Image Reconstruction for Source Trajectories with Kinks

Yu Zou, Xiaochuan Pan, and Emil Sidky

Abstract—We describe a formula for the inversion of the x-ray transform with a general trajectory. Based upon the formula, we derive algorithms for image reconstruction from the x-ray transform. The numerical results show that exact ROI images can be reconstructed by use of the derived algorithms from (truncated) data acquired with trajectories with kinks.

I. INTRODUCTION

In certain applications of cone-beam CT, the x-ray source trajectory may not be smooth. Instead, it has a finite number of kinks (i.e., singularities). Examples of such trajectories include the circle-circle, circle-line, and circle-helix-circle trajectories. Some of these trajectories have been investigated previously, and both approximated and quasi-exact algorithms have been developed for image reconstruction in these situations. Recently, Katsevich has studied image reconstruction by use of the exact filtered backprojection (FBP) algorithm from data acquired with a circle-line trajectory [1]. Recently, there has been significant development on image reconstruction for general cone-beam trajectories [2, 3, 4].

In this work, we describe a formula for the inversion of the x-ray transform with a general trajectory. Based upon the formula, we first derive algorithms for image reconstruction from the x-ray transform. Subsequently, we show that the derived algorithms are directly applicable to reconstructing images for source trajectories, such as the aforementioned circle-circle and circle-line trajectories, with a finite number of singularities (i.e., the kinks). We have also performed numerical studies, and the quantitative results in these studies indicate that exact ROI images can be reconstructed by use of the derived algorithms from (truncated) data acquired with trajectories with kinks.

II. THEORY

A. Cone-beam data function

We assume that the subject to be imaged is confined in a cylinder of radius $R_s$. Therefore, the support of the object function $f(\tilde{r})$ satisfies

\[ f(\tilde{r}) = 0 \quad x^2 + y^2 > R_s^2, \tag{1} \]

where $\tilde{r} = (x, y, z)^T$ and the central axis of the support cylinder coincides with the $z$-axis.

Suppose the x-ray source trajectory can be characterized by a vector $\tilde{r}_0(\lambda)$ that is a continuous function of a parameter $\lambda$, which is the path length over the trajectory. In the fixed-coordinate system, we write $\tilde{r}_0(\lambda) = (x_0(\lambda), y_0(\lambda), z_0(\lambda))^T$, and the distance between a point on the trajectory and the $z$-axis is thus given by

\[ R(\lambda) = \sqrt{x_0^2(\lambda) + y_0^2(\lambda)}. \tag{2} \]

We assume that $R(\lambda) > R_s$. Therefore, the trajectory does not go through the support cylinder.

The trajectory considered could be a smooth or piecewise smooth function of $\lambda$. A chord-line is defined as a straight line intersecting with the trajectory at two points $\tilde{r}_0(\lambda_a)$ and $\tilde{r}_0(\lambda_b)$. Without the loss of generality, it is assumed that $\lambda_a \leq \lambda_b$. One can use

\[ \hat{e}_c = \frac{\tilde{r}_0(\lambda_b) - \tilde{r}_0(\lambda_a)}{|\tilde{r}_0(\lambda_b) - \tilde{r}_0(\lambda_a)|} \tag{3} \]

to denote the direction of the chord-line.

The cone-beam projection of the image function is defined as

\[ D(\tilde{r}_0(\lambda), \hat{\beta}) = \int_0^\infty dt \, f(\tilde{r}_0(\lambda) + t \hat{\beta}), \tag{4} \]

where the unit vector $\hat{\beta}$, indicating the projection direction of an individual x-ray passing through the point $\tilde{r}'$, can be written as

\[ \hat{\beta} = \frac{\tilde{r}' - \tilde{r}_0(\lambda)}{|\tilde{r}' - \tilde{r}_0(\lambda)|}, \tag{5} \]

and $\tilde{r}' \in \mathbb{R}^3$.

B. Inversion formula

Any point $\tilde{r}$ on the chord-line can be expressed as

\[ \tilde{r} = \frac{1}{2} [\tilde{r}_0(\lambda_a) + \tilde{r}_0(\lambda_b)] + x_c \hat{e}_c, \quad x_c \in \mathbb{R}. \tag{6} \]

Furthermore, we refer to the segment on the chord-line between points $\tilde{r}_0(\lambda_a)$ and $\tilde{r}_0(\lambda_b)$ as the chord. For a helical
where \( r' \in \mathbb{R}^3 \), and \( \nu_c \) denotes the spatial frequency with respect to \( x_c \).

Applying Eq. (13) to Eq. (8) yields

\[
f_c(x_c, \lambda_a, \lambda_b) = \frac{1}{2\pi^2} \int_{\mathbb{R}} \frac{dx'_c}{x_c - x_c} g_c(x'_c, \lambda_a, \lambda_b), \quad (14)
\]

where \( x_c \in \mathbb{R} \). The result in Eq. (14) provides an algorithm for reconstructing the image on a chord from knowledge of the backprojection over the entire chord-line. As shown below, however, by exploiting the fact that the image support is confined on the chord, one can reconstruct the image on the chord from knowledge of the backprojection only on the chord.

We use \( x_{s1} \) and \( x_{s2} \) to denote the endpoints of the intersection of the chord with the support cylinder, which we refer to as the support segment on the chord. Because the object function is confined within the support cylinder and because the trajectory never passes through the support cylinder, we observe that \( [x_{s1}, x_{s2}] \in [-l, l] \), i.e., the support segment of the object function on the chord-line is always within the chord. This observation can be exploited for deriving algorithms capable of reconstructing the image on a chord by requiring knowledge of the backprojection only on the support segment.

Performing the Hilbert transform with respect to \( x_c \) on both sides of Eq. (14), we obtain

\[
g_c(x_c, \lambda_a, \lambda_b) = 2 \int_{\mathbb{R}} \frac{dx'_c}{x'_c - x_c} f_c(x'_c, \lambda_a, \lambda_b) = 2 \int_{x_{s1}}^{x_{s2}} \frac{dx'_c}{x'_c - x_c} f_c(x'_c, \lambda_a, \lambda_b), \quad (15)
\]

where \( x_c \in \mathbb{R} \), and parameters \( x_{s1} \) and \( x_{s2} \) satisfy \( x_{s1} \in (-\infty, x_{s1}] \) and \( x_{s2} \in [x_{s2}, \infty) \), respectively. Clearly, the last part of Eq. (15) was obtained by exploiting the fact that \( f_c(x_c, \lambda_a, \lambda_b) = 0 \) for \( x_c \notin [x_{s1}, x_{s2}] \). It is important to point out [3] that, for \( x_c \in [-l, l] \), only the first term, i.e., the physical data \( D(\hat{r}_0(\lambda), \hat{\beta}) \) of the extended data function in Eq. (11) contributes to the backprojection image \( g_c(x_c, \lambda_a, \lambda_b) \). Therefore, the derivation below considers the contribution only from the physical data term \( D(\hat{r}_0(\lambda), \hat{\beta}) \).

The result in Eq. (15) represents a Hilbert transform on a finite interval. Its inversion can be expressed as [5, 6]

\[
f_c(x_c, \lambda_a, \lambda_b) = \frac{1}{2\pi} \sqrt{(x_{s2} - x_c)(x_c - x_{s1})} \int_{\mathbb{R}} \frac{dx'_c}{x_c - x'_c} g_c(x'_c, \lambda_a, \lambda_b) + 2\pi D(\hat{r}_0(\lambda_a)), \quad (16)
\]

where

\[
g_c(x'_c, \lambda_a, \lambda_b) = \Pi_c(x'_c) \sqrt{(x_{s2} - x'_c)(x'_c - x_{s1})} g_c(x'_c, \lambda_a, \lambda_b), \quad (17)
\]

and \( \Pi_c(x'_c) = 1 \) if \( x'_c \in [x_{s1}, x_{s2}] \) and 0 if \( x'_c \notin [x_{s1}, x_{s2}] \). It is interesting to note that the first term of Eq. (16) indicates...
explicitly a shift-invariant filtering (i.e., the Hilbert transform) over the entire $x'_{s}$-axis. Such a shift-invariant form may have practical significance because it can generally be calculated efficiently by use of the fast-Fourier-transform technique.

It can be observed in Eq. (17) that the image on the chord can be obtained exactly from knowledge of the backprojection onto a support segment, specified by $x_c \in [x_{s1}, x_{s2}]$, on the chord. It is this important fact forming the basis for exact image reconstruction on a chord possibly from projections containing transverse truncation. We refer to this algorithm as the backprojection-implicit filtering (BPF) algorithm because it backprojects the modified data (i.e., the integration over $\lambda$ in obtaining $g\rho(x', \lambda_0)$ before performing the 1D Hilbert transform of the weighted backprojection image (i.e., the integration over $x'_{s}$). We can also derive a filtered backprojection (FBP) algorithm for image reconstruction on the chord from minimum data (MD) and thus refer to it as the MD-FBP algorithm. Similar to the BPF algorithm, the MD-FBP algorithm can also accommodate image reconstruction on a chord from truncated data.

III Numerical Studies

We consider the scanning of the Shepp-Logan phantom by use of two trajectories that have kinks. As shown in Figs. 1a and 1b, the first trajectory consists of two circles, whereas the second trajectory includes a line and a circle. We assume that the central point of the largest ellipsoid of the Shepp-Logan phantom is at the origin of the fixed-coordinate system.

The circle-circle trajectory in Fig. 1a can be mathematically written as

$$\tilde{r}_0(\lambda) = (R_0 \cos \lambda, R_0 \sin \lambda, 0) \quad \lambda \in [-\frac{3\pi}{2}, 0) \quad (18)$$

$$\tilde{r}_0(\lambda) = (R_0 \cos \lambda, 0, R_0 \sin \lambda) \quad \lambda \in [0, \frac{3\pi}{2})$$

where the first circle is in the $x$-$y$ plane, and the second circle is in the $x$-$z$ plane. A kink appears at $\lambda = 0$.

The circle-line trajectory in Fig. 1b is described by

$$\tilde{r}_0(\lambda) = (R_0 \cos \lambda, R_0 \sin \lambda, 0) \quad \lambda \in [-\frac{3\pi}{2}, 0) \quad (19)$$

$$\tilde{r}_0(\lambda) = (R_0, 0, h\lambda) \quad \lambda \in [0, \frac{\pi}{2})$$

where $R_0 = 285$ mm and $h = 128$ mm/rev. Again, a kink occurs at $\lambda = 0$.

We generated cone-beam data from the 3D Shepp-Logan phantom by use of each of these two trajectories in Fig. 1 with $R_0 = 285$ mm. We assume a constant distance $S = 502.5$ mm between the source and detector plane. A 2D detector plane of $512 \times 256$ square detector-element is assumed with element size of 0.78 mm. The long side of the rectangle detector plane is along the tangential direction of the circles, whereas the short side is along the $z$-axis for $\lambda \in [-\frac{\pi}{2}, 0)$ and the $y$-axis for $\lambda \in [0, \frac{\pi}{2})$. The simulated cone-beam data were sampled at 1024 views along the source trajectory that were uniformly distributed over the scanned range of the parameter $\lambda$.

