

Tomographic Reconstruction Using an Adaptive Tetrahedral Mesh Defined by a Point Cloud

Arkadiusz Sitek*, Member, IEEE, Ronald H. Huesman, Fellow, IEEE, and Grant T. Gullberg, Fellow, IEEE

Abstract—Medical images in nuclear medicine are commonly represented in three dimensions as a stack of two-dimensional images that are reconstructed from tomographic projections. Although natural and straightforward, this may not be an optimal visual representation for performing various diagnostic tasks. A method for three-dimensional (3-D) tomographic reconstruction is developed using a point cloud image representation. A point cloud is a set of points (nodes) in space, where each node of the point cloud is characterized by its position and intensity. The density of the nodes determines the local resolution allowing for the modeling of different parts of the image with different resolution. The reconstructed volume, which in general could be of any resolution, size, shape, and topology, is represented by a set of nonoverlapping tetrahedra defined by the nodes. The intensity at any point within the volume is defined by linearly interpolating inside a tetrahedron from the values at the four nodes that define the tetrahedron. This approach creates a continuous piecewise linear intensity over the reconstruction domain. The reconstruction provides a distinct multiresolution representation, which is designed to accurately and efficiently represent the 3-D image. The method is applicable to the acquisition of any tomographic geometry, such as parallel-, fan-, and cone-beam; and the reconstruction procedure can also model the physics of the image detection process. An efficient method for evaluating the system projection matrix is presented. The system matrix is used in an iterative algorithm to reconstruct both the intensity and location of the distribution of points in the point cloud. Examples of the reconstruction of projection data generated by computer simulations and projection data experimentally acquired using a Jaszczak cardiac torso phantom are presented. This work creates a framework for voxel-less multiresolution representation of images in nuclear medicine.

Index Terms—Iterative reconstruction, positron emission tomography (PET), single photon emission computed tomography (SPECT), tomography.

I. INTRODUCTION

TODAY, diagnostic radiology depends heavily upon digital representation of images in three dimensions to form diagnostic decisions. This has been due to the advancement in computing hardware and digital storage capability that has been developed in parallel with new and sophisticated data acquisition hardware that provides a tremendous amount of digital data for displaying medical images in various types of image format.

Manuscript received March 10, 2006; revised May 1, 2006. This work was supported in part by the National Institutes of Health under Grant R01-HL50663, Grant R01-EB00121, and Grant R01-HL71253, and in part by the U.S. Department of Energy under Contract DE-AC02-05CH11231. Asterisk indicates corresponding author.

*A. Sitek is with the Lawrence Berkeley National Laboratory, Berkeley, CA 94720 USA (e-mail: asitek@lbl.gov).

R. H. Huesman and G. T. Gullberg are with the Lawrence Berkeley National Laboratory, Berkeley, CA 94720 USA.

Digital Object Identifier 10.1109/TMI.2006.879319

Recently, significant effort has been spent in developing even more accurate and efficient ways to display, reconstruct, and analyze medical images. This paper presents a new voxel-less method for representation of reconstructed images and outlines approaches that can be used for three-dimensional (3-D) tomographic reconstruction of nuclear medicine images using point clouds and tetrahedral grids. The motivation for this work is the hypothesis that the new point cloud image representation presented in this paper is superior in performing diagnostic tasks to standard voxel-based image representation. Furthermore, the tetrahedral geometry proposed here may be able to take advantage of the graphics hardware used for visualization of the reconstructed images.

A standard way of representing 3-D images formed from the reconstruction of tomographic projections is to divide the reconstruction volume into a regular array of small voxels and assume that the reconstructed medium is constant inside each voxel. Such methods can be described as using constant basis functions on a regular grid [1]. This approach can be improved by introducing nonconstant basis functions such as blobs [2], [3]. Blobs are basis functions that are defined on a bounded region that extends over several voxels. Linear integrals through the blobs can be computed analytically, thereby increasing performance of the image reconstruction process. It has been shown that this approach increases accuracy compared to the standard constant voxel-based approach to image representation [4]. The irregular image representation in two dimensions was studied by Brankov *et al.* [5] where triangular pixels and a finite element formulation to efficiently reconstruct a series of two-dimensional (2-D) transverse slices in emission tomography were used. A maximum likelihood-expectation maximization (ML-EM) algorithm [6] was implemented to reconstruct image values at the nodes of a fixed finite element-like 2-D triangular mesh. The authors used computer simulations to demonstrate that a multiresolution approach improves the accuracy of the tomographic image reconstruction in 2-D over that of regular grid based methods.

If a regular grid is used, this means that in order to represent small lesions with small voxels, there will be large portions of the grid that will just be “empty” space. For tomographic applications, this increases the number of unknowns in the inverse problem. Our approach is to use a more efficient image description based on nonuniform sampling where image samples are placed most densely in image regions having fine detail and less dense in regions with coarse detail. These grid points can be thought of as forming a cloud of points called a “point cloud.” In our work, we will use the term “point cloud” to refer to the collection of points that define the image. There is a correspondence between the description used in this work and *mesh*

modeling. Mesh modeling is the formation of a mesh of points from an already prescribed digital image. Mesh modeling has been an important part of continuum mechanics but recently has been applied in computer graphics, specifically to medical imaging analysis [7]. These approaches in engineering and computer graphics have recently been put on a sounder mathematical basis using wavelet theory [8]–[12].

For a prescribed point cloud, it is necessary to have a particular set of basis functions in order to define the image throughout the 3-D space. In the work presented in this paper, an image is represented as a point cloud (a set of irregularly placed points) in 3-D space with the volume segmented by a set of tetrahedrons defined by these points. Using tetrahedral image representation and linear interpolation between the nodes of each tetrahedron, a continuous representation of the intensity is achieved that cannot be achieved by regular voxels with constant intensity. The Delaunay algorithm [13] is used to tetrahedralize the volume from a specified point cloud. It is important to note that the linear interpolation inside each tetrahedron implies that the intensity along any line going through the tetrahedron will change linearly. This also implies that intensity along any line through the volume will be continuous and piecewise linear.

