Published in final edited form as:

Phys Med Biol. 2013 May 07; 58(9): 3001–3022. doi:10.1088/0031-9155/58/9/3001.

Practical implementation of tetrahedral mesh reconstruction in emission tomography

R. Boutchko¹, A. Sitek², and G. T. Gullberg¹

¹Lawrence Berkeley National Laboratory, 1 Cyclotron Rd., M.S. 55R0121, Berkeley, CA 94720, USA

²Massachusetts General Hospital, White 427, 55 Fruit St., Boston, MA 02114, USA

Abstract

This paper presents a practical implementation of image reconstruction on tetrahedral meshes optimized for emission computed tomography with parallel beam geometry. Tetrahedral mesh built on a point cloud is a convenient image representation method, intrinsically three-dimensional and with a multi-level resolution property. Image intensities are defined at the mesh nodes and linearly interpolated inside each tetrahedron. For the given mesh geometry, the intensities can be computed directly from tomographic projections using iterative reconstruction algorithms with a system matrix calculated using an exact analytical formula. The mesh geometry is optimized for a specific patient using a two stage process. First, a noisy image is reconstructed on a finely-spaced uniform cloud. Then, the geometry of the representation is adaptively transformed through boundary-preserving node motion and elimination. Nodes are removed in constant intensity regions, merged along the boundaries, and moved in the direction of the mean local intensity gradient in order to provide higher node density in the boundary regions. Attenuation correction and detector geometric response are included in the system matrix. Once the mesh geometry is optimized, it is used to generate the final system matrix for ML-EM reconstruction of node intensities and for visualization of the reconstructed images. In dynamic PET or SPECT imaging, the system matrix generation procedure is performed using a quasi-static sinogram, generated by summing projection data from multiple time frames. This system matrix is then used to reconstruct the individual time frame projections. Performance of the new method is evaluated by reconstructing simulated projections of the NCAT phantom and the method is then applied to dynamic SPECT phantom and patient studies and to a dynamic microPET rat study. Tetrahedral mesh-based images are compared to the standard voxel-based reconstruction for both high and low signal-to-noise ratio projection datasets. The results demonstrate that the reconstructed images represented as tetrahedral meshes based on point clouds offer image quality comparable to that achievable using a standard voxel grid while allowing substantial reduction in the number of unknown intensities to be reconstructed and reducing the noise.

1. Introduction

A significant portion of modern nuclear emission tomography research is dedicated to improving quality of the reconstructed single photon emission computed tomography (SPECT) and positron emission tomography (PET) images. One method of improving image quality is finding new methods of image representation optimized for specific imaging tasks.

Non-uniform point clouds organized into a tetrahedral mesh have been shown to be a convenient and sparse method of representing SPECT, PET and x-ray computed tomography (CT) images. Two-dimensional triangular meshes in tomography were introduced by Brankov and his coworkers in (Yang *et al* 2003, Brankov *et al* 2004). Direct application of 3D meshes in emission tomography was proposed by Sitek in (Sitek *et al* 2006) and further developed in (Pereira and Sitek 2010). Our paper describes a development of this method, optimized for dynamic SPECT and PET imaging.

The goal of dynamic SPECT is to obtain information about the evolution of radiotracer distribution in organs and tissues with time (Gullberg *et al* 2010). In order to achieve this goal, multiple projection datasets (sinograms) are acquired consecutively. Acquisition time for each sinogram is considerably shorter than that in standard static SPECT. This translates into low signal-to-noise ratio (SNR) in the sinogram and poor quality of the reconstructed images. The effective projection time can be increased by summing up the individual time frame sinograms, which can result in a potential SNR increase. Images reconstructed from this quasi- static sinogram provide prior information that allows one to create a tetrahedral mesh-based image representation optimized for the specific geometry of the study. The optimized representation is then used to reconstruct individuals sinogram frames.

This manuscript is organized as follows. In Section 2, we list the main properties of tetrahedral meshes, provide a detailed description of each step of image reconstruction on a tetrahedral mesh, and give the basic algorithm to be used in our work. Section 3 describes the imaging experiments conducted in order to illustrate our method using either simulated or acquired projection data. The results of these experiments are shown in Section 4. Sections 5 and 6 provide discussion of the method applicability to different types of studies and the summary of our work.

2. Methods

The presentation of tomographic reconstruction on point-cloud-based tetrahedral meshes is organized in four stages. First, we briefly describe basic facts about tetrahedral meshes and their application in tomography. Then we explain our method of mesh generation and adaptive optimization of the mesh geometry. Next, we describe our implementation of supplementary imaging techniques such as attenuation and other corrections and data visualization. Finally, we lay out the step-by-step algorithm for applying our method in an imaging study.

2.1. Point cloud and tetrahedral meshes in tomography

This subsection explains the notations and definitions used in our manuscript and lists the main results from (Sitek *et al* 2006), where tetrahedral mesh application in 3D tomography was introduced.

2.1.1. Image representation—Our goal is to represent a compactly supported distribution $f(\mathbf{r})$ with a finite number of parameters. We achieve this by using a *tetrahedral mesh* defined on a *point cloud*. The cloud, a finite set of nodes with positions $\mathbf{r}_k$, is denoted as $\mathcal{R}_K$, where K is the total number of nodes. These nodes form a mesh of M non-

overlapping tetrahedra denoted by $\mathcal{T}_M$. Each tetrahedron is defined by the indices of its four vertices

$$T_M \equiv \{\mathbf{r}_{m1}, \mathbf{r}_{m2}, \mathbf{r}_{m3}, \mathbf{r}_{m4}\} \subset \mathcal{R}_K, \quad (1)$$

ordered so that

$$V_m = \frac{[(\mathbf{r}_{m2} - \mathbf{r}_{m1}) \times (\mathbf{r}_{m3} - \mathbf{r}_{m1})] \cdot (\mathbf{r}_{m4} - \mathbf{r}_{m1})}{12} > 0. \quad (2)$$

Expression (2) is equal to the volume of the m^{th} tetrahedron. Two sets, $\mathcal{R}_K$ and $\mathcal{T}_M$, define the point cloud with tetrahedral mesh, which we denote by $\mathcal{P}_{K,M}$. The mesh geometry is constructed using an adaptive optimization process described in the next subsection.

An intensity value I_k is associated with each node in $\mathcal{R}_K$; the complete set of intensities is denoted as $\mathcal{I}_K$. For each tetrahedron T_m , there exists a linear function $f_m(\mathbf{r})$ that interpolates intensities I_k so that $f_m(\mathbf{r}_k) = I_k$. This existence is subject to the condition (2). A union of all $f_m(\mathbf{r})$ defines a continuous piecewise-linear three-dimensional function $f_{\mathcal{P},I}(\mathbf{r})$. With the right choice of node intensities, this function constitutes a representation of a 3D distribution $f(\mathbf{r})$. The intensities are chosen to satisfy the L^2 difference norm of the approximation:

$$\mathcal{I}_K = \arg \min_I \int_{\mathbb{R}^3} (f(\mathbf{r}) - f_{\mathcal{P},I}(\mathbf{r}))^2 d\mathbf{r} \quad (3)$$

The closeness of the representation to the original distribution depends on the properties of the represented function and on the cloud geometry. Often, good representation is achieved by straightforward sampling of f at the mesh nodes $I_k = f(\mathbf{r}_k)$.