![Fig. 2. Images of the Shepp-Logan phantom reconstructed by use of the proposed BPF algorithm from the data acquired along a circle-circle trajectory in 1a. Images in the left, middle, and right panels are on 2D slices specified by $x = 0$ cm, $y = -2.7$ cm, and $z = 0$ cm, respectively. The display window is $[1.0, 1.05]$.](image1)

![Fig. 3. Images of the Shepp-Logan phantom reconstructed by use of the proposed BPF algorithm from the data acquired along a circle-circle trajectory in 1b. Images in the left, middle, and right panels are on 2D slices specified by $x = 0$ cm, $y = -2.7$ cm, and $z = 0$ cm, respectively. The display window is $[1.0, 1.05]$.](image2)

We have applied the BPF and MD-FBP algorithms to reconstructing images from the simulated data for the circle-circle and circle-line trajectories. In this summary, we present the results obtained only with the BPF algorithm and will report the results obtained with the MD-FBP algorithm at the meeting. In Figs. 2 and 3, we show the images reconstructed from cone-beam data acquired with the circle-circle and circle-line trajectories. We have also performed quantitative comparison between the reconstructed and true values within the Shepp-Logan phantom. These results clearly demonstrated that the proposed BPF and MD-FBP algorithms can accurately recon-
structure images on chords from (truncated) data acquired with trajectories with a finite number of kinks.

IV Conclusion

We described a formula for image reconstruction on chords of a general trajectory with a finite number of singularities (i.e., kinks). Based upon the formula, the BPF algorithm can be derived for reconstructing ROI images from truncated data. We have applied the BPF algorithm to reconstruct ROI images from cone-beam projection data by use of the circle-circle and circle-line trajectories. Both these two trajectories have singularities. Results of our numerical studies confirm that the BPF algorithm accurately reconstruct ROI images from data acquired with a general, non-smooth trajectory.

Based upon the formula, we have also derived additional reconstruction algorithms that first perform data filtration before backprojecting the filtered data into image space. We refer to these algorithms as the filtered backprojection (FBP) algorithm and the minimum-data filtered backprojection (MD-FBP) algorithm, respectively. It is interesting to point out that the MD-FBP algorithm can exactly reconstruct ROI images from truncated data, whereas the FBP algorithm cannot. We have also performed numerical studies to evaluate and compare image reconstructions by use of the BPF, FBP, and MD-FBP algorithms and will present these theoretical and numerical results at the meeting.

V Acknowledgments

This work was supported in part by National Institutes of Health grants EB000225, EB002765, and EB003913. Its contents are solely the responsibility of the authors and do not necessarily represent the official views of the National Institutes of Health. The authors thank Mr. Dan Xia for discussion and also gratefully acknowledge the use of the Chiba City Linux cluster in the Mathematics and Computer Science Division of Argonne National Laboratory.

References

Monte Carlo Simulation of Scatter in a Cone-beam Micro-CT Scanner

Noel Black and Jens Gregor

Abstract—Scattered radiation is a potential source for artifacts in x-ray computed tomography (CT). In case scatter is a non-negligible contributing factor to degraded image quality, the projection data should be appropriately corrected. To determine whether such correction is necessary, we have developed our own Monte Carlo code and present a true cone-beam scatter study for a particular commercial micro-CT scanner. The experimental results presented include beam stop based validation as well as a simulation of detected primary and scatter radiation for a laboratory mouse. We find scatter contamination of the projection data to be negligible and conclude that scatter correction need not be performed.

I. INTRODUCTION

Scattered radiation is a potential source for artifacts in x-ray computed tomography (CT). When a substantial amount of scatter contaminates the projection data, the reconstructed image may for example suffer from low-contrast detectability as well as cupping and streaking. The magnitude of scatter contamination depends strongly on the type of scanner and the object being imaged. In case scatter is a non-negligible contributing factor to degraded image quality, the projection data should be appropriately corrected.

Beekman and coworkers [9], [4] recently used Monte Carlo simulation to estimate the effects of scatter for one particular commercial cone-beam micro-CT scanner, the SkyScan 1076 (SkyScan, Aartselaar, Belgium). They found that scatter contamination can affect the accuracy of the reconstructed attenuation values by as much as 15%. As a means for correction, they proposed an iterative method in which the image is first reconstructed with no scatter correction. Monte Carlo simulation is subsequently used to compute scatter projections. These are then subtracted from the original projection data and a new image is reconstructed. The process may be repeated to higher order as necessary.

Chow et al. [3] studied scatter contamination for another cone-beam micro-CT scanner, namely, the MicroCAT II (CTI Concorde Microsystems, Knoxville, TN). They found scatter to cause an error in the reconstructed attenuation coefficients of about 10% and reported a scatter-to-primary ratio, henceforth S/P, of about 30% for a mouse-sized phantom object.

In this paper we present a true cone-beam scatter study for the MicroCAT II. For this purpose, we have developed our own Monte Carlo code as described next. The experimental results presented include beam stop based validation as well as a simulation of detected primary and scatter radiation for a laboratory mouse. We find scatter contamination of the projection data to be negligible and conclude that scatter correction need not be performed.

II. IMPLEMENTATION OF PHOTON TRANSPORT PHYSICS

The Monte Carlo code tracks each photon from production at the source until it is absorbed in the object or leaves the voxel space of the object and is either detected or misses the detector. We confine our attention to the physics of the x-ray beam and of photon transport in the object, assuming that all photons incident on the detector are actually detected.

Source energy spectra are generated from the TASMIP code of Boone and Seibert [2], which polynomially interpolates measured constant potential tungsten anode x-ray spectra at 1 keV intervals.

The Monte Carlo code includes the dominant interactions of radiation in matter at energies below 1 MeV: photoelectric absorption, coherent scattering, and incoherent scattering. At each voxel, the code determines by sampling which (if any) interaction occurs, using the exponential decay law and interaction cross sections to form probability distributions. The probability \( P_{pe} \) for the photon to be absorbed by the photoelectric effect is computed as

\[
P_{pe} = \left(1 - e^{-\mu_{tot}d}\right) \frac{\mu_{pe}}{\mu_{tot}}
\]

where \( d \) is the path length of the photon trajectory through the voxel, \( \rho \) is the material density, and \( \mu_{tot} = \mu_{pe} + \mu_{coh} + \mu_{incoh} \). The probabilities for coherent scattering, incoherent scattering, and noninteraction are computed analogously. The attenuation coefficients \( \mu_{pe}, \mu_{coh}, \) and \( \mu_{incoh} \) are taken from the XCOM program developed at NIST [7].

The history of a photoelectrically absorbed photon is terminated. A photon that does not interact in the voxel continues to the next voxel intersected by its trajectory.

The scattering angle \( \theta \) of a coherently scattered photon is determined by sampling the coherent scattering differential cross section

\[
\left(\frac{d\sigma}{d\Omega}\right)_{coh} = \frac{r_e^2}{2} (1 + \cos^2(\theta)) F_m^2(x),
\]

where

\[
x = \frac{E}{hc} \sin(\theta/2).
\]

Here \( r_e \) is the classical electron radius, \( F_m(x) \) is the molecular coherent scattering form factor, \( h \) is the Planck constant, and \( c \) is the speed of light.


c is the speed of light, and \( E \) is the photon energy. The azimuthal angle \( \phi \) is sampled from a flat distribution. Form factors for water and Lucite are taken from Peplow et al. [8]. They have shown that measured molecular form factors must be used for agreement between calculation and experiment, and that most animal tissues appear similar to water in their coherent scattering distributions. For the mouse simulation, therefore, coherent scatter from any of the mouse tissues is modeled using the form factor for water. Fig. (1) compares the coherent scattering distributions. For the mouse simulation, \( \chi^6 \) monoenergetic photons at 20 keV in Lucite are assumed to be absorbed, and their histories are terminated. Incoherent scattering functions are expected distribution, again with \( \chi^6 \) monoenergetic \( \phi \) azimuthal angle and the uniform distribution resulting from simulation with the theoretically expected distribution.

The direction of an incoherently scattered photon is determined by sampling the incoherent scattering differential cross section

\[
\left( \frac{d\sigma}{d\Omega} \right)_{incoh} = \frac{r^2}{2} \left( \frac{E'}{E} \right)^2 \left( \frac{E'}{E} + \frac{E}{E'} + \cos^2(\theta) - 1 \right) S_n(x)
\]

and the uniform \( \phi \) distribution. The scattered photon transfers energy and momentum to an atomic electron in the collision and emerges with energy \( E' \) given by

\[
E' = E/\left[1 + (E/m_e c^2)(1 - \cos(\theta))\right],
\]

which merely states energy conservation for the interaction. \( m_e \) is the electron mass. Photons with energy less than 5 keV are assumed to be absorbed, and their histories are terminated. Incoherent scattering functions \( S_n(x) \) are formed by the independent atomic model, using the data of Hubble et al. [6]. Relative atomic abundances for muscle, soft tissue, and bone which are used in the mouse simulation are taken from XCOM [7]. Fig. (2) compares the incoherent scattering angular distribution resulting from simulation with the theoretically expected distribution, again with \( \chi^6 \) monoenergetic photons at 20 keV incident on Lucite.

The code uses the combined multiple recursive random number generator of the SPRNG [5] library to prevent periodicity from becoming an issue.

### III. Validation

Experimental projections were made of phantoms composed of Lucite slabs and lead stops in the CT micro-scanner. The lead stops are 1.6mm thick and are roughly square, with edge sizes of approximately 6mm and higher, in 2mm increments. The scanner has a tungsten anode x-ray source and was operated at an anode voltage of 80 kVp and an anode current of 500 \( \mu \)A. The exposure time was 375 \( \mu \)s. The source to detector distance \( L_{SD} \) is 359.68 mm and the source to center of rotation distance \( L_{SC} \) is 274.52 mm. The effective dimension of a detector cell \( w_{det} \) is 161.76 \( \mu \)m, with a binned resolution of 512 \( \times \) 768 cells.

Fig. (3) shows the experimental and simulation results for an 80 kVp x-ray beam scattering on a 12.0 mm thick Lucite slab with the 10mm lead stop on a line through the projection of the stop. The experimental projection of the stop has noticeably rounded edges compared with the simulated projection. This is probably due to two main effects. First, the focal spot size of the real x-ray beam is finite, whereas the simulation regards it as a point. A finite focal spot size will blur edges. Second, the lead pieces are not precisely of uniform thickness; the edges become crimped and distorted when the stops are cut out, since lead is a very soft material and precision machining was not possible. The agreement is therefore expected to be better for larger size stops, for which the rounded experimental edges on opposite sides of the stop are nonoverlapping.