One of the major goals of this paper is to formulate the tomographic reconstruction problem using a representation of the reconstruction volume with a set of points defined as a point cloud. Although the tetrahedral representation of the volume is used in this work, there are other choices that can be used for the volume interpolations. For example, irregular voxels created using Voronoi diagrams [14], [15] in 3-D defined by point clouds or spline volume interpolation [16] with a set of control points defined by point clouds can be used. In this work the tetrahedral approach was used because it is relatively easy compared to other approaches to efficiently formulate the tomographic inverse problem using linear interpolation. Using tetrahedral image representation with linear interpolation allows one to model the reconstructed intensity as a continuous function over the reconstruction domain. Each point (node) of the point cloud has a specific location with a particular intensity. The volume of the reconstruction is divided into nonoverlapping tetrahedral cells with each tetrahedron defined by four nodes from the point cloud. The intensity is linearly interpolated throughout each tetrahedral volume from values at the nodes.

This paper is organized as follows. In Section II, details of the definition of a point cloud and details of the volume interpolation using point clouds are presented. Section III describes the algorithm for calculating the system matrix that can be used with an iterative reconstruction algorithm for parallel-, fan-, and cone-beam imaging geometries. Section IV presents an example of the reconstruction algorithm—an adaptive multiresolution iterative reconstruction algorithm. This algorithm is applied to the reconstruction of the projection data generated by computer simulations and projection data experimentally acquired using the Jaszczak cardiac torso phantom. An example of a possible strategy for creating a point cloud is given. In Section V, we outline and discuss some important aspects of the proposed methods. Finally, Section VI summarizes and concludes this work.

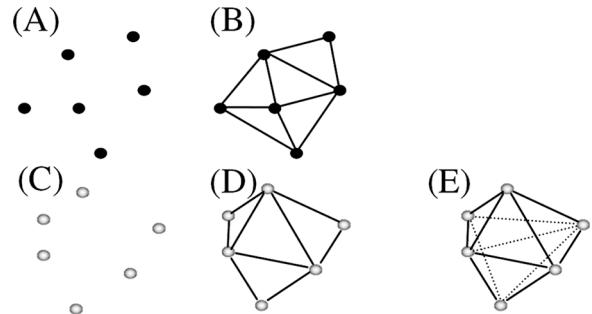


Fig. 1. Results of the Delaunay tetrahedralization algorithm. (A) Six nodes in two dimensions to be triangulated. (B) Result of Delaunay algorithm. The triangulation forms a convex hull. (C) Six nodes (point cloud) in three dimensions to be tetrahedralized. (D) shows results of the Delaunay algorithm with hidden edges not displayed and (E) shows hidden edges displayed. Tetrahedralization of these six nodes created four tetrahedrons.

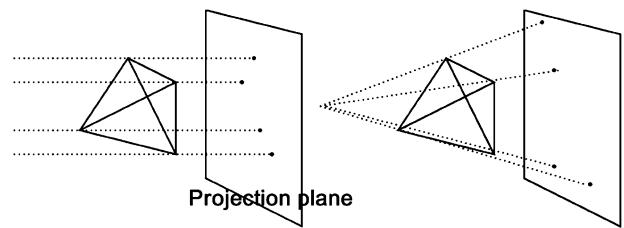


Fig. 2. Illustration of projections of a tetrahedron in parallel and cone beam geometries. In both cases, projections are defined by a set of projections of four nodes that define the tetrahedron.

II. POINT CLOUDS AND TETRAHEDRAL REPRESENTATION OF THE RECONSTRUCTION VOLUME

The reconstruction volume is defined by a set of points called a point cloud. Each point i of the point cloud (*node*) has location x_i and intensity I_i . There is no restriction on the location of the points or their intensity. The volume of the reconstruction region Ω is divided into nonoverlapping tetrahedra with each tetrahedron defined by four nodes from the point cloud.

The Delaunay algorithm [13] was developed to create tetrahedrons from a point cloud. The original Delaunay algorithm creates a tetrahedralization that is a convex hull of the set of points defined by the point cloud. Fig. 1(A) shows a point cloud in two-dimensions and Fig. 1(B) shows the result of the triangulation. The code developed by Si¹ [17], [18] was adapted to perform the Delaunay tetrahedralization. The code [17] is optimized for speed and computes a Delaunay tetrahedralization of 40 000 randomly distributed points in 4.8 s and 1 000 000 points in 3 min using a 3.06-GHz Intel processor [18]. Fig. 1(C)–(E) illustrates a simple 3-D example of the tetrahedralization of six nodes that creates four tetrahedrons.

For tomography, the reconstructed intensity is specified by linearly interpolating between the values of four nodes that define each tetrahedron. The important point to make here is that the linear interpolation for points inside each tetrahedron implies that the intensity along any line going through a tetrahedron will change linearly. This also implies that the intensity along any line going through the reconstruction volume is a piecewise linear function. A 2-D function resulting from a 2-D

