List Mode EM Reconstruction of Compton Scatter Camera Images in 3-D[1]

Scott J. Wilderman[2], N.H. Clinthorne[3], J.A. Fessler[3,4,5], C-h. Hua[3,4], and W. Les Rogers[3,4]

[2]Department of Nuclear Engineering and Radiologic Sciences, [3]Division of Nuclear Medicine, [4]Department of Biomedical Engineering, [5]Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, Michigan 48109

Abstract

A method has been developed for List Mode EM reconstruction of Compton scattering camera images in 3-D, using a previously reported 2-D technique and refining and adapting it to three dimensions. Spatial variation in the system sensitivity is determined by an approximate numerical integration which accounts for solid angle effects, absorption and escape probabilities, and variation in the differential angular scattering cross section. The method for computing the system transition probabilities uses a similar method to determine values in pixels along exact back-projected cones for each detected event, and uses pre-computed values of the inherent system resolution (which includes the effects of spatial and energy measurement resolution and Doppler broadening) to model the response in pixels neighboring the back-projected cone. The algorithm has been parallelized, permitting reconstruction of images using larger number of detected events in relatively constant time by adding additional processors. Results are presented using 3-D simulated data.

I. INTRODUCTION

List-mode Expectation Maximization (EM) methods [1, 2, 3] have recently been applied to the Compton camera reconstruction problem [4, 5]. The method is particularly appealing for this problem because the number of system variables is limited to $N_\gamma N^3$, if N_γ is the number of detected events and N is the dimension of the image space. By contrast, the number of combinations of position and energy measurements which describe a Compton camera in a traditional iterative reconstruction approach can be as large as 10 billion for each N^3 pixels.

The list-mode Maximum Likelihood problem is posed as follows: Let Y be all possible measured projection data, accumulated as individual measurements Y_i, and taking $Y_i \to 1$ for each detected particle, and $Y_i \to 0$ for the infinite number of possible events not detected in the current measurement. The maximization step can then be written as

$$\frac{1}{\lambda_j} \sum_i Y_{ij}^E - s_j \bigg|_{\lambda_j = \lambda_j^{(l+1)}} = 0 \qquad (1)$$

and the expectation step as

$$Y_{ij}^E = Y_i \frac{\lambda_j t_{ij}}{\sum_k t_{ik} \lambda_k} \qquad (2)$$

leading to the iteration (indexed by l)

$$\lambda_j^{(l+1)} = \frac{\lambda_j^{(l)}}{s_j} \sum_i \frac{Y_i t_{ij}}{\sum_k t_{ik} \lambda_k^{(l)}}. \qquad (3)$$

In the above s_j is the probability that a photon emitted from pixel j is detected anywhere, and t_{ij} the probability that γ_i was emitted from pixel j. Barrett *et al*[2] and Parra [3] have proven that this treatment of Y holds (here we ignore any time dependence of the measurement), with the condition that as the detected Y_i do not span the space of all possible events, s_j must be computed as the integral over all possible detected events i, including those for which $Y_i = 0$.

In earlier work [4, 5], a technique for determining the required system matrix coefficients needed in the EM algorithm was developed, in which relative transition probabilities t_{ij} were computed and stored for only those pixels intersecting the back-projected cone corresponding to each detected particle. Transition probabilities for pixels within a given angular range of the intersected pixels were then computed on the fly at each iteration step, with values based on the inherent cone-spread function for the device, which is a function of the energy and spatial resolution in the detectors, as well as the degree of Doppler broadening of the Compton scattered photon energy spectrum. In 2-D, this approximation reduces the average number of elements t_{ij} which must be stored to roughly $2NN_\gamma$, but still allows for modeling of the entire $N^2 N_\gamma$ system. The sensitivities s_j were computed by performing a crude numerical integral over the first detector volume for each pixel j and assuming uniform response in the second detector. Use of this reduced data set permits a full EM iteration to be computed in about 1 minute on a high end Unix workstation, for a 500,000 count problem solved on a 64 x 64 image space.

II. METHODS

In the current work we deploy more accurate computation of the sensitivities and the system matrix coefficients; apply the method to the 3-D case; and parallelize the algorithm using the MPI package [6] to permit running on multiple networked workstations. Since the efficacy of the algorithm has been shown to be limited by data size rather than speed, parallelization brings a linear increase in the size of the problem which can be treated, with constant computation speed (limited to message passing overhead).