If \( T_S \) is the intensity of detected radiation behind the lead stop and \( T_{SP} \) is intensity of detected radiation in the same region but with no lead stop, the S/P ratio should be given by \( S/P = T_{SP}/(T_{SP} - T_S) \) assuming the stop blocks all radiation incident upon it. If a fraction \( x \) of the incident primary radiation is unblocked by the stop, then \( S/P = (T_{SP} - xT_S)/T_{SP} \). (\( x \) can be calculated from the attenuation coefficients for lead.) The scatter to primary ratios are calculated in this way for an 80 kVp beam incident on a 12.0 mm thick slab of Lucite with lead stops of various sizes. A region of interest interior to the lead stop shadow region is selected for tabulation. This region is smaller than the shadow region to avoid edge effects. The agreement is good for the 10mm and larger stops: \( S/P_{exp} = 97\%\), and \( S/P_{sim} = 97\%\). The agreement worsens as the size of the stop decreases, as expected. The relative error is 0.96\%, 16.39\%, and 33.97\%. for the 10mm, 8mm, and 6mm stops respectively.

Fig. (4) plots the simulation results for \( S/P \) as a function of Lucite thickness. One set of points tabulates photons over the entire detector and the other set tabulates photons over a small region of interest at the center of the detector. The latter is larger in general for a given Lucite thickness. \( S/P \) is just below 14\% for the 24.0mm thick Lucite block and decreases approximately linearly to zero with decreasing Lucite thickness.

### IV. Simulation Based on a Mouse Reconstruction

We are interested in the amount of detected scatter in a projection of a typical mouse object. Fig. (5a) shows a slice
of a $256 \times 256 \times 511$ mouse reconstruction, obtained using the simultaneous iterative reconstruction technique (SIRT) [1]. To convert this to an input attenuation map for the simulation each voxel must be labeled as a material type with known attenuation coefficients and scattering properties. A typical mouse is composed primarily of soft tissue, muscle, water, and bone, and is assumed to be surrounded by air. We therefore want to threshold the attenuation map to create an indexed map. Fig. (5b) shows the plot of a line through the attenuation map with material thresholds superimposed. This line is the $x$-voxel = 100 line of the slice shown in Fig. (5a); it is approximately the mouse axis and is expected to intersect each of the main mouse component materials as most of it is interior to the mouse. The thresholds are ordered according to their total attenuation coefficients, but their exact placement is somewhat arbitrary. The results of the simulation are not expected to depend significantly on the exact threshold levels, as the attenuation and scattering properties of water, soft tissue, and muscle are extremely similar.

Fig. (5c) shows the resulting indexed mouse map. The skull, ribs, and tailbone are visible as bone and correspond nicely with the same features in the original attenuation map of Fig. (5a). With this thresholding, the mouse composition turns out to be 0.69% air, 25.42% water, 23.54% muscle, 43.20% soft tissue, and 7.15% bone.

Fig. (6) shows the results of the simulation on the indexed mouse attenuation map with $2 \times 10^9$ photons. The detected primary radiation map embodies a mouse projection that is quite detailed; the skeleton is nicely revealed and small features like the ribs are clearly visible. The detected scattered radiation has the form of a slowly varying bulge roughly centered on the detector, perhaps offset slightly. The primary map is used to segment the mouse from the background, as we are interested in the amount of scattered radiation detected in the region behind the mouse. The scatter to primary ratio in this region of interest is $S/P = 0.98\%$.

Fig. (7) shows the results of the simulation with $2 \times 10^9$ photons on the same mouse indexed only with soft tissue. The primary map is now structureless; it shows well the outline of the mouse and indicates the thickness of the mouse in the beam direction. The scatter map looks very similar to that resulting from the five component mouse. The scatter to primary ratio in the region behind the soft tissue mouse is $S/P = 0.92\%$, which is only slightly less than that for the five component mouse. Both simulations find an average $S/P$ of approximately 1%, which should be considered quite small in the sense that it is probably not statistically correctable. Fig. (8) shows the primary and scatter maps for both mouse objects along the $x$-voxel = 100 reference line. The primary radiation map for the five component mouse projection has noticeable structure compared to that for the soft tissue mouse, but two primary maps have basically the same shape along this line. The scatter maps for the two mice along the reference line appear very similar, that from the five component mouse being perhaps...
slightly larger. Note that the scatter is magnified by a factor of 50 so its form can be seen on the plot.

V. Conclusions

The average S/P value is 0.98% with a maximum of 1.88% for the five-component mouse. The average and maximum S/P values for the soft tissue mouse are 0.92% and 1.60%, respectively. These ratios are all so small that scatter correction, which is inherently statistical, is probably not worthwhile for MicroCAT II studies involving mice.

The Monte Carlo code developed here can be used to simulate any scanner simply by modifying the scanner geometry. Future applications could involve untangling beam hardening effects from scatter contamination effects. These two error sources induce very similar artifacts, namely cupping, streaking, and reduction of image contrast. It would also be relatively simple to simulate SPECT imaging by modeling the x-ray source characteristics and detector geometry specific for that modality.

REFERENCES

Development of 3D Cone-Beam Micro-CT Reconstruction

Hsiao-Mei Fu, Ho-Shiang Chueh, Wen-Kai Tasi, and Jyh-Cheng Chen

Abstract—Purpose: Cone-beam micro-CT can offer comprehensive 3D information and high resolution images. This technology can be used with other imaging modalities in the preclinical application of molecular imaging research. Image reconstruction algorithm affects final image spatial resolution, which is the primary topic of this study. We use two types of different methods which are analytic and iterative algorithms to reconstruct the micro-CT images for comparison. Materials and Methods: In this study, projection data of the defrise phantom were obtained using in-house designed micro-CT and images were reconstructed by analytic algorithms, tent-FDK algorithm and FBP. For validation of the iterative reconstruction algorithm (transmission maximum-likelihood algorithm), we have done the simulations using the shepp-logan digital phantom. Human tooth sample and mice bone sample data were reconstructed by analytic and iterative algorithms. Results: The defrise phantom results show the coronal view of the images reconstructed by tent-FDK and FBP algorithm. From the profile of the results, image reconstructed by tent-FDK has sharp edge and uniform pixel value in the high density layer. Comparison of the results of human tooth sample and mice bone sample reconstructed by the two kinds of algorithms, the image reconstructed by maximum-likelihood algorithm has less artifacts and more correct pixel value standing for the attenuation coefficient. Conclusion: Tent-FDK algorithm is a kind of useful analytic method to reconstruct cone-beam CT data. We also develop another iterative method to reconstruct images for low dose x-ray cone-beam CT. The micro-CT images for preclinical study are reconstructed by these two kinds of methods.

Index Terms—micro-CT, 3Dimage reconstruction, cone -beam

I. INTRODUCTION

X-ray computed tomography is an imaging modality that produces cross-sectional images representing the X-ray attenuation properties of the body. Image formation is obtained by using thin x-ray beam and scanning the entire field of view [1]. Parallel beam and fan-beam geometry were used popularly in clinical computed tomography machines. Cone-beam geometry is another approach to acquire the information of attenuated signals. A cone-beam x-ray computed tomography is defined by a focal point with a two dimensional detector. Unlike traditional computed tomography using linear array detector, it can obtain volume information by a planar detector in a single scan.

In three-dimensional computed tomography study, the analytic reconstruction algorithm is the primary method of choice to reconstruct images. The FDK algorithm offered by Feldkamp · Davis · Kress is the first available analytic algorithm to reconstruct three-dimensional cone-beam CT data [2]. However, it may cause image blurring in major axis direction. There are some modified algorithms from the FDK algorithm. G.Wang brought up the generalized Feldkamp algorithm in 1993 [3]. Tent-FDK is another modified algorithm which used rebinned CT sinograms to reconstruct the cone-beam CT data by M.Grass in 2000 [4]. Different from analytic methods, statistical iterative method is another approach to reconstruct CT images. Statistical character of photons is used in this kind of algorithm and expected image is iteratively calculated [5,6]. The first paper to propose a maximum-likelihood approach for transmission tomography appears to be due to Rockmore and Macovski in 1977. In 1995, Jolyon A.Browne derived a gradient-based algorithm from the earlier mathematical modeling of Lange-Carson. The maximum-likelihood algorithm is a kind of statistical iterative algorithm for transmission finite-beamwidth computed tomography [5]. We develop a cone-beam micro-CT system to acquire volume information and reconstruct images by two kinds of above mentioned methods. One of them is T-FDK algorithm and the other is transmission maximum-likelihood (TML) algorithm. In this study, we compare and evaluate the micro-CT data reconstructed by these two methods.

II. MATERIALS AND METHODS

A. Tent-FDK algorithm

For a given 2D detector we can acquire fully 2D projections of the object. Stacking all the projections results in a 3D dataset, this can be rearranged as sinograms. The cone-beam 3D dataset can be rebinned into parallel-beam geometry. Let S be the distance from the source to the detector. Cone-beam to parallel-beam rebinning can be applied:

\[
d^\varphi_{\max} = S \sin(\beta_{\max})
\]

and

\[
d^\rho_{\max} = S \tan(\gamma_{\max}) \cos(\beta_{\max})
\]

where \( \beta \) and \( \gamma \) are the fan-angle and cone-angle. \( d^\varphi \) and \( d^\rho \) are the indexes of rebinned dataset of parallel-beam in rectangular plane. The schematic view of rebinning step is given in Fig.1. Therefore, we can reconstruct images by the rebinned sinograms [4].

H.M. Fu, H.S. Chueh, W.K. Tasi, and J.C. Chen are with the Institute of Radiological Sciences, National Yang-Ming University, Taipei 112, Taiwan, ROC. Telephone:886-2-28267000-5807.
E-Mail:g39220004@ym.edu.tw, g39020015@ym.edu.tw, s19104024@ym.edu.tw and jccchen@ym.edu.tw
B. Transmission maximum-likelihood (TML) algorithm

The number $\bar{y}_{jk}$ of photons exiting ray $j$ of beam $k$ are Poisson distributed with a mean expressed by

$$\bar{y}_{jk} = I_{jk} \exp(-\sum_i u_i * L_{ijk})$$

(3)

$L_{ijk}$ is the intersection length of every pixel index. We assume the detected photon number follows Poisson distribution. The conditional probability density function of the photon number is

$$P(y_k | \mu) = \frac{\bar{y}_k^{y_k}}{y_k!} \exp(-\bar{y}_k)$$

(4)

$y_k$, the number of photons exiting beam $k$, is the summation of $y_{jk}$ over $j \in \text{beam } k$.

$$\bar{y}_k = \sum_{j \in k} \bar{y}_{jk}$$

(5)

The likelihood function is defined as the probability of $Y$, which is the vector notation for the set of all $y_k$. Owing to statistical independence, this probability is expressed as

$$P(Y | \mu) = \prod_k P(y_k | \mu)$$

(6)

The main principle of ML estimation is to solve for the set of pixel x-ray linear attenuation coefficients denoted by the vector quantity $u = (u_1, u_2, u_3, \ldots)^T$, which maximizes the log-likelihood function.

$$\lambda(u) = \ln P(Y | \mu) = \sum_k \{ -I_{jk} \sum_{i \in j} \exp(-\sum_{l \in i} u_l * L_{ijk})$$

$$+ y_k \left( \ln I_{jk} + \ln \sum_{i \in j} \exp(-\sum_{l \in i} u_l * L_{ijk}) \right) - \ln y_k! \}$$

(7)

And we maximize the log-likelihood function by gradient-based iterative algorithm [5].

