

Exact Fan-Beam Reconstruction With Arbitrary Object Translations and Truncated Projections

Jan Hoskovec, Rolf Clackdoyle, Laurent Desbat, and Simon Rit

Abstract—This article proposes a new method for reconstructing two-dimensional (2D) computed tomography (CT) images from truncated and motion contaminated sinograms. The type of motion considered here is a sequence of rigid translations which are assumed to be known. The algorithm first identifies the sufficiency of angular coverage in each 2D point of the CT image to calculate the Hilbert transform from the local “virtual” trajectory which accounts for the motion and the truncation. By taking advantage of data redundancy in the full circular scan, our method expands the reconstructible region beyond the one obtained with chord-based methods. The proposed direct reconstruction algorithm is based on the Differentiated Back-Projection with Hilbert filtering (DBP-H). The motion is taken into account during back-projection which is the first step of our direct reconstruction, before taking the derivatives and inverting the finite Hilbert transform. The algorithm has been tested in a proof-of-concept study on Shepp-Logan phantom simulations with several motion cases and detector sizes.

Index Terms—Computed tomography, image reconstruction, motion compensation, reconstruction algorithms.

I. INTRODUCTION

IN CLASSICAL, two-dimensional (2D) computed tomography (CT), image reconstruction consists of recovering an unknown image from its measured ray-sums in the plane. Traditionally, the reconstruction procedure, as introduced in the 1970s and deployed on clinical scanners, consists of back-projecting the sinogram data which have been processed by a ramp filter [1]. Multiple issues can cause reconstruction artifacts, undermining the image quality. The ones we address here are object motion and incomplete data due to lateral truncation of the acquired projections. If these issues are ignored, the former will blur structures present in the object and the latter will usually create a bright ring at the truncation edge and low-frequency artifacts throughout the reconstructed image.

Manuscript received August 18, 2015; revised November 12, 2015; accepted December 23, 2015. Date of publication May 24, 2016; date of current version June 21, 2016. This work was supported in part by a grant from la Région Rhône-Alpes and by grant ANR-12-BS01-0018 (DROITE project) from the Agence Nationale de la Recherche (France).

J. Hoskovec is with Laboratoire Hubert Curien, CNRS UMR 5516, Université Jean Monnet, Saint-Étienne, France, and also with CREATIS, CNRS UMR 5220, Inserm U1206, INSA-Lyon, Université Claude Bernard-Lyon 1, Centre Léon Bérard, Lyon, France.

R. Clackdoyle is with Laboratoire Hubert Curien, CNRS UMR 5516, Université Jean Monnet, Saint-Étienne, France.

L. Desbat is with TIMC-IMAG, CNRS UMR 5525, Université Grenoble Alpes, Grenoble, France.

S. Rit is with Université de Lyon, CREATIS, CNRS UMR 5220, Inserm U1206, INSA-Lyon; Université Claude Bernard-Lyon 1, Centre Léon Bérard, Lyon, France (e-mail: simon.rit@creatis.insa-lyon.fr).

Digital Object Identifier 10.1109/TNS.2016.2519242

With traditional ramp-filtered backprojection, successful reconstruction requires that every ray-sum of image space is measured or known to be zero. Missing non-zero ray-sums will cause errors in the ramp-filtered sinogram, because the ramp filter has infinite support. In the early 2000s, new reconstruction methods were proposed [2]–[7], broadly based on the idea of splitting the ramp filter into its components of the derivative and the Hilbert transform. Replacing the ramp filter by these two steps makes it possible to perform a local operation (the derivative) before backprojecting the data whereas the large support operation (Hilbert transform) takes place in the image space. Thus, in the backprojected image, missing data only affect the regions where the data are missing, and with specialized techniques for handling truncated Hilbert transforms, mathematically correct region-of-interest (ROI) reconstruction becomes possible. This family of reconstruction algorithms is variously referred to as DBP-H (Differentiated Back-Projection with Hilbert filtering), DBP or BPF (Back-Projection–Filtration). It was later shown [8], [9] that one can take this approach a step further, and also perform the derivative in image space. With this strategy, the backprojection step is applied to the raw sinogram data, which provides more flexibility for coping with unusual scanning geometries, as we will demonstrate here.

Compensation techniques for motion artifacts have been investigated since computed tomography became widespread in the 1980s. Early approaches consisted of removing motion artifacts during post-processing [10]. Later, a method appeared which exploits sinogram features to remove those due to object motion [11]. Current methods typically incorporate motion properties into the reconstruction formula, using a parametric motion model [12]–[16] or a vector field describing the measured motion [17], [18]. Some methods also estimate the motion parameters from sinogram properties [19], [20], whereas others only deal with the motion-compensated reconstruction, assuming the motion to be known, for example, from measurements taken on an external device [13], [16], [21]. In this work, we are not considering deformations such as caused by patient respiration. We only address *rigid* motions of the object being scanned.

A scan of an object undergoing rigid translation is equivalent to that of a stationary object with a perturbed, “virtual” source trajectory. Therefore, works dealing with perturbed source trajectories [19]–[22] also fall into the scope of motion compensation. Recently, Kim et al. [21] used an iterative algorithm to account for virtual trajectories arising from motion-corrected helical CT acquisitions. However, we are concerned with analytic algorithms in this work. Xia et al. [22] proposed a method of chord-based reconstruction from a helical

scan with perturbed source trajectory. Chord (or PI-line) based algorithms are based on the ideas of DBP-H algorithms in three-dimensional (3D) reconstructions from scans obtained with a helical or saddle-curve, rather than circular, source paths [5], [22], [23]. Dealing with perturbed source paths generally consists of including the modification of the trajectory in the reconstruction formulas. In Xia's case [22], the source trajectory perturbation was included in the weighting scheme used at backprojection, as well as in the choice of the sinogram coordinates at which the derivative was calculated for a given point on the reconstructed chord. The sinogram data derivative was taken with respect to the gantry angle, which is typical for algorithms working in three dimensions and does not favor correction of motion-induced artifacts. Xia's method could easily be collapsed to fan-beam if one chooses the chords on which to backproject beforehand (these chords are unique for a given voxel in the case of a helical scan, but there is an infinite number of them for any planar scan, such as a circular scan). Yu *et al.* [19], [20] explicitly address object motion which they also model as a source trajectory perturbation. In [19], two fan-beam reconstruction algorithms—one based on the conventional FBP, one of a DBP-H type—were proposed. The DBP-H style algorithm is equivalent to that of [22] collapsed into 2D, but with a different approach for the derivative, given as a weighted sum of the partial derivatives with respect to the gantry angle and to the on-detector position (just like in [2]). In [20], compensation for rotation of the object around the origin was added to the problem; otherwise the reconstruction method was the same as in [19]. This rotation was treated as a transformation of the gantry angle at both the derivative and backprojection steps.

This work proposes a method of exact reconstruction from a standard 2D fan-beam full scan of an object undergoing a sequence of arbitrary translations in the plane of the source trajectory. Its principal interest is to analytically handle both motion compensation and projection truncation, particularly truncation arising from the motion. Previous methods, such as those of [19], [20], [22], could be adjusted to achieve these objectives but our work includes several novel features. One is that the region of exact (mathematically correct) reconstruction can be automatically indicated. Another feature is an increased capability of our algorithm by mathematically recasting the measurements into parallel projections before applying the DBP-H method thereby overcoming certain limitations of chord-based methods applied to fan-beam projections. A third feature of our work is that the derivative is performed in image space (as proposed by Zeng [8]), so the backprojection to be performed as the first step, which simplifies the motion correction. A fourth novel feature is that the truncation arising from the motion is identified analytically and independently of the measured fan-beam projections.

We make no claim in this paper that our method in its current form has specific real-world applications, such as rigid patient motion correction in clinical CT. Our work lies entirely in the two-dimensional context. However, we feel that analytic methods for motion correction in image reconstruction is a worthwhile endeavor because analytic methods often provide more insight on important issues such as data sufficiency throughout the reconstruction zone. Data sufficiency becomes

a complicated issue when arbitrary rigid motions are considered, because the effective source trajectory is altered, and needs to be checked for tomographic completeness at each point. Furthermore the motion can generate truncation of some regions of the object which further complicates the data completeness issue. The contribution of this work is a step in the direction of analytic correction of arbitrary 3D rigid motions such as a clinical CT scan, or any scanning situation where the object undergoes a known rigid motion.