¹<http://tetgen.berlios.de>

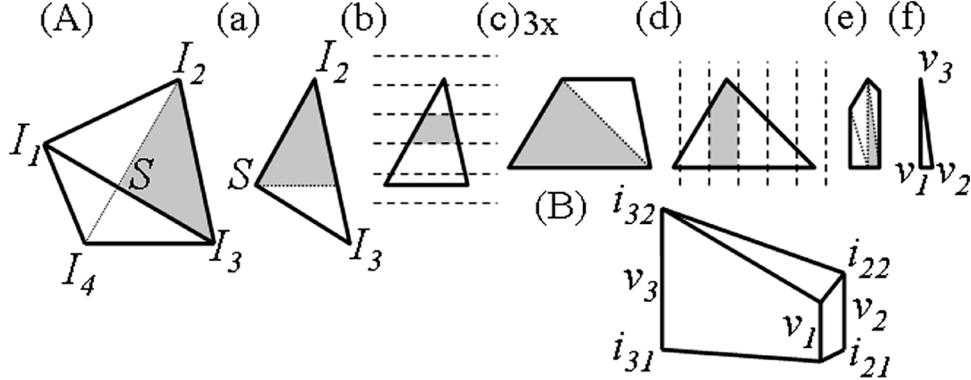


Fig. 3. Projection of a tetrahedron onto a digitized 2-D projection plane is illustrated as a pyramid with quadrilateral base. Thickest point of this pyramid is S . Obviously, the thickness at the projection of nodes 1, 2, 3, and 4 is 0. To simplify the calculation, the bases of the pyramids are divided into triangles, and the values at the corners are interpolated. These triangles are divided further (a) into upper- and lower-parts. Upper part is superimposed in (b) onto rows of the projection bins. Intersection of the upper-part of the triangle with each row creates a trapezoid [shaded area in (b)]. This trapezoid is shown in (c) enlarged three times. Each trapezoid is divided into two triangles and both triangles are superimposed onto columns of projection bins shown in (d). Intersection of each column with the triangles in general creates at pentagon shown in (e) that is divided into a maximum of four triangles. For one triangle illustrated in (f) the thicknesses at the corners are v_1 , v_2 , and v_3 , and are calculated by interpolation using the thicknesses of the initial pyramid. Lower part of the triangle in (b) and the projection bins are flipped vertically and processed identically as the upper-part of the triangle. (B) Slanted prism in (f) is shown enlarged with values of intensities interpolated from I_1, \dots, I_4 . At the base i_{11} , i_{21} , and i_{31} are the values at the corners, and i_{12} , i_{22} , and i_{32} are the values at the vertical corners. For clarity symbols i_{11} and i_{12} were not shown in (B).

cut though the reconstruction volume will also result in a continuous 2-D function.

III. FORMATION OF THE SYSTEM MATRIX

In order to formulate an inverse problem using a set of points (point cloud) defined in Section II, we start with the imaging equation

$$P_j = \sum_{i=1}^M \alpha_{j,i} I_i \quad (1)$$

where P_j is the value of the tomographic projection acquired at bin j . It is assumed that the projection bin has rectangular shape. I_i is the value of the reconstructed intensity at the i th point of the point cloud with M being the total number of points. The matrix α , which relates the projection vector \mathbf{P} to the image vector \mathbf{I} , is called the system matrix. In order to use any iterative algorithm for the tomographic reconstruction, the system matrix α has to be specified. The matrix α describes the physics and the geometry of the tomographic data acquisition. Once the system matrix is known, an algorithm such as the ML-EM algorithm [6], [19] can be used for the reconstruction.

In order to estimate the system matrix α for a point cloud and tetrahedral grid, an algorithm was developed for projecting a tetrahedron onto the detector plane. The algorithm consists of two steps: 1) projection of the nodes and 2) calculation of the weights for the projected tetrahedrons. In the first step, the four nodes that define each tetrahedron are projected as shown in Fig. 2 onto the projection plane according to the geometry of the tomographic apparatus. In the second step, the values of the projection matrix elements are calculated using the positions of the four projected nodes with respect to the projection bin positions. The two-step method is aimed at making the second step independent of the first, thus, independent of projection geometry, so it can be used for any imaging geometry (parallel-, fan-, and cone-beam). However, in order to be able to separate these

two steps for all projection geometries, the assumption has to be made that the magnification angles (Fig. 2) for cone- and fan-beams are small, which is usually true because tetrahedron sizes are small compared to the distances from the detector to the focal points. This assumption is needed because for converging geometries the interpolation between the projected nodes is no longer linear for large magnification angles.

The first step of the algorithm—the projection of the nodes—is straightforward (Fig. 2), thus, we concentrate on describing the method that is used for calculating the weights for the projection of a tetrahedron onto the pixelized projection plane. Obviously, the projections of the tetrahedrons are 2-D, but for the sake of clarity, let us visualize these projections as pyramids. The third dimension is the thickness through the tetrahedron. The height h of the pyramids is $h = 3V/B$, where V is the tetrahedron or pyramid volume and B is the area of the pyramid base. The thickest point of the pyramid is at S (Fig. 3). In general, a contribution to the projection from a single tetrahedron will be an integral of the volume of the pyramid (thickness corresponds to intensity) over the area intersected by a projection bin with a projected pyramid. In order to overcome the high computational load, we developed an algorithm that analytically calculates the exact integrals of intensities over volumes of pyramids corresponding to projections of tetrahedrons onto the projection plane.

The schematic of the algorithm is presented in Fig. 3. The bases of the pyramids are divided into triangles. Each of these triangles is further divided in upper- and lower-parts [Fig. 3(A),(a)]. The upper part is superimposed onto rows of the projection bins [Fig. 3(A),(b)]. One intersection, which forms a trapezoid, is separated and shown in [Fig. 3(A),(c)]. This is subdivided further into triangles. The lower triangle is shown superimposed onto the columns of the projection bins [Fig. 3(A),(d)]. One of the intersections forms a pentagon, which is divided into four triangles [Fig. 3(A),(e)]. One of these triangles is shown in Fig. 3(A),(f). The thicknesses at the

corners are denoted as v_1 , v_2 , and v_3 . These are determined by interpolating the thicknesses of the initial pyramid. The final triangle illustrated in Fig. 3(A),(f) is the result of subdividing triangles, trapezoids, and pyramids up the line. This triangle is illustrated in Fig. 3(B) as a slanted prism with values c_{ij} obtained by interpolating the intensities I_1, \dots, I_4 of the original pyramid. The calculation needs to be performed for all the subdivisions. For example, the calculation needs to be performed for all the triangles in Fig. 3(A),(e), likewise for all of the segments in Fig. 3(A),(d), and so forth up the line. The subdivision in triangular segments forms an efficient method for calculating the projection weights for each tetrahedron.

