Exact helical reconstruction using native cone-beam geometries

Frédéric Noo[1]**, Jed Pack**[1] **and Dominic Heuscher**[2]

[1] Department of Radiology, University of Utah, Salt Lake City, UT 84112, USA
[2] Philips Medical Systems, Cleveland, OH, USA

E-mail: noo@doug.med.utah.edu

Abstract
This paper is about helical cone-beam reconstruction using the exact filtered backprojection formula recently suggested by Katsevich (2002a *Phys. Med. Biol.* **47** 2583–97). We investigate how to efficiently and accurately implement Katsevich's formula for direct reconstruction from helical cone-beam data measured in two native geometries. The first geometry is the curved detector geometry of third-generation multi-slice CT scanners, and the second geometry is the flat detector geometry of C-arms systems and of most industrial cone-beam CT scanners. For each of these two geometries, we determine processing steps to be applied to the measured data such that the final outcome is an implementation of the Katsevich formula. These steps are first described using continuous-form equations, disregarding the finite detector resolution and the source position sampling. Next, techniques are presented for implementation of these steps with finite data sampling. The performance of these techniques is illustrated for the curved detector geometry of third-generation CT scanners, with 32, 64 and 128 detector rows. In each case, resolution and noise measurements are given along with reconstructions of the FORBILD thorax phantom.

1. Introduction

Since the introduction of multi-slice CT scanners in 1998, helical cone-beam reconstruction has become an important subject of research in x-ray computed tomography (CT). Initially, the divergence of the beams could be neglected because there were only four detector rows (see, e.g., Taguchi and Aradate (1998) and Schaller *et al* (2000)). But, this situation was only of transitional nature: CT scanners with 16 rows appeared on the market in 2001, while industrial efforts are currently being made at developing scanners with 32 rows and more. With more than four detector rows, the divergence of the beams cannot be neglected for accurate imaging and dedicated cone-beam (CB) reconstruction algorithms must be considered.

Over the last six years, numerous algorithms have been suggested for helical cone-beam reconstruction. These algorithms can be listed into four classes: (i) rebinning algorithms, (ii) FDK-like algorithms, (iii) quasi-exact algorithms and (iv) exact algorithms.

The rebinning algorithms aim at estimating the 2D sinogram of oblique slices covering the region-of-interest (ROI). Once these sinograms have been estimated, the oblique slices can be reconstructed using any 2D reconstruction methods, and the ROI can be obtained subsequently by interpolation of the oblique slices. The quality of the reconstruction depends here mainly on the estimate of the sinograms. Estimates that seem satisfactory for 16-row scanners can be obtained using the oblique surface reconstruction algorithm (Heuscher 1999, Chen *et al* 2000) or the nutating slice algorithm (Larsson *et al* 1998), which is also called ASSR method in the literature (Kachelriess *et al* 2000). Alternatively, an exact estimate of the sinograms can be obtained using the PHI method of Schaller *et al* (2000), but the overscan and the computational and memory efforts are significantly more demanding in that case.

The FDK-like algorithms are extensions to the helix of the algorithm designed by Feldkamp, Davis and Kress for circular scans (Feldkamp *et al* 1984). These algorithms process the data in three steps: 1D ramp filtering, data weighting and finally CB backprojection. The data weighting is introduced to smooth out discontinuities at the end of the data range used to reconstruct a given slice. It neglects the beam divergence and acts pretty much the same way as the weighting functions used in spiral CT (see Crawford and King (1990)). Good reconstruction results have been reported for large numbers of detector rows (see, e.g., Turbell (1999) and Sourbelle (2002)). However, the existing FDK-like methods for the helix suffer one drawback: artefacts in the reconstruction cannot be easily predicted or detected due to a lack of theory supporting the approximations involved in the development of the algorithm.

The quasi-exact algorithms are discretizations of analytical reconstruction formulae developed from extensions of Grangeat's theory for cone-beam reconstruction using the 3D Radon transform. They include the zero-boundary method of Defrise *et al* (2000), the virtual-circle method of Kudo *et al* (2000) and implementation of the local-ROI method with Hilbert filtering (see Tam (2000)). These algorithms are not exact only because they process incorrectly a small number of Radon samples, so that the data can be processed sequentially as in the filtered-backprojection algorithms. The Radon samples that are processed wrong correspond to the planes that have three intersections with the helix and are such that the line joining the first intersection to the third intersection passes through the field-of-view (see Tam *et al* (2002) for more details). The quasi-exact algorithms have currently the disadvantage of being more sensitive to discretization errors and less computationally efficient than the rebinning and FDK-like methods.

Exact helical cone-beam reconstruction can be achieved using the PHI method of Schaller *et al* (2000), which was already mentioned, the four-step local ROI method of Tam (2000) and Lauritsch *et al* (2000), which is a filtered backprojection (FBP) formulation of the PHI method with comparable computational effort, and the formula of Katsevich (see either Katsevich (2002a) or Katsevich (2003)). Alternatively, exact reconstruction can also be achieved by adding a correction image to the output of quasi-exact algorithms (see Tam *et al* (2002)). In this paper, the attention is focused on reconstruction using Katsevich's formula.

Katsevich's formula represents an important breakthrough for helical cone-beam reconstruction because it is of filtered backprojection type with simple 1D shift-invariant filtering. It was successfully tested in Köhler *et al* (2002), which provides a comparison between wedge reconstruction and reconstruction with the original Katsevich formula, and in Katsevich *et al* (2003). However, in each case, few details were given on how to process the data. Katsevich's formula was basically published in a very general form that disregards the detector geometry.

This work investigates how to efficiently and accurately implement Katsevich's formula for direct reconstruction from data measured in two specific detector geometries. The first geometry is the curved detector geometry of third-generation CT scanners, and the second geometry is the flat detector geometry of C-arm systems and of most industrial CB CT scanners. These geometries, which are described in section 2, are referred to as native geometries to emphasize their effective implementation in CT scanners. Section 3 gives a summary of Katsevich's formula. Next, sections 4 and 5 focus on reconstruction from data measured in the curved and flat detector geometries, respectively. We have determined processing steps that apply directly to such measured data and yield an implementation of Katsevich's formula as a final outcome. These steps are first given using continuous-form equations, disregarding the finite detector resolution and the source position sampling. Next, techniques are presented for their implementation with finite data sampling. The performance of these techniques is illustrated in section 6 for the curved detector geometry, with 32, 64 and 128 detector rows.

2. Description of the CB measurements

Conceptually, CB projections can be described with or without the use of a specific detector geometry. The description without a specific geometry is very general and thereby highly convenient for the development of mathematical reconstruction formulae. However, it offers little insight for implementation. On the other hand, a specific description of the projections often hampers theoretical developments but is highly convenient for the development of an efficient reconstruction algorithm. Below, the general description of the CB projections is reviewed, then the specific expression for the curved detector geometry and the flat detector geometry is given. In each case, a helical CB geometry of axis z, radius R_o and helix pitch P is considered.

Throughout the paper, the 3D object density to be reconstructed is either denoted as f, $f(x, y, z)$ or $f(\underline{x})$ with $\underline{x} = (x, y, z)$. It is assumed that f is zero outside the cylinder $x^2 + y^2 < R^2$ for a given $R < R_o$. The region $x^2 + y^2 < R^2$ is called the field-of-view (FOV). There is no restriction on the size of the FOV other than $R < R_o$.

2.1. General formulation

By definition, a CB projection of f is the set of integrals of f on all half-lines that diverge from a given point in space, called the vertex point. In CT, the vertex point represents the position of the focal spot of the x-ray source relative to the patient.

In this paper, the CB projections are assumed to be known for vertex points on a helix of axis z, radius R_o and pitch P. Each vertex point is thus given by the value of the vector

$$\underline{a}(\lambda) = \left[R_o \cos(\lambda + \lambda_0), R_o \sin(\lambda + \lambda_0), z_0 + P \frac{\lambda}{2\pi} \right] \tag{1}$$

at some angle λ. At $\lambda = 0$, the vertex point is in the plane $z = z_0$ at polar angle λ_0. As λ increases, the vertex point moves in the positive z-direction at the pace of a distance P per 2π-increase of λ. In third-generation diagnostic x-ray CT scanners, λ represents the rotation angle of the source–detector assembly, while P is the table feed per turn. The range of variation for λ generally depends on the axial extent and location of the region-of-interest, on the radial size of the FOV, and also on the formula used for reconstruction. With Katsevich's formula and a ratio $R/R_o = 25/57 \simeq 0.44$, an overscan of about $145°$ is needed (see Katsevich *et al* (2003)).
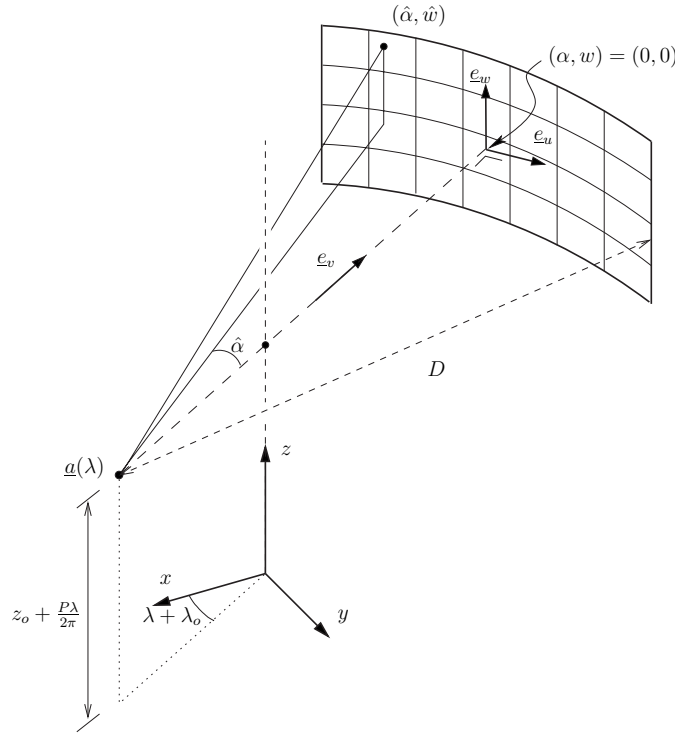
Figure 1. Geometry of data acquisition with a curved detector panel.

The CB projection from the vertex point $\underline{a}(\lambda)$ is

$$g(\lambda, \underline{\theta}) = \int_0^{+\infty} dt \ f(\underline{a}(\lambda) + t\underline{\theta}) \qquad \underline{\theta} \in S \tag{2}$$

where S is the set of all possible unit vectors in space. The CB projection is thus a function of $\underline{\theta} \in S$ parametrized by λ, the angular position of the vertex point.

2.2. Curved detector geometry

This geometry is that of third-generation multi-slice CT scanners. The CB projection is measured using a curved panel of detectors that is shaped to fit geometrically on a cylinder of radius D centred on the vertex point and parallel to the z-axis. See figure 1.

The curved detector panel consists of $N_{rows} \times N_{cols}$ detectors. The detector columns are parallel to the z-axis, while the detector rows form circle arcs parallel to each other and to the (x, y)-plane. The location of (the centre of) a detector in the panel is specified by two parameters, α and w. The parameter w is the signed distance from the centre of the detector to the z-plane passing through the vertex point. The parameter α is the (signed) angle between the plane that contains the vertex point and the z-axis and the plane parallel to the z-axis through the vertex point and the detector centre. It is assumed that there is no truncation in α, i.e the detector panel is large enough to cover the entire FOV in α. On the other hand, the w-extent of the detector panel only allows the coverage of a segment of the object in z. The size of this segment must be taken into account in the selection of the helix pitch P.

Using the curved detector panel, the CB projection appears as a function $g_{(c)}$ of λ, α and w. To describe this function, it is useful to introduce rotated coordinate axes

$$\underline{e}_u(\lambda) = [-\sin(\lambda + \lambda_0), \cos(\lambda + \lambda_0), 0] \tag{3}$$

$$\underline{e}_v(\lambda) = [-\cos(\lambda + \lambda_0), -\sin(\lambda + \lambda_0), 0] \tag{4}$$

$$\underline{e}_w = [0, 0, 1]. \tag{5}$$

As illustrated in figure 1, these axes are obtained from the x, y and z axes by a rotation of angle $\pi/2 + \lambda + \lambda_0$ about the z-axis. By definition,

$$g_{(c)}(\lambda, \alpha, w) = g(\lambda, \underline{\theta}_c) \tag{6}$$

with

$$\underline{\theta}_c = \frac{1}{\sqrt{D^2 + w^2}} (D \sin \alpha \, \underline{e}_u(\lambda) + D \cos \alpha \, \underline{e}_v(\lambda) + w \, \underline{e}_w). \tag{7}$$

Conversely, for a direction $\underline{\theta}$ pointing towards the detector panel

$$g(\lambda, \underline{\theta}) = g_{(c)}(\lambda, \alpha_c, w_c) \tag{8}$$

with

$$\alpha_c = \arctan \frac{\underline{\theta} \cdot \underline{e}_u(\lambda)}{\underline{\theta} \cdot \underline{e}_v(\lambda)} \qquad w_c = \frac{D(\underline{\theta} \cdot \underline{e}_w)}{\sqrt{1 - (\underline{\theta} \cdot \underline{e}_w)^2}}. \tag{9}$$

Note that w_c is independent of λ, since $\underline{e}_w$ is independent of λ.

2.3. Flat detector geometry

In this geometry, the CB projection is measured using a flat panel of detectors parallel to the z-axis and of orientation changing with the vertex point so as to be always orthogonal to the plane containing the vertex point and the z-axis (see figure 2). The detector plane is thus orthogonal to the unit vector $\underline{e}_v(\lambda)$ of equation (4). For convenience, it is assumed that the distance between the detector plane and the vertex point is a constant, denoted by D.

The detector panel consists of $N_{\text{rows}} \times N_{\text{cols}}$ detectors and is positioned so that the detector rows and columns are, respectively, parallel to the $\underline{e}_u$ and $\underline{e}_w$ axes of equations (3) and (5). The location of (the centre of) a detector in the panel is specified by two Cartesian coordinates, u and w. These coordinates are, respectively, measured along $\underline{e}_u$ and $\underline{e}_w$ and are such that $(u, w) = (0, 0)$ is at the orthogonal projection of the vertex point onto the detector plane. It is assumed that there is no truncation in u, i.e. the detector panel is large enough to cover the entire FOV in u. On the other hand, the w-extent of the detector panel only allows the coverage of a segment of the object in z. The size of this segment must be taken into account in the selection of the helix pitch P.