The paper is organized as follows: In Section II, we establish the notation, review standard DBP-H reconstruction theory and repeat the derivation of Zeng [8] for the backproject-first variation. We also introduce our notation for arbitrary displacements during the 2π fan-beam scan. In Section III, we describe the two distinct components of the algorithm in conceptual terms. The first part concerns an analysis of the truncation pattern for each point, thus establishing whether or not a "Hilbert point" can be computed there, independent of the sampling of the fan-beam sinogram. The second part involves finding lines of Hilbert points, along which the finite inverse Hilbert transform can be computed. The calculation of Hilbert points is described, where the main issue is the conversion of fan-beam data into (irregularly spaced) samples of the parallel-beam angular variable. In Section IV, two simulation studies are presented that are designed to illustrate the features of our reconstruction algorithm. Both simulations used the standard Shepp-Logan phantom. The first simulated motion was a continuously translating object, and the second motion pattern was three small movements during a largely stationary scan. Sections V and VI are the discussion and conclusions.

II. THEORY

A. Notation

Let us consider an X-ray source moving on a circular path and fix the origin at the center of its rotation. Let $\vec{x} = (x, y)^T$ be a point in the source plane. Let $f(\vec{x})$ be the density function describing the scanned object, assumed to be zero everywhere outside a known support region. Parallel projections of such an object can be written as

$$\bar{p}(s, \theta) = \int_{-\infty}^{\infty} f(s\vec{\theta} + r\vec{\theta}^\perp) dr \quad (1)$$

with $\vec{\theta} = (-\sin \theta, \cos \theta)^T$ and $\vec{\theta}^\perp = (\cos \theta, \sin \theta)^T$.

Fan-beam projections can be written as

$$p(\alpha, \beta) = \int_0^{\infty} f(\vec{v}_\beta - t\vec{\alpha}_\beta) dt \quad (2)$$

with $\vec{v}_\beta = R\vec{\beta} = R(-\sin \beta, \cos \beta)^T$ and $\vec{\alpha}_\beta = (-\sin(\alpha + \beta), \cos(\alpha + \beta))^T$. All the angles α, β, θ are measured counter-clockwise, β and θ from the vertical axis, α from the line connecting \vec{v}_β to the origin, as illustrated in Fig. 1. The source path radius is denoted R .

It can be shown that a ray in fan-beam geometry can also be expressed in parallel geometry using the equivalent parameters

$$\theta = \alpha + \beta - \frac{\pi}{2} \quad (3)$$

$$s = R \sin \alpha, \quad (4)$$

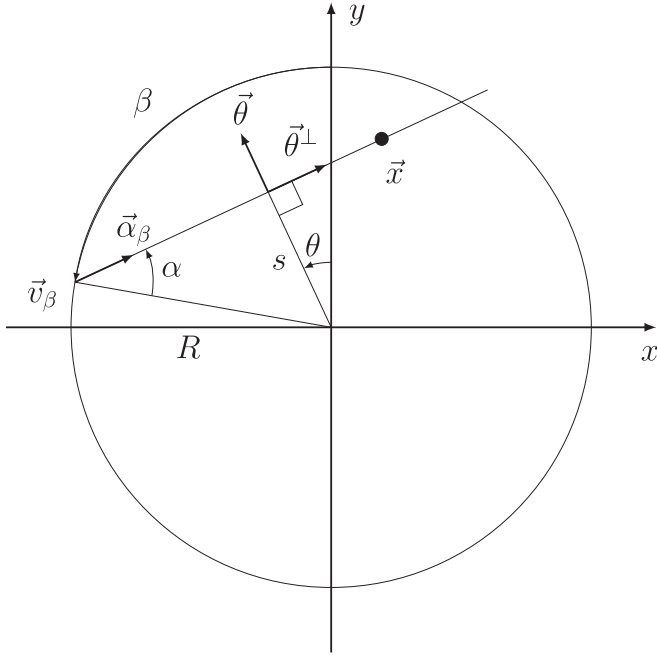


Fig. 1. The fan-beam projection geometry. The source is at \vec{v}_β and the trajectory radius is R . A measurement ray in the fan-beam geometry is parametrized by (α, β) . Also shown are the equivalent parameters (s, θ) expressing the same ray in the parallel projection geometry.

therefore,

$$p(\alpha, \beta) = \bar{p}\left(R \sin \alpha, \alpha + \beta - \frac{\pi}{2}\right). \quad (5)$$

B. Reconstruction Formula

Our method is based on parallel-beam reconstruction theory, using the DBP-H approach combined with the derivative calculated in image space. Here we present this known theory for the situation of no object motion and no truncated projections. In Section II-C below, we discuss how we adapt the method for fanbeam projections involving object translations and truncation.

Let us write the backprojected derivative of parallel projections, with the backprojection starting at an angle ϕ and covering an angle of π

$$b_\phi(\vec{x}) = \int_\phi^{\phi+\pi} \left. \frac{\partial}{\partial s} \bar{p}(s, \theta) \right|_{s=\vec{x} \cdot \vec{\theta}} d\theta. \quad (6)$$

It is known (e.g. [3], [8]) that this backprojection of the derivative is linked to the 1-D Hilbert transform of the image f by

$$b_\phi(\vec{x}) = -2\pi H_\phi f(\vec{x}) \quad (7)$$

where H_ϕ denotes the image operation of performing 1D Hilbert transforms along the vector $\vec{\phi}^\perp = (\cos \phi, \sin \phi)^\top$,

$$\begin{aligned} H_\phi f(\vec{x}) &= f(s\vec{\phi} + r\vec{\phi}^\perp) * h(r) \\ &= \int_{-\infty}^{\infty} \frac{f(s\vec{\phi} + (r-r')\vec{\phi}^\perp)}{\pi r'} dr' \end{aligned} \quad (8)$$

with $\vec{\phi} = (-\sin \phi, \cos \phi)^\top$. We will usually take $\phi = 0$, so

$$H_0 f(\vec{x}) = \int_{-\infty}^{\infty} \frac{f(r-r', s)}{\pi r'} dr'. \quad (9)$$

Now, closely following the approach of Zeng [8], we interchange the order of the differentiation and backprojection in (6), as follows. Considering $\bar{p}(\vec{x} \cdot \vec{\theta}, \theta)$ as the backprojected image of the θ projection, we have

$$\frac{\partial}{\partial x} \bar{p}(\vec{x} \cdot \vec{\theta}, \theta) = \left(\frac{\partial}{\partial s} \bar{p} \right) (\vec{x} \cdot \vec{\theta}, \theta) (-\sin \theta), \quad (10)$$

$$\frac{\partial}{\partial y} \bar{p}(\vec{x} \cdot \vec{\theta}, \theta) = \left(\frac{\partial}{\partial s} \bar{p} \right) (\vec{x} \cdot \vec{\theta}, \theta) (\cos \theta). \quad (11)$$

From (6), using the trigonometrical identity $\cos^2 \theta + \sin^2 \theta = 1$ and verifying that we are allowed to permute the derivative and the sum, we can then write

$$\begin{aligned} b_\phi(\vec{x}) &= \int_\phi^{\phi+\pi} \frac{\partial}{\partial s} \bar{p}(\vec{x} \cdot \vec{\theta}, \theta) d\theta \\ &= \int_\phi^{\phi+\pi} \left[\frac{\partial}{\partial x} \bar{p}(\vec{x} \cdot \vec{\theta}, \theta) (-\sin \theta) + \frac{\partial}{\partial y} \bar{p}(\vec{x} \cdot \vec{\theta}, \theta) (\cos \theta) \right] d\theta \\ &= \frac{\partial}{\partial x} b_{s,\phi}(\vec{x}) + \frac{\partial}{\partial y} b_{c,\phi}(\vec{x}) \end{aligned} \quad (12)$$

with

$$b_{s,\phi}(\vec{x}) = \int_\phi^{\phi+\pi} \bar{p}(\vec{x} \cdot \vec{\theta}, \theta) (-\sin \theta) d\theta \quad (13)$$

$$b_{c,\phi}(\vec{x}) = \int_\phi^{\phi+\pi} \bar{p}(\vec{x} \cdot \vec{\theta}, \theta) (\cos \theta) d\theta. \quad (14)$$

This formula allows us to make two weighted backprojections $b_{s,\phi}$ and $b_{c,\phi}$ of unprocessed sinogram data and take the derivative in image space afterwards in order to compute $b_\phi(\vec{x})$. Our approach is to convert the measurements to parallel projections from fan-beam. We can now convert directly into the parallel backprojected image, while simultaneously handling the motion of the object.

From (7), we have $H_\phi f(\vec{x}) = (-1/2\pi)b_\phi(\vec{x})$, so the 1D Hilbert transform of the object function $f(\vec{x})$ can be directly obtained from $b_\phi(\vec{x})$. In order to recover $f(\vec{x})$, as in every DBP-H method, we use the assumption that the function f vanishes outside a known support region. Thus, in the case of horizontal filtering ($\phi = 0$), we assume that $f(x, y) = 0$ for x outside the known interval $[L + \epsilon, U - \epsilon]$ while $H_0 f(x, y)$ is known for all $x \in [L, U]$. The limits L and U depend on y . We can then apply the following formula [24] to each horizontal line:

$$\begin{aligned} f(x, y) &= \frac{-1}{\sqrt{(x-L)(U-x)}} \\ &\times \left(\int_L^U \frac{\sqrt{(x'-L)(U-x')}}{\pi(x-x')} H_0 f(x', y) dx' + C(y) \right). \end{aligned} \quad (15)$$

The term C is constant for each line and is directly proportional to the ray-sum of the line the finite Hilbert inversion is being applied to: $C(y) = (-1/\pi)\bar{p}(y, 0)$.

