0 - 1}{2} \leq t - 1 \leq \frac{\mu_0 - 1}{2} \quad (32)$$

Hence,

$$s_0 - (\pi - \cos^{-1}(\mu_0))(1 + \mu_0) \leq s_b \leq s_0 - \cos^{-1}(\mu_0)(1 - \mu_0). \quad (33)$$

Using the monotony of $x_3(x_{10}, x_{20}, s_b)$ and dichotomy, it is straightforward to search a unique s_b satisfying $x_3(x_{10}, x_{20}, s_b) = x_{30}$ in the interval

$$[s_0 - (\pi - \cos^{-1}(\mu_0))(1 + \mu_0), s_0 - \cos^{-1}(\mu_0)(1 - \mu_0)].$$

4. Implementation for equispacial data

First, the cone beam projections are uniformly sampled with sampling intervals $\Delta s, \Delta u, \Delta \nu$ for s, u and ν as follows:

$$g(s_k, u_m, \nu_n) \quad 0 \leq k < K, |m| \leq M, |n| \leq N, \quad (34)$$

where k, m and n are the indexes of sampling points for s, u and ν , respectively.

4.1. First order derivatives of cone-beam data

Since in the general case there are noise and errors in $g(s, u, \nu)$, to compute its derivatives a smooth convolution is performed with a gauss function

$$G(s, u, \nu) = \frac{1}{(\sqrt{2\pi})^3 \sigma_s \sigma_u \sigma_\nu} \exp \left(-\frac{s^2}{2\sigma_s^2} - \frac{u^2}{2\sigma_u^2} - \frac{\nu^2}{2\sigma_\nu^2} \right). \quad (35)$$

where σ_s , σ_u , and σ_ν are standard deviations serving as smoothing factors for the three dimensions, respectively. The partial derivatives of $g(s, u, \nu)$ are computed as following:

$$\frac{\partial}{\partial s}g(s, u, \nu) \approx \frac{\partial}{\partial s}g(s, u, \nu) * G(s, u, \nu) = g(s, u, \nu) * \frac{\partial}{\partial s}G(s, u, \nu), \quad (36a)$$

$$\frac{\partial}{\partial u}g(s, u, \nu) \approx \frac{\partial}{\partial u}g(s, u, \nu) * G(s, u, \nu) = g(s, u, \nu) * \frac{\partial}{\partial u}G(s, u, \nu), \quad (36b)$$

$$\frac{\partial}{\partial \nu}g(s, u, \nu) \approx \frac{\partial}{\partial \nu}g(s, u, \nu) * G(s, u, \nu) = g(s, u, \nu) * \frac{\partial}{\partial \nu}G(s, u, \nu), \quad (36c)$$

where the star * signifies a convolution operation. Using $3 \times 1 \times 1$, $1 \times 3 \times 1$ and $1 \times 1 \times 3$ windows to capture effects of σ_s , σ_u and σ_ν respectively, it results in the special 2-points difference formulae

$$D_g^s(s_k, u_m, \nu_n) \approx \frac{g(s_{k+1}, u_m, \nu_n) - g(s_{k-1}, u_m, \nu_n)}{2\Delta s} \quad (37a)$$

$$D_g^u(s_k, u_m, \nu_n) \approx \frac{g(s_k, u_{m+1}, \nu_n) - g(s_k, u_{m-1}, \nu_n)}{2\Delta u} \quad (37b)$$

$$D_g^\nu(s_k, u_m, \nu_n) \approx \frac{g(s_k, u_m, \nu_{n+1}) - g(s_k, u_m, \nu_{n-1})}{2\Delta \nu} \quad (37c)$$

Then, Eq. (20) is discretized as:

$$D_g(s_k, u_m, \nu_n) = D_g^s(s_k, u_m, \nu_n) + \frac{D^2 + u_m^2}{D} D_g^u(s_k, u_m, \nu_n) + \frac{u_m \nu_n}{D} D_g^\nu(s_k, u_m, \nu_n), \quad (38)$$

If we denote $s_{k+0.5} = (s_k + s_{k+1})/2$, $u_{m+0.5} = (u_m + u_{m+1})/2$, and $\nu_{n+0.5} = (\nu_n + \nu_{n+1})/2$, and use $2 \times 2 \times 2$ windows, we have an alternative form for Eq. (38):

$$\begin{aligned} D_g(s_{k+0.5}, u_{m+0.5}, \nu_{n+0.5}) &= D_g^s(s_{k+0.5}, u_{m+0.5}, \nu_{n+0.5}) \\ &+ \frac{D^2 + u_{m+0.5}^2}{D} D_g^u(s_{k+0.5}, u_{m+0.5}, \nu_{n+0.5}) \\ &+ \frac{u_{m+0.5} \nu_{n+0.5}}{D} D_g^\nu(s_{k+0.5}, u_{m+0.5}, \nu_{n+0.5}) \end{aligned} \quad (39)$$

where 8-points difference formulae are assumed:

$$D_g^s(s_{k+0.5}, u_{m+0.5}, \nu_{n+0.5}) \approx \frac{\sum_{m'=0}^1 \sum_{n'=0}^1 g(s_{k+1}, u_{m+m'}, \nu_{n+n'}) - \sum_{m'=0}^1 \sum_{n'=0}^1 g(s_k, u_{m+m'}, \nu_{n+n'})}{4\Delta s} \quad (40a)$$