A measurement i in a Compton camera system consists of a first detector interaction position $\vec{z_1}$, a second detector interaction position $\vec{z_2}$, and an energy of interaction in the scatter detector E. The factors contributing to the computation of the probability of the emission of a gamma in pixel j

resulting in a measurement i include terms relating to the emission probability, the interaction and escape probabilities in the detectors, the scattering angles subtended by the interaction positions, Doppler broadening of the scattered gamma energy, and uncertainties introduced in the measurement and recording process, among other things. For the sake of simplicity, we choose here to ignore errors in the measurement of the detected positions and scattering in the source. It can be shown then that [7]

$$P(\vec{z_1}, \vec{z_2}, E|\vec{z_0}) = \frac{1}{8\pi^2}\frac{\mu_C}{\mu_t}\left(1 - e^{-\mu_t z_{d11}}\right) f_\Omega(\theta, E) \quad (4)$$
$$e^{-\mu_t z_{d12}}\left(1 - e^{-\mu_t z_{d2}}\right)\frac{\cos(\phi_1)\cos(\phi_2)}{d_{01}^2 d_{12}^2}.$$

In the above expression, $P(\vec{z_1}, \vec{z_2}, E|\vec{z_0})$ is the probability of a particle emitted at $\vec{z_0}$ (which is contained in voxel j) being measured as having interacted at positions $\vec{z_1}$ and $\vec{z_2}$ with energy E; $\frac{\mu_C}{\mu_t}$, the ratio of the Compton to total absorption cross section in the first detector; z_{d11}, the distance traveled in the first detector before the interaction; z_{d12}, the distance traversed in leaving the detector after the scatter; z_{d2}, the distance traveled in the capture detector prior to the interaction; d_{01}^2, the distance to the first interaction; and d_{12}^2, the distance to the second; $\cos(\phi_1)$, the angle of incidence on the face of the first detector and $\cos(\phi_2)$, the angle on hitting the second; and $f_\Omega(\theta, E)$, the probability of scattering through θ with energy E. This function includes the convolution of the energy measurement resolution with the doubly differential cross section which combines the Klein-Nishina cross section with the Doppler broadened spectra. The sensitivities s_j can be calculated then by integrating $P(\vec{z_1}, \vec{z_2}, E|\vec{z_0})$ over the pixel ($\vec{z_0}$), volume of the first detector ($\vec{z_1}$), volume of the second detector ($\vec{z_2}$), and all possible energies (E) between the upper and lower thresholds on the system,

$$s_j = \int_j \int_{D1} \int_{D2} \int_{E1}^{E2} P(\vec{z_1}, \vec{z_2}, E|\vec{z_0}) d\vec{z_0} d\vec{z_1} d\vec{z_2} dE. \quad (5)$$

In the current work, these integrals are computed by a simple numerical integration, in which $P(\vec{z_1}, \vec{z_2}, E|\vec{z_0})$ is evaluated at 4 mesh points inside each pixel, 4 over each of the individual elements of the scatter detector, and 120 points over the larger area capture detector. Most of the components of the expression in (5) can be determined easily, and tables of $f_\Omega(\theta, E)$ can be prepared in advance and for each point in the numerical integral (θ is defined exactly by the three positions). For the current work, however, the effect of Doppler broadening on the sensitivities is ignored, and so $f_\Omega(\theta, E)$ is taken as just the Klein-Nishina cross section convolved with the system energy resolution. The integral of (5) thus becomes a triple sum over the pixel j and the detector volumes of the factors in the expression for $P(\vec{z_1}, \vec{z_2}, E|\vec{z_0})$.

The system transition probabilities t_{ij} are given by the integral of $P(\vec{z_1}, \vec{z_2}, E|\vec{z_0})$ over the emitting pixel j. As described in a previous work [5], to avoid the lengthy computation involved in performing this integral over every pixel for every particle and to avoid having to store this large number of transition probabilities, we compute only the track

through the image space of the back-projection assuming perfect data, and assume that the relative weights of the neighboring pixels can be inferred from the inherent system resolution, which is computed in advance. We are left then with simply integrating the expression for $P(\vec{z_1}, \vec{z_2}, E|\vec{z_0})$ over the pixel volume. In the current work, again the Doppler broadening is ignored, and the integral over the pixel volume is approximated by assuming that $P(\vec{z_1}, \vec{z_2}, E|\vec{z_0})$ is constant over the relatively small and usually distant pixel volume.