The reconstruction formula has been discussed in parallel-beam geometry. The measurements and the motion, however, arise in fan-beam. We implement the parallel beam method of (13–14, 12, 7, 15) by converting from fan-beam projections while simultaneously handling arbitrary translations and truncation; see Section III.

C. Motion and Truncation

We assume a single scan of $[0, 2\pi]$ and we associate time with the angle β of the scan. When the object is undergoing a sequence of translations, the object function at a given moment, characterized by the gantry angle β , is the object function at reference time, displaced by a vector \vec{d}_β . Let us partition the total scan $[0, 2\pi]$ into N intervals, $0 = \beta_0 < \beta_1 < \beta_2 < \dots < \beta_N = 2\pi$. We describe the motion using a set of displacement vectors $\{\vec{\Delta}_k\}$, $k = 1 \dots N$. Each section corresponds to the interval on which the displacement $\vec{\Delta}_k$ occurs, with constant velocity $\vec{m}_k = \vec{\Delta}_k / (\beta_k - \beta_{k-1})$. The velocity $\vec{m}(\beta)$ at the current position β is

$$\vec{m}(\beta) = \begin{cases} \vec{m}_1, \beta \in (0, \beta_1) \\ \vec{m}_2, \beta \in (\beta_1, \beta_2) \\ \dots \\ \vec{m}_N, \beta \in (\beta_{N-1}, 2\pi). \end{cases} \quad (16)$$

The total displacement when the source is at position β can then be written as

$$\vec{d}_\beta = \int_0^\beta \vec{m}(\beta') d\beta'. \quad (17)$$

If now we fix the reference time at $\beta = 0$, we obtain the following relation linking the displaced image f_β to the reference image f_0 :

$$f_\beta(\vec{x} + \vec{d}_\beta) = f_0(\vec{x}). \quad (18)$$

When a projection is acquired at a gantry angle β of the moving object f_β , the ray-sum (2) becomes

$$\begin{aligned} p(\alpha, \beta) &= \int_0^\infty f_\beta(\vec{v}_\beta - t\vec{\alpha}_\beta) dt \\ &= \int_0^\infty f_0(\vec{v}_\beta - t\vec{\alpha}_\beta - \vec{d}_\beta) dt. \end{aligned} \quad (19)$$

This is equivalent, up to a change in α , to a projection of the static reference object acquired with a virtual (perturbed) source trajectory $\vec{w}_\beta = \vec{v}_\beta - \vec{d}_\beta$. This equivalence is useful when considering a reference frame fixed on the moving object. Note that β is *not* the polar angle for the virtual source \vec{w}_β . To reconstruct f_0 , the quantity \vec{x} on the right-hand-side of (6), (12)–(14) would be replaced by $\vec{x} + \vec{d}_\beta$.

In the stationary case, the circular field-of-view (FOV) is defined as the intersection of the triangular regions that are visible from each source position. The triangular regions are defined by a fixed symmetric angular aperture α_{\max} of the source, corresponding to the finite detector extent. For a moving object, we define the FOV in the reference frame of the object

and we say that a point is within the FOV if it is visible in all the parallel projections. Depending on the motion, the shape of the resulting FOV is usually quite different from the circular FOV arising in the stationary case, and can easily be much larger. Examples of the behavior of these FOVs will be further discussed in Section IV.

III. ALGORITHM

We first consider the situation where there is no motion, and we examine the case of a point \vec{x} inside the circular FOV. The reconstruction algorithm relies on the intermediate calculation of $b_\phi(\vec{x})$ via (13–14) which requires, for the calculation of the integral over θ , that for all lines passing through \vec{x} , the line integral (measurement) $\bar{p}(\vec{x} \cdot \vec{\theta}, \theta)$ must be available, where angle θ is the orientation of the line with respect to the horizontal. Since the fan-beam source undergoes a 2π rotation, each line passing through \vec{x} is measured twice, which suggests a certain redundancy when computing $b_\phi(\vec{x})$. On the other hand, any point outside the FOV will, by definition, have the property that there exist some lines passing through \vec{x} but not passing through the FOV. Examining one of these lines through \vec{x} but outside the FOV, we know that when this line emanated from the x-ray source, it must have been truncated, in one instance at least. Note that, from the definition of the FOV, there is no immediate reason why the second measurement of this line should also have been truncated when measured from another x-ray source on the line. However, for the case of no motion, and since we are assuming a *symmetric* truncation aperture α_{\max} , these lines will always be truncated for both instances. Consequently, for any point \vec{x} outside the FOV, it will not be possible to find a full 180 degrees of values θ with which to calculate $b_\phi(\vec{x})$.

For the case with object motion, the situation changes. The definition of the FOV being the region for which all points are always visible during the scan is too stringent for the purposes of image reconstruction. The redundancy of two measurements per line could rescue a measurement that was truncated for one position of the source, but, due to a motion of the object, was within the source aperture for the second movement. Thus, it may be the case that parts of the object that are outside the FOV might nonetheless admit the calculation of $b_\phi(\vec{x})$. An essential first step of our algorithm is to perform this analysis to determine for which parts of the object the calculation of $b_\phi(\vec{x})$ is possible.

Once it is known for which points \vec{x} all 180 degrees of lines pass through the point, the algorithm passes to stage two which is the calculation of $b_\phi(\vec{x})$ for all these points, followed by the finite Hilbert inversion step to complete the reconstruction.

A. Part 1. Truncation and Angular Sufficiency

The purpose of the first part of the algorithm is to identify for which points \vec{x} (in practice, which pixels) the calculation of $b_\phi(\vec{x})$ can be achieved. We refer to these points as Hilbert points because $H_\phi f(\vec{x}) = (-1/2\pi)b_\phi(\vec{x})$. Assuming a circular scan of radius R , identification of the Hilbert points \vec{x} depends on the fan-beam truncation angle α_{\max} , and on \vec{d}_β , the description of the motion. This part of the algorithm can be achieved

analytically since the motion description and trajectory \vec{v}_β are known for all $\beta \in [0, 2\pi]$. The issue of handling a finite number of measured fan-beam projections only plays a role in part 2 of the algorithm.

The identification of Hilbert points is broken into two tasks. The first is to calculate a *local* virtual trajectory, corresponding to each point \vec{x} and taking into account the truncation. The second part is to analyze the truncation pattern to determine if all lines passing through \vec{x} intersect the non-truncated region of the local trajectory. If so, then all $\theta \in [0, \pi)$ will be available for the integration of (13) and (14). If not, we consider that essential measurements are missing, and that \vec{x} is not a Hilbert point, i.e., that $b_\phi(\vec{x})$ cannot be calculated from the measurements for the specified motion and truncation. Note that identification of the Hilbert points is independent of the collection of measured fan-beam projections; in particular, it is independent of the number of projections. However, it requires a complete (“continuous”) description of the motion \vec{d}_β for $\beta \in [0, 2\pi]$.

To construct the local virtual trajectory, we consider each angular position β and we apply the condition $|\alpha'| \leq \alpha_{\max}$ where α' is the ray angle of the displaced point $\vec{x} + \vec{d}_\beta$ in the projection \vec{v}_β . Under the assumption that no part of the object moves outside the circular trajectory, the angle α' will always lie inside the range $(-\pi/2, \pi/2)$ and α' can be found using the expression

$$\sin \alpha' = \frac{\vec{\beta}^\perp \cdot (\vec{x} + \vec{d}_\beta)}{L} \quad (20)$$

where $\vec{\beta}^\perp = (\cos \beta, \sin \beta)^T$ and L is the length from the shifted point to the source (or from the static point to the shifted source), $L = \|\vec{x} + \vec{d}_\beta - \vec{v}_\beta\|$ (Fig. 2). To emphasize the dependence on \vec{x} and on the movement, we sometimes write $\alpha'_{\beta, \vec{x}}$ rather than α' . Note that this calculation is conceptually in the reference frame of the circular trajectory whereas we normally fix the reference frame to the moving object. We now construct a binary-valued truncation function $t(\vec{x}, \beta)$ for the point \vec{x} which specifies which parts of the virtual trajectory are missing (due to truncation):

$$t(\vec{x}, \beta) = \begin{cases} 0 & \text{if } |\alpha'| \leq \alpha_{\max}, \\ 1 & \text{otherwise.} \end{cases} \quad (21)$$

Recall that the virtual trajectory, as seen by all points in the object (if there were no truncation) is $\vec{w}_\beta = \vec{v}_\beta - \vec{d}_\beta$. The truncation function describes which intervals from \vec{w}_β are removed to form a local virtual trajectory, “seen” by the point \vec{x} . Example local virtual trajectories are shown in Section IV.