$$D_g^u(s_{k+0.5}, u_{m+0.5}, \nu_{n+0.5}) \approx \frac{\sum_{k'=0}^1 \sum_{n'=0}^1 g(s_{k+k'}, u_{m+1}, \nu_{n+n'}) - \sum_{k'=0}^1 \sum_{n'=0}^1 g(s_{k+k'}, u_m, \nu_{n+n'})}{4\Delta u} \quad (40b)$$

$$D_g^\nu(s_{k+0.5}, u_{m+0.5}, \nu_{n+0.5}) \approx \frac{\sum_{k'=0}^1 \sum_{m'=0}^1 g(s_{k+k'}, u_{m+m'}, \nu_{n+1}) - \sum_{k'=0}^1 \sum_{m'=0}^1 g(s_{k+k'}, u_{m+m'}, \nu_n)}{4\Delta \nu} \quad (40c)$$

Evidently, with various combinations of $\sigma_s, \sigma_u, \sigma_\nu$ values we can obtain more complex formulae to compute the derivatives of $g(s, u, \nu)$.

4.2. Filtration of derivative data

This step is more complicated than other steps. According to Eq. (21), for any $s \in \mathbb{R}$, $s_1 - s \in [-\pi + 0.5\Delta, \pi - 0.5\Delta]$ with $\Delta = 2\cos^{-1}(r/R)$. Thus, for all s_k , the interval $[-\pi + 0.5\Delta, \pi - 0.5\Delta]$ is uniformly sampled at $s_q, |q| \leq Q$. Using Eq. (22), the following matrix is calculated:

$$\nu_{q,m} = \frac{Dhs_q}{2\pi R} + \frac{hs_q \text{ctg}(s_q)}{2\pi R} u_m, \quad (41)$$

where the only exception is that $\nu_{0,m} = \frac{h}{2\pi R} u_m \cdot D_g(s_k, s_q, u_m, \nu_{q,m})$ is obtained by re-sampling $D_g(s_k, u_m, \nu_n)$ via linear interpolation along the direction of ν . After that, the filtering processes are executed as follows:

$$\Psi^{(1)}(s_k, s_q, u_m, \nu_{q,m}) = D_g(s_k, s_q, u_m, \nu_{q,m}) \frac{D}{\sqrt{D^2 + u_m^2 + \nu_{q,m}^2}}, \quad (42a)$$

$$\begin{aligned} \Psi^{(2)}(s_k, s_q, u_m, \nu_{q,m}) &\approx \sum_{m'=-M}^{m-1} \frac{\Psi^{(1)}(s_k, s_q, u_{m'}, \nu_{q,m'}) \Delta u}{(u_{m'} - u_m)} \\ &+ \sum_{m'=m+1}^M \frac{\Psi^{(1)}(s_k, s_q, u_{m'}, \nu_{q,m'}) \Delta u}{(u_{m'} - u_m)}, \end{aligned} \quad (42b)$$

$$\Psi^{(3)}(s_k, s_q, u_m, \nu_{q,m}) = \frac{\sqrt{D^2 + u_m^2 + \nu_{q,m}^2}}{D} \Psi^{(2)}(s_k, s_q, u_m, \nu_{q,m}). \quad (42c)$$

Because Formula Eq. (42b) is a Hilbert transform, it can be implemented using FFT (see appendix D). Once again, $\Psi(s_k, u_m, \nu_n)$ is obtained by re-sampling $\Psi^{(3)}(s_k, s_q, u_m, \nu_{q,m})$. Given s_k, u_m, ν_n , we search the maximal q denoted as $q_{d\max}$ satisfying $\nu_{q_{d\max},m} < \nu_n$. Similarly, we find the minimal q denoted as $q_{u\min}$ satisfying $\nu_{q_{u\min},m} > \nu_n$. Then, $\psi(s_k, u_m, \nu_n)$ is estimated by interpolating between $\Psi^{(3)}(s_k, s_{q_{d\max}}, u_m, \nu_{q_{d\max},m})$ and $\Psi^{(3)}(s_k, s_{q_{u\min}}, u_m, \nu_{q_{u\min},m})$ linearly.

4.3. Backprojection of filtered data

For a given \mathbf{x} , $\mathbf{I}_{PI}(\mathbf{x})$ is determined in terms of $s_b(\mathbf{x})$ and $s_t(\mathbf{x})$. At the same time, k_b and k_t are determined to satisfy that $s_{(k_b-1)} < s_b(\mathbf{x}) < s_{k_b}$ and $s_{k_t} < s_t(\mathbf{x}) < s_{(k_t+1)}$. Then, the backprojection step is implemented as follows:

$$f(\mathbf{x}) = -\frac{\Delta s}{2\pi^2} \sum_{k=k_b}^{k=k_t} \frac{\psi(s_k, \tilde{u}(s_k, \mathbf{x}), \tilde{v}(s_k, \mathbf{x}))}{|\mathbf{x} - \mathbf{y}(s_k)|}, \quad (43)$$

where

$$\tilde{u}(s_k, \mathbf{x}) = \frac{D(\mathbf{x} - \mathbf{y}(s_k)) \bullet \mathbf{d}_1}{(\mathbf{x} - \mathbf{y}(s_k)) \bullet \mathbf{d}_3}, \quad (44a)$$

$$\tilde{v}(s_k, \mathbf{x}) = \frac{D(\mathbf{x} - \mathbf{y}(s_k)) \bullet \mathbf{d}_2}{(\mathbf{x} - \mathbf{y}(s_k)) \bullet \mathbf{d}_3} \quad (44b)$$