Using the flat detector panel, the CB projection appears as a function $g_{(f)}$ of λ, u and w such that

$$g_{(f)}(\lambda, u, w) = g(\lambda, \underline{\theta}_f) \tag{10}$$

with

$$\underline{\theta}_f = \frac{1}{\sqrt{u^2 + D^2 + w^2}} (u \underline{e}_u(\lambda) + D \underline{e}_v(\lambda) + w \underline{e}_w). \tag{11}$$

Conversely, for a direction $\underline{\theta}$ pointing towards the detector panel

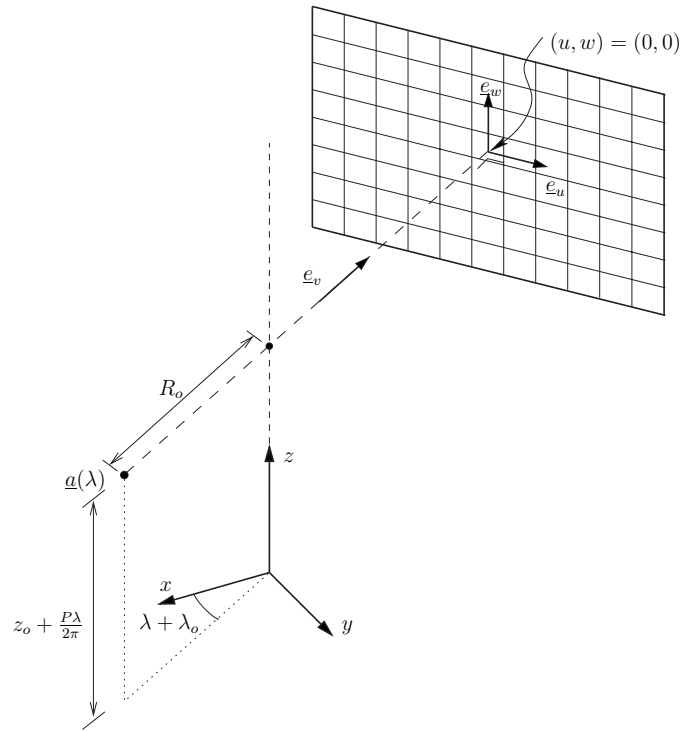$$g(\lambda, \underline{\theta}) = g_{(f)}(\lambda, u_f, w_f) \tag{12}$$

Figure 2. Geometry of data acquisition with a flat detector panel.

with

$$u_f = D \frac{\underline{\theta} \cdot \underline{e}_u(\lambda)}{\underline{\theta} \cdot \underline{e}_v(\lambda)} \qquad w_f = D \frac{\underline{\theta} \cdot \underline{e}_w}{\underline{\theta} \cdot \underline{e}_v(\lambda)}. \tag{13}$$

Note that with the adopted notations, there is an easy link between the CB projection on the curved panel and the CB projection on the flat panel:

$$g_{(c)}(\lambda, \alpha, w) = g_{(f)}(\lambda, D \tan \alpha, w/\cos \alpha). \tag{14}$$

3. The Katsevich formula

This section gives a description without proof of Katsevich's formula (Katsevich 2002a, 2003) for exact helical CB reconstruction. Aside from being exact, this formula presents two additional key properties that make it very attractive. First, it is essentially a filtered-backprojection reconstruction formula with 1D shift-invariant filtering and is therefore efficient. Second, z-truncation is allowed, i.e. the values of any CB projection do not need to be known for all rays passing through the object in the axial (z) direction. In fact, the formula is nearly optimal in terms of required data coverage in the axial direction for a given helix pitch.

3.1. Geometrical tools

The Katsevich formula makes use of π-lines and of particular planes that will be called here κ-planes.

A π-line is any segment of line that connects two vertex points separated by less than one helix turn (i.e. separated by less than $360°$ in λ). π-lines obey the following property: for every point $\underline{x} \in$ FOV, there is a π-line passing through $\underline{x}$ and this π-line is unique. See Danielsson *et al* (1997) or Defrise *et al* (2000) for a proof. Geometrically, the helix segment that connects the extremities of the unique π-line through a given point defines a $180°$ data coverage of that point. For this reason, Danielsson *et al* (1997) conjectured that axially-truncated CB data on the helix segment that connects the extremities of the π-line through a given point should be enough to exactly reconstruct f at that point. This conjecture was proved by Katsevich who derived a formula satisfying this property (Katsevich 2003).

A κ-plane is any plane that has three intersections with the helix such that one intersection is at mid-way between the two others (in λ). There exists a family of κ-planes of interest for each vertex point $\underline{a}(\lambda)$. These planes are specified using an angle $\psi \in (-\pi, \pi)$. The κ-plane of angle ψ at $\underline{a}(\lambda)$, denoted by $\mathcal{K}(\lambda, \psi)$, is, by definition, the plane that contains $\underline{a}(\lambda)$, $\underline{a}(\lambda + \psi)$ and $\underline{a}(\lambda + 2\psi)$. The unit vector normal to $\mathcal{K}(\lambda, \psi)$ is denoted by $\underline{n}(\lambda, \psi)$. Requesting the angle between $\underline{n}(\lambda, \psi)$ and the z-axis to be acute, we have

$$\underline{n}(\lambda, \psi) = \frac{(\underline{a}(\lambda + \psi) - \underline{a}(\lambda)) \times (\underline{a}(\lambda + 2\psi) - \underline{a}(\lambda))}{\|(\underline{a}(\lambda + \psi) - \underline{a}(\lambda)) \times (\underline{a}(\lambda + 2\psi) - \underline{a}(\lambda))\|} \operatorname{sign}(\psi) \tag{15}$$

where the symbol $\times$ refers to a cross product. As ψ tends to zero, the κ-plane $\mathcal{K}(\lambda, \psi)$ converges to the plane that is tangent to the helix at $\underline{a}(\lambda)$ and parallel to $\underline{e}_v(\lambda)$ (equation (4)):

$$\lim_{\psi \to 0} \underline{n}(\lambda, \psi) = \frac{1}{\sqrt{1 + 4\pi^2 R_o^2 / P^2}} \left(-\underline{e}_u + \frac{2\pi R_o}{P} \underline{e}_w \right)$$

$$= \frac{1}{\sqrt{1 + 4\pi^2 R_o^2 / P^2}} [\sin(\lambda + \lambda_0), -\cos(\lambda + \lambda_0), 2\pi R_o / P]. \tag{16}$$

Together, κ-planes and π-lines obey the following important property. Consider a point $\underline{x}$ within the FOV and let $\lambda_i(\underline{x})$ and $\lambda_o(\underline{x})$ define the extremities of the π-line through that point, with $\lambda_i(\underline{x}) < \lambda_o(\underline{x})$. For any $\lambda \in [\lambda_i(\underline{x}), \lambda_o(\underline{x})]$, there exists a κ-plane $\mathcal{K}(\lambda, \psi)$ at $\underline{a}(\lambda)$ that contains $\underline{x}$. This property holds for any FOV size $R < R_o$. However, unicity may not hold, that is there may be more than one κ-plane at $\underline{a}(\lambda)$ that contains $\underline{x}$. Between these various κ-planes, there exist only one with minimum value for $|\psi|$. This plane is such that $\lambda_i(\underline{x}) \leqslant \lambda + 2\psi \leqslant \lambda_o(\underline{x})$ and is used in Katsevich's formula below. See Katsevich (2003) for more details.

3.2. The reconstruction formula

Following the results of Katsevich, $f(\underline{x})$ with $\underline{x}$ in the FOV is reconstructed using CB backprojection over the segment of the helix joining the extremities of the π-line passing through $\underline{x}$. The backprojection involves filtered data, denoted by $g^F(\lambda, \underline{\theta})$. Specifically,

$$f(\underline{x}) = -\frac{1}{2\pi} \int_{\lambda_i(\underline{x})}^{\lambda_o(\underline{x})} d\lambda \frac{1}{\|\underline{x} - \underline{a}(\lambda)\|} g^F \left(\lambda, \frac{\underline{x} - \underline{a}(\lambda)}{\|\underline{x} - \underline{a}(\lambda)\|} \right) \tag{17}$$

where $\lambda_i(\underline{x})$ and $\lambda_o(\underline{x})$ define the extremities of the π-line through $\underline{x}$ with $\lambda_i(\underline{x}) < \lambda_o(\underline{x})$.

To obtain $g^F(\lambda, \underline{\theta})$, it is first needed to get $g'(\lambda, \underline{\theta})$, the derivative of $g(\lambda, \underline{\theta})$ with respect to λ at fixed direction $\underline{\theta}$:

$$g'(\lambda, \underline{\theta}) = \lim_{\varepsilon \to 0} \frac{g(\lambda + \varepsilon, \underline{\theta}) - g(\lambda, \underline{\theta})}{\varepsilon}. \tag{18}$$

Once $g'(\lambda, \underline{\theta})$ is known, $g^F(\lambda, \underline{\theta})$ is computed using the equation

$$g^F(\lambda, \underline{\theta}) = \int_0^{2\pi} d\gamma \, h_H(\sin \gamma) g'(\lambda, \cos \gamma \underline{\theta} + \sin \gamma (\underline{\theta} \times \underline{m}(\lambda, \underline{\theta}))). \qquad (19)$$

In this expression, $h_H(s)$ is the kernel of the Hilbert transform, that is

$$h_H(s) = -\int_{-\infty}^{+\infty} d\sigma \, i \, \mathrm{sgn}(\sigma) \, \mathrm{e}^{\mathrm{i}2\pi\sigma s} = \frac{1}{\pi s} \qquad (20)$$

and vector $\underline{m}(\lambda, \underline{\theta})$ is normal to the κ-plane $\mathcal{K}(\lambda, \psi)$ of the smallest $|\psi|$ value that contains the line of direction $\underline{\theta}$ through $\underline{a}(\lambda)$. This vector is given by equation (15) using the relevant value of ψ; this value is between $-\pi/2 - \arcsin(R/R_o)$ and $\pi/2 + \arcsin(R/R_o)$ when $(\lambda, \underline{\theta})$ defines a half-line passing through the FOV.

Note from (17) that $g^F(\lambda, \underline{\theta})$ only needs to be known for vectors $\underline{\theta}$ that specify the direction of the lines starting at $\underline{a}(\lambda)$ and passing through points $\underline{x}$ in the FOV for which $\lambda_i(\underline{x}) < \lambda < \lambda_o(\underline{x})$. For such vectors $\underline{\theta}$, there always exists a κ-plane at $\underline{a}(\lambda)$ that contains the line of direction $\underline{\theta}$ through $\underline{a}(\lambda)$ (see the discussion at the end of section 3.1). Hence, $\underline{m}(\lambda, \underline{\theta})$ always exist and enforcing $|\psi|$ to be minimum in the selection of the κ-plane defines it uniquely and appropriately for Katsevich's formula to be correct (Katsevich 2003).

As explained in Katsevich (2003), to appreciate the filtering (convolution) nature of equation (19), it is worth considering a fixed κ-plane, $\mathcal{K}(\lambda, \psi)$, and evaluating g^F for all lines $(\lambda, \underline{\theta})$ that lie in $\mathcal{K}(\lambda, \psi)$ and are thus such that $\underline{m}(\lambda, \underline{\theta}) = \underline{n}(\lambda, \psi)$. Let $\underline{c}_1$ and $\underline{c}_2$ be two unit orthogonal vectors parallel to $\mathcal{K}(\lambda, \psi)$ with $\underline{c}_1 \times \underline{c}_2 = \underline{n}(\lambda, \psi)$. With these vectors, the direction of any line $(\lambda, \underline{\theta})$ in $\mathcal{K}(\lambda, \psi)$ can be written in the form $\underline{\theta}(\delta) = \cos \delta \underline{c}_1 + \sin \delta \underline{c}_2$, where δ is some angle. Consequently,

$$g^F(\lambda, \underline{\theta}(\delta)) = \int_0^{2\pi} d\gamma \, h_H(\sin \gamma) g'(\lambda, \cos(\delta - \gamma)\underline{c}_1 + \sin(\delta - \gamma)\underline{c}_2) \qquad (21)$$

which is a convolution formula.

The proof of Katsevich formula can be found in Katsevich (2003). It should be noted that the formula presented here above is only one version of Katsevich's formula. As explained in Katsevich (2002a) and Katsevich *et al* (2003), it is possible to reformulate the reconstruction formula so that there is no need to take a derivative in λ. The outcome is still an FBP formula, but it involves four different filtered projections and also two backprojection steps, one given by equation (17) and a second one similar to equation (17) but with a weighting factor $1/\|\underline{x} - \underline{a}(\lambda)\|^2$ instead of $1/\|\underline{x} - \underline{a}(\lambda)\|$. This modified formula was disregarded for computational efficiency reasons.

Another flexibility in Katsevich's formula is the definition of the κ-planes. The definition given in section 3.1 is one formulation out of many (see equations (4) to (6) in Katsevich (2002a)). This definition was selected as it seems to be the most natural one according to Katsevich's results (Katsevich 2003).

4. Reconstruction in the geometry of the curved detector panel

This section consists of three parts. In the first part, the steps to implement Katsevich's formula using the sampling geometry of the curved detector panel are given. In the second part, the number of detector rows needed for reconstruction using a given helix pitch is explained. In the last part, implementation strategies are discussed for various steps of the algorithm. A proof of the results given in this section can be found in the appendix.

4.1. The algorithm

Using the curved-panel geometry, the reconstruction is achieved in two steps:

Step 1: filtering. Each projection $g_{(c)}(\lambda, \alpha, w)$ is modified into a filtered projection $g_{(c)}^F(\lambda, \alpha, w)$ according to the following steps:

CF1: derivative at constant direction. Compute from the data $g_{(c)}(\lambda, \alpha, w)$, $g_1(\lambda, \alpha, w)$ such that

$$g_1(\lambda, \alpha, w) = g'(\lambda, \underline{\theta}_c) \tag{22}$$

with $\underline{\theta}_c$ given by equation (7) and $g'(\lambda, \underline{\theta}_c)$, the derivative of g with respect to λ at constant line direction $\underline{\theta}_c$, i.e

$$g'(\lambda, \underline{\theta}_c) = \lim_{\varepsilon \to 0} \frac{g(\lambda + \varepsilon, \underline{\theta}_c) - g(\lambda, \underline{\theta}_c)}{\varepsilon} \tag{23}$$

(see equation (18)).