2.1.2. Projection of a tetrahedron—Tetrahedral representation allows a convenient definition of discrete linear projections as encountered in tomography. Expressed in Cartesian coordinates with the detector in xy -plane, one projection bin is defined as

$$P_j = \int_{j\text{-th pixel}} dx dy \int_{-\infty}^{\infty} dz f(\mathbf{r}). \quad (4)$$

This equation describes an integral of f within an infinite prism with the cross-section of one detector bin, generally, a square. If f is represented by $f_{\mathcal{P},I}$, each tetrahedron can be projected separately. An intersection of a tetrahedron and an infinite prism is shaped like an irregular polyhedron of one of several types with two examples shown in Figure 1. Inside each type of these polyhedra, the integral of a linear function can be represented as an algebraic expression. For example, if the m^{th} tetrahedron is completely enclosed within the projection prism, then the projection is simply

$$P_{j,m} = \int_{T_m} f_m(\mathbf{r}) d\mathbf{r} = V_m \frac{I_{m_1} + I_{m_2} + I_{m_3} I_{m_4}}{4}. \quad (5)$$

For other possible intersection geometries, the resulting projection integrals are more cumbersome, although still relatively straightforward algebraic expressions. Detailed discussion of possible geometries and methods to facilitate their calculation by using GPU computation is provided in (Sitek *et al* 2006).

2.1.3. 1D and 2D cases—A tetrahedron is a 3D convex hull of its four vertices or a 3D *simplex*. Many features of tetrahedral meshes are preserved in simplex meshes in lower dimensions, where they are easier to visualize and understand. Below, we give a short description of the 1D and 2D analogues of the tetrahedral mesh. These analogues will be used throughout our work for illustration purposes or to gain additional insight for the tetrahedral mesh construction and optimization.

A one-dimensional simplex is a straight line segment. A simplex mesh representation of a function $f(x)$ is a piecewise linear polynomial with irregularly distributed nodes as shown in Figure 2. Unlike in the higher-dimensional cases, there is no ambiguity in the mesh construction, since the “chords” of the mesh always connect consecutive nodes.

A 2D simplex is a triangle. A triangular mesh is built on an irregular point cloud in a non-unique way using the process of triangulation. Delaunay triangulation (Berg *et al* 2008) is a popular technique that produces convenient meshes for most non-degenerate point cloud geometries. In Figure 3, we show the Delaunay triangulation of a small point cloud produced by Matlab® and the corresponding representation of an arbitrary 2D distribution that uses the sampled values of the approximated functions as the node intensities $I_k = f(\mathbf{r}_k)$. Since triangular and tetrahedral meshes are similar, we use 2D analogues to illustrate some of the optimization operations in the following sections.

2.1.4. Tomographic reconstruction—If the cloud and mesh geometry $\mathcal{P}_{KM}$ is known, we can define the *system matrix*, which allows us to define the *forward projection* as a linear system that relates the node intensities and the projections:

$$\tilde{P}_m = \sum_K S_{mk} I_k + \text{Noise} \quad (6)$$

or, for the radionuclide emission tomography

$$\tilde{P}_m = \text{Poisson} \left(\sum_K S_{mk} I_k \right). \quad (7)$$

The k^{th} column of the system matrix can be viewed as the total set of projections (5) of a cloud with the intensity $I_k = 1$ and all other intensities $I_n \neq k = 0$.

The *tomographic problem* is an inverse problem to (7), used to determine the $\mathcal{Q}_K$ from a set of measured projections $\{\tilde{P}_M\}$. (Only vertex intensities $\mathcal{Q}_K$ are computed during image reconstruction; intensities inside each tetrahedron are uniquely determined by the vertex values.) A number of techniques can be applied to reconstruct the underlying image from projections (i.e., to solve the inverse problem). For the strong signal/low noise scenario, filtered back projection (FBP) has been successfully applied in 2D case by Brankov and colleagues (Yang *et al* 2003, Brankov *et al* 2004, Brankov *et al* 2005, Gonzalo and Brankov 2008) and in 3D, both gradient search and expectation maximization algorithms have been applied by Sitek and colleagues (Sitek *et al* 2006, Pereira and Sitek 2010). We use maximum likelihood expectation maximization (ML-EM) method for all image intensity reconstruction tasks. FBP, gradient search, and different versions of EM are all standard image reconstruction methods well described in the literature (Wernick and Aarsvold 2004).

2.2. Mesh optimization

This subsection describes the techniques used for generation and adaptive optimization of tetrahedral meshes based on point clouds in SPECT and PET imaging. In order to develop a methodology that is specific for the radionuclide emission tomography, we make an important assumption about the nature of 3D images to be reconstructed. These images are assumed to consist of regions of constant or near-constant intensity. Thin boundaries between these regions are well defined and characterized by a high absolute value of the image intensity gradient $|\nabla f|$. The constant intensity parts of an image correspond to tissues within the same internal organs, where the radiotracer concentration does not vary significantly except in regions with decreased or increased tracer uptake. Since detecting such regions is often the main task of the scanning procedure, it is important that the voxel size in those regions is sufficiently small to resolve high-spatial-frequency features. We ensure high node density in selected regions by using a mesh generation method that relies on coarsening of an initially dense mesh instead of refinement of an initially- sparse mesh (Sitek *et al* 2006), thus lowering the chances of missing relatively small regions where increased resolution is required. Here and elsewhere in this article, by “resolution” we mean the image resolution, e.g., voxels per unit volume, not the resolution of the imaging system.

2.2.1. Mesh coarsening—Two strategies of reducing the number of nodes in the mesh (and therefore, the unknown intensities to be reconstructed) are used: removing center-nodes in constant-intensity regions or merging a pair of nearby nodes that have the same intensity. In both cases, the mesh is analyzed to compute two supplementary sets of variables for each node k , R_k^N and T_k^N . R_k^N is a set of all neighbors of node k . Since two nodes, connected by a single chord of the mesh, are guaranteed to be the forming vertices of at least one tetrahedron, we define this set as:

$$\mathcal{R}_K^N: \forall \mathbf{r}_j \in \mathcal{R}_K^N \exists T_m: \{\mathbf{r}_k, \mathbf{r}_j\} \subset T_m. \quad (8)$$

The second is a set of all tetrahedra that include the node k :

$$\mathcal{T}_k^N: \forall T_m \in \mathcal{T}_k, \mathbf{r}_k \in T_m. \quad (9)$$

If the node k is in the middle of a constant-intensity region, i.e., all its immediate neighbors have the same intensities, this node can be removed and the mesh connectivity adjusted without changing the intensity distribution f . More rigorously, if for a small threshold ϵ_1 ,

$$\frac{|I_j - I_k|}{\min(I_k, I_j)} < \epsilon_1 \quad \forall \mathbf{r}_j \in \mathcal{R}_k^N, \quad (10)$$

then the node k can be removed from the mesh with the total norm of the approximation changing by less than $I_k \epsilon_1 \left(\sum_{m \in T_k^N} V_m \right) / 4$. The region formed by all tetrahedra in T_k is a polyhedron with triangular facets. When the k^{th} node is removed, this polyhedron is re-meshed by moving the node k to the position of one of its neighbors j . Tetrahedra that included both nodes k and j are discarded from the original mesh. In order to preserve the shape of the reconstructed region, no node from either side of the region (typically, a cube) is moved inside the region, no node from a rib of this tetrahedron is moved either inside or onto a facet, and no nodes from a 3D corner of the reconstructed object are removed at all. A simple symmetry check is performed to ensure the integrity of the new mesh: we require that as k is replaced by j , volumes (2) of all tetrahedra in the mesh remain positive. If several neighbors satisfy symmetry check as a candidates for the remeshing target, we pick a neighbor that maximizes the volume of the smallest of the new tetrahedra created by remeshing. This condition helps to sustain non-degenerate mesh structure.