For the slanted prism in Fig. 3(B), we derived an analytical formula for the integral Δ of the linearly changing intensity inside the slanted prism

$$\Delta = \frac{A}{12} [2(\bar{c}_1 v_1 + \bar{c}_2 v_2 + \bar{c}_3 v_3) + \bar{c}_1 v_2 + \bar{c}_1 v_3 + \bar{c}_2 v_1 + \bar{c}_2 v_3 + \bar{c}_3 v_1 + \bar{c}_3 v_2] \quad (2)$$

where the thickness of the prism at the corners is v_1 , v_2 , and v_3 , and A is the area of the triangle base. The values \bar{c}_1 , \bar{c}_2 , and \bar{c}_3 are averages calculated from values of the intensities at the corners. The values $c_{11}, c_{21}, c_{31}, c_{12}, c_{22}$, and c_{32} [Fig. 3(B)] are the intensities obtained by linearly interpolating the values at the tetrahedron nodes I_1, \dots, I_4 , where $\bar{c}_1 = (c_{11} + c_{12})/2$, etc. In general, since c_{ij} will be some linear combination of I_n , we see from (2) that the integral will also be some linear combination of I_n 's. Calculating all the integrals (Δ 's) for each prism of the decomposed pyramid (each pyramid corresponds to a projection of one tetrahedron), the exact projection of a tetrahedron can be obtained. With this approach, we are able to exactly project very small, as well as, very large tetrahedrons. For example, tetrahedrons smaller than the size of a projection bin can also be projected exactly. This feature is very important in our approach since, in general, we do not restrict the size of tetrahedrons that constitute the reconstruction area.

Fig. 4 presents two projections of the same tetrahedron but made from different angles. There are no aliasing artifacts due to the magnification of the edge [Fig. 4(C)]. The profile [Fig. 4(D)] is a quadratic piecewise function due to the linear contribution from the linearly changing intensity and the linearly changing thickness of the tetrahedron. The quadratic behavior of this profile can also be deduced from (2). Since v and c are linear, Δ must be a quadratic function.

The system matrix is calculated as follows. A single node j is set to the value of 1 and the others to 0. To obtain the j th column of the system matrix, all tetrahedrons are projected for each angle. The values in projection bins after the projection are equal to elements of the j th column of the system matrix. Obviously, to speed up the calculation, only tetrahedrons that contain node j (nonzero node) need to be projected since all others have intensities equal to zero.

IV. RECONSTRUCTION EXAMPLES

Two examples are presented of the reconstruction of a point cloud segmented into nonoverlapping tetrahedra. One is the reconstruction of computer simulated projections and the other

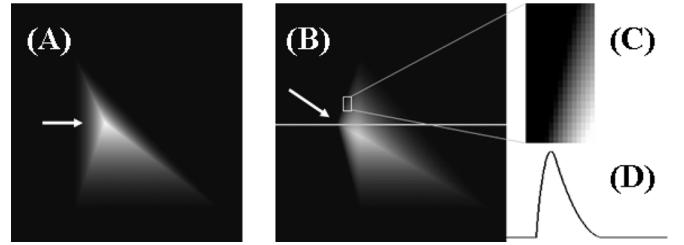


Fig. 4. Images (A) and (B) present projections of the same tetrahedron from two different angles onto a 256×256 pixel projection plane using the technique in Fig. 3. Intensity for only one node (marked by arrow) is nonzero and the intensities of the other three nodes are zero. Image (C) is a magnification of the rectangular region marked on image (B) showing the pixelization. Plot (D) shows a profile through the projection (B). Location of the profile is marked by the line in (B).

is the reconstruction of projections acquired from a SPECT phantom study. The volume of the reconstruction is divided into nonoverlapping tetrahedra with each tetrahedron defined by four nodes from the point cloud. The values of the intensities were reconstructed using the ML-EM algorithm [6], [19] with a system matrix calculated using the method described in Section III. No regularization (except postsmothing) was used because although well developed for the standard voxel approach the regularization methods have not yet been developed and optimized for point cloud image representation. In order to be fair in the comparison we decided to use the straightforward ML-EM algorithm. The reconstruction involved determining both the intensity of the nodes as well as the location of additional nodes. The algorithm that was used in this paper for the addition of nodes is just an example and in no way do we claim that it is optimal. Other methods of node placement are presented in Section V.

If the minimum number of nodes required to construct the point cloud is N , the following algorithm was used:

- Step 1) start with a coarse regular grid with n nodes where $n \ll N$ nodes (e.g., $n = 1000$);
- Step 2) tetrahedralize to obtain the number of tetrahedrons T (Section II);
- Step 3) calculate the system matrix (Section III);
- Step 4) reconstruct intensities at the nodes using the ML-EM algorithm;
- Step 5) add $n_a T$ ($0 < n_a < 1$) number of nodes to the point cloud at the centers of the tetrahedrons with the largest variations in the reconstructed intensities and with volumes $V > V_{\min}$;
- Step 6) if $(n = n + n_a T) < N$ disregard previous tetrahedralization and go to 2 else END.

All steps in the above algorithm were discussed in the previous sections except Step 5), where the nodes are added to the point cloud. In the current implementation of the algorithm, we added nodes at the centers of the tetrahedrons with the highest variations in the intensities v between the four nodes that define each tetrahedron (please see Section V for some other criteria for node placements that can be used). The v for a tetrahedron with I_1, \dots, I_4 was calculated as $v = \sum_{i=1}^4 (I_i - \bar{I})^2$, where $\bar{I} = \sum_{i=1}^4 I_i / 4$. In order to prevent over-sampling, nodes were added to tetrahedrons only if the tetrahedron volumes were

TABLE I
COMPUTING TIMES FOR FIVE CYCLES OF THE ALGORITHM

Cycle	1	2	3	4	5
# tetrahedrons	4k	14k	47k	141k	445k
Delaunay	<0.1	0.3	0.6	1.6	5.5
System Matrix	56	172	470	1469	4424
ML-EM per iteration			3.1	4.2	- ^a

All values are in seconds.

Data for Intel Xeon 3.4GHz. Software compiled and optimized for speed with Microsoft Visual Studio C++ compiler.

^aA value could not be measured accurately due to insufficient memory causing extensive swapping.

above some threshold V_{\min} . The variable n_a is an arbitrary parameter describing the number of nodes to be added at each iteration. In this paper, we used $V_{\min} = 0.1a^3$, where a was the size of the square projection bin, and $n_a = 0.25$. Please note that when using such an algorithm the initial shape of the reconstruction area is unchanged because a point can be added only inside existing tetrahedrons.