$\psi(s_k, \tilde{u}(s_k, \mathbf{x}), \tilde{v}(s_k, \mathbf{x}))$ is estimated by bilinear interpolation. To reduce artifacts due to the endpoint effects, we treat $s_b(\mathbf{x})$ as the interpolation between $s_{(k_b-1)}$ and s_{k_b} , and $s_t(\mathbf{x})$ as the interpolation between $s_{(k_t+1)}$ and s_{k_t} . Then, formula Eq. (43) is refined as:

$$f(\mathbf{x}) = -\frac{\Delta s}{2\pi^2} \sum_{k=k_b-1}^{k=k_t+1} \frac{\psi(s_k, \tilde{u}(s_k, \mathbf{x}), \tilde{v}(s_k, \mathbf{x}))\lambda(k)}{|\mathbf{x} - \mathbf{y}(s_k)|}, \quad (45)$$

with the coefficients

$$\lambda(k) = \begin{cases} 0.5\lambda_b^2 & k = k_b - 1 \\ 0.5 + \lambda_b - 0.5\lambda_b^2 & k = k_b \\ 1 & k_b + 1 \leq k \leq k_t - 1, \\ 0.5 + \lambda_t - 0.5\lambda_t^2 & k = k_t \\ 0.5\lambda_t^2 & k = k_t + 1 \end{cases} \quad (46)$$

$$\lambda_b = \frac{s_{k_b} - s_b(\mathbf{x})}{\Delta s}. \quad (47a)$$

$$\lambda_t = \frac{s_t(\mathbf{x}) - s_{k_t}}{\Delta s} \quad (47b)$$

4.4. Selection of parameters

In the above describing implementation of the Katsevich algorithm, there are four basic discretization parameters, which are K, M, N, Q . K is the number of sampling points in s . An appropriate K is selected based on the requirement on the desired resolution and the length of the interval of s . In the

Table 1
Cone-beam imaging parameters used in the numerical simulation

Scanning radius (R)	75 cm
Source to detector distance (D)	150 cm
Helical pitch (h)	25 cm
Object radius (r)	25 cm
Number of projections per turn	512
Scanning range ($K = 1536$)	$s \in [-3\pi, 3\pi]$
Detector size ($M = 136, N = 45$)	273×91
Detector width (Δu)	0.391 cm
Detector height ($\Delta \nu$)	0.391 cm
Number of lines $L(s_1)$ per projection ($Q = 64$)	129
Reconstruction matrix	$257 \times 257 \times 257$

local coordinate system shown in Fig. 2, one can determine the maximum value of $|u|_{\max}$ and $|\nu|_{\max}$ of the Tam-Danielson window from formula Eq. (17).

$$|u|_{\max} = \frac{D \sin \Delta}{1 - \cos \Delta} \quad (48a)$$

$$|\nu|_{\max} = \frac{Dh(2\pi - \Delta)}{2\pi R(1 - \cos \Delta)} \quad (48b)$$

where $\Delta = 2 \cos^{-1}(r/R)$. Similarly, a proper Δ_u and M covering the field of view $|u|_{\max} = M \cdot \Delta_u$ are determined. Assume $\Delta \nu = \Delta u$ and $|\nu|_{\max} = N \cdot \Delta \nu$, N is determined to be consistent with the involved detection window.

$$N = \left\lceil \frac{h}{2\pi R} \frac{2\pi - \Delta}{\sin \Delta} M \right\rceil \quad (49)$$

Since $L(s_1)$ varies substantially as a function of s_1 , based on the simulation results reported in the following section we suggest to use $0.6 N < Q < 1.5 N$ to achieve the desired resolution. Considering the local derivative operation, M and N may be slightly larger than the above theoretical values.

5. Simulation results

Our algorithmic steps for implementing the Katsevich algorithm were numerically evaluated using the 3D Shepp-Logan phantom [7]. As shown in Fig. 2, cone-beam spiral data were collected with a planar detector array in the simulation. In the following, unless otherwise stated, the parameters should remain the same as in Table 1, Formulae Eqs (38) and (45) were used for image reconstruction, and the Hilbert transform was performed in the Fourier domain using a rectangular apodization window.

To demonstrate the effect of artifact reduction using Eq. (45), Fig. 5 gives the reconstructed slices using Eqs (43) and (45) respectively. It can be observed in Fig. 5 that the range of the PI segment is very important to the reconstruction quality. Specifically, significant artifacts were caused if the endpoints of the PI segment were not treated appropriately, i.e., not interpolated continuously.