The second node-removal method can be applied in regions with changing intensities, including boundary regions. If two neighbors are sufficiently close and have close intensities, they are merged to a single node positioned half-way between the original close neighbors. This is defined in terms of two thresholds. If for two neighbors k and j

$$\frac{|I_j - I_k|}{\min(I_k, I_j)} < \epsilon_2 \quad \text{and} \quad \left| \mathbf{r}_k - \mathbf{r}_j \right| < d, \quad (11)$$

then one of the nodes is discarded (it is irrelevant, which one, let us assume it is the node j), two tetrahedra that included both j and k are removed from the mesh, and the position and intensity of the remaining node are adjusted

$$\mathbf{r}_k \rightarrow \frac{\mathbf{r}_k + \mathbf{r}_j}{2}, I_k \rightarrow \frac{I_k + I_j}{2}. \quad (12)$$

Parameter ϵ_2 is typically small, and parameter d is commensurate with the detector pixel dimension. The same mesh integrity constraints as described above is applied, plus both merged nodes have to be either inside the reconstructed volume, on the same facet, or on the same rib. Examples of both mesh-coarsening operations are given in Figure 4 using the same 2D example as previously shown in Figure 3

2.2.2. Node motion—In addition to the node elimination, the optimization procedure includes adaptive node motion with the intention to better outline the boundaries in the image while preserving the mesh structure. We choose the direction of the motion so that the nodes move towards the assumed location of the boundary. Since the true image is not known *a priori*, this direction is chosen for each node based on the intensities and positions of the moved node and its immediate neighbors. In order to define a method of establishing the motion direction, we will consider an 1D example. The k^{th} node of the 1D mesh characterized by its position x_k and intensity I_k “knows” positions and intensities of its two neighbors, $x_{k\pm 1}$ and $I_{k\pm 1}$. The gradients (in this particular case, derivatives) on both sides of the k^{th} node are $f'_{\pm} = (I_{k\pm 1} - I_k)/(x_{k\pm 1} - x_k)$. The next step depends on the signs of $f'_{\pm}$. Suppose, both derivatives are positive. In this case, if the boundary is on the left side of node k , then $f'_- > f'_+$, while if the boundary is to its right side, then $f'_- < f'_+$. The latter case is shown in a simple illustration in Figure 5(a). In that case, moving the node to the right creates a better local representation of $f(x)$. The rule is reversed if both $f'_{\pm} < 0$. If the two derivatives have different signs or the same sign but also the same magnitude, then no motion is applied to the node of interest.

In the 2D and 3D cases, we use the same approach as in the example above, except it has to be preceded by determining the local gradient unit vector

$$\hat{\eta} = \frac{\nabla f}{|\nabla f|}, \quad (13)$$

following an obvious assumption that near a boundary the gradient is perpendicular to this boundary. Inside each tetrahedron the function $f_P(\mathbf{r})$ is linear, therefore its gradient is uniform. We denote ∇f inside the m^{th} tetrahedron as ∇_m and compute the local gradient as

$$\nabla f = \sum_{m \in \mathcal{T}_k} w_{m,k} \nabla_m \quad (14)$$

Here, the weighting factors $w_{m,k}$ are related to $\Omega_{m,k}$, the solid angle inside the m^{th} tetrahedron at the vertex $\mathbf{r}_k$,

$$w_{m,k} = \frac{\Omega_{m,k}}{\sum_{m' \in \mathcal{T}_k} \Omega_{m',k}}. \quad (15)$$

Solid angles $\Omega_{m,k}$ are computed from the node positions using a formula derived in (Oosterom *et al* 1983). When $\tilde{\eta}$ is known, we have to determine whether the motion has to be in $+\hat{\eta}$ or $-\hat{\eta}$ direction. In direct analogy with the 1D case, we compute “upstream” and “downstream” gradients at the k^{th} node:

$$\nabla_{\pm} = \sum_{m \in \mathcal{T}_k} w_{m,k} (\nabla_m \cdot \eta) \times \begin{cases} 1, & (\nabla_m \cdot \eta) \geq 0, \\ 0, & \text{otherwies.} \end{cases} \quad (16)$$

In terms of these quantities, we compute the desired direction displacement of the k^{th} node as

$$\hat{\delta}\mathbf{r}_k = \begin{cases} \eta & \nabla_+ > |\nabla f| > \nabla_-, \\ -\eta & \nabla_+ < |\nabla f| < \nabla_-, \\ 0, & \text{otherwies.} \end{cases} \quad (17)$$

The magnitude of the shift $\delta\mathbf{r}_k = \alpha \hat{\delta}\mathbf{r}_k$ is selected as a fraction of the smallest nearest neighbor distance $|\mathbf{r}_j - \mathbf{r}_k|$ for $\mathbf{r}_j \in \mathcal{R}_k^N$. Initially, we multiply distance by a semi-arbitrary value $\alpha = 0.1|\eta_+ - \eta_-|/|\nabla f|$, although other starting values of α can be chosen. If following the motion $\mathbf{r}_k \rightarrow \mathbf{r}_k + \delta\mathbf{r}_k$, the mesh integrity conditions (2) are not satisfied for all $T_m \in T_k$, then α is changed to $\alpha/2$ and the motion attempt is repeated.

The proposed node position adjustment algorithm is based on 1D analogies and general considerations. Although at this time we do not have a rigorous proof of the efficiency of this approach, results of our computational experiments provide a strong numerical evidence that this method does allow one to improve the mesh representation of a typical SPECT or PET image. Results of our method application in a 2D case are shown in Figure 5(b). The illustration shows triangular mesh representation of a well-known Shepp-Logan phantom. A 128×128 fraction of the 512×512 pixel phantom was generated. A 32×32 square grid of nodes was created and used to generate a mesh of 1,922 triangles. The node intensities were sampled from the original phantom. The number of nodes was then reduced to 236 and the number of triangles to 441 using the algorithm described in subsection 2.2.1. Several iterations of node motion were performed. After each iteration, the node intensities were computed from (3) using the known true phantom values. The work was performed using Matlab routines phantom for phantom generation, delaunay for triangulation, tri2cart for mapping triangular mesh data onto a pixel grid, and fminsearch and fmincon to perform the minimization (3).

2.3. Corrections

2.3.1. Attenuation correction—In SPECT imaging, attenuation correction is often required. An attenuation coefficient map μ is obtained from an x-ray CT sinogram of the patient in the format of a voxel-based matrix. When needed, the attenuation correction is computed by weighting the system matrix element S_{jk} by $\exp(\int \mu ds)$ where the integration

line connects the mesh node k to the detector bin j . This integral is computed from the attenuation matrix using straightforward ray tracing.

2.3.2. Diverging ray correction—Even though the ideal model of parallel beam SPECT assumes that only photons orthogonal to the detector plane are counted, in reality, a certain fraction of non-orthogonal rays is also detected. In ray-driven voxel-based reconstruction, the presence of divergent rays is taken into account by adding appropriately weighted line integrals. We would like to preserve the convenience of calculating the system matrix elements as described in paragraph 2.1.2. This is achieved using the following strategy. After the projection of the point cloud with one non-zero intensity has been computed, as discussed above, the detector matrix is convolved with a point-spread kernel calculated from the value of the distance from k^{th} node to the detector plane. The convolution kernel used to reconstruct the data acquired using our SPECT camera is a Gaussian and parametrized as described in Appendix A.