In practice, we store the function $t(\vec{x}, \beta)$ as the sequence of endpoints β_k where the function $t(\vec{x}, \beta)$ transitions between truncated and untruncated, and these endpoints are found by applying (20) and (21) for all $\beta \in [0, 2\pi]$ sampled very finely. In our implementation, the function $t(\vec{x}, \beta)$ was evaluated at 360,000 samples of β for each \vec{x} . (This is a much higher number than the actual number of projections used in the simulations).

Having obtained the truncation description $t(\vec{x}, \beta)$, we now check if the point \vec{x} is completely sampled by the local virtual trajectory. Note that a line from \vec{x} to the virtual trajectory

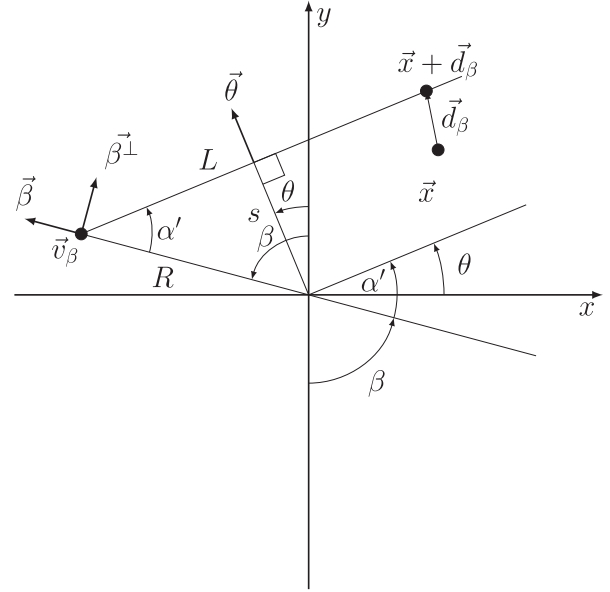


Fig. 2. The angle α' is seen to satisfy $\sin \alpha' = \vec{\beta}^\perp \cdot (\vec{x} + \vec{d}_\beta)/L$ and to satisfy $R \sin \alpha' = (\vec{x} + \vec{d}_\beta) \cdot \vec{\theta}$.

location \vec{w}_β will lie at angle θ to the horizontal if and only if $\vec{\theta} \cdot (\vec{w}_\beta - \vec{x}) = 0$. The condition for \vec{x} to be a Hilbert point is

$$\forall \theta \in [0, \pi), \exists \beta \in [0, 2\pi] \text{ such that} \quad (22)$$

$$\vec{\theta} \cdot (\vec{w}_\beta - \vec{x}) = 0 \text{ and } t(\vec{x}, \beta) = 0,$$

which means that all lines through \vec{x} must intersect the (truncated) local virtual trajectory. In practice, for each fixed \vec{x} , we converted non-truncated segments of \vec{w}_β into intervals in $\theta \in [0, \pi)$ (by extracting polar angles of $\vec{w}_\beta - \vec{x}$ at the endpoints of non-truncated segments) and we examined the union of these intervals. If the union covered $[0, \pi)$, then the point \vec{x} was deemed a Hilbert point.

In our implementation, we apply two shortcuts for finding Hilbert points. We observe that if there is a continuous segment in θ which is of length π or longer, then the condition for a Hilbert point is automatically satisfied. The second shortcut involves noting that if for a certain point \vec{x} there is no truncation at all for $\beta \in [0, 2\pi]$ then a θ -segment of length π or larger is assured. We consider three situations for a Hilbert point \vec{x} to arise: (i) there is no truncation in β for all $[0, 2\pi]$; (ii) there is a continuous segment in θ of length π or longer; or (iii) no such continuous segment exists, but all lines through \vec{x} intersect the non-truncated part of the virtual sinogram.

B. Part 2. Calculation of $b_\phi(\vec{x})$ and Hilbert Reconstruction

In part 1, the Hilbert points were identified. A Hilbert point \vec{x} satisfies the property that all lines through \vec{x} intersect the virtual trajectory \vec{w}_β , thereby ensuring full $[\phi, \phi + \pi)$ coverage of θ for calculating $b_\phi(\vec{x})$. The Hilbert points were determined from the motion and truncation parameters only; the available projections from the sampled fan-beam sinogram were not considered. On the other hand, for each Hilbert point \vec{x} , the calculation of $b_\phi(\vec{x})$ will use *all* projections for which the point \vec{x} is not truncated.

Part 2 of the algorithm consists of three components. The first is the identification of the angle ϕ for the Hilbert reconstruction step; second component is the calculation of $b_\phi(\vec{x})$ from the sinogram values $p(\alpha, \beta)$; and the third component is the Hilbert reconstruction step.

The reconstruction is performed on lines of contiguous Hilbert points that extend outside the object support on both sides. The object support is assumed known, and the reconstruction is carried out in the reference frame of this object at $\beta = 0$. It is often possible to use a set of parallel lines, all lying at the same angle ϕ with respect to the horizontal. Note that reconstruction of f_0 may not be possible for some Hilbert points, if the Hilbert point does not lie on a suitable line of Hilbert points. A Hilbert point which lies on a suitable line is called a *reconstruction point*. Usually, there is a reasonably obvious choice of ϕ which is suitable for all reconstruction points. A systematic brute force method would be to examine each Hilbert point by considering all line segments passing through the point and extending in both directions outside the object support. If one of these line segments consists only of Hilbert points, then all points along the line are reconstruction points, and for these points, ϕ is the angle of the line.

We now turn to the second component of the algorithm, with the assumption that a single global value of ϕ applies to all reconstruction points, although the same steps would be applied if ϕ varied with each point. We let N_β be the number of fan-beam projections and we index the samples of β as β_k , $k = 1, 2, \dots, N_\beta$. For each reconstruction point \vec{x} , all N_β fan-beam projections $p(\cdot, \beta_k)$ will contribute to $b_\phi(\vec{x})$, except for those for which $t(\vec{x}, \beta_k) = 1$. For this subset of the N_β projections, we calculate $\alpha'_{\beta_k, \vec{x}}$ using (20), and define $\theta_k = \alpha'_{\beta_k, \vec{x}} + \beta_k - \pi/2$ according to (3). Now we let $\bar{\theta}_k = \phi + ((\theta_k - \phi) \bmod \pi)$ so $\bar{\theta}_k \in [\phi, \phi + \pi)$.

This subset of directions $\{\bar{\theta}_k\}$ is sorted into a subsequence such that $\phi \leq \bar{\theta}_{k_1} \leq \bar{\theta}_{k_2} \leq \dots \leq \bar{\theta}_{k_{N_\beta(\vec{x})}} \leq \phi + \pi$ where $N_\beta(\vec{x}) \leq N_\beta$ is the number of fan-beam projections for which $t(\beta_k, \vec{x}) = 0$, i.e. the number of fan-beam projections for which \vec{x} is not truncated. These ordered but unevenly-spaced samples $\bar{\theta}_k$ will be used for the integrals in (13) and (14) to compute $b_s(\vec{x})$ and $b_c(\vec{x})$. We define $\Delta\theta_j = (\bar{\theta}_{k_{j+1}} - \bar{\theta}_{k_{j-1}})/2$, understanding that $\bar{\theta}_{k_0} = \phi$ and $\bar{\theta}_{k_{N_\beta(\vec{x})+1}} = \phi + \pi$. We let $p_j = p(\alpha'_{\beta_{k_j}, \vec{x}}, \beta_{k_j})$ for short, and from (3)–(5) we see that $p_j = \bar{p}(R \sin \alpha'_{\beta_{k_j}, \vec{x}}, \bar{\theta}_{k_j})$ and, observing that $R \sin \alpha'_{\beta_{k_j}, \vec{x}} = (\vec{x} + \vec{d}_{\beta_{k_j}}) \cdot \bar{\theta}_j$ (see Fig. 2), we calculate $b_{s,\phi}(\vec{x})$ and $b_{c,\phi}(\vec{x})$ from the fan-beam sinogram samples p_j :

$$b_{s,\phi}(\vec{x}) = \sum_{j=1}^{N_\beta(\vec{x})} -\sin \bar{\theta}_{k_j} p_j \Delta\theta_j \quad (23)$$

$$b_{c,\phi}(\vec{x}) = \sum_{j=1}^{N_\beta(\vec{x})} \cos \bar{\theta}_{k_j} p_j \Delta\theta_j \quad (24)$$

and from (12), $b_\phi(\vec{x}) = (\partial/\partial x)b_{s,\phi}(\vec{x}) + (\partial/\partial y)b_{c,\phi}(\vec{x})$.