C. Materials

To verify the algorithm, we use the digital phantom (shepp-logan phantom) to simulate the projection data and reconstruct them by the FBP and the ML algorithm [4, 5]. Then the defrise phantom (shown in Fig.2), human tooth sample and mice bone sample were imaged using our Micro-CT. The in-house designed micro-CT includes X-ray-tube · GOS-CMOS detector(1024 ×128 array) · rotation stage and computer system. The reconstructed image size can be 512 × 512 · 256×256 or 128 ×128, FOV is about 2 × 2 (cm$^2$). The micro-CT system setting is shown in Fig.3. The real system is shown in Fig.4.

III. RESULTS

The defrise phantom image shown in Fig.5 is used to test the T-FDK algorithm. From the results of the coronal images reconstructed by the two algorithms, there is almost no difference between them. However, the profile of the coronal view shown in Fig.6 reveals that the result of tent-FDK algorithm has more uniform in region of every high density layer.
Fig. 5. (a) The coronal view of defrise phantom image reconstructed by the FBP. (b) The coronal view of defrise phantom image reconstructed by the T-FDK.

Fig. 6. (a) The profile of the coronal image reconstructed by the FBP algorithm. (b) The profile of the coronal image reconstructed by the T-FDK algorithm.

For verifying the TML algorithm based on parallel beam assumption, the 2D Shepp-logan digital phantom was used. We generated 120 projections by radon transform and reconstructed it by FBP and TML algorithm for comparison. Fig. 7(a) shows the FBP reconstruction of the Shepp-logan phantom, it can be seen that there are clear streak artifacts. Fig. 7(b) shows the image reconstructed by the TML algorithm. Although the contrast is not better than Fig. 7(a), the streak artifacts has been reduced.

Fig. 7. (a) FBP reconstruction for Shepp-logan phantom 120 projections. (b) TML reconstructed image. The iteration number is 50.

Fig. 8. (a) Human tooth sample reconstructed by the T-FDK algorithm. (b) TML reconstructed image. The iteration number is 30.

Fig. 9. (a) Mice bone sample reconstructed by the T-FDK algorithm. (b) TML reconstructed image. The iteration number is 40.

Fig. 10. 3D image of the mice bone marrow cavity was reconstructed by surface rendering.

The human tooth sample and the mice bone sample were imaged by the in-house designed micro-CT. They were all obtained with 400 projections and reconstructed by the two...
kinds of algorithms. For conforming to the assumption of the TML algorithm, the projection data were rebinned to the parallel geometry in a virtual plane first. Figures 8 - 9 show the results of the real data reconstructed by these two algorithms, tent-FDK and TML algorithm. From the results, the streak artifacts are reduced by the TML algorithm. In Fig.9(a), the ring artifacts which are produced by non-uniform detector response are also reduced by the TML algorithm shown in Fig.9(b). Fig.10 shows the 3D image of the bone marrow cavity reconstructed by surface rendering, the inside honeycomb structure can been seen.

IV. DISCUSSION

From the results of the defrise phantom experiments, it seems that there’s no difference between the two reconstructed images due to the small cone-angle which is $2.5^\circ$. From the profile in the tent-FDK of the coronal view, the effect of rebinned step emerged that regions of every high density layer is smooth and uniform. The overall behaviour of the result of the tent-FDK method is smoother and the rebinned step is efficacious [4]. Our results has the consistent outcome. The digital phantom (shepp-logan phantom) is used to verify the TML algorithm. The TML algorithm is based on parallel beam geometry. Although the cone-beam micro-CT data is not a parallel beam data set, it is rebinned and conformed to the assumption. We implement the TML algorithm on the simulation data produced from the digital phantom first. Images from the simulation are shown in Fig.7. There is distinct streak artifact in the image of Fig.7 (a) but not in the image of Fig.7 (b). In a paper[5], the edge of details of the simulated data was clearer by using the ML algorithm. In our simulation, the images reconstructed by the TML algorithm are clearer and conform to the previous study. Besides, the pixel value in Fig.7 (b) is proportional to the attenuation coefficient and there is no negative value. We use the same algorithm on the human tooth sample and mice bone sample. The streak artifacts are also reduced by the algorithm in Fig.8 (b) and Fig.9 (b). Micro-CT is usually used to evaluate the biopsies of bone. In a study, the 3D display helps to see the bone structure [7]. The bone marrow cavity can be reconstructed by surface rendering with suitable threshold choice.

In the TML algorithm, when to stop the iteration is an issue problem of the method. We adopt the RMSD (root mean square difference) to determine to iteration number. The RMSD is expressed as

$$\text{RMSD}^{u}_{n+1} = \sqrt{\frac{1}{N} \sum_{j=1}^{N} (\hat{u}_j^{n+1} - \hat{u}_j^n)^2}$$

(8)

The RMSD of every iteration for the mice bone is shown in Fig.11. It is decreasing until iteration number 40.

V. CONCLUSION AND FUTURE WORK

T-FDK algorithm is a useful method to reconstruct the 3D cone-beam CT images. Iterative reconstruction method is developed to become useful algorithm for reconstructing transmission images. The TML algorithm can improve the low dose micro-CT image quality. In the future, we will speed up the algorithm and develop an EM method to solve the ML estimation problem. The image quality of different reconstructed algorithms will be analyzed quantitatively.

REFERENCES

Cluster reconstruction strategies for microCT/nanoCT scanners

Xuan Liu, Alexander Sasov

Abstract — We describe here a cluster type reconstruction for microCT and nanoCT scanners, using a few dedicated fast PC’s. The aim is to speed up reconstruction to cope with the ever-increasing data size of our scanners. The goal is achieved in two folds: firstly, divide a reconstruction task among fast PC’s; secondly, optimize reconstruction process itself by making use of full capacity of the PC’s and by optimizing reconstruction algorithm in various ways. The algorithm of choice is straightforward Feldkamp cone-beam reconstruction. Preliminary results show approximately 4N times increase in speed (N: number of PC, tested up to N=5. The factor 4 is gained by algorithm optimization) increase in speed.

Key words: microCT, nanoCT, cone-beam, cluster reconstruction, optimization

I INTRODUCTION

Micro computerized tomography (microCT/nanoCT) technique is being used in an increasing range of applications such as biomedical, geology and material research (see our website [1] for examples of these application fields). Driven by these applications, the resolution of microCT scanners is being improved dramatically, which inevitably leads to fast increase in data sizes. In many cases the long reconstruction time becomes a limiting factor to make full advantage of available hardware. This motivates us to develop a cluster type reconstruction to speed up reconstruction. Since the reconstruction problem does not scale up linearly, datasets with different format require different optimization methods.

The reconstruction program described here serve several commercially available microCT/nanoCT scanners: SkyScan 1076-in vivo microCT for small animal imaging, SkyScan 1172/1072 high resolution general purpose microCT, SkyScan 1074 portable scanner, SkyScan 1078 ultrafast microCT with a scanning/reconstruction cycle less than a minute, and SkyScan 2011 nanoCT instrument. Table I shows more details of these scanners. All scanners have a cone-beam geometry with a circular orbit. The 1076 and 1172/1072 scanners can do spiral scan as well (reconstructed by another program). Two types of circular-orbit configuration are used: rotating stage with fixed source-detector pair for in-vitro scanners, and rotating gantry with fixed animal bed for in-vivo scanners. All scanners can be operated in either short-scan (180+ degrees) or full-scan (360 degrees) mode. The in-vivo scanner 1076 allows for multi-bed scans to increase the axial FOV. Most scanners can shift detector during acquisition to double transaxial FOV. As shown in table I, projections from these scanners range from 512x512 (binning during acquisition) to 8Kx2K (detector shift during acquisition). As determined by Shannon’s law, more projection views would be needed as well for finer pixel sizes to obtain the potential high resolution.

We use Feldkamp cone-beam reconstruction [2] for all scanners. Due to the large data format, iterative algorithms are for the time being out of consideration. In case of short-scan, the Parker’s filter [3] is used to correct for data redundancy. Despite the cone-angle (3 - 20 degrees in axial direction) in use and measurement noise, this algorithm still yields fairly good images. For image examples, we refer you to [1].

<table>
<thead>
<tr>
<th>Scanner</th>
<th>Geometry</th>
<th>Detector</th>
<th>Pixel size</th>
<th>FOV (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SkyScan-1172a</td>
<td>Rotating stage</td>
<td>4000x2300</td>
<td>&lt;0.9µm</td>
<td>35/68</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1280x1024</td>
<td>&lt;1.6µm</td>
<td>20/37</td>
</tr>
<tr>
<td>SkyScan-1076a</td>
<td>Rotating gantry</td>
<td>4000x2300</td>
<td>9/18/35µm</td>
<td>35 /</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>68</td>
</tr>
<tr>
<td>SkyScan-2011</td>
<td>Rotating stage</td>
<td>1280x1024</td>
<td>120-150nm</td>
<td>11</td>
</tr>
<tr>
<td>SkyScan-1078</td>
<td>Rotating gantry</td>
<td>1280x1024</td>
<td>47/94µm</td>
<td>48</td>
</tr>
<tr>
<td>SkyScan-1074a</td>
<td>Rotating stage</td>
<td>768x576</td>
<td>40/22µm</td>
<td>30/16</td>
</tr>
</tbody>
</table>

The scanner models 1172 and 1076 may roughly double the transaxial field of view by shifting the detector. To speed up acquisition, pixels can be binned to reduce the image size. These two techniques introduce a large variation in projection volume: 512³ – 1Kx8Kx2K, with resulting image volume of 512³ – 2Kx8Kx8K.

Data pre-processing includes detector flat-field correction, smoothing or filtering, ring artifacts correction and beam-hardening correction. The flat-field correction is done during acquisition. To reduce ring artifacts caused by inhomogeneous detector response, small random movement of the detector has been introduced and further correction with median filter can be applied during reconstruction. For beam-hardening effects, a set of hardware filters made of aluminum or copper can be used during acquisition to reduce this effect. In the software, the linear model described in [4] is used. Due to the large variety of scanning objects, user interaction is however often required to adjust beam-hardening correction level. For scanners with rotating-stage geometry, source-detector and rotation center may be misaligned. This is also corrected during pre-processing.
II SCANNING GEOMETRY AND DATA

A. Scanning geometry

As shown in table 1, we have several scanner models. Among them, the new generation 1172-high-resolution model has the most flexible geometry and the largest data format, which imposes the highest request for reconstruction. To simplify description, we generalize all scanners to a circular-orbit cone-beam geometry as shown in fig. 1, using 1172 scanner as a typical instance.

![Circular-orbit cone-beam geometry](image)

Fig. 1. Generalized scanner geometry: the X-ray source located at S and the detector centered at O’ form a rigid body which rotate on a circular orbit around point O, the center of FOV. The field-of-view, in which the scanning object should locate, is centered at point O in a x-y-z coordinate system. In case of rotating-stage geometry, it is the object which rotates around z-axis at point O, while the source-detector pair remains stationary.