2.4. Point cloud visualization

To visualize a point cloud/tetrahedral mesh, we project it onto a regular voxel-based grid. In order to minimize the resampling approximation error, this process is done in a manner similar to calculating the parallel projections in (4). The intensity inside the tetrahedron T_m varies linearly as $f_m(\mathbf{r})$, intensity inside the j^{th} pixel is uniform and equal to p_j . If we define the spatial domain of the tetrahedron by $\{T_m\}$, the domain of the pixel by $\{p_j\}$, and their intersection as $\Omega_{mj} = \{T_m\} \cup \{p_j\}$, then the pixel intensities are computed as

$$p_j = \sum_m \int_{\Omega_{mj}} f_m(\mathbf{r}) \cdot \quad (18)$$

In order to compute (18), we use the same logic as in computing the projections of tetrahedra. As shown in Figure 6, the intersection of any tetrahedron with a voxel layer (a horizontal slab in the Figure) can be represented as a combination of several tetrahedra. Integrating these tetrahedra then is equivalent to projecting them onto a virtual detector using equation (4).

Equation (18) can be used to visualize any slice that intersects the point cloud. By varying the pixel size, the desired visualization resolution can be achieved. We see this approach as advantageous to simple sampling of the nodes at a regular grid because it preserves the L_1 norm of the image and better represents the fine features of a multiresolution grid.

2.5. Algorithm

The exact algorithm of mesh generation and optimization in nuclear emission tomography varies little for a static and for a dynamic acquisition. For a static acquisition, a single set of projection data is reconstructed to yield a single (static) 3D image. In a dynamic acquisition, several noisy projection datasets are acquired for several consecutive time frames as the radiotracer redistributes inside the subject and used to reconstruct dynamic images, usually, extremely noisy. A less noisy projection dataset can be formed by summing the projection

data from the later-stage time frames (after most of the tracer redistribution has occurred). Usually, this summed projection is similar in its noise properties to a standard static projection dataset.

The mesh-formation and image reconstruction algorithm for both PET and SPECT consists of the following steps:

- i. A parallel-geometry projection data array $\tilde{P}$ is formed for the static sinogram and, if needed, for each dynamic sinogram. The dimensions of the projection array are $M = N_r \times N_z \times N_{\text{ang}}$, with the subscripts corresponding to the radial, axial, and angular data subdivisions. We assume a symmetric radial field of view (FOV) and identical pixel size in axial and radial directions, which is typically the case for SPECT. The description below can be adjusted to a shifted radial FOV or non-uniform inter-slice and intra-slice resolution in a straightforward manner.
- ii. A regular and uniform $(N_r + 1) \times (N_r + 1) \times (N_z + 1)$ node array is formed by placing the nodes in the corners of the voxel grid. A tetrahedral mesh is formed by subdividing each cubic voxel into five tetrahedra (Sitek *et al* 2005).
- iii. The tetrahedral system matrix S is computed for the current geometry. For SPECT, divergent ray correction, attenuation correction, and other possible corrections are applied as needed. Using the system matrix and the static projection data, the original high-resolution image is reconstructed. If the projections are noisy, post-smoothing is desired; if the ML-EM or other EM algorithm is used, it is apodized after a few (5–10) iterations to ensure smoother images.
- iv. The reconstructed image is examined for consistency with the input data. If this is the original reconstruction, the image intensity values and major features in the FOV are established and used to define the mesh optimization thresholds $\epsilon_{1,2}$ and d . If a higher resolution is desired in one portion of the image (e.g., in the heart region), smaller threshold values can be specified for the higher-resolution portion.
- v. Mesh coarsening and node motion steps are performed. After each optimization step, we verify that the mesh integrity is preserved by checking two geometric conditions. Condition 1 requires that the volumes of all tetrahedra are positive and larger than a small threshold limit, usually $1/100^{\text{th}}$ of the underlying voxel volume. Condition 2 requires that all of the nearest neighbor distances are above a prescribed minimum, usually, $1/10^{\text{th}}$ of the original voxel spacing.
- vi. The above step is repeated while gradually increasing the optimization thresholds $\epsilon_{1,2}$ and d from zero to the pre-set values. Usually, 8–10 repetitions are performed.
- vii. The new system matrix S is computed for the sparser optimized mesh and used in the new ML-EM reconstruction of the node intensities.

- viii. If the resulting images require (or at least allow) further coarsening and motion, the algorithm is returned to step v. Otherwise, the static image reconstruction is completed. We stop mesh-coarsening process when for pre-set thresholds (10) and (11), no new nodes are removed during the last iteration. This is usually achieved after 5–10 repetitions of the process.
- ix. In case of dynamic SPECT or PET, the current system matrix built for the sparse and optimized mesh is used to reconstruct each of the dynamic time frame datasets.

3. Imaging Experiments

Several imaging experiments using both simulated and acquired emission projection data have been conducted to illustrate the performance of the proposed methods.

3.1. Numerical phantom study

Noise-dependent performance of the tetrahedral mesh-based image reconstruction was studied using simulated SPECT projections of the numerical NCAT phantom (Segars 2001, Segars *et al* 2008). In order to generate the noisy projections, we used a $128 \times 128 \times 80$ -voxel section of the NCAT phantom extending from the mid-abdomen to upper thorax (see Figure 7) with the default activity values (55 units for liver and the heart tissue, 2 units for blood and bones, 4 units for most other tissues). Seventy-two 128×80 planar projections were generated in the step-and-shoot mode over a full 360° angular FOV using a voxel-driven parallel X-ray transform. The resolution of the projection was then reduced by a factor of two to yield a $64 \times 40 \times 72$ projection dataset $\{\tilde{P}\}$ (these sampling parameters are similar to those, produced by a typical imaging protocol used in our cardiac SPECT patient exams). This dataset was used as the mean to generate several series of J sets of data with Poisson distributed noise

$$P_n^j(\eta) = \text{Poisson}\left(\frac{\bar{P}_n}{\eta}\right) \quad (19)$$

Here, index $j = 1, 2, \dots, J$ denotes the particular Poisson noise implementation and η denotes the noise level.

All of the projection sets were reconstructed on both a standard $64 \times 64 \times 40$ voxel grid and a point cloud-based tetrahedral mesh using the algorithm described in the previous section. The size of the voxels in the reconstruction grid was the same as the size of the pixels in the simulated projections. Thirty-five iterations of ML-EM algorithm were used for all reconstruction algorithm implementations. Reconstructed data, both voxel and mesh node intensities, were scaled by the noise level:

$$\{V\}^j(\eta) = \eta \times \text{MLEMreconstruction}(\{P\}^j(\eta)), \quad (20)$$

In order to isolate the comparative analysis on the image reconstruction, neither diverging geometry nor attenuation were modeled at both projection generation and image reconstruction stages. No additional rotation or resampling operations were applied: the tetrahedral mesh was mapped onto the standard $64 \times 64 \times 40$ voxel grid, after which the results of the two reconstruction methods were compared. For both of these methods and for each value of η , the statistical mean value of the m^{th} voxel $\bar{V}_m(\eta)$ and the standard deviation $\sigma_m(\eta)$ were computed from the set of J images. The region of interest (ROIs)-averaged SNR was then computed as

$$\text{SNR}_{\text{ROI}}(\eta) = 2\log_{10}\left(\frac{\bar{V}_m(\eta)}{\sigma_m(\eta)}\right)_{\text{ROI}} \quad (21)$$

and the bias values B_{ROI} were computed as

$$B_{\text{ROI}} = \langle \bar{V}_{\text{reconstructed}} \rangle_{\text{ROI}} / \langle V_{\text{trueNCAT}} \rangle_{\text{ROI}}, \quad (22)$$

where $\langle \rangle_{\text{ROI}}$ denotes averaging over an ROI. The original NCAT phantom was used to allocate ROIs corresponding to the heart muscle tissue, blood pool, liver, and also the whole torso including connecting tissues, bones, muscles, and internal organs. For each of these ROIs, we computed the mean SNR for different values of input noise η and different numbers of ML-EM iterations.