The final component of Part 2 of the algorithm is the finite Hilbert inverse method described, for $\phi = 0$, by (15). This implementation is straight-forward: the values of $b_\phi(\vec{x})$ have been calculated as outlined above, for all reconstruction points.

TABLE I
PARAMETERS OF THE FOVS USED IN THIS STUDY: DETECTOR HALF-WIDTH u WITH CORRESPONDING FAN HALF-WIDTH α_{max} AND FOV RADIUS r . THE A PRIORI PHANTOM SUPPORT HAS THE MAJOR HALF-AXIS OF 116 MM AND THE MINOR HALF-AXIS OF 90 MM

	Large FOV	Small FOV
u (mm)	167.25	138.75
α_{max} (deg)	19.21	16.12
r (mm)	118.45	99.97

Implementation details for the inverse Hilbert transform can be found in [3], [5], [25]. If a non-zero value of ϕ is used, then implementation can be conveniently performed by calculating $b_\phi(\vec{x})$ for reconstruction points \vec{x} lying on a rotated grid, such that the horizontal lines of the rotated grid match the rows of the array. The reconstructed image will be rotated by ϕ but would not have been subject to any more interpolations compared to the $\phi = 0$ case.

IV. SIMULATIONS

A. Setup

For the simulations used to validate our algorithm, we used an off-centered slice of the 3D Shepp-Logan phantom [1]. We tested our algorithm against a wide range of motions; we selected two of them for presentation here, for two different truncation values α_{max} . In addition, we performed a simulation experiment where the projection data did not undergo precisely the motion that was given to the reconstruction algorithm, and we called this the “noisy” motion case.

The scan geometry was circular with a radius of $R = 360$ mm and a flat detector at 480 mm from the source with 0.5 mm pixel spacing. The detector half-width u was fixed with regards to the desired FOV radius: slightly larger than the major half-axis of the phantom’s elliptic support for the large FOV, and between the lengths of the minor and the major half-axes for the reduced FOV. As an *a priori* used in the finite Hilbert inversion step, the phantom’s support was assumed to be an ellipse of a vertical half-axis of 116 mm and a horizontal half-axis of 90 mm. The detector, fan and static FOV dimensions are shown in Table I. In all cases, 720 regularly sampled projections were used. Two motions were used in our experiments:

1) *Motion 1, Large FOV*: The first motion was a sequence of six translations of a substantial amplitude, with displacement vectors of magnitude roughly 10% to 30% of the radius of the source trajectory. The motion is illustrated in Fig. 3 and its parameters are listed in Table II. The large FOV was used with this motion.

2) *Motion 2, Small FOV*: A second motion consisted of three small linear translations occurring during three short time intervals (see Table III and Fig. 4). The small FOV was used with this motion.

In both motions, the object did not end the scan at its original (starting) position, therefore the virtual trajectories were open. These virtual trajectories are shown in Fig. 3 and 4.

This second motion was also used for the study of robustness of the algorithm. Two kinds of perturbations were simulated. First, Gaussian noise with a standard deviation of 0.1% was

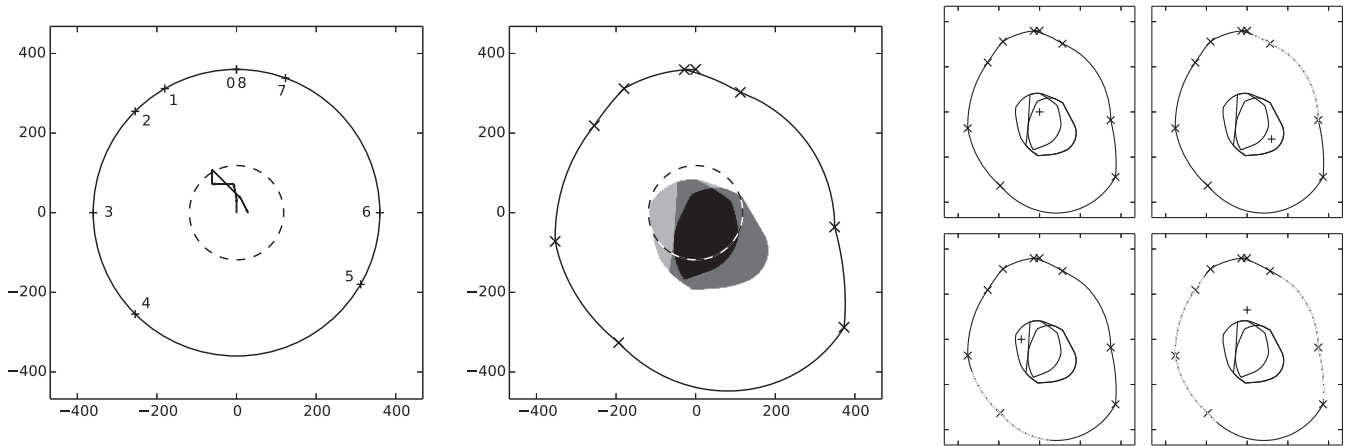


Fig. 3. Left: FOV and a schematic illustration of Motion 1 (displacement vectors $\vec{\Delta}_k$ and angles β_k). Middle: Virtual trajectory and different zones of the FOV obtained with the first motion. The static circular FOV is also superimposed for comparison with the dynamic FOV. The crosses at the virtual trajectory mark angles β_k , where translations begin and end. The different colors in the background image mean the following. Black: No truncation occurred for points within this region. Dark gray: Points covered for a continuous segment of length π or more. Only these two first categories of points could be reconstructed using a chord-based method. Light gray: Projection data redundancy allowed the reconstruction from multiple shorter segments of coverage. White: not a Hilbert point (truncation too large). Right: Examples of local virtual trajectories for particular points belonging to the four possible truncation scenarios. From top left: Black point, dark gray point, light gray point, white point.

TABLE II
PARAMETERS OF THE FIRST MOTION USED IN THE SIMULATIONS

Phase # (k)	β_k	β_{k+1}	Displacement vector ($\vec{\Delta}_k$)
0	0°	30°	$\vec{0}$
1	30°	45°	$R(0, 0.1)^T$
2	45°	90°	$R(-0.02, 0.1)^T$
3	90°	135°	$R(-0.15, 0)^T$
4	135°	240°	$R(0, 0.1)^T$
5	240°	270°	$R(0.2, -0.2)^T$
6	270°	340°	$\vec{0}$
7	340°	360°	$R(0.05, -0.1)^T$

TABLE III
PARAMETERS OF THE SECOND MOTION USED IN THE SIMULATIONS

Phase # (k)	β_k	β_{k+1}	Displacement vector ($\vec{\Delta}_k$)
0	0°	70°	$\vec{0}$
1	70°	90°	$R(-0.05, -0.02)^T$
2	90°	159°	$\vec{0}$
3	159°	185°	$R(0.08, 0.02)^T$
4	185°	240°	$\vec{0}$
5	240°	260°	$R(0.04, -0.05)^T$
6	260°	360°	$\vec{0}$

added to the projection data. Second, we simulated the case of badly estimated displacements. The reconstruction algorithm was still given the parameters listed in Table III, but the actual displacements occurring during the simulation of the sinogram were slightly different. To the start-stop angles β_1 through β_6 , a random Gaussian error with standard deviation of 1° was added. To the nonzero vectors among $\vec{\Delta}_k$, small vectors were added, with randomly chosen directions and with magnitudes selected from a gaussian distribution with standard deviation $0.001R$ which was roughly 1% to 2% of the individual displacements. The values of these “true” simulated motion parameters are listed Table III, whereas the “misestimated” values in Table IV were used for the reconstruction algorithm.

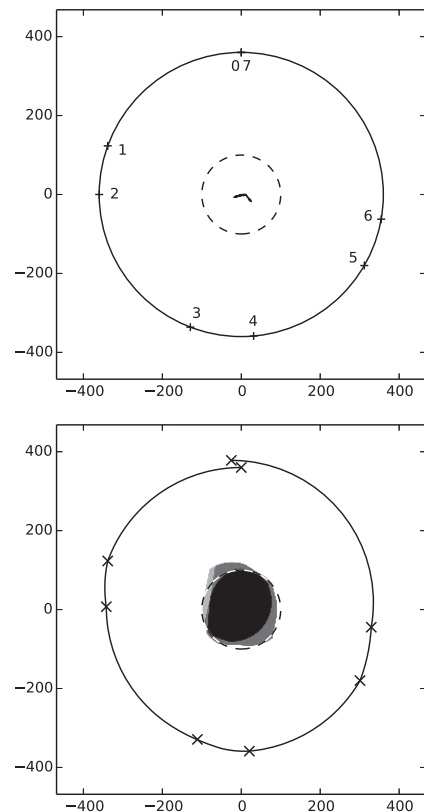


Fig. 4. Top: Representation of Motion 2. Bottom: Virtual trajectory, static FOV and different zones of the FOV obtained with the second motion. The color code used is the same as with the first motion on Fig. 3.