The application of the method above to 3-D is straightforward. We simply apply the 2-D algorithm described above to successive planes in the image space. We therefore increase the number of elements for which we must store data elements in going from the 2-D to the 3-D case by a factor equal to the dimension of the image space, to approximately $2N^2 N_\gamma$. There are two potential drawbacks to this method, however. First, as the 2-D method is based on the intersections of conics with planes in the image space, the current 3-D method involves computations of weights for area elements in given planes, rather than volume elements in the image space. Thus weights are computed differently for nearest neighbor voxels depending on whether the neighbor is lateral in the initial 2-D reference plane or the next reference plane. Second, when the track of the initial particle is almost parallel to the orientation of the image plane, the numerical uncertainty involved in computing the conic intersection with the plane is so large that the data must be rotated and the computation done in a different orientation, and then used in that alternative orientation when computing neighboring weights during the iterations. Based on the results from for perfect point sources presented later, neither of these problems appear to have significant effects. Indeed, images of point sources symmetric in both lateral and longitudinal dimensions.

The recursion formula of (3) lends itself easily to parallelization, even though the sums are over pixels in one case and projection data in the other. The data is split by dividing the particles among the processors. This allows for some time saving in the computation of the weights t_{ij}, as, in fact, different processors need not have access to the weights computed by each other. For a problem with M processors and N_γ photons, in the current implementation, each processor first computes for its allocation of N_γ/M photons the weights t_{ij} it requires, and a synchronization of the nodes is then imposed. Next, each processor computes the sum in the iteration formula, $\sum_{M_i} Y_i t_{ij} / \sum_k t_{ik} \lambda_k^{(l)}$. Another synchronization is imposed, and these results are then collected and tabulated by the master processor, which then solves (3) for $\lambda_j^{(l+1)}$. Values of this updated back-projection is then broadcast to all the slave nodes, which then update their local copies of $\sum_k t_{ik} \lambda_k^{(l)}$ for the N_γ/M photons which they are processing. Thus, the only data that must be transfered between the processors are the M sets of the N^3 values of $\sum_{M_i} Y_i t_{ij} / \sum_k t_{ik} \lambda_k^{(l)}$ which must passed up by each slave, and the N^3 values of $\lambda_j^{(l+1)}$ broadcast by the controlling process at each update. The $2N_\gamma N^2/M$ weights are local to each processor. For memory limited problems such as this, this organization allocates resources most efficiently.

The MPI (Message-Passing Interface) [6] software was used to perform the inter-processor communication tasks required here.

III. RESULTS

Results are given here for simulated data generated by Monte Carlo using a geometry similar to that of the C-SPRINT [8] system. C-SPRINT consists of a 32 by 8 array of 1.4 mm silicon detector elements .3 mm thick as the scatter detector and the SPRINT second detector. Projection data was generated by Monte Carlo simulation using the program SKEPTIC [9], which has been extensively employed and tested in the simulation of Compton scatter cameras [8, 4].

Two phantoms have been modeled. The first consists of two point sources in a cold background, one centered with respect to the the front face of C-SPRINT, and one at a position .24 cm behind and .16 cm to the right of the center point. 3-D measurements were simulated by performing separate simulations of rotations of the phantom through 36 equal 10 degree steps. A 31x31x31 image space with .02 cm pixels was used in the reconstructions, with 8208 photons from the central point and 7491 events from the off-axis point modeled. For the purposes of validating the method, exact energy and position measurement data was used, and Doppler broadening of the scattered gamma energy was ignored. The initial back-projection and computation of the weights took approximately 3 minutes on a Sparc Ultra 1 workstation and each iteration roughly 20 seconds. No parallelization was required.