3.2. Physical phantom study

SPECT projection datasets of the Jaszczak torso phantom were simulated using the database described in detail in (Sitek *et al* 2006b). The database was created by acquiring multiple projections of the torso phantom separately filling its “organs” with $^{99\text{m}}\text{Tc}$ at different known concentrations and using a dual head SPECT-CT system (GE Millennium VG3). The 3/8” crystal and the Low Energy High Resolution (LEHR) collimator was used. (GE VPC-45K collimator: 54×40 cm field of view; 86 300 hexagonal holes with 1.5 mm diameter, 35 mm length, and 0.2 mm septal width; penetration ratio at 140 KeV is 0.3.) This database allows generation of multiple projection datasets corresponding to the same target distribution of the radiotracer, but containing independent noise realizations. The phantom had four separate portions that could be filled with a radiotracer of different concentrations: myocardium wall, myocardium cavity (“blood”), intestinal region, and the rest of the torso (“background”), as well as tracer-free lungs and spine regions. Each projection of the generated datasets was cropped to 88×48 4.42-mm pixels, containing the complete projection view of the phantom. The number of angular views was 120 for a complete 360° rotation of the gantry in step-and-shoot mode. An attenuation map of the phantom was acquired in the same experimental set-up automatically registered with the emission data by the scanner software. The same attenuation map was used to generate system matrices for both the standard voxel grid and for the tetrahedral mesh.

3.3. Patient imaging study

A dynamic cardiac SPECT patient study was performed at University of California San Francisco (UCSF) hospital using the same model SPECT camera and the same LEHR collimator as described in subsection 3.2. Two detector heads were mounted on a gantry, continuously rotating at 72 seconds per rotation and acquiring one histogrammed dataset every second. Active detector area consisted of 40×64 square 8.84×8.84 mm pixels. A dose of 24 mCi of ^{99m}Tc -sestamibi was injected into the patient following a pharmacologically induced stress achieved by a 4-minute injection of Persantine®. Continuous gantry rotation data acquisition started simultaneously with the radiotracer injection and persisted for 28.8 minutes until 24 full 360° datasets were acquired. The pharmacological stress was reversed 3 minutes into the acquisition by injection of Aminophylline®. A longitudinally truncated $64 \times 20 \times 72$ pixel dataset that contained the heart was used to reconstruct images. The static dataset was formed by summing up the projection data from rotations 2 through 24. The static dataset with a total of 1.6×10^7 counts and one of the dynamic datasets with the total of 7.5×10^5 counts were reconstructed.

The datasets were reconstructed using both a standard voxel approach and tetrahedral mesh approach. Diverging beam and attenuation correction were included in both reconstructions. The ML-EM algorithm apodized after 35 iterations was used in all image reconstruction calculations. The $64 \times 64 \times 20$ 8.84-mm voxel reconstruction was later resampled onto a $32 \times 32 \times 32$ 3.54-mm voxel sub-volume oriented along the major view axes of the heart. The resampling was performed using the Matlab interp3 routine with linear approximation. The tetrahedral mesh reconstruction was projected both on the standard voxel grid and on the higher-spatial resolution grid oriented along the heart long axis using the direct visualization technique described in 2.4.

3.4. MicroPET Rodent Study

A microPET scan of a rat was performed at UCSF Center for Molecular and Functional Imaging using a Siemens Inveon microPET scanner. A dose of 1.24 mCi of ^{18}F FDHROL was injected into the tail vein of the animal at the beginning of the acquisition and imaged for 60 minutes. A single 20-second time frame projection dataset was selected to illustrate our method in this paper. Out of 159 axial slices, 32 slices containing the heart region were selected for reconstruction and processing. The 3.1×10^7 events were binned into a parallel-beam SPECT-like $128 \times 32 \times 160$ projection dataset, with 160 angular views uniformly sampled on 180° . The radial and axial pixel size in the sinogram was 0.796 and 0.815 mm respectively. For the sake of simplicity and better comparison, a uniform 0.8-mm pixel size was assumed for both dimensions; the resulting small deformation of the reconstructed images was ignored.

Similarly to the human patient study, the projection data were reconstructed on both a regular $128 \times 128 \times 32$ voxel grid and a point cloud-based tetrahedral mesh. Thirty-five iterations of the ML-EM algorithm were used both times, with no additional corrections for attenuation or scatter applied. The voxel image was resampled and the tetrahedral result was directly projected onto a $64 \times 64 \times 64$ voxel grid with 0.53 mm spatial resolution rotated along the heart's major axes.

4. Results

4.1. Numerical phantom

$J = 128$ projection datasets of the thorax portion of NCAT phantom were generated for each of six noise levels $\eta = \{1, 4, 16, 64, 256, 1024\}$. The corresponding mean count-rate in the generated projections changed from 230 counts per pixel for $\eta = 1$ (1.1×10^7 total counts) to 0.23 counts per pixel for $\eta = 1024$ (4.2×10^4 total counts). The simulated projection data were reconstructed and analyzed using the method described in subsection 3.1.

The mesh optimization was performed using a projection dataset, created by summing up all 128 datasets for $\eta = 1$. Ten iterations of mesh optimization were performed gradually increasing the intensity thresholds $\epsilon_1 = \epsilon_2$ defined in Equation (10) and Equation (11) from 0.05 to 0.15 in the heart region and from 0.1 to 0.3 elsewhere. A maximum merging distance was gradually increased from 1 to 1.6 pixel widths in the heart region and from 1 to 2.5 elsewhere. The original mesh of $65 \times 65 \times 41 = 173\,225$ nodes and 819\,200 tetrahedra was sparsened to 35\,245 nodes and 185\,441 tetrahedra, a five-fold reduction in node density. The node density reduction was 1.3-fold inside the heart ROI, 3.0-fold elsewhere within the torso, and 14.2-fold outside of the torso-occupied region of the total field of view.

Figure 8 shows the voxel- and tetrahedron-mesh NCAT reconstructions for each value of the input noise η reconstructed using 30 ML-EM iterations. The two techniques differ mostly in the level of the background noise and in the representation of boundaries of constant-intensity regions. As could be argued from the general volume element size considerations, the voxel-based images exhibit more background noise for all η . At low η , the boundaries in the tetrahedral mesh-based images appear to be less pronounced, especially, in the outer regions optimized using higher values of ϵ_1 and ϵ_2 . At medium input noise, the boundaries in the two types of images appear to be similarly well-defined. For large η , the boundaries are masked by noise in the voxel- and by sliver-like artifacts in the tetrahedral-based images. Figure 9 shows the NCAT slices for $\eta = 64$ with the ML-EM reconstruction apodized after different number of iterations $N_{\text{ML-EM}}$. Figure 10 (a) and (b) shows the dependence of mean SNRs for several important ROIs for different η and $N_{\text{ML-EM}}$. The plots confirm the visually observable effect that SNR diminishes as either parameter increases. Figure 10 (c) and (d) shows the values of bias (22) computed for the heart tissue, the blood cavity, and the liver as a function of $N_{\text{ML-EM}}$ for two noise levels. For sufficiently large ROIs and for low input noise levels, both representations achieve satisfactory levels of reconstructed intensities (for perfect reconstruction, $B = 1$ is expected), however, performance of the mesh-based reconstruction appears to be somewhat better. For smaller ROIs for the high input noise case, an optimum number of $N_{\text{ML-EM}}$ appears to be at about 30–50. Finally, Figure 11 shows voxel-by-voxel variances for one of the reconstruction scenarios, illustrating the intuitively expected result of drastic decrease in variance inside constant-intensity regions, where the mean tetrahedron size of the mesh reconstruction is large.