CF2: length-correction weighting. Compute

$$g_2(\lambda, \alpha, w) = \frac{D}{\sqrt{D^2 + w^2}} g_1(\lambda, \alpha, w) \tag{24}$$

CF3: forward height rebinning. Let α_m be half the fan angle defined by the size of the FOV and the helix radius, i.e. $\alpha_m = \arcsin(R/R_o)$. This step consists in the computation of

$$g_3(\lambda, \alpha, \psi) = g_2(\lambda, \alpha, w_\kappa(\alpha, \psi)) \tag{25}$$

for all $\psi \in [-\pi/2 - \alpha_m, \pi/2 + \alpha_m]$, with

$$w_\kappa(\alpha, \psi) = \frac{DP}{2\pi R_o} \left(\psi \cos \alpha + \frac{\psi}{\tan \psi} \sin \alpha \right). \tag{26}$$

At fixed ψ, equation (26) describes a curve in the detector area that will be called κ-curve of angle ψ. This curve is the intersection between the detector panel and the κ-plane $\mathcal{K}(\lambda, \psi)$ (see figure 3).

CF4: 1D Hilbert transform in α. Compute at constant ψ

$$g_4(\lambda, \alpha, \psi) = \int_{-\pi/2}^{+\pi/2} d\alpha' h_H(\sin(\alpha - \alpha')) g_3(\lambda, \alpha', \psi) \tag{27}$$

where h_H is the kernel of the Hilbert transform (see equation (20)).

CF5: backward height rebinning. Compute

$$g_5(\lambda, \alpha, w) = g_4(\lambda, \alpha, \hat{\psi}(\alpha, w)) \tag{28}$$

where $\hat{\psi}(\alpha, w)$ is angle ψ of the smallest absolute value that satisfies the equation

$$w = \frac{DP}{2\pi R_o} \left(\psi \cos \alpha + \frac{\psi}{\tan \psi} \sin \alpha \right). \tag{29}$$

(Looking at steps CF3, CF4 and CF5 together, we see that the value of $g_5(\lambda, \alpha, w)$ at a given detector location (α_o, w_o) is found from a convolution of the values of g_2 on the κ-curve of the smallest $|\psi|$-value that passes through (α_o, w_o).)

CF6: post-cosine weighting. Compute

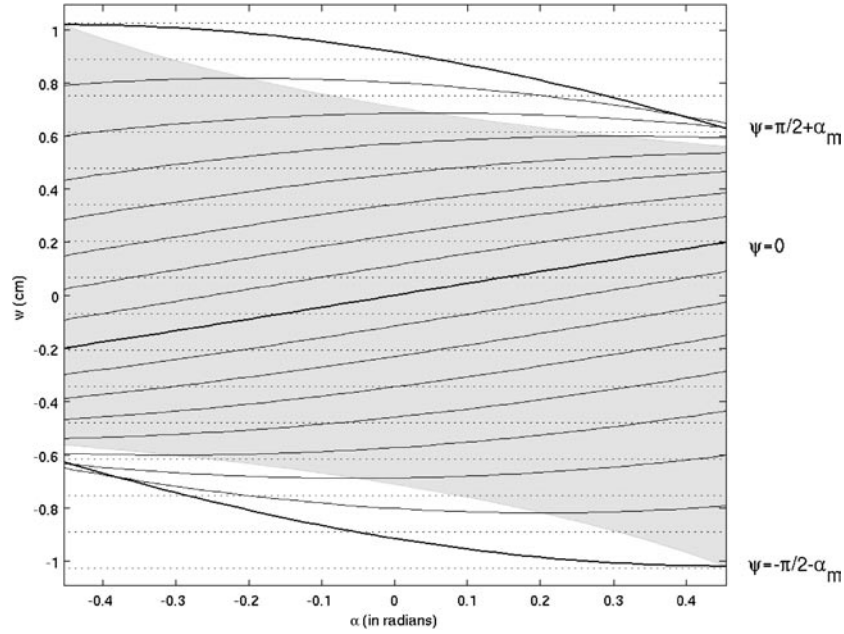$$g_{(c)}^F(\lambda, \alpha, w) = \cos \alpha \, g_5(\lambda, \alpha, w). \tag{30}$$

Figure 3. Illustration of κ-curves for a curved detector panel with 16 detector rows. The horizontal dotted lines correspond to the central line of each detector row.

Step 2: backprojection. The filtered projection $g_{(c)}^F(\lambda, \alpha, w)$ is backprojected to obtain f at each point $\underline{x} = (x, y, z)$ of the FOV according to the formula

$$f(\underline{x}) = \frac{1}{2\pi} \int_{\lambda_i(\underline{x})}^{\lambda_o(\underline{x})} d\lambda \frac{1}{v^*(\lambda, \underline{x})} g_{(c)}^F(\lambda, \alpha^*(\lambda, \underline{x}), w^*(\lambda, \underline{x})) \tag{31}$$

where $\lambda_i(\underline{x})$ and $\lambda_o(\underline{x})$ define the extremities of the π-line through $\underline{x}$ with $\lambda_i(\underline{x}) < \lambda_o(\underline{x})$ and

$$v^*(\lambda, \underline{x}) = R_o - x \cos(\lambda + \lambda_0) - y \sin(\lambda + \lambda_0) \tag{32}$$

$$\alpha^*(\lambda, \underline{x}) = \arctan\left(\frac{1}{v^*(\lambda, \underline{x})}(-x \sin(\lambda + \lambda_0) + y \cos(\lambda + \lambda_0))\right) \tag{33}$$

$$w^*(\lambda, \underline{x}) = \frac{D \cos \alpha^*(\lambda, \underline{x})}{v^*(\lambda, \underline{x})} \left(z - z_0 - \frac{P}{2\pi}\lambda\right). \tag{34}$$

4.2. Number of detector rows needed

As can be expected, a minimum number of detector rows are required for the above algorithm to be applicable. This number depends on the specific value of the helix parameters, P and R_o, and also on the size R of the FOV. Basically, the number of detector rows must be large enough so that the filtered data can be computed at any point needed for the CB backprojection.

The detector region needed for the CB backprojection of equation (31) is known in the literature as the Tam–Danielsson window (see Danielsson *et al* (1997) and Tam *et al* (1998)). This region is the set of (α, w)-points such that

$$\alpha \in [-\alpha_m, \alpha_m] \qquad w_{\text{bottom}}(\alpha) \leqslant w \leqslant w_{\text{top}}(\alpha) \tag{35}$$

Table 1. Maximum helix pitch and corresponding pitch factor $PD/(N_{rows}R_o d_w)$ for a curved detector ($R = 25$ cm, $R_o = 57$ cm and $R_o d_w/D = 0.75$ mm)

N_{rows}	4	8	16	32	64	128
Maximum helix pitch P (cm)	0.31	0.73	1.56	3.24	6.58	13.28
Maximum pitch factor	1.03	1.21	1.30	1.35	1.37	1.38

where α_m is the half fan angle (see step CF3),

$$w_{\text{bottom}}(\alpha) = -\frac{DP}{2\pi R_o}\frac{\pi/2+\alpha}{\cos\alpha} \qquad w_{\text{top}}(\alpha) = \frac{DP}{2\pi R_o}\frac{\pi/2-\alpha}{\cos\alpha}. \tag{36}$$

To compute g_c^F at any point in that region, the filtering steps CF1 to CF6 must be carried out. If these steps were local, no more data points than those in the Tam–Danielsson window would be needed. However, the Hilbert transform of filtering step CF4 is a non-local operation, so extra data points are needed. To find all required data points, we need to look at the area covered by the κ-curve of equation (26) when ψ varies between $-\pi/2 - \alpha_m$ and $\pi/2 + \alpha_m$.

Figure 3 illustrates a few κ-curves with $\psi \in [-\pi/2 - \alpha_m, \pi/2 + \alpha_m]$. From this figure, it appears clear that using $\psi \in [-\pi/2 - \alpha_m, \pi/2 + \alpha_m]$ allows all points in the Tam–Danielsson window to be covered, as stated in Katsevich (2003); this justifies the requested range of variation for ψ in filtering step CF3. Also, the area covered by the κ-curves when ψ varies between $-\pi/2 - \alpha_m$, and $\pi/2 + \alpha_m$ is seen to be between the detector lines of equation $w = w_{\text{top}}(-\alpha_m)$ and $w = w_{\text{bottom}}(\alpha_m)$. Hence, the number of detector rows needed is

$$\begin{aligned} N_{\text{rows}} &= \frac{d_w + w_{\text{top}}(-\alpha_m) - w_{\text{bottom}}(\alpha_m)}{d_w} \\ &= 1 + \frac{DP}{\pi R_o d_w}\frac{\pi/2+\alpha_m}{\cos\alpha_m} \end{aligned} \tag{37}$$

if d_w is the thickness of the detector rows. Table 1 outlines the relation between N_{rows} and the maximum allowable helix pitch for parameters that are representative of a third-generation CT scanner. Table 1 also provides the corresponding value of the pitch factor, $PD/(N_{\text{rows}}R_o d_w)$, which is widely used in medical CT.

4.3. Implementation strategies

This section discusses implementation strategies for the steps of the algorithm in section 4.1. For efficiency reasons, only linear interpolation techniques are considered. It is assumed that N_s projections have been measured at a uniform rate of N projections per 2π-rotation. The angle λ for these projections is $\lambda_k = k\Delta\lambda$ with $\Delta\lambda = 2\pi/N$ and $k = 0, \ldots, N_s - 1$. The curved detector panel consists of $N_{\text{cols}} \times N_{\text{rows}}$ detectors of width d_α and height d_w at locations

$$\begin{aligned} \alpha_i &= (i + \delta - (N_{\text{cols}} - 1)/2)\Delta\alpha & i &= 0, \ldots, N_{\text{cols}} - 1 \\ w_j &= (j - (N_{\text{rows}} - 1)/2)d_w & j &= 0, \ldots, N_{\text{rows}} - 1 \end{aligned} \tag{38}$$

where $\Delta\alpha = d_\alpha/D$ and δ is either $1/4$ or 0, depending on whether or not a quarter-detector offset is considered. The available CB data are thus $g_{(c)}(\lambda_k, \alpha_i, w_j)$ with indices i, j and k running as shown above.

4.3.1. Filtering step CF1. The problem here is to compute samples of $g_1(\lambda, \alpha, w)$ from the data $g_{(c)}(\lambda_k, \alpha_i, w_j)$ according to equation (22). We suggest computing the derivative

at interlaced sample positions using pairs of consecutive projections. Such an approach is favourable to minimize memory access while limiting the loss of resolution associated with this derivative step. In any case, a controlled amount of smoothing may want to be added later, in step CF4, when applying the Hilbert transform.

The inspection of equation (22) induces a straightforward approach to compute g_1 at interlaced source positions $\lambda_{k+1/2} = \lambda_k + \Delta\lambda/2$, namely

$$g_1(\lambda_{k+1/2}, \alpha, w) \simeq \frac{g(\lambda_{k+1}, \underline{\theta}_c) - g(\lambda_k, \underline{\theta}_c)}{\Delta\lambda} \tag{39}$$

where $\underline{\theta}_c$ is given by equation (7), using $\lambda_{k+1/2}$ for λ. The above formula represents one of two techniques suggested. For this formula to be useful, it is needed to convert the values of g into values of $g_{(c)}$. This can be done using equations (8) and (9) with $\underline{\theta}_c$ for $\underline{\theta}$. The result is

$$g_1(\lambda_{k+1/2}, \alpha, w) \simeq \frac{g_{(c)}(\lambda_{k+1}, \alpha_{\text{right}}, w) - g_{(c)}(\lambda_k, \alpha_{\text{left}}, w)}{\Delta\lambda} \tag{40}$$

with

$$\alpha_{\text{right}} = \arctan\left(\frac{\underline{\theta}_c \cdot \underline{e}_u(\lambda_{k+1})}{\underline{\theta}_c \cdot \underline{e}_v(\lambda_{k+1})}\right) \qquad \alpha_{\text{left}} = \arctan\left(\frac{\underline{\theta}_c \cdot \underline{e}_u(\lambda_k)}{\underline{\theta}_c \cdot \underline{e}_v(\lambda_k)}\right). \tag{41}$$

Using the expression of $\underline{\theta}_c$ (equation (7) with $\lambda_{k+1/2}$ for λ) and of vectors $\underline{e}_u$ and $\underline{e}_v$ (equations (3) and (4)), we get $\alpha_{\text{right}} = \alpha + \Delta\lambda/2$ and $\alpha_{\text{left}} = \alpha - \Delta\lambda/2$, so that

$$g_1(\lambda_{k+1/2}, \alpha, w) \simeq \frac{g(\lambda_{k+1}, \alpha + \Delta\lambda/2, w) - g(\lambda_k, \alpha - \Delta\lambda/2, w)}{\Delta\lambda}. \tag{42}$$

This formula can be applied at each detector location (α_i, w_j) to build the derivative data $g_1(\lambda_{k+1/2}, \alpha_i, w_j)$, with $k = 0, \ldots, N_s - 2, i = 0, \ldots, N_{\text{cols}} - 1, j = 0, \ldots, N_{\text{rows}} - 1$. In this process, linear interpolation in α may be required to get $g(\lambda_{k+1}, \alpha_i + \Delta\lambda/2, w_j)$ and $g(\lambda_k, \alpha_i - \Delta\lambda/2, w_j)$.