The SkyScan-1172 has a rotating-stage geometry. Contrary to most scanners of this kind, it has a novel architecture in which both the sample stage (centered at O as shown in fig. 1) and the detector are movable along line SO’. This architecture allows favorable optimization among resolution, FOV size and scanning speed. The maximum source-detector distance (SO’) is 340 mm, while the sample stage can move freely between the source and the detector. The largest detector by this scanner uses a 10-megapixel (4000x2300 ) 12-bit cooled digital CCD camera which is coupled by optical fibers to scintillators. If working in the detector-shift mode, the detector area can be doubled. Table II summarizes the system geometry for SkyScan 1172 scanner.

<table>
<thead>
<tr>
<th>Focal length (SO)</th>
<th>16 – 190mm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. projection image format</td>
<td>8000 x 2300 pixels</td>
</tr>
<tr>
<td>Cone angle (horizontal)</td>
<td>10 – 36 degrees</td>
</tr>
<tr>
<td>Cone angle (vertical)</td>
<td>5 – 9 degrees</td>
</tr>
<tr>
<td>Transaxial FOV (diameter)</td>
<td>6 – 70 mm</td>
</tr>
</tbody>
</table>

B. Data acquisition

All scanners operate in a step-and-shoot mode. According to the application needs, a few parameters can be adjusted freely: the rotation step which is often coupled to the image size chosen; the exposure time per frame and the number of frames to average per angular position; with or without hardware filter to reduce beam-hardening effects; x-ray source voltage etc.. In addition to these, the 1172 scanners can also adjust the detector and scanning stage positions.

Though it is out of the scope of this article, it is worthy of mentioning that studies still need to be done to optimize these parameters in function of scanning time and reconstruction time, the latter being the major limiting factor in throughput if large data format is used. Table III. gives you an idea of the typical scanning times.

<table>
<thead>
<tr>
<th>Cross-section Matrix (pixels)</th>
<th>Rotation step (deg.)</th>
<th>Scan time (minutes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1Kx1K</td>
<td>0.6 – 0.9</td>
<td>5 - 15</td>
</tr>
<tr>
<td>2Kx2K</td>
<td>0.3 – 0.6</td>
<td>20 – 60</td>
</tr>
<tr>
<td>4Kx4K</td>
<td>0.2 – 0.3</td>
<td>90 – 250</td>
</tr>
<tr>
<td>8Kx8K</td>
<td>0.2</td>
<td>180 - 500</td>
</tr>
</tbody>
</table>

III RECONSTRUCTION

A. Algorithm of choice: Feldkamp algorithm

Although the scanning geometry depicted in fig. 1 does not satisfy Tuy’s condition for exact reconstruction, good image quality can be still obtained by the Feldkamp’s [2] algorithm, which is simple, robust and efficient.

As well known, the back-projection step in the algorithm takes most of the computation time. The usual ray-tracing method becomes impractical due to the large data format and the cone-beam geometry: the whole image volume would require roughly from 500 MB to 512 GB memory and it would be extremely inefficient to traverse across the cross-sections. Therefore the back-projection is done by voxel-driven method.

In the voxel-driven backprojection, the major computation process is to find the (u,v) coordinates of the impacting point (P(u,v) on the detector surface) of the X-ray passing through a point P(x,y,z) in the object space, as depicted in figure 1. Let’s consider a fixed source-detector position θ and a certain projection line S-P-P’, and denote the (u,v) coordinates as U(θ, x,y,z) and V(θ,x,y,z). To simplify things, we can shift the detector plane so that point O’ and O coincide and we can denote the source-detector distance (which equals to the focal length now) as D. The following can be easily obtained:

\[ U(\theta, x,y,z) = \frac{(ycos\theta - xsin\theta)}{[1 + (ysin\theta + xcos\theta)] / D} \]
V(θ,x,y,z) = z / [ 1 + ( y sinθ + x cosθ ) / D ]

Note that U(θ, x,y,z) is independent of z-coordinate.

So far, three voxel-driven backprojection strategies have been implemented:

1) Reconstruct per slice, holding both image and projections in memory. This is fast but it is only possible when the volume to be reconstructed is small enough to fit into the available memory. This strategy is mostly used in “preview” mode, while trial reconstruction is performed to adjust reconstruction parameters.

2) Reconstruct per slice, holding projections needed in memory. This strategy does not take the advantage of the z-independence of U(θ, x,y,z). If the full projection volume is too large, the reconstruction is divided accordingly. However, this strategy will fail if the volume of projection needed for 1 cross-section exceeds the available memory.

3) Reconstruct per angle, holding cross-sections in memory and keeping the needed projection data on extended memory (hard disk). This way we can make use of the fact that U(θ, x,y,z) is independent of z-coordinate: for each angle θ, U(θ, x,y,z) and V(θ,x,y,z) are pre-computed and then used to reconstruct all cross-sections in memory. If the full cross-section volume is too large, it is divided accordingly. Although the hard disk is used as extended memory, we still notice about 25% speed increase in general. This method works as long as the memory is big enough to hold 2 cross-section images and there is enough disk space to hold the temporary projection data. It is therefore preferred than the reconstruct-per-slice method. This is the method currently in use.

B. Cluster reconstruction: architecture

Our cluster reconstruction consists of 2 programs: the user-interface (UI) program and the reconstruction program. Both programs run under windows. The two programs communicate with each other using a client-server model. It is designed to run on a few dedicated PC’s connected by fast network. Figure 2 shows our network configuration as a typical situation: 4 identical reconstruction nodes (dual Intel-Xeon 3.4 GHZ processors, 800MHZ FSB, 2 GB dual-channel RAM) connected to a gigabit(10/100/1000) network, to which a host computer (dual Intel-Xeon 2.8 GHZ processors, 533 MHZ FSB, 1 GB RAM ) is also connected. As a matter of fact, the cluster reconstruction can be also run on one and the same computer, or on any other suitable PCs on the network if delays due to network are acceptable. Data are exchanged via shared data drive through the network.

C. Cluster reconstruction: distribution and optimization

The reconstruction is designed for use interactively. One of the concern is that the reconstruction task should be divided such that all PC’s will finish a certain task simultaneously, so that no PC will lag behind. Dividing the task into many small pieces and distributing them whenever a PC is available seems to be a logical solution at first sight, but it is not efficient due to the cone geometry (convolution may take a very significant part of the reconstruction time at off-centered slices). Instead, we choose to divide the reconstruction task only once, taking the computing power of the PC’s into account.

An obvious way is to divide the number of cross-sections accordingly. This method works rather well for data format not larger than 2000x2000. For larger dataset, certainly those of 8000x8000, we have observed severe reconstruction speed variation across axial positions. This is largely due to the inclined angle at off-centered slices. To incorporate this effect and other options in the software (possibility to reconstruct only part of the full volume), convolution and backprojection speeds are recorded automatically per PC and this information is then used to compute estimated reconstruction time for a given task on a given PC. The nice thing of this method is that it takes into account not only the cone geometry but also the computing power of the individual PC. This division method has been implemented and is being in use. It works sufficiently well for most situations.

The major optimization in implementation has been done so far is the different strategies in backprojection, as
described in section III. Further improvement in speed can be expected by fine-tuning of the reconstruction, taking the computer processor architecture into account.

Table IV shows a few example of the reconstruction speed in function of data size and the axial position of the cross-sections.

<table>
<thead>
<tr>
<th>Data matrix</th>
<th>Recon.Time Per slice, per PC</th>
<th>Recon.Time On 4 PC’s (full volume)</th>
</tr>
</thead>
<tbody>
<tr>
<td>207 of 1Kx1K-&gt;1K x 1Kx1K</td>
<td>1 s</td>
<td>4m 20s</td>
</tr>
<tr>
<td>311 of 2000x2000-&gt;1942 x 2000x2000</td>
<td>4.9 s</td>
<td>42 m</td>
</tr>
<tr>
<td>952 of 4Kx2K-&gt;2Kx4Kx4K</td>
<td>1 m</td>
<td>~ 10 hours*</td>
</tr>
<tr>
<td>978 of 8Kx2K-&gt;2Kx8Kx8K</td>
<td>9 m</td>
<td>~ 3 days*</td>
</tr>
</tbody>
</table>

* Extrapolated based on reconstruction of the central 100 slices using 4 PC’s .

IV CONCLUSIONS AND DISCUSSIONS

We have developed a cluster type reconstruction program for microCT and nanaoCT data ranging from 512x512x512 to 2Kx8Kx8K. The preliminary results show that it is practical to have reconstructions up to 2Kx4Kx4K. 2Kx8Kx8K reconstruction still requires days which is less practical, but it is possible.

V ACKNOWLEDGEMENT

The authors thank the colleagues at SkyScan for providing data and their willingness to test and evaluate the program.

VI REFERENCES

Three-dimensional Focus of Attention for Iterative Cone-Beam Micro-CT Reconstruction

Thomas M. Benson and Jens Gregor

Abstract—Three-dimensional iterative reconstruction of high-resolution CT data poses significant implementation difficulties due to the associated computation and storage burdens. In this paper we present a data-driven preprocessing technique called focus of attention that heuristically separates the image and projection domains into object and background before reconstruction. Significant savings are then achieved in terms of reconstruction time and required memory by removing the background regions from computational consideration and focusing entirely on the object regions. We present the results of using this technique for a three-dimensional reconstruction of real data acquired from a micro-CT scanner.

I. INTRODUCTION

When applied to small animal imaging and other high-resolution applications, three-dimensional iterative reconstruction poses significant implementation difficulties due to the extensive computational burden. We recently demonstrated the advantages of utilizing distributed computing techniques as well as ordered subsets to reduce the total reconstruction time [1], [2]. Additionally, we have presented a data-driven focus of attention (FOA) scheme that reduces the amount of projection and image data to be considered during reconstruction [1], [3]. The approach in the previous work is to extract a computational region of interest in the form of a cylinder with a convex cross-section that is large enough to enclose the entire object being imaged. We shall henceforth refer to this approach, which is relatively simple to implement since it disregards the axial variations of the object, as 2D extended FOA. In this paper, we present an algorithm for true 3D FOA. We show that the resulting computational region of interest is much smaller and that this translates into substantial computational savings. We present experimental results based on mouse data acquired with a MicroCAT II (CTI-Concorde Microsystems, Knoxville, TN), which is a circular orbit cone-beam micro-CT system for small animal imaging [4].