There are several ways to compute the derivatives of $g(s, u, \nu)$. Fig. 6 compares the reconstruction results using Eqs (38) and (39) respectively. It can be seen in Fig. 6 that Eq. (38) performs better than

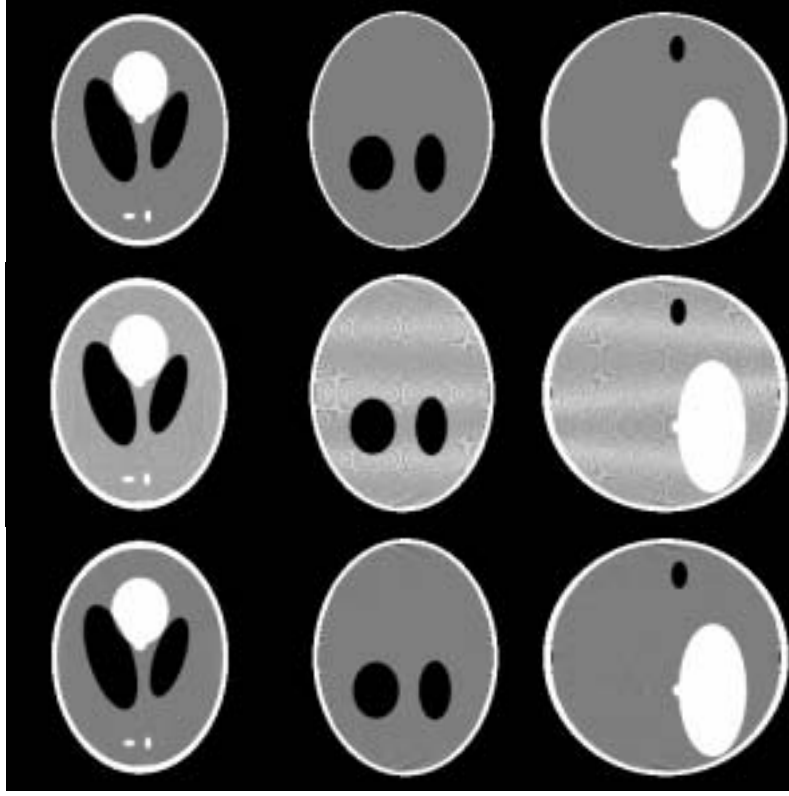


Fig. 5. Reconstruction results of different backprojection formulae. The first row are the original slices of 3D head phantom, while the second and the third are the reconstruction results by formulae Eqs (43) and (45), respectively. The left column are the slices at $z = -6.25$ cm, the middle column are the slices at $y = 0$ cm, and the right column are the slices at $x = 0$ cm. Gray range $[1.01, 1.03]$ is linear mapped into $[0, 255]$.

Table 2
PSNR of the slice $z = -6.25$ cm with respect to Q ($N = 45$)

Q	5	10	15	20	25	30	35	50
PSNR(dB)	23.854	25.451	25.849	26.020	26.078	26.135	26.165	26.179
Q	45	50	55	60	64	65	70	75
PSNR(dB)	26.189	26.195	26.215	26.216	26.226	26.222	26.226	26.230

Eq. (39). A major reason is that Eq. (38) may have better captured partial derivative information than Eq. (39), since Eq. (38) involves a larger range along the direction of interest.

A common mistake is that the partial derivatives of u and v with respect to s are not needed. In fact, (u, v) define a local detection coordinates and are associated with s by the unit vectors $\mathbf{d}_1, \mathbf{d}_2$ and \mathbf{d}_3 , as shown in Eqs (16) and (17). To illustrate the errors due to this misunderstanding, Fig. 7 presents the reconstruction results when the partial derivative of u and v were omitted.

Undoubtedly, the reconstruction precision depends on the selection of parameter Q . We know that Q may be smaller than Q_m , where Q_m satisfies that the maximum distance between adjacent lines $L(s_1)$ along direction \mathbf{d}_2 is small than $\Delta\nu$. For the parameters in Table 1, we found that $Q_m = 73$. To demonstrate the effect of Q on image quality, we reconstructed the 3D phantom with different values of Q . Table 2 shows the peak-signal-noise-ratio (PSNR) of the reconstructed slice $z = -6.25$ cm with

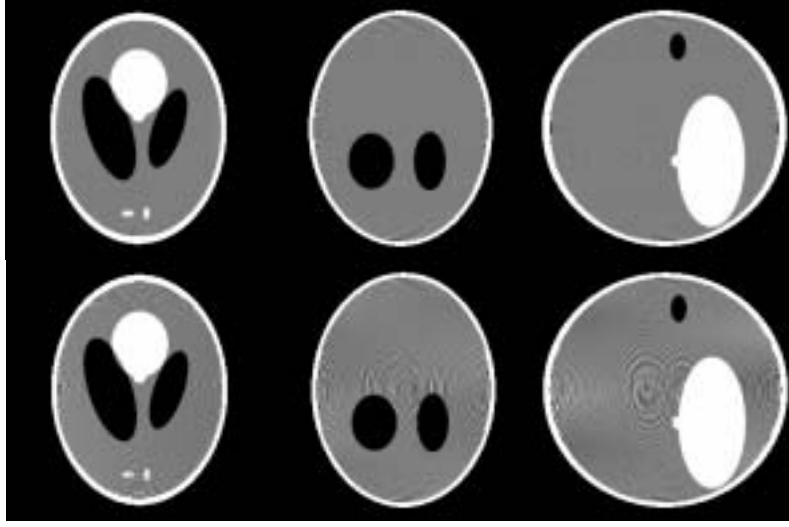


Fig. 6. Reconstruction results of different derivative formulae. The first row are reconstructed by formula Eq. (38), while the second row are reconstructed by formula Eq. (39). The left column are the slices at $z = -6.25$ cm, the middle column are the slices at $y = 0$ cm, and the right column are the slices at $x = 0$ cm. Gray range $[1.01, 1.03]$ is linear mapped into $[0, 255]$.