Another approach to get an expression for samples of g_1 is to apply the chain differentiation rule to equation (8). This rule yields the following equation

$$g'(\lambda, \underline{\theta}) = \left(\frac{\partial g_{(c)}}{\partial\lambda} + \frac{\partial g_{(c)}}{\partial\alpha}\frac{\partial\alpha_c}{\partial\lambda} + \frac{\partial g_{(c)}}{\partial w}\frac{\partial w_c}{\partial\lambda}\right)(\lambda, \alpha_c, w_c). \tag{43}$$

That is, upon the use of (9) for the expression of α_c and w_c,

$$g'(\lambda, \underline{\theta}) = \left(\frac{\partial g_{(c)}}{\partial\lambda} + \frac{\partial g_{(c)}}{\partial\alpha}\right)(\lambda, \alpha_c, w_c). \tag{44}$$

Or equivalently,

$$g_1(\lambda, \alpha, w) = g'(\lambda, \underline{\theta}_c) = \left(\frac{\partial g_{(c)}}{\partial\lambda} + \frac{\partial g_{(c)}}{\partial\alpha}\right)(\lambda, \alpha, w). \tag{45}$$

From this last equation, we suggest computing g_1 at interlaced sample positions in λ and α, using the following formula for $k = 0, \ldots, N_s - 2, i = 0, \ldots, N_{\text{cols}} - 2, j = 0, \ldots, N_{\text{rows}} - 1$:

$$\begin{aligned}
g_1(\lambda_{k+1/2}, \alpha_{i+1/2}, w_j) \simeq{}& \frac{g_1(\lambda_{k+1}, \alpha_i, w_j) - g_1(\lambda_k, \alpha_i, w_j)}{2\Delta\lambda} \\
&+ \frac{g_1(\lambda_{k+1}, \alpha_{i+1}, w_j) - g_1(\lambda_k, \alpha_{i+1}, w_j)}{2\Delta\lambda} \\
&+ \frac{g_1(\lambda_k, \alpha_{i+1}, w_j) - g_1(\lambda_k, \alpha_i, w_j)}{2\Delta\alpha} \\
&+ \frac{g_1(\lambda_{k+1}, \alpha_{i+1}, w_j) - g_1(\lambda_{k+1}, \alpha_i, w_j)}{2\Delta\alpha}
\end{aligned} \tag{46}$$

where $\alpha_{i+1/2} = \alpha_i + \Delta\alpha/2$. The first two terms in this expression form an estimate of $\partial g_{(c)}/\partial\lambda$ at $(\lambda_{k+1/2}, \alpha_{i+1/2}, w_j)$, using the average of estimates at $(\lambda_{k+1/2}, \alpha_i, w_j)$ and $(\lambda_{k+1/2}, \alpha_{i+1}, w_j)$. The last two terms form an estimate of $\partial g_{(c)}/\partial\alpha$ at $(\lambda_{k+1/2}, \alpha_{i+1/2}, w_j)$, using the average of estimates at $(\lambda_k, \alpha_{i+1/2}, w_j)$ and $(\lambda_{k+1}, \alpha_{i+1/2}, w_j)$.

4.3.2. Filtering step CF3. Once samples in ψ have been specified, this step can be implemented using a simple linear interpolation in w. We suggest considering an odd number of samples uniformly distributed over the range of variation of ψ:

$$\psi_l = l\Delta\psi \qquad l = -M, \ldots, M \tag{47}$$

with $\Delta\psi = (\pi + 2\alpha_m)/(2M)$.

Unfortunately, it turns out that the selection of M is difficult: figure 3, which illustrates κ-curves for $M = 8$ and $N_{\text{rows}} = 16$, shows that the w-separation between the κ-curves is not uniform across the Tam–Danielsson window. To use linear interpolation in w, we suggest selecting M such that the maximum separation between the κ-curves at the half fan angle is equal to the detector row thickness, that is we suggest selecting M such that

$$\frac{\mathrm{d}w_\kappa}{\mathrm{d}\psi}(\alpha = -\alpha_m, \psi = \pi/2 + \alpha_m) \simeq \frac{d_w}{\Delta\psi} = \frac{d_w}{(\pi + 2\alpha_m)/(2M)} \tag{48}$$

with w_κ given by equation (26). At maximum allowable helix pitch, this criteria yields

$$2M \simeq N_{\text{rows}} \left(1 + \left(\frac{\pi}{2} + \alpha_m\right)\tan\alpha_m\right). \tag{49}$$

That is, $2M/N_{\text{rows}} \simeq 2$ for a half fan angle α_m of $26°$.

Note that using twice as many κ curves as detector rows doubles the computational time for step CF4. A more sophisticated (but less efficient) interpolation scheme in w for step CF3 could allow a reduction in the number of κ-curves to be considered, and thereby yield an overall gain in efficiency. However, this line of research was not investigated in this work.

4.3.3. Filtering step CF4. This filtering step is very similar to that encountered in conventional fan-beam CT. Following Natterer (1986), we suggest approximating the convolution kernel $h_H(\sin\alpha)$ by the following expression, based on (20) and Nyquist's sampling theory:

$$h_H(\sin\alpha) = \frac{\alpha}{\sin\alpha}h_H(\alpha) \simeq -\frac{\alpha}{\sin\alpha}\int_{-\frac{1}{2\Delta\alpha}}^{\frac{1}{2\Delta\alpha}} \mathrm{d}\sigma \, \mathrm{i}\,\mathrm{sgn}(\sigma)\,\mathrm{e}^{\mathrm{i}2\pi\sigma\alpha}A(2\sigma\,\Delta\alpha)$$

$$= \frac{1}{\Delta\alpha}H\left(\frac{\alpha}{\Delta\alpha}\right) \tag{50}$$

where $A(x)$ is any given apodization window and

$$H(t) = \frac{(t\Delta\alpha)}{\sin(t\Delta\alpha)}\int_0^1 \mathrm{d}u \, \sin(\pi t u)A(u). \tag{51}$$

For filtering without apodization, $A(u) = 1$ over the whole integration range of u. For Hanning apodization, $A(u) = \cos^2(\pi u/2)$.

Using H, the convolution formula (27) is discretized into the following discrete convolution formula:

$$g_4(\lambda, \alpha_i, \psi_l) \simeq \sum_{p=0}^{N_{\text{cols}}-1} H(i - p)g_3(\lambda, \alpha_p, \psi_l) \tag{52}$$

or

$$g_4(\lambda, \alpha_i, \psi_l) \simeq \sum_{p=0}^{N_{\text{cols}}-1} H(i - p - 1/2) g_3(\lambda, \alpha_{p+1/2}, \psi_l) \tag{53}$$

depending on whether the filtering step CF1 is achieved using (42) or (46). Each of these formulae can be implemented efficiently using the FFT.

Recall that using (46) implies a shift of the samples in α. Formula (53) is defined so that the samples in α are brought back to their original position.

4.3.4. Filtering step CF5. This step is in appearance complicated because equation (29) is non-linear in ψ and may have more than one solution (the solution of the smallest absolute value is the desired one). Fortunately, there is a way to get around these difficulties using simple linear interpolation. The trick is to realize that by definition,

$$g_4(\lambda, \alpha_i, \psi_l) = g_5(\lambda, \alpha_i, w_\kappa(\alpha_i, \psi_l)) \tag{54}$$

with w_κ given by (26). So

$$g_5(\lambda, \alpha_i, w) \simeq (1 - c(\alpha_i, w, l)) g_4(\lambda, \alpha_i, \psi_l) + c(\alpha_i, w, l) g_4(\lambda, \alpha_i, \psi_{l+1}) \tag{55}$$

for $w_\kappa(\alpha_i, \psi_l) < w < w_\kappa(\alpha_i, \psi_{l+1})$, with $c(\alpha_i, w, l) = (w - w_\kappa(\alpha_i, \psi_l))/(w_\kappa(\alpha_i, \psi_{l+1}) - w_\kappa(\alpha_i, \psi_l))$.

To get $g_5(\lambda, \alpha_i, w_j)$ for a given λ, a given i and any j such that (α_i, w_j) is within the Tam–Danielsson (which is everything needed for the backprojection), we suggest applying (55) over successive values of l, the index that defines the value of ψ. More specifically, for $\alpha_i \geqslant 0$, we suggest using (55) to compute $g_5(\lambda, \alpha_i, w_j)$ at any j such that $w_\kappa(\alpha_i, \psi_l) < w_j < w_\kappa(\alpha_i, \psi_{l+1})$, with l going from $l = -M$ towards $l = M$ as long as $w_\kappa(\alpha_i, \psi_{l+1})$ is larger than $w_\kappa(\alpha_i, \psi_l)$. For $\alpha_i < 0$, the same approach is taken but with l going from $l = M$ towards $l = -M$ as long as $w_\kappa(\alpha_i, \psi_{l-1}) < w_\kappa(\alpha_i, \psi_l)$. This change of loop direction according to α_i sign is enough to guarantee that $g_5(\lambda, \alpha_i, w_j)$ is estimated from the κ-curve with the smallest $|\psi|$ for any w_j, as requested in step CF5. See figure 3 for an illustration of the behaviour of the κ-curves.

4.3.5. Step 2: backprojection. The main difficulty with this step is the voxel dependence of the range of projections over which the CB backprojection is to be carried out. Fortunately, the Tam–Danielsson window can be used to simplify the situation. Let $\chi(\alpha, w)$ be the characteristic function of this window, i.e. $\chi(\alpha, w) = 1$ if $w_{\text{bottom}}(\alpha) < w < w_{\text{top}}(\alpha)$ and 0 otherwise (see (36)). Using $\chi(\alpha, w)$, the backprojection formula (31) can be rewritten in the form

$$f(\underline{x}) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \mathrm{d}\lambda \frac{1}{v^*(\lambda, \underline{x})} \chi(\alpha, w) g_{(c)}^F(\lambda, \alpha, w) \Big|_{\alpha = \alpha^*(\lambda, \underline{x}), w = w^*(\lambda, \underline{x})}. \tag{56}$$

So, the backprojection can be implemented by multiplying each filtered projection by χ and then backprojecting the outcome without worrying about the range of integration. This approach is however not quite as simple as it looks because the function χ is discontinuous.

To avoid significant reconstruction artefacts, it is needed to replace χ by a smooth function χ_a, the optimal definition of which is not easy to find. We suggest using the empirical expression

$$\chi_a(\alpha, w) = \begin{cases} 0 & \text{if} \quad w > w_{\text{top}}(\alpha) + ad_w \\ \frac{w_{\text{top}}(\alpha) + ad_w - w}{2ad_w} & \text{if} \quad w_{\text{top}}(\alpha) - ad_w < w < w_{\text{top}}(\alpha) + ad_w \\ 1 & \text{if} \quad w_{\text{bottom}}(\alpha) + ad_w < w < w_{\text{top}}(\alpha) - ad_w \\ \frac{w - w_{\text{bottom}}(\alpha) + ad_w}{2ad_w} & \text{if} \quad w_{\text{bottom}}(\alpha) - ad_w < w < w_{\text{bottom}}(\alpha) + ad_w \\ 0 & \text{if} \quad w < w_{\text{bottom}}(\alpha) - ad_w \end{cases} \quad (57)$$

where a is an adjustable parameter. With this smooth function, each sampled filtered projection contributes a quantity

$$\delta f(\underline{x}) = \frac{\Delta\lambda}{2\pi v^*(\lambda, \underline{x})} \chi_a(\alpha, w) g_{(c)}^F(\lambda, \alpha, w)\Big|_{\alpha=\alpha^*(\lambda, \underline{x}), w=w^*(\lambda, \underline{x})} \quad (58)$$

to the reconstruction. To compute this quantity efficiently, it should be noted that $v^*(\lambda, \underline{x})$ and $\alpha^*(\lambda, \underline{x})$ are both independent of z. Additionally, $w^*(\lambda, \underline{x}) = \eta(\lambda, \underline{x})(z - z_0 - P\lambda/(2\pi))$ with $\eta(\lambda, \underline{x})$, some function independent of z (compare with (34)). We suggest therefore looping over x and y first and compute $v^*(\lambda, \underline{x})$, $\alpha^*(\lambda, \underline{x})$ and $\eta(\lambda, \underline{x})$. The loop over z can then be performed with z limited to the range of values such that $w_{\text{bottom}}(\alpha^*) - ad_w < w^*(\lambda, \underline{x}) < w_{\text{top}}(\alpha^*) + ad_w$. To determine this range efficiently, the function w_{top} and w_{bottom} should be tabulated.

The method presented hereabove is pretty straightforward but most likely of accuracy depending on the regularization parameter a, the optimal value of which is not clear. Below, we present an alternative approach that avoids the use of any regularization parameter. This approach is analogous to the method used in Katsevich (2002b) to smooth the edges of the Tam–Danielsson window in an FDK-like method based on π-interval backprojection.

Consider the following two functions of λ, defined for a given voxel $\underline{x}$:

$$r_{\text{in}}(\lambda) = w_{\text{top}}(\alpha^*(\lambda, \underline{x})) - w^*(\lambda, \underline{x}) \quad (59)$$

$$r_{\text{out}}(\lambda) = w^*(\lambda, \underline{x}) - w_{\text{bottom}}(\alpha^*(\lambda, \underline{x})). \quad (60)$$

By definition, these functions are such that $r_{\text{in}}(\lambda_i(\underline{x})) = 0$ and $r_{\text{out}}(\lambda_o(\underline{x})) = 0$, where $\lambda_i(\underline{x})$ and $\lambda_o(\underline{x})$ define the extremities of the π-line containing $\underline{x}$. Therefore, using the first two terms of the Taylor series of $r_{\text{in}}(\lambda)$ and $r_{\text{out}}(\lambda)$ around $\lambda_i(\underline{x})$ and $\lambda_o(\underline{x})$, respectively, it is found that

$$\lambda_i(\underline{x}) \simeq \lambda + \frac{r_{\text{in}}(\lambda)\Delta\lambda}{r_{\text{in}}(\lambda - \Delta\lambda) - r_{\text{in}}(\lambda)} \qquad \text{if} \quad \lambda_i(\underline{x}) < \lambda < \lambda_i(\underline{x}) + \Delta\lambda \quad (61)$$

$$\lambda_o(\underline{x}) \simeq \lambda + \frac{r_{\text{out}}(\lambda)\Delta\lambda}{r_{\text{out}}(\lambda - \Delta\lambda) - r_{\text{out}}(\lambda)} \qquad \text{if} \quad \lambda_o(\underline{x}) - \Delta\lambda < \lambda < \lambda_o(\underline{x}). \quad (62)$$

These two formulae provide a way to get accurate estimates of the limits of integration, $\lambda_i(\underline{x})$ and $\lambda_o(\underline{x})$, for the cone-beam backprojection at $\underline{x}$. We suggest using these formulae with the trapezoidal rule for CB backprojection at $\underline{x}$. The outcome is that each sampled filtered projection should contribute a quantity

$$\delta f(\underline{x}) = \rho(\lambda, \underline{x}) \frac{\Delta\lambda}{2\pi v^*(\lambda, \underline{x})} g_{(c)}^F(\lambda, \alpha, w)\Big|_{\alpha=\alpha^*(\lambda, \underline{x}), w=w^*(\lambda, \underline{x})} \quad (63)$$

to the reconstruction, with

$$\rho(\lambda, \underline{x}) = \begin{cases} 0 & \text{if} \quad \lambda < \lambda_i(\underline{x}) - \Delta\lambda \\ (1 + d_{\text{in}})^2/2 & \text{if} \quad \lambda_i(\underline{x}) - \Delta\lambda < \lambda < \lambda_i(\underline{x}) \\ \frac{1}{2} + d_{\text{in}} - d_{\text{in}}^2/2 & \text{if} \quad \lambda_i(\underline{x}) < \lambda < \lambda_i(\underline{x}) + \Delta\lambda \\ 1 & \text{if} \quad \lambda_i(\underline{x}) + \Delta\lambda < \lambda < \lambda_o(\underline{x}) - \Delta\lambda \\ \frac{1}{2} + d_{\text{out}} - d_{\text{out}}^2/2 & \text{if} \quad \lambda_o(\underline{x}) - \Delta\lambda < \lambda < \lambda_o(\underline{x}) \\ (1 + d_{\text{out}})^2/2 & \text{if} \quad \lambda_o(\underline{x}) < \lambda < \lambda_o(\underline{x}) + \Delta\lambda \\ 0 & \text{if} \quad \lambda > \lambda_o(\underline{x}) + \Delta\lambda \end{cases} \tag{64}$$

where $d_{\text{in}} = (\lambda - \lambda_i(\underline{x}))/\Delta\lambda$ and $d_{\text{out}} = (\lambda_o(\underline{x}) - \lambda)/\Delta\lambda$.