II. ITERATIVE RECONSTRUCTION

The typical goal of an iterative reconstruction is to find an approximate solution to the linear system

\[ A\mu = p \]  

where \( a_{ij} \) represents the contribution of the \( j \)th voxel to the attenuation of the \( i \)th projection ray, \( \mu_j \) represents the attenuation coefficient of the \( j \)th voxel, and \( p_i \) represents the log-normalized recording for the \( i \)th projection ray. Many of the algorithms used for iterative reconstruction can be compactly expressed using forward and backprojections of the system matrix, or \( A\mu \) and \( A^T p \), respectively. The work presented here is based on an ordered subsets version of Simultaneous Iterative Reconstruction Technique (SIRT) [5], [6] which is given by

\[ \mu_{(k+1)} = \mu_{(k)} + \frac{\sum a_{ij}(p_i - a_{ij}\mu_{(k)})}{\sum a_{ij}} \]  

where \( a_{ij}, \mu_j \), and \( p_i \) are as before. This algorithm can also be written using the aforementioned projectors as

\[ \mu_{(k+1)} = \mu_{(k)} + \{ A^T [(p - A\mu_{(k)}) \odot r] \} \odot c \]  

where \( r \) and \( c \) are row and column sums of \( A \), i.e., \( r_i = \sum_j a_{ij} \) and \( c_j = \sum_i a_{ij} \), respectively, and \( \odot \) denotes element-wise division.

The computational cost of SIRT, as well as most other iterative algorithms, is proportional to two times the number of non-zero system matrix elements. Since the system matrix can be incredibly large, parallel computing is warranted in order to achieve reasonable computing times for high-resolution reconstructions. This, however, does not solve the problem entirely. The memory requirements associated with an iterative algorithm are tremendous even when computing the system matrix on-the-fly. Consider, for example, the task of reconstructing a \( 512 \times 512 \times 1022 \) image. Using single-precision floating-point arithmetic, close to 1 GB of memory is required to store a single copy of the image. For SIRT, two additional image sized data structures are needed, namely, one for the backprojection computed update values and one for the normalization values corresponding to the column sums. Memory is, of course, also required for storage of the projection data as well as for communicating intermediate results between nodes. All in all, over 3 GB of memory is required per node when reconstructing the aforementioned image size using 360 projections of dimension \( 512 \times 1022 \).

In addition to imposing memory constraints, the large image data structures increase reconstruction time due to interprocessor communications. As mentioned above, we have previously described a reconstruction framework based on distributed computing and ordered subsets. As part of this work, we demonstrated that the increase in interprocessor communications associated with ordered subsets becomes a significant portion of a parallel reconstruction. Fortunately, most images contain many voxels that represent nothing but uninteresting...
background. Identifying and discarding these voxels prior to reconstruction has several advantages. The system matrix effectively contains fewer non-zero elements thus making the forward and backprojections compute proportionally faster. Furthermore, the background voxels need neither be stored nor communicated, yielding critical memory and communications savings.

III. Focus of Attention

Focus of attention (FOA) is a heuristic, data-driven preprocessing technique that reduces the amount of projection and image data considered during reconstruction. The idea can be stated as segmenting both data spaces into object and background in order to subsequently allow all computational resources to be concentrated on the latter. The original FOA algorithm, which was developed first for 2D PET and then adapted for projection based 3D MRI, was based on the theory of support functions and involved solving a quadratic programming problem [7], [8]. For 3D cone-beam micro-CT, we later found that the divergent nature of the sampling geometry called for a slightly different approach [3]. Here we take that approach one step further.

We proceed with an intuitive introduction of 3D FOA and then present our algorithm. Imagine a set of shadowgrams acquired by shining a light at an object from multiple directions. Backprojecting these shadowgrams would yield a region enveloping the original object. Basically, we aim to implement this backprojection process, threshold the enveloping region, and form a 3D FOA region mask from the result. We present our algorithm in Algorithm 1 and address the details in the remainder of this section.

Algorithm 1 Calculation of a 3D FOA region.
1: for all projections \( p \) do
2:  segment \( p \) into object and background
3:  apply median filter to \( p \)
4:  apply dilation filter to \( p \)
5:  for all voxels \( v \) do
6:      project \( v \) to detector plane
7:      if \( v \) projects to object detector pixel then
8:         increment counter for \( v \)
9:      end if
10:  end for
11: end for
12: for all voxels \( v \) do
13:  if counter for \( v \) exceeds threshold \( t \) then
14:     add \( v \) to object
15:  else
16:     add \( v \) to background
17:  end if
18: end for
19: extract FOA region from object

The first issue to address in the algorithm is the segmentation of the projection data into object and background. We perform the segmentation by comparing the non-normalized projection data value for a given detector pixel against the non-normalized blank scan recording for the same pixel. If the detector recording is less than the blank scan recording by a certain amount, presumably due to attenuation, then the detector pixel is considered to represent object data. For this work, we define detector object pixels to be those for which the projection data value is less than 98% of the corresponding blank scan value. With real data sets, the segmentation resulting from this process will likely be noisy in the sense that some pixels will be included in the object that should not be and vice versa. Thus, we employ two standard image processing techniques, namely, median filtering and dilation [9], to improve the segmentation. The \( 3 \times 3 \) median filter reduces the salt and pepper noise resulting from the initial segmentation. Also, we do not want the FOA region to be too tight around the object as that may affect reconstruction quality, especially if part of the object is mistakenly labeled as background. Therefore, we perform a dilation on the thresholded projection using a \( 9 \times 9 \) structuring element to fill holes and expand the FOA region. Finally, we determine the first and last detector object pixel indices in each column and label all pixels in between as object pixels. This serves to fill any holes that may exist in the columns and completes the segmentation of the projection data for a single projection.

Next, we calculate the FOA region in the image space. To do this, we project each voxel to the detector plane by forming the line connecting the X-ray source and the voxel center and calculating the intersection of this line with the detector plane. If the detector plane intersection occurs within an object detector pixel, then we increment a counter corresponding to the voxel. For purposes of faster implementation, and to prevent holes in the columns of voxels, we determine the first and last voxel indices in a given column that project to object detector pixels and increment the counters for all voxels between them. We perform this process for each projection, thus yielding a total count of the number of projections for which each voxel is considered to be in the object. By choosing a threshold value, we then segment the image space into object and background. For the work presented here we use a threshold value corresponding to 90% of the projections, which is quite conservative. Figure 1 depicts several steps of the FOA algorithm for a single transaxial slice of the 3D FOA region.

We parallelize the FOA algorithm by distributing the projections to the available reconstruction nodes. The algorithm implementation is also multi-threaded such that the projections assigned to a given reconstruction node are further distributed to each processor available locally. After each node has completed its projections, we perform interprocessor communications to compute the final counter values for each voxel and to distribute the projection space FOA regions to all nodes. Finally, we continue with step 12 on each node to complete the algorithm.

IV. Results and Conclusions

We have reconstructed a 3D image of a mouse using data from a MicroCAT II scanner that was configured to have a coneangle of approximately 18 degrees. The mouse received
and memory usage was 111. The reconstruction time was reduced to the total) were identified as background voxels. In this case, described above, approximately of memory per node. Using the true 3D FOA approach took 161 million voxels (3D FOA run-times. Without applying a FOA technique, we would have to store attenuation coefficients for over 267 million voxels per image. Since our current computing environment does not have enough memory per node to support such a reconstruction, we cannot compare the FOA and non-FOA reconstruction timings. However, we can compare the 2D extended FOA and the true 3D FOA run-times.

When using the 2D extended FOA approach, approximately 161 million voxels (60% of the total) were identified as belonging to the background. The reconstruction subsequently took 2 hours and 19 minutes and used approximately 1.7 GB of memory per node. Using the true 3D FOA approach described above, approximately 201 million voxels (75% of the total) were identified as background voxels. In this case, the reconstruction time was reduced to 1 hour and 33 minutes and memory usage was 1.2 GB per node. Less than two minutes of this total were devoted to computing the FOA region.

The final reconstruction time for the 3D FOA technique is approximately 33% less than that of the 2D extended FOA method. The additional 40 million background voxels identified by the 3D FOA region correspond to a 37% reduction in object voxels over the 2D extended FOA region. This 37% smaller image data set translates nearly linearly to the MPI communication savings, which are 36% lower for the 3D FOA method. In addition, the computational savings in the system model calculations and projectors is approximately 31% compared to the 2D extended FOA method. These savings together yield the approximately 33% reduction in total reconstruction time. Figure 2 presents several slices of the final reconstruction.

V. CONCLUDING REMARKS

We have presented a technique that can be used to significantly reduce the computational burden associated with high-resolution cone-beam CT reconstructions. Since focus of attention is data-driven, the extent of the savings depends upon the size and shape of the object being scanned. Many objects of interest are significantly smaller than the entire voxel space and will thus have a considerable number of background voxels. Furthermore, there are several parameters that can be manipulated to affect the size of the FOA region, including the threshold levels in the projection and image spaces as well as the median and dilation filter sizes. While we did not notice any loss in the quality of the reconstructions, further study is warranted to analyze the quantitative impact of focus of attention.

REFERENCES

Fig. 2. Mouse reconstructions: Coronal slice 265 (top), sagittal slice 220 (center), transaxial slices 340 (bottom left) and 640 (bottom right). The full reconstruction volume is $512 \times 512 \times 1022$ voxels. The images are thresholded to improve contrast.
Fast reconstruction of 3D TOF PET data by mashing into coarse angles and Most Likely Slice ReBinning

Stefaan Vandenberghe, Margaret Daube-Witherspoon, Robert Lewitt, Joel Karp

Abstract—Faster scintillators like LaBr and LSO have sparked renewed interest in PET scanners with time-of-flight information. The time-of-flight information adds another dimension to the dataset compared to conventional 3D PET. Because storing the data in histoprojections (sinogram with TOF information as an extra dimension) increases the memory requirements significantly, listmode data storage and listmode reconstruction become advantageous. Here we show by simulations and analytical reconstruction that angular sampling can be reduced significantly compared to what is required for conventional PET. Even quite coarse angular mashing does not reduce image quality. This principle can also be extended to the axial direction. By using the time-of-flight information, rebinning the 3D data into 2D data becomes more accurate compared to conventional PET. The advantages of the proposed methods are threefold. Data storage is reduced compared to a sinogram with Time Of Flight information. Compared to listmode format we have the advantage of a predetermined storage space and faster reconstruction. Data are compressed from 3D to 2D which permits fast analytical or iterative reconstruction methods.

Index Terms—TOF-PET, 3D-PET, reconstruction, rebinning

I. INTRODUCTION

PET systems with time-of-flight (TOF) information [1,2] were developed during the 1980’s. Due to the low stopping power and low light output of the detector material used in these systems (primarily BaF$_2$), these systems could not compete with non-TOF based systems based on BGO crystals. Recently there has been an increased interest [3,4] in TOF-PET based on fast scintillators with a sufficiently high stopping power and high light output like LSO and LaBr. Different reconstruction methods were developed in the past for TOF-PET data. This work investigates the necessary angular sampling in transverse and axial directions for use with these methods.

II. RECONSTRUCTION FOR TOF PET

A. TOF kernels

TOF PET acquisition can be described well with Gaussian kernels [5]. The acquired position information by lines of response (LORs) detected along a certain direction follows a 3D Gaussian function with maximum at the source position and a full width at half maximum $\text{FWHM}_\text{TOF}$ along the TOF direction. In the two perpendicular directions it follows a Gaussian with a $\text{FWHM}_\text{Res}$.