4.2. Physical phantom study

A series of 50 separate complete datasets of the Jaszczak phantom were generated with the ratio of intensity concentrations in the heart muscle compartment to that in background

being approximately 4:1. Blood and liver regions were filled with the same concentration of activity as the background. The mean number of events recorded in the $88 \times 48 \times 120$ datasets was 1.23×10^6 .

A sum of all fifty datasets was used to perform the original reconstruction. Five iterations of the node-optimization resulted in the generation of a mesh of 569 610 tetrahedra based on 110 108 nodes, a 3.4-fold reduction relative to the $88 \times 88 \times 48$ voxel grid representation (2.6-fold reduction in the heart region). Mesh coarsening intensity thresholds $\epsilon_1 = \epsilon_2$ were increased from 0.01 to 0.2 for the regions near the heart and from 0.1 to 0.2 for the rest of the region of reconstruction. A node-merging distance threshold d was set to 2.5 cm at the heart region and 5 cm elsewhere. The resulting mesh was used to reconstruct each of the individual 50 datasets using 8, 32, 128, and 512 iterations of ML-EM. Analogous computations were performed using a standard voxel grid.

Figure 12(a–c) shows the slices of the volumes obtained using the mesh-based and the voxel-based reconstruction algorithms using 8, 32, and 128 iterations. The voxel-based images have significantly higher noise levels, however, the tetrahedral mesh-based images exhibit slight but visible sliver artifacts. The variance in images 12(b) is shown in Figure 12(c). The plot in Figure 12(d) illustrates the dependence of the image SNR on the number of iterations $N_{\text{ML-EM}}$. Despite the obvious SNR advantage in using fewer iterations, some finer details such as the area to the left of the “spine” in the axial slices can be solved only when $N_{\text{ML-EM}} > 30$.

4.3. Patient study

In order to reconstruct the cardiac SPECT study acquired during the experiment described in subsection 3.3, the original tetrahedral mesh with $65 \times 65 \times 21 = 88\,725$ nodes and 409 600 tetrahedra was generated using the static sinogram. Approximately ten iterations of node removal and motion resulted in an optimized mesh with 20 002 nodes and 102 102 tetrahedra. Mesh coarsening intensity thresholds $\epsilon_1 = \epsilon_2$ were increased from 0.01 to 0.2 for the regions near the heart and from 0.1 to 0.4 for the rest of the region of reconstruction. A node-merging distance threshold d was set to 2.5 cm at the heart region and 5 cm elsewhere. This mesh was used to reconstruct the static dataset and the dynamic dataset corresponding to the 21st time frame out of 24.

Axial, coronal, and sagittal slices through the complete region of reconstruction for static and dynamic reconstructions are shown in Figure 13. Following the trend observed in the phantom studies, in static SPECT, voxel-based images exhibit slightly higher background noise yet possibly better-resolved boundaries between regions with different intensity. In the dynamic frame reconstructions, image noise in the voxel case is much higher, while no resolution advantage is observed.

Figure 14 shows the static and dynamic images reconstructed using voxels and tetrahedral meshes, rotated and resampled (for voxels) or directly projected (for tetrahedra) onto the higher-resolution grid aligned with the major heart axes. Visual comparison between Figure 14(a) and (b) shows that the tetrahedron-based reconstruction appears to have sharper, while the voxel-based reconstruction is more visually appealing, probably because of the

additional smoothing effect caused by the process of resampling. At the same time, the lower background noise advantage of the tetrahedral mesh is not as well defined in the heart region where little node density reduction occurred.

4.4. MicroPET rat study

Images reconstructed for a single time frame in the microPET experiment are shown in Figure 15(a) for the voxel-based reconstruction and in Figure 15(b) for the tetrahedral-mesh-based reconstruction. The voxel grid consisted of $128 \times 128 \times 32 = 542\,288$ cubic voxels. The tetrahedral mesh was relaxed from the original 549 153 nodes and 2 711 440 tetrahedra to 40 407 nodes and 225 794 tetrahedra. The coarsening intensity thresholds $\epsilon_1 = \epsilon_2$ were set to 0.02 in the heart region and to 0.1 elsewhere, the node merging threshold was set to, respectively 3 and 6 voxel spacings (2.4 mm and 4.8 mm) near the heart and elsewhere. As in the SPECT experiments discussed before, tetrahedral mesh offers significant reduction of the background noise.

5. Discussion

As seen in the illustrations in the previous section, employing tetrahedral meshes to reconstruct noisy SPECT and PET data results in visible reduction in both the number of unknown intensities to be reconstructed and in the background noise, two facts clearly related to each other. Higher SNR, especially in large constant-intensity regions, is achieved, while no significant loss in pixel resolution is observed. The accuracy of the reconstruction is comparable to that of the voxel-based approach (and possibly better for the extremely high noise levels). In some cases, moderate-level sliver-like artifacts can be detected, but they become significant only in the extremely low SNR cases. However, any statements claiming that tetrahedral mesh approach is better or worse than the conventional voxel-based approach to image reconstruction would be premature at this stage. Systematic studies comparing image quality and performance for the two methods are needed in order to make such claims. In the future, we plan to perform a quantitative study evaluating the image quality of the tetrahedral-mesh based images. Several important issues that need to be addressed during such a study include: analysis of possible image distortions and sliver-like artifacts that can be caused by improper mesh geometries; formalizing the selection of the mesh optimization threshold parameters for different input noise levels and different sizes of the expected features to be recognized; analysis of the convergence of both the mesh optimization process and the reconstruction of node intensities; optimizing the numerical implementation of the mesh-based method so its computational efficiency can be adequately compared with conventional methods; finally, implementing basic smoothing and denoising priors used during image reconstruction. Using a non-uniform tetrahedral mesh is essentially similar to an edge-preserving smoothing prior on a voxel grid, and comparing the performance of the mesh-based reconstruction to a prior-constrained voxel-based reconstruction would be very interesting.

Before some of the important questions outlined above are addressed, we can discuss the main expected advantages and disadvantages of our approach only speculatively. At such level, the main advantages of the method would be the possible SNR gain, the reduction of

the intensity variables to be recovered, and the fact that the images are reconstructed on discrete deformable meshes. Reduced number of unknowns suggests that a point cloud can be a convenient image representation when reconstructing dynamic image data directly from projections. Computational techniques that can benefit from this include factor analysis of dynamic series (Sitek *et al* 2000) or B-spline based reconstruction with non-uniform pixelation previously applied to dynamic small animal SPECT by our group (Reutter *et al* 2011). Implementation of the tomographic problem on a discrete mesh makes our approach a convenient starting point for a number of finite element based methods, including modeling of mechanical and other properties of tissues (Sitek *et al* 2002, Veress *et al* 2008).