To implement (63) most efficiently, we suggest, as before, looping first over x and y, then over z, with the loop over z restricted to the values of z such that $w_{\text{bottom}}(\alpha^*) < w^*(\lambda, \underline{x}) < w_{\text{top}}(\alpha^*)$. In this process, the filtered projections should always be kept in memory for three consecutive values of λ. That way, the contribution at λ such that $\lambda_i(\underline{x}) - \Delta\lambda < \lambda < \lambda_i(\underline{x})$ can be added when $\lambda_i(\underline{x}) < \lambda < \lambda_i(\underline{x}) + \Delta\lambda$, and similarly, the contribution at λ such that $\lambda_o(\underline{x}) < \lambda < \lambda_o(\underline{x}) + \Delta\lambda$ can be added when $\lambda_o(\underline{x}) - \Delta\lambda < \lambda < \lambda_o(\underline{x})$. This storage is necessary because the condition $w_{\text{bottom}}(\alpha^*) < w^*(\lambda, \underline{x}) < w_{\text{top}}(\alpha^*)$ on z prevents the computation of these contributions at the time the conditions $\lambda_i(\underline{x}) - \Delta\lambda < \lambda < \lambda_i(\underline{x})$ and $\lambda_o(\underline{x}) < \lambda < \lambda_o(\underline{x}) + \Delta\lambda$ are met. Note that to always be able to compute $\rho(\lambda, \underline{x})$ quickly, the values of $\alpha^*(\lambda, \underline{x})$, $v^*(\lambda, \underline{x})$ and $\eta(\lambda, \underline{x})$ at all (x, y) should be kept in memory for three consecutive values of λ along with the filtered projections.

Finally, since $g_{(c)}^F$ is only known at discrete sample locations, a preinterpolation in α and w is suggested so that in both, (58) and (63), $g_{(c)}^F$ at (α^*, w^*) can be obtained using the nearest neighbour interpolation, which is faster than bilinear interpolation.

5. Reconstruction in the flat detector panel geometry

Like the previous section, this section consists of three parts. In the first part, the steps to implement Katsevich's formula using the sampling geometry of the flat detector panel are given. In the second part, the number of detector rows needed for reconstruction using a given helix pitch is explained. In the last part, implementation strategies are discussed for some steps of the algorithm. A proof of the results given in this section can be found in the appendix.

5.1. The algorithm

Using the flat-panel geometry, the reconstruction is achieved in two steps.

Step 1: filtering. Each projection $g_{(f)}(\lambda, u, w)$ is modified into $g_{(f)}^F(\lambda, u, w)$ according to the following filtering steps:

FF1: derivative at constant direction. Compute from $g_{(f)}(\lambda, u, w)$, $g_1(\lambda, u, w)$ such that

$$g_1(\lambda, u, w) = g'(\lambda, \underline{\theta}_f) \tag{65}$$

with $\underline{\theta}_f$ given by equation (11) and $g'(\lambda, \underline{\theta}_f)$, the derivative of g with respect to λ at constant line direction, i.e.

$$g'(\lambda, \underline{\theta}_f) = \lim_{\varepsilon \to 0} \frac{g(\lambda + \varepsilon, \underline{\theta}_f) - g(\lambda, \underline{\theta}_f)}{\varepsilon} \tag{66}$$

(see equation (18)).

FF2: length-correction weighting. Compute

$$g_2(\lambda, u, w) = \frac{D}{\sqrt{u^2 + D^2 + w^2}} g_1(\lambda, u, w). \tag{67}$$

FF3: forward height rebinning. Compute for all $\psi \in [-\pi/2 - \alpha_m, \pi/2 + \alpha_m]$

$$g_3(\lambda, u, \psi) = g_2(\lambda, u, w_\kappa(u, \psi)) \tag{68}$$

with

$$w_\kappa(u, \psi) = \frac{DP}{2\pi R_o} \left(\psi + \frac{\psi}{\tan \psi} \frac{u}{D} \right). \tag{69}$$

Recall from section 4 that α_m is half the fan angle defined by the size of the FOV and the helix radius, i.e. $\alpha_m = \arcsin(R/R_o)$. At fixed ψ, the function $w_\kappa(u, \psi)$ defines a line in the detector plane, called κ-line of angle ψ, which is the intersection between the detector panel and the κ-plane $\mathcal{K}(\lambda, \psi)$

FF4: 1D Hilbert transform in u. Compute at constant ψ

$$g_4(\lambda, u, \psi) = \int_{-\infty}^{+\infty} du' h_H(u - u') g_3(\lambda, u', \psi) \tag{70}$$

where h_H is the kernel of the Hilbert transform (see equation (20)).

FF5: backward height rebinning. Compute

$$g_{(f)}^F(\lambda, u, w) = g_4(\lambda, u, \hat{\psi}(u, w)) \tag{71}$$

where $\hat{\psi}(u, w)$ is the angle ψ of the smallest absolute value that satisfies the equation

$$w = \frac{DP}{2\pi R_o} \left(\psi + \frac{\psi}{\tan \psi} \frac{u}{D} \right). \tag{72}$$

(Looking at steps FF3, FF4 and FF5 together, it appears clear that $g_{(f)}^F(\lambda, u_0, w_0)$ is obtained from a convolution of the data $g_2(\lambda, u, w)$ on the κ-line of the smallest $|\psi|$ value through (u_0, w_0).)

Step 2: backprojection. The filtered projection $g_{(f)}^F(\lambda, u, w)$ is backprojected to build f at each point $\underline{x} = (x, y, z)$ of the FOV according to the formula

$$f(\underline{x}) = \frac{1}{2\pi} \int_{\lambda_i(\underline{x})}^{\lambda_o(\underline{x})} d\lambda \frac{1}{v^*(\lambda, \underline{x})} g_{(f)}^F(\lambda, u^*(\lambda, \underline{x}), w^*(\lambda, \underline{x})) \tag{73}$$

where $\lambda_i(\underline{x})$ and $\lambda_o(\underline{x})$ define the extremities of the π-line through $\underline{x}$ with $\lambda_i(\underline{x}) < \lambda_o(\underline{x})$ and

$$v^*(\lambda, \underline{x}) = R_o - x \cos(\lambda + \lambda_0) - y \sin(\lambda + \lambda_0) \tag{74}$$

$$u^*(\lambda, \underline{x}) = \frac{D}{v^*(\lambda, \underline{x})} (-x \sin(\lambda + \lambda_0) + y \cos(\lambda + \lambda_0)) \tag{75}$$

$$w^*(\lambda, \underline{x}) = \frac{D}{v^*(\lambda, \underline{x})} \left(z - z_0 - \frac{P}{2\pi} \lambda \right). \tag{76}$$

5.2. Number of detector rows needed

A minimum number of detector rows is needed for the above algorithm to be applicable. Basically, there must be enough detector rows to compute the filtered data at any point in the Tam–Danielsson window, which is the detector region needed for the CB backprojection of equation (73). The Tam–Danielsson window is the set of (u, w) points such that

$$u \in [-u_m, u_m] \qquad w_{\text{bottom}}(u) \leqslant w \leqslant w_{\text{top}}(u) \tag{77}$$
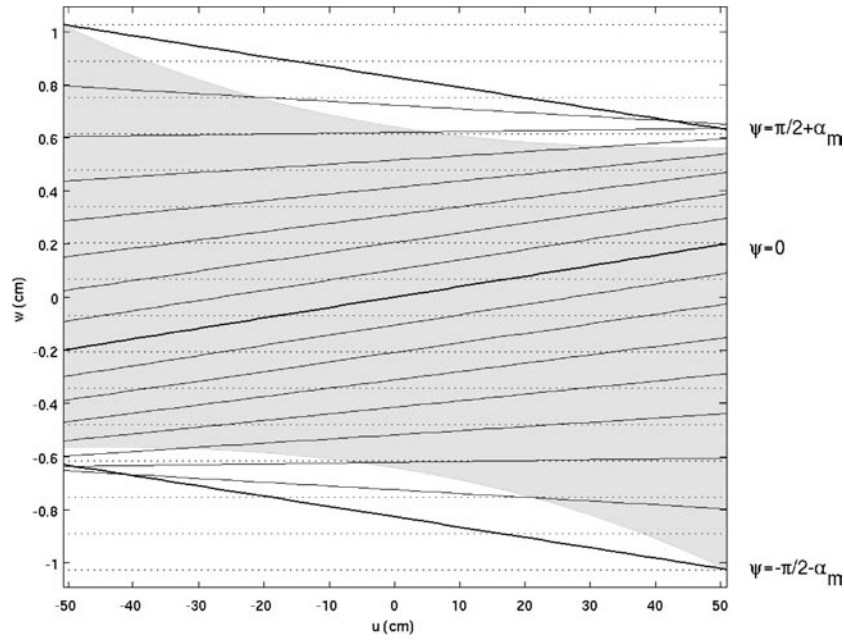
Figure 4. Illustration of κ-lines for a flat detector panel with 16 detector rows. The horizontal dotted lines correspond to the central line of each detector row.

with $u_m = D \tan \alpha_m$ and

$$
w_{\text{bottom}}(u) = -\frac{P}{2\pi R_o D}(u^2 + D^2)(\pi/2 + \arctan(u/D))
$$

$$
w_{\text{top}}(u) = \frac{P}{2\pi R_o D}(u^2 + D^2)(\pi/2 - \arctan(u/D)). \tag{78}
$$

To compute the filtered data at any point in the Tam–Danielson window, it is needed to know the data on any κ-line that intersects this window. As stated in Katsevich (2003), it is sufficient to consider $\psi \in [-\pi/2 - \alpha_m, \pi/2 + \alpha_m]$ as all points in the Tam–Danielson window belong to a κ-line of angle $\psi \in [-\pi/2 - \alpha_m, \pi/2 + \alpha_m]$. Figure 4 illustrates a few of these κ-lines. From this figure, it can be seen that the area between the detector lines $w = w_{\text{top}}(-u_m)$ and $w = w_{\text{bottom}}(u_m)$ is needed. Hence, the number of required detector rows is

$$
\begin{aligned}
N_{\text{rows}} &= \frac{d_w + w_{\text{top}}(-u_m) - w_{\text{bottom}}(u_m)}{d_w} \\
&= 1 + \frac{P}{\pi R_o D d_w}\left(u_m^2 + D^2\right)(\pi/2 + \arctan(u_m/D)) \tag{79}
\end{aligned}
$$

if d_w is the thickness of the detector rows. For example, with $R = 25$ cm, $R_o = 57$ cm and $R_o d_w/D = 0.75$ mm, the maximum allowable helix pitch for a 64-row detector panel is 5.92 cm.

5.3. Implementation strategies

As can be seen from a comparison of sections 4.1 and 5.1, the reconstruction steps for the flat detector are very similar to those given for the curved detector. The development

of implementation techniques for the flat detector is therefore pretty straightforward once implementation techniques are known for the curved detector. Actually, except for the filtering step FF1, the techniques of sections 4.3 only require minor changes to be applied to the flat detector. So, below we only discuss details for implementation of filtering step FF1. For this description, it is assumed that the data is sampled at

$$\lambda_k = k\Delta\lambda \qquad\qquad k = 0, \ldots, N_s - 1 \qquad (80)$$

$$u_i = (i + \delta - (N_{\text{cols}} - 1)/2)d_u \qquad i = 0, \ldots, N_{\text{cols}} - 1 \qquad (81)$$

$$w_j = (j - N_{\text{cols}}/2)d_w \qquad\qquad j = 0, \ldots, N_{\text{rows}} - 1 \qquad (82)$$

where d_u is the detector width while $\Delta\lambda$, δ, and d_w are defined as in section 4.3.

5.3.1. Filtering step FF1. The problem here is to compute samples of $g_1(\lambda, u, v)$ from the data $g_{(f)}(\lambda_k, u_i, w_j)$ according to equation (65). Following the same reasoning as in section 4.3, a first approach that can be taken is a direct discretization of (65) giving

$$g_1(\lambda_{k+1/2}, u, w) \simeq \frac{g(\lambda_{k+1}, \underline{\theta}_f) - g(\lambda_k, \underline{\theta}_f)}{\Delta\lambda} \qquad (83)$$

where $\underline{\theta}_f$ is given by equation (11), using $\lambda_{k+1/2}$ for λ. To apply this formula, it is needed to convert the values of g into values of $g_{(f)}$ using equations (12) and (13).

A second approach for the computation of $g_1(\lambda, u, v)$ is to apply the chain differentiation rule to equation (12) to obtain

$$g'(\lambda, \underline{\theta}) = \left(\frac{\partial g_{(f)}}{\partial \lambda} + \frac{\partial g_{(f)}}{\partial u} \frac{\partial u_f}{\partial \lambda} + \frac{\partial g_{(f)}}{\partial w} \frac{\partial w_f}{\partial \lambda} \right)(\lambda, \alpha_f, w_f). \qquad (84)$$

That is, upon the use of (13) for the expression of u_f and w_f,

$$g'(\lambda, \underline{\theta}) = \left(\frac{\partial g_{(f)}}{\partial \lambda} + \frac{u^2 + D^2}{D} \frac{\partial g_{(f)}}{\partial u} + \frac{uw}{D} \frac{\partial g_{(f)}}{\partial w} \right)(\lambda, \alpha_f, w_f). \qquad (85)$$

Or equivalently,

$$g_1(\lambda, \alpha, w) = g'(\lambda, \underline{\theta}_f) = \left(\frac{\partial g_{(f)}}{\partial \lambda} + \frac{u^2 + D^2}{D} \frac{\partial g_{(f)}}{\partial u} + \frac{uw}{D} \frac{\partial g_{(f)}}{\partial w} \right)(\lambda, u, w). \qquad (86)$$

From this last equation, we suggest computing g_1 at interlaced sample positions in λ, u and w, using the following formula for $k = 0, \ldots, N_s - 2$, $i = 0, \ldots, N_{\text{cols}} - 2$, $j = 0, \ldots, N_{\text{rows}} - 2$:

$$g_1(\lambda_{k+1/2}, u_{i+1/2}, w_{j+1/2}) \simeq \sum_{I=i}^{i+1} \sum_{J=j}^{j+1} \frac{g_1(\lambda_{k+1}, u_I, w_J) - g_1(\lambda_k, u_I, w_J)}{4\Delta\lambda}$$

$$+ \frac{u_{i+1/2}^2 + D^2}{D} \sum_{K=k}^{k+1} \sum_{J=j}^{j+1} \frac{g_1(\lambda_K, u_{i+1}, w_J) - g_1(\lambda_K, u_i, w_J)}{4d_u}$$

$$+ \frac{u_{i+1/2} w_{j+1/2}}{D} \sum_{K=k}^{k+1} \sum_{I=i}^{i+1} \frac{g_1(\lambda_K, u_I, w_{j+1}) - g_1(\lambda_K, u_I, w_j)}{4d_w}. \qquad (87)$$

This formula is the flat-detector version of formula (46) for the curved detector.