B. Data formats

In non-TOF based PET scanners data are usually binned on-the-fly into 2D or 3D sinograms. These sinograms are then reconstructed using projection-based analytical or iterative reconstruction methods. TOF information adds an extra dimension to the data. To store binned data with TOF information would increase the size of the already large projection data for 3D PET scanners by a factor equal to the number of TOF bins. For listmode storage the increase in datasure by adding the TOF information is more modest: it increases by $n\text{events} \times n\text{TOFbits}$. Listmode-based reconstruction is rather slow, however, because forward and backprojection is on an event by event basis. In TOF reconstruction histoprojections (also called preimages) are often used. These are projections with an extra dimension for the TOF difference. When converting the TOF difference to spatial coordinates one sees that they are similar to images rotated along the projection direction. These images are blurred in the projection direction with the TOF kernel (Gaussian with $\text{FWHM}_\text{TOF}$) and with the resolution kernel (Gaussian with $\text{FWHM}_\text{Res}$) in the perpendicular direction. An example of these histoprojections is given in Figure 1

C. Reconstruction methods

A number of reconstruction methods [7,8,9] were developed for TOF-PET in the past. These methods can be classified into different groups.

1. Most Likely Position (MLP) Reconstruction: given an LOR (direction) and a time difference one can calculate the point in image space with the highest probability of being the annihilation point. This is a very fast image reconstruction method using a low amount of memory. Each event is used to update the most likely point in image space with a 1. After the full data set is collected (and image space is completely updated) one can apply an inverse filter to obtain a reconstructed image. This method delivers noisy [6] images because the angular dependence of the probability distribution is not taken into account. It is however a very fast and efficient method. Even in the 80’s it was presented as a real time reconstruction method.

2. Confidence weighted (CW) analytical reconstruction. In this method one takes the probability density function (PDF) into account. Two different implementations are possible. In the...
histoprocession based method data along close orientations is collected. For each event a histoprocession is updated with the PDF (2D-Gaussian with TOF-FWHM and spatial resolution FWHM). After all events are collected, an inverse filter is applied, and the histoprojections are reprojected into image space. In the image space based method, the image space is updated for each event with the rotated PDF. After all events are collected, an image space inverse filter is applied.

3. Iterative methods: due to the nature of the data most iterative methods start from listmode data. One does a Gaussian weighted (TOF direction) forward and backprojection, and uses an iterative algorithm (e.g. the ML-EM method). After a few iterations an appropriate reconstruction is obtained.

Comparisons in the past have shown that iterative reconstruction leads to lower variance compared to CW analytical methods. MLP methods are worse than CW methods because of the increased noise. The method proposed below groups listmode events with close angular orientations (transverse and axially) into histoprojections or preimages. This allows us to speed up reconstructions significantly because forward- and backprojections are per orientation instead of per event. In this work a CW method is used: the inverse filter is applied on the histoprojections. In the next paragraph it is explained why angular sampling can be significantly reduced for TOF-PET.

III. ANGULAR SAMPLING FOR 2D PET AND 2D TOF-PET

A. Angular sampling

For the majority of reconstruction algorithms, image space is sampled by a certain number of voxels $\xi$ in each direction. The size of each voxel ($\Delta x$) is mostly chosen well below the spatial resolution. It is important to select enough angular samples in the projection data to be able to reconstruct the image.

Conventional 2D PET: For each event in the FOV (gray area with radius R in figure 2.a.) one wants to minimize the error due to angular sampling. The biggest error occurs at the edge of the FOV. If this error at the edge of the FOV is less than one pixel then data can be reconstructed correctly. If the number of pixels in x and y directions is equal to $\xi$ then the angular sampling criterion is (see figure 2.a.): $\Delta \phi = \text{atan}(2/\xi)$.

2D TOF-PET: Assume we have a 2D PET system with FWHM$_{TOF}$ and we define $g = R / \text{FWHM}_{TOF}$ where R is radius of the object. With TOF differences, the points can be placed closely to their original position in image space. Sufficient angular sampling is achieved if in this step the error (due to discrete angles) is less than one pixel. In figure 2.b it is shown that this can be achieved by sampling with $\Delta \phi \sim \text{atan}(2g/\xi)$. This can be intuitively understood by looking at the case when FWHM$_{TOF}$ is equal to FWHM$_{Res}$. In this case, the pdf is a symmetric gaussian and is independent of the angle of the LOR. All data can be collected in one histoprocession (preimage) and the same reconstruction will be produced as by using histoprojections for each angle.

B. Simulation

One slice of the MCAT phantom (256x256 matrix, 2 mm pixelsize) was used to generate simulated PET and TOF-PET data (300 ps FWHM$_{TOF} \sim 45$ mm). The simulation was noise free to avoid effects of limited statistics. Analytical reconstruction was used: PET data were reconstructed with Filtered Backprojection (FBP) and TOF-PET data with CW backprojection. A varying number of angles was used to reconstruct the data. Results are shown in Figure 3.a and b. At least 45 angles are needed to obtain a good reconstruction from regular PET data (Figure 3.a.). It is clear that TOF-PET reconstructions can be retrieved from significantly fewer angles: 8 angles already gives a good reconstruction (Figure 3.b).

C. Mashing

A ring PET scanner will however measure a large number of angles. The data size can be reduced by combining close angles. The same slice of the MCAT phantom was simulated to generate PET and TOF-PET data (300 ps FWHM$_{TOF}$) for 90 different angles. The same reconstruction method as before was used. Data from LORs with close orientations were mashed into a varying number of preimages. In the mashing process a coordinate transformation is performed (Figure 4.). The length $l = c \frac{\Delta t}{2}$ along the 3D LOR (oblique angle $\theta$ with the transverse plane) is projected onto the transverse plane and the resulting coordinate in that plane is $l' = l \cos \theta$. The radial distance $s$ and the transverse angle $\phi$ of the 2D LOR are the conventional coordinates used in PET. When a limited number of histoprojections is chosen, one can calculate the equivalent radial distance $s'$ and the equivalent length $l''$ along the LOR of the Most Likely Position in the closest angular histoprocession $\phi_n$. Similar methods were described earlier in [9]. The 2D part of Figure 5 illustrates how listmode events are organized in different histoprojections. Results are shown in Figure 6. It is clear that good TOF-PET reconstructions can be retrieved from a limited number of angles.

IV. REBINNING FOR 3D PET AND 3D TOF-PET

A. Most Likely Slice ReBinning

Different rebinning methods have been proposed in the past. The use of Single Slice ReBinning (SSRB) [10] provides accurate results for objects near the central axis and shows degrading resolution when going radially off-center. The
resolution loss is largest in the axial direction and tangential direction. Multi Slice ReBinning (MSRB) [11] spreads the coincidence over the different slices that the LOR crosses. Fourier ReBinning (FORE) is a better approximation than SSRB and works well for scanners with a large axial acceptance angle. It is mentioned in the paper [12] that the algorithm finds a virtual TOF estimate by using the Fourier transform of an oblique sinogram. This suggests that using the time-of-flight information will reduce the “rebinning error”. We use the TOF information (like first mentioned in [12] and recently in [13]) to assign the listmode event to the most likely slice. Using the TOF information, we rebin the data to the most likely slice. Assume we have an LOR with endpoint 1 \((x_1,y_1,z_1)\) and endpoint 2 \((x_2,y_2,z_2)\) and the TOF measurement was \(\Delta t = t_2 - t_1\).

The Most Likely Slice is then given by:

\[
z_{ML} = \frac{(z_1 + z_2)}{2} - c \frac{\Delta t (z_2 - z_1)}{2d}
\]

with \(c\) the speed of light and \(d\) (the length of the LOR) equal to

\[
d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}
\]

The second term of \(z_{ML}\) adds a TOF based correction to the SSRB method (which is first term only). It is interesting to calculate the maximum error (misplacement in wrong slice) due to the rebinning method. If the axial angle of the LOR is \(\theta\) and the spatial TOF uncertainty is \(\text{FWHM}_{\text{TOF}}\), the slice misplacement has a distribution with \(\text{FWHM}_{\text{slice}} = \sin \theta \times \text{FWHM}_{\text{TOF}}\): eg. the error for the most oblique LOR in a system with 15 degree axial maximum angle and 300 ps \(\text{FWHM}_{\text{TOF}}\) is 12 mm. For less oblique LORs the error is smaller.

**B. Simulation**

Two pointgrids were simulated for a 3D TOF-PET system (5 mm spatial \(\text{FWHM}\) and 424 ps \(\text{FWHM}_{\text{TOF}}\)) with a maximal axial angle of 15 degrees. The center of the scanner is at \((x,y,z)=(0,0,0)\). In pointgrid 1 points were placed 5 cm spaced from each other in the radial direction \((x=0, 5, 10, 15, 20\) and \(25\) mm) and 5 cm spaced in the axial direction \((z=0, 5\) and \(10\) cm). The radial distance between points is the same for pointgrid 2. In the axial direction the spacing is now 1 cm and there are 5 rows \((z=0, 1, 2, 3, 4)\). The data were obtained using the Gate Monte Carlo simulator which has accurate modeling of all interactions in the detector. Listmode data with detector pairs are converted to an LOR with endpoints \((x_1,y_1,z_1)\) and \((x_2,y_2,z_2)\) and \(\Delta t = t_2 - t_1\). Data were rebinned into histograms using the MLSRB method and the SSRB method. Coronal slices through the preimages are shown in Figure 7a, b, c and d. In Figure 7a, we show the rebinned data of pointgrid 1 using MLSRB. In Figure 7b, we show the same data using SSRB. Here it is already clear that using the TOF information significantly reduces the blurring due to rebinning. Figure 7c shows the rebinned data of pointgrid 2 using MLSRB. It is clear that the points can clearly be separated even in the histoprojections. Using the SSRB method (Fig 7.d) the radically off-center points can not be separated anymore.

**V. Conclusion**

It is shown that with TOF information, the angular sampling requirements are reduced. This allows us to mask the data in quite coarse angles. Using the TOF information to rebin the 3D data to the most likely slice is shown to yield images with good axial resolution. Both methods are useful to achieve fast reconstruction methods as forward and/or backprojection needs to be done for only a limited number of angles.

**Acknowledgment**

This work was supported in part by the National Institutes of Health under Grants R21EB001684 and R01-EB002131. We would like to thank Samuel Matej, Gerd Muehllener and Lucretiu Popescu for useful discussions on this topic.

**References**

Figure 1: Histoprojections at different angles.

Figure 2: Angular sampling for 2D PET (2.A) and angular sampling for 2D TOFPET (2.B).