Two potential disadvantages of our approach are the possibility of masking of small features during mesh optimization and the increased computational complexity. The loss of feature resolution can potentially be caused by excessive node coarsening that can hide or distort the shapes of boundaries of constant-intensity regions. We plan to minimize the likelihood of this effect by developing the criteria for automatic regulation of the mesh-optimization process. The issue of computational complexity is most noticeable at the pre-reconstruction stage of the algorithm implementation. The number of floating point operations (flops) for generating a system matrix for K voxels and M projection bins is $\sim K \times M$ both for cubic voxels and for a cloud of K points. Each ML-EM iteration consists of one forward- and one backprojection operation, each requiring the number of flops proportional to the number of non-zero entries in the system matrix N_S . As the result, the first three steps of the algorithm described in subsection 2.5 require approximately as much time as the complete image reconstruction process for a standard voxel-based method. Each step of the subsequent mesh optimization requires $\sim C \times K$ operations, where C is proportional to the mean number of nearest neighbors or neighboring tetrahedra for each node. The system matrix generation, image reconstruction, and optimization steps are then repeated several times (5 to 20 times in our experience) with both K and N_S gradually decreasing. Thus, while the computation time for the tetrahedral mesh approach is an order of magnitude greater than that for the cubic voxel approach when reconstructing a single image, this time difference is reduced or inverted when multiple datasets have to be reconstructed using the same system matrix, such as in dynamic imaging.

When compared to previously reported work on tetrahedral mesh-based image reconstruction, an important feature of our approach is that we do not use a purely geometry-based method such as Delaunay tessellation to construct the mesh $\mathcal{T}_M$ for the point cloud geometry $\mathcal{R}_K$. Geometry-driven tessellation methods are likely to create mesh elements, triangles or tetrahedra, that pierce boundaries resulting in wedge-shaped artifacts partially visible in Figure 5. Our adaptive optimization of both the mesh structure and the node distribution allows us to achieve a sparse non-uniform cloud geometry while preserving the intensity boundaries that outline image features of different sizes. Our optimization algorithm is prevented from creating degenerate (zero-volume) mesh elements by enforcing the non-negativity constraint (2) with zero in the right hand side of the inequality substituted by a small positive ϵ . However, even finite volume near-degenerate tetrahedra and triangles may cause computational errors and instabilities, therefore further improvement to the adaptive mesh optimization method is an important part of our future work. In particular, we plan to add another optimization step that would allow us to

“collapse” degenerately elongated or flat tetrahedra and transform them into facets or ribs of the neighboring mesh elements. At this stage, the mesh coarsening parameters (intensity and distance thresholds for different regions) are selected mostly *ad hoc*; systematic methodology is needed for the future applications.

6. Summary

A method of tomographic reconstruction on non-uniform point clouds connected by tetrahedral meshes has been explained in detail with applications for parallel-beam SPECT and PET imaging. The method includes an adaptive mesh optimization algorithm that allows significant reduction in the number of unknown intensities to be reconstructed from projections without reducing the spatial resolution of the reconstructed images. The proposed algorithm includes diverging geometry and attenuation corrections as well as a convenient visualization technique. The method has been successfully applied to reconstruct images both from projections simulated using the digital NCAT phantom and from experimental data, including a human torso phantom study, a dynamic cardiac patient SPECT study and a cardiac microPET study of a rat.

Acknowledgments

The work presented in this article has been funded in part by National Institutes of Health grants R01-HL50663 and R01-EB00121 and by the Director, Office of Science, Office of Biological and Environmental Research of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. The authors would like to thank Dr. Bryan Reutter (formerly LBNL), Dr. Youngho Seo (UCSF) and Mr. Andrew Hernandez (formerly UCSF) for their help in acquiring and preparing the experimental data and Dr. W. Paul Segars, Department of Bioengineering, Duke University for providing the digital NCAT phantom.

Appendix A

Point response in parallel-beam collimator SPECT

This appendix explains the point response correction kernel used to generate the system matrix. Parallel-beam collimation in SPECT does not produce true parallel beams, instead a narrow diverging beam of rays is measured by the detector. Figure A1(a–b) shows a detector view collected from a small sub-millimeter sphere soaked in ^{99m}Tc acquired using our SPECT camera. The point-spread function is a non-uniform circular blob with slightly hexagonal features that reflect the hexagonal cell structure of the collimator. As shown in Figure A1(c). The blob can be fitted by a Gaussian-like function centered at $\mathbf{r}_0$,

$$A \exp(-|\mathbf{r} - \mathbf{r}_0|^2 / \sigma^2).$$

The width of this Gaussian exhibits strong linear dependence on distance d from the point source to the detector. Measuring the point response for d varying between 13 and 60 cm, we arrive to the following approximation for σ :

$$\sigma = ad + b, \quad a = 0.02567, \quad b = 0.21, \quad (\text{A.1})$$

where a is dimensionless and the units for b , d and σ are centimeters.

Figure A1. Projection of a point source on the parallel-hole collimator of GE Millennium VG 3 SPECT camera used to acquire data in section 4. (a) Standard view. (b) Same as (a), but the display window and level adjusted to illustrate hexagonal symmetry of the collimator. (c) Pixel intensities fitted to a Gaussian-like function.

References

- de, Berg M., Cheong, O., van, Kreveld M., Overmars, M. Computational Geometry: Algorithms and Applications. Third Edition. Springer-Verlag Berlin Heidelberg; 2008.
- Brankov JG, Yang Y, Wernick MN. "Tomographic image reconstruction based on content-adaptive mesh model,". *IEEE Trans Med Imaging*. 2004; 23(2):202–212. [PubMed: 14964565]
- Brankov JG, Yang Y, Wernick MN. "Spatio-temporal processing of gated SPECT images using deformable mesh modeling,". *Med Phys*. 2005; 32(9):2839–2849. [PubMed: 16266098]
- Gonzalo R, Brankov JG. "Mesh model 2D reconstruction operator for SPECT,". *Proc. SPIE*. 2008; 6913:69132L.
- Gullberg GT, Reutter BW, Sitek A, Maltz J, Budinger TF. "Dynamic single photon emission computed tomography- Basic principles and cardiac applications,". *Phys. Med. Biol*. 2010; 55:R111–R191. [PubMed: 20858925]
- Oosterom A, Van Strackee J. "The Solid Angle of a Plane Triangle,". *IEEE Trans Bio-Med Eng*. 1983; 30(2):125–126.
- Pereira NF, Sitek A. "Evaluation Of A 3D Point Cloud Tetrahedral Reconstruction Method,". *Phys. Med. Biol*. 2010; 55:5341–5361. [PubMed: 20736496]
- Reutter BW, Huesman RH, Brennan K, Boutchko R, Hanrahan S, Gullberg GT. "Longitudinal Evaluation of Fatty Acid Metabolism in Normal and Spontaneously Hypertensive Rat Hearts with Dynamic MicroSPECT Imaging,". *Int J Molecular Imaging A*. 2011:893129.
- Segars, WP. Chapel Hill, NC: Development of a new dynamic NURBS-based cardiac-torso (NCAT) phantom Ph.D. dissertation, Univ. North Carolina; 2001.
- Segars WP, Mahesh M, Beck TJ, Frey EC, Tsui BW. "Realistic CT simulation using the 4D XCAT phantom,". *Med Phys*. 2008; 35:3800–3808. [PubMed: 18777939]
- Sitek A, Di Bella EVR, Gullberg GT. "Factor analysis with a priori knowledge - application in dynamic cardiac SPECT,". *Phys. Med. Biol*. 2000; 45:2619–2638. [PubMed: 11008961]
- Sitek A, Klein GJ, Gullberg GT, Huesman RH. "Deformable model of the heart with fiber structure,". *IEEE Trans Nucl Sci*. 2002; 49(3):789–793.
- Sitek A, Klein GJ, Reutter BW, Huesman RH, Gullberg GT. "Measurement of the Biomechanics of 3-D Cardiac Function with Gated Nuclear Medicine Studies,". 2005 IEEE Nuclear Science Symposium Conference Record, IEEE NSS-MIC, Puerto Rico, M08-3 (2005). 2005
- Sitek A, Huesman RH, Gullberg GT. "Tomographic reconstruction using an adaptive tetrahedral mesh defined by a point cloud,". *IEEE T Med Imaging*. 2006; 25:1172–1179.