The Hilbert filtering direction was chosen along the horizontal axis, which corresponds to the angle $\phi = 0$. The reconstructible area is a horizontal band defined by the map of Hilbert points (see Section III-A) and the object support, and is represented in the next section by horizontal dotted lines.

TABLE IV
PARAMETERS OF THE MOTION USED IN THE “MISESTIMATED”
SIMULATION

Phase # (k)	β_k	β_{k+1}	Displacement vector ($\vec{\Delta}_k$)
0	0°	70.250°	$\vec{0}$
1	70.250°	89.175°	$\begin{pmatrix} -0.050002 \\ -0.020014 \end{pmatrix} R$
2	89.175°	158.348°	$\vec{0}$
3	158.348°	184.270°	$\begin{pmatrix} 0.080326 \\ 0.019864 \end{pmatrix} R$
4	184.270°	240.758°	$\vec{0}$
5	240.758°	258.227°	$\begin{pmatrix} 0.040095 \\ -0.050038 \end{pmatrix} R$
6	258.227°	360°	$\vec{0}$

B. Results

Fig. 5 and 6 show the reconstructions obtained, side by side with the respective pre-determined FOVs. For these cases where motion was either absent or exactly measured, the results match the expectations. All these horizontal lines of Hilbert points were accurately reconstructed, which began and ended in the visible area at both sides of the support. This margin was sometimes relatively small, as illustrated in the case of Motion 2 (Fig. 6), where the right side of the reconstructible zone barely exceeded the phantom support.

Fig. 7 shows two reconstructions obtained with the second motion under harsher conditions (noisy sinogram or inaccurately estimated motion). The additive noise in the sinogram merely causes a high-frequency “grain” noise in the reconstructed image.

On the other hand, when the motion was not accurately estimated, we observe several streak artifacts, as well as “stripe” artifacts towards the bottom of the image. The latter are caused by the constant $C(y)$ in the Hilbert inversion step in (15) because it is obtained from $C(y) = (-1/\pi)\bar{p}(R \sin \alpha', 0)$ and a poor estimation of the displacement directly affects the computation of α' and, therefore, of the constant. A possible explanation of the fact that these artifacts only appeared in that specific location is that the individual motions occurred over short periods of time, one of which corresponded to the gantry angles where the horizontal projections across these zones were acquired.

Table V contains the results of the root-mean-square error (RMSE) of all reconstructions obtained with the small FOV and, in the dynamic cases, the second motion used. The reconstructed values were compared to the exact defined values of the phantom, but only within the reconstructibility zone—at the intersection of the pre-computed FOV with the phantom support, but only across valid horizontal filtering lines. There seems to be little difference between the static reconstruction and its motion compensated counterpart—the dynamic reconstruction yields a RMSE only about 4% higher. The Gaussian noise within the sinogram has even less impact, increasing the RMSE by less than 2%. The misestimation of motion parameters, on the other hand, increases the overall RMSE significantly (by more than 60%). However, the artifacts caused by the misestimation are mostly local.

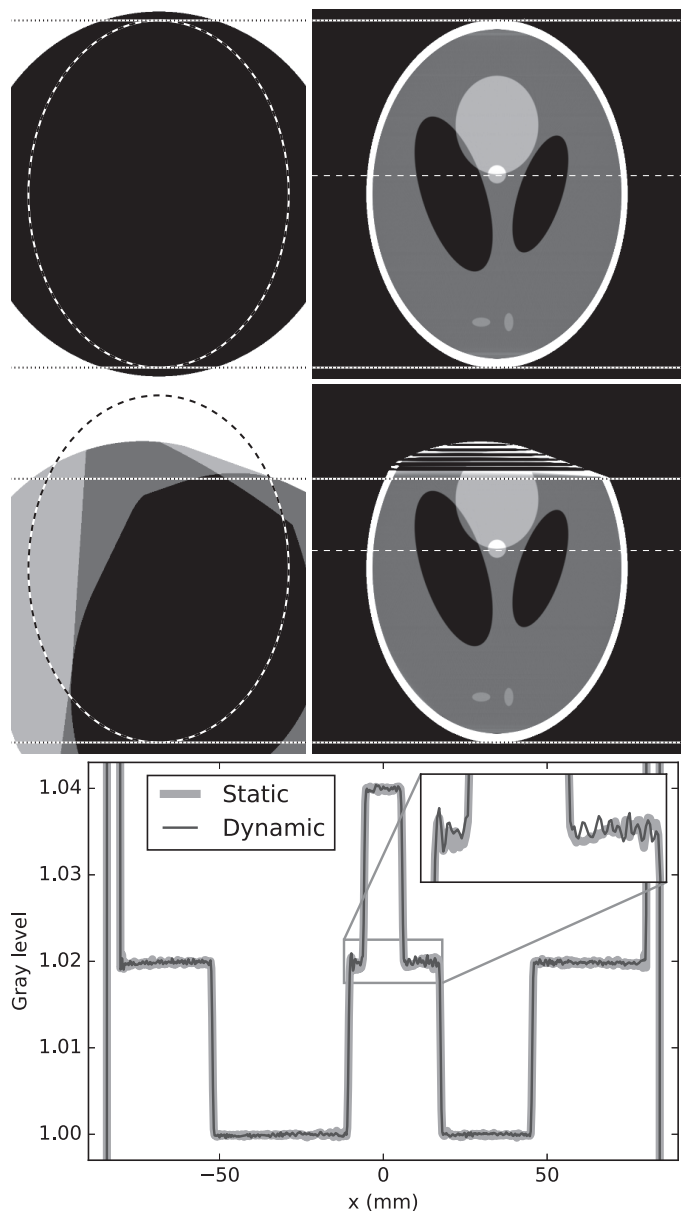


Fig. 5. The reconstructed images obtained without motion (top) and in presence of Motion 1 (middle). Gray window: [1.00,1.06]. In the left column, pre-detected FOV and phantom support for comparison. The horizontal dotted lines indicate the limits of the reconstructible area of our algorithm with horizontal Hilbert filtering. At the bottom: Horizontal profile across the corresponding dashed lines drawn on the reconstructed images ($y = 12$ mm).

The horizontal profiles (bottom row of Fig. 5, 6 and 7) suggest that the reconstruction quality is satisfactory—as long as the motion was correctly estimated. In a clinical case, there would probably be a trade-off between image quality and sensitivity to errors in the motion estimation.

V. DISCUSSION

We have shown that our proposed algorithm is able to simultaneously handle lateral truncation and compensation for arbitrary translations whereas previous methods have focused on one of the two issues only. Rigid object motion is equivalent to having a perturbed, “virtual” source trajectory with a

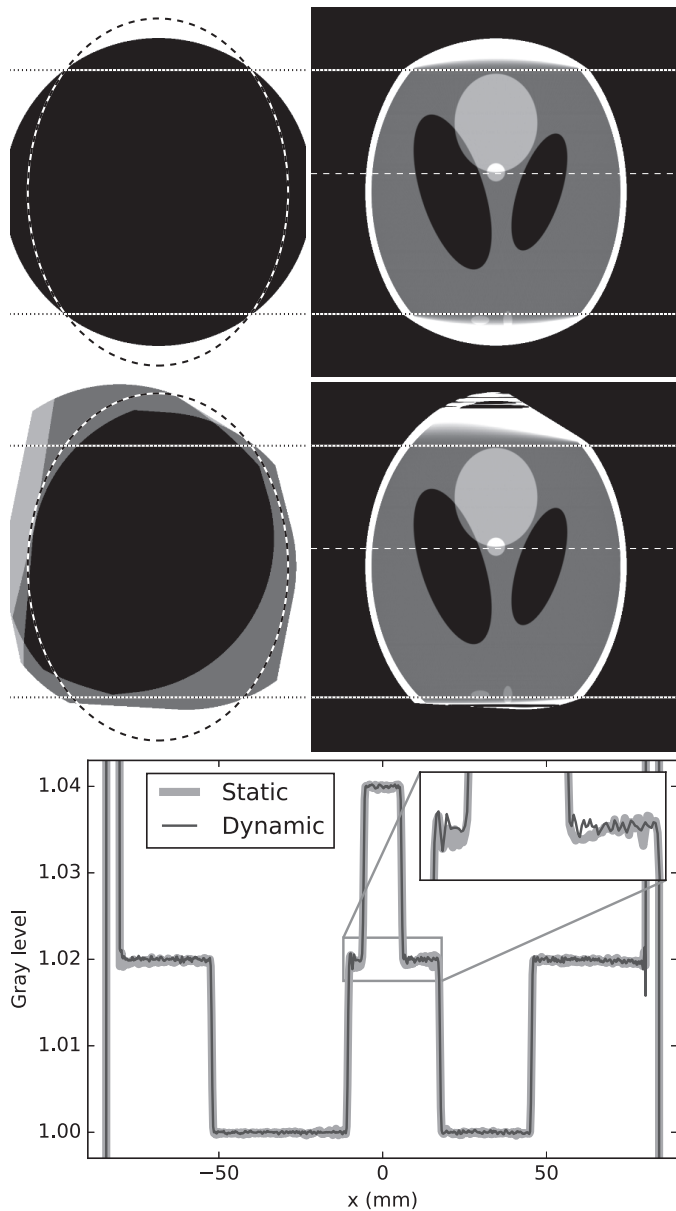


Fig. 6. The reconstructed images obtained without motion (top) and in presence of Motion 2 (middle). Gray window: [1.00,1.06]. In the left column, pre-detected FOV and phantom support for comparison. The horizontal dotted lines indicate the limits of the reconstructible area of our algorithm with horizontal Hilbert filtering. At the bottom: Horizontal profile across the corresponding dashed lines drawn on the reconstructed images ($y = 12$ mm).