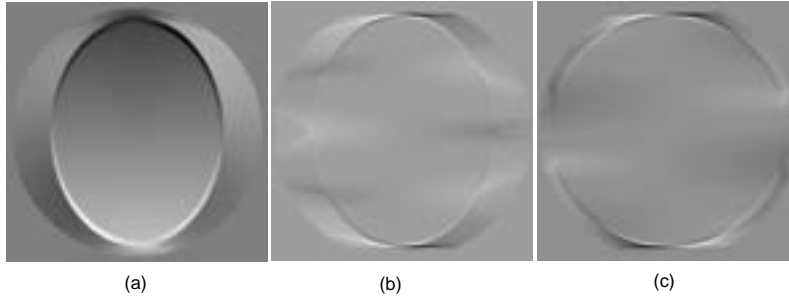


Fig. 7. Reconstruction results when the partial derivatives of u and v are omitted. From left to right are the reconstructed slices at $z = -6.25$ cm, $y = 0$ cm and $x = 0$ cm, respectively. Gray range $[0, 1]$ is adapted linear mapped into $[0, 255]$.

different values of Q . It can be seen in Table 2 that the reconstruction quality is insensitive to the variation in Q . When Q varies from 25 to 75, PSNR only increases 0.15 dB, and we cannot visually distinguish the differences among the reconstructions, as shown in Fig. 8. Based on our experience, we suggest that it $0.6 N < Q < 1.5 N$.

Formula Eq. (42b) can be regarded as a Hilbert transform, hence it can be implemented in the Fourier domain. Since the impulse response of the Hilbert transform is of an infinite length, we need use a window function of a finite length. Typical window functions include rectangular, Hamming, Hanning, Blackman, and Kaiser windows [8]. Fig. 9 gives the reconstructed slices using different window functions. It can be concluded that the rectangular window is the optimal in terms of the image quality.

6. Discussions and conclusion

While preparing this manuscript, it came to our attention that Noo, Pack and Heuscher published their procedures for exact spiral cone-beam reconstruction using native cone-beam geometries [9]. Since

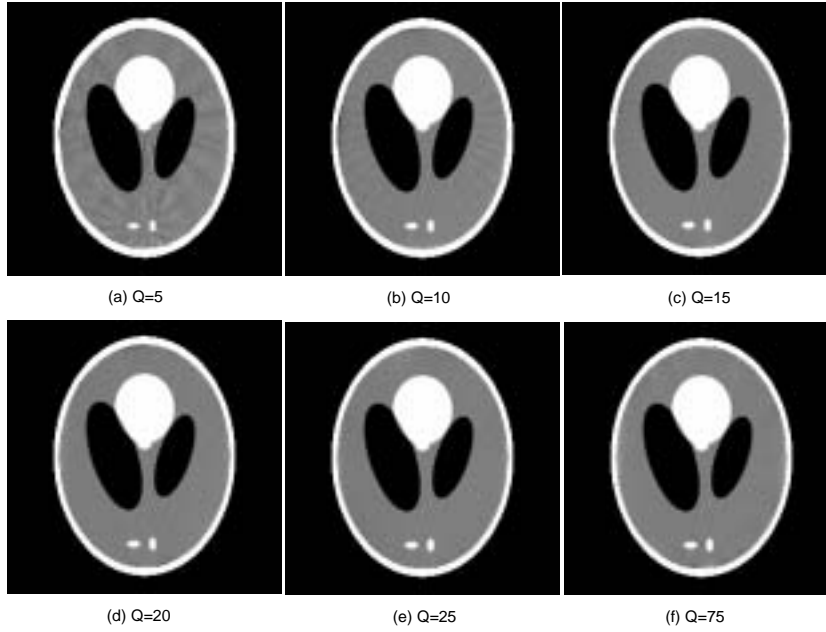


Fig. 8. Reconstructed results of slice $z = -6.25$ cm for different Q . Gray range $[1.015, 1.025]$ is linear mapped into $[0, 255]$.

both their work and ours are about the implementation of the Katsevich algorithm, there are substantial similarities in the two papers. However, we would like to emphasize that there are also significant differences between them, which make them actually complement each other. Specifically, our paper differs from that by Noo, Pack and Heuscher in the following aspects. Primarily, our implementation steps are not the same as theirs. Our experimental results show that (1) the endpoints of the PI segment should be treated continuously or the reconstruction artifacts would be evident; (2) in the derivative operation, a 2-point difference formulae outperforms a 8-point counterpart; (3) the reconstruction quality is insensitive to Q around N ; for example, $0.6 N < Q < 1.5 N$; and (4) if the Hilbert transform is implemented in the Fourier domain the rectangular window function is the optimal in terms of the image quality. Also, we have analyzed image artifacts associated with inappropriate selections of the imaging parameters. Furthermore, we have decided to open our source codes in MatLab on “<http://dolphin.radiology.uiowa.edu/ge/CTL/Software/Katsevich>” as an Appendix to this paper, which makes the Katsevich algorithm complete transparent to the peers.