- Sitek A, Reutter BW, Gullberg GT. "Method of Generating Multiple Sets of Experimental Phantom Data,". *it J Nucl Med*. 2006; 47:1187–1192. [PubMed: 16818954]
- Veress AI, Weiss JA, Huesman RH, Reutter BW, Sitek A, Taylor S, Yang Y, Gullberg GT. "Regional changes in the diastolic deformation of the left ventricle for SHR and WKY rats using ^{18}F FDG based microPET technology and hyperelastic warping,". 2008; 36(*Ann Biomed Eng*):1104–1117.
- Wernick M, Aarsvold J. *Emission tomography: the fundamentals of PET and SPECT*, The Fundamentals of PET and SPECT Series (Elsevier Academic Press). 2004
- Yang Y, Wernick MN, Brankov JG. "A fast algorithm for accurate content-adaptive mesh generation,". *IEEE Trans Image Process*. 2003; 12(8):866–881. [PubMed: 18237961]

Figure 1.

Two particular cases of computing the projection of a single tetrahedron onto one detector pixel. (a) A large tetrahedron is projected onto multiple pixels. The polyhedron formed by the intersection of the projected tetrahedron and semi-infinite prism defined by the detector bin is computed, subdivided into individual tetrahedra, and projected separately using scenario (b), when the projected tetrahedron is fully enclosed in the projection-defining prism and the projection integral is given by (5).

Figure 2.
One-dimensional analogue of a tetrahedral mesh built on a point cloud: piecewise linear polynomial with irregularly placed nodes.

Figure 3.

Two-dimensional analogue of a tetrahedral mesh built on a point cloud: triangular-mesh. (a) Original function $f(\mathbf{r})$ and an arbitrary node distribution $\mathcal{R}_K$. (b) Piecewise-bilinear interpolation on a triangular mesh $\mathcal{T}_M$ formed on $\mathcal{R}_K$ using Delaunay triangulation method. Node intensities are assigned by direct sampling of the original function $I_k = f(\mathbf{r}_k)$.

Figure 4.

Mesh optimization example: coarsening step illustrated on a 2D mesh. (Left) The original mesh, with arrows pointing out at the locations where nodes are to be removed. (Right) The same remeshing constraints as described above are applied to perform one node elimination and one node merging. A node in the center of a constant-intensity region in the bottom-right corner is eliminated. Two neighboring nodes of the same intensity in the center are merged. No visible change in the mesh-represented image occurs.

Figure 5.

(a) Node motion, 1D illustration. Moving a node in the direction of a larger absolute value of the gradient allows to better outline the boundary. The same concept of moving nodes toward a likely boundary is applied in 2D and 3D. (b) Node motion, 2D illustration. *Left:* A 128×128 pixel fragment of modified Shepp-Logan phantom. *Center:* Representation of the phantom fragment on a 1,922-triangle mesh built on a sparse 32×32 node grid. The grid was further coarsened to 236-node/441-triangle geometry using the node elimination methodology from subsection 2.2, with no visible changes in the approximated image. *Right:* The 236-node mesh after the motion step. At each step, the node intensities are calculated iteratively using the criterion (3).

Figure 6.

Projecting a tetrahedron onto a voxel grid: a tetrahedron intersection with a single voxel layer is a triangular prism or a tetrahedron or a combination thereof. This intersection can be subdivided into several individual tetrahedra. E.g., the tetrahedra on the right comprise the part of the original tetrahedron fully contained within a single voxel layer. These tetrahedra are then projected onto this layer using the same approach as described in Figure 1.

Figure 7. NCAT phantom, 64×40 slices through the spatially trimmed $64 \times 40 \times 40$ volume. Dashed lines denote relative positions of the orthogonal slices. From left to right, the image displays three axial slices $z = \{11, 15, 19\}$ and one coronal slice $y = 18$. Relative intensity units are used here and in all subsequently shown images.

Figure 8.

The same NCAT phantom slices as in Figure 7 reconstructed using the 35 245-node tetrahedral mesh grid (top row of each sub-figure) and the 163 840-voxel grid (bottom rows) for different noise levels η as explained in equations (19) and (20). The noise levels in (a–f) are: 1, 4, 16, 64, 256, and 1024. The original 64×40 resolution is retained in order to avoid scaling-caused smoothing effects.

Figure 9.

NCAT phantom slices reconstructed using different number of iterations $N_{\text{ML-EM}}$ using mesh-based reconstruction (top rows) and voxel-based reconstruction (bottom rows). Noise level $\eta = 64$. The values of $N_{\text{ML-EM}}$ are: (a) 5, (b) 15, (c) 40, (d) 100. (f) Intensity profiles for the 15-iteration reconstruction.

Figure 10.

(a) SNR vs noise level for the $N_{\text{ML-EM}} = 30$ iteration reconstruction measured in four ROIs: myocardium tissue (heart), blood, liver, and complete torso. (b) SNR vs No. of iterations for $\eta = 4$ and $\eta = 1024$, heart and torso ROIs. (c) Biases for constant intensity regions of heart tissue, blood, and liver for $\eta = 4$ for different $N_{\text{ML-em}}$. The biases were computed using (22). Boundary regions were excluded from the ROIs in order to avoid partial volume effects. (d) Same as the previous plot (including the same legend) for noise level $\eta = 1024$

Figure 11.
Variances for $\eta = 256$, 50 iterations for mesh-based (top) and voxel-based (bottom) reconstruction.

Figure 12.

(a) Jaszczak phantom reconstruction, three slices through the $88 \times 88 \times 48$ volume, 8 ML-EM iterations, 4.42 mm pixels, *top*: mesh-based, *bottom*: voxel-based. (b) Same as previous, 32 iterations. (c) Same as previous, 128 iterations. (d) Variances for the 32-iteration reconstruction. (e) SNR as the function of $N_{\text{ML-EM}}$ for three ROIs. *Note 1*: The maximum displayed intensity for the variance images (d) is $1/10^{\text{th}}$ of that of $I_{\max}$ in the images (a–c). *Note 2*: The green arrows in images (a–c) point at the feature where the intensities are not reconstructed correctly after fewer iterations.

Figure 13.

Patient study, complete region of reconstruction slices, 8.84 mm pixel size: (a) Voxel reconstruction, quasi-static projection dataset. (b) Tetrahedral mesh reconstruction, static dataset. (c) Voxel reconstruction for a single time frame. (d) Tetrahedral mesh reconstruction for a single time frame.

Figure 14.

Patient study results. (a) Static projections, voxel-based reconstruction resampled onto a rotated grid. (b) Static projections, tetrahedron reconstruction rotated and projected onto a voxel grid. (c) Dynamic projections, resampled voxel reconstruction. (d) Dynamic projections, rotated tetrahedron reconstruction.

Figure 15.

Dynamic time frame microPET images of a rat's heart, rotated and projected onto a 0.53 mm grid: (a) Voxel-grid based reconstruction. (b) Tetrahedral mesh-based reconstruction.