Table 2. Simulation parameters.

Number of detectors per row (N_{cols})	672
Detector height (d_w)	0.13684 cm
Detector width (d_α)	0.14083 cm
Quarter detector offset	yes
Maximum FOV radius (R)	25 cm
Focal spot size	0.09 cm $\times$ 0.12 cm
Source-to-detector distance (R_o)	104 cm
Source-to-rotation-axis distance (D)	57 cm
Number of projections per turn	1160

6. Evaluation

This section illustrates the performance of the techniques presented in section 4 for reconstruction in the curved detector geometry. The evaluation was carried out using geometrical parameters that are representative of current third-generation CT scanners. See table 2.

6.1. Visual image impression and artefacts

Simulated data of the FORBILD thorax phantom[3] were computed using analytical formulae for line integrals through all objects composing the phantom. For this computation, the detectors and the focal spot were divided into 3×3 subelements, then each data point was obtained using a realistic average of the 81 line integrals connecting individual subdetectors and subsources (that is the exponential of minus each line integral was computed and the data point was obtained as minus the log of the average of these exponentials). In doing so, three values were considered for the number of detector rows: 32, 64 and 128. In each case, the largest possible helix pitch was used (see table 1 for the precise values of the helix pitch), the initial z-position of the source was $z_0 = -5$ cm and the initial angular position of the source was $\lambda_0 = 0$ rad.

Reconstruction from the computer-simulated data was performed on a grid of $600 \times 384 \times 116$ cubic voxels of side 0.75 mm. The centre of the voxel with lowest x, lowest y and lowest z coordinates was at -22.46254 cm, -14.3625 cm and -2.6 cm, respectively. Three versions of the reconstruction algorithm were implemented to account for the two suggested ways to apply the filtering step CF1 and also the backprojection. The first version of the algorithm used the derivative (42) with the backprojection (63), the second version of the algorithm used the derivative (46) with the backprojection (63) and the third version of the algorithm used the derivative (46) with the backprojection (58). In each version, the Hilbert transform was applied without apodization and the backprojection was implemented with a preinterpolation factor equal to 8. Also, the radius of the FOV was 21 cm, and the number of κ-planes was defined with M equal to the number of detector rows. In version 3 of the algorithm, two values were considered for the regularization parameter a: $a = 0.5$ and $a = 0.125$.

Reconstruction results are displayed at four different locations (two axial slices, one coronal slice and one sagittal slice). Figure 5 illustrates the expected output, that is the artifact-free structures of the FORBILD thorax phantom in these slices, while figures 6 and 7 show reconstructions. In each case, the slices are displayed using a highly compressed

---

[3] See www.imp.uni-erlangen.de/forbild/english/results/index.htm for more information on FORBILD phantoms.
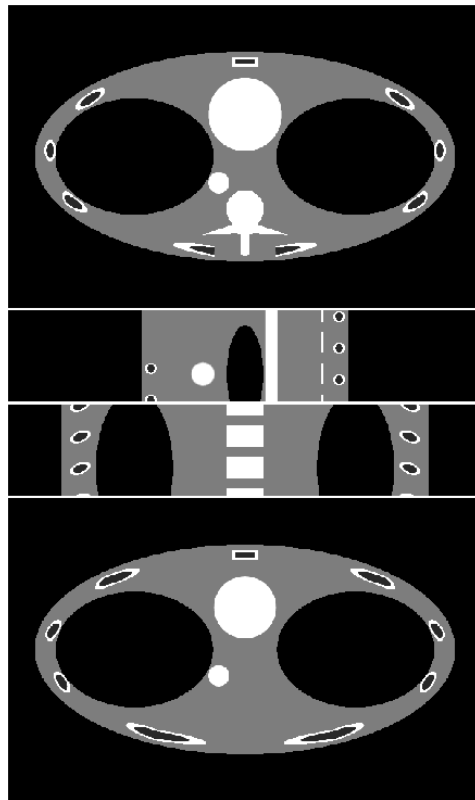
Figure 5. Original slices through the FORBILD thorax phantom. From top to left: slice $z = -0.35$ cm, slice $x = -3.3375$ cm, slice $y = -4.8375$ cm, and slice $z = 1.9$ cm. Each voxel value was obtained as the average of $3 \times 3 \times 3$ density values, taken at locations uniformly distributed inside the voxel.

greyscale of 60 HU centred on the 0 HU value. This compressed greyscale and the selection of most slices near object edges enhance the visibility of discretization artefacts.

Figure 6 compares reconstructions obtained with the three different versions of the algorithm for 64 detector rows. This figure shows that using the derivative (42) produces smoother images. This smoothing can be intuitively understood by noting that $\Delta\lambda = 4\Delta\alpha$ for the sampling parameters under consideration, so formula (42) involves samples that are separated by a distance $4\Delta\alpha$ in α. On the contrary, the derivative (46) retains the resolution in α. Formula (42) may be more natural than (46) as it follows more closely the definition of the derivative in Katsevich's formula. However, it does not allow resolution in α to be retained when the sampling rate in λ is low.

From figure 6, it also appears clear that the backprojection technique (58) is very sensitive to the selection of the regularization parameter a. For the reconstructions with version 3 of the algorithm, the χ-window was applied after the filtered data were preinterpolated by a factor 8. Under these conditions, $a = 0.125$ seemed to produce the best result (visually speaking). That is, it was found that the smooth transition from zero to one in χ_a should take place over no more than two pixels (after preinterpolation, the pixel width is $d_w/8$). Considering a larger value for a, such as 0.5, created a much smoother χ_a but resulted in significant streak artefacts. This observation suggests that Katsevich's formula is very sensitive to the way data
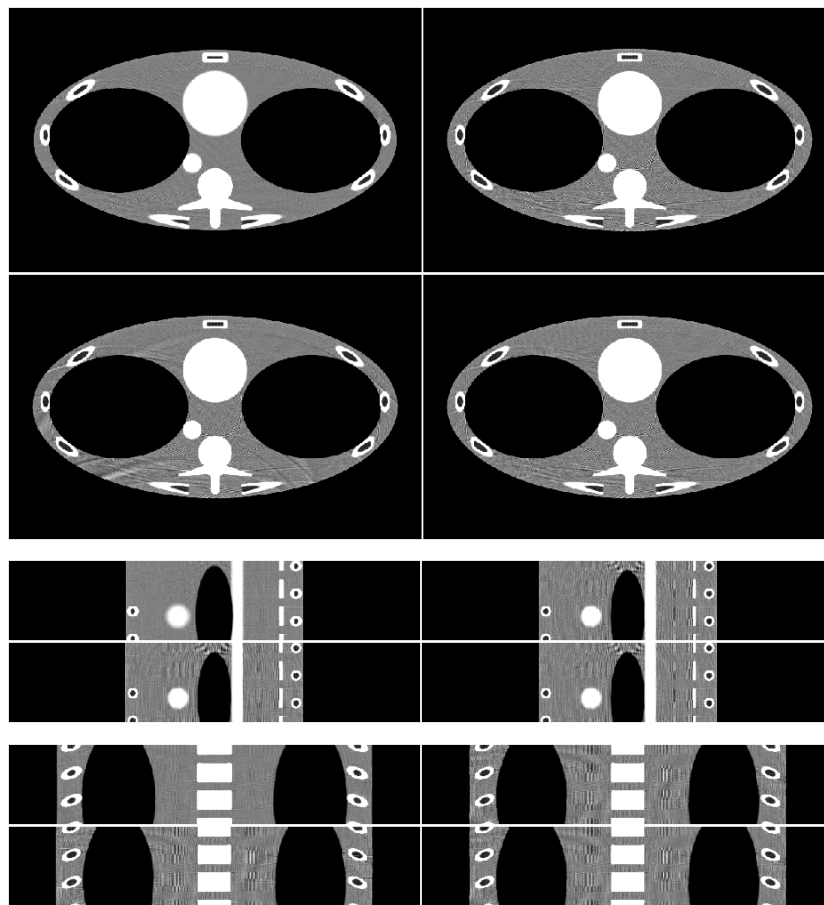
Figure 6. Reconstructions of the FORBILD thorax phantom in curved detector geometry with 64 detector rows, using three different versions of the algorithm. The first set of four images corresponds to the slice $z = -0.35$ cm, the second set to the slice $x = -3.3375$ cm and the last set to the slice $y = -4.8375$ cm. In each set, the top left reconstruction is obtained using (42) for the derivative and (63) for the backprojection the top right reconstruction is obtained using (46) for the derivative and (63) for the backprojection and the bottom reconstructions are obtained using (46) for the derivative and (58) for the backprojection, with $a = 0.5$ (bottom left) or $a = 0.125$ (bottom right). The images are displayed with a compressed greyscale of 60 HU centred on the 0 HU value.

outside the Tam–Danielsson window are incorporated in the reconstruction. As illustrated with backprojection (63), a voxel-dependent smoothing of χ appears more robust, while requiring little additional effort as discussed below.

Table 3 provides a comparison of required CPU times for versions 2 and 3 of the algorithm. For both versions, the total CPU time and the CPU time for filtering only (steps CF1 through CF6) are reported for optimized reconstruction on a PC Pentium IV (2.4 GHz). These CPU times are given for 32, 64 and 128 detector rows (with $R = 21$ cm and with maximum helix pitch in each case). Table 3 shows that, in the worst case, version 2 is only 2.5% slower than version 3, and that the filtering steps represent a significant fraction of the computational effort (25%, 34% and 44% for 32, 64 and 128 rows, respectively). This latter observation

Table 3. CPU times on a PC Pentium IV (2.4 GHz) for two different versions of the algorithm (see section 6.1 for details). The given times are for reconstruction on a grid of $600 \times 384 \times 116$ voxels in the curved detector geometry of table 2.

	Version 2		Version 3	
N_{rows}	Filtering time (s)	Total time (s)	Filtering time (s)	Total time (s)
32	415	1638	415	1608
64	497	1458	497	1431
128	607	1378	607	1344

is not surprising since the filtering steps involve a number of FFTs of length 2048. Table 3 also shows that, as the number of detector rows increases, the total CPU time decreases while the filtering CPU time increases. To understand these behaviours, it is worth noting that the number of projections processed for the given volume was 4224, 2510 and 1530 for 32, 64 and 128 detector rows, respectively. So, as the number of detector rows increases, more FFTs of length 2048 are required ($1530 \times 256 = 391\,680$ for 128 rows against $4224 \times 64 = 270\,336$ for 32 rows) and the filtering CPU time consequently increases. Conversely, as the number of detector rows increases, less projections are involved in the backprojection and the quantities (32) and (33), which are only dependent on x and y, are computed less often. So, there is a decrease in the backprojection effort and this decrease outweights the increase in filtering effort to provide a decrease in total CPU time.

Figure 7 compares reconstructions with 32, 64, and 128 detector rows using version 2 of the algorithm. These reconstructions show that some slight splay artefacts appear in the axial slices as the number of detector rows decreases. The magnitude of these artefacts was in general below 5 HU. However, this observation suggests that Katsevich's formula becomes more sensitive to discretization errors as the number of detector rows decreases (assuming the largest possible helix pitch is consistently used). Finally, note that when the number of detector rows is 128 the helix does not extend far enough in the z direction to provide sufficient data for reconstruction of all voxels; this is clearly visible in the bottom right image of figure 7.

6.2. Resolution measurements

The resolution obtained using version 2 of the algorithm was evaluated at maximum helix pitch for 32, 64 and 128 detector rows. This evaluation was achieved from data of a phantom made up from 20 spheres of radius 0.015 cm in the plane $z = 0$. The helix extended from $-P$ to P and crossed the plane $z = 0$ on the x-axis, while the spheres were uniformly distributed on two half-circles of centre $\underline{x} = 0$ in the region $y \geqslant 0$ (see figure 8). There were ten spheres per half-circle and the radii of the two half-circles were 8.8 cm and 17.6 cm. No spheres were placed in the region $y < 0$ since the choice of the helix implies a symmetry about the x-axis. Note that the detector size at the centre of the FOV is $Rd_\alpha/D = 0.075$ cm, which is five times larger than the radius of the spheres; the spheres approximate 3D Dirac (δ) impulses.