In conclusion, we have not only described how to program the Katsevich algorithm in the planar detector geometry but also studied effects of various implementation options on the image quality. Because the Katsevich-type spiral cone-beam reconstruction is quite complicated and has become an active new area, our results should be valuable for those who are interested in evaluating, improving and extending this new approach.

Acknowledgment

This work is partially supported by NIH/NIBIB grants (EB002667 and EB004287).

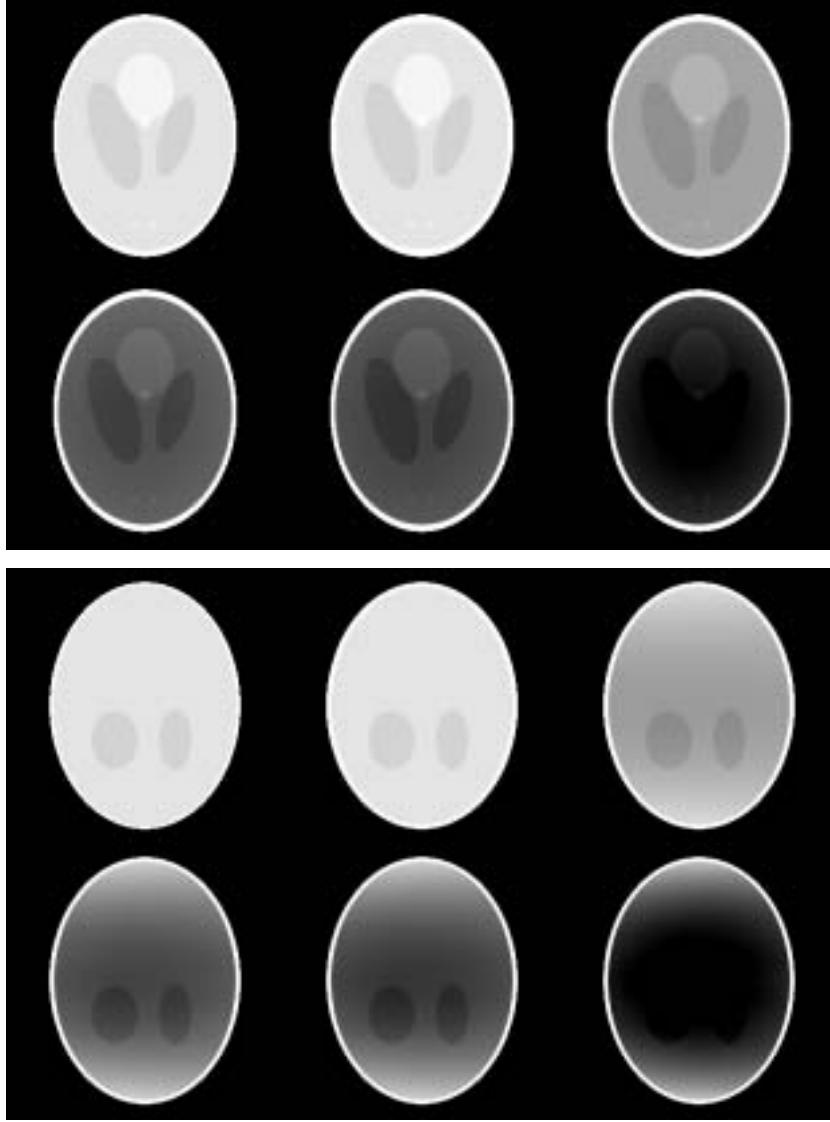


Fig. 9. Reconstruction results using different window function. The top are the slices at $z = -6.25$ cm and the bottom are the slices at $y = 0$ cm. From left to right, top to bottom, the first one is original slice, the others are reconstructed results with rectangular, Kaiser, Hamming, Hanning, Blackman window function, respectively. Gray range $[0.75, 1.05]$ is linear mapped into $[0, 255]$.

Appendix A. Derivative of formulae Eq. (19)

For a given $\beta \in S^2$, its projection position in the local coordinate system can be expressed as:

$$u = \frac{D\beta \bullet \mathbf{d}_1}{\beta \bullet \mathbf{d}_3}, \quad (\text{A1a})$$

$$\nu = \frac{D\beta \bullet \mathbf{d}_2}{\beta \bullet \mathbf{d}_3}, \quad (\text{A1b})$$