In terms of computer complexity, the most computationally expensive steps are steps 2, 3, and 4. The times needed for calculation of these steps are listed in the Table I.

A. Computer Simulations

A computer simulation was performed to evaluate the proposed methods. The object consisted of a big sphere with diameter of $64a$ and density 1. The unit of length was equal to the size of the projection bin a . The object also contained three smaller spheres with diameters $16a$, $4a$, and $4a$ positioned off-center inside the large sphere with intensities 0.5, 4, and 2, respectively.

Simulated parallel projection data consisted of 32 projections 128×128 square bins each acquired over an angular range of 180° . The projection for one angle is shown in Fig. 5(A). The value of each projection bin was determined by numerical integration of line integrals going through the object where each line integral was calculated analytically. For each projection bin 64 line integrals were considered. The simulated projections were reconstructed using six cycles of the algorithm. The following numbers of ML-EM iterations were used 1, 1, 2, 10, 200, and 400 for six subsequent cycles.

As can be seen in Fig. 5(C) and (D), the quality of the projection of the tomographic reconstruction increases as more nodes are included in the point cloud. The projection of the final reconstruction with 121 000 nodes [Fig. 5(B)] is virtually indistinguishable from the original projection presented in Fig. 5(A). Fig. 6 presents cuts through the reconstruction area for different cycles of the reconstruction algorithm. Notice that the density of the nodes is increased at the edges of the myocardium. The images in Fig. 6(A)–(D) correspond to the images in Fig. 5(C)–(F). The projection in Fig. 5(F) is from a reconstruction yielding 67 908 nodes.

B. Experimental Phantom Studies

The method was also tested using experimentally acquired phantom data. We used the Jaszczak anthropomorphic cardiac torso phantom (Data Spectrum, Hillsborough, NC). Three different sets of projections were generated with 2:1, 5:1, and 10:1 ratio of activity of ^{99m}Tc placed in the myocardium compartment and in the background that included blood pool, torso, and

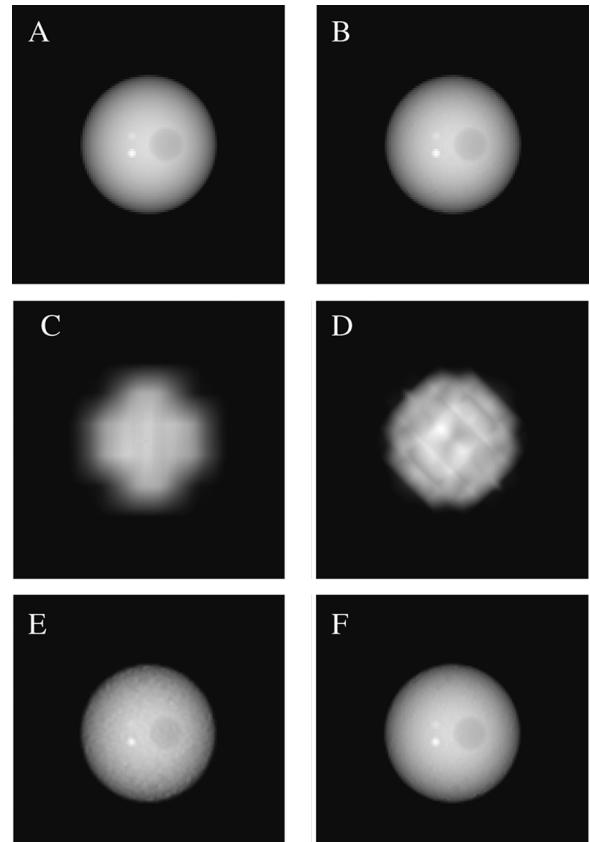


Fig. 5. Images present a single original projection of the numerical phantom (A) and a final reconstructed projection of the grid with 121 000 nodes (B). Images (C)–(F) correspond to reconstructed projections of the grid for four different cycles of the reconstruction algorithm (they correspond to grids presented in Fig. 6(A)–(D)). Increased accuracy for subsequent cycles can clearly be seen.

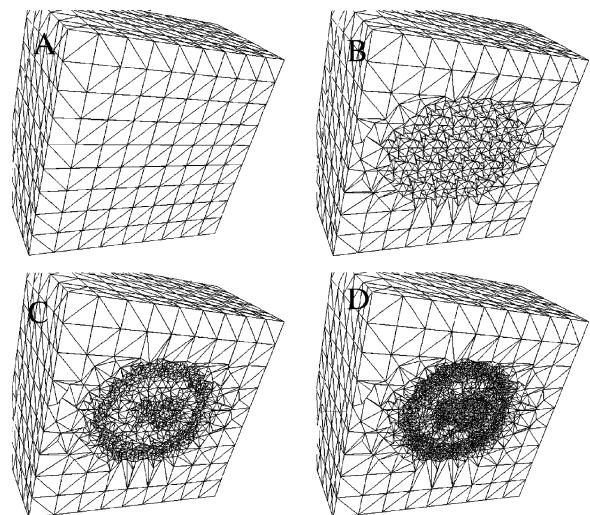


Fig. 6. Results of the reconstruction using a tetrahedral grid and a numerical phantom. Images (A)–(D) present cuts through the multiresolution grid at four different cycles of the reconstruction algorithm. (A) corresponds to an initial coarse mesh with 1000 nodes. Meshes (B), (C), and (D) correspond to the third, fourth, and fifth subsequent iterations with 6949, 21456, and 67 908 nodes, respectively. Density of the nodes increases mainly around the edges of the object.

liver compartments. There was no activity placed in the spinal cord and in the lungs. Sixty projections in 128×128 matrix were acquired over 180° using a GE Millennium VG3 camera with

parallel collimators. The acquisition was performed at 140 keV with a 20% energy window. The numbers of counts in the 3-D sinograms were equal to 3.72, 3.84, and 4.05 million, respectively.