Computer-simulated data of the 20-sphere phantom was created with 14×14 subdetectors and 3×3 subsources. Next, each sphere was reconstructed on an individual (centred) grid of 50^3 voxels of side 0.01 cm. From each reconstruction, we determined the surfaces of points where the density was half the maximum value; these surfaces can be seen all together in figure 8 for 32 detector rows. Then, from these 'full-width-half-maximum (FWHM)' surfaces, we computed for each sphere the FWHM in z, the average FWHM in the (x, y)-plane, and
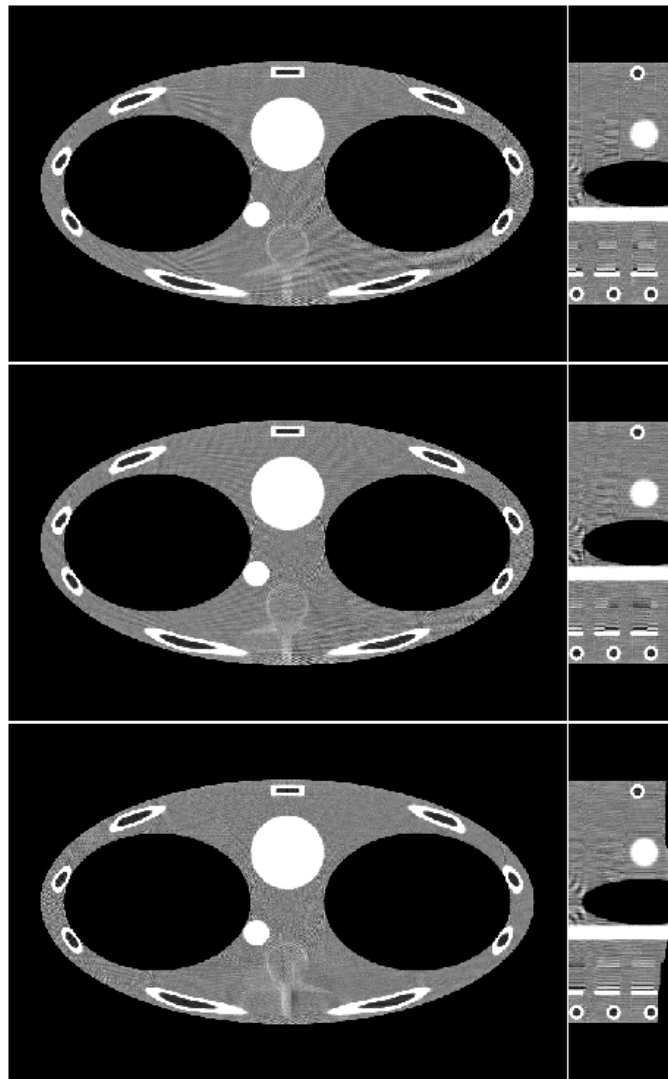
Figure 7. Reconstructions of the FORBILD thorax phantom in the curved detector geometry using the second version of the algorithm (i.e. (46) for the derivative and (63) for the backprojection) with 32 detector rows (top), 64 detector rows (middle) and 128 detector rows (bottom). *Left*: slice $z = 1.9$ cm, which is 1 mm away from the boundary of one vertebra. *Right*: slice $x = -3.3375$ cm. The images are displayed with a compressed greyscale of 60 HU centred on the 0 HU value.

the overall averaged FWHM with its standard deviation. The results are displayed in figure 9 (left column).

Figure 9 shows that the resolution at a given location is in general anisotropic and also that the resolution varies with both the radial position and the angular position in the FOV. In particular, as we step away from the centre of the FOV, the resolution gets more anisotropic and the spatial variance in the angular direction gets worse, especially with the number of detector rows increasing.
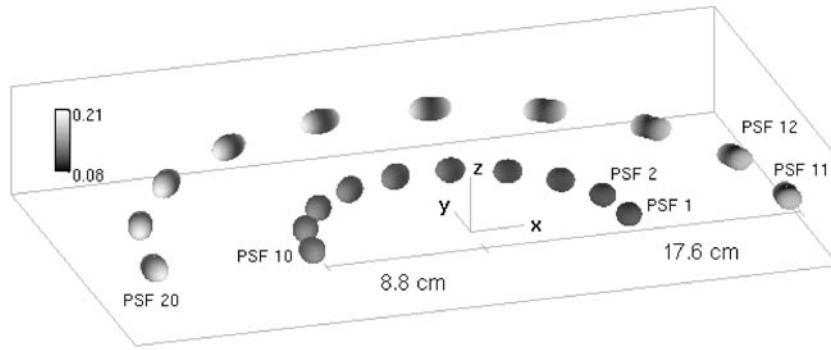
Figure 8. FWHM surfaces obtained from individual reconstructions of 20 spheres of radius 0.015 cm uniformly distributed on two half-circles within the field-of-view. The number of detector rows was 32.

The results in figure 9 are given for illustration of the behaviour of Katsevich's formula with data sampled in a conventional CT geometry. Conceptually, the resolution is affected by several factors including the detector size, the focal spot size, the source–position sampling, the selected discretization formulae, the magnification effect and the data range used for reconstruction at each location. Unfortunately, a complete analysis of how these factors interplay to produce the results in figure 9 is beyond the scope of this paper. However, the magnification effect seems to explain some important parts of the observed behaviour, as explained below.

On one hand, the magnification effect induces anisotropy in resolution. Indeed, due to the magnification effect, the resolution at a given location varies from one direction to another and since this difference is not accounted for in the frequency cut-off of the Hilbert transform (50) nor in the height rebinning steps, anisotropy in (x, y, z) is introduced (see section V.I.2 in Natterer (1986) for a discussion of this effect in 2D fan-beam tomography). As expected, this anisotropy gets more significant when moving away from the centre (compare the results for the spheres at 8.8 cm from the centre to those for the spheres at 17.6 cm). In figure 9, we see also that the resolution in z is better than the resolution in the (x, y)-plane, even though the detector height and detector width are comparable. These results are not surprising as we expect the z-resolution to be mostly governed by the 1D frequency response of the interpolation steps in w, while the (x, y)-resolution is mostly governed by a 2D frequency response (for the x-ray source far enough away and disregarding interpolation effects, the z-resolution would be that of a sinc-function, while the (x, y)-resolution would be that of a jinc-function[4]).

On the other hand, since a different angular range of data is used for reconstruction at different locations, the average magnification effect varies spatially, and since this variation is not accounted for in the data processing, a spatially varying resolution is introduced in the reconstruction. For example, spheres 1 and 10 are reconstructed from the data with $|\lambda| < \pi/2 - \eta$ and $|\lambda| < \pi/2 + \eta$, respectively, where $\eta = \arcsin(8.8/57)$, so the average magnification effect is larger for sphere 1 than sphere 10 and indeed the resolution for sphere 1 is better than for sphere 10. Unfortunately, with increasing distance from the centre, additional effects seem to enter into play and a complete understanding becomes more complicated.

---

[4] The FWHM of $\text{sinc}(x) = \sin(\pi x)/(\pi x)$ is 0.6 while the FWHM of $\text{jinc}(x) = J_1(\pi x)/(\pi x)$ is 0.7, where J_1 is the Bessel function of order 1.
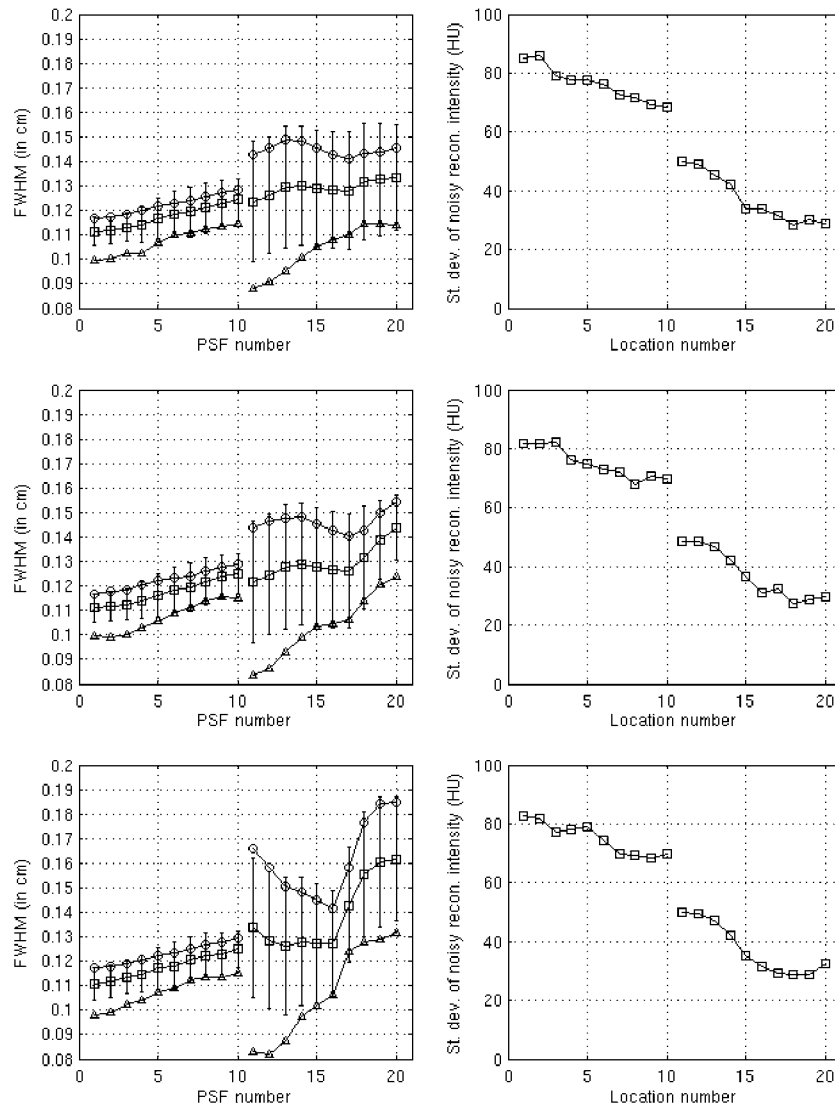
Figure 9. *Left*: resolution measurements obtained from FWHM reconstructed surfaces of the 20 spheres displayed in figure 8. The in-plane (x, y) resolution is shown with circles, while the z-resolution is shown with triangles and the average resolution with squares and added standard deviation bars. *Right*: noise measurements (standard deviation in density value) at the location of the spheres for a water cylinder of radius 20 cm with 200 000 photons per data point. *Top*: 32 detector rows. *Middle*: 64 detector rows. *Bottom*: 128 detector rows.

Recall from (53) that the Hilbert transform is applied with a half-pixel shift. We would like to point out at this stage that this half-pixel shift has an important impact on resolution. Figure 10 shows the resolution measurements obtained for 32 detector rows when the Hilbert transform is applied without half-pixel shift; a significant loss of in-plane resolution can be observed in comparison with figure 9.
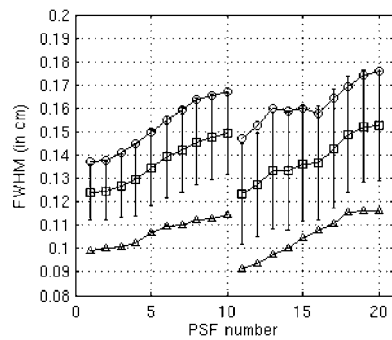
Figure 10. Resolution measurement for 32 detector rows when the Hilbert transform is applied without half-pixel shift.

6.3. Noise measurements

Noise measurements were achieved using a uniform cylinder of radius 20 cm centred on the z-axis. Computer-simulated data of that cylinder was created with Poisson noise based on an emission of 200 000 photons per data point (using 3×3 subdetectors and subsources). Reconstructions were then performed on individual grids of 10^3 cubic voxels of side 0.75 mm centred on the locations of the spheres used for resolution measurements. The standard deviation in the density values obtained on these individual grids are displayed in figure 9 (right column). For each subset of ten locations at same distance from the FOV centre, the results appear fairly consistent with the FWHM measurements in the left column of figure 9; the noise is higher at locations where the FWHM is smaller and vice versa. The drop in standard deviation from the first subset to the other is caused by the change in average path length, which is shorter at the locations that are most away from the centre.

7. Discussion and conclusions

Helical cone-beam reconstruction using Katsevich's formula (Katsevich 2002a) has been investigated. Katsevich's formula represents a breakthrough for multi-slice helical CT. However, it was published with very few implementation details, using detector-independent notations. Reconstructions results were reported later in Köhler *et al* (2002) and also in Katsevich *et al* (2003), but again with very few implementation details. These results demonstrated the accuracy of Katsevich's formula but let the subject of efficient implementation in native detector geometries be one of further investigation. This paper presented the results of such an investigation for two native detector geometries. In a first stage, processing steps that can be applied directly to the data have been determined for reconstruction. Next, implementation strategies taking into account data sampling have been suggested. These strategies have been implemented and their performance has been illustrated. The implementation strategies appear relatively efficient as they use only linear interpolation methods, 1D convolution and z-independent backprojection weighting. The performance was illustrated using three versions of the algorithm for reconstruction from data in the curved detector geometry of third-generation CT scanners. The version 2, which uses the derivative (46) and the backprojection (63), was preferred. However, results with the other versions were also displayed to illustrate the sensitivity of Katsevich's formula to implementation strategies. Regarding that aspect (sensitivity), the initial differentiation step and the backprojection over

the π-range of each voxel were found to be the steps the most prone to resolution loss and/or discretization errors.

As mentioned in section 3, different definitions can be taken for the κ-planes. The definition selected in this paper seems to be the most natural one and was also the one selected in Köhler *et al* (2002) and in Katsevich *et al* (2003). At this stage, no clear arguments could be found in favour of using another definition.

In Katsevich *et al* (2003), an alternative expression of Katsevich's formula was used for reconstruction. This alternative expression avoids the derivative with respect to the source position and was thought to be more appropriate for reconstruction with the number of projections usually encountered in CT (Katsevich 2002a). However, our algorithm, which uses the derivative, provides results that are at least as good as those published in Katsevich *et al* (2003), with the advantage of being computationally more efficient. Indeed, our algorithm involves a single filtered projection and a single backprojection, while the algorithm in Katsevich *et al* (2003) involves four different filtered projections and also two different backprojections. It seems to us that an efficient implementation of the formula in Katsevich *et al* (2003) would require twice more time for filtering plus an additional fraction of effort in backprojection, and could therefore be 1.5 to 2 times slower.

The experimental results of section 6 were obtained using a quarter detector offset for the data simulation. However, this data acquisition feature offers practically no resolution benefits for reconstruction with Katsevich's formula because the average angular range of data used for reconstruction at each voxel is $180°$. The quarter detector offset is often used in clinical applications to obtain an increased resolution and therefore there is a need to find an exact reconstruction formula that is as attractive as Katsevich's but can benefit from quarter detector offset. Besides the quarter detector offset, another feature often used to gain resolution is the so-called flying focal spot (Kalender 2000). This feature was not considered in this paper but should not be difficult to use with little, if any, changes in our implementation of Katsevich's formula.

The implementation techniques described in sections 5.3 and 6.3 are only suggestions. It is possible that more efficient (in speed and accuracy) methods can be found. Actually, it is hoped that this work will help to achieve further developments. In particular, we are currently investigating the possibility of processing the data in wedge geometry as defined below, to reduce the shift-variance of the resolution and increase the reconstruction speed. The following result has already been obtained for the curved detector geometry of section 5.