Figure 3.a: 2D PET reconstruction from a varying number of angles (3.a) and 2D TOF reconstruction from a varying number of angles (3.b).
Figure 4. Coordinate transformation used for mashing 2D TOF PET data

Figure 5. Reconstruction with coarse angular sampling

Figure 6. Angular mashing and reconstruction for 2D TOF-PET

Figure 7. Rebinning of 3D TOF-PET data with MLSRB and SSRB into histoprojections (a) pointgrid 1 rebinned with MLSRB, (b) pointgrid 1 rebinned with SSRB, (c) pointgrid 2 with MLSRB and (d) pointgrid 2 with SSRB
Strategy to Estimate Statistical Normalization Parameters of 3D Panel PET Detector with DOI

Mu Chen, Christian Michel, Merence Sibomana, Vladimir Y. Panin, Jeih-San Liow, Ziad Burbar, and Micheal E. Casey

Abstract— A statistical normalization technique was developed for the CPS dual-layer HRRT scanner. A component based model was proposed to include crystal sensitivity, geometric response, and layer identification factors. A maximum likelihood based approach was used to estimate these factors. Emission data were acquired on the HRRT scanner with a rotating rod source in listmode format to preserve counts information in each line of response. A conjugate gradient algorithm was implemented to obtain the maximum likelihood estimation of normalization factors. These factors were then used to construct the normalization sinogram array. Uniform phantom data were acquired and reconstructed to compare the direct to the statistical normalization technique. The results show that the statistical normalization technique achieves better image noise levels than the direct technique. The improved technique could be used to shorten the acquisition time of the normalization. This paper presents a new strategy to estimate the components of the statistical norm model. The geometric component is measured once at factory and other components ($\varepsilon$, $\beta$, and $\gamma$) are measured from separate scans.

I. INTRODUCTION

Normalization factors are corrections that compensate for the inherent non-uniformity of the PET detector pair efficiency. These non-uniformities can be the result of systematic variation in solid angle and distance between detector pairs, drift in electronics and PMT tube gains, and/or variations among crystal responses. Inaccurate normalization can result in artifacts, poor uniformity, and increased noise in the reconstructed images. The traditional normalization technique for HRRT is a direct method in which a known source of activity (a rotating rod) is scanned, then correction factors are obtained as a ratio between predicted and measured counts [1]. The accuracy of this calibration technique is limited by the statistics of the measured counts. For scanners with a large number of line of response (LOR), to achieve acceptable statistical noise levels at a low axial compression (span 3 for HRRT), a very large number of counts, i.e. a very long scan (a few days), is required.

Component-based method [2]–[5] was introduced to improve the statistical accuracy of the normalization factors. This approach models LOR sensitivity as the product of the individual coincident detector efficiencies, a geometrical factor which accounts for the gamma ray incidence angle, and some other factors. It reduces the number of degrees of freedom in the normalization model so the number of counts required can be reduced.

Traditionally, factors of the component-based normalization are computed by averaging over multiple LORs. When the model includes more complicated components such as scatter, block pattern, and dead-time, the averaging method involves more approximation and loses its accuracy.

Statistical estimation of normalization factors has been investigated [6]–[8] to estimate the complicated components simultaneously with the iterative technique. Also by modeling the statistical noise, it could further reduce the total scan time to acquire the normalization.

In this work, we propose a model for the CPS HRRT dual-layer scanner and test its performance. The proposed model includes crystal sensitivity, geometric response, and layer identification factors. A maximum likelihood based approach was used to estimate these factors. A conjugate gradient algorithm was implemented to obtain the maximum likelihood estimation of normalization factors. These factors were then used to construct the normalization sinogram array. Uniform phantom data were then acquired on HRRT and reconstructed to compare the direct to the new normalization technique.

II. MODEL AND METHODS

The CPS HRRT scanner has 8 panel detectors (Fig. 1). Each detector consists two layers of $72 \times 104$ LSO-LYSO crystals. The layer identification is implemented by the phoswich technique [9]. Each detector head is in coincidence with opposing 5 detector heads. The total possible number of LOR is $4,485,611,520$.

Emission data were acquired for 50 hours in listmode format with a rotating $^{68}$Ge rod source. The listmode data then was histogrammed for each LOR. Poisson noise was added to the data to simulate a 10 hour normalization scan to test performance at low statistics. Stored data of counts in each
LOR were also sorted into sinogram format (i.e., parallel projections using nearest-neighborhood rebin approximation) to create direct normalization.

The mean of the true (prompt-delay) detected events is:  
\[ \bar{y} = P_{\text{norm}} \mathbf{x}, \]  
where \( P_{\text{norm}} \) is the normalization factor and \( \mathbf{x} \) is the projected source distribution. The scatter component was ignored since we used a line source in air. The Poisson log likelihood for normalization factors is:  
\[ L(\varepsilon, g, \ldots) = \sum_{n=1}^{N} \gamma_n \log \bar{y}_n(\varepsilon, g, \ldots) - \bar{y}_n(\varepsilon, g, \ldots) \tag{1} \]

where \( \varepsilon \) is crystal efficiency that describes the intrinsic efficiency of the detector crystal, and \( g \) is geometric factor that describes the variation due to the geometric position of the detector pair.

For each of the four layer combinations (front-front, front-back, back-front, and back-back), the ratio of the expected mean value of counts in LOR \( i,j \) to the source distribution is:  
\[ \left( \frac{\bar{y}}{x} \right)_{p,q,i,j} = \sum_{p'=0}^{1} \sum_{q'=0}^{1} g_{pq,i,j} \beta_{p'i} \varepsilon_{p'i} \beta_{q'j} \varepsilon_{q'j} \tag{2} \]

where, \((p,q)\) denotes a layer-pair (0: back; 1: front); \((i,j)\) denotes a crystal-pair, and  
\[ \beta_{p'i} = \left\{ \begin{array}{ll} \beta_{pi}, & \text{when } p' = p \\ (1 - \beta_{pi}) \gamma_{pi}, & \text{when } p' \neq p \end{array} \right. \tag{3} \]

\( g_{pq,i,j} \) is the geometric factor for LOR \((p,q,i,j)\). \( \varepsilon_{pi} \) is the crystal efficiency for crystal \((p,i)\). \( \beta_{pi} \) is the ratio of the total number of counts of crystal \(i\) in layer \(p\) being correctly identified as in layer \(p\), \(p = 0,1\). \( \gamma_{pi} \) is the probability of the counts of crystal \(i\) crosstalked to layer \(p\) being accepted in layer \(p\). The reason for \( \gamma \) not being one is that events identified in the wrong layer may not fit in the energy window for that layer to be accepted as valid events.

For example, the expression of Equation 2 for back-back layer pairs LOR \((0,0,i,j)\) is:  
\[ \left( \frac{\bar{y}}{x} \right)_{0,0,i,j} = g_{00,i,j} \beta_{0i} \varepsilon_{0i} \beta_{0j} \varepsilon_{0j} + g_{01,i,j} \beta_{0i} \varepsilon_{0i} (1 - \beta_{1j}) \gamma_{0j} \varepsilon_{1j} + g_{10,i,j} \beta_{1i} \varepsilon_{1i} \beta_{1j} \varepsilon_{0j} + g_{11,i,j} \beta_{1i} \varepsilon_{1i} (1 - \beta_{1j}) \gamma_{0j} \varepsilon_{1j} \]

Fig. 2 illustrates two panel detectors and the cylinder shell formed by the rotating rod source. Due to the symmetric property of the HRRT scanner, many LORs are geometrically equivalent. This property can be used to reduce the dimension of the geometric factor. We found the geometric factors associated with those unique LORs are those with LORs started from only one quadrant of detector one (see the shaded area of Fig 2).

The maximum likelihood estimation is obtained with the conjugate gradient algorithm [10]. Five iterations were used with 3 sub-iterations of line search at each main iteration. After each line search was completed, normalization factors were truncated so that \( \beta, \gamma \in (0,1) \), and \( \varepsilon, g > 0 \).

### III. Results and Discussions

#### A. Validation of the model

Normalization factors, including crystal efficiency (\( \varepsilon \)), layer identification factor (\( \beta \)), and layer crosstalk acceptance factor (\( \gamma \)), for back and front layers are displayed in Fig. 3. The means of \( \beta \) for the front and back layers are 0.91 and 0.99. The means of \( \gamma \) for the front and back layers are 0.11 and 0.11, respectively. Thus, overall only a few percent of events are
Fig. 3. Normalization factors for each crystal. Left: $\varepsilon$, crystal efficiency; Center: $\beta$, layer identification factor; Right: $\gamma$, layer crosstalk acceptance factor. In each figure, Top: back layer; Bottom: front layer.

identified and accepted in the wrong layer. However localized effects are apparent and can be seen as vertical streaks in $\varepsilon$ and $\beta$ of each layer. Also since $\varepsilon$ and $\beta$ are multiplicative factors to each other most of the time, it is quite difficult to separate them.

The normalized uniform cylinder sinograms using direct method and the proposed statistical component-based method with 10 and 50 hour norm acquisition are displayed in Fig. 4. In both the 10 and 50 hour cases, the statistical method provides less noise in normalized sinograms compared to the direct method.

The reconstructed uniform phantom images are displayed in Fig. 5. Noise in the images are measured as the relative standard deviation of voxel values in the volume of interest, which is a 30 pixel radius, 80 pixel long cylinder inside the phantom. The values are shown in Table I.

| TABLE I |
|------------------|----------|----------|
| VOLUME RELATIVE STANDARD DEVIATION |
|                  | 10 hour  | 50 hour  |
| statistical norm | 0.1952   | 0.1513   |
| direct norm      | 0.2297   | 0.1663   |

In both 50 hour and 10 hour normalization cases, the images reconstructed with the new normalization have superior image uniformity and less noise. However, some vertical (axial) artifacts remain in the component-based normalization image. They are probably due to the fact that scatter component may require a different normalization and deadtime correction for normalization should be accounted for. These preliminary results were published at MIC04 [11].
B. Multi-step strategy

The above approach requires knowledge of counts in each of the $4.9 \times 10^9$ LORs. Currently the way to obtain it is to acquire long scan data in listmode, then histogram into LOR mode. Huge disk space is required to store the listmode file and processing the listmode file takes considerable amount of time. Online processing of the listmode events and estimation of normalization factors is highly desirable. A multi-step approach which is similar to [4] is proposed. First, a long normalization scan is acquired and histogrammed into LOR. The geometric factors, which should be the same among all HRRT scanners, are estimated with this long scan. Second, on each HRRT scanner to be normalized, a short normalization scan is acquired and crystal efficiencies and other factors are estimated with an online algorithm. Normalization matrix is created using saved geometric and other factors afterward.

At this stage, during the online processing, coincident events are histogrammed into counts per crystal. Counts per crystal are later used as crystal efficiency and layer identification factors are ignored.

Estimated crystal efficiencies are displayed in Fig. 6. Reconstructed segment zero images compared to the full model are displayed in Fig. 7. The streak artifact in the simplified model images can be clearly seen. It is very likely that the artifact is due to the lack of layer identification factors in this model.

IV. CONCLUSIONS

In conclusion, we have developed a normalization model for the CPS dual-layer HRRT scanner and a statistical iterative approach to estimate its parameters. This technique could be used to shorten the acquisition time of the normalization while maintaining the noise level in the reconstructed image. Future studies will be performed with modeling separate scatter normalization and including deadtime parameters. A multi-step approach is being investigated to further reduce the scan and processing time.

REFERENCES

Fig. 7. Reconstructed segment zero images with the online histogramming (top) compared to the full model (bottom).