Hence, we have

$$\frac{\partial u}{\partial s} = \left(\frac{D\beta \bullet \mathbf{d}_1}{\beta \bullet \mathbf{d}_3} \right)' = \frac{D\beta \bullet \mathbf{d}'_1}{\beta \bullet \mathbf{d}_3} - \frac{D\beta \bullet \mathbf{d}_1 \beta \bullet \mathbf{d}'_3}{(\beta \bullet \mathbf{d}_3)^2}, \quad (\text{A2a})$$

$$\frac{\partial \nu}{\partial s} = \left(\frac{D\beta \bullet \mathbf{d}_2}{\beta \bullet \mathbf{d}_3} \right)' = \frac{D\beta \bullet \mathbf{d}'_2}{\beta \bullet \mathbf{d}_3} - \frac{D\beta \bullet \mathbf{d}_2 \beta \bullet \mathbf{d}'_3}{(\beta \bullet \mathbf{d}_3)^2}. \quad (\text{A2b})$$

Since $\mathbf{d}'_1 = \mathbf{d}_3$, $\mathbf{d}'_3 = -\mathbf{d}_1$, $\mathbf{d}'_2 = 0$, we obtain

$$\frac{\partial u}{\partial s} = \frac{D\beta \bullet \mathbf{d}_3}{\beta \bullet \mathbf{d}_3} + \frac{D(\beta \bullet \mathbf{d}_1)^2}{(\beta \bullet \mathbf{d}_3)^2}, \quad (\text{A3a})$$

$$\frac{\partial \nu}{\partial s} = \frac{D\beta \bullet \mathbf{d}_2 \beta \bullet \mathbf{d}_1}{(\beta \bullet \mathbf{d}_3)^2}. \quad (\text{A3b})$$

Using (A1), it follows readily that

$$\frac{\partial u}{\partial s} = \frac{D^2 + u^2}{D}, \quad (\text{A4a})$$

$$\frac{\partial \nu}{\partial s} = \frac{u\nu}{D}. \quad (\text{A4b})$$

Appendix B. Derivative of Eq. (22) for $L(s_1)$

Let $\psi(s) = s/2$ and $s = 0$ without loss of generality, we have

$$s_1 = s_2/2, s_2 \in [-2\pi + \Delta, 2\pi - \Delta], s_1 \in [-\pi + 0.5\Delta, \pi - 0.5\Delta]. \quad (\text{B1})$$

Hence $\mathbf{y}(s)$, $\mathbf{y}(s_2)$, and $\mathbf{ny}(s_1(s, s_2))$ can be expressed as:

$$\mathbf{y}(s) = (R, 0, 0), \quad (\text{B2a})$$

$$\mathbf{y}(s_1) = \left(R \cos(s_1), R \sin(s_1), \frac{hs_1}{2\pi} \right), \quad (\text{B2b})$$

$$\mathbf{y}(s_2) = \left(R \cos(2s_1), R \sin(2s_1), \frac{hs_1}{\pi} \right). \quad (\text{B2c})$$

Therefore, we have

$$(\mathbf{y}(s_1) - \mathbf{y}(s)) \times (\mathbf{y}(s_2) - \mathbf{y}(s)) = 2R(1 - \cos(s_1)) \left(\frac{hs_1}{2\pi} \sin(s_1), -\frac{hs_1}{2\pi} \cos(s_1), R \sin(s_1) \right). \quad (\text{B3})$$

In the case of $s_1 \neq 0$, the equation for the plane determined by $\mathbf{y}(s)$, $\mathbf{y}(s_2)$ and $\mathbf{y}(s_1(s, s_2))$ is given by

$$(\mathbf{x} - \mathbf{y}(s)) \cdot \left(\frac{hs_1}{2\pi} \sin(s_1), -\frac{hs_1}{2\pi} \cos(s_1), R \sin(s_1) \right) = 0. \quad (\text{B4})$$

Since an arbitrary point on the detector plane can be described by $(R - D, u, \nu)$, we have

$$\nu = \frac{Dhs_1}{2\pi R} + \frac{hs_1 \text{ctg}(s_1)}{2\pi R} u. \quad (\text{B5})$$

In the case of $s_1=0$, we have

$$\nu = \frac{h}{2\pi R} u. \quad (\text{B6})$$

If $s \neq 0$, (B5) becomes

$$\nu = \frac{Dh(s_1 - s)}{2\pi R} + \frac{h(s_1 - s) \text{ctg}(s_1 - s)}{2\pi R} u. \quad (\text{B7})$$

Appendix C. Derivation of Eq. (23)

Shown as in Fig. 4, for fixed s and β , we can find the intersection position P along the direction β in the detector plane with local coordinates (u, ν, D) . There exist s_1 and $L(s_1)$ containing P . Pick $P' \in L(s_1)$, whose coordinates can be written $(\tilde{u}, \tilde{\nu}, D)$. The angle between OP and OP' is denoted as γ . Then, we have

$$|\sin \gamma| = \frac{|\vec{OP} \times \vec{OP'}|}{|\vec{OP}| \cdot |\vec{OP'}|} = \frac{\sqrt{(\tilde{u} - u)^2 D^2 + (\nu - \tilde{\nu})^2 D^2 + (u\tilde{\nu} - \tilde{u}\nu)^2}}{|\vec{OP}| \cdot |\vec{OP'}|}. \quad (\text{C1})$$