The 3-D image consisting of a point cloud of nonoverlapping tetrahedral elements was reconstructed from the sinograms using five cycles of our algorithm with 2, 2, 10, 200, and 200 iterations of the ML-EM algorithm in each cycle, respectively. Another 3-D image consisting of the standard rectangular cubic grid of $128 \times 128 \times 128$ with a voxel size a was reconstructed from the same set of sinograms using 200 iterations of the ML-EM algorithm. No attenuation or detector response was modeled in the reconstruction of either image.

Following the reconstruction by both methods, the reconstructed images were regularized by applying a 3-D Gaussian filter with a full-width at half-maximum (FWHM) of $2a$. This is a straight forward procedure for a regular grid and somewhat more involved for a tetrahedral grid. If I_1, I_2, \dots, I_K are intensities of nodes directly connected to a node with intensity I_0 , where K is the number of nodes, then the regularized value R_0 is

$$R_0 = \frac{\sum_{k=0}^K I_k e^{-ad_k^2}}{\sum_{t=0}^K e^{-ad_t^2}} \quad (3)$$

where $a = 4 \ln 2/w_{1/2}$, and d_k and d_t are the distances from node 0 to nodes k and t , respectively, and $w_{1/2}$ is the FWHM of the Gaussian curve. The results of the reconstructions are presented in Fig. 7.

V. DISCUSSION

A method is presented for image representation and tomographic reconstruction based on unstructured distribution of point clouds with non-overlapping tetrahedrons. An important result of the paper is the formulation of the inverse problem using point cloud image representation and the development of an efficient method for calculating the system matrix for a tetrahedral segmentation of the image space. This approach has the potential of high impact in nuclear medicine imaging because the new image representation may prove to be superior to standard voxel-based images. The main motivation of this work was to create a framework for further investigation of the point cloud image representation. The choice of the image representation was dictated by the hypothesis that with smaller number of unknowns the tomographic inverse problem will be better posed. Second, our image representation (tetrahedron based) is designed to take advantage of readily available graphics hardware in order to explore 3-D visualization modes. This graphics hardware support of our configuration was one of the main reasons for this development, and once fully developed may become a very important tool for medical image visualization. This is an area of research that we are actively pursuing. There are several other important issues associated with the methods presented that we would like to emphasize in the following discussion.

One of the important aspects of our method is that a tetrahedral geometry of the reconstruction volume can be visualized in

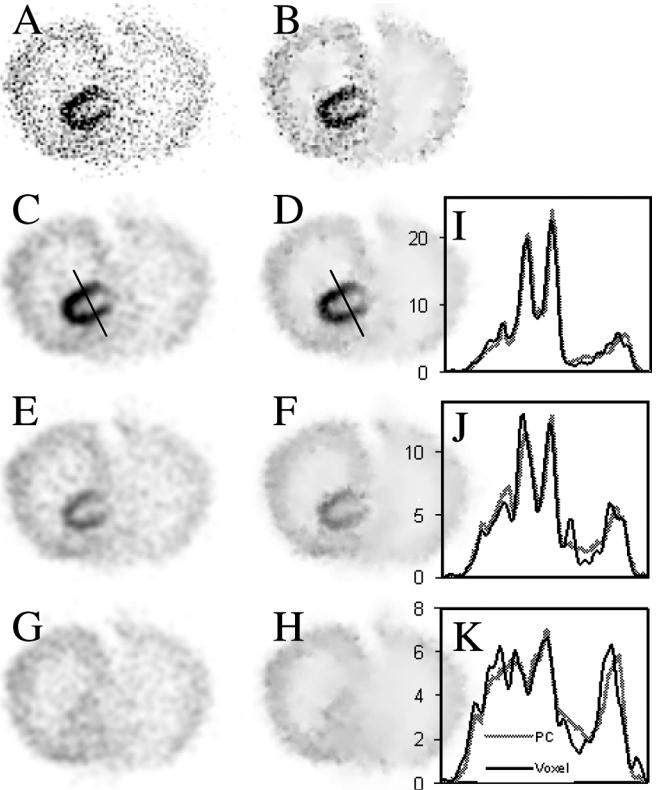


Fig. 7. Images are slices through the volume of the Jaszczak torso phantom with cardiac insert reconstructed from projections acquired using the GE Millennium VG3 SPECT camera. (A) Cut through the volume reconstructed on a $128 \times 128 \times 128$ cubic grid using 200 iterations of the ML-EM algorithm. (B) Cut though the volume reconstructed to give a point cloud consisting of 62 000 nodes. This point cloud was obtained by five cycles of the reconstruction algorithm. Images (C) and (D) represent the same cuts as (A) and (B) after the application of the 3-D Gaussian filter with a FWHM of $2a$. Images (E) and (F) correspond to the same slice but with ratio of activity between the heart and background of 5:1, and (G) and (H) with 2:1. Graphs (I)–(K) present the profiles through the images corresponding to lines presented in (C) and (D).

3-D using commodity graphics processing units (GPUs). In recent years, the development of graphics hardware has grown so fast that the GPU performance doubles every six months. The rapid development of the GPU is driven by the video game industry. Currently, the NVidia GeForceFX 6800 GPU (NVidia Corporation, Santa Clara, CA) contains a chip that performs floating point operations almost an order of magnitude faster than a 3-GHz Pentium 4 central processing unit (CPU). The GPU has already been used for tomographic reconstructions using an iterative reconstruction algorithm with the standard, regularly distributed, voxelized reconstruction region to achieve 8–10 times faster reconstruction times compared to reconstruction times for a standard implementation on a CPU [20] and [21]. The GPU could also be used for tomographic reconstruction with tetrahedral grids. Keep in mind that volume rendering performed on the GPU is very similar to a projection operation in tomographic reconstruction. The GPU can be adapted for the most computationally demanding tasks associated with iterative reconstruction algorithms, which is the creation of the system matrix. The imaging application in [20] and [21] used texture mapping techniques for which it was convenient to reconstruct on regular grids. However, since the texturing techniques used in

[20] and [21] are not well suited for the application to tetrahedral grids [22], new algorithms for the visualization of reconstructions of tetrahedral elements need to be developed or adopted from the literature dealing with visualization of unstructured grids in computer graphics [22]–[25].