stationary object. In the absence of static truncation, as in [22], the combination of a tight FOV with object motion may also substantially change the FOV, as shown in Fig. 3 for Motion 1. The motion will change which portion of space is never truncated (black areas); it will create areas which will be scanned by a continuous arc for the application of chord-based reconstruction techniques [5], [22], [23] (dark gray areas); and some areas will still be sampled at least once for any direction (light gray areas). Our direct reconstruction method handles these three situations to define where Hilbert points can be computed using (22).

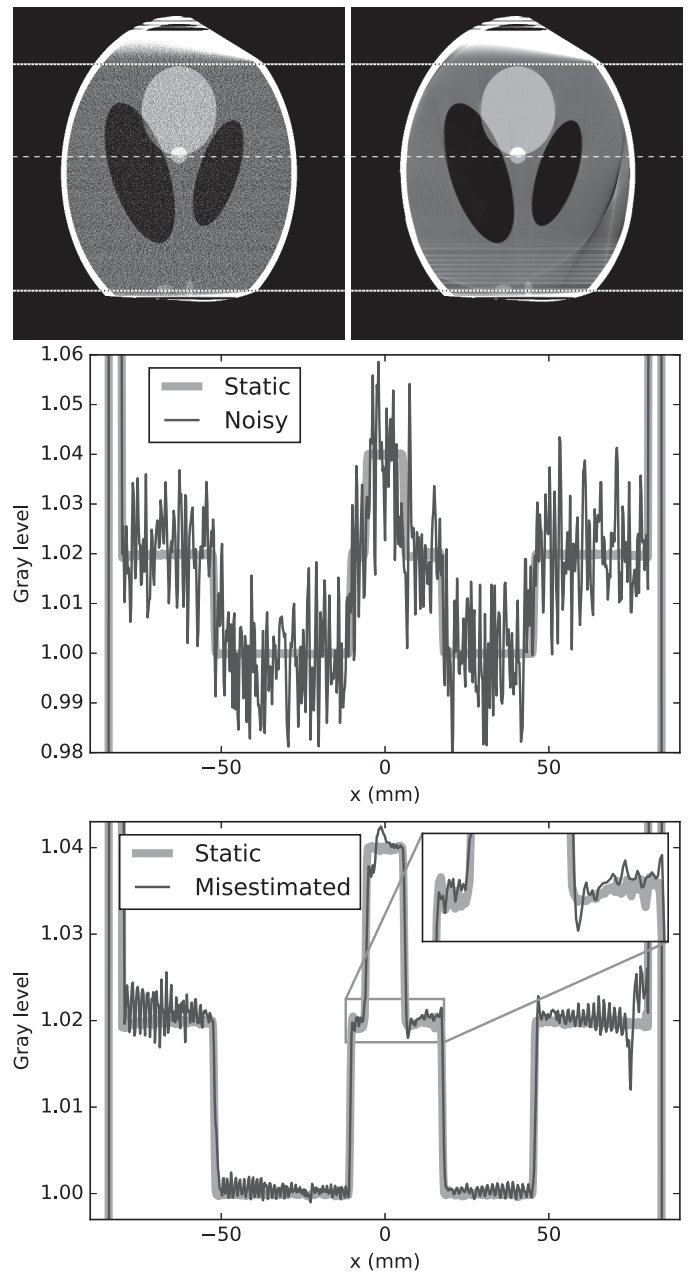


Fig. 7. Top: The reconstructed images obtained with the second motion with Gaussian noise added to the sinogram (left) and non-ideally estimated motion (right). Middle: Horizontal profiles across the reconstruction obtained with a noisy sinogram compared to the static, noiseless case ($y = 12$ mm). Bottom: Horizontal profiles across the reconstruction obtained with non-ideally estimated motion, also compared to the static, noiseless case ($y = 12$ mm).

As with other methods based on the finite Hilbert inversion, a Hilbert point is only reconstructible if it belongs to a segment along which Hilbert points are calculable, with both ends of the segment outside the object support. The reconstructible area would be larger if other directions than the horizontal were also considered. They would need to be chosen for each reconstruction manually from the region of Hilbert points (the three grey/black zones of Figs. 3 and 4), depending on the knowledge of the object support and the motion. It would then be possible to partition the image into multiple regions, where the best-suited direction would be used.

TABLE V
ROOT-MEAN-SQUARE ERROR WITHIN THE RECONSTRUCTIBLE ZONE
(COMPARED TO THE EXACT SIMULATED PHANTOM VALUES)

Simulation	RMSE
Static (Fig. 6, top, right)	0.0455
Dynamic (Fig. 6, middle, right)	0.0474
Static noisy (reconstruction image not shown)	0.0463
Dynamic noisy (Fig. 7, top, left)	0.0482
Misestimated (Fig. 7, top, right)	0.0775

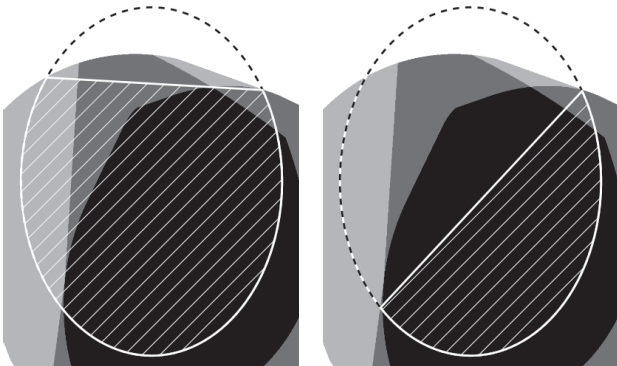


Fig. 8. Superposition of the largest reconstructible areas (hatched) with oblique inverse Hilbert filtering on the FOV image with Motion 1, with our algorithm (left) and chord-based techniques (right).

The enlargement of the area where Hilbert points can be computed with our algorithm can lead to a substantial increase of the part of the object which is accurately reconstructed, compared to chord-based techniques. This increase is nicely illustrated with the first motion in Fig. 8 where it is shown that the largest reconstructible area is substantially larger with our algorithm even if oblique inverse Hilbert filtering were applied. Iterative reconstruction techniques would further increase the reconstructible area since only one extremity of the segment must be outside the object support [7], [26] with the disadvantage of a substantial increase of computational time due to both the necessity for multiple iterations, and the fact that the entire object needs to be modelled (represented in pixels) even though only a portion of it will be accurately reconstructed.

The sampling of the measurements needed to calculate a given Hilbert point is irregular and specific to the object motion. It has been accounted for by discretizing (6) in (23) and (24). Although computationally intensive because the measurements p_j must be sorted for each pixel of the backprojection b_ϕ according to the θ_j angles of the measurement lines, this solution increases the area where Hilbert points can be calculated. Another advantage is the use of all projections corresponding to measured lines intersecting a given point, unlike chord-based techniques which only use the data acquired between the two end points of the chord. This full use of all measurements should substantially decrease the CT image noise compared to chord-based techniques when reconstructing from noise-corrupted projection data.

A further innovation is Part 1 of the algorithm. Reconstructions could be performed without this step of analytically establishing the truncation behavior at each point;

the rebinning to the parallel geometry in Part 2 could have been performed blindly, but would have left the difficulty of identifying the truncation based on the uneven sampling in θ_k . Heuristic thresholds would have had to be applied, to limit $\bar{\Delta}_\theta$ from growing too large, but without being able to separate the situations of gaps in θ due to truncation from gaps due to large steps in the virtual trajectory caused by the sampling of the fan-beam projections. Our method completely avoids this awkward complication.