Let $g_{PB}^F(\theta, u, w)$ represents filtered data rebinned into wedge geometry, i.e.

$$\frac{1}{\cos^2 \alpha} g_{(c)}^F(\lambda, \alpha, w) = g_{PB}^F(\theta = \lambda - \alpha, u = R_o \sin \alpha, w). \quad (88)$$

Using $g_{PB}^F(\theta, u, w)$, the backprojection equation (31) can be rewritten in the form

$$f(\underline{x}) = \frac{1}{2\pi R_o} \int_{\theta_i(\underline{x})}^{\theta_o(\underline{x})} \mathrm{d}\theta \, g_{PB}^F(\theta, u^*, w^*) \quad (89)$$

with $u^* = R_o \sin \alpha^*$, $\theta_i(\underline{x}) = \lambda_i(\underline{x}) - \alpha^*(\lambda_i(\underline{x}), \underline{x})$ and $\theta_o(\underline{x}) = \lambda_o(\underline{x}) - \alpha^*(\lambda_o(\underline{x}), \underline{x})$.

That is, upon rebinning to wedge geometry, the weighting factor in the backprojection disappears, thereby allowing possibly a more efficient implementation with less shift-variance in resolution. Preliminary results using wedge rebinning can be found in Köhler *et al* (2002).

Acknowledgments

Appendix

This appendix shows how the algorithms of sections 4 and 5 can be derived from the reconstruction formula of Katsevich. The algorithm of section 5, which is relative to the flat array of detectors, is first derived directly from Katsevich's formula. Next, the algorithm for the curved detector, which is given in section 4, is derived using the change of variables defined by equation (14).

A.1. Flat array of detectors

First, we determine the intersection between the detector plane at position λ and $\mathcal{K}(\lambda, \psi)$, the κ-plane of angle ψ at $\underline{a}(\lambda)$. This intersection was referred to as κ-line of angle ψ in section 5.

Recall that $\mathcal{K}(\lambda, \psi)$ is the plane that contains $\underline{a}(\lambda)$, $\underline{a}(\lambda + \psi)$ and $\underline{a}(\lambda + 2\psi)$. To find the equation of the κ-line of angle ψ, we calculate the coordinates of two points that belong to this line. The first point, (u_1, w_1), is the intersection of the detector plane with the line that join $\underline{a}(\lambda)$ to $\underline{a}(\lambda + \psi)$. The second point, (u_2, w_2) is the intersection of the detector plane with the line that join $\underline{a}(\lambda)$ to $\underline{a}(\lambda + 2\psi)$. Using equation (1) and the rotated axes of equations (3), (4) and (5), we find

$$\underline{a}(\lambda + \psi) - \underline{a}(\lambda) = R_o \sin \psi \, \underline{e}_u(\lambda) + R_o(1 - \cos \psi)\underline{e}_v(\lambda) + (P\psi)/(2\pi)\underline{e}_w. \tag{90}$$

So, from (13), with $\underline{a}(\lambda + \psi) - \underline{a}(\lambda)$ used for $\underline{\theta}$,

$$u_1 = D\frac{\sin \psi}{1 - \cos \psi} \qquad w_1 = \frac{DP}{2\pi R_o}\frac{\psi}{1 - \cos \psi} \tag{91}$$

and consequently

$$u_2 = D\frac{\sin 2\psi}{1 - \cos 2\psi} \qquad w_2 = \frac{DP}{2\pi R_o}\frac{2\psi}{1 - \cos 2\psi}. \tag{92}$$

From these coordinates, it is straightforward to find the equation of the κ-line of angle ψ:

$$w_\kappa = \frac{DP}{2\pi R_o}\left(\psi + \frac{\psi}{\tan \psi}\frac{u}{D}\right) \tag{93}$$

as announced by equation (69).

Now, we focus on the expression of the filtered data $g^F(\lambda, \underline{\theta})$ for a given direction $\underline{\theta}$. Following equation (19), we consider the line of direction $\underline{\beta} = \cos \gamma \underline{\theta} + \sin \gamma (\underline{\theta} \times \underline{m}(\lambda, \underline{\theta}))$ through $\underline{a}(\lambda)$. By construction, this line sweeps a κ-plane as γ varies, so its intersection with the detector plane moves on a κ-line as γ varies. We denote this intersection as (u', w'_κ) and let (u, w_κ) be the intersection between the detector plane and the line of direction $\underline{\theta}$ through $\underline{a}(\lambda)$, so that $(u', w'_\kappa) = (u, w_\kappa)$ for $\gamma = 0$. See figure 11. Referring to the parameters in that figure, we note from trigonometric properties of triangle that $lL = rr' \sin \gamma$ and $l\Delta L = r'r'' \sin \Delta\gamma$. Hence,

$$\frac{\Delta L}{L} = \frac{r''}{r}\frac{\sin \Delta\gamma}{\sin \gamma}. \tag{94}$$

But, (u, w_κ) and (u', w'_κ) belong to a same (κ-) line, whatever γ, so $\Delta L/L = \Delta u'/(u' - u)$ and (94) is equivalent to

$$\frac{\Delta u'}{u' - u} = \frac{r''}{r}\frac{\sin \Delta\gamma}{\sin \gamma}. \tag{95}$$

Viewing $\Delta\gamma$ as an infinitesimal quantity, this relation becomes

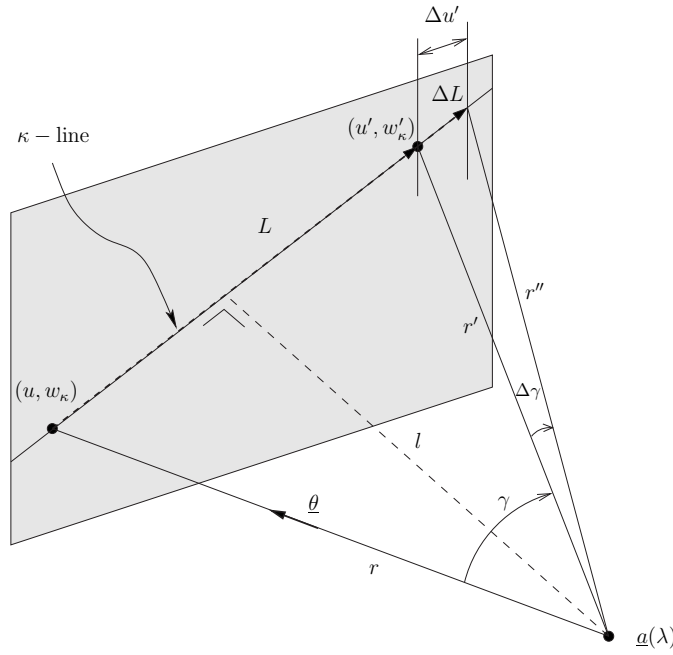$$\frac{du'}{u' - u} = \frac{r'}{r}\frac{d\gamma}{\sin \gamma} \tag{96}$$

Figure 11. Link between parameter γ in Katsevich's formula and flat detector coordinates.

that is

$$h_H(\sin\gamma)\,\mathrm{d}\gamma = -\frac{r}{r'}h_H(u-u')\,\mathrm{d}u' \tag{97}$$

where h_H is the kernel of the Hilbert transform given in equation (20)), $r = \left(u^2 + D^2 + w_\kappa^2\right)^{1/2}$, and $r' = ((u')^2 + D^2 + (w'_\kappa)^2)^{1/2}$ (as illustrated in figure 11, r and r' are the distances between $\underline{a}(\lambda)$ and the detector points (u, w) and (u', w'), respectively).

The insertion of (97) into (19) yields

$$g^F(\lambda, \underline{\theta}) = -\frac{r}{D}\int_{-\infty}^{+\infty}\mathrm{d}u'\,h_H(u-u')\frac{D}{r'}g_1(\lambda, u', w'_\kappa) \tag{98}$$

with g_1 given by (65). Comparing this expression with the expanded formulae of the algorithm in section 5.1, it appears clear that

$$g^F(\lambda, \underline{\theta}) = -\frac{r}{D}g^F_{(f)}(\lambda, u, w_\kappa) \tag{99}$$

where $g^F_{(f)}$ is the filtered projection of equation (71). To completely establish the algorithm of section 5.1, it remains only to note that

$$\frac{r}{D\|\underline{x}-\underline{a}(\lambda)\|} = \frac{1}{(\underline{x}-\underline{a}(\lambda))\cdot\underline{e}_v(\lambda)} \tag{100}$$

when $\underline{\theta} = (\underline{x}-\underline{a}(\lambda))/\|\underline{x}-\underline{a}(\lambda)\|$. So,

$$\frac{-1}{2\pi\|\underline{x}-\underline{a}(\lambda)\|}g^F\left(\lambda, \frac{\underline{x}-\underline{a}(\lambda)}{\|\underline{x}-\underline{a}(\lambda)\|}\right) = \frac{1}{2\pi(\underline{x}-\underline{a}(\lambda))\cdot\underline{e}_v(\lambda)}g^F_{(f)}(\lambda, u, w_\kappa)$$

$$= \frac{1}{2\pi v^*(\lambda, \underline{x})}g^F_{(f)}(\lambda, u, w_\kappa) \tag{101}$$

where $v^*(\lambda, \underline{x})$ is as in equation (74).

A.2. Curved array of detectors

The algorithm for the curved array is obtained directly from the equations of the algorithm for the flat array by replacing w by $w/\cos\alpha$ and u by $D\tan\alpha$, as suggested by equation (14). Following this substitution, the equation of the κ-line (93) becomes

$$w_\kappa = \frac{DP}{2\pi R_o}\cos\alpha\left(\psi + \frac{\psi}{\tan\psi}\tan\alpha\right) = \frac{DP}{2\pi R_o}\left(\psi\cos\alpha + \frac{\psi}{\tan\psi}\sin\alpha\right) \tag{102}$$

which is the equation of the κ curve of equation (26). Similarly, $D/((u')^2 + D^2 + (w')^2)^{1/2}$ becomes $D\cos\alpha'/(D^2 + (w')^2)^{1/2}$ and

$$u - u' = D(\tan\alpha - \tan\alpha') = D\frac{\sin(\alpha - \alpha')}{\cos\alpha\cos\alpha'} \tag{103}$$

so that

$$h_H(u - u')\,\mathrm{d}u' = h_H(\sin(\alpha - \alpha'))\frac{\cos\alpha\cos\alpha'}{D}\frac{D}{\cos^2\alpha'}\,\mathrm{d}\alpha'$$
$$= h_H(\sin(\alpha - \alpha'))\frac{\cos\alpha}{\cos\alpha'}\,\mathrm{d}\alpha'. \tag{104}$$

The combination of all these results together yields the equations of the algorithm for the curved array from the equations for the flat array.

References

Chen L, Heuscher D J and Liang Y 2000 Oblique surface reconstruction to approximate cone-beam spiral data in multi-slice CT *Proc. SPIE* **4123** 279–84

Crawford C R and King K F 1990 Computed tomography scanning with simultaneous patient translation *Med. Phys.* **17** 967–82

Danielsson P E, Edholm P, Eriksson J and Magnusson Seger M 1997 Towards exact reconstruction for helical cone-beam scanning of long objects. A new detector arrangement and a new completeness condition *Proc. 1997 Meeting on Fully 3D Image Reconstruction in Radiology and Nuclear Medicine (Pittsburgh, PA)* ed D W Townsend and P E Kinahan pp 141–4

Defrise M, Noo F and Kudo H 2000 A solution to the long-object problem in helical cone-beam tomography *Phys. Med. Biol.* **45** 623–43

Feldkamp L A, Davis L C and Kress J W 1984 Practical cone-beam algorithm *J. Opt. Soc. Am.* A **6** 612–9

Heuscher D J 1999 Spiral cone beam scans using 2D surface reconstructions *Proc. 1999 Int. Meeting on Fully 3D Image Reconstruction* pp 204–7

Kachelriess M, Schaller S and Kalender W A 2000 Advanced single-slice rebinning in cone-beam spiral CT *Med. Phys.* **27** 754–72

Kalender W A 2000 *Computed Tomography* (Publicis MCD)

Katsevich A I 2002a Analysis of an exact inversion algorithm for spiral cone-beam CT *Phys. Med. Biol.* **47** 2583–97

Katsevich A I 2002b Microlocal analysis of an FBP algorithm for truncated spiral cone-beam data *J. Fourier Anal. Appl.* **8** 407–25

Katsevich A I 2003 An improved exact filtered backprojection algorithm for spiral computed tomography *Adv. Appl. Math.* at press

Katsevich A I, Lauritsch G, Bruder H, Flohr T and Stierstorfer K 2003 Evaluation and empirical analysis of an exact FBP algorithm for spiral cone-beam CT *Proc. SPIE* **5032** 663–74

Köhler Th, Bontus C, Brown K, Heuscher D, Grass M, Shechter G and Proksa R 2002 Evaluation of helical cone-beam CT reconstruction algorithms *Conference Record of the 2002 IEEE Med. Imaging Conf. (Norfolk)* at press

Kudo H, Noo F and Defrise M 2000 Quasi-exact filtered backprojection algorithm for long-object problem in helical cone-beam tomography *IEEE Trans. Med. Imaging* **19** 902–21

Larsson G L, Ruth C C and Crawford C R 1998 Nutating slice CT image reconstruction: apparatus and method *US Patent* 5802134

Lauritsch G, Tam K C, Sourbelle K and Schaller S 2000 Exact local region-of-interest reconstruction in spiral cone-beam filtered-backprojection CT: numerical implementation and first image results *Proc. SPIE* **3979** 520–32

Natterer F 1986 *The Mathematics of Computerized Tomography (Classics in Applied Mathematics 32)* (SIAM)

Schaller S, Flohr T, Klingenbeck K, Kraus J, Fuchs T and Kalender W A 2000 Spiral interpolation algorithm for multi-slice spiral CT. Part 1: theory *IEEE Trans. Med. Imaging* **19** 822–34

Schaller S, Noo F, Sauer F, Tam K C, Lauritsch G and Flohr T 2000 Exact Radon rebinning algorithm for the long object problem in helical cone-beam CT *IEEE Trans. Med. Imaging* **19** 361–75

Sourbelle K 2002 Performance evaluation of exact and approximate cone-beam algorithms in spiral computed tomography *PhD Thesis* Erlangen University, Germany

Taguchi K and Aradate H 1998 Algorithm for image reconstruction in multi-slice helical CT *Med. Phys.* **25** 550–61

Tam K C 2000 Exact local regions-of-interest reconstruction in spiral cone-beam filtered-backprojection CT. Theory *Proc. SPIE* **3979** 506–19

Tam K C, Lauritsch G and Sourbelle K 2002 Filtering point spread function in backprojection cone-beam CT and its applications in long object imaging *Phys. Med. Biol.* **47** 2685–703

Tam K C, Samarasekera S and Sauer F 1998 Exact cone-beam CT with a spiral scan *Phys. Med. Biol.* **43** 1015–24

Turbell H 1999 Three-dimensional image reconstruction in circular and helical computed tomography *Tech. Rep. Lic. Thesis no. 760* Linköping University, Sweden