Results of reconstructions are presented in figures 1 through 3. All images are from the 20th iteration, and represent the slices at Z = 0.0, .12, and .24 cm. It is quite clear from the figures that the method is able to produce point images with the correct position in all three dimensions, and that the images are symmetric in all three axes. Further, the computed relative intensity of the two points is .916, which is in excellent agreement with the .913 ratio between the input data points. The images from the slice between the planes of the points are zero to within 12 orders of magnitude.
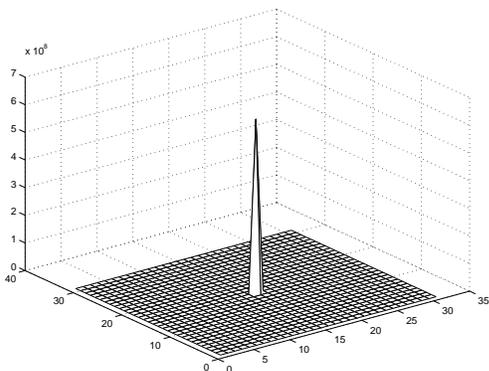
Fig. 1 Z = 0.0 slice, 20th iteration

The second phantom is a cylinder of radius 2.5 cm and
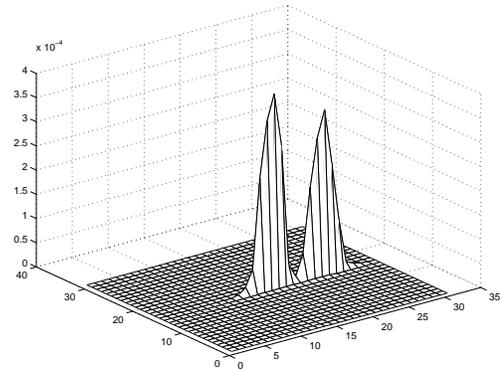
Fig. 2 Z = +.12 slice, 20th iteration
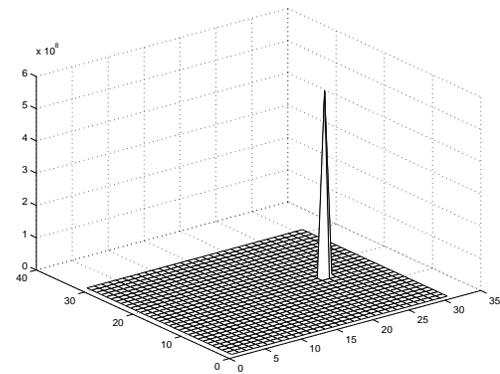
Fig. 3 Z = +.24 slice, 20th iteration

background intensity 1, with two hot spots of relative intensity 10. and two cold spots of intensity 0. The hot and cold spots are centered 1.5 cm from the origin, and have radii of .2 and .4 cm. The height of the cylinder is to 2.5 cm. Monte Carlo simulations were done and reconstructions performed in parallel, using 100,000 particles on each of 2 Unix workstations and a 3x32x32 grid of .2 cm pixels. Results for the central slice are presented for the 10th, 20th, and 40th iterations in figure 4 through 6. Both hot spots are clearly visible, and as the iteration converges, they approach their expected relative size and intensities. The is more clearly demonstrated in figure 7, which is a cross sectional plot through the hot spots. The cold spots are also visible though less discernible. Further, with increasing iterations, because no regularization was used, the level of noise blurs the effect.