Note that (u, ν) and $(\tilde{u}, \tilde{\nu})$ satisfy Eq. (22), we have

$$\sin \gamma = \frac{(\tilde{u} - u)DC_{s,s_1}}{|\vec{OP}| \cdot |\vec{OP'}|}, \quad (\text{C2})$$

where C_{s,s_1} is constant for fixed s and s_1 , and

$$C_{s,s_1} = \sqrt{1 + \left(\frac{h(s_1 - s)}{2\pi R \sin(s_1 - s)} \right)^2}. \quad (\text{C3})$$

Assume that $P'' \in L(s_1)$ is close to P' with coordinates $(\tilde{u}, \tilde{\nu})$, and the angle between OP' and OP'' is $\Delta\gamma$. Then, we have

$$\sin(\Delta\gamma) = \frac{(\tilde{u} - u)DC_{s,s_1}}{|\overrightarrow{OP'}| \bullet |\overrightarrow{OP''}|}. \quad (C4)$$

By (C4) and (C2), we have

$$\frac{\sin(\Delta\gamma)}{\sin \gamma} = \frac{(\tilde{u} - u)|\overrightarrow{OP}|}{(\tilde{u} - u)|\overrightarrow{OP'}|}. \quad (C5)$$

If $\Delta\gamma$ is so small that $\Delta\gamma \approx \sin(\Delta\gamma)$, we have

$$\frac{d\gamma}{\sin \gamma} = \frac{d\tilde{u}|\overrightarrow{OP}|}{(\tilde{u} - u)|\overrightarrow{OP'}|} = \frac{d\tilde{u}\sqrt{D^2 + u^2 + \nu^2}}{(\tilde{u} - u)\sqrt{D^2 + \tilde{u}^2 + \tilde{\nu}^2}}. \quad (C6)$$

Appendix D. Implementation of formula Eq. (42b) using FFT

In fact, formula Eq. (25b) can be rewritten as:

$$\Psi^{(2)}(s, u, \nu) = -\pi(\Psi^{(1)}(s, u, \nu) * H(u)) \quad (D1)$$

where star * signifies the convolution operator and $H(u)$ is the kernel function of Hilbert transform in continuous domain with the expression:

$$H(u) = \frac{1}{\pi u} \quad (D2)$$

If discretize (D1), one can get Eq. (42b) directly, that is

$$\Psi^{(2)}(s_k, s_q, u_m, \nu_{q,m}) \approx -\pi(\Psi^{(1)}(s_k, s_q, u_m, \nu_{q,m}) * h(m)) \quad (D3)$$

with discretized Hilbert transform kernel function:

$$h(m) = \begin{cases} \frac{1}{\pi m} & m \neq 0 \\ 0 & m = 0 \end{cases} \quad (D4)$$

Since (D4) has an infinite duration, which lead to non-realizable filter, we can truncate $h(m)$ by a window function $w(m)$ with finite length. Typical window functions include rectangular, Hamming, Hanning, Blackman, and Kaiser windows [8]. Denote $h'(m) = h(m)w(m)$, one can get:

$$\Psi^{(2)}(s_k, s_q, u_m, \nu_{q,m}) \approx -\pi(\Psi^{(1)}(s_k, s_q, u_m, \nu_{q,m}) * h'(m)) \quad (D5)$$

According to the theory of signal processing, (D5) can be computed by FFT in frequency domain. Considering the finite length of the digital Hilbert transform kernel function, (D4) is refined as:

$$\tilde{h}(m) = \begin{cases} \frac{2 \sin^2(\pi m/2)}{\pi m} & m \neq 0 \\ 0 & m = 0 \end{cases} \quad (\text{D6})$$

References

- [1] G. Wang, T.H. Lin, P.C. Cheng and D.M. Shinozaki, A general cone-beam reconstruction algorithm, *IEEE Transaction on Medical Imaging* **12**(3) (1993), 486–496.
- [2] H. Kudo, F. Noo and M. Defrise, Cone-beam filtered-backprojection algorithm for truncated data, *Physics in Medicine and Biology* **43** (1993), 2885–2909.
- [3] A. Katsevich, Theoretically exact FBP-type inversion algorithm for spiral CT, *SIAM J. Appl. Math.* **62**(6) (2002), 2012–2026.
- [4] A. Katsevich, An improved exact filtered backprojection algorithm for spiral computed tomography, *Advance in Applied Mathematics* **32** (2004), 681–697.
- [5] A. Katsevich, Analysis of an exact inversion algorithm for spiral cone-beam CT, *Phys. Med. Biol.* **47** (2002), 2583–2597.
- [6] M. Defrise, F. Noo and H. Kudo, A solution to the long-object problem in helical cone-beam tomography, *Phys. Med. Bio.* **45** (2000), 623–643.
- [7] G. Wang, T.H. Lin, P.C. Cheng and D.M. Shinozaki, A general cone-beam reconstruction algorithm, *IEEE Transaction on Medical Imaging* **12** (3) (1993), 486–496.
- [8] P.S.R. Diniz, E.A.B. da Silva and S.L. Netto, *Digital signal processing system analysis and design*, Publishing house of electronics industry, Beijing, 2002.
- [9] F. Noo, J. Pack and D. Heuscher, Exact helical reconstruction using native cone-beam Geometries, *Phys. Med. Biol.* **48** (2003), 3787–3818.