The current reconstruction implementation presented as an example in this paper uses a suboptimal algorithm for the placement of additional nodes. Nodes are added to the point cloud in Step 5) of the algorithm (Section IV), specifically to tetrahedrons with the highest deviation in intensity between the four nodes. The more optimal and probably the more effective placement would be to add nodes to centers of tetrahedrons with the highest mean curvature. Mean curvature is defined as the average of the three eigenvalues of the Hessian matrix of the intensity function [26]. Other algorithms, such as the minimum description length (MDL) principle [27], can also be used for determining the optimal number of nodes and their locations. However, the computational complexity for determining the second derivative for every node or the computation time needed for optimization using the MDL implementation may offset the benefit of adding addition nodes. Other approaches that could be used for determination of the point locations is to reconstruct the image first using the regular grid, then based on this reconstruction distribute the points in optimal locations. This approach, however, would slightly limit the practical values of our method because it would require performing the standard reconstruction first. However, some fast analytical techniques (such as filtered backprojection [1]) can be used for the initial voxel-based reconstruction, increasing only minimally the computational burden of this method. Once the voxel-based image estimate is reconstructed, points can be distributed, for example, using the approach developed in [28]. The other interesting issue that needs to be studied is whether the algorithms for distribution of points should be the same for all kinds of nuclear medicine studies. For example, one would expect to have nodes distributed differently for oncology studies where low-contrast lesions may be positioned on low intensity background than for cardiac studies where nodes probably should be concentrated at the boundaries of the myocardium. These issues need to be investigated before any strong conclusion as to what is the optimal algorithm for node distribution can be drawn. We plan to investigate this in future work.

One of the most important future steps in the development of this method is the implementation of attenuation and detector response correction for PET and SPECT. The implementation of the attenuation correction is straightforward and could be done when calculating the system matrix by modifying the intensity at each node j from 1 to $\exp(-\int \mu dl)$, where the line integral goes from the node location to the location of the projection bin and integrates the distribution of attenuation coefficients in the attenuation map. Depth-dependent correction for the detector response can also be implemented in the system matrix by applying an appropriate smoothing filter to the projection of each tetrahedron. This will model the effect of depth dependent blurring. The amount of smoothing will depend on the distance from the node j to the particular projection bin. These corrections will be implemented in the future.

The algorithm presented in this paper lacks regularization features that are needed in order to implement MAP iterative reconstruction algorithms [6], [29]–[32]. A large amount of research literature exists on various regularization and filtering methods for regular grids. These methods are necessary to reduce noise for viewing by readers in clinics using 2-D displays. Similar regularization tools need to be developed for tetrahedral mesh defined by a point cloud.

The other direction of research is using the 3-D visualization to suppress noise in the reconstructed images. It is expected that the noise can be suppressed (at least to some extent) using 3-D visualization techniques, such as volume rendering that averages out the noise by integrating intensities. The visualization aspects of the point clouds and tetrahedral grids are an important continuation and extension of this work that we will investigate.

VI. SUMMARY AND CONCLUSION

In summary, we have described a new fully 3-D method for tomographic reconstruction using a reconstruction domain defined by a point cloud. A method was presented for the continuous representation of reconstructed images in three dimensions by linear interpolating within nonoverlapping tetrahedrons defined by points of the point cloud. An ingenious method was presented for efficiently calculating the system matrix for iterative reconstruction for this somewhat more complex but yet more efficient image representation. Using data from a computer simulation and from an experimental phantom study, examples of tomographic reconstructions were presented using this developed methodology.

We believe that our image representation is better suited for representation of 3-D images due to the adaptive multiresolution nature of point clouds, thus providing an efficient representation of medical images. Since these images are represented by a set of unstructured points that can easily be moved in time without destroying the reconstruction framework, the extension of this work could involve the application of the point cloud model to motion tracking and motion compensation directly in the tomographic reconstruction process similar to the work of Gilland *et al.* [33], [34]. Although we believe that the impact of this work on medical imaging will be high, a strict evaluation and comparison to standard reconstructions methods need to be performed before drawing any strong conclusions.

Three-dimensional display of medical images is becoming more and more popular with advances in hardware and computational speed. However, reconstructed images are often still represented by a stack of 2-D images. This work provides an alternate fully 3-D image representation for which the full advantage will be realized only in conjunction with 3-D visualization, which is a future direction of our work.

ACKNOWLEDGMENT

The authors would like to thank S. Webb for proofreading the manuscript.

REFERENCES

- [1] G. T. Herman, *Image Reconstruction from Projections: The Fundamentals of Computerized Tomography*. New York: Academic, 1980.