IV. CONCLUSIONS

A technique for list-mode EM reconstruction of Compton scatter camera images in 3-D has been developed. The method, which provides order of N saving in memory requirement, has been implemented in parallel using the MPI message-passing interface. Images reconstructed from distributed source phantom data generated by Monte Carlo simulation are
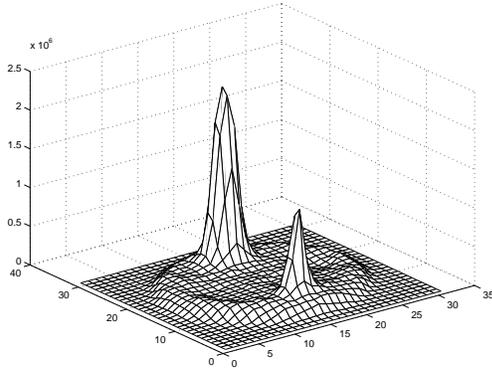
Fig. 4 Central slice, cylindrical phantom 10th iteration
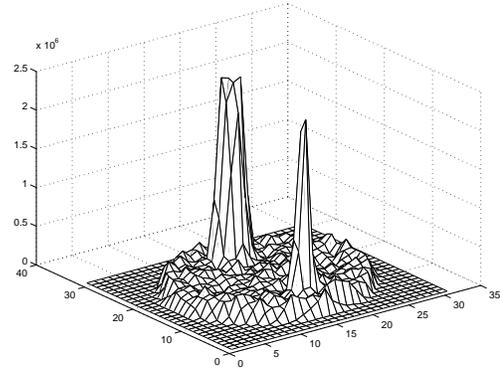
Fig. 6 Central slice, cylindrical phantom 40th iteration
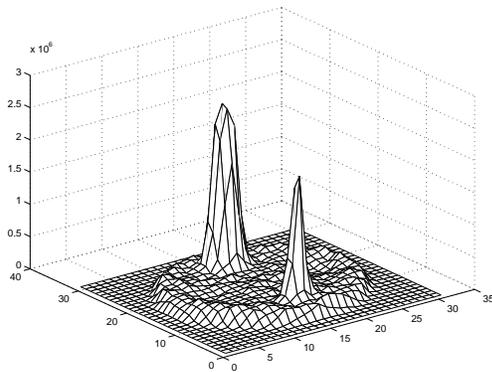
Fig. 5 Central slice, cylindrical phantom 20th iteration
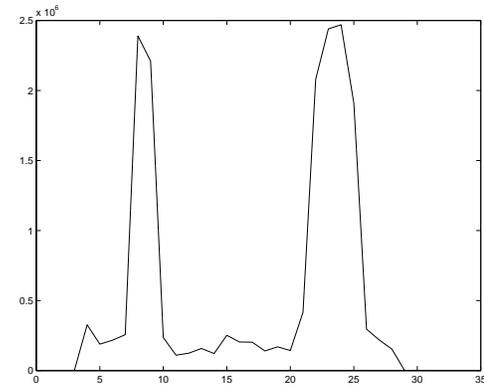
Fig. 7 Cross section through central slice, cylindrical phantom 40th iteration

presented, demonstrating validity of the technique. Using this algorithm, a problem with 100,000 particles to be reconstructed on 64x64x64 image space would still require 5 1 GB nodes in order to fit into memory. Further work is required before the method could be applied to more realistic problems.

V. REFERENCES

[1] T. Hebert, R. Leahy, M. Singh, "Three-dimensional maximum-likelihood reconstruction for an electronically collimated single-photon-emission imaging system," *J. Opt. Soc. Am., vol. 7, 1990* pp. 1305-1313.

[2] H.H. Barrett, T. White, and L.C. Parra, "List-mode likelihood," *J. Opt. Soc. Am., vol. 14, 1997* pp. 2914-2923.

[3] L.C. Parra and H.H. Barrett, "List-mode likelihood: EM algorithm and image quality estimation demonstrated on 2-D PET," *IEEE Trans. Med. Imag., vol. 17, 1998* pp. 228-235.

[4] S.J. Wilderman, J.A. Fessler, N.H. Clinthorne, and W.L. Rogers, "List-mode maximum likelihood reconstruction of Compton scatter camera images in nuclear medicine," *Proceedings of the 1998 IEEE Nuclear Science Symposium*, vol 3, (1998), p1716-1720.

[5] S.J. Wilderman, J.A. Fessler, N.H. Clinthorne, J.W. LeBlanc, and W.L. Rogers, "Improved modeling of system response in list-mode EM reconstruction of Compton scatter camera images,"

submitted to IEEE Transactions on Nuclear Science, 1999.

[6] W. Gropp, E. Lusk, and R. Thakur, *Using MPI-2: Advanced Features of the Message-Passing Interface*, MIT Press, 1999.

[7] T. Kragh, *private communication, Oct 1999.*

[8] J.W. LeBlanc, X. Bai, N.H. Clinthorne, C-h. Hua, D. Meier, W.L. Rogers, D.K. Wehe, P. Weilhammer, S.J. Wilderman, "^{99m}Tc Imaging Performance of the C-SPRINT Compton Camera," presented at *IEEE Nucl. Sci. Sym. and Med. Imaging Conf., Seattle, WA, 1999.*

[9] S.J. Wilderman, "Vectorized algorithms for the Monte Carlo simulation of kilovolt electron and photon transport," *University of Michigan, Ann Arbor, MI, Ph. D. dissertation, 1990.*