CMIG449

PERGAMON

Computerized Medical Imaging and Graphics 00 (2001) 000–000

Computerized Medical Imaging and Graphics

www.elsevier.com/locate/compmedimag

Automatic axis generation for virtual bronchoscopic assessment of major airway obstructions

R.D. Swift[a], A.P. Kiraly[a], A.J. Sherbondy[a], A.L. Austin[a], E.A. Hoffman[b], G. McLennan[b], W.E. Higgins[a,b,*]

[a]Department of Electrical Engineering, Penn State University, 121 Electrical Engineering East, University Park, PA 16802, USA
[b]College of Medicine, University of Iowa, Iowa City, IA 52242, USA

Received 5 July 2001; accepted 29 August 2001

Abstract

Virtual bronchoscopy (VB) has emerged as a paradigm for more effective 3D CT image evaluation. Systematic evaluation of a 3D CT chest image using VB techniques, however, requires precomputed guidance data. This guidance data takes the form of central axes, or centerlines, through the major airways. We propose an axes-generation algorithm for VB assessment of 3D CT chest images. For a typical high-resolution 3D CT chest image, the algorithm produces a series of airway-tree axes, corresponding airway cross-sectional area measurements, and a segmented airway tree in a few minutes on a standard PC. Results for phantom and human airway-obstruction cases demonstrate the efficacy of the algorithm. Also, the algorithm is demonstrated in the context of VB-based 3D CT assessment. © 2001 Elsevier Science Ltd. All rights reserved.

Keywords: Virtual bronchoscopy; Virtual endoscopy; Centerlines; 3D pulmonary imaging; Path planning; CT bronchography; 3D imaging

1. Introduction

Major airway obstruction is a common problem arising from lung cancer, benign processes, and other tumors that commonly metastasize the airways. Impending obstruction to the trachea is life threatening, while obstruction to the left or right main bronchi are associated with significant morbidity. These patients need adequate evaluation prior to an intervention such as laser resection, balloon bronchoplasty, stent insertion, cryotherapy or brachytherapy. Three-dimensional (3D) computed-tomography (CT) pulmonary images are commonly used for such evaluation [1]. The evaluation of these images is generally done manually with film records [2]. A film record shows a series of transverse-oriented two-dimensional (2D) slice images. The physician performs 3D mental reconstruction of anatomical structures depicted in the film to evaluate the case. While often suitable for the radiologist, this form of evaluation is typically inadequate for the bronchoscopist or surgeon, who need precise 3D 'road maps' to the surgical sites of interest [3].

Recently, *virtual bronchoscopy* (VB) has emerged as a paradigm for more effective 3D CT image evaluation [4–12]. VB is a sub-branch of the new field often referred to as *virtual endoscopy* [13,14]. When related to the chest, virtual endoscopy involves the use of computer-based image-processing and graphical techniques to observe structures inside the 'virtual chest environment', as defined by a 3D CT pulmonary image. Since all evaluation is computer-based, the user can navigate through the 3D chest environment with great flexibility.

3D CT images used in tandem with VB techniques can help the physician evaluate patients suffering from major airway obstruction. Systematic evaluation of a 3D CT chest image using VB techniques, however, requires precomputed guidance data [8]. This guidance data generally takes the form of central axes, or centerlines, through the major airways. Such axes not only give a means for evaluating a 3D CT chest image, but also potentially provide road maps for subsequent bronchoscopic procedures (the bronchoscope passes through the airways!).

We propose a semi-automatic axes-generation algorithm for VB assessment of 3D CT chest images. The proposed algorithm can generate a series of airway-tree axes for a typical high-resolution 3D CT chest scan in a few minutes on a standard PC. It is integrated into a PC-based system for VB CT assessment and follow-on bronchoscopic guidance

[*] Corresponding author. Tel.: +1-814-865-0186; fax: +1-814-863-5341. *E-mail address:* weh2@psu.edu (W.E. Higgins).

0895-6111/01/$ - see front matter © 2001 Elsevier Science Ltd. All rights reserved.
PII: S0895-6111(01)00035-0

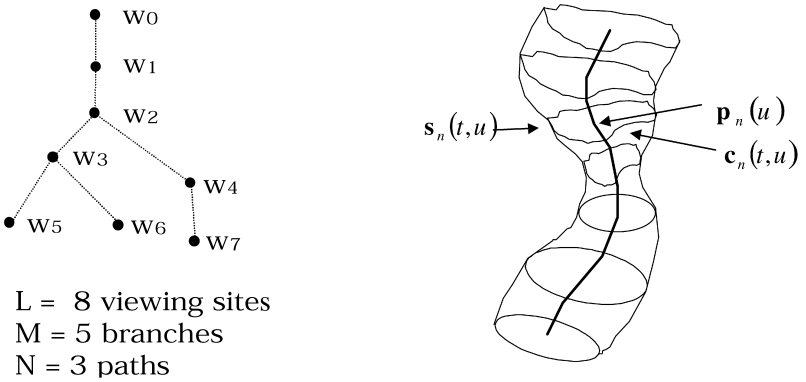

Fig. 1. Models used in the two-satge axes-generation algorithm. Left: example of a Discrete Model for a tree made up of eight viewing sites $\mathbf{w}_l, l = 0,2,\ldots,7$; Right: example of a Generalized-Cylinder (GC) Model for a single path, where $\mathbf{s}_n(t,u)$ is the GC surface function specified by spine (central axis) $\mathbf{p}_n(u)$ and contour function $\mathbf{c}_n(t,u)$. See Section 3 for more.

[15,16]. The results, given for phantom and human cases, demonstrate the efficacy of the algorithm. The human results focus on the problem of assessing major airway obstructions.

Section 2 introduces the problem of axes generation for 3D CT chest images. Section 3 describes the proposed axes-generation method. Section 4 provides a large series of results employing phantom and human data. Finally, Section 5 offers concluding remarks.

2. Problem overview

Mathematical notation and problem constraints are defined within the context of finding the central axes of the major airways depicted in a 3D CT pulmonary image V. We assume that V consists of a contiguous series of 16-bit slices spaced Δz apart. The transverse-plane sample spacings are Δx and Δy. A voxel is denoted by (x,y,z) and voxel (x,y,z)'s intensity value is given by $V(x,y,z)$. The airway tree consists of a complex, branching, connected set of dark airways surrounded by relatively bright airway walls. Limited spatial resolution, noise, partial volume effects, and image-reconstruction artifacts complicate the extraction of the airway-tree axial structure.

We assume that an algorithm devised for extracting the major airways and associated axes (also called *paths*) abides by the following requirements: (a) paths evolve slowly along their extent, thereby providing a smooth trajectory for navigation; (b) paths for a 'complete' tree are generated (i.e. smooth paths through multiple airways are generated in one pass); (c) the extracted airways conform to the measurable morphology and gray-scale characteristics of a typical 3D CT pulmonary image; (d) the algorithm preserves the homotopy of the original structure; (e) paths approximate the medial (central) axes of the branches in the structure; (f) airway cross-sections change little from one point along a path to the next.

Previously proposed axes-generation techniques have employed either manual image interaction or automatic processing. Manual approaches have a user either fully trace a path, do significant manual segmentation, or specify a priori sites (key frames) that are later interpolated into a complete path [4,13,17]. Unfortunately, a 3D image's complexity and dimensionality make such procedures prohibitive for practical use, especially for images generated by new multidetector helical CT scanners [18]. Automated approaches have employed: (1) segmentation followed by 3D skeletonization [8,12,19–23]; (2) morphological operations [24–26]; (3) active contour modeling [27]; (4) tubular structure analysis, using differential geometry [23,28–33]. These techniques either lead to imprecise or missing axes (particularly when pathologies and noise may be present), require inordinate processing time, or are not well-suited to our specific problem of defining precise smooth central axes suitable for navigation through a complex branching-tree structure. For example, many techniques only give one path [21,22,27,29,31]. Many techniques do not give smooth navigation axes, but instead give coarser digitized axes [8,19,20,23–25,32]. A few proposed techniques for tracking vessels in coronary angiograms considered a problem similar to ours [26,28]. These methods used tree continuity assumptions and an adaptive tracking filter matched to the expected gray level profile of a vessel cross-section. These methods, while limited to 2D projection images of 3D structures, motivate our algorithm for true 3D airway analysis.

Our algorithm follows a two-stage approach. The first stage computes a *Discrete Model* that consists of a sparsely spaced set of data corresponding to the major airway axes. The second stage then uses the Discrete Model to define a smooth set of airway axes, a segmented airway tree, and other measurement data; these data constitute the *Generalized Cylinder (GC) Model* [34–37]. Fig. 1 schematically illustrates these two model components.

The basic flow of the two-stage algorithm is as follows. The user first specifies a starting point of interest for the airway tree, generally in the proximal end of the trachea. Then, to perform the Stage-1 calculation of the Discrete Model, an adaptive 3D searching technique steps through the 3D image, building up a sparsely sampled set of central axes for the major airways. Stage-2 then applies a cubic-spline analysis and contour-finding procedure to the Discrete Model data to give a generalized cylinder representation for each airway central axis; this gives the GC Model. Section 3 gives a complete description of the two-stage algorithm and the model components.

3. Axes-definition algorithm

Section 3.1 defines many of the analytical quantities constituting the Discrete Model and GC Model. Section 3.2 summarizes the two-stage algorithm. Finally, Section 3.3 and Section 3.4 gives further details on key algorithm steps.

3.1. Two-stage model definition

The Discrete Model consists of a sparse set of viewing-site, branch, and path data defining the central axes of the major airways. An individual point along an axis will be referred to as a *viewing site*. A set of contiguous viewing sites between an axis end point and branch point or between two branch points will be called a *branch*. A *branch point* is a viewing site where a single branch divides into two or more separate branches. One complete axis, or *path*, consists of a subset of connected branches, where the first and last path branches terminate the path with endpoints. More specifically, the Discrete Model consists of L viewing sites

$$\mathbf{w}_l, \qquad l = 0, 1, \ldots, L-1. \tag{1}$$

M branches

$$\mathbf{b}_m = \{\mathbf{w}_{m,0}, \mathbf{w}_{m,1}, \ldots, \mathbf{w}_{m,\mathrm{END}_m}\}, \qquad m = 0, 1, \ldots, M-1 \tag{2}$$

and N paths

$$\mathbf{p}_n = \{\mathbf{b}_0, \mathbf{b}_{n,1}, \ldots, \mathbf{b}_{n,\mathrm{END}_n}\}, \qquad n = 0, 1, \ldots, N-1, \tag{3}$$

where $L \geq 2$, $M \geq 1$, and $N \geq 1$ (Technically, a tree can consist of only one branch and one path; i.e. $M = N = 1$. In this case, a branch can be terminated by two end points.). Each viewing site $\mathbf{w}_l$ belongs to only one branch unless it is a branch point. The special viewing site $\mathbf{w}_0$, referred to as the *root site*, is picked manually, as discussed in Section 3.3. The root site $\mathbf{w}_0$ starts the tree. Hence, it always starts the first branch $\mathbf{b}_0$ and first path $\mathbf{p}_0$. Per the definition (2), each branch $\mathbf{b}_m$ consists of a set of contiguous viewing sites from set (1). For the 3D chest problem, we assume that branch $\mathbf{b}_0$ corresponds to the trachea and terminates at the main carina. Branch $\mathbf{b}_0$ is a member of all paths. Per (3), a subset of connected branches starting from $\mathbf{b}_0$ and terminating at a branch $\mathbf{b}_{n,\mathrm{END}_n}$ that has an endpoint form a path $\mathbf{p}_n$, where the constituent branches are from branch set (2). In general, branches $\mathbf{b}_m$ can belong to many paths.