- [2] W. Chlewicki, F. Hermansen, and S. B. Hansen, "Noise reduction and convergence of Bayesian algorithms with blobs based on the Huber function and median root prior," *Phys. Med. Biol.*, vol. 49, pp. 4717–4730, 2004.
- [3] R. M. Lewitt and S. Matej, "Overview of methods for image reconstruction from projections in emission computed tomography," *Proc. IEEE*, vol. 91, no. 10, pp. 1588–1611, Oct. 2003.
- [4] S. Matej, G. T. Herman, T. K. Narayan, S. S. Furuike, R. M. Lewitt, and P. E. Kinahan, "Evaluation of task-oriented performance of several fully 3-D PET reconstruction algorithms," *Phys. Med. Biol.*, vol. 39, pp. 355–367, 1994.
- [5] J. G. Brankov, Y. Yang, and M. N. Wernick, "Tomographic image reconstruction based on a content-adaptive mesh model," *IEEE Trans. Med. Imag.*, vol. 23, no. 2, pp. 202–212, Feb. 2004.
- [6] K. Lange and R. Carson, "EM reconstruction algorithms for emission and transmission tomography," *J. Comput. Assist. Tomogr.*, vol. 8, pp. 306–316, 1984.
- [7] A. Singh, D. Terzopoulos, and D. B. Goldgof, *Deformable Models in Medical Image Analysis*. Los Alamitos, CA: IEEE Comput. Soc. Press, 1998.
- [8] I. Daubechies, I. Guskov, P. Schroder, and W. Sweldens, "Wavelets on irregular point sets," *Philos. Trans. Roy. Soc. Lond. A, Math. Phys. Eng. Sci.*, vol. 357, pp. 2397–2413, 1999.
- [9] T. Frese, C. A. Bouman, and K. Sauer, "Adaptive wavelet graph model for bayesian tomographic reconstruction," *IEEE Trans. Image Process.*, vol. 11, no. 7, pp. 756–770, Jul. 2002.
- [10] S. Mallat, *A Wavelet Tour of Signal Processing*, 2nd ed. New York: Academic, 1999.
- [11] R. D. Nowak and E. D. Kolaczyk, "A statistical multiscale framework for Poisson inverse problems," *IEEE Trans. Inf. Theory*, vol. 46, no. 5, pp. 1811–1825, Aug. 2000.
- [12] R. M. Willett and R. D. Nowak, "Platelets: A multiscale approach for recovering edges and surfaces in photon-limited medical imaging," *IEEE Trans. Med. Imag.*, vol. 22, no. 3, pp. 332–350, Mar. 2003.
- [13] B. N. Delaunay, "Sur la sphère vide," *Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk*, vol. 7, pp. 793–800, 1934.
- [14] F. Aurenhammer, "Voronoi diagrams—A survey of a fundamental geometric data structure," *ACM Computing Survey*, vol. 23, pp. 345–405, 1991.
- [15] P. Cignoni, C. Montani, and R. Scopigno, "DeWall: A fast divide & conquer Delaunay triangulation algorithm in $E \wedge d$," *Comput. Aided Des.*, vol. 30, pp. 333–341, 1998.
- [16] J. Hua, Y. He, and H. Qin, "Trivariate simplex splines for inhomogeneous solid modeling in engineering design," *J. Comput. Inf. Sci. Eng.*, vol. 5, pp. 149–157, 2005.
- [17] H. Si and K. Gartner, "An algorithm for three-dimensional constrained Delaunay tetrahedralizations," presented at the 4th Int. Conf. Eng. Comput. Technol., Lisbon, Portugal, Sep. 7–9, 2004.
- [18] H. Si, TetGen. ver. 1.3, Berlin, Germany, 2005 [Online]. Available: <http://tetgen.berlios.de>
- [19] L. A. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Trans. Med. Imag.*, vol. MI-1, no. 1, pp. 113–122, Feb. 1982.
- [20] K. Chidlow and M. T. , "Rapid emission tomography reconstruction," presented at the 2003 Eurographics/IEEE TVCG Workshop Volume Graphics, Tokyo, Japan, Jul. 7–8, 2003.
- [21] F. Xu and K. Mueller, "Accelerating popular tomographic reconstruction algorithms on commodity PC graphics hardware," *IEEE Trans. Nucl. Sci.*, vol. 52, no. 3, pp. 654–663, Jun. 2005.
- [22] S. P. Callahan, M. Ikits, J. L. D. Comba, and C. T. Silva, "Hardware-assisted visibility sorting for unstructured volume rendering," *IEEE Trans. Vis. Comput. Graphics*, vol. 11, no. 3, pp. 285–295, May/Jun. 2005.
- [23] R. Cook, N. Max, C. T. Silva, and P. L. Williams, "Image-space visibility ordering for cell projection volume rendering of unstructured data," *IEEE Trans. Vis. Comput. Graphics*, vol. 10, no. 6, pp. 695–707, Nov./Dec. 2004.
- [24] M. Ikits, J. Kniss, A. Lefohn, and C. Hansen, *GPU Gems: Programming Techniques, Tips, and Tricks for Real-Time Graphics*. Boston, MA: Addison Wesley, 2004.
- [25] M. Weiler, M. Kraus, M. Merz, and T. Ertl, "Hardware-based view-independent cell projection," *IEEE Trans. Vis. Comput. Graphics*, vol. 9, pp. 163–175, 2003.
- [26] E. Kreyszig, *Differential Geometry*. Mineola, NY: Dover, 1991.
- [27] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465–471, 1978.
- [28] J. G. Brankov, Y. Yang, and M. N. Wernick, "Content-adaptive 3-D mesh modeling for representation of volumetric images," presented at the 2002 Int. Conf. Image Process., Jun. 24–28, 2002.
- [29] M. T. Chan, G. T. Herman, and E. Levitan, "A Bayesian approach to PET reconstruction using image-modeling Gibbs priors: Implementation and comparison," *IEEE Trans. Nucl. Sci.*, vol. 44, no. 3, pp. 1347–1354, Jun. 1997.
- [30] J. A. Fessler and S. D. Booth, "Conjugate-gradient preconditioning methods for shift-variant PET image reconstruction," *IEEE Trans. Image Process.*, vol. 8, no. 5, p. 688, May 1999.
- [31] Q. Jinyi and R. M. Leahy, "Resolution and noise properties of MAP reconstruction for fully 3-D PET," *IEEE Trans. Med. Imag.*, vol. 19, no. 5, pp. 493–506, May 2000.
- [32] D. S. Lalush and B. M. Tsui, "A generalized Gibbs prior for maximum *a posteriori* reconstruction in SPECT," *Phys. Med. Biol.*, vol. 38, pp. 729–741, 1993.
- [33] D. R. Gilland, B. A. Mair, J. E. Bowsher, and R. J. Jaszcak, "Simultaneous reconstruction and motion estimation for gated cardiac ECT," *IEEE Trans. Nucl. Sci.*, vol. 49, no. 5, pp. 2344–2349, Oct. 2002.
- [34] D. R. Gilland, B. A. Mair, and J. Sun, "Joint 4D reconstruction and motion estimation in gated cardiac ECT," presented at the Fully 3-D Image Reconstruction Conf., Salt Lake City, UT, Jul. 6–9, 2005.