The motion was assumed to be known in this study. A first experiment has demonstrated a substantial impact on the reconstructed image of random noise in the specification of the motion (Fig. 6). The robustness of the algorithm to a mismatch of the specified motion to the true motion should be further studied. On the other hand, the effect of statistical noise in the projections does not seem to be significant according to the preliminary test.

Throughout this work, we have assumed a full 360 degree fan-beam scan. However, our method applies equally well for other scanning arcs. In particular, for any type of reduced scans [2], [27], the processing would remain identical because the complementary component of the shortscan trajectory could be considered as a region where *all* image points are truncated. By the same reasoning, our approach also works for any collection of arcs on the circle.

For future developments, extending our algorithm to handle any general in-plane rigid motion should be straightforward, as for example the change in the reconstruction part of the algorithm in [19] to that in [20] was minimal. However, rigid motion is often 3D in practice, e.g., head movements of the patient during the acquisition [21]. Whether it would be feasible to expand the algorithm to a 3D situation is a more difficult question, as the “backproject first, then differentiate” approach has, to our knowledge, not yet been explored in three dimensions.

Fully 3D motion can be handled by iterative algorithms. In this case, identification of the 3D reconstructible area is still a difficult question. For example, an empirical solution based on the fraction of non-truncated projections at each point has been proposed [21]. While it may be effective in practice, some areas might be reconstructible despite truncation and non-reconstructible areas may occur with no truncation such as the extreme example [21] of object rotation with the source resulting in only one non-truncated projection acquired repeatedly. Recently, this group has proposed a rigorous method to identify non-reconstructible areas based on violating Tuy’s condition but cannot ensure reconstructibility in the complementary region [28]. On the other hand, in our 2D approach, we rigorously identified a reconstructible area but it might not be maximal, as mentioned above.

VI. CONCLUSION

We have proposed a direct algorithm for motion-compensated reconstruction from truncated fan-beam projections. The algorithm first analyses the reconstructible area using local “virtual” trajectories which account for the motion and the truncation. Our reconstruction technique handles data redundancy from discontinuous segments of these

local “virtual” trajectories which expands the reconstructible area compared to chord-based algorithms. Satisfactory results were obtained with Shepp-Logan phantom simulations with several motion cases and detector sizes.

REFERENCES

- [1] A. C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*, Piscataway, NJ, USA: IEEE Press, 1988.
- [2] F. Noo, M. Defrise, R. Clackdoyle, and H. Kudo, “Image reconstruction from fan-beam projections on less than a short scan,” *Phys. Med. Biol.*, vol. 47, pp. 2525–2546, 2002.
- [3] F. Noo, R. Clackdoyle, and J. D. Pack, “A two-step Hilbert transform method for 2D image reconstruction,” *Phys. Med. Biol.*, vol. 49, no. 17, pp. 3903–3923, 2004.
- [4] R. Clackdoyle, F. Noo, J. Guo, and J. Roberts, “Quantitative reconstruction from truncated projections in classical tomography,” *IEEE Trans. Nucl. Sci.*, vol. 51, no. 5, pp. 2570–2578, Oct. 2004.
- [5] J. D. Pack, F. Noo, and R. Clackdoyle, “Cone-beam reconstruction using the backprojection of locally filtered projections,” *IEEE Trans. Med. Imag.*, vol. 24, no. 1, pp. 70–85, 2005.
- [6] L. Yu, Y. Zou, E. Y. Sidky, C. A. Pelizzari, P. Munro, and X. Pan, “Region of interest reconstruction from truncated data in circular cone-beam CT,” *IEEE Trans. Med. Imag.*, vol. 25, no. 7, pp. 869–881, 2006.
- [7] M. Defrise, F. Noo, R. Clackdoyle, and H. Kudo, “Truncated Hilbert transform and image reconstruction from limited tomographic data,” *Inverse Probl.*, vol. 22, no. 3, p. 1037, 2006.
- [8] G. L. Zeng, “Image reconstruction via the finite Hilbert transform of the derivative of the backprojection,” *Med. Phys.*, vol. 34, no. 7, pp. 2837–2843, 2007.
- [9] Y. Wei, “Comment on image reconstruction via the finite hilbert transform of the derivative of the backprojection,” *Med. Phys.*, vol. 34, no. 10, pp. 4068–4070, 2007.
- [10] O. Helenon, D. Chanin, M. Laval-Jeantet, and J. Frija, “Artifacts on lung CT scans: Removal with fourier filtration,” *Radiology*, vol. 171, pp. 572–574, 1989.
- [11] D. Zerfowski, “Motion artifact compensation in CT,” *Proc. Medical Imaging Conf., Int. Soc. Optics and Photonics*, 1998, pp. 416–424.
- [12] C. Crawford, K. King, C. Ritchie, and J. Godwin, “Respiratory compensation in projection imaging using a magnification and displacement model,” *IEEE Trans. Med. Imag.*, vol. 15, no. 3, pp. 327–332, 1996.
- [13] R. R. Fulton, S. R. Meikle, S. Eberl, J. Pfeiffer, and C. J. Constable, “Correction for head movements in positron emission tomography using an optical motion-tracking system,” *IEEE Trans. Nucl. Sci.*, vol. 49, no. 1, pp. 116–123, Feb. 2002.
- [14] S. Roux, L. Desbat, A. Koenig, and P. Grangeat, “Exact reconstruction in 2D dynamic CT: Compensation of time-dependent affine deformations,” *Phys. Med. Biol.*, vol. 49, no. 11, p. 2169, 2004.
- [15] L. Desbat *et al.*, “Compensation of some time dependent deformations in tomography,” *IEEE Trans. Med. Imag.*, vol. 26, no. 2, pp. 261–9, 2007.
- [16] J. Nuyts, J.-H. Kim, and R. Fulton, “Iterative CT reconstruction with correction for known rigid motion,” in *Proc. 11th Int. Meeting Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*, 2011, pp. 132–135.
- [17] D. Schäfer, J. Borgert, V. Rasche, and M. Grass, “Motion-compensated and gated cone beam filtered back-projection for 3-D rotational X-ray angiography,” *IEEE Trans. Med. Imag.*, vol. 25, no. 7, pp. 898–906, 2006.
- [18] S. Rit, D. Sarrut, and L. Desbat, “Comparison of analytic and algebraic methods for motion-compensated cone-beam CT reconstruction of the thorax,” *IEEE Trans. Med. Imag.*, vol. 28, no. 10, pp. 1513–1525, 2009.
- [19] H. Yu, Y. Wei, J. Hsieh, and G. Wang, “Data consistency based translational motion artifact reduction in fan-beam CT,” *IEEE Trans. Med. Imag.*, vol. 25, no. 6, pp. 792–803, 2006.
- [20] H. Yu and G. Wang, “Data consistency based rigid motion artifact reduction in fan-beam CT,” *IEEE Trans. Med. Imag.*, vol. 26, no. 2, pp. 249–260, 2007.
- [21] J.-H. Kim, J. Nuyts, A. Kyme, Z. Kuncic, and R. Fulton, “A rigid motion correction method for helical computed tomography (CT),” *Phys. Med. Biol.*, vol. 60, no. 5, pp. 2047–2073, Mar. 2015.
- [22] D. Xia, E. Sidky, L. Yu, Y. Zou, and X. Pan, “Exact ROI image reconstruction with perturbed source trajectories in C-arm CT,” *Proc. IEEE Nuclear Science Symp. Conf. Rec.*, 2005, vol. 4, p. 4.
- [23] H. Yu, S. Zhao, Y. Ye, and G. Wang, “Exact BPF and FBP algorithms for nonstandard saddle curves,” *Med. Phys.*, vol. 32, no. 11, pp. 3305–3312, 2005.
- [24] S. G. Michlin and A. H. Armstrong, *Integral Equations and Their Applications to Certain Problems in Mechanics, Mathematical Physics and Technology*, 2nd edition, London, U.K.: Pergamon, 1957.
- [25] F. Dennerlein, F. Noo, H. Schondube, G. Lauritsch, and J. Hornegger, “A factorization approach for cone-beam reconstruction on a circular short-scan,” *IEEE Trans. Med. Imag.*, vol. 27, no. 7, pp. 887–896, 2008.
- [26] R. Clackdoyle and M. Defrise, “Tomographic reconstruction in the 21st century,” *IEEE Signal Process. Mag.*, vol. 27, no. 4, pp. 60–80, 2010.
- [27] D. Parker, “Optimal short scan convolution reconstruction for fanbeam CT,” *Med. Phys.*, vol. 9, no. 2, pp. 254–257, 1982.
- [28] T. Sun, R. Clackdoyle, R. Fulton, and J. Nuyts, “Quantification of local reconstruction accuracy for helical CT with motion correction,” *Proc. IEEE Nuclear Science Symp. and Medical Imaging Conf.*, 2014.