For Discrete Model shown in Fig. 1, the complete tree is represented as

$$\mathbf{b}_0 = \{\mathbf{w}_0, \mathbf{w}_1, \mathbf{w}_2\}, \qquad \mathbf{b}_1 = \{\mathbf{w}_2, \mathbf{w}_3\},$$

$$\mathbf{b}_2 = \{\mathbf{w}_2, \mathbf{w}_4, \mathbf{w}_7\}, \qquad \mathbf{b}_3 = \{\mathbf{w}_3, \mathbf{w}_5\},$$

$$\mathbf{b}_4 = \{\mathbf{w}_3, \mathbf{w}_6\}, \qquad \mathbf{p}_0 = \{\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_3\},$$

$$\mathbf{p}_1 = \{\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_4\}, \qquad \mathbf{p}_2 = \{\mathbf{b}_0, \mathbf{b}_2\}.$$

The two viewing sites $\mathbf{w}_2$ and $\mathbf{w}_3$ are branch points.

A viewing site is more completely expressed as $\mathbf{w}_l = \{\mathbf{s}_l, \mathbf{d}_l\}$, where

$$\mathbf{s}_l = [\,x_l \quad y_l \quad z_l\,], \qquad \mathbf{d}_l = [\,d_l^x \quad d_l^y \quad d_l^z\,]. \tag{4}$$

$\mathbf{s}_l$ is the 3D *location* of $\mathbf{w}_l$; $\mathbf{d}_l$ is a 3D unit vector pointing in the viewing *direction* for $\mathbf{w}_l(\|\mathbf{d}_l\| = 1)$; x_l, y_l and z_l are the x, y, and z components of $\mathbf{s}_l$; and d_l^x, d_l^y, and d_l^z are the respective x, y, and z components of $\mathbf{d}_l$. At each viewing site $\mathbf{w}_l$, the viewing direction $\mathbf{d}_l$ defines a local coordinate frame such that the local z-axis is aligned with the axis of the tree at $\mathbf{s}_l$. As we will clarify in Section 3.3, adjacent contiguous viewing sites are not necessarily equally spaced.

The Discrete Model captures essential top-level topological structure of the airway tree in an efficient data structure. To limit redundancy, viewing sites are shared among branches and branches are shared among paths. But the Discrete Model does not contain detailed smooth paths and tree structure. For VB-based navigation through a 3D image, smooth paths are necessary. To meet this requirement, we propose the Generalized Cylinder Model, which builds upon the concept of a generalized cylinder (GC) [34–37].

A generalized cylinder is a generalization of a right cylinder. The axis of a GC is not confined to be a straight line. Instead, it can be an arbitrary open curved trajectory. Also, instead of a circular 2D cross-section, the cross-section of a GC can be an arbitrary closed contour subject to certain continuity constraints. GCs are particularly attractive, since their compact representation permit efficient computer implementation. The GC Model contains detailed structural information for each path. Each Discrete-Model path $\mathbf{p}_n$ is transformed into a GC that is composed of a *spine* (or primary axis) $\mathbf{p}_n(u)$ and a *contour function* $\mathbf{c}_n(t,u)$. With the spine and contour functions, the surface $\mathbf{s}_n(t,u)$ of the GC can be obtained. Fig. 1 depicts an example GC. Cubic B-splines are used to represent the spine and contour functions. For VB navigation, a spine corresponds to a desired smooth navigation path through an airway and the contour function defines the airway wall's endoluminal surface about the spine.

Complete details on the B-spline functions representing the spines and contour functions are given in Section 3.4 and Ref. [38]. Below, we summarize the quantities defining these functions. The spine $\mathbf{p}_n(u)$ is a continuous, open, parameterized curve in $\Re^3$ given by

$$\begin{aligned} &\mathbf{p}_n(u) = [\, x_n(u) \quad y_n(u) \quad z_n(u)\,], \qquad 0 \le u \le u_n^{\max}, \\ &n = 0, 1, \ldots, N-1, \end{aligned} \tag{5}$$

where u is a distance parameter as one moves along the path from beginning to end, and $x_n(u)$, $y_n(u)$, and $z_n(u)$ are B-spline functionals that determine x, y, and z position in 3D image space. Note that each point of the spine (5) also has a direction vector associated with it

$$\mathbf{d}_n(u) = [\, d_n^x(u) \quad d_n^y(u) \quad d_n^z(u)\,], \tag{6}$$

as in the case of the Discrete Model. The direction $\mathbf{d}_n(u)$ is not necessary for defining the spine per se, but it is necessary for virtual-endoscopic navigation and exploration as done for the examples of Section 4.3. As Section 3.3 describes, the viewing sites comprising Discrete-Model path $\mathbf{p}_n$ serve as interpolation points, or *knot points*, for the B-spline $\mathbf{p}_n(u)$. The B-spline representation for the spine permits equispaced sampling (per u or arc length) along the path. At each point u along the spine $\mathbf{p}_n(u)$, the tangent to the spine, $\mathrm{d}\mathbf{p}_n(u)/\mathrm{d}u$, defines the z-axis of a local coordinate frame. In this local coordinate frame, the contour function

$$\mathbf{c}_n(t,u) = [\, x_n^c(t,u) \quad y_n^c(t,u) \quad 0\,], \qquad 0 \le t \le 1 \tag{7}$$

represents the generalized cylinder's orthogonal cross-section, where t is the distance parameter as one moves along the closed contour from beginning to end, u is the position along the nth GC spine $\mathbf{p}_n(u)$, and $x_n^c(t,u)$ and $y_n^c(t,u)$ are the B-spline functionals that determine the x and y position in the 2D plane with respect to the distance parameter t. The contour function is a closed, parameterized B-Spline curve in the local coordinate frame constrained to the 2D plane orthogonal to the path tangent at u. Hence, the z component of $\mathbf{c}_n(t,u) = 0$. Finally, a point on the surface of the nth GC is given by

$$\mathbf{s}_n(t,u) = \mathbf{p}_n(u) + \mathbf{c}_n(t,u)\mathbf{R}_n(u), \tag{8}$$

where $\mathbf{R}_n(u)$ is a 3×3 rotation matrix that rotates the global 3D image z-axis to the z-axis of the GC's local coordinate frame at point $\mathbf{p}_n(u)$; i.e. this rotation is given by the relationship between the global z-axis and $\mathrm{d}\mathbf{p}_n(u)/\mathrm{d}u$.

3.2. Two-stage algorithm summary

The two-stage algorithm is summarized below. Complete details on the steps below are discussed fully in Sections 3.3 and 3.4 to follow.

Stage 1: discrete-model calculation

1. Starting with a given 3D 16-bit chest image V, the user specifies the root site $\mathbf{w}_0 = \{\mathbf{s}_0, \mathbf{d}_0\}$, somewhere near the proximal end of the trachea.
2. Use $\mathbf{w}_0$ to begin a queue of pending viewing sites. Start branch $\mathbf{b}_0$ with $\mathbf{w}_0$, and start path $\mathbf{p}_0$ with branch $\mathbf{b}_0$.
3. Pick a viewing site $\mathbf{w}_l$ from the queue.
4. Perform a local 2D oblique-slice analysis at $\mathbf{w}_l$ to estimate the airway structure's cross-section. If the analysis passes a set of stopping criteria, then $\mathbf{w}_l$ corresponds to an endpoint; this terminates the current branch and path-go to step 6. Otherwise, use the centroid of the extracted cross-section as the refined viewing-site location $\mathbf{s}_l$ for $\mathbf{w}_l$.
5. Perform a 3D spherical search about the refined viewing site $\mathbf{w}_l$ to locate new viewing sites to analyze. If new viewing sites are found, then add them to the queue.
6. If either the cross-sectional analysis or spherical search reveals that $\mathbf{w}_l$ corresponds to an endpoint or branch point, then appropriately update the active branch and path information.
7. If the queue is not empty, return to step 3. Otherwise, the recorded viewing site, branch, and path information defines the final Discrete Model, per (1–3).

Stage 2: generalized-cylinder model calculation

1. For each path $\mathbf{p}_n$ in the Discrete Model data, $n = 0, 1, \ldots, N-1$, compute the spine and contour functions of the associated generalized cylinder:
   (a) Perform a B-spline analysis to compute the spine $\mathbf{p}_n(u)$.
   (b) At equally spaced samples $\hat{u} = 0, u_1, u_2, \ldots, u_n^{\max}$, along the spine, compute the contour function $\mathbf{c}_n(t, \hat{u})$ using a B-spline analysis. Also, compute the GC's cross-sectional area at $\hat{u}$.
2. Construct a segmented airway-tree image by merging the surface functions (8) for all GCs in the GC Model. Also, compute endoluminal cross-sectional area measurements of the tree.

The final outputs are the smooth paths $\mathbf{p}_n(u)$, $n = 0, 1, \ldots, N-1$, cross-sectional area measurements of the airway tree along the paths, the basic viewing-site/branch/path connectivity relationships per (1–3), and the segmented airway tree.

3.3. Stage-1 details

This section gives complete detail for the Stage-1 Discrete Model calculation. The step numbers below refer to the steps outlined in Section 3.2.

Step 1: specify root site

The root site $\mathbf{w}_0$ is specified manually by examining the 3D image data in an appropriate image viewer. We have used virtual-endoscopy systems built by our group for PCs and Sun workstations for this purpose [8,15]. This site is easily picked by locating any point near the proximal end ('top') of the trachea. This point need not be precisely located in the center of a tracheal cross-section, since the two-stage algorithm 'straightens out' the trajectory of travel.

Step 4: 2D oblique-slice analysis

First, a 2D slice of data that is centered about $\mathbf{s}_l$ and orthogonal to $\mathbf{d}_l$ is sampled from the given 3D image V. This is depicted in Fig. 2. Next, a set of equally spaced

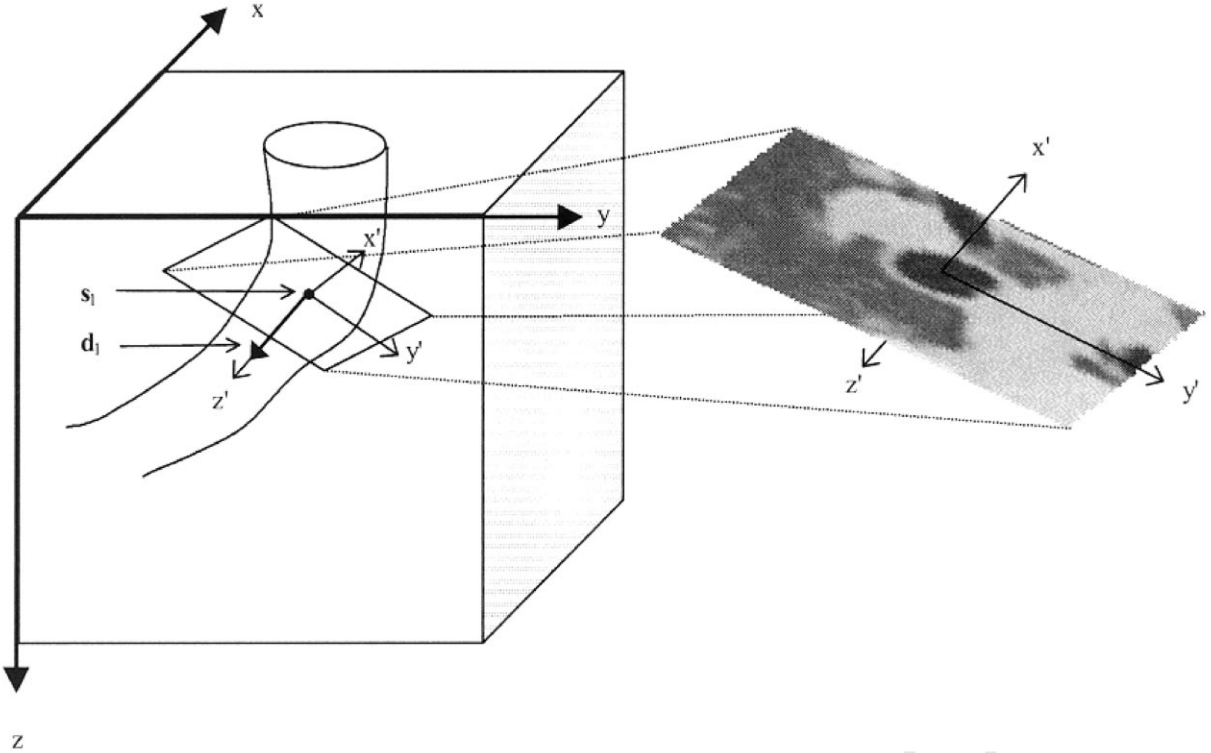

Fig. 2. 2D oblique slice sampled from V about viewing site $\mathbf{w}_l$. Local coordinate frame (x', y', z') of the slice is defined by location $\mathbf{s}_l$ and direction vector $\mathbf{d}_l$. Direction vector $\mathbf{d}_l$ is aligned with the airway axis and the z axis (z') of the local coordinate frame. The left side of the figure shows the 3D image's global (x, y, z) coordinate frame.

rays perpendicular to $\mathbf{d}_l$ are cast radially outward from $\mathbf{s}_l$ on this slice. Points encountered by a ray are computed by using 3D nearest-neighbor interpolation on the original 3D image. In our results, we cast 16 rays. If a ray strikes an image point at distance r_b having image value $f(r_\mathrm{b})$ that is greater than a prescribed threshold, then this marks a point on the airway lumen's boundary; see Fig. 3. The complete set of boundary contour points provide an initial estimate of the airway lumen's boundary at viewing site $\mathbf{w}_l$. In a typical CT chest image 'air' values are near -1000 HU versus ap; $+100$ HU (or greater) for an airway wall. Hence, picking a threshold tends to be straightforward. We have also used a gradient-based technique and a local half-maximum approach to find where a ray strikes the airway lumen, but have not observed these methods to offer appreciable differences in the path calculation [17,38].

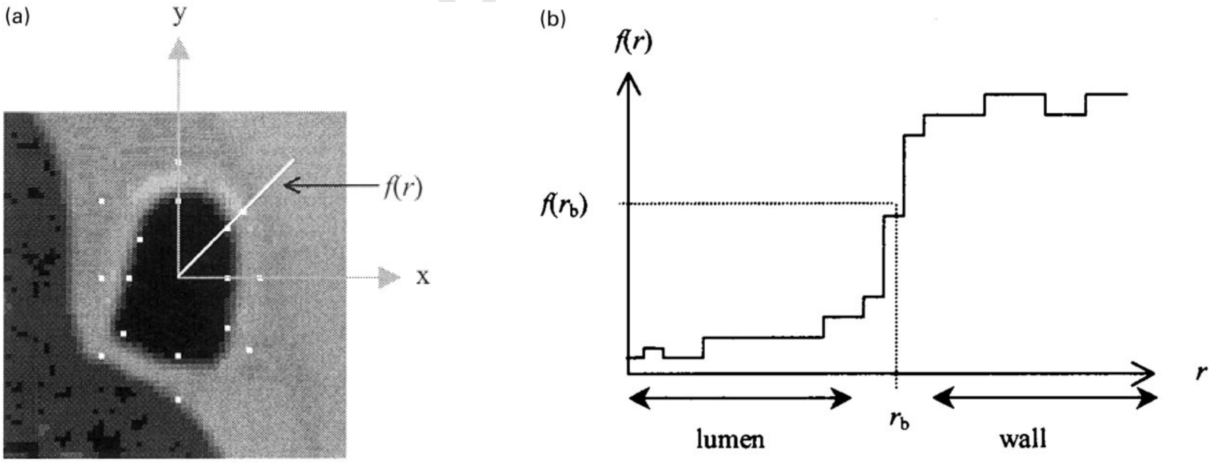

Fig. 3. Contour boundary finding. Rays are cast out from $\mathbf{s}_l$ on the oblique slice. When a ray strikes a point r_b that has a gray-level value $f(r_b)$ above a preset threshold, then this point specifies a endoluminal boundary point. For 3D CT data, the air appears black (near -1000 HU) and the brighter wall is generally above 100 HU. (a) 2D oblique cross section sampled at $\mathbf{s}_l$. (b) Gray-scale image profile $f(r)$ along a ray.

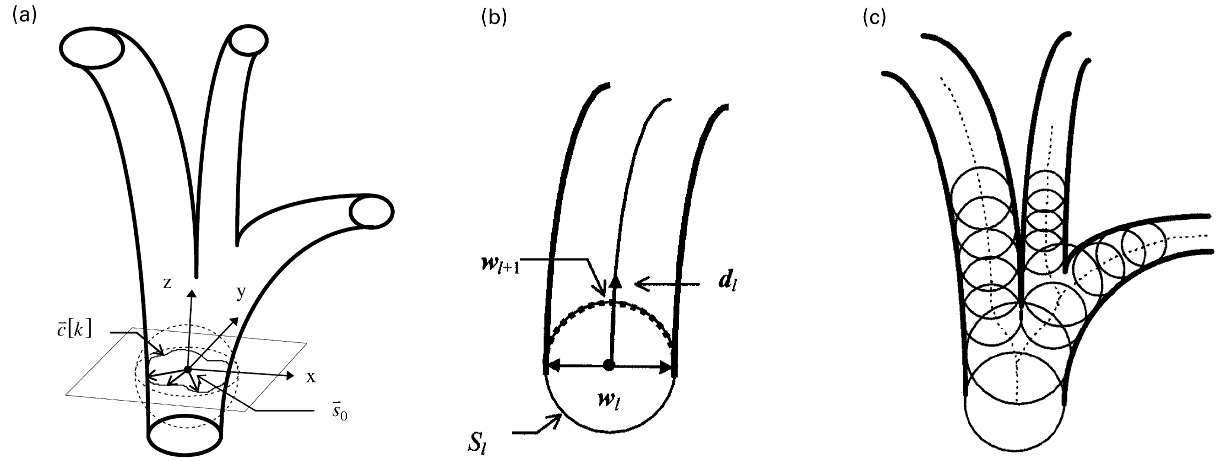

Fig. 4. Construction of the 3D tessellated spherical operator S_l about the current viewing site w_l. As described in the text, "air" regions (contours C_s) passing through predefined tessellated points on the sphere's surface are located to find new viewing sites to examine along the tree. (a) Construction of the spherical operator. (b) Side view of 3D search about $\mathbf{w}_{l+1}$ A new viewing site is found. (c) 2D side view of evolving tree using a series of spheres.

Two stopping criteria determine whether a particular viewing site is an endpoint: if the computed airway contour (a) has a radius below a prescribed minimum or (b) has a shape that is too far from circular (measured by computing the standard deviation of the ray lengths). We set the minimum radius to be 1/2 the coarsest spatial resolution dimension (usually the slice thickness Δz). We vary the standard deviation parameter depending on the operating conditions. The standard-deviation stopping criteria detects situations where a contour has a sudden break in it and leaks into the parenchyma; this happens when the airway wall becomes very thin or the airway becomes very small. If the viewing site $\mathbf{w}_l$ passes either of these stopping criteria, it is deemed to be an endpoint—this terminates the currently active branch and path. Otherwise, the airway is sufficiently large and has a suitable shape. Thus, the analysis of this viewing site continues onto step 5. The centroid of the contour is used as a modified estimate of the viewing site's location $\mathbf{s}_l$ Also, the minimum radius r_l of a circle that fully encloses the contour is calculated.

Step 5: 3D spherical operator search

The goal of this step is to perform a 3D search for airways emanating from the currently active viewing site. This is done by finding where air patches, which correspond to evolving airways, intersect the surface of an appropriately constructed tessellated 3D sphere situated about $\mathbf{s}_l$. The intersecting spherical air patches indicate new regions to move to and help define new viewing sites to examine. The discussion below describes the construction and use of the sphere.

The modified viewing-site location $\mathbf{s}_l$ and minimum radius r_l from step 4 are used to define a 3D tessellated spherical search operator S_l. The radius of this sphere is given by $R_s = r_l + \delta r$, where $\delta r > 0$ is the minimum detectable airway wall thickness. We choose δr so that it equals the minimum spatial resolution of the image (typically Δx or Δy). By its construction, the spherical operator S_l will be just large enough to completely contain the cross-sectional contour at $\mathbf{w}_l$. At the same time, the sphere is small enough so that all airway branches emanating from $\mathbf{w}_l$ can be detected. The resulting sphere is shown by the dashed lines in Fig. 4a. Note that the sphere is constructed to cover approximately a spherical volume in space. This gives isotropic search coverage to all directions. But the actual sampled form of the sphere is typically an ellipsoid, since the slice thickness Δz tends to be larger than the transverse-plane sample spacings, Δx and Δy.

In order to approximate a spherical shell, we use a tessellated spheroid generation technique developed by Paeth [39]. This results in a unit spheroid consisting of a list of vertices and normal vectors that form small triangles to approximate the surface of a sphere. Fig. 5a depicts the top view of a spheroid of depth 3 with vertices numbered counterclockwise out from the center. The depth corresponds to the factor of angular division within one octant of the sphere. The radius of the spheroid can be adjusted by scaling the vertex locations. Paeth gives complete detail on this structure [39, p. 179–90].

All airway branches emanating from the volume covered by the spherical operator S_l will form closed 'air' patches C_s at the intersections of airway walls and the sphere's surface. See Fig. 5b. Detecting airway branches, and hence new viewing sites to search, then amounts to finding all connected spherical surface patches that intersect air regions C_s. If the 3D image point situated at a sphere vertex P_1 is below the acceptable threshold for air (as used for the 2D oblique-slice analysis), then this vertex is deemed to be contained by air patch C_s. All connected sphere vertices intersecting C_s are deemed to constitute C_s. Rather than the traditional notions of 4- and 6-connectedness of a 2D pixel set, we perform connected-component analysis on the *surface of a sphere*. For our implemented case of a tessellated approximation to a sphere, we treat the vertices as a grid of points and the line segments connecting them (face

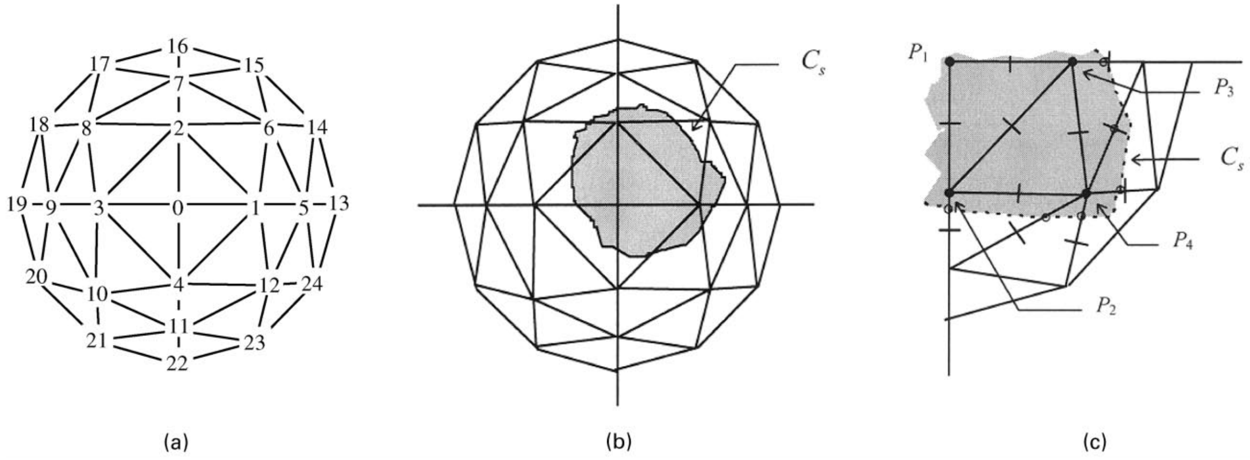

Fig. 5. Form of tessellated spherical operator S_l. A tessellated sphere with depth $n = 3$ is depected. (a) Top view of spherical operator's tessellation. (b) Top view of sphere impinging on contour C_s. (c) Detail of tessellated sphere about contour C_s.

edges) as the connectivity paths. See Fig. 5c; vertices P_1, P_2, P_4, and P_3 are found to be a connected cluster of sphere vertices passing through C_s. At least three connected vertices must be found to justify a valid cluster. Given the centroid c_s of a cluster, c_s specifies the location of a new viewing site to search and the vector $(\mathbf{w}_l - c_s)$ specifies this new site's direction. Note, that neither a viewing site's location $\mathbf{s}_l$ or direction $\mathbf{d}_l$ is confined to integer coordinates.

The search is actually done over a hemisphere of directions situated about $\mathbf{s}_l$ even though the entire tessellated sphere can be necessary for constructing the surface patches, with the orientation of the hemisphere determined by the current direction $\mathbf{d}_l$. The spherical operator's 'snug' construction about $\mathbf{w}_l$ guarantees that all evolving airway branches are detected by the operator. A new viewing site $\mathbf{w}_{l+1}$ is always situated at a distance R_s from $\mathbf{w}_l$, as depicted in Fig. 4b; i.e. the search moves forward as far as possible and connected sites are not necessarily equidistant. Fig. 4c illustrates a 2D side view of how the 3D spherical search evolves during the Discrete Model's construction.

The sphere's depth is updated for each examined viewing site and is determined as follows. First, we assume that the smallest airway endoluminal structure that can be detected must have a radius $r_{\min}$ of at least 1/2 the slice thickness Δz. Since Δz is generally greater than the in-plane resolutions, Δx and Δy, the worst case is for a horizontally oriented airway in the x–y plane of the scanned image that is visible on only a single slice. We define h as the distance from the sphere center to the midpoint of a chord connecting two neighboring vertices and e as the maximal deviation of the chord from a true sphere of radius R_s (Fig. 6a). Note that triangle facets tessellating the sphere vary in shape slightly across the sphere's surface. The worst case is when a given facet is an equilateral triangle with sides of length b (the maximal chord length) and when all three of its vertices are precisely on the inner wall of an airway's cross-section. If the position of the triangle is translated at all or any of the triangle's sides are shortened, then one or more vertices must fall outside the airway's interior and hence result in a missed detection. The sphere's depth must be large enough to prevent this occurrence. Fig. 6b depicts this situation. Combining the

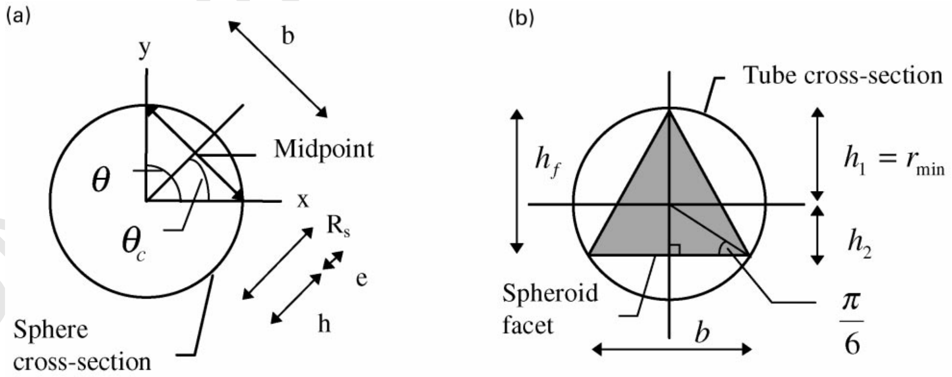

Fig. 6. Determing spheroid depth based on chord length between vertices. (a) Cross section of spherical operator. (b) Single triangular "facet" aligned with airway cross section.

relationships between b, $r_{\min}$, and spheroid radius R_s gives the required depth [38]:

$$\text{Sphere depth} = \frac{\pi}{4}\left(\sin^{-1}\left(\frac{r_{\min(1+\sin\frac{\pi}{6})}}{2R_s\, cos\frac{\pi}{6}}\right)\right)^{-1}. \tag{9}$$

This spheroid will then be guaranteed to have a tesselated surface with a fine enough resolution to detect any subsequent branching airway points subject to the minimum allowable size.

Step 6: update branch and path relationships

At the end of the analyses of Steps 4 and 5, the Discrete Model requires updating. This requires some straightforward book-keeping, briefly highlighted below. Three cases arise as a result of the current viewing site's analysis:

- If the current viewing site passes the stopping criteria (step 4) or if the spherical search finds no subsequent viewing sites, then $\mathbf{w}_l$ is an endpoint: terminate the current branch and path.
- If one new viewing site is found, append it to the current branch.
- If two (or more) viewing sites are found, then $\mathbf{w}_l$ corresponds to a branch point. Terminate the current branch and generate new branches for each new viewing site. The first viewing site of each new branch is the current viewing site $\mathbf{w}_l$, and the second viewing site is the respective newly found viewing site. The connectivity between the current branch and the new branches is recorded in the Discrete Model. Also, the currently active path is continued along one of the new branches, while a new path is initiated using the other new viewing site.

3.4. Stage-2 Details

This section gives more detail for the two steps of Stage 2. A generalized cylinder is computed for each Discrete-Model path $\mathbf{p}_n$. As stated earlier, the GC requires a spine function $\mathbf{p}_n(u)$ and a contour function $c_n(t,u)$. Both of these quantities are computed using well-known cubic uniform B-splines [40]. Our GC-Model development adapts the efforts of Shani et al. to our 3D tree problem [35].

Step 1(a): spine calculation

The spine $\mathbf{p}_n(u)$ of the nth GC is derived from $\mathbf{p}_n$ and is represented as an open cubic uniform B-spline function. The basic steps of this calculation are as follows: (a) the viewing sites $\mathbf{w}_l$ along $\mathbf{p}_n$ serve as *knot points* $\mathbf{s}_l$; (b) the knot points $\mathbf{s}_l$ are used to compute *control points* V_l; (c) the path's B-spline representation $\mathbf{p}_n(u)$ is made up of a set of piecewise connected *span* functions $C_i(u)$, which are a function of the control points V_l. All essential detail for these calculations appears below.

Consider a Discrete-Model path $\mathbf{p}_n$ made up of P viewing sites

$$\text{p}_n = \{\mathbf{w}_0\ \mathbf{w}_1\ \mathbf{w}_2\ \mathbf{w}_3\ \mathbf{w}_4\ \mathbf{w}_5\ \mathbf{w}_6\ \mathbf{w}_7\ \ldots\ \mathbf{w}_{P-2}\ \mathbf{w}_{P-1}\}.. \tag{10}$$

The viewing-site locations of the path, written as a P element vector

$$\mathbf{S} = \{\mathbf{s}_0\ \mathbf{s}_1\ \mathbf{s}_2\ \mathbf{s}_3\ \mathbf{s}_4\ \mathbf{s}_5\ \mathbf{s}_6\ \mathbf{s}_7\ \ldots\ \mathbf{s}_{P-2}\ \mathbf{s}_{P-1}\}. \tag{11}$$

serve as *knot points* or *interpolation points* for the spine $\mathbf{p}_n(u)$ (We drop the subscript n for much of this discussion to simplify notation.). The spine $\mathbf{p}_n(u)$ is guaranteed to pass through these points in 3D space. A set of $P+2$ *control points* V_i, $i = 0, 1, \ldots, P+1$, are derived from the P knot points, where each V_i is a point in 3D space. Each consecutive group of four control points

$$\bar{V}_i = [V_{i-1}\ \ V_i\ \ V_{i+1}\ \ V_{i+2}]^{\mathrm{T}}, \qquad i = 1, 2, \ldots, P-1 \tag{12}$$

are used to form a series of $P-1$ third-order continuous *span* polynomials $C_i(u)$, where

$$C_i(u) = [\,u^3\quad u^2\quad u\quad 1\,][\mathbf{C}]\bar{V}_i \tag{13}$$

and the 4×4 matrix $\mathbf{C}$ is given by

$$\mathbf{C} = \frac{1}{6}\begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{bmatrix}. \tag{14}$$

So, for $0 \le u < 1$

$$\mathbf{p}_n(u) = C_i(u'), \tag{15}$$

where

$$i = \text{int}\{u(P-1)\} + 1, \qquad i = 1, 2, \ldots, P-1 \tag{16}$$

and

$$u' = (u(P-1)) - i + 1, \tag{17}$$

where int(a) represents the integer part of a.

The only detail remaining is the computation of the control points. The knot points are related to the control points through the following matrix relationship:

$$\mathbf{S} = \mathbf{D}\mathbf{V}, \tag{18}$$

where $\mathbf{V}$ is a P-element subset of the control points given by the vector

$$\mathbf{V} = [\,V_1\quad V_2\quad V_3\quad V_4\quad V_5\quad \ldots\quad V_P\,]^{\mathrm{T}},$$

and, for third-order B-splines, the $P \times P$ matrix $\mathbf{D}$ is given by the well-known relation

$$D=\frac{1}{6}\begin{bmatrix}6&0&0&0&\cdots&&&0\\1&4&1&0&&&&0\\0&1&4&1&&&&0\\0&0&1&4&&&&0\\\vdots&&&&\ddots&&&\vdots\\&&&&&1&4&1\\0&0&0&0&\cdots&0&0&6\end{bmatrix}=\frac{1}{6}\begin{bmatrix}a_{1,1}&a_{1,2}&0&0&\cdots&&&0\\a_{2,1}&a_{2,2}&a_{2,3}&0&&&&0\\0&a_{3,2}&a_{3,3}&a_{3,4}&&&&0\\0&0&a_{4,3}&a_{4,4}&&&&0\\\vdots&&&&\ddots&&&\vdots\\&&&&&a_{P-1,P-2}&a_{P-1,P-1}&a_{P-1,P}\\0&0&0&0&\cdots&0&0&a_{P,P}\end{bmatrix}. \tag{19}$$

(The terminating control points V_0 and V_{P+1} are discussed below) **D** can undergo L–U decomposition:

$\mathbf{D}=\mathbf{LU}$.

Since **D** is tridiagonal, both the lower triangular matrix **L** and the upper triangular matrix **U** are bidiagonal:

$$\mathbf{L}=\frac{1}{6}\begin{bmatrix}1&0&0&0&\cdots&&&0\\l_1&1&0&0&&&&0\\0&l_2&1&0&&&&0\\0&0&l_3&1&&&&0\\\vdots&&&&\ddots&&&\vdots\\&&&&&l_{P-2}&1&0\\0&0&0&0&\cdots&0&l_{P-1}&1\end{bmatrix},$$

$$\mathbf{U}=\begin{bmatrix}u_1&w_1&0&0&\cdots&&&0\\0&u_2&w_2&0&&&&0\\0&0&u_3&w_3&&&&0\\0&0&0&u_4&&&&0\\\vdots&&&&\ddots&&&\vdots\\&&&&&0&u_{P-1}&w_{P-1}\\0&0&0&0&\cdots&0&0&u_P\end{bmatrix}. \tag{20}$$

The sparse forms of the matrices above permit efficient calculation of the control points via the following simple recurrence relations. Let

$$y_1=\mathbf{s}_0. \tag{21}$$

From Eqs. (19) and (20),

$$u_1=a_{1,1} \tag{22}$$

and, for $i=2,\ldots,P$,

$$y_i-\mathbf{s}_{i-1}-l_{i-1}y_{i-1}, \tag{23}$$

where

$$l_{i-1}=\frac{u_{i-1}}{a_{i,i-1}}, \tag{24}$$

$$u_i=a_{i,i}-l_{i-1}w_{i-1}, \tag{25}$$

$$w_i=a_{i,i+1}. \tag{26}$$

Given the values of y_i, w_i, and u_i, $i=1,2,\ldots,P$, we can now get the control points:

$$V_P=\frac{y_P}{u_P}, \tag{27}$$

$$V_i=\frac{y_i-w_iV_{i+1}}{u_i},\qquad i=P-1,\ldots,1. \tag{28}$$

Finally, the terminating control points are given by the following and guarantee that the local curvature of the spine equals 0 at its ends:

$$V_0=2V_1-V_2,\qquad V_{P+1}=2V_P-V_{P-1}. \tag{29}$$

Thus, overall, to compute $\mathbf{p}_n(u)$, the following procedure is followed: (a) use Eqs. (10) and (11) to set up the knot points; (b) use the starting condition (21) and relations (19), (20) and (22)–(26) to compute the intermediate quantities y_i, u_i, and w_i, and Eqs. (27)–(29) to get the control points; (c) apply Eqs. (13)–(17). Finally, we point out that direction vectors $\mathbf{d}_n(u)$, Eq. (6), can be computed for each spine $\mathbf{p}_n(u)$, using the same procedure as above by using the $\mathbf{d}_n$ of Eq. (4) instead of the $\mathbf{s}_n$.

Step 1(b): contour-function calculation

The calculation of the contour functions again uses B-spline analysis as used for the spine calculations. For each value $\hat{u}$ along the nth GC's spine $\mathbf{p}_n(\hat{u})$, the following steps give the contour function $\mathbf{c}_n(t,\hat{u})$:

1. Find a set of $P-1$ contour points defining the endoluminal surface at $\mathbf{p}_n(\hat{u})$. This is done using the same ray-casting technique of Stage 1, step 4, with $P-1=16$. This gives a set of points c_i, $i=0,1,\ldots,P-2$, that serve as knot points in the spline calculations. These points, and other calculations, involve 2D calculations in the local coordinate frame for $\mathbf{c}_n(t,\hat{u})$. To define a complete closed contour consistent with the needs of the spline calculations, we add the point $C_{P-1}=C_0$ to give the vector of knot points

$$\mathbf{S}_c=[\,c_0\quad c_1\quad c_2\quad c_3\quad c_4\quad \ldots\quad c_{P-2}\quad c_0\,]^{\mathrm{T}}.$$

2. To find the B-spline representation, we use the same

development as for the spine calculation, but make the following substitutions:

(a) Vector $\mathbf{S}_c$ takes the place of $\mathbf{S}$ in Eq. (18).

(b) The $P \times P$ matrix $\mathbf{D}$ in Eq. (18) is replaced by the $(P-2) \times (P-2)$ matrix $\mathbf{D}_c$, where

$$\mathbf{D}_c = \frac{1}{6}\left[\begin{array}{cccccccc} 4 & 1 & 0 & 0 & \cdots & & & 1 \\ 1 & 4 & 1 & 0 & & & & 0 \\ 0 & 1 & 4 & 1 & & & & 0 \\ 0 & 0 & 1 & 4 & & & & 0 \\ \vdots & & & & & \ddots & & \vdots \\ & & & & & 1 & 4 & 1 \\ 1 & 0 & 0 & 0 & \cdots & 0 & 1 & 4 \end{array}\right].$$

This form accounts for the overlapping endpoints at indices 0 and $P-1$.

(c) The control-point end conditions (29) are replaced by

$$V_p = V_0, \qquad V_{P+1} = V_1.$$

(d) $\mathrm{P}-1$ spans again are used to represent the closed contour, but they are parameterized on t instead of u. Thus, for $0 \le t < 1$,

$$\mathbf{c}_n(t, \hat{u}) = C_i(t'),$$

where

$$i = \mathrm{int}\{t(P-1)\} + 1, \qquad i = 1, 2, \ldots, P-1$$

and

$$t' = (t(P-1)) - i + 1.$$

The calculation of the contour functions is then done as for the spine functions, with the obvious substitutions (e.g. c_0 is substituted for $\mathbf{s}_0$ in Eq. (21), etc.).

Step 2: construct segmented tree

After the spine and contour functions are computed, a segmented image is readily generated by applying Eq. (8). The image is built up by combining the GCs for all paths $n = 0, 1, \ldots, N-1$, and sample values $\hat{u} = 0, u_1, u_2, \ldots, u_n^{\max}$. At each sampled value $\hat{u}$, the contour function $\mathbf{c}_n(t, \hat{u})$ is computed. The voxels covered by the contour are included in the segmented image (nearest-neighbor interpolation used). Also, endoluminal cross-sectional area calculations are readily done using the relation

$$\mathrm{area}\{\mathbf{c}_n(t, \hat{u})\} = \frac{\sum_{k=0}^{P-3} (c_k \times c_{k+1}) + (c_{P-2} \times c_0)}{2}, \qquad (30)$$

where '×' is for vector cross-product [39]. The Eq. (30) used for the area calculation for $\mathbf{c}_n(t, \hat{u})$ is based on the $(P-1)$-sided polygon defined by the points c_i, $i = 0, \ldots, P-2$. The polygon can be looked upon as enclosing $(P-1)$ triangles, each of which contributes an area term to Eq. (30).

4. Results

The axes-generation algorithm has been tested on a wide range of data. In this section we present phantom and human results. We also give applications of the algorithm to virtual-bronchoscopic assessment of human 3D CT pulmonary images.

4.1. Phantom results

We created a computer-generated phantom consisting of a single tube that branches into two tubes. Voxels located inside the wall of either of two analytically defined curved tubes are set to 127 (8-bit gray-scale) and background voxels are set to zero. These mathematically generated tubes have circular curvature (i.e. they each form one quadrant of a circle when viewed from the side) with a radius of 64 voxels. Assuming that each voxel is an isotropic cube of dimensions $(1\ \mathrm{mm})^3$, then the radius is 64 mm and the image dimensions are 128 mm × 128 mm × 128 mm. The inner radius of the each tube's cross-section (lumen) is 10 mm and the outer radius is 15 mm. To give a noisy image, we applied a window-average filter using a 3 × 3 × 3 kernel and then added Gaussian white noise ($\sigma = 10$ gray levels).

No operator intervention was required for finding an appropriate seed point $\mathbf{w}_0$, since the mathematical center is voxel location (64,64,1) where the two tubes overlap. Fig. 7a shows sagittal and coronal weighted-sum projections of the phantom image along with automatically-generated paths. Fig. 7b shows cross-sectional area measurements of each of the two paths along with the true cross-sectional area of a single tube. Note the fluctuation in cross-sectional area near the bifurcation point at distance 32 mm. This is expected because the cross-sectional area increases as the two tubes begin to diverge until they no longer overlap. After the bifurcation point is passed (distance = 45 mm), the cross-section estimates are for the non-overlapping portions of each tube, and the area measurement remains within a few percent of the known 314.15 mm^2 (dashed line) for each of the tubes (black and white lines).

To test the method for anisotropically sampled image data, we down-sampled the same phantom in z by a factor of 5 resulting in voxel dimensions of 1 mm × 1 mm × 5 mm. This was accomplished by averaging the gray-scale values at a given x–y coordinate over 5 slices to form a single slice. Then, we added noise as before. Fig. 7c,d shows the results of the algorithm using this image. Despite the far lower resolution along the z-axis (note the stair-step effect in the projection images), the path tracking performed well. As expected, area estimation degraded due to the

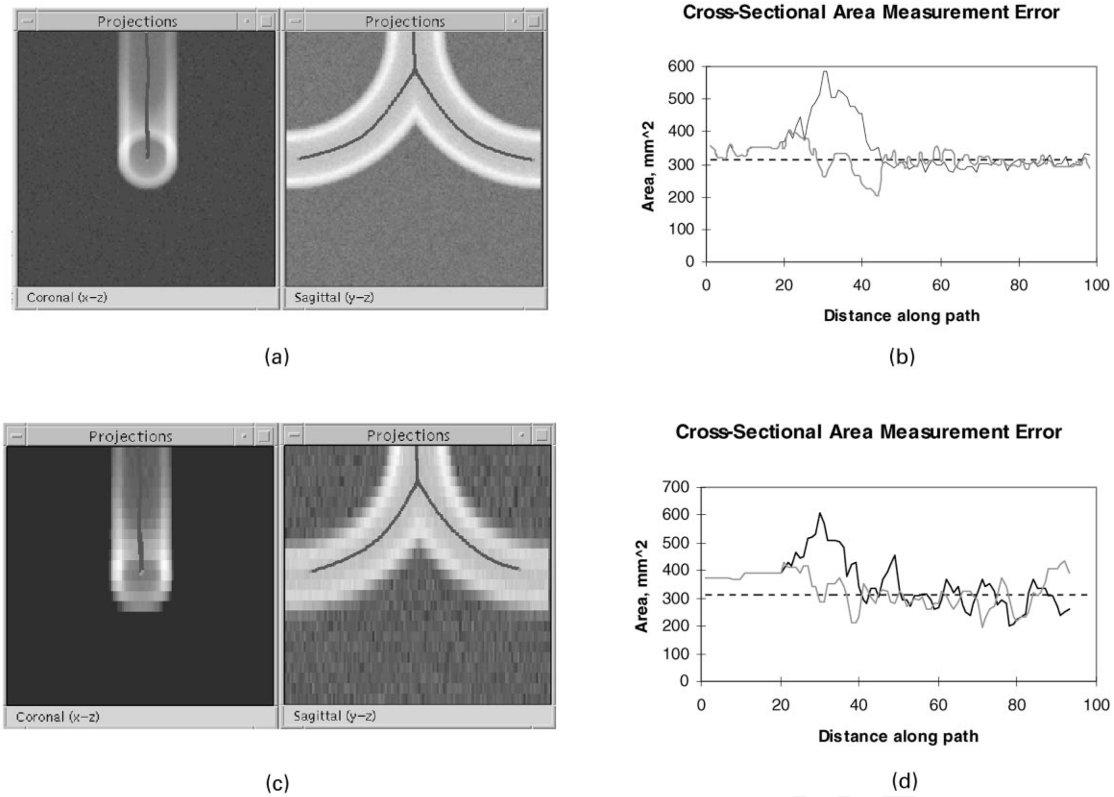

Fig. 7. Computer-Generated Branching Tubular Phantom. Top row uses an isotropically sampled 3D image; i.e., $\Delta x = \Delta y = \Delta z$. Bottom row uses an anistrophically sampled phantom; i.e., $\Delta x = \Delta y = 5\Delta z$. Part (a): extracted 3D axes are the dark lines. Part (b) Computed cross-sectional areas for the two separate branches closely follow the ideal value from the exact analytically defined tube. Part (c): extracted 3D axes are the dark lines. Part (d): Computed cross-sectional areas for the two branches still follow the ideal value from the exact analytically defined tube, despite the high down-sampling factor in the digitized image data. (a) Weighted-Sum Projection Images of Tubular Phantom. (b) Cross-Sectional areas for two branches. (c) Projection Images of Anisotropic Tubular Phantom. (d) Cross-Sectional areas for two branches.

coarse z-resolution. Exhaustive phantom results for tubes of other sampling factors and for cone phantoms, given in [38], confirm the efficacy of the algorithm.

4.2. Human studies

We next validated the methodology on a series of human cases. These cases had earlier undergone standard 3D EBCT (electron beam CT) scanning. They all involved upper-airway obstructions, such as a stenosed airway from an impinging cancer or a collapsed/compressed lung. All 3D CT images consisted of 512×512 transverse-plane slices, with a slice thickness $\Delta z = 3.0$ mm. The number of slices in the cases ranged from 24 to 80; scans were done to focus on the region of interest. The transverse-plane resolution ranged from $\Delta x = \Delta y = 0.410$ mm to 0.684 mm (One case involved 230 0.531 mm-thick slices, with $\Delta x = \Delta y = 0.531$ mm.).

We selected thirteen cases that had previously undergone prior manual path definition using the VIDA 3D analysis software [17,41]. The previous analyses done for these cases involved a skilled technician manually segmented one airway of interest. Axis measurements were next obtained using a package called TGA (Tube Geometry Analysis) contained in VIDA [41]. The basic steps performed in this analysis are as follows: (1) manually segment the airway of interest in all CT slices; (2) perform a shape-based interpolation of the segmented airway, to give a higher-resolution form of the segmented airway; (3) perform TGA analysis to semi-automatically find a central axis through the segmented airway. These steps required about 2 h of user intervention for a typical case.

To compare results, we applied our proposed algorithm using the starting point of the manual analysis as the root site $\mathbf{w}_0$. Generation of the output results required roughly two minutes of computer processing per case (700 MHz Pentium-III PC used). Fig. 8 summarizes the results. These results focus on the differences between the central axes defined manually and automatically. Note the strong agreement between the results. We point out that the manually generated results, even though they were made painstakingly, are not ‘perfect’ ground truth. This is because manual segmentation, shape-based interpolation, and the semi-automatic path extraction method all introduce errors. If anything, the automated results are more reliable. Fig. 9 shows pictorial and numerical results for a typical human case in this study.

	coordinate difference	normalized coordinate difference
mean	2.21	0.37
std. dev.	0.38	0.02

Fig. 8. Comparison of axis accuracy between manual and automatic methods. Thirteen 3D human CT cases, previously analyzed manually using VIDA are included in the study. "Coordinate Difference" refers to mean 3D image coordinate difference between a point on a manually defined axis and the corresponding automatically computed axis (1 unit = 1 voxel). "Normalized Coordinate Difference" is computed by normalizing the coordinate difference by the average diameter of the manually defined axis (1 unit = 1 diameter unit). The average airway diameters for the cases ranged from 3.42mm (5.0 voxels) to 12.31mm (9 voxels). The standard deviation ("std. dev." is computed for these two measures over the 13-case database.

4.3. Application to virtual bronchoscopy

We have devised a PC-based system for virtual-bronchoscopic (VB) assessment of high-resolution 3D CT chest images. The system permits a user to build a multimedia HTML case report for a given 3D image. The basic steps in this evaluation are as follows:

1. Begin a case study by selecting a root site for the airway tree.
2. Run the automated axes-generation algorithm to extract the main airway axes, airway cross-sectional area values, and segmented airway tree.
3. Use the system's many visual tools to peruse regions along the extracted axes. During this analysis, interesting snapshots and movies can be saved.

The details of this system are described elsewhere [15,16]. It is well-acknowledged that precomputed guidance data are

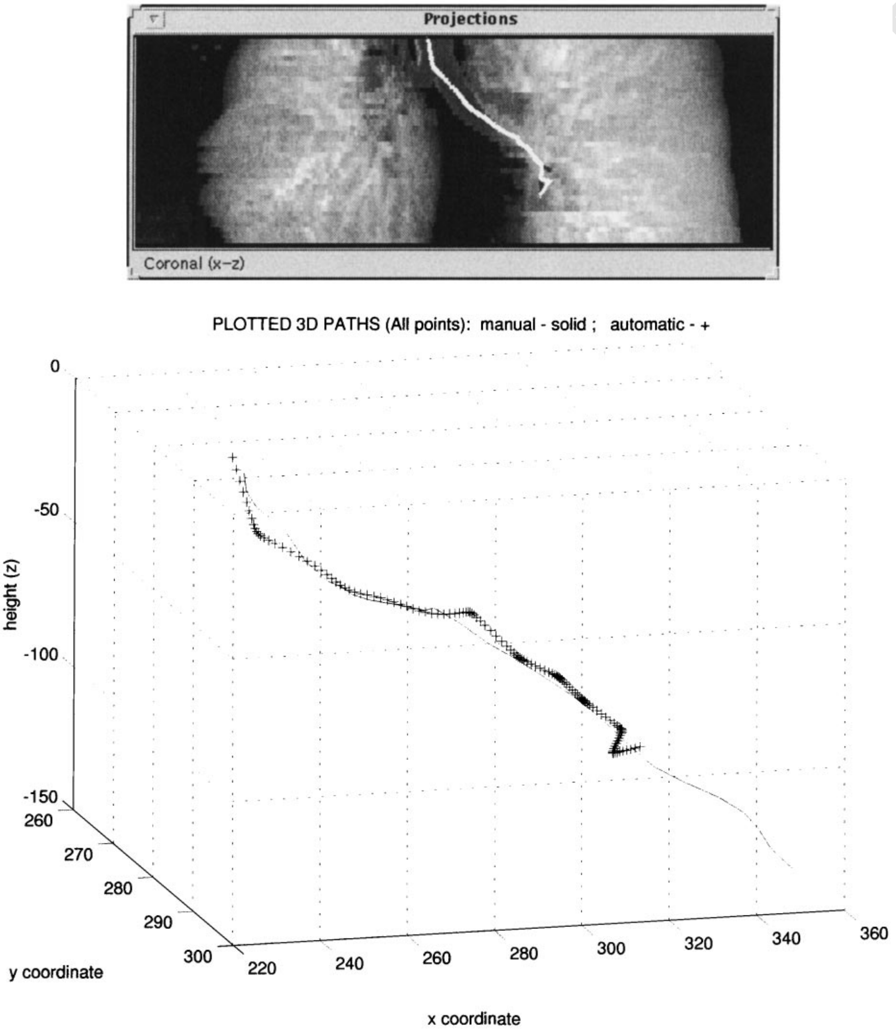

Fig. 9. Manual verus automatic axis generation for a specific human case. The 3D EBCT image consisted of 40 512 × 512 3.0mm-thick slices with $\Delta x = \Delta y = 0.684$mm. Average coordinate difference = 1.49; normalized coordinate difference = 0.138. Top: Coronal maximum-intensity projection showing automatically computed axis. The axis consists of 123 sites. An automated segmentation approach was used to remove the bones and mediastinal structures form the data before the projection was computed [8]. Bottom: 3D view of manual (solid line) and automatically-computed (+ line) axes.

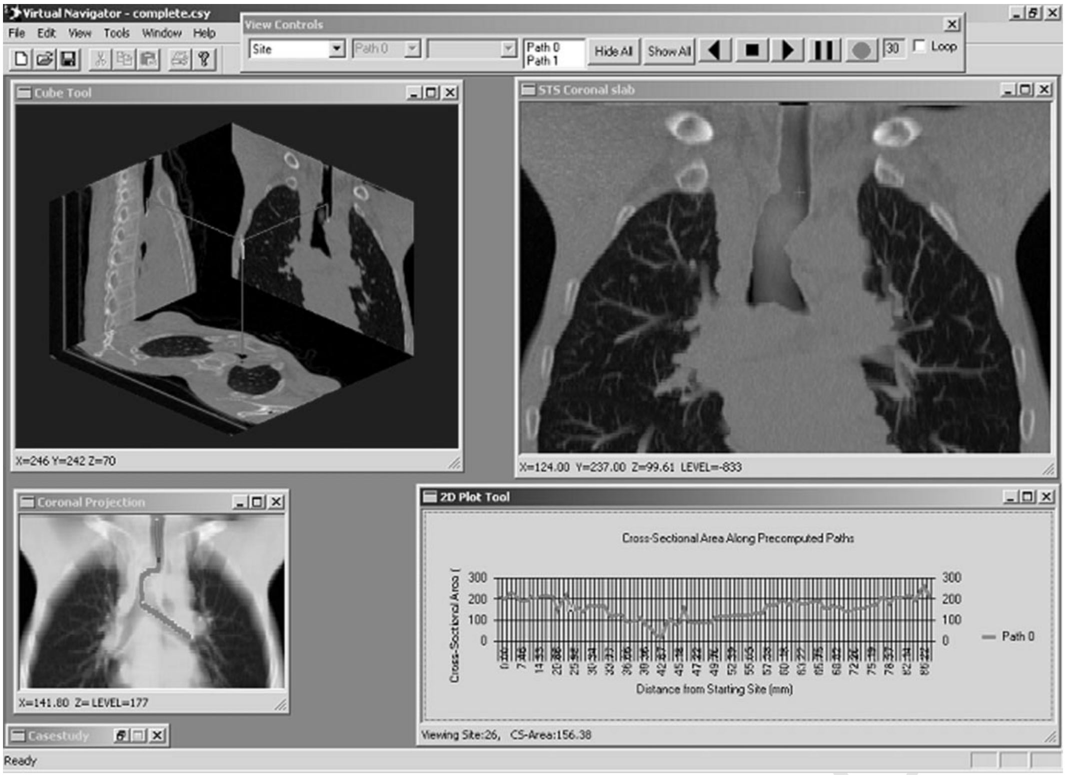
Virtual Navigator - complete.csy
File Edit View Tools Window Help
View Controls
Site
Path 0
Path 0
Path 1
Hide All
Show All
30
☐ Loop
Cube Tool
X=246 Y=242 Z=70
STS Coronal slab
X=124.00 Y=237.00 Z=99.61 LEVEL=-833
Coronal Projection
X=141.80 Z= LEVEL=177
2D Plot Tool
Cross-Sectional Area Along Precomputed Paths
Cross-Sectional Area (
300
200
100
0
Distance from Starting Site (mm)
Path 0
Viewing Site:26, CS-Area:156.38
Casestudy
Ready

Fig. 10. VB assessment of a human suffering from tracheomalacia. The input 3D CT image was $512 \times 512 \times 133$ ($\Delta x = \Delta y = 0.586$ mm; $\Delta z = 1.5$ mm). The proposed axis-generation algorithm was used to extract a path through the region of the collapse. This axis is depicted as a line on the Coronal Projection tool (maximum-intensity projection computed, data between $200 \leq y \leq 300$ used to create view). The computed airway cross-sectional values along this axis are depicted on the Plot Tool. The large dot on the Plot Tool represents a site selected by the user. All other activated tools (Coronal Projection, Cube Tool, and STS Coronal Slab) depict renditions of the 3D CT data at this selected site. The Cube Tool shows a composite of the standard three MPR (transverse, sagittal, and coronal) slices at this site; the pointers on these views indicate the direction of travel along the axis for this site. The STS Coronal Slab shows a coronal front-to-back depth-weighted maximum thin slab (slab thickness = 20, depth of vision = 30) at the site [42]. These views clearly show the manifestation of the collapse in several different ways.

required for effective VB-based evaluation of a 3D CT scan. We have applied the axes-generation algorithm proposed in this paper for guidance. In this section we give two examples.

Example #1: A patient suffering from tracheomalacia (collapsed trachea) underwent an EBCT scan. Using a single 20-sec breath hold, a 3D EBCT image made up of 133 contiguous slices was reconstructed. Each slice consists of 512×512 voxels (slice thickness $\Delta z = 1.5$ mm axial-plane resolution $\Delta x = \Delta y = 0.586$ mm). As discussed in Section 3.3 (step 1), we used the VB system to define a root site. An axis through the collapsed region was computed. This calculation required approximately 1 min of computation on a standard 700 MHz PC. This axis served as guidance data for interacting with the VB system.

Fig. 10 depicts a snapshot of the VB system. The solid line on the Coronal Projection image is the computed axis. Also, the Plot Tool shows a plot of cross-sectional area versus distance along the axis. One site near the ‘bend’ in the collapse was selected on the Plot Tool. This site is then highlighted on all other activated tools (Coronal Projection, Coronal Sliding Thin-Slab, and Cube Tools). The various views clearly show both visually and numerically the nature and extent of the collapse.

Example #2: A patient with a previously inserted stent, underwent an EBCT scan. As before, using a single 20-sec breath hold, a 3D EBCT scan was done. The reconstructed image consisted of 123 contiguous slices (512×512 voxels per slice, slice thickness $\Delta z = 1.5$ mm, axial-plane resolution $\Delta x = \Delta y = 0.586$ mm). We used the VB system to define a root site. A complete set of axes through the major airways was extracted. This analysis again took on the order of 1 min on a 700 MHz PC.

Fig. 11 depicts part of the analysis of this case. The Coronal Projection shows the extracted airway tree axes. The 3D Surface Tool also depicts the airway tree axes in addition to the segmented airway tree. The STS Coronal Slab and Cube Tool clearly show manifestations of the stent. The small cross-section views give shots at a particular site. Again, the system composite view provides many renditions of the region of interest.

5. Discussion

For typical high-resolution 3D CT chest images, the proposed axes-generation algorithm requires less than a

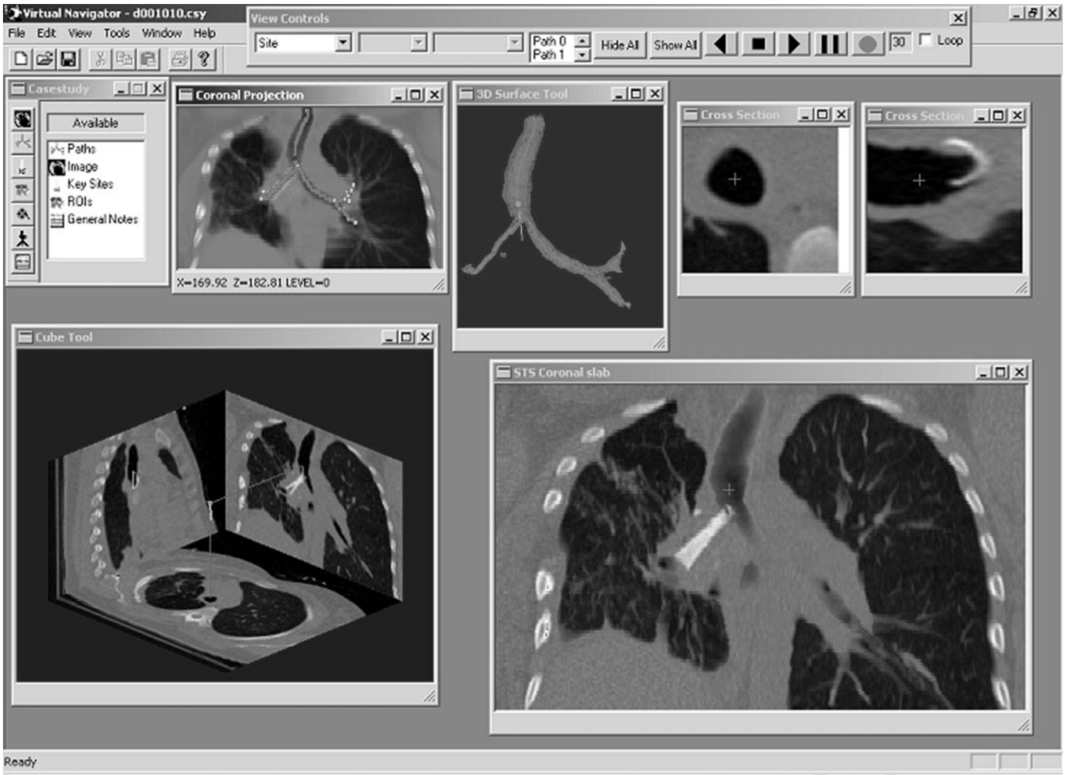

Fig. 11. VB assessment of a human having a misplaced stent encroaching stent. The input 3D CT image was 512 × 512 × 123 ($\Delta x = \Delta y = 0.586$ mm; $\Delta z = 1.5$ mm). Our axis-generation method was used to compute a complete set of axes through the major airways and a segmented airway tree. These axes are depicted on the Coronal Projection Tool (maximum-intensity projection computed, data between $200 \leq y \leq 300$ used to create view) and 3D Surface Tool. A site near the base of the stent by the main carina was selected on the 3D Surface Tool; this selected site is depicted as a large ball and needle in the 3D Surface Tool's view. All other activated tools (Coronal Projection, Cube Tool, STS Coronal Slab, and two Cross-Section Tools) depict renditions of the 3D CT data at this selected site. The Cube Tool shows a composite of the standard three MPR (transverse, sagittal, and coronal) slices at this site; the pointers on these views indicate the direction of travel along the axis for this site; the Coronal view clearly shows a longitudinal view of the encroaching stent. The STS Coronal Slab shows the geometry of the stent and surrounding structures at the site (front-to-back depth-weighted maximum, slab thickness = 20, depth of vision = 30). The Cross-Section Tool views show locally orthogonal views at the selected site. Finally, the 3D Surface Tool clearly shows the narrowness of the airway at the location of the stent.

few minutes of computation time on a standard PC. The algorithm uses the gray-scale data directly, and it does not require any interpolation or prior segmentation. With small modification, the algorithm could be applicable to other 3D tree-finding problems. Further work could be done in adapting the approach to find paths to preidentified suspect sites, such as airway narrowings and cancerous lymph nodes. Also, the method can have difficulty in cases where the airway wall becomes very thin, as a result of partial-voxel artifact. For such cases, the method can escape into the lungs and find many superfluous axes. Potential solutions to this problem are to incorporate iterative refinement and to have more robust wall-detection methods.

The method has been tested on phantom, animal, and human cases. This paper presented results for phantom and human cases. Refs. [38,43] give validation for animal cases. These results demonstrate the efficacy of the method. Further, the method has been integrated into Sun and PC-based software systems for virtual bronchoscopy [8,9,15,16,44]. Perhaps most significantly, we have successfully applied the method to live VB-based guidance of bronchoscopy for phantom and animal cases [15,16,44].

6. Summary

Major airway obstruction is a common problem arising from lung cancer, benign processes, and other tumors that commonly metastasize the airways. Three-dimensional (3D) computed-tomography (CT) pulmonary images are often used for evaluating such cases. The physician typically evaluates a case by using 3D mental reconstruction of anatomical structures depicted in the images. While often suitable for the radiologist, this form of evaluation is typically inadequate for the bronchoscopist, who need precise 3D 'road maps' to the surgical sites of interest.

Virtual bronchoscopy (VB) has emerged as a paradigm for more effective 3D CT image evaluation. Systematic evaluation of a 3D CT chest image using VB techniques, however, requires precomputed guidance data. This guidance data takes the form of central axes, or centerlines, through the major airways. We propose an axes-generation algorithm for VB assessment of 3D CT chest images. For a typical high-resolution 3D CT chest image, the algorithm produces a series of airway-tree axes, corresponding airway cross-sectional area measurements, and a segmented airway tree in a few minutes on a standard PC.

Our algorithm follows a two-stage approach. The first stage computes a *Discrete Model* that consists of a sparsely spaced set of data corresponding to the major airway axes. The second stage then uses the Discrete Model to define a smooth set of airway axes, a segmented airway tree, and other measurement data; these data constitute the *Generalized Cylinder* (*GC*) *Model*. The basic flow of the two-stage algorithm is as follows. The user first specifies a starting point of interest for the airway tree, generally in the proximal end of the trachea. Then, to perform the Stage-1 calculation of the Discrete Model, an adaptive 3D searching technique steps through the 3D image, building up a sparsely sampled set of central axes for the major airways. Stage-2 then applies a cubic-spline analysis and contour-finding procedure to the Discrete Model data to give a generalized cylinder representation for each airway central axis; this gives the GC Model.

Quantitative results for phantom and human airway-obstruction cases demonstrate the efficacy of the algorithm. Also, the algorithm is used in conjunction with a PC-based system for virtual-bronchoscopic (VB) assessment of high-resolution 3D CT chest images. In this context, the algorithm is demonstrated for two cases involving major airway obstructions.

Acknowledgements

This work was partially supported by grant #CA74325 from the National Cancer Institute of the NIH and by the Whitaker Foundation. The authors wish to thank Janice Cook-Granroth for her analysis. Early unrefered and incomplete versions of this work appeared in the conference proceedings [43,45].

References

[1] Naidich DP, Gruden JF, McGuinness G, McCauley DI, Bhalla MB. Volumetric CT (VCT) of the airways. J Thoracic Imag 1997;12: 11–28.

[2] Potchen EJ, Grainger RG, Greene R. Pulmonary Radiology. Philadelphia: W.B. Saunders, 1993.

[3] Arroliga AC, Matthay RA. The role of bronchoscopy in lung cancer. Clinics Med 1993;14:87–98.

[4] Vining DJ, Liu K, Choplin RH, Haponik EF. Virtual bronchoscopy: relationships of virtual reality endobronchial simulations to actual bronchoscopic findings. Chest 1996;109:549–53.

[5] Rubin GD, Beaulieu CF, Argiro V, Ringl H, Norbash AM, Feller JF, Dake MD, Jeffrey RB, Napel S. Perspective volume rendering of CT and MR images: applications for endoscopic imaging. Radiology 1996;199:321–30.

[6] Summers RM, Feng DH, Holland SM, Sneller MC, Shelhamer JH. Virtual bronchoscopy: segmentation method for real-time display. Radiology 1996;200:857–62.

[7] Summers RM. Navigational aids for real-time virtual bronchoscopy. Am J Roentgenol 1997;168:1165–70.

[8] Higgins WE, Ramaswamy K, Swift R, McLennan G, Hoffman EA. Virtual bronchoscopy for 3D pulmonary image assessment: state of the art and future needs. Radiographics 1998;18:761–78.

[9] Ramaswamy K, Higgins WE. Interactive dynamic navigation for virtual endoscopy. Comput Biol Med 1999;29:303–31.

[10] Haponik EF, Aquino SL, Vining DJ. Virtual bronchoscopy. Clinics Chest Med 1999;20:201–17.

[11] Aquino SL, Vining DJ. Virtual bronchoscopy. Clinics Chest Med 1999;20:725–30.

[12] Mori K, Hasegawa J, Suenaga Y, Toriwaki J. Automated anatomical labeling of the bronchial branch and its application to the virtual bronchoscopy system. IEEE Trans Med Imag 2000;19:103–14.

[13] Lorensen WE, Jolesz FA, Kikinis R. The exploration of cross-sectional data with a virtual endsocope. Interactive Technol New Health Paradigm 1995;Jan:221–30.

[14] Shahidi R, Argiro V, Napel S, Gray L, McAdams HP, Rubin GD, Beaulieu CF, Jeffrey RB, Johnson A. Assessment of several virtual endoscopy techniques using computed tomography and perspective volume rendering. Visualization in biomedical computing. Hohne KH, Kikinis R, editors. LNCS 1996;1131:512–28.

[15] Sherbondy AJ, Kiraly AP, Austin AL, Helferty JP, Wan S, Turlington JZ, Hoffman EA, McLennan G, Higgins WE. Virtual bronchoscopic approach for comining 3D CT and endoscopic video. Clough A, Chen CT, editors. SPIE Med Imag 2000: Physiol Funct Multidimension Images 2000;3978:104–16.

[16] Helferty JP, Sherbondy AJ, Hoffman EA, McLennan G, Higgins WE. Experiments in virtual-endoscopy guidance of bronchoscopy. In: Chen C, Clough AV, editors. SPIE Med Imag 2001: Physiol Funct Multidimension Images, 4321. 2001. p. 111–21.

[17] McLennan G, Shamsolkottabi S, Hoffman EA. Assessment of major airway obstruction using image analysis of digital CT information. SPIE Med Imag: Phys Funct Multidimension Images 1996;2709:197–208.

[18] Beutel J, Kundel HL, Van Metter RL. Handbook of Medical Imaging, vol. 1: physics and psychophysics. Bellingham, Washington: SPIE Press, 2000 Chapter 8.

[19] Higgins WE, Karwoski RA, Spyra WJT, Ritman EL. System for analyzing true three-dimensional angiograms 1996;15:377–85.

[20] Wood S, Zerhouni E, Hoford J, Hoffman E, Mitzner W. Measurement of three-dimensional lung tree structures by using computed tomography. J Appl Physiol 1995;79(5):1687–97.

[21] Hong L, Kaufman A, Wei Y, Viswambharan A, Wax M, Liang Z. 3D virtual colonoscopy. Proceedings of Biomed. Vis'95, 30 Oct 1995, p. 26–32 and 83.

[22] Ge Y, Stelts DR, Wang J, Vining DJ. Computing the centerline of a colon: a robust and efficient method based on 3D skeletons. J Comput Assist Tomog 1999;23(5):786–94.

[23] Sauret V, Goatman KA, Fleming JS, Bailey AG. Semi-automated tabulation of the 3D topology and morphology of branching networks using CT: application to the airway tree. Phys Med Biol 1999;44:1625–38.

[24] Pitsupati C. Geometric analysis of dynamic three-dimensional tree structures. PhD Thesis, Johns Hopkins University, 1996.

[25] Tozaki T, Kawata Y, Niki N, Ohmatsu H, Eguchi K, Moriyama N. Three-dimensional analysis of lung area using thin slice CT images 1996;2709:2–11.

[26] Haris K, Efstratiadis SN, Maglaveras N, Pappas C, Courassas J, Louridas G. Model-based morphologica segmentation and leabelling of coronary angiograms. IEEE Trans Med Imag 1999;18:1003–15.

[27] Klein AK, Lee F, Amini AA. Quantitative coronary angiography with deformable spline models. IEEE Trans Med Imag 1997;16:468–82.

[28] Sun Y. Automated identification of vessel contours in coronary arteriograms by an adaptive tracking algorithm 1989;8:78–88.

[29] Wang G, Vannier MW. GI tract unraveling by spiral CT 1995;2434:307–15.

[30] Williams J, Wolff L. Analysis of the pulmonary vascular tree using differential geometry based vector fields. Comput Vision Image Understanding 1997;65:226–36.

[31] Paik DS, Beaulieu CF, Jeffrey RB, Rubin GD, Napel S. Automated flight path planning for virtual endoscopy. Med Phys 1998;25: 629–37.

[32] Wink O, Niessen WJ, Viergever MA. Fast delineation and visualization of vessels in 3-D angiographic images. IEEE Trans Med Imag 2000;19:337–46.

[33] Krissian K, Malandain G, Ayache N. Model-based detection of tubular structures in 3D images. Comput Vision Image Understanding 2000;80:130–71.

[34] Verdonck B, Bloch I, Maitre H, Vandermeulen D, Suetens P, Marchal G. Accurate segmentation of blood vessels from 3D medical images. IEEE International Conference on Image Processing, 16–19 September 1996, p. III-311–4.

[35] Shani U, Ballard DH. Splines as embeddings for generalized cylinders. CVGIP 1984;27:129–56.

[36] Sequeira J, Ebel R, Schmitt F. Three-dimensional modeling of tree-like anatomical structures. Comput Med Imag Graph 1993;17:333–7.

[37] Kitamura K, Tobis JB, Sklansky J. Estimating the 3-D skeletons and transverse areas of coronary arteries from biplane angiograms 1988;7:173–87.

[38] Swift RD. Analysis and visualization of tubular structures in three-dimensional images. PhD Thesis, The Pennsylvania State University, May 1998.

[39] Paeth A, editor. Graphics gems, Vol. V. Boston: Academic Press, 1995.

[40] Rogers DF, Adams AJ. Mathematical elements for computer graphics. New York: McGraw-Hill, 1990.

[41] D'Souza ND, Reinhardt JM, Hoffman EA. ASAP: interactive quantification of 2D airway geometry. 10–15 February 1996; 2709: 180–96.

[42] Turlington JZ, Higgins WE. New techniques for efficient sliding thin-slab volume visualization. IEEE Trans Med Imag 2001;19(August).

[43] Swift RD, Higgins WE, Hoffman EA, McLennan G, Reinhardt JM. Automatic axis generation for 3D virtual bronchoscopic assessment. Hoffman E, editor. SPIE Conf Med Imag 1998: Image Process 1998;3337:73–84 February 21–27.

[44] Helferty JP, Zhang C, McLennan G, Higgins WE. Videoendoscopic distortion correction and its application to virtual guidance of endoscopy. IEEE Trans Med Imag 2001;19(July).

[45] McLennan G, Hoffman EA, Swift RD, Higgins WE. Virtual bronchoscopic assessment of major airway obstructions. Clough A, Chen CT, editors. SPIE Med Imag 1999: Physiol Funct Multidimension Images 1999;3660:313–20.

Roderick D. Swift received the BS degree in electrical engineering from Duke University and the MS and PhD degrees in electrical engineering from Penn State University. He has been employed at Mitre Corporation, Reston, VA, and IBM Corporation, Austin, Texas. His research interests are in medical image processing and visualization.

Atilla P. Kiraly received BA degrees in mathematics and physics from Hobart College. He has also received the MS degree in computer engineering from Penn State University. He has been employed previously at MicroSoft, Redmond WA, and Advanced Interface Technologies, State College, PA. He is currently a PhD candidate in computer engineering at Penn State University. His research interests are in medical image processing and virtual endoscopy.

Anthony J. Sherbondy received BS and MS degrees in electrical engineering from Penn State University. He is currently a research associate in radiology at Stanford University. His research interests are medical imaging and large-scale software-system design.

Allen L. Austin received BS and MS degrees in electrical engineering from Penn State University. He is currently employed at Advanced Interfaces Technologies, State College, PA. His areas of interest are image processing and virtual reality.

Eric A. Hoffman received BA degrees in physiology and psychology from Antioch College. He also received the PhD in physiology from the University of Minnesota. He has previously been an assistant professor of radiology at the Mayo Medical School, Rochester, MN, and an associate professor of radiology at the University of Pennsylvania, Philadelphia. He is currently a professor of radiology and biomedical engineering at the University of Iowa, where he serves as director of the Division of Physiologic Imaging. His research interests are in pulmonary imaging and medical image processing.

Geoffrey McLennan received the MBBS in medicine at the University of Adelaide, Australia. He has held various positions at the University of Adelaide and the University of Iowa. He is currently a professor of medicine in the University of Iowa College of Medicine. He also serves as Director of Bronchoscopy Service in the University of Iowa College of Medicine. His research interests are in pulmonary medicine and imaging.

William E. Higgins received the BS degree in electrical engineering from the Massachusetts Institute of Technology, Cambridge, MA, and the MS and PhD degrees in electrical engineering from the University of Illinois, Urbana-Champaign. He has held positions previously at the Honeywell Systems and Research Center, Minneapolis, MN, and the Mayo Clinic, Rochester, MN. He is currently a professor of electrical engineering, computer science and engineering, and bioengineering at the Pennsylvania State University. In addition he is an adjunct professor of radiology at the University of Iowa. His research interests are in multidimensional medical image processing and visualization and in virtual